Introduction

Thank you for purchasing the SMARTDAC+ GX10/GX20/GP10/GP20/GM10 Series (hereafter referred to as the recorder, GX, GP, or GM). This manual explains the dedicated commands for the recorder. To ensure correct use, please read this manual thoroughly before beginning operation. For details on the functions related to SMARTDAC+ series options, see also the manual for the options.

Notes

• The contents of this manual are subject to change without prior notice as a result of continuing improvements to the instrument’s performance and functions.
• Every effort has been made in the preparation of this manual to ensure the accuracy of its contents. However, should you have any questions or find any errors, please contact your nearest YOKOGAWA dealer.
• Copying or reproducing all or any part of the contents of this manual without the permission of YOKOGAWA is strictly prohibited.
• The TCP/IP software of this product and the documents concerning it have been developed/created by YOKOGAWA based on the BSD Networking Software, Release 1 that has been licensed from the Regents of the University of California.

Trademarks

• SMARTDAC+ and SMARTDACPLUS are registered trademarks of Yokogawa Electric Corporation.
• Microsoft and Windows are registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
• Adobe and Acrobat are registered trademarks or trademarks of Adobe Systems Incorporated.
• Kerberos is a trademark of Massachusetts Institute of Technology (MIT).
• Bluetooth is a trademark or registered trademark of Bluetooth SIG Inc.
• Company and product names that appear in this manual are registered trademarks or trademarks of their respective holders.
• The company and product names used in this manual are not accompanied by the registered trademark or trademark symbols (® and ™).

Revisions

December 2012 1st Edition
February 2013 2nd Edition
May 2013 3rd Edition
May 2014 4th Edition
December 2014 5th Edition
August 2015 6th Edition
December 2015 7th Edition
March 2016 8th Edition
June 2017 9th Edition
June 2018 10th Edition
July 2018 11th Edition
## Recorder Version and Functions Described in This Manual

<table>
<thead>
<tr>
<th>Edition</th>
<th>Product</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GX/GP: Version 1.01 and later</td>
<td>—</td>
</tr>
<tr>
<td>2</td>
<td>GX/GP: Version 1.02 and later</td>
<td>Feature additions.</td>
</tr>
<tr>
<td>3</td>
<td>GX/GP: Version 1.03 and later</td>
<td>Electromagnetic relay type analog input modules have been added. Feature additions.</td>
</tr>
<tr>
<td>4</td>
<td>GX/GP: Version 2.01 and later</td>
<td>Support for GX20/GP20 large memory type and expandable I/O has been added. Support for new modules (current (mA) input, low withstand voltage relay, and DI/DO) has been added. Feature additions. Advanced security function (/AS option) Custom display function (/CG option) EtherNet/IP communication (/E1 option) WT communication (/E2 option) Log scale function (/LG option) Etc.</td>
</tr>
<tr>
<td>5</td>
<td>GX/GP: Version 2.02 and later GM: Version 2.02 and later</td>
<td>Describes the GM. Feature additions. Bluetooth communication (/C8 option) [GM] USB communication [GM] Pulse input (DI module)</td>
</tr>
<tr>
<td>6</td>
<td>GX/GP: Version 2.02 and later GM: Version 2.03 and later</td>
<td>Advanced security function (/AS option) is added to the GM.</td>
</tr>
<tr>
<td>8</td>
<td>GX/GP: Version 3.02 and later GM: Version 3.02 and later</td>
<td>Port limitation setting of DARWIN compatible communication has been added.</td>
</tr>
<tr>
<td>9</td>
<td>GX/GP: Version 4.01 and later GM: Version 4.01 and later</td>
<td>Support for new modules (analog output, high-speed AI, 4-wire RTD, PID control) Feature additions. Program control (/PG option) Logic math function (/MT option) Support for new measurement modes (high-speed AI, dual interval)</td>
</tr>
<tr>
<td>10</td>
<td>GX/GP: Version 4.02 and later GM: Version 4.02 and later</td>
<td>Calibration correction of communication channel has been added.</td>
</tr>
<tr>
<td>11</td>
<td>GX/GP: Version 4.03 and later GM: Version 4.03 and later</td>
<td>Support for new modules (High withstand voltage AI).</td>
</tr>
</tbody>
</table>
How to Use This Manual

This manual explains the dedicated communication commands for the recorder and how to use them. For details on the features of the recorder and how to use it, see the following manuals.

- Model GX10/GX20/GP10/GP20 Paperless Recorder First Step Guide (IM 04L51B01-02EN)
- Model GX10/GX20/GP10/GP20 Paperless Recorder User’s Manual (IM 04L51B01-01EN)
- Data Acquisition System GM First Step Guide (IM 04L55B01-02EN)
- Data Acquisition System GM User’s Manual (IM 04L55B01-01EN)

Conventions Used in This Manual

<table>
<thead>
<tr>
<th>Unit K</th>
<th>Denotes 1024. Example: 768K (file size)</th>
</tr>
</thead>
<tbody>
<tr>
<td>k</td>
<td>Denotes 1000.</td>
</tr>
</tbody>
</table>

Markings

- Improper handling or use can lead to injury to the user or damage to the instrument. This symbol appears on the instrument to indicate that the user must refer to the user’s manual for special instructions. The same symbol appears in the corresponding place in the user’s manual to identify those instructions. In the manual, the symbol is used in conjunction with the word “WARNING” or “CAUTION.”

WARNING
Calls attention to actions or conditions that could cause serious or fatal injury to the user, and precautions that can be taken to prevent such occurrences.

CAUTION
Calls attention to actions or conditions that could cause light injury to the user or cause damage to the instrument or user’s data, and precautions that can be taken to prevent such occurrences.

Note
Calls attention to information that is important for the proper operation of the instrument.
Contents

Introduction ........................................................................................................................... i
Recorder Version and Functions Described in This Manual....................................................... ii
How to Use This Manual ........................................................................................................ iii
Conventions Used in This Manual ............................................................................................ iii

Chapter 1 Using Dedicated Commands (General)

1.1 Operations over an Ethernet Network ................................................................................. 1-1
  1.1.1 Preparing the Instrument ................................................................................................ 1-1
  1.1.2 Sending Commands and Receiving Responses ............................................................. 1-1

1.2 Operations over the Serial Interface (RS-232, RS-422/485, USB, Bluetooth) .................... 1-2
  1.2.1 Preparing the Instrument .............................................................................................. 1-2
  1.2.2 Sending Commands and Receiving Responses ............................................................. 1-2
  1.2.3 RS-232 Connection Procedure (GX/GP) ..................................................................... 1-3
  1.2.4 RS-422/485 Connection Procedure ............................................................................. 1-6
  1.2.5 USB Connection Procedure (GM) ............................................................................... 1-9
  1.2.6 Bluetooth Connection Procedure (GM, /C8 option) ..................................................... 1-10

Chapter 2 Commands and Responses

2.1 Command Transmission and Recorder Responses .............................................................. 2-1
  2.1.1 General Communication .............................................................................................. 2-1
  2.1.2 Command Types and Functions ................................................................................... 2-2
  2.1.3 Command Syntax ........................................................................................................ 2-2
  2.1.4 Recorder Responses .................................................................................................... 2-4

2.2 List of Commands .............................................................................................................. 2-5
  2.2.1 Setting Commands ...................................................................................................... 2-5
  2.2.2 Output Commands ....................................................................................................... 2-8
  2.2.3 Operation Commands .................................................................................................. 2-9
  2.2.4 Communication Control Commands ......................................................................... 2-9
  2.2.5 Instrument Information Commands .......................................................................... 2-10
  2.2.6 Conditions for Executing Commands ....................................................................... 2-10
  2.2.7 How to Use Commands .............................................................................................. 2-11
  2.2.8 Device Nomenclature in Command Descriptions ...................................................... 2-11

2.3 Parameters ....................................................................................................................... 2-12
  2.3.1 Measuring Range Parameters .................................................................................... 2-12
  2.3.2 Parameter Notation and Range .................................................................................. 2-12
  2.3.3 Specifying a Range ....................................................................................................... 2-13

2.4 Setting Commands ............................................................................................................ 2-14

2.5 Output Commands ............................................................................................................ 2-94

2.6 Operation Commands ....................................................................................................... 2-101

2.7 Communication Control Commands .................................................................................. 2-113

2.8 Instrument Information Output Commands ....................................................................... 2-115

2.9 Responses to Commands .................................................................................................. 2-117
  2.9.1 Affirmative Response (For commands other than output request commands) ............. 2-117
  2.9.2 Negative Response ..................................................................................................... 2-117
  2.9.3 Data Output Response ............................................................................................... 2-118
  2.9.4 Output in Response to RS-422/485 Commands .......................................................... 2-120

2.10 ASCII Output Format ....................................................................................................... 2-121
  2.10.1 Most Recent Channel Data (FData) .......................................................................... 2-121
  2.10.2 Most Recent (DO Channel) Status (FRelay) ............................................................... 2-122
  2.10.3 Internal Switch Status (FRelay) ................................................................................ 2-124
  2.10.4 Latest re-transmission output (AO channel) state (FTransStatAO) ............................. 2-124
  2.10.5 Users Who Are Currently Logged In (FUser) ............................................................ 2-126
  2.10.6 All Users Who Are Currently Logged In (FUser) ....................................................... 2-128
  2.10.7 Instrument Address (FAddr) ..................................................................................... 2-130
  2.10.8 Recorder status (FStat) ............................................................................................ 2-131
  2.10.9 Alarm Summary (FLog) ............................................................................................ 2-134
  2.10.10 Message Summary (FLog) ...................................................................................... 2-135
  2.10.11 Event log (FLog) .................................................................................................. 2-136
  2.10.12 Error Log (FLog) ................................................................................................. 2-136
  2.10.13 Address Setting Log (FLog) ..................................................................................... 2-137
  2.10.14 General Communication Log (FLog) ...................................................................... 2-138
Appendix

Appendix 1 ASCII Character Codes .............................................................. App-1
Appendix 2 Login Procedure .................................................................. App-2
When Using the Login Function .......................................................... App-2
When Not Using the Login Function ................................................... App-3
Appendix 3 Output Flow Chart of External Storage Medium Files and File Lists ........ App-4
Example for Outputting File aaaa.dtd ............................................... App-4
Example for Outputting a File List ................................................... App-5
Appendix 4 FIFO Data Output Flow Chart ........................................... App-6
Overview of the FIFO Buffer .......................................................... App-6
Example of FIFO Buffer Operation ................................................ App-6
Appendix 5 Check Sum Calculation Method ........................................ App-7
Appendix 6 Data Group Name and Data Name for WT Communication ........ App-8
Parameters p5 (Data Group Name) and p6 (Data Name) for SWattData Command App-8
Appendix 7 Bluetooth Communication Connection Flow Chart .............. App-14
1.1 Operations over an Ethernet Network

You can control the recorder by sending commands from a PC over an Ethernet network. There are various types of commands: setting commands, output commands, operation commands, communication control commands, and instrument information output commands.

1.1.1 Preparing the Instrument

Recorder Configuration
Configure the recorder to connect to the Ethernet network that you want to use. For instructions on how to configure the recorder, see section 1.17, “Configuring the Ethernet Communication Function” in the Model GX10/GX20/GP10/GP20 Paperless Recorder User’s Manual (IM 04L51B01-01EN) or section 2.18, “Configuring the Ethernet Communication Function,” in the Data Acquisition System GM User’s Manual (IM 04L55B01-01EN).

PC
The PC that you will use must meet the following requirements.
• The PC is connected to the Ethernet network that you want to use.
• The PC can run programs that you have created (see section 1.1.2, “Sending Commands and Receiving Responses,” below).

1.1.2 Sending Commands and Receiving Responses

Programs
When you send a command to the recorder, it will return a response. You can control the recorder by writing a program that sends commands and processes responses and then executing the program. You need to create the programs.
Example: If you send the command “FData,0,0001,0020” from your PC to the recorder, the recorder will return the most recent data of channels 0001 to 0020 in ASCII code.
For details on commands and responses, see chapter 2, “Commands and Responses.”

Notes on Creating Programs
• When Not Using the Login Function
  You can start using commands immediately after communication is established with the recorder.

• When Using the Login Function
  Log in to the recorder using a system administrator account or a normal user account that is registered in the recorder. Log in by connecting to the recorder and then sending the “CLogin” command.

• Port Number
  The default port number is “34434.” You can change the port number using the SServer command.
1.2 Operations over the Serial Interface (RS-232, RS-422/485, USB, Bluetooth)

You can control the recorder by sending commands from a PC through the serial interface. There are various types of commands: setting commands, output commands, operation commands, communication control commands, and instrument information output commands. Except for a few special commands, the commands are the same as those used over an Ethernet network.

1.2.1 Preparing the Instrument

Connection


Recorder Configuration

Configure the recorder to use serial communication. For instructions on how to configure the recorder, see section 1.18, “Configuring the Serial Communication Function (/C2 and /C3 options)” in the Model GX10/GX20/GP10/GP20 Paperless Recorder User’s Manual (IM 04L51B01-01EN) or section 2.19, “Configuring the Serial Communication Function (/C3 option),” section 2.20, “Configuring the USB Communication Function,” or section 2.21, “Configuring the Bluetooth Communication Functions,” in the Data Acquisition System GM User’s Manual (IM 04L55B01-01EN).

PC

The PC that you will use must meet the following requirements.

- The PC is connected to the recorder through the serial interface.
- The PC can run programs that you have created (see section 1.2.2, “Sending Commands and Receiving Responses,” below).

1.2.2 Sending Commands and Receiving Responses

Programs

When you send a command to the recorder, it will return a response. You can control the recorder by writing a program that sends commands and processes responses and then executing the program. You need to create the programs.

Example: If you send the command “FData,0,0001,0020” from your PC to the recorder, the recorder will return the most recent data of channels 0001 to 0020 in ASCII code.

For details on commands and responses, see chapter 2, “Commands and Responses.”

Notes on Creating Programs

- For RS-232 (GX/GP), USB communication (GM), Bluetooth (GM, /C8 option)

When you connect a PC to the recorder through the serial interface, the recorder will be ready to receive commands.

- For RS-422/485

The device that receives an open command (ESC O) from a PC will be ready to receive commands. The connection will close in the following situations.

- When the recorder receives a connection-close command (ESC C).
1.2.3 **RS-232 Connection Procedure (GX/GP)**  
Connect a cable to the 9-pin D-sub RS-232 connector.

**Connection**

- **Connector pin arrangement and signal names**

![Connector Diagram]

Each pin corresponds to the signal indicated below. The following table shows the signal name, RS-232 standard, JIS, and ITU-T standard signals.

<table>
<thead>
<tr>
<th>Pin</th>
<th>Signal Name</th>
<th>Name</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>RD</td>
<td>104</td>
<td>BB(RXD) Received data Input signal to the GX/GP.</td>
</tr>
<tr>
<td>3</td>
<td>SD</td>
<td>103</td>
<td>BA(TXD) Transmitted data Output signal from the GX/GP.</td>
</tr>
<tr>
<td>5</td>
<td>SG</td>
<td>102</td>
<td>AB(GND) Signal ground</td>
</tr>
<tr>
<td>7</td>
<td>RS</td>
<td>105</td>
<td>CA(RTS) Request to send Handshaking signal when receiving data from the PC. Output signal from the GX/GP.</td>
</tr>
<tr>
<td>8</td>
<td>CS</td>
<td>106</td>
<td>CB(CTS) Clear to send Handshaking signal when receiving data from the PC. Input signal to the GX/GP.</td>
</tr>
</tbody>
</table>

1. Pins 1, 4, 6, and 9 are not used.

**Signal direction**

- **Connection example**

- **OFF-OFF/XON-XON**

- **CS-RS(CTS-RTS)**

- **XON-RS(XON-RTS)**

The connection of RS on the PC and CS on the GX/GP is not necessary. However, we recommend that you wire them so that the cable can be used in either direction.
Handshaking

When using the RS-232 interface for transferring data, it is necessary for equipment on both sides to agree on a set of rules to ensure the proper transfer of data. The set of rules is called handshaking. Because there are various handshaking methods that can be used between the GX/GP and the PC, you must make sure that the same method is chosen by both the GX/GP and the PC.

You can choose any of the four methods on the GX/GP in the table below.

<table>
<thead>
<tr>
<th>Handshaking</th>
<th>Data transmission control (Control used when sending data to a PC)</th>
<th>Data Reception Control (Control used when receiving data from a PC)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Software Handshaking</td>
<td>Hardware Handshaking</td>
</tr>
<tr>
<td>OFF-OFF</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>XON-XON</td>
<td>Yes(^1)</td>
<td>Yes(^3)</td>
</tr>
<tr>
<td>XON-RS</td>
<td>Yes(^1)</td>
<td>Yes(^2)</td>
</tr>
<tr>
<td>CS-RS</td>
<td>Yes(^2)</td>
<td>Yes(^4)</td>
</tr>
</tbody>
</table>

Yes  Supported.

1  Stops transmission when X-OFF is received. Resume when X-ON is received.
2  Stops sending when CS (CTS) is false. Resumes when it is true.
3  Sends X-OFF when the receive data buffer is 3/4 full. Sends X-ON when the receive data buffer is 1/4 full.
4  Sets RS (RTS) to False when the receive data buffer is 3/4 full. Sets RS (RTS) to True when the receive data buffer becomes 1/4 full.

• OFF-OFF

**Data transmission control**

There is no handshaking between the GX/GP and the PC. The “X-OFF” and “X-ON” signals received from the PC are treated as data, and the CS signal is ignored.

**Data reception control**

There is no handshaking between the GX/GP and the PC. When the received buffer becomes full, all of the data that overflows are discarded.

RS  True (fixed).

• XON-XON

**Data transmission control**

Software handshaking is performed between the GX/GP and the PC. When an “X-OFF” code is received while sending data to the PC, the GX/GP stops the data transmission. When the GX/GP receives the next “X-ON” code, the GX/GP resumes the data transmission. The CS signal received from the PC is ignored.

**Data reception control**

Software handshaking is performed between the GX/GP and the PC. When the amount of used area in the received buffer reaches to 3/4 full (192 bytes for R2.01 and earlier; 6144 bytes for R2.02 and later), the GX/GP sends an “X-OFF” code. Then, when the amount of used area decreases to 1/4 bytes (64 bytes for R2.01 and earlier; 2048 bytes for R2.02 and later), the GX/GP sends an “X-ON” code.

RS  True (fixed).

• XON-RS

**Data transmission control**

The operation is the same as with XON-XON.

**Data reception control**

Hardware handshaking is performed between the GX/GP and the PC. When the amount of used area in the received buffer reaches to 3/4 full (192 bytes for R2.01 and earlier; 6144 bytes for R2.02 and later), the GX/GP sets “RS=False.” Then, when the amount of used area decreases to 1/4 bytes (64 bytes for R2.01 and earlier; 2048 bytes for R2.02 and later), the GX/GP sets “RS=True.”
• CS-RS

Data transmission control
Hardware handshaking is performed between the GX/GP and the PC. When the CS signal becomes False while sending data to the PC, the GX/GP stops the data transmission. When the CS signal becomes True, the GX/GP resumes the data transmission. The “X-OFF” and “X-ON” signals are treated as data.

Data reception control
The operation is the same as with XON-RS.

Note

• The PC program must be designed so that the received buffers of both the GX/GP and the PC do not become full.
• If you select XON-XON, send the data in ASCII format.
1.2.4 RS-422/485 Connection Procedure

Connect a cable to the terminal.

Connection

• Connecting the Cable

As shown in the figure below, remove approximately 6 mm of the covering from the end of the cable to expose the conductor. Keep the exposed section from the end of the shield within 5 cm.

![Connection Diagram](image)

<table>
<thead>
<tr>
<th>Two-wire system</th>
<th>Four-wire system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electric potential of the shield</td>
<td>Electric potential of the shield</td>
</tr>
<tr>
<td>Shield</td>
<td>Shield</td>
</tr>
<tr>
<td>FG</td>
<td>SDB+</td>
</tr>
<tr>
<td>SG</td>
<td>SDA−</td>
</tr>
<tr>
<td>RDB+</td>
<td>RDA−</td>
</tr>
</tbody>
</table>

Recommended torque for tightening the screw: 0.2 N·m

• Signal names

Each terminal corresponds to the signal indicated below.

<table>
<thead>
<tr>
<th>Signal Name</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>FG</td>
<td>Frame ground of the recorder.</td>
</tr>
<tr>
<td>SG</td>
<td>Signal ground.</td>
</tr>
<tr>
<td>SDB+</td>
<td>Send data B (+).</td>
</tr>
<tr>
<td>SDA−</td>
<td>Send data A (−).</td>
</tr>
<tr>
<td>RDB+</td>
<td>Receive data B (+).</td>
</tr>
<tr>
<td>RDA−</td>
<td>Receive data A (−).</td>
</tr>
</tbody>
</table>

Connecting to the host device

The figure below illustrates the connection of the recorder to a host device. If the port on the host device is an RS-232 interface, connect a converter.

![Connection Diagram](image)

Connection example to the host device

A connection can be made with a host device having a RS-232, RS422, or RS-485 port. In the case of RS-232, a converter is used. See the connection examples below for a typical converter terminal. For details, see the manual that comes with the converter.

<table>
<thead>
<tr>
<th>RS-422/485 Port</th>
<th>Converter</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDA(−)</td>
<td>TD(−)</td>
</tr>
<tr>
<td>SDB(+)</td>
<td>TD(+)</td>
</tr>
<tr>
<td>RDA(−)</td>
<td>RD(−)</td>
</tr>
<tr>
<td>RDB(+)</td>
<td>RD(+)</td>
</tr>
<tr>
<td>SG</td>
<td>SHIELD</td>
</tr>
<tr>
<td>FG</td>
<td>EARTH</td>
</tr>
</tbody>
</table>

There is no problem of connecting a 220-Ω terminator at either end if YOKOGAWA’s PLCs or temperature controllers are also connected to the communication line.
• Four-wire system
Generally, a four-wire system is used to connect to a host device. In the case of a four-wire system, the transmission and reception lines need to be crossed over.

- **RS-422/485 terminal on the recorder**
  - Termination (external) 120 Ω 1/2W or greater

- **Do not connect terminators to #1 through #n-1.**

• Two-wire system
Connect the transmission and reception signals with the same polarity on the RS-422/485 terminal block. Only two wires are used to connect to the external device.

- **RS-422/485 terminal on the recorder**
  - Termination (external) 120 Ω 1/2W or greater

- **Do not connect terminators to #1 through #n-1.**

**Note**
- The method used to eliminate noise varies depending on the situation. In the connection example, the shield of the cable is connected only to the recorder’s ground (one-sided grounding). This is effective when there is a difference in the electric potential between the computer’s ground and the recorder’s ground. This may be the case for long distance communications. If there is no difference in the electric potential between the computer’s ground and the recorder’s ground, the method of connecting the shield also to the computer’s ground may be effective (two-sided grounding). In addition, in some cases, using two-sided grounding with a capacitor connected in series on one side is effective. Consider these possibilities to eliminate noise.
- When using the two-wire interface (Modbus protocol), the 485 driver must be set to high impedance within 3.5 characters after the last data byte is sent by the host computer.
1.2  Operations over the Serial Interface (RS-232, RS-422/485, USB, Bluetooth)

Serial interface converter
The recommended converter is given below.
SYSMEX RA CO., LTD./MODEL RC-770X, LINE EYE/SI-30FA, YOKOGAWA/ML2

Some converters not recommended by Yokogawa have FG and SG pins that are not isolated. In this case, do not follow the diagram on the previous page (do not connect anything to the FG and SG pins). Especially in the case of long distance communications, the potential difference that appears may damage the recorder or cause communication errors. For converters that do not have the SG pin, they can be used without using the signal ground. For details, see the manual that comes with the converter.

On some non-recommended converters, the signal polarity may be reversed (A/B or +/- indication). In this case, reverse the connection.

For a two-wire system, the host device must control the transmission driver of the converter in order to prevent collisions of transmit and received data. When using the recommended converter, the driver is controlled using the RS (RTS) signal on the RS-232.

When instruments that support only the RS-422 interface exist in the system
When using the four-wire system, up to 32 recorders can be connected to a single host device. However, this may not be true if instruments that support only the RS-422 interface exist in the system.

When YOKOGAWA's recorders that support only the RS-422 interface exist in the system
The maximum number of connection is 16. Some of YOKOGAWA's conventional recorders (HR2400 and µR, for example) only support the RS-422 driver. In this case, only up to 16 units can be connected.

Note
In the RS-422 standard, 10 is the maximum number of connections that are allowed on one port (for a four-wire system).

Terminator
When using a multidrop connection (including a point-to-point connection), connect a terminator to the recorder if the recorder is connected to the end of the chain. Do not connect a terminator to a recorder in the middle of the chain. In addition, turn ON the terminator on the host device (see the manual of the host device). If a converter is being used, turn ON its terminator. The recommended converter is a type that has a built-in terminator.
Select the appropriate terminator (120 Ω), indicated in the figure, according to the characteristic impedance of the line, the installation conditions of the instruments, and so on.
1.2.5 **USB Connection Procedure (GM)**

The procedure to connect a GM to the PC via USB is shown below. For instructions on how to use the PC, see the user’s manual for your PC.

**Configuring the GM**

Turn the USB communication function on (default value is on). For the procedure, see section 2.19, “Configuring the USB Communication Function,” in the Data Acquisition System GM User’s Manual (IM 04L55B01-01EN).

**Connecting the GM to the PC**

Connect a USB cable to the USB port.

USB port (USB mini B type)

If the PC is connected to a network environment, a USB driver will be automatically installed. If it does not, check the download link for the driver at our website below, and install the driver.


When the USB driver installation is complete, a COM port will be assigned.

Connect using the following communication conditions.

- Baud rate: 115200
- Parity: None
- Data length: 8 bits
- Stop bits: 1 bit
- Handshake: Off: Off
1.2.6 Bluetooth Connection Procedure (GM, /C8 option)

The procedure to connect a GM to the PC via Bluetooth is shown below. For instructions on how to use the PC, see the user’s manual for your PC.

Configuring the GM

Turn the Bluetooth function on (default value is on). For the procedure, see section 2.20, “Configuring the Bluetooth Communication Function,” in the Data Acquisition System GM User’s Manual (IM 04L55B01-01EN).

Connecting the GM to the PC

1 Check whether the BT LED in the GM status display area is on. If the LED is off, hold down the GM USER1 key for at least 3 seconds.

The BT LED in the GM status display area turns on, the GM enters the connection standby state.

2 Perform a pairing operation from the PC.

A 6-digit authentication code appears on the GM’s 7 segment LED. Check that this authentication code matches that shown on the PC, and pair the devices. When pairing is complete, a COM port will be assigned.

Note

The GM stores up to eight entries of pairing information. This information is retained even when the power is turned off.

The pairing operation is not necessary in subsequent connections.

3 Perform the operation for connecting from the PC to the GM.

See “Appendix 7 Bluetooth Communication Connection Flow Chart” and section “2.2.7 How to Use Commands”.

---

[Image of BT LED on GM status display area]
Chapter 2 Commands and Responses

2.1 Command Transmission and Recorder Responses

2.1.1 General Communication
The recorder can work with various applications through the use of commands. The communication that is achieved through commands is referred to as “general communication.”

2.1.2 Command Types and Functions
The following types of commands are available. The first character of command names represents the command type. For example, in the command “SRangeAI,” “S” represents the command type. The second and subsequent characters represent the contents of commands.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation commands</td>
<td>Commands that start with “O.” These commands are used to operate the recorder.</td>
</tr>
<tr>
<td>Example: OSetTime</td>
<td></td>
</tr>
<tr>
<td>Setting commands</td>
<td>Commands that start with “S.” These commands change the recorder settings.</td>
</tr>
<tr>
<td>Example: SRangeAI</td>
<td></td>
</tr>
<tr>
<td>Output commands</td>
<td>Commands that start with “F.” These commands cause the recorder to output measured data and other types of data.</td>
</tr>
<tr>
<td>Example: FData</td>
<td></td>
</tr>
<tr>
<td>Communication Control commands</td>
<td>Commands that start with “C.” These commands control the communication with the recorder.</td>
</tr>
<tr>
<td>Example: CCheckSum</td>
<td></td>
</tr>
<tr>
<td>Instrument information output commands</td>
<td>Commands that start with an underscore. These commands cause the recorder to output its instrument information.</td>
</tr>
<tr>
<td>Example: _MFG</td>
<td></td>
</tr>
</tbody>
</table>

2.1.3 Command Syntax
A Single Command
A single command consists of a command name, parameters, delimiters, and terminator. The command name is written in the beginning, and parameters follow. Delimiters are used to separate the command name from parameters and between each parameter. A delimiter is a symbol that indicates a separation. A terminator is attached to the end of a command.

Command name, parameter 1, parameter 2 terminator

Example of a Command
SRangeAI,0001,VOLT,2V,OFF,-15000,18000,0

Commands in a Series (Setting commands only)
You can send multiple setting commands in a series. When writing a series of commands, separate each command with a sub delimiter. A sub delimiter is a symbol that indicates a separation. A terminator is attached to the end of the series. The maximum number of bytes that can be sent at once is 8000 bytes (8000 characters).

Command name, parameter 1, parameter 2; command name, parameter 1 terminator

(Command 1) (Command 2)

Sub delimiter
2.1 Command Transmission and Recorder Responses

**Notes on Writing Commands in a Series**
- Only setting commands can be written in a series.
- Queries (see the next section) cannot be written in a series.
- If there is an error in one of the commands in a series, the commands before it are canceled, and those after it are not executed.

**Example of a Command**

```
SRangeAI,0001,VOLT,2V,OFF,-15000,18000,0;SRangeAI,0002,SKIP
```

**Queries**

Queries are used to inquire the recorder settings. To send a query, append a question mark to the command name or parameter. When the recorder receives a query, it returns the relevant setting as a character string in an appropriate syntax. Queries can be used on some of the available setting and operation commands.

**Command name? terminator**

**Command name,parameter1? terminator**

**Examples of Queries and Responses**

<table>
<thead>
<tr>
<th>Query</th>
<th>Example of Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRangeAI?</td>
<td>SRangeAI,0001,VOLT,2V,OFF,-20000,20000,0</td>
</tr>
<tr>
<td></td>
<td>SRangeAI,0002,..........................</td>
</tr>
<tr>
<td></td>
<td>...........................................</td>
</tr>
<tr>
<td>SRangeAI,0001?</td>
<td>SRangeAI,0001,VOLT,2V,OFF,-20000,20000,0</td>
</tr>
</tbody>
</table>

**Command Names**

A command name is a character string consisting of up to 16 alphanumeric characters. The first character represents the command type.

**Notes on Writing Commands Names**
- Command names are not case sensitive.
- Spaces before the character string are ignored.
Parameters

Parameters are characteristic values that are attached to commands.

Notes on Writing Parameters

• Write parameters in their appropriate order.
• Spaces around and in the middle of parameters are ignored. Exception is the character strings that users specify.
• You can omit the setting command parameters that do not need to be changed from their current settings. If you omit parameters, write only the delimiters.
• If parameters are omitted and there are multiple delimiters at the end of the command, those delimiters can be omitted.
  Example: SRangeAI,0001,,1800,0 terminator
  SRangeAI,0001,VOLT,2V,terminator

There are two types of parameters: predefined expressions and user-defined character strings.

How to Write User-Defined Character Strings (Parameters)

• Enclose user-defined character strings in single quotation marks.
  Example The command for setting the channel 0001 tag to “SYSTEM1” is shown below.
  STagIO,0001,'SYSTEM1'
• There are two types of user-defined character strings depending on the type of characters that can be used.

  Character Strings Consisting Only of Characters in the ASCII Code Range (0x00 to 0x7f)
  In this manual, applicable parameters are indicated with “ASCII.”
  Example p3 Tag number (up to 16 characters, ASCII)

  You can use alphanumeric characters and some of the symbols. For the ASCII characters that you can use, see appendix 1.

  Character Strings Consisting of Characters in the UTF-8 Code Range
  In this manual, applicable parameters are indicated with “UTF-8.”
  Example p2 Tag (up to 32 characters, UTF-8)

  UTF-8 codes include ASCII codes. You can use UTF-8 characters, including the ASCII characters above. For the ASCII characters that you can use, see appendix 1.

Delimiters

Commas are used as delimiters.

Sub delimiters

Semicolons are used as sub delimiters.

Terminators

“CR+LF” is used as a terminator, meaning “CR” followed by “LF.” Expressed in ASCII code, it is 0x0d0x0a.
2.1.4 Recorder Responses

The recorder returns the following responses to commands.

- If the recorder successfully completes the processing of a received output request command, it outputs the requested data.
- If the recorder successfully completes the processing of a received command that is not an output request command, it outputs an affirmative response.
- If a command syntax error, setting error, or other error occurs, the recorder outputs a negative response.

For each command the recorder receives, it returns a single response. The controller (PC) side must process commands and responses in accordance with this command-response rule. If the command-response rule is not followed, the operation of the recorder is not guaranteed. For details on the response syntax, see 2.9 Responses to Commands.
2.2 List of Commands

Unless specified otherwise, AI, AO, DI, DO, PI, and PID represent I/O channel types.
- AI  Analog input
- AO  Analog output
- DI  Digital input
- DO  Digital output
- PI  Pulse input
- PID PID control

### 2.2.1 Setting Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description (Required Options) [Applicable Models]</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Measurement Operation Setting Commands</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SScan</td>
<td>Scan interval</td>
<td>2-14</td>
</tr>
<tr>
<td>SScanGroup</td>
<td>Scan group (module scan interval)</td>
<td>2-14</td>
</tr>
<tr>
<td>SModeAI</td>
<td>AI module</td>
<td>2-15</td>
</tr>
<tr>
<td>SModeAICurrent</td>
<td>Current input type AI module</td>
<td>2-15</td>
</tr>
<tr>
<td>SBOLmtAI</td>
<td>Upper and lower burnout limits of AI module</td>
<td>2-16</td>
</tr>
<tr>
<td>SBOLmtAICurrent</td>
<td>Upper and lower burnout limits of current input type AI module</td>
<td>2-16</td>
</tr>
<tr>
<td>SModeDI</td>
<td>DI module</td>
<td>2-16</td>
</tr>
<tr>
<td>SModePID</td>
<td>PID control module</td>
<td>2-16</td>
</tr>
<tr>
<td>SScaleOver</td>
<td>Detection of values that exceed the scale</td>
<td>2-17</td>
</tr>
<tr>
<td>Recording Basic Setting Commands</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMemory</td>
<td>Recording mode</td>
<td>2-17</td>
</tr>
<tr>
<td>SMemKeyConfirm</td>
<td>Record confirmation action [GX/GP]</td>
<td>2-17</td>
</tr>
<tr>
<td>SDispData</td>
<td>Display data recording</td>
<td>2-17</td>
</tr>
<tr>
<td>SEventData</td>
<td>Event data recording</td>
<td>2-18</td>
</tr>
<tr>
<td>Recording Channel Setting Commands</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRecDisp</td>
<td>Channel for recording display data</td>
<td>2-19</td>
</tr>
<tr>
<td>SRecEvent</td>
<td>Channel for recording event data</td>
<td>2-19</td>
</tr>
<tr>
<td>SRecManual</td>
<td>Channel for recording manual sampled data</td>
<td>2-20</td>
</tr>
<tr>
<td>Batch Setting Commands</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SBatch</td>
<td>Batch function</td>
<td>2-20</td>
</tr>
<tr>
<td>STextField</td>
<td>Batch text</td>
<td>2-20</td>
</tr>
<tr>
<td>Data Save Setting Commands</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDirectory</td>
<td>Name of directory to save data</td>
<td>2-20</td>
</tr>
<tr>
<td>SFileHead</td>
<td>File header</td>
<td>2-20</td>
</tr>
<tr>
<td>SFileName</td>
<td>File naming rule</td>
<td>2-21</td>
</tr>
<tr>
<td>SMediaSave</td>
<td>Automatic data file saving</td>
<td>2-21</td>
</tr>
<tr>
<td>SFileFormat</td>
<td>Display/event data file format</td>
<td>2-21</td>
</tr>
<tr>
<td>I/O Channel (AI/AO/DI/DO/PI/PID) Setting Commands</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRangeAI</td>
<td>Measurement range of AI channel</td>
<td>2-22</td>
</tr>
<tr>
<td>SRangeAO</td>
<td>Measurement range of pulse input type AI channel</td>
<td>2-23</td>
</tr>
<tr>
<td>SRangePulse</td>
<td>Measurement range of pulse input channel</td>
<td>2-24</td>
</tr>
<tr>
<td>SRangeDI</td>
<td>Measurement range of DI channel</td>
<td>2-25</td>
</tr>
<tr>
<td>SRangeDO</td>
<td>DO channel operation</td>
<td>2-25</td>
</tr>
<tr>
<td>SRangeAO</td>
<td>AO channel operation</td>
<td>2-26</td>
</tr>
<tr>
<td>SMoveAve</td>
<td>Moving average</td>
<td>2-26</td>
</tr>
<tr>
<td>SFilter</td>
<td>First-order lag filter</td>
<td>2-27</td>
</tr>
<tr>
<td>SBurnOut</td>
<td>Behavior when a sensor burns out</td>
<td>2-27</td>
</tr>
<tr>
<td>SRjc</td>
<td>Reference junction compensation method</td>
<td>2-27</td>
</tr>
<tr>
<td>SAAlarm</td>
<td>Alarm</td>
<td>2-28</td>
</tr>
<tr>
<td>SAAlarmHyst</td>
<td>Alarm hysteresis</td>
<td>2-28</td>
</tr>
<tr>
<td>SAAlarmDelay</td>
<td>Alarm delay time</td>
<td>2-29</td>
</tr>
<tr>
<td>STag</td>
<td>Tag</td>
<td>2-29</td>
</tr>
<tr>
<td>SColor</td>
<td>Channel color</td>
<td>2-29</td>
</tr>
<tr>
<td>SZone</td>
<td>Waveform display zone</td>
<td>2-30</td>
</tr>
<tr>
<td>SScale</td>
<td>Scale display [GX/GP]</td>
<td>2-30</td>
</tr>
<tr>
<td>SBar</td>
<td>Bar graph display</td>
<td>2-30</td>
</tr>
<tr>
<td>SPartial</td>
<td>Partial expanded display [GX/GP]</td>
<td>2-30</td>
</tr>
<tr>
<td>SColorS</td>
<td>Color scale band</td>
<td>2-30</td>
</tr>
<tr>
<td>SAlarmMark</td>
<td>Alarm mark</td>
<td>2-31</td>
</tr>
<tr>
<td>SValue</td>
<td>Upper/lower limit display characters</td>
<td>2-31</td>
</tr>
<tr>
<td>SCaliB</td>
<td>Calibration correction</td>
<td>2-32</td>
</tr>
<tr>
<td>SPreset</td>
<td>AO channel preset action</td>
<td>2-33</td>
</tr>
<tr>
<td>Math Channel Setting Commands</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMathBasic</td>
<td>Math action (/MT)</td>
<td>2-33</td>
</tr>
<tr>
<td>SKConst</td>
<td>Constant (/MT)</td>
<td>2-34</td>
</tr>
<tr>
<td>SRangeMath</td>
<td>Computation expression (/MT)</td>
<td>2-34</td>
</tr>
<tr>
<td>STlog</td>
<td>TLOG (/MT)</td>
<td>2-34</td>
</tr>
<tr>
<td>SRolAve</td>
<td>Rolling average (/MT)</td>
<td>2-34</td>
</tr>
<tr>
<td>SAAlarm</td>
<td>Alarm (/MT)</td>
<td>2-35</td>
</tr>
<tr>
<td>SAlarmHyst</td>
<td>Alarm hysteresis (/MT)</td>
<td>2-35</td>
</tr>
<tr>
<td>SAlarmDelay</td>
<td>Alarm delay time (/MT)</td>
<td>2-35</td>
</tr>
<tr>
<td>STag</td>
<td>Tag (/MT)</td>
<td>2-36</td>
</tr>
<tr>
<td>SColor</td>
<td>Channel color (/MT)</td>
<td>2-36</td>
</tr>
<tr>
<td>SZone</td>
<td>Waveform display zone (/MT)</td>
<td>2-36</td>
</tr>
<tr>
<td>SScale</td>
<td>Scale display (/MT) [GX/GP]</td>
<td>2-36</td>
</tr>
<tr>
<td>SBar</td>
<td>Bar graph display (/MT)</td>
<td>2-36</td>
</tr>
<tr>
<td>SPartial</td>
<td>Partial expanded display (/MT) [GX/GP]</td>
<td>2-36</td>
</tr>
<tr>
<td>SColor</td>
<td>Color scale band (/MT)</td>
<td>2-37</td>
</tr>
<tr>
<td>SAlarmMark</td>
<td>Alarm mark (/MT)</td>
<td>2-37</td>
</tr>
<tr>
<td>Communication Channel Setting Commands</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRangeCom</td>
<td>Measurement range (/MC)</td>
<td>2-38</td>
</tr>
<tr>
<td>SValueCom</td>
<td>Preset operation (/MC)</td>
<td>2-38</td>
</tr>
<tr>
<td>SWDCom</td>
<td>Watchdog timer (/MC)</td>
<td>2-38</td>
</tr>
<tr>
<td>SAlarmCom</td>
<td>Alarm (/MC)</td>
<td>2-38</td>
</tr>
<tr>
<td>SAlarmHyst</td>
<td>Alarm hysteresis (/MC)</td>
<td>2-39</td>
</tr>
<tr>
<td>SAlarmDelay</td>
<td>Alarm delay time (/MC)</td>
<td>2-39</td>
</tr>
<tr>
<td>STag</td>
<td>Tag (/MC)</td>
<td>2-39</td>
</tr>
<tr>
<td>SColor</td>
<td>Channel color (/MC)</td>
<td>2-39</td>
</tr>
</tbody>
</table>
### 2.2 List of Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SZoneCom</td>
<td>Waveform display zone (/MC)</td>
<td>2-39</td>
</tr>
<tr>
<td>SScaleCom</td>
<td>Scale display (/MC) [GX/GP]</td>
<td>2-39</td>
</tr>
<tr>
<td>SBarCom</td>
<td>Bar graph display (/MC)</td>
<td>2-40</td>
</tr>
<tr>
<td>SPartialCom</td>
<td>Partial expanded display (/MC)[GX/GP]</td>
<td>2-40</td>
</tr>
<tr>
<td>SBandCom</td>
<td>Color scale band (/MC)</td>
<td>2-40</td>
</tr>
<tr>
<td>SAlmMarkCom</td>
<td>Alarm mark (/MC)</td>
<td>2-40</td>
</tr>
<tr>
<td>SCALibUseCom</td>
<td>Calibration correction use On/Off</td>
<td>2-41</td>
</tr>
<tr>
<td>SCALibCom</td>
<td>Calibration correction</td>
<td>2-41</td>
</tr>
<tr>
<td>SAlmLimit</td>
<td>Rate-of-change alarm interval</td>
<td>2-42</td>
</tr>
<tr>
<td>SIndivAlmACK</td>
<td>Individual alarm ACK</td>
<td>2-42</td>
</tr>
<tr>
<td>SAlmSts</td>
<td>Alarm display hold/nonhold</td>
<td>2-42</td>
</tr>
<tr>
<td>STimer</td>
<td>Timer</td>
<td>2-42</td>
</tr>
<tr>
<td>SMatchTimer</td>
<td>Match time timer</td>
<td>2-43</td>
</tr>
<tr>
<td>SEventAct</td>
<td>Event action</td>
<td>2-43</td>
</tr>
<tr>
<td>SReport</td>
<td>Report type (/MT)</td>
<td>2-45</td>
</tr>
<tr>
<td>SRepData</td>
<td>Report data (/MT)</td>
<td>2-46</td>
</tr>
<tr>
<td>SRepTemp</td>
<td>Report output (/MT)</td>
<td>2-46</td>
</tr>
<tr>
<td>SdigitalSign</td>
<td>Electronic signature inclusion (/MT)</td>
<td>2-46</td>
</tr>
<tr>
<td>SRepCh</td>
<td>Report channel (/MT)</td>
<td>2-46</td>
</tr>
<tr>
<td>SRepBatchInfo</td>
<td>Batch information output (/MT)</td>
<td>2-47</td>
</tr>
<tr>
<td>SLcd</td>
<td>LCD [GX/GP]</td>
<td>2-47</td>
</tr>
<tr>
<td>SViewAngle</td>
<td>View angle [GX/GP]</td>
<td>2-47</td>
</tr>
<tr>
<td>SBackColor</td>
<td>Screen background color [GX/GP]</td>
<td>2-47</td>
</tr>
<tr>
<td>SGrpChange</td>
<td>Automatic group switching time [GX/GP]</td>
<td>2-47</td>
</tr>
<tr>
<td>SAutoJump</td>
<td>Jump default display operation [GX/GP]</td>
<td>2-47</td>
</tr>
<tr>
<td>SCalFormat</td>
<td>Calendar display format [GX/GP]</td>
<td>2-48</td>
</tr>
<tr>
<td>SBarDirect</td>
<td>Bar graph display direction [GX/GP]</td>
<td>2-48</td>
</tr>
<tr>
<td>SChgMonitor</td>
<td>Value modification from the monitor</td>
<td>2-48</td>
</tr>
<tr>
<td>STrdWave</td>
<td>Trend waveform display [GX/GP]</td>
<td>2-48</td>
</tr>
<tr>
<td>STrdScale</td>
<td>Scale [GX/GP]</td>
<td>2-48</td>
</tr>
<tr>
<td>STrdLine</td>
<td>Trend line width, grid [GX/GP]</td>
<td>2-48</td>
</tr>
<tr>
<td>STrdRate</td>
<td>Trend interval switching [GX/GP]</td>
<td>2-49</td>
</tr>
<tr>
<td>STrdKind</td>
<td>Trend type [GX/GP]</td>
<td>2-49</td>
</tr>
<tr>
<td>STrdPartial</td>
<td>Partial expanded trend display [GX/GP]</td>
<td>2-49</td>
</tr>
<tr>
<td>SMsgBasic</td>
<td>Message writing</td>
<td>2-49</td>
</tr>
<tr>
<td>SGroup</td>
<td>Display group</td>
<td>2-49</td>
</tr>
<tr>
<td>STrripLine</td>
<td>Display group trip line</td>
<td>2-50</td>
</tr>
<tr>
<td>SScilamp</td>
<td>Scale bitmap image usage [GX/GP]</td>
<td>2-50</td>
</tr>
<tr>
<td>SMessage</td>
<td>Message</td>
<td>2-50</td>
</tr>
</tbody>
</table>

### System Setting Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>STimeZone</td>
<td>Time zone</td>
<td>2-50</td>
</tr>
<tr>
<td>SDateBasic</td>
<td>Gradual time adjustment</td>
<td>2-50</td>
</tr>
<tr>
<td>SDateFormat</td>
<td>Date format</td>
<td>2-50</td>
</tr>
<tr>
<td>SDst</td>
<td>Daylight saving time</td>
<td>2-51</td>
</tr>
<tr>
<td>SLang</td>
<td>Language</td>
<td>2-51</td>
</tr>
<tr>
<td>STemp</td>
<td>Temperature unit</td>
<td>2-51</td>
</tr>
<tr>
<td>SPoint</td>
<td>Decimal point type</td>
<td>2-51</td>
</tr>
<tr>
<td>SFailAct</td>
<td>Fail relay operation (/FL) [GX/GP]</td>
<td>2-51</td>
</tr>
<tr>
<td>SFailSts</td>
<td>Instrument status to output (/FL) [GX/GP]</td>
<td>2-52</td>
</tr>
<tr>
<td>SPrtn</td>
<td>Printer</td>
<td>2-52</td>
</tr>
<tr>
<td>SLed</td>
<td>LED indicator operation [GX/GP]</td>
<td>2-52</td>
</tr>
<tr>
<td>SSound</td>
<td>Sound [GX/GP]</td>
<td>2-52</td>
</tr>
<tr>
<td>SInstruTag</td>
<td>Instruments tag</td>
<td>2-52</td>
</tr>
<tr>
<td>SConfCmt</td>
<td>Setting file comment</td>
<td>2-52</td>
</tr>
<tr>
<td>USBInput</td>
<td>USB input device [GX/GP]</td>
<td>2-53</td>
</tr>
<tr>
<td>SGrpInput</td>
<td>Auto logout for serial communication (/C2 or /C3)</td>
<td>2-54</td>
</tr>
</tbody>
</table>

### Time Setting Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMatchTime</td>
<td>Match time timer</td>
<td>2-42</td>
</tr>
<tr>
<td>SEventAct</td>
<td>Event action</td>
<td>2-43</td>
</tr>
</tbody>
</table>

### Report Setting Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SReport</td>
<td>Report type (/MT)</td>
<td>2-45</td>
</tr>
<tr>
<td>SRepData</td>
<td>Report data (/MT)</td>
<td>2-46</td>
</tr>
<tr>
<td>SRepTemp</td>
<td>Report output (/MT)</td>
<td>2-46</td>
</tr>
<tr>
<td>SdigitalSign</td>
<td>Electronic signature inclusion (/MT)</td>
<td>2-46</td>
</tr>
<tr>
<td>SRepCh</td>
<td>Report channel (/MT)</td>
<td>2-46</td>
</tr>
<tr>
<td>SRepBatchInfo</td>
<td>Batch information output (/MT)</td>
<td>2-47</td>
</tr>
</tbody>
</table>

### Display Setting Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLcd</td>
<td>LCD [GX/GP]</td>
<td>2-47</td>
</tr>
<tr>
<td>SViewAngle</td>
<td>View angle [GX/GP]</td>
<td>2-47</td>
</tr>
<tr>
<td>SBackColor</td>
<td>Screen background color [GX/GP]</td>
<td>2-47</td>
</tr>
<tr>
<td>SGrpChange</td>
<td>Automatic group switching time [GX/GP]</td>
<td>2-47</td>
</tr>
<tr>
<td>SAutoJump</td>
<td>Jump default display operation [GX/GP]</td>
<td>2-47</td>
</tr>
<tr>
<td>SCalFormat</td>
<td>Calendar display format [GX/GP]</td>
<td>2-48</td>
</tr>
<tr>
<td>SBarDirect</td>
<td>Bar graph display direction [GX/GP]</td>
<td>2-48</td>
</tr>
<tr>
<td>SChgMonitor</td>
<td>Value modification from the monitor</td>
<td>2-48</td>
</tr>
<tr>
<td>STrdWave</td>
<td>Trend waveform display [GX/GP]</td>
<td>2-48</td>
</tr>
<tr>
<td>STrdScale</td>
<td>Scale [GX/GP]</td>
<td>2-48</td>
</tr>
<tr>
<td>STrdLine</td>
<td>Trend line width, grid [GX/GP]</td>
<td>2-48</td>
</tr>
<tr>
<td>STrdRate</td>
<td>Trend interval switching [GX/GP]</td>
<td>2-49</td>
</tr>
<tr>
<td>STrdKind</td>
<td>Trend type [GX/GP]</td>
<td>2-49</td>
</tr>
<tr>
<td>STrdPartial</td>
<td>Partial expanded trend display [GX/GP]</td>
<td>2-49</td>
</tr>
<tr>
<td>SMsgBasic</td>
<td>Message writing</td>
<td>2-49</td>
</tr>
<tr>
<td>SGroup</td>
<td>Display group</td>
<td>2-49</td>
</tr>
<tr>
<td>STrripLine</td>
<td>Display group trip line</td>
<td>2-50</td>
</tr>
<tr>
<td>SScilamp</td>
<td>Scale bitmap image usage [GX/GP]</td>
<td>2-50</td>
</tr>
<tr>
<td>SMessage</td>
<td>Message</td>
<td>2-50</td>
</tr>
</tbody>
</table>

### Internal Switch Setting Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSwitch</td>
<td>Internal switch operation</td>
<td>2-53</td>
</tr>
</tbody>
</table>

### Serial Communication Setting Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSerialBasic</td>
<td>Serial communication basics (/C2 or /C3)</td>
<td>2-53</td>
</tr>
<tr>
<td>SModMaster</td>
<td>Modbus master (/C2/MC or /C3/MC)</td>
<td>2-54</td>
</tr>
<tr>
<td>SModMCmd</td>
<td>Modbus master transmission command (/C2/MC or /C3/MC)</td>
<td>2-54</td>
</tr>
<tr>
<td>SSerialAutoOut</td>
<td>Auto logout for serial communication (/C2 or /C3)</td>
<td>2-54</td>
</tr>
</tbody>
</table>

### Ethernet Communication Setting Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIPaddress</td>
<td>IP address information</td>
<td>2-54</td>
</tr>
<tr>
<td>SClient</td>
<td>Client function</td>
<td>2-55</td>
</tr>
<tr>
<td>SClientEncrypt</td>
<td>Client Communication Encryption</td>
<td>2-55</td>
</tr>
<tr>
<td>SDns</td>
<td>DNS information</td>
<td>2-55</td>
</tr>
<tr>
<td>SDhcp</td>
<td>DHCP client</td>
<td>2-55</td>
</tr>
<tr>
<td>SftpKind</td>
<td>File transfer via FTP</td>
<td>2-55</td>
</tr>
<tr>
<td>SftpCnct</td>
<td>FTP client connection</td>
<td>2-56</td>
</tr>
<tr>
<td>SMailHead</td>
<td>Mail header</td>
<td>2-56</td>
</tr>
<tr>
<td>SMailBasic</td>
<td>Common section of the mail</td>
<td>2-57</td>
</tr>
<tr>
<td>SMail</td>
<td>Destination and behavior for each mail type</td>
<td>2-57</td>
</tr>
<tr>
<td>SMailAlarm</td>
<td>Alarm notification mail target channels</td>
<td>2-57</td>
</tr>
<tr>
<td>SMailAlarmLevel</td>
<td>Alarm notification mail target alarm levels</td>
<td>2-57</td>
</tr>
<tr>
<td>SMailAlarmDetect</td>
<td>Alarm notification mail target alarm detection method</td>
<td>2-58</td>
</tr>
<tr>
<td>SMailTime</td>
<td>Scheduled transmission times</td>
<td>2-58</td>
</tr>
</tbody>
</table>
### 2.2 List of Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSntpCnct</td>
<td>SNTP client</td>
<td>2-58</td>
</tr>
<tr>
<td>SMdModClient</td>
<td>Modbus client operation (/MC)</td>
<td>2-58</td>
</tr>
<tr>
<td>SMdModList</td>
<td>Modbus client connection destination server (/MC)</td>
<td>2-58</td>
</tr>
<tr>
<td>SMdModCCmd</td>
<td>Modbus client transmission command (/MC)</td>
<td>2-58</td>
</tr>
<tr>
<td>SServer</td>
<td>Server function</td>
<td>2-59</td>
</tr>
<tr>
<td>SServerEncrypt</td>
<td>Server communication encryption</td>
<td>2-59</td>
</tr>
<tr>
<td>SKeepAlive</td>
<td>Keepalive</td>
<td>2-59</td>
</tr>
<tr>
<td>STimeout</td>
<td>Communication timeout</td>
<td>2-59</td>
</tr>
<tr>
<td>SFtpFormat</td>
<td>FTP server directory output format</td>
<td>2-59</td>
</tr>
<tr>
<td>SMdModDelay</td>
<td>Modbus server delay response</td>
<td>2-59</td>
</tr>
<tr>
<td>SMdModLimit</td>
<td>Modbus server connection limit</td>
<td>2-60</td>
</tr>
<tr>
<td>SMdModList</td>
<td>IP address to allow connection to Modbus server</td>
<td>2-60</td>
</tr>
<tr>
<td>SWattList</td>
<td>WT communication connection server (/E2)</td>
<td>2-60</td>
</tr>
<tr>
<td>SWattClient</td>
<td>WT communication operation (/E2)</td>
<td>2-60</td>
</tr>
<tr>
<td>SWattData</td>
<td>WT data allocation to communication channel (/E2)</td>
<td>2-60</td>
</tr>
<tr>
<td>SKdccCnct</td>
<td>KDC connection destination (/AS)</td>
<td>2-61</td>
</tr>
<tr>
<td>SAuthKey</td>
<td>Certification key (/AS)</td>
<td>2-61</td>
</tr>
<tr>
<td>SDarwinCnvCh</td>
<td>Darwin channel conversion (Darwin compatible communication)</td>
<td>2-61</td>
</tr>
<tr>
<td>SDarwinPortLimit</td>
<td>Port limitation of DARWIN compatible communication</td>
<td>2-61</td>
</tr>
<tr>
<td>SSSLMPClient</td>
<td>SLMP client operation (/E4)</td>
<td>2-61</td>
</tr>
<tr>
<td>SSSLMPList</td>
<td>SLMP connection destination server (/E4)</td>
<td>2-61</td>
</tr>
<tr>
<td>SSSLMPcmd</td>
<td>SLMP client transmission command (/E4)</td>
<td>2-62</td>
</tr>
<tr>
<td><strong>Security Setting Commands</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSecurty</td>
<td>Security function</td>
<td>2-62</td>
</tr>
<tr>
<td>SKdc</td>
<td>Password management (/AS)</td>
<td>2-63</td>
</tr>
<tr>
<td>SOPwPass</td>
<td>Password to unlock operation [GX/GP]</td>
<td>2-63</td>
</tr>
<tr>
<td>SOPwLimit</td>
<td>Operation lock details [GX/GP]</td>
<td>2-63</td>
</tr>
<tr>
<td>SUser</td>
<td>User settings</td>
<td>2-64</td>
</tr>
<tr>
<td>SUserLimit</td>
<td>Authority of user</td>
<td>2-65</td>
</tr>
<tr>
<td>SSignIn</td>
<td>Sign In (/AS)</td>
<td>2-65</td>
</tr>
<tr>
<td>SSignInTitle</td>
<td>Sign in title (/AS)</td>
<td>2-65</td>
</tr>
<tr>
<td>SSignInLimit</td>
<td>Sign in property (/AS)</td>
<td>2-65</td>
</tr>
<tr>
<td>SBTPassword</td>
<td>Bluetooth password (/I8)</td>
<td>2-66</td>
</tr>
<tr>
<td>SWebCustomMenu</td>
<td>Web monitor screen</td>
<td>2-66</td>
</tr>
<tr>
<td>SSessionSecurity</td>
<td>Web session security function (/AS) [GM]</td>
<td>2-66</td>
</tr>
<tr>
<td>SWebTimeOut</td>
<td>Web auto logout (/AS) [GM]</td>
<td>2-66</td>
</tr>
<tr>
<td><strong>Local Setting Commands</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMonitor</td>
<td>Monitor screen display information [GX/GP]</td>
<td>2-67</td>
</tr>
<tr>
<td>SMultiPattern</td>
<td>Multi panel division [GX/GP]</td>
<td>2-67</td>
</tr>
<tr>
<td>SMultiKind</td>
<td>Multi panel [GX/GP]</td>
<td>2-68</td>
</tr>
<tr>
<td>SHomeMonitor</td>
<td>Standard display information [GX/GP]</td>
<td>2-68</td>
</tr>
<tr>
<td>SHomeKind</td>
<td>Standard display [GX/GP]</td>
<td>2-68</td>
</tr>
<tr>
<td>SFavoriteMonitor</td>
<td>Favorite screen display information [GX/GP]</td>
<td>2-69</td>
</tr>
<tr>
<td>SFavoriteKind</td>
<td>Favorite screen [GX/GP]</td>
<td>2-69</td>
</tr>
<tr>
<td><strong>Multi Batch Setting Commands (IBT)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMltTextField</td>
<td>Batch text</td>
<td>2-70</td>
</tr>
<tr>
<td>SMltFileHead</td>
<td>File header</td>
<td>2-70</td>
</tr>
<tr>
<td>SMltFileName</td>
<td>File naming rule</td>
<td>2-70</td>
</tr>
<tr>
<td>SMltGroup</td>
<td>Display group</td>
<td>2-70</td>
</tr>
<tr>
<td>SMltTripline</td>
<td>Display group trip line</td>
<td>2-71</td>
</tr>
<tr>
<td>SMltSclBmp</td>
<td>Scale bitmap</td>
<td>2-71</td>
</tr>
<tr>
<td>SMltMulti</td>
<td>Multi panel pattern</td>
<td>2-71</td>
</tr>
<tr>
<td>SMltMultiKind</td>
<td>Multi panel type</td>
<td>2-71</td>
</tr>
<tr>
<td><strong>Bluetooth Communication Setting Commands</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SBluetooth</td>
<td>Bluetooth communication function (/C8) [GM]</td>
<td>2-72</td>
</tr>
<tr>
<td>SBTID</td>
<td>Bluetooth communication ID (/C8) [GM]</td>
<td>2-72</td>
</tr>
<tr>
<td>SBTTimeOut</td>
<td>Bluetooth communication timeout (/C8) [GM]</td>
<td>2-72</td>
</tr>
<tr>
<td><strong>USB Setting Commands</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUsbFunction</td>
<td>USB function [GM]</td>
<td>2-72</td>
</tr>
<tr>
<td>SUsbAutoLOut</td>
<td>USB auto logout [GM]</td>
<td>2-72</td>
</tr>
<tr>
<td><strong>Web Setting Commands</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SWebCustomMenu</td>
<td>Web monitor screen</td>
<td>2-72</td>
</tr>
<tr>
<td><strong>Schedule Management Setting Commands</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSchedule</td>
<td>Schedule management</td>
<td>2-72</td>
</tr>
<tr>
<td>SScheduleText</td>
<td>Schedule management text</td>
<td>2-73</td>
</tr>
<tr>
<td><strong>Dual Interval Function Setting Commands</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDualGroup</td>
<td>Scan group number of the display group</td>
<td>2-73</td>
</tr>
<tr>
<td>SMasterScanGrp</td>
<td>Scan group (master scan interval)</td>
<td>2-73</td>
</tr>
<tr>
<td><strong>Control Setup Parameter Setting Commands</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCtrlMode</td>
<td>Control mode</td>
<td>2-73</td>
</tr>
<tr>
<td>SCtrlScan</td>
<td>Control period</td>
<td>2-74</td>
</tr>
<tr>
<td>SCtrlAction</td>
<td>Control</td>
<td>2-74</td>
</tr>
<tr>
<td>SCtrlType</td>
<td>Control type</td>
<td>2-74</td>
</tr>
<tr>
<td>SCtrlLoopAction</td>
<td>Loop control</td>
<td>2-74</td>
</tr>
<tr>
<td>SCtrlSPPID</td>
<td>Number of SP groups, number of PID groups</td>
<td>2-75</td>
</tr>
<tr>
<td>SCtrlALNo</td>
<td>Number of alarms</td>
<td>2-75</td>
</tr>
<tr>
<td>SCtrlAlmMode</td>
<td>Alarm mode</td>
<td>2-75</td>
</tr>
<tr>
<td>SCtrlIRegist</td>
<td>Contact registration</td>
<td>2-75</td>
</tr>
<tr>
<td>SCtrlRelay</td>
<td>Relay action</td>
<td>2-78</td>
</tr>
<tr>
<td>SCtrlRangeAI</td>
<td>Measurement input range</td>
<td>2-79</td>
</tr>
<tr>
<td>SCtrlBurnOut</td>
<td>Burnout mode</td>
<td>2-79</td>
</tr>
<tr>
<td>SCtrlRjC</td>
<td>RJC temperature</td>
<td>2-80</td>
</tr>
<tr>
<td>SCtrlFilter</td>
<td>First-order lag filter of the AI terminal of the PID control module</td>
<td>2-80</td>
</tr>
<tr>
<td>SCtrlCalibAI</td>
<td>Calibration correction of the AI terminal of the PID control module</td>
<td>2-80</td>
</tr>
<tr>
<td>SCtrlRangeAO</td>
<td>Range of the AO terminal of the PID control module</td>
<td>2-82</td>
</tr>
</tbody>
</table>
### 2.2 List of Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCtrlOutOperate</td>
<td>Output value manual output operation type</td>
</tr>
<tr>
<td>STagIO</td>
<td>PID control module channel display (tag, tag No.)</td>
</tr>
<tr>
<td>SColorIO</td>
<td>PID control module channel display (color)</td>
</tr>
<tr>
<td>SZoneIO</td>
<td>PID control module channel display (zone high limit, zone low limit)</td>
</tr>
<tr>
<td>SScaleIO</td>
<td>PID control module channel display (scale display position, number of scale divisions)</td>
</tr>
<tr>
<td>SBarIO</td>
<td>PID control module channel display (bar display position, number of bar divisions)</td>
</tr>
<tr>
<td>SPartialO</td>
<td>PID control module channel display (partial)</td>
</tr>
<tr>
<td>SValueO</td>
<td>PID control module channel display (upper and lower limit string)</td>
</tr>
</tbody>
</table>

### 2.2.2 Output Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description (Required Options)</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>FData</td>
<td>Outputs the most recent channel data [Applicable Models]</td>
<td>2-94</td>
</tr>
<tr>
<td>FRelay</td>
<td>Outputs the most recent relay and internal switch status</td>
<td>2-94</td>
</tr>
<tr>
<td>FTransStatAO</td>
<td>Outputs the most recent re-transmission (AO channel) status</td>
<td>2-94</td>
</tr>
<tr>
<td>FFifoCur</td>
<td>Outputs channel FIFO data</td>
<td>2-94</td>
</tr>
<tr>
<td>FSnap</td>
<td>Takes a snapshot [GX/GP]</td>
<td>2-95</td>
</tr>
<tr>
<td>FUser</td>
<td>Outputs the user level</td>
<td>2-95</td>
</tr>
<tr>
<td>FAddr</td>
<td>Outputs the IP address</td>
<td>2-95</td>
</tr>
<tr>
<td>FStat</td>
<td>Outputs the GX/GP status</td>
<td>2-95</td>
</tr>
<tr>
<td>FLog</td>
<td>Outputs the log</td>
<td>2-95</td>
</tr>
<tr>
<td>FEventLog</td>
<td>Outputs a detail event log (/AS)</td>
<td>2-96</td>
</tr>
<tr>
<td>FMedia</td>
<td>Outputs external storage medium and internal memory information</td>
<td>2-96</td>
</tr>
<tr>
<td>FCnf</td>
<td>Outputs settings data</td>
<td>2-97</td>
</tr>
<tr>
<td>FChInfo</td>
<td>Outputs decimal place and unit information</td>
<td>2-98</td>
</tr>
<tr>
<td>FSysConf</td>
<td>Queries the system configuration and reconfigures modules</td>
<td>2-98</td>
</tr>
<tr>
<td>FBTDevInfo</td>
<td>Bluetooth device information output (/C8) [GM]</td>
<td>2-98</td>
</tr>
<tr>
<td>FReminder</td>
<td>Outputs reminder information (/AH)</td>
<td>2-98</td>
</tr>
<tr>
<td>FCtrlData</td>
<td>Control data output</td>
<td>2-98</td>
</tr>
<tr>
<td>FCtrlNo</td>
<td>SP number and PID number output</td>
<td>2-99</td>
</tr>
</tbody>
</table>

**Control Operation Parameter Setting Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCtrlBackColor</td>
<td>Background color [GX/GP]</td>
</tr>
<tr>
<td>SCtrlOutOperate</td>
<td>Output value manual output operation type</td>
</tr>
<tr>
<td>STagIO</td>
<td>PID control module channel display (tag, tag No.)</td>
</tr>
<tr>
<td>SColorIO</td>
<td>PID control module channel display (color)</td>
</tr>
<tr>
<td>SZoneIO</td>
<td>PID control module channel display (zone high limit, zone low limit)</td>
</tr>
<tr>
<td>SScaleIO</td>
<td>PID control module channel display (scale display position, number of scale divisions)</td>
</tr>
<tr>
<td>SBarIO</td>
<td>PID control module channel display (bar display position, number of bar divisions)</td>
</tr>
<tr>
<td>SPartialO</td>
<td>PID control module channel display (partial)</td>
</tr>
<tr>
<td>SValueO</td>
<td>PID control module channel display (upper and lower limit string)</td>
</tr>
</tbody>
</table>

**Program Control Setting Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPrmColor</td>
<td>Loop color</td>
</tr>
<tr>
<td>SPrmDispDetail</td>
<td>Auto message printing, screen switching</td>
</tr>
</tbody>
</table>

**Control Event Action Setting Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCtrlEventAct</td>
<td>Control event action</td>
</tr>
</tbody>
</table>

**Logic Math Setting Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLogicMath</td>
<td>Logic math expression</td>
</tr>
<tr>
<td>SWConst</td>
<td>Variable constant</td>
</tr>
</tbody>
</table>

**Control Display Setting Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCtrlGroupSW</td>
<td>Control group use On/Off</td>
</tr>
<tr>
<td>SCtrlGroupName</td>
<td>Control group name</td>
</tr>
<tr>
<td>SCtrlSplit</td>
<td>Control group divisions [GX/GP]</td>
</tr>
<tr>
<td>SCtrlGroup</td>
<td>Loop to assign to control group</td>
</tr>
<tr>
<td>SCtrlTag</td>
<td>Loop tag, tag comment</td>
</tr>
<tr>
<td>SCtrlDispDV</td>
<td>Deviation display band</td>
</tr>
</tbody>
</table>
2.2 List of Commands

2.2.3 Operation Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSendTime</td>
<td>Sets the time</td>
<td>2-101</td>
</tr>
<tr>
<td>ORec</td>
<td>Starts or stops recording</td>
<td>2-101</td>
</tr>
<tr>
<td>OAlarmAck</td>
<td>Clears alarm output (alarm acknowledgement)</td>
<td>2-101</td>
</tr>
<tr>
<td>OExecRec</td>
<td>Generates a manual trigger, executes a sample,</td>
<td>2-101</td>
</tr>
<tr>
<td></td>
<td>takes a snapshot, or causes a timeout</td>
<td></td>
</tr>
<tr>
<td>OExecSNTP</td>
<td>Queries the time using SNTP</td>
<td>2-102</td>
</tr>
<tr>
<td>OMessage</td>
<td>Writes a message</td>
<td>2-102</td>
</tr>
<tr>
<td>OPassword</td>
<td>Changes the password</td>
<td>2-102</td>
</tr>
<tr>
<td>OMath</td>
<td>Starts, stops, or resets</td>
<td>2-102</td>
</tr>
<tr>
<td></td>
<td>computation or clears the status display</td>
<td></td>
</tr>
<tr>
<td>OSaveConf</td>
<td>Saves setting data</td>
<td>2-103</td>
</tr>
<tr>
<td>OSaveConfAll</td>
<td>Saves setting data at once</td>
<td>2-103</td>
</tr>
<tr>
<td>OCommCh</td>
<td>Sets a communication channel to a value</td>
<td>2-103</td>
</tr>
<tr>
<td>OEmail</td>
<td>Starts or stops the e-mail transmission function</td>
<td>2-103</td>
</tr>
<tr>
<td>OMBRestore</td>
<td>Recovers Modbus manually</td>
<td>2-104</td>
</tr>
<tr>
<td>OMReset</td>
<td>Resets a relative timer</td>
<td>2-104</td>
</tr>
<tr>
<td>OMReset</td>
<td>Resets the match time timer</td>
<td>2-104</td>
</tr>
<tr>
<td>OCmdRelay</td>
<td>Outputs the DO channel and internal switch</td>
<td>2-104</td>
</tr>
<tr>
<td></td>
<td>status</td>
<td></td>
</tr>
<tr>
<td>OBatName</td>
<td>Sets a batch name</td>
<td>2-104</td>
</tr>
<tr>
<td>OBatComment</td>
<td>Sets a batch comment</td>
<td>2-105</td>
</tr>
<tr>
<td>OBatText</td>
<td>Sets a batch text</td>
<td>2-105</td>
</tr>
<tr>
<td>ODispRate</td>
<td>Switches the trend interval [GX/GP]</td>
<td>2-105</td>
</tr>
<tr>
<td>OLoadConf</td>
<td>Loads setting data</td>
<td>2-105</td>
</tr>
<tr>
<td>OLoadConfAll</td>
<td>Loads setting data at once</td>
<td>2-106</td>
</tr>
<tr>
<td>OSeriApply</td>
<td>Applies serial communication settings</td>
<td>2-106</td>
</tr>
<tr>
<td>OIPApply</td>
<td>Applies the IP address</td>
<td>2-106</td>
</tr>
<tr>
<td>OInit</td>
<td>Clears measured data and initializes setting</td>
<td>2-106</td>
</tr>
<tr>
<td>OUsbFApply</td>
<td>Applies USB communication settings</td>
<td>2-107</td>
</tr>
<tr>
<td>OBtApply</td>
<td>Applies Bluetooth communication settings [GM]</td>
<td>2-107</td>
</tr>
<tr>
<td>OBTClearList</td>
<td>Clears the Bluetooth connection list [IC8]</td>
<td>2-107</td>
</tr>
</tbody>
</table>

2.2.4 Communication Control Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>OLoginAssist</td>
<td>Assists login [GX/GP]</td>
<td>2-107</td>
</tr>
<tr>
<td>OSendValue</td>
<td>Assists touch panel operation [IC8/GP]</td>
<td>2-107</td>
</tr>
<tr>
<td>OUserLockACK</td>
<td>User locked ACK [IAS]</td>
<td>2-107</td>
</tr>
<tr>
<td>OKeyLock</td>
<td>Key lock on/off [GM]</td>
<td>2-107</td>
</tr>
<tr>
<td>OErrorClear</td>
<td>Clears the error display [GM]</td>
<td>2-108</td>
</tr>
<tr>
<td>OSLMPRestore</td>
<td>Manually restores SLMP [IC8/E4]</td>
<td>2-108</td>
</tr>
<tr>
<td>OtransChAO</td>
<td>Individual re-transmission output [AO channel]</td>
<td>2-108</td>
</tr>
<tr>
<td>OtransAllAO</td>
<td>Collective re-transmission output [AO channel]</td>
<td>2-108</td>
</tr>
<tr>
<td>OCmdAO</td>
<td>Manual output setting</td>
<td>2-108</td>
</tr>
<tr>
<td>OinitPara</td>
<td>Individual setting parameter initialization</td>
<td>2-108</td>
</tr>
<tr>
<td>OCtrlAM</td>
<td>Auto/manual/cascade operation switching</td>
<td>2-109</td>
</tr>
<tr>
<td>OCtrlSR</td>
<td>Operation start/stop switching</td>
<td>2-109</td>
</tr>
<tr>
<td>OCtrlRL</td>
<td>Remote/local switching</td>
<td>2-109</td>
</tr>
<tr>
<td>OCtrlAT</td>
<td>Auto-tuning request</td>
<td>2-109</td>
</tr>
<tr>
<td>OCtrlSPN</td>
<td>Selects the target setpoint number</td>
<td>2-109</td>
</tr>
<tr>
<td>OCtrlLMO</td>
<td>Sets the manual output setpoint</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlPAT</td>
<td>Pattern number switching</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlMode</td>
<td>Program operation start or stop</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlHOLD</td>
<td>Hold operation</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlADV</td>
<td>Advance operation</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlSP</td>
<td>Sets the target setpoint</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlTSR</td>
<td>Sets the final target setpoint</td>
<td>2-111</td>
</tr>
<tr>
<td>OCtrlRTIME</td>
<td>Sets the segment remaining</td>
<td>2-111</td>
</tr>
<tr>
<td>OCtrlStSeg</td>
<td>Sets the start segment number</td>
<td>2-111</td>
</tr>
<tr>
<td>OCtrlDlyTime</td>
<td>Sets the starting time of program operation</td>
<td>2-111</td>
</tr>
<tr>
<td>OCtrlLoadPAT</td>
<td>Loads a program pattern file</td>
<td>2-112</td>
</tr>
<tr>
<td>OCtrlLoadPATAll</td>
<td>Collectively Loads program pattern files</td>
<td>2-112</td>
</tr>
<tr>
<td>OCtrlSavePAT</td>
<td>Saves a program pattern file</td>
<td>2-112</td>
</tr>
<tr>
<td>OCtrlSavePATAll</td>
<td>Collectively saves program pattern files</td>
<td>2-112</td>
</tr>
<tr>
<td>OCtrlDelPAT</td>
<td>Deletes a pattern file</td>
<td>2-112</td>
</tr>
</tbody>
</table>

2.2.5 Commands and Responses

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>OMath</td>
<td>Starts or stops the e-mail</td>
<td>2-105</td>
</tr>
<tr>
<td></td>
<td>transmission function</td>
<td></td>
</tr>
<tr>
<td>OMBRestore</td>
<td>Recovers Modbus manually</td>
<td>2-104</td>
</tr>
<tr>
<td>OMBWrite</td>
<td>Writes a message</td>
<td>2-102</td>
</tr>
<tr>
<td>OCmdRelay</td>
<td>Outputs the DO channel and internal switch</td>
<td>2-104</td>
</tr>
<tr>
<td></td>
<td>status</td>
<td></td>
</tr>
<tr>
<td>OBatName</td>
<td>Sets a batch name</td>
<td>2-104</td>
</tr>
<tr>
<td>OBatComment</td>
<td>Sets a batch comment</td>
<td>2-105</td>
</tr>
<tr>
<td>OBatText</td>
<td>Sets a batch text</td>
<td>2-105</td>
</tr>
<tr>
<td>ODispRate</td>
<td>Switches the trend interval [GX/GP]</td>
<td>2-105</td>
</tr>
<tr>
<td>OLoadConf</td>
<td>Loads setting data</td>
<td>2-105</td>
</tr>
<tr>
<td>OLoadConfAll</td>
<td>Loads setting data at once</td>
<td>2-106</td>
</tr>
<tr>
<td>OSeriApply</td>
<td>Applies serial communication settings</td>
<td>2-106</td>
</tr>
<tr>
<td>OIPApply</td>
<td>Applies the IP address</td>
<td>2-106</td>
</tr>
<tr>
<td>OInit</td>
<td>Clears measured data and initializes</td>
<td>2-106</td>
</tr>
<tr>
<td></td>
<td>setting data</td>
<td></td>
</tr>
<tr>
<td>OUsbFApply</td>
<td>Applies USB communication settings</td>
<td>2-107</td>
</tr>
<tr>
<td>OBtApply</td>
<td>Applies Bluetooth communication settings [GM]</td>
<td>2-107</td>
</tr>
<tr>
<td>OBTClearList</td>
<td>Clears the Bluetooth connection list [IC8]</td>
<td>2-107</td>
</tr>
<tr>
<td>OLoginAssist</td>
<td>Assists login [GX/GP]</td>
<td>2-107</td>
</tr>
<tr>
<td>OSendValue</td>
<td>Assists touch panel operation [IC8/GP]</td>
<td>2-107</td>
</tr>
<tr>
<td>OUserLockACK</td>
<td>User locked ACK [IAS]</td>
<td>2-107</td>
</tr>
<tr>
<td>OKeyLock</td>
<td>Key lock on/off [GM]</td>
<td>2-107</td>
</tr>
<tr>
<td>OErrorClear</td>
<td>Clears the error display [GM]</td>
<td>2-108</td>
</tr>
<tr>
<td>OSLMPRestore</td>
<td>Manually restores SLMP [IC8/E4]</td>
<td>2-108</td>
</tr>
<tr>
<td>OtransChAO</td>
<td>Individual re-transmission output [AO channel]</td>
<td>2-108</td>
</tr>
<tr>
<td>OtransAllAO</td>
<td>Collective re-transmission output [AO channel]</td>
<td>2-108</td>
</tr>
<tr>
<td>OCmdAO</td>
<td>Manual output setting</td>
<td>2-108</td>
</tr>
<tr>
<td>OinitPara</td>
<td>Individual setting parameter initialization</td>
<td>2-108</td>
</tr>
<tr>
<td>OCtrlAM</td>
<td>Auto/manual/cascade operation switching</td>
<td>2-109</td>
</tr>
<tr>
<td>OCtrlSR</td>
<td>Operation start/stop switching</td>
<td>2-109</td>
</tr>
<tr>
<td>OCtrlRL</td>
<td>Remote/local switching</td>
<td>2-109</td>
</tr>
<tr>
<td>OCtrlAT</td>
<td>Auto-tuning request</td>
<td>2-109</td>
</tr>
<tr>
<td>OCtrlSPN</td>
<td>Selects the target setpoint number</td>
<td>2-109</td>
</tr>
<tr>
<td>OCtrlLMO</td>
<td>Sets the manual output setpoint</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlPAT</td>
<td>Pattern number switching</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlMode</td>
<td>Program operation start or stop</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlHOLD</td>
<td>Hold operation</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlADV</td>
<td>Advance operation</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlSP</td>
<td>Sets the target setpoint</td>
<td>2-110</td>
</tr>
<tr>
<td>OCtrlTSR</td>
<td>Sets the final target setpoint</td>
<td>2-111</td>
</tr>
<tr>
<td>OCtrlRTIME</td>
<td>Sets the segment remaining</td>
<td>2-111</td>
</tr>
<tr>
<td>OCtrlStSeg</td>
<td>Sets the start segment number</td>
<td>2-111</td>
</tr>
<tr>
<td>OCtrlDlyTime</td>
<td>Sets the starting time of program operation</td>
<td>2-111</td>
</tr>
<tr>
<td>OCtrlLoadPAT</td>
<td>Loads a program pattern file</td>
<td>2-112</td>
</tr>
<tr>
<td>OCtrlLoadPATAll</td>
<td>Collectively Loads program pattern files</td>
<td>2-112</td>
</tr>
<tr>
<td>OCtrlSavePAT</td>
<td>Saves a program pattern file</td>
<td>2-112</td>
</tr>
<tr>
<td>OCtrlSavePATAll</td>
<td>Collectively saves program pattern files</td>
<td>2-112</td>
</tr>
<tr>
<td>OCtrlDelPAT</td>
<td>Deletes a pattern file</td>
<td>2-112</td>
</tr>
</tbody>
</table>

2.2.6 Commands and Responses

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>OMath</td>
<td>Starts or stops the e-mail</td>
<td>2-105</td>
</tr>
<tr>
<td></td>
<td>transmission function</td>
<td></td>
</tr>
<tr>
<td>OMBRestore</td>
<td>Recovers Modbus manually</td>
<td>2-104</td>
</tr>
<tr>
<td>OMBWrite</td>
<td>Writes a message</td>
<td>2-102</td>
</tr>
<tr>
<td>OCmdRelay</td>
<td>Outputs the DO channel and internal switch</td>
<td>2-104</td>
</tr>
<tr>
<td></td>
<td>status</td>
<td></td>
</tr>
<tr>
<td>OBatName</td>
<td>Sets a batch name</td>
<td>2-104</td>
</tr>
<tr>
<td>OBatComment</td>
<td>Sets a batch comment</td>
<td>2-105</td>
</tr>
<tr>
<td>OBatText</td>
<td>Sets a batch text</td>
<td>2-105</td>
</tr>
<tr>
<td>ODispRate</td>
<td>Switches the trend interval [GX/GP]</td>
<td>2-105</td>
</tr>
<tr>
<td>OLoadConf</td>
<td>Loads setting data</td>
<td>2-105</td>
</tr>
<tr>
<td>OLoadConfAll</td>
<td>Loads setting data at once</td>
<td>2-106</td>
</tr>
<tr>
<td>OSeriApply</td>
<td>Applies serial communication settings</td>
<td>2-106</td>
</tr>
<tr>
<td>OIPApply</td>
<td>Applies the IP address</td>
<td>2-106</td>
</tr>
<tr>
<td>OInit</td>
<td>Clears measured data and initializes</td>
<td>2-106</td>
</tr>
<tr>
<td></td>
<td>setting data</td>
<td></td>
</tr>
<tr>
<td>OUsbFApply</td>
<td>Applies USB communication settings</td>
<td>2-107</td>
</tr>
<tr>
<td>OBtApply</td>
<td>Applies Bluetooth communication settings [GM]</td>
<td>2-107</td>
</tr>
<tr>
<td>OBTClearList</td>
<td>Clears the Bluetooth connection list [IC8]</td>
<td>2-107</td>
</tr>
</tbody>
</table>
2.2 List of Commands

2.2.5 Instrument Information

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>_MFG</td>
<td>Outputs the instrument manufacturer</td>
<td>2-115</td>
</tr>
<tr>
<td>_INF</td>
<td>Outputs the instrument’s product name</td>
<td>2-115</td>
</tr>
<tr>
<td>_COD</td>
<td>Outputs the instrument’s basic specifications</td>
<td>2-115</td>
</tr>
<tr>
<td>_VER</td>
<td>Outputs the instrument’s firmware version information</td>
<td>2-115</td>
</tr>
<tr>
<td>_OPT</td>
<td>Outputs the instrument’s option installation information</td>
<td>2-115</td>
</tr>
<tr>
<td>_TYP</td>
<td>Outputs the instrument’s temperature unit, and daylight saving time installation information</td>
<td>2-115</td>
</tr>
<tr>
<td>_ERR</td>
<td>Outputs the instrument’s error number information</td>
<td>2-115</td>
</tr>
<tr>
<td>_UNS</td>
<td>Outputs the instrument’s unit configuration information</td>
<td>2-115</td>
</tr>
<tr>
<td>_UNR</td>
<td>Outputs the instrument’s unit configuration information</td>
<td>2-115</td>
</tr>
<tr>
<td>_MDS</td>
<td>Outputs the instrument’s module configuration information</td>
<td>2-115</td>
</tr>
<tr>
<td>_MDR</td>
<td>Outputs the instrument’s module configuration information</td>
<td>2-115</td>
</tr>
</tbody>
</table>

2.2.6 Conditions for Executing Commands

A command can be executed only when the recorder can execute the setting change or operation that the command specifies. Commands are invalid in the following circumstances.

- The recorder is not in a condition to accept the operation.
  - For example, if the recorder is not recording, you cannot write a message.

- If the recorder does not have the function or is not using the function.

The “Description” column in section 2.2.1, "Setting Commands" contains the recorder suffix codes that are required for using the commands.

- If the login function is in use, the command cannot be used at the user level that the user is logged in at.

- User restriction is placed on the operation.

The following table lists the commands that are invalid according to the limitation types (p1 of the SOpelLimit command or p2 of the SUSerLimit command).

<table>
<thead>
<tr>
<th>Limitation Type</th>
<th>Invalid Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td>ORec</td>
</tr>
<tr>
<td>Math</td>
<td>OMath</td>
</tr>
<tr>
<td>DataSave</td>
<td>OExecRec, OMTRest</td>
</tr>
<tr>
<td>Message</td>
<td>OMessage</td>
</tr>
<tr>
<td>Batch</td>
<td>OBatName, OBatComment, OBatText</td>
</tr>
<tr>
<td>AlarmACK</td>
<td>OAlarmAck</td>
</tr>
<tr>
<td>Comm</td>
<td>OEMail, OIPApply, OMBArestore, OSLMPRestore</td>
</tr>
</tbody>
</table>

DispOpe SHomeKind, SHomeMonitor, SFavoriteKind, SFavoriteMonitor, Smonitor, SMultiPattern, SMultiKind, ODispRate

DateSet OExecSNTP, OSetTime

ChangeSet Sxxxx¹, OLoadConf, OLoadConfAll, OInit³, OCtrlLoadPAT, OCtrlLoadPATAll

File OLoadConf, OLoadConfAll, OSaveConf, OSaveConfAll, OCtrlLoadPAT, OCtrlSavePAT, OCtrlLoadPATAll, OCtrlSavePATAll, Fmedia

System OInit, FSysConf (when pl is specified)

Out OCmdRelay, OCommCh

CalibSet² SCalibIO, SSchedule, SScheduleText, OLoadConfAll, OLoadConf⁴, OInit³, SCalibUseCom, SCalibCom

ControlIN OCtrlRL

ControlOUT OCtrlISR, OCtrlAM, OCtrlMO

Tuning OCtrlAT

Program OCtrlMODE, OCtrl1HOLD, OCtrl1ADV

¹ Setting commands except for SHomeKind, SHomeMonitor, SFavoriteKind, SFavoriteMonitor, Smonitor, SMultiPattern, SMultiKind, and SCalib[Note], SCalibUseCom, SCalibCom

² Can be specified with the SUserLimit command when the advanced security function (IAS option) is in use on instruments whose version is 2.02 or later.

³ Cannot be executed if initialization items include SECURITY or OTHERS items.

⁴ Cannot be executed if load items include CALIB items.

⁵ Cannot be executed if initialization items include CALIB items.

- The command is not applicable to the model.

For commands that can be used only on certain models, the models are listed in the “Description” column in section 2.2.1, “Setting Commands,” to section 2.2.4, “Communication Control Commands.” (Examples: [GX/GP], [GM])

The applicable models for the following commands are further reduced.

<table>
<thead>
<tr>
<th>Command</th>
<th>Applicable Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>SViewAngle</td>
<td>GX10, GP10</td>
</tr>
<tr>
<td>SMultiPattern</td>
<td>GX20, GP20</td>
</tr>
<tr>
<td>SMultiKind</td>
<td>GX20, GP20</td>
</tr>
</tbody>
</table>

*Note: Only when the advanced security function (IAS option) is in use on instruments whose version is 2.02 or later.*
2.2.7 How to Use Commands

When Using Ethernet
- When not using the login function
  When you connect a PC to the recorder, the recorder will be ready to receive commands.
- When using the login function
  Establish communication with the recorder, and log in using a registered user account (CLogin command). After you finish the operation, log out (CLogout command).

When Using RS-232 (GX/GP)
- When you wire and connect a PC to the GX/GP, the GX/GP will be ready to receive commands.
- When using the login function, log in using a registered user account (CLogin command). After you finish the operation, log out (CLogout command).

When Using RS-422/485
- The device that is opened with an open command (ESC o) will be ready to receive commands.
- When using the login function, log in using a registered user account (CLogin command). After you finish the operation, log out (CLogout command).
- To close the connection, send the close command (ESC c).

When Using USB Communication (GM)
- When not using the login function
  When you connect a PC to the GM, the GM will be ready to receive commands.
- When using the login function
  Log in using a registered user account (CLogin command) to establish a connection. After you finish the operation, log out (CLogout command). You can also use the auto logout function (SUsbAutoLOut command).
- To remove a GM, perform a device removal procedure on the PC to disconnect, and then remove the cable.

When Using Bluetooth (GM, /C8 option)
- When not using the login function
  When the Bluetooth password function is enabled, use a command to start communication (CBTConnect) to send the password. When a connection is established, the GM will be ready to receive commands.
- When using the login function
  In addition to the procedure above, log in using a registered user account (CLogin command). After you finish the operation, log out (CLogout command). You can also use the auto logout function (SBTTimeOut command).
- To disconnect, perform a device removal procedure on the PC.

Note
- For the login operation, see appendix 2, “Login Procedure.”
- For details on Bluetooth connection, see appendix 7, “Bluetooth Communication Connection Flow Chart.”

2.2.8 Device Nomenclature in Command Descriptions

The following nomenclature is used in the command descriptions in section 2.4 to distinguish the devices.

<table>
<thead>
<tr>
<th>Nomenclature</th>
<th>Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recorder</td>
<td>Both GX/GP and GM</td>
</tr>
<tr>
<td>Main unit</td>
<td>Both GX/GP and GM main units</td>
</tr>
<tr>
<td>GX/GP main unit</td>
<td>GX/GP main unit</td>
</tr>
<tr>
<td>GM main unit</td>
<td>GM main unit</td>
</tr>
<tr>
<td>GX20-1/GP20-1</td>
<td>GX20/GP20 standard type</td>
</tr>
<tr>
<td>GX20-2/GP20-2</td>
<td>GX20/GP20 large memory type</td>
</tr>
<tr>
<td>GM10-1</td>
<td>GM10 standard type</td>
</tr>
<tr>
<td>GM10-2</td>
<td>GM10 large memory type</td>
</tr>
<tr>
<td>Expandable I/O</td>
<td>GX/GP Expandable I/O</td>
</tr>
<tr>
<td>Sub unit</td>
<td>GM sub unit</td>
</tr>
</tbody>
</table>
2.3 Parameters

This section describes parameters.

2.3.1 Measuring Range Parameters

AI Channel Span

Specify the span using an integer.
Example If the range is -2.0000 V to 2.0000 V and you want to set the span lower limit to 0.5000 V and the span upper limit to 1.8000 V, set the parameters to 5000 and 18000, respectively.

Scaling

Scaling is possible on AI and DI channels. Scaling is specified by a mantissa and decimal place.
Example To set the scaling to -10.00 to 20.00, set the scaling lower limit to -1000, scaling upper limit to 2000, and the decimal place to 2. The decimal place value represents the number of digits to the right of the decimal point.

Math Channel and Communication Channel Span

Set the span of math channels and communication channels using a mantissa and decimal place.
Example To set the span to 1.00 to 2.00, set the scaling lower limit to 1000, scaling upper limit to 2000, and the decimal place to 3.

2.3.2 Parameter Notation and Range

The table below shows the principle parameter notations and ranges of values.

<table>
<thead>
<tr>
<th>Type</th>
<th>Notation and Range of Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>[GX/GP]</td>
<td></td>
</tr>
<tr>
<td>Unit number</td>
<td>No expandable I/O 0</td>
</tr>
<tr>
<td></td>
<td>Expandable I/O installed 0 to 6</td>
</tr>
<tr>
<td>[GX/GP]</td>
<td>When the unit is GX10/GP10 0 to 2</td>
</tr>
<tr>
<td>Module number</td>
<td>When the unit is GX20/GP20 0 to 9</td>
</tr>
<tr>
<td>[GM]</td>
<td>No sub unit 0</td>
</tr>
<tr>
<td></td>
<td>Sub unit installed 0 to 6</td>
</tr>
<tr>
<td>[GM]</td>
<td>Main unit 0 to 9</td>
</tr>
<tr>
<td></td>
<td>Sub unit 0 to 6</td>
</tr>
<tr>
<td>AI channel</td>
<td>Specify as &quot;unit number+module number+channel.&quot;</td>
</tr>
<tr>
<td>DI channel</td>
<td>Example The AI channel whose unit number is 0, module number is 1, and channel number is 02 is 0102.</td>
</tr>
<tr>
<td>DO channel</td>
<td></td>
</tr>
<tr>
<td>PI channel</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Notation and Range of Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID channel</td>
<td>Specify as &quot;unit number+slot number+channel.&quot;</td>
</tr>
<tr>
<td></td>
<td>Example The PID channel whose unit number is 0, slot number is 1, and channel number is 02 is 0102.</td>
</tr>
<tr>
<td></td>
<td>Note that the PID channel number changes depending on the channel type. The following table shows the association.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Channel type</th>
<th>Channel number (4 digits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PV</td>
<td>**01, **04</td>
</tr>
<tr>
<td>SP</td>
<td>**02, **05</td>
</tr>
<tr>
<td>OUT</td>
<td>**03, **06</td>
</tr>
<tr>
<td>AI</td>
<td>**07, **08</td>
</tr>
<tr>
<td>AO</td>
<td>**09, **10</td>
</tr>
<tr>
<td>DI</td>
<td>**11 to **18</td>
</tr>
<tr>
<td>DO</td>
<td>**19 to **26</td>
</tr>
</tbody>
</table>

Math channel

GX20-2/GP20-2: 001 to 200
GX10/GP10: 001 to 050
GM10: 001 to 100

For SGroup and SMailAlarm commands, insert "A" in front.
Example A001

If the measurement mode is dual interval, the number of channels is halved.

Communication channel

GX10/GP10:001 to 050
GX20-1/GP20-1: 001 to 300
GX20-2/GP20-2: 001 to 500
GM10-1: 001 to 300
GM10-2: 001 to 500

For SGroup and SMailAlarm commands, insert "C" in front.
Example C001

If the measurement mode is dual interval, the number of channels is halved.

Number of channels for recording display data

GX10/GP10: 001 to 100
GX20-1/GP20-1: 001 to 500
GX20-2/GP20-2: 001 to 1000
GM10-1: 1 to 500
GM10-2: 1 to 1000

Number of channels for recording event data

GX10/GP10: 001 to 100
GX20-1/GP20-1: 001 to 500
GX20-2/GP20-2: 001 to 1000
GM10-1: 1 to 500
GM10-2: 1 to 1000

Number of channels for recording manual sampled data

GX10/GP10/GX20-1/GP20-1: 1 to 50
GX20-2/GP20-2: 1 to 100
GM10-1: 1 to 50
GM10-2: 1 to 100

Number of report channels

GX10/GP10: 1 to 50
GX20/GP20: 1 to 60
GM10: 1 to 60

Number of display groups

GX10/GP10: 1 to 30
GX20-1/GP20-1: 1 to 50
GX20-2/GP20-2: 1 to 60
GM10-1: 1 to 50
GM10-2: 1 to 60
### Type Notation and Range of Values

<table>
<thead>
<tr>
<th>Type</th>
<th>Notation and Range of Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of channels that can be registered</td>
<td>GX10/GP10: 10</td>
</tr>
<tr>
<td>to display groups</td>
<td>GX20/GP20: 20</td>
</tr>
<tr>
<td>Modbus server setting number</td>
<td>GM10: 20</td>
</tr>
<tr>
<td>Modbus command number</td>
<td>GX10/GP10/GX20-1/GP20-1: 1 to 16</td>
</tr>
<tr>
<td>(Ethernet)</td>
<td>GX20-2/GP20-2: 1 to 32</td>
</tr>
<tr>
<td>Modbus command number (serial communication)</td>
<td>GX10/GP10: 1 to 50</td>
</tr>
<tr>
<td></td>
<td>GX20/GP20: 1 to 100</td>
</tr>
<tr>
<td>Server setting number for WT communication</td>
<td>GX10/GP10: 1 to 8</td>
</tr>
<tr>
<td></td>
<td>GX20/GP20: 1 to 16</td>
</tr>
<tr>
<td>Communication channel allocation number for WT communication</td>
<td>GX10/GP10: 1 to 50</td>
</tr>
<tr>
<td></td>
<td>GX20/GP20: 1 to 300</td>
</tr>
<tr>
<td>Number of users that can be registered</td>
<td>Advanced security function (/AS) not installed or disabled: 1 to 50</td>
</tr>
<tr>
<td>(user number)</td>
<td>Advanced security function (/AS) enabled: 1 to 100</td>
</tr>
<tr>
<td>Number of batch groups in use for the multi batch function (/BT)</td>
<td>GX10/GP10: 2 to 6</td>
</tr>
<tr>
<td></td>
<td>GX20-1/GX20-1: 2 to 6</td>
</tr>
<tr>
<td></td>
<td>GX20-2/GP20-2: 2 to 12</td>
</tr>
<tr>
<td>Batch group number when the multi batch function is enabled</td>
<td>1 to (number of batch groups in use)</td>
</tr>
<tr>
<td>Number of display groups when the multi batch function (/BT) is enabled</td>
<td>GX10/GP10: 1 to 6</td>
</tr>
<tr>
<td></td>
<td>GX20-1/GX20-1: 1 to 6</td>
</tr>
<tr>
<td></td>
<td>GX20-2/GP20-2: 1 to 12</td>
</tr>
<tr>
<td>Schedule registration number</td>
<td>GX10/GP10: 1 to 6</td>
</tr>
<tr>
<td></td>
<td>GX20-1/GX20-2: 1 to 6</td>
</tr>
<tr>
<td></td>
<td>GX20-2/GP20-2: 1 to 12</td>
</tr>
<tr>
<td>Number of control groups</td>
<td>GX10-1/GX20-1/GM10-1: 1 to 5</td>
</tr>
<tr>
<td></td>
<td>GX20-2/GM10-2: 1 to 10</td>
</tr>
<tr>
<td>Loop number</td>
<td>L001 to L652</td>
</tr>
<tr>
<td></td>
<td>To specify the loop number, add an &quot;L&quot; in front, and specify as &quot;unit number+module number+loop number.&quot;</td>
</tr>
<tr>
<td></td>
<td>Example: The loop whose unit number is 0, module number is 0, and loop number is 1 is L001.</td>
</tr>
<tr>
<td>PID module terminal number</td>
<td>The terminal numbers in a PID module consists of AI, AO, DI, and DO.</td>
</tr>
<tr>
<td></td>
<td>AI: 1 or 2, AO: 1 or 2, DI: 1 to 8, DO: 1 to 8</td>
</tr>
<tr>
<td>Logic math number</td>
<td>GX10: 1 to 20</td>
</tr>
<tr>
<td></td>
<td>GX20-1/GX20-2, GM10-1/GM10-2: 1 to 50</td>
</tr>
</tbody>
</table>

### 2.3.3 Specifying a Range

When specifying consecutive channel numbers or group numbers in a setting command, you can specify them using a range instead of specifying each number one by one.

- Use a hyphen to separate the first number and the last number. For I/O channels, you can specify a range that spans over multiple slots that modules are installed in.
- You can specify the minimum number by omitting the number before the hyphen and the maximum number by omitting the number after the hyphen. If you want to specify all numbers from the first number to the last number, specify only the hyphen.

#### Example 1
- To specify 3 to 10: “3-10”
- To specify 3 to the maximum number: “3-”
- To specify the first number to 10: “-10”
- To specify all numbers: “-”

#### Example 2
- A command that sets the channel ranges of AI modules installed in slots 0 to 2 to Skip.
  - SRangeAI,0001-0210,Skip or SRangeAI,-0210,Skip
- If a different module is installed in slot 1, queries will work, but setting commands will result in error.
2.4 Setting Commands

SScan
Scan Interval
Sets the scan interval.

Syntax
SScan,p1,p2
p1 Scan group (1 or 2)
When the measurement mode is Normal or High speed, this is fixed to 1. When the measurement mode is Dual interval, you can select 1 or 2.
p2 Scan interval (see "Description")

Query
SScan[,p1]?

Example
Set the scan interval of scan group 1 to 1 second.
SScan,1,1s

Description
• You cannot use this command to configure settings while recording is in progress.
• You cannot use this command to configure settings while computation is in progress.
• The scan interval (p2) can be set in the following range.

<table>
<thead>
<tr>
<th>Measurement mode</th>
<th>Options (p2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>100ms, 200ms, 500ms, 1s, 2s, 5s (Notes *1, *2, *3)</td>
</tr>
<tr>
<td>High speed</td>
<td>1ms, 2ms, 5ms, 10ms, 20ms, 50ms, 100ms, 200ms, 500ms, 1s, 2s, 5s</td>
</tr>
<tr>
<td>Dual interval</td>
<td>1ms, 2ms, 5ms, 10ms, 20ms, 50ms, 100ms, 200ms, 500ms, 1s, 2s, 5s (Notes *4, *5, *6, *7)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module type (type, suffix code)</th>
<th>Shortest scan interval (when installed in the main unit)</th>
<th>Shortest scan interval (when installed in an expandable I/O)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analog input module (high-speed AI, -H0)</td>
<td>1ms</td>
<td>100ms</td>
</tr>
<tr>
<td>(Universal, -U2)</td>
<td>100ms</td>
<td>100ms</td>
</tr>
<tr>
<td>(Electromagnetic relay, -T1)</td>
<td>500ms</td>
<td>500ms</td>
</tr>
<tr>
<td>(Low withstand voltage relay, -L1)</td>
<td>1s</td>
<td>1s</td>
</tr>
<tr>
<td>(Current input, -C1)</td>
<td>100ms</td>
<td>100ms</td>
</tr>
<tr>
<td>(4-wire RTD, -R1)</td>
<td>100ms</td>
<td>100ms</td>
</tr>
<tr>
<td>(High withstand voltage, -V1)</td>
<td>500ms</td>
<td>500ms</td>
</tr>
<tr>
<td>Digital input module</td>
<td>100ms</td>
<td>100ms</td>
</tr>
<tr>
<td>Digital output module</td>
<td>100ms</td>
<td>100ms</td>
</tr>
<tr>
<td>Analog output module</td>
<td>100ms</td>
<td>100ms</td>
</tr>
<tr>
<td>Digital input/output module</td>
<td>100ms</td>
<td>100ms</td>
</tr>
<tr>
<td>Pulse input module</td>
<td>100ms</td>
<td>100ms</td>
</tr>
<tr>
<td>Expansion module</td>
<td>100ms</td>
<td>100ms</td>
</tr>
<tr>
<td>PID control module</td>
<td>100ms</td>
<td>100ms</td>
</tr>
</tbody>
</table>

*1 If an electro-magnetic relay type analog input module is installed, scan interval less than or equal to 500 ms cannot be specified.

*2 If a low withstand voltage relay type analog input module is installed, scan interval less than or equal to 200 ms cannot be specified.

*3 When the multi-batch function (/BT) is enabled, you cannot set the scan interval to 200 ms or less.

*4 If high-speed AI type analog input module is installed, scan interval less than or equal to 50 ms cannot be specified.

*5 If an electro-magnetic relay type analog input module is assigned to a scan group, scan interval less than or equal to 500 ms cannot be specified.

*6 If a low withstand voltage relay type analog input module is assigned to a scan group, scan interval less than or equal to 200 ms cannot be specified.

*7 For dual interval measurement, scan interval cannot be set less than 50 ms on scan group 2.

The following three conditions must be met for an option to be a valid scan interval.

1 When specifying a scan interval of 50 ms or less, the number of channels must be within the limits. (For the limits to the number of channels for each scan interval, see the main unit’s User’s Manual (IM 04L51B01-01EN, or IM 04L55B01-01EN).)

2 When specifying a scan interval of 50 ms or less, the number of channels must be within the limits. (For the limits to the number of channels for each scan interval, see the main unit’s User’s Manual (IM 04L51B01-01EN, or IM 04L55B01-01EN).)

3 The shortest scan interval of the modules assigned to a scan group is shorter than the scan interval of the scan group. The following table shows the shortest scan interval of each module.

SScanGroup
Scan Group
Sets the scan group of the module.

Syntax
SScanGroup,p1,p2,p3
p1 Unit number
p2 Module number
p3 Scan group (1)

Query
SScanGroup[,p1[,p2]]?

Example
Set the module installed in the main unit, whose module number is 2 in scan group 1.
SScanGroup,0,2,1

Description
• You cannot use this command to configure settings while recording is in progress.
• You cannot use this command to configure settings while computation is in progress.
• The scan group (p3) can be set in the following range.

<table>
<thead>
<tr>
<th>Measurement mode</th>
<th>Options (p3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal, High speed</td>
<td>Fixed to 1</td>
</tr>
<tr>
<td>Dual interval</td>
<td>1 or 2</td>
</tr>
</tbody>
</table>

The following three conditions must be met for an option to be a valid scan group.

1 For a module installed in the main unit, assign the module to a scan group with a scan interval of 50 ms or less.
If the scan interval is 50 ms or less, the number of installed modules and the number of channels must be within their limits.

The shortest scan interval of the module must be shorter than the scan interval of the scan group.

### SModeAI

**AI Module**

Sets the mode and A/D integration time or noise rejection mode of an AI module (excluding current input type AI modules).

**Syntax**

```text
SModeAI,p1,p2,p3,p4
```

- **p1**: Unit number
- **p2**: Module number
- **p3**: Mode (see "Description.")
- **p4**: A/D integration time or noise rejection mode
  - A/D integration time: Auto, 50Hz, 60Hz, Common
  - Noise rejection mode (high-speed AI type): Off, 50Hz, 60Hz, Common

**Query**

```text
SModeAI[,p1[,p2]]?
```

**Example**

For the module installed in the main unit, whose module number is 2, set the mode to 10CH and the AD integration time to Auto.

```text
SModeAI,0,2,10CH,Auto
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- There are limitations on the allowable combinations of scan interval and p3 and p4. See the explanation for the SModeAI command.

#### Module type Mode (p3)

- Universal type (-U2)
- Current input type (-C1)
- High withstand voltage (-V1)
- Electromagnetic relay type (-T1)
- Low withstand voltage relay type (-L1)
- 4-wire RTD type (-R1)
- High-speed AI type (-H0)

#### Scan interval Mode (p3) Integration time (p4)

<table>
<thead>
<tr>
<th>Scan interval</th>
<th>Mode (p3)</th>
<th>Integration time (p4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto</td>
<td>50Hz</td>
<td>60Hz</td>
</tr>
<tr>
<td>Common</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### 2.4 Setting Commands

### SModeAICurrent

**Current Input Type AI Module**

Sets the mode and A/D integration time of an current input type AI module.

**Syntax**

```text
SModeAICurrent,p1,p2,p3,p4
```

- **p1**: Unit number
- **p2**: Module number
- **p3**: Mode
  - 2CH: 2 channel mode
  - 10CH: 10 channel mode
- **p4**: AD integration time (Auto, 50Hz, 60Hz, Common)

**Query**

```text
SModeAICurrent[,p1[,p2]]?
```

**Example**

For the module installed in the main unit, whose module number is 2, set the mode to 10CH and the AD integration time to Auto.

```text
SModeAICurrent,0,2,10CH,Auto
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- There are limitations on the allowable combinations of scan interval and p3 and p4. See the explanation for the SModeAI command.
### SBOLmtAI

#### Upper and Lower Burnout Limits of AI Module
Sets the burnout limits for the general signal range of an AI module (excluding current input type AI modules).

**Syntax**
```
SBOLmtAI,p1,p2,p3,p4
```
- **p1**: Unit number
- **p2**: Module number
- **p3**: Lower burnout limit for the general signal range. Percentage of the specified span
  -20.0 to -5.0% (-200 to -50)
- **p4**: Upper burnout limit for the general signal range. Percentage of the specified span
  105.0 to 120.0% (1050 to 1200)

**Query**
```
SBOLmtAI[,p1[,p2]]?  
```

**Example**
For the module installed in the main unit, whose module number is 2, set the lower burnout limit for the general signal range to -10% and the upper burnout limit for the general signal range to 110%.
```
SBOLmtAI,0,2,-100,1100
```

**Description**
- This command is invalid for 4-wire RTD modules.
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

### SBOLmtAICurrent

#### Upper and Lower Burnout Limits of Current Input Type AI Module
Sets the burnout limits for the general signal range of a current input type AI module.

**Syntax**
```
SBOLmtAICurrent,p1,p2,p3,p4
```
- **p1**: Unit number
- **p2**: Module number
- **p3**: Lower burnout limit for the general signal range. Percentage of the specified span
  -20.0 to -5.0% (-200 to -50)
- **p4**: Upper burnout limit for the general signal range. Percentage of the specified span
  105.0 to 120.0% (1050 to 1200)

**Query**
```
SBOLmtAICurrent[,p1[,p2]]?  
```

**Example**
For the module installed in the main unit, whose module number is 2, set the lower burnout limit for the general signal range to -10% and the upper burnout limit for the general signal range to 110%.
```
SBOLmtAICurrent,0,2,-100,1100
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

### SModeDI

#### DI Module
Sets the mode of a DI module.

**Syntax**
```
SModeDI,p1,p2,p3
```
- **p1**: Unit number
- **p2**: Module number
- **p3**: Mode (Normal, Remote)
  - Normal: DI input
  - Remote: Remote control input

**Query**
```
SModeDI[,p1[,p2]]?  
```

**Example**
Set the module with a module number of 2 as a remote control input module.
```
SModeDI,0,2,Remote
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- Only one module can be set to remote. If different modules are set to remote numerous times, the last module will be the remote module.
- For modules installed in an expandable I/O or sub unit, p3 is fixed to Normal.
- Pulse input is valid on products with the math function (/MT option).
- When the measurement mode is set to high speed, p3 is fixed to Remote.
### SScaleOver

**Detection of Values That Exceed the Scale**
Sets how to detect measurement over-range.

**Syntax**

```plaintext
SScaleOver, p1
/P1
```

- **FREE** Assume scale over-range when the measurement range is exceeded.
- **OVER** Assume scale over-range when ±105% of the scale is exceeded.

**Query**

`SScaleOver?`

**Example**

Assume scale over-range when the measurement range is exceeded.

`SScaleOver,FREE`

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- The setting specified with this command is valid if at least one module is installed.

### SMemKeyConfirm

**Record Confirmation Action [GX/GP]**
Sets the record confirmation action.

**Syntax**

```plaintext
SMemKeyConfirm, p1
```

- **p1** Enable or disable confirmation screen (Off, On)

**Query**

`SMemKeyConfirm?`

**Example**

Show the confirmation screen.

`SMemKeyConfirm,On`

**Description**

- When the multi batch function (/BT) is enabled, this is fixed to On.

### SDispData

**Display Data Recording**
Sets the display data recording mode.

**Syntax**

```plaintext
SDispData, p1, p2
```

- **p1** Recording interval (5s, 10s, 15s, 30s, 1min, 2min, 5min, 10min, 15min, 20min, 30min, 1h, 2h, 4h, 10h)/div.
- **p2** File save interval (10min, 20min, 30min, 1h, 2h, 3h, 4h, 6h, 8h, 12h, 1day, 2day, 3day, 5day, 7day, 14day, 31day)

**Query**

`SDispData?`

**Example**

Set the recording interval to 1 minute and file save interval to 12 hours.

`SDispData,1min,12h`

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- You cannot choose a recording interval that is shorter than the scan interval.
- You cannot choose a recording interval that is not an integer multiple of the scan interval.
- File save interval is valid when display data recording is enabled (recording mode of the `SMemory` command).
- This command is invalid when the measurement mode is set to high speed or dual interval.
## SEventData

### Event Data Recording

Sets the event data recording mode.

**Syntax**

```
SEventData,p1,p2,p3,p4,p5,p6
```

<table>
<thead>
<tr>
<th>p1</th>
<th>Scan group (1 or 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>p2</td>
<td>Recording interval (see &quot;Description.&quot;)</td>
</tr>
<tr>
<td>p3</td>
<td>Operation mode</td>
</tr>
<tr>
<td></td>
<td>Free</td>
</tr>
<tr>
<td></td>
<td>Starts recording at recording start and stops recording at recording stop.</td>
</tr>
<tr>
<td></td>
<td>SingleTrigger</td>
</tr>
<tr>
<td></td>
<td>After a trigger event occurs, the recorder will record for the specified time and stop.</td>
</tr>
<tr>
<td></td>
<td>RepeatTrigger</td>
</tr>
<tr>
<td></td>
<td>After a trigger event occurs, the recorder will record for the specified time and stop. Then, the recorder will enter the trigger-wait state.</td>
</tr>
<tr>
<td>p4</td>
<td>Data length (2min, 5min, 10min, 20min, 30min, 1h, 2h, 3h, 4h, 6h, 8h, 12h, 1day, 2day, 3day, 5day, 7day, 14day, 31day)</td>
</tr>
<tr>
<td>p5</td>
<td>Pre-trigger (0, 5, 25, 50, 75, 95, 100) [%]</td>
</tr>
<tr>
<td>p6</td>
<td>Trigger source key (Off, On)</td>
</tr>
</tbody>
</table>

**Query**

```
SEventData[,p1]?
```

**Example**

Record event data in Free mode at a recording interval of 1 second. Separate the data into different files every 2 hours.

```
SEventData,1,1s,Free,2h
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- You cannot choose a recording interval that is shorter than the scan interval.
- You cannot choose a recording interval that is not an integer multiple of the scan interval.
- p1 = 2 is valid when the measurement mode is set to dual interval.
- The recording interval (p2) can be set in the following range.

<table>
<thead>
<tr>
<th>Measurement mode</th>
<th>Options (p2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>100ms, 200ms, 500ms, 1s, 2s, 5s, 10s, 15s, 20s, 30s, 1min, 2min, 5min, 10min, 15min, 20min, 30min</td>
</tr>
<tr>
<td>High speed</td>
<td>1ms, 2ms, 5ms, 10ms, 20ms, 50ms, 100ms, 200ms, 500ms, 1s, 2s, 5s, 10s, 15s, 20s, 30s, 1min, 2min, 5min, 10min, 15min, 20min, 30min</td>
</tr>
<tr>
<td>Dual interval</td>
<td>1ms, 2ms, 5ms, 10ms, 20ms, 50ms, 100ms, 200ms, 500ms, 1s, 2s, 5s, 10s, 15s, 20s, 30s, 1min, 2min, 5min, 10min, 15min, 20min, 30min</td>
</tr>
</tbody>
</table>

1. There are limits to the recording interval (p2) depending on the model, scan interval, and number of recording channels (see SRecEvent).
2. Data length (p4) can be set to 2min or 5min when the measurement mode is set to high speed.
3. Data length (p4) can be set to 5min when the measurement mode is set to dual interval.
4. This setting is valid when event data recording is enabled (recording mode of the SMemory command).
5. When the advanced security function (/AS) is enabled, p3 is fixed to Free.
6. When the multi batch function (/BT) is enabled, p3 is fixed to Free.

<table>
<thead>
<tr>
<th>Scan interval</th>
<th>Recording intervals less than or equal to 50 ms</th>
</tr>
</thead>
<tbody>
<tr>
<td>1ms</td>
<td>1ms, 2ms, 5ms, 10ms, 20ms, 50ms</td>
</tr>
<tr>
<td>2ms</td>
<td>2ms, 10ms, 20ms, 50ms</td>
</tr>
<tr>
<td>5ms</td>
<td>5ms, 10ms, 20ms, 50ms</td>
</tr>
<tr>
<td>10ms</td>
<td>10ms, 20ms, 50ms</td>
</tr>
<tr>
<td>20ms</td>
<td>20ms</td>
</tr>
<tr>
<td>50ms</td>
<td>50ms</td>
</tr>
</tbody>
</table>

- The recording intervals (p2) less than or equal to 50 ms can be set in the following range.
**SRecDisp**

**Channel for Recording Display Data**

Sets the channel for recording display data.

**Syntax**

\[ \text{SRecDisp}(p1, p2, p3) \]

- **p1**: Number (see “Description”)
- **p2**: Channel type
  - Off: Do not record display data.
  - IO: I/O channel
  - Math: Math channel
  - Com: Communication channel
- **p3**: Channel number

**Query**

\[ \text{SRecDisp}(p1) \]

**Example**

Assign the display data of I/O channel 0005 to number 10 and record.

\[ \text{SRecDisp}(10, 10, 0005) \]

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- You cannot use this command to configure settings when the measurement mode is set to dual interval.
- If p2=Off, you cannot set p3.
- There is a limit to the number of recording channels depending on the recording interval (SDispData command).

<table>
<thead>
<tr>
<th>Recording Interval</th>
<th>Number of Recording Channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 s/div</td>
<td>100</td>
</tr>
<tr>
<td>10 s/div</td>
<td>200</td>
</tr>
<tr>
<td>15 s/div or higher</td>
<td>500</td>
</tr>
</tbody>
</table>

For the large memory type (GX20/2/GP20/2/GM10/2), the following table applies.

<table>
<thead>
<tr>
<th>Recording Interval</th>
<th>Number of Recording Channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 s/div</td>
<td>100</td>
</tr>
<tr>
<td>10 s/div</td>
<td>200</td>
</tr>
<tr>
<td>15 s/div or higher</td>
<td>500</td>
</tr>
</tbody>
</table>

- You cannot set a channel more than once.

**SRecEvent**

**Channel for Recording Event Data**

Sets the channel for recording event data.

**Syntax**

\[ \text{SRecEvent}(p1, p2, p3, p4) \]

- **p1**: Scan group (1 or 2)
- **p2**: Number (see “Description”)
- **p3**: Channel type
  - Off: Do not record event data.
  - IO: I/O channel
  - Math: Math channel
  - Com: Communication channel
- **p4**: Channel number

**Query**

\[ \text{SRecEvent}(p1, p2) \]

**Example**

Assign the event data of I/O channel 0006 to number 11 and record.

\[ \text{SRecEvent}(11, 10, 0006) \]

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- If p3=Off, you cannot set p4.
- This setting is valid when event data recording is enabled (recording mode of the SMemory command).
- There is a limit to the number of recording channels depending on the recording interval (SDispData command).

**Recording Interval**

<table>
<thead>
<tr>
<th>Number of Recording Channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 ms</td>
</tr>
<tr>
<td>200 ms</td>
</tr>
<tr>
<td>500 ms or more</td>
</tr>
</tbody>
</table>

For the large memory type (GX20/2/GP20/2/GM10/2), the following table applies.

<table>
<thead>
<tr>
<th>Recording Interval</th>
<th>Number of Recording Channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>When recording only event data</td>
<td>When recording display data and event data</td>
</tr>
<tr>
<td>100 ms</td>
<td>500</td>
</tr>
<tr>
<td>200 ms</td>
<td>200</td>
</tr>
<tr>
<td>500 ms or more</td>
<td>1000</td>
</tr>
<tr>
<td>1 s or more</td>
<td>1000</td>
</tr>
</tbody>
</table>

- If the recording interval is 50 ms or less, there are limits to the number of recording channels depending on the model and measurement mode.

**High speed**

**Recording Interval**

<table>
<thead>
<tr>
<th>Number of Recording Channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 ms</td>
</tr>
<tr>
<td>200 ms</td>
</tr>
<tr>
<td>500 ms or more</td>
</tr>
</tbody>
</table>

For the large memory type (GX20/2/GP20/2/GM10/2), the following table applies.

<table>
<thead>
<tr>
<th>Recording Interval</th>
<th>Number of Recording Channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>When recording only event data</td>
<td>When recording display data and event data</td>
</tr>
<tr>
<td>100 ms</td>
<td>500</td>
</tr>
<tr>
<td>200 ms</td>
<td>200</td>
</tr>
<tr>
<td>500 ms or more</td>
<td>1000</td>
</tr>
<tr>
<td>1 s or more</td>
<td>1000</td>
</tr>
</tbody>
</table>

- You cannot set a channel more than once.
- When the measurement mode is set to dual interval and p3 = IO, you can set the channels of only the applicable scan groups.
- If the measurement mode is set to dual interval, p3 = Math and p3 = Com can be specified only on scan groups operating at the master scan interval.
## SRecManual

**Channel for Recording Manual Sampled Data**
Sets the channel for recording manual sampled data.

**Syntax**
```
SRecManual, p1, p2, p3
```

- **p1** Number (1 to 50)
- **p2** Channel type
  - `Off`: Do not record manual sampled data.
  - `IO`: I/O channel
  - `Math`: Math channel (/MT)
  - `Com`: Communication channel (/MC)
- **p3** Channel number

**Query**
```
SRecManual[,p1]?
```

**Example**
Assign the manual sampled data of I/O channel 0003 to number 2 and record.
```
SRecManual, 2, IO, 0003
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- If `p2=Off`, you cannot set `p3`.
- You cannot set a channel more than once.

## SBatch

**Batch Function**
Configures the batch function's basic settings.

**Syntax**
```
SBatch, p1, p2, p3, p4
```

- **p1** Enable or disable (Off, On)
- **p2** Number of lot number digits (Off, 4, 6, 8)
  - `Off`: Do not use lot numbers.
  - 4: 4-digit lot number
  - 6: 6-digit lot number
  - 8: 8-digit lot number
- **p3** Auto increment (Off, On)
- **p4** Recording start screen (Comment, TextField)
  - `Comment`: Batch comment
  - `TextField`: Text field

**Query**
```
SBatch?
```

**Example**
Enable the batch function. Use 4-digit lot numbers. Automatically increment the lot number in the next operation.
```
SBatch, On, 4, On, TextField
```

**Description**
- For the characters that you can use in the directory name (p1), see Appendix 1.
- The following character strings cannot be used for directory names.

<table>
<thead>
<tr>
<th>Character String</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUX</td>
</tr>
<tr>
<td>CON</td>
</tr>
<tr>
<td>PRN</td>
</tr>
<tr>
<td>NUL</td>
</tr>
<tr>
<td>CLOCK</td>
</tr>
<tr>
<td>CLOCK$</td>
</tr>
<tr>
<td>COM0 to COM9</td>
</tr>
<tr>
<td>LPT0 to LPT9</td>
</tr>
</tbody>
</table>
- You cannot use a character string that starts or ends with a period or space for directory names.

## STextField

**Batch Text**
Sets a batch text.

**Syntax**
```
STextField, p1, p2, p3
```

- **p1** Field number (1 to 24)
- **p2** Title (up to 20 characters, UTF-8)
- **p3** Character string (up to 30 characters, UTF-8)

**Query**
```
STextField[,p1]?
```

**Example**
For field number 3, set the field title to "OPERATOR" and the character string to "RECODER1."
```
STextField, 3, 'OPERATOR', 'RECODER1'
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- This command is valid only when the multi batch function (/BT) is disabled.

## SDirectory

**Name of Directory to Save Data**
Sets the name of the directory to save data.

**Syntax**
```
SDirectory, p1
```

- **p1** Directory name (up to 20 characters, ASCII)

**Query**
```
SDirectory?
```

**Example**
Set the directory name to "DATA0."
```
SDirectory, 'DATA0'
```

**Description**
- For the characters that you can use in the directory name (p1), see Appendix 1.
- The following character strings cannot be used for directory names.

<table>
<thead>
<tr>
<th>Character String</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUX</td>
</tr>
<tr>
<td>CON</td>
</tr>
<tr>
<td>PRN</td>
</tr>
<tr>
<td>NUL</td>
</tr>
<tr>
<td>CLOCK</td>
</tr>
<tr>
<td>CLOCK$</td>
</tr>
<tr>
<td>COM0 to COM9</td>
</tr>
<tr>
<td>LPT0 to LPT9</td>
</tr>
</tbody>
</table>
- You cannot use a character string that starts or ends with a period or space for directory names.

## SFileHead

**File Header**
Sets the file header character string.

**Syntax**
```
SFileHead, p1
```

- **p1** File header (up to 50 characters, UTF-8)

**Query**
```
SFileHead?
```

**Example**
Set the file header to "GX_DATA."
```
SFileHead, 'GX_DATA'
```

**Description**
- This command is valid only when the multi batch function (/BT) is disabled.
### SFileName

**File Naming Rule**
Sets the file naming rule for data files.

**Syntax**
```
SFileName,p1,p2
```
- **p1** File naming rule
  - Date
  - Serial
  - Batch
- **p2** Specified file name (up to 16 characters, ASCII)

**Query**
```
SFileName?
```

**Example**
Set the file naming rule to “Date.” Set the specified file name to “Recorder1_data.”
```
SFileName,Date,’Recorder1_data’
```

**Description**
- If the batch setting is disabled (SBatch: p1=Off), you cannot specify p1=Batch.
- For the characters that you can use in the specified file name (p2), see Appendix 1.
- This command is valid only when the multi batch function (/BT) is disabled.

### SMediaSave

**Automatic Data File Saving**
Sets the auto saving of data files to an external storage medium.

**Syntax**
```
SMediaSave,p1,p2
```
- **p1** Auto saving to an external storage medium
  - (GX/GP: Off, On)
  - (GM: Off, On, Fixed to On when the advanced security function (/AS) is enabled and the log in via communication is enabled.)
- **p2** Media FIFO (Off, On)

**Query**
```
SMediaSave?
```

**Example**
Enable the auto saving to the external storage medium and media FIFO.
```
SMediaSave,On,On
```

### SFileFormat

**Display/Event Data File Format**
Sets the file format of display data files and event data files.

**Syntax**
```
SFileFormat,p1
```
- **p1** File format (Binary, Text)

**Query**
```
SFileFormat?
```

**Example**
Create files in text format.
```
SFileFormat,Text
```

**Description**
- The types of data that you can set file formats for are display data and event data.
- The file saving methods that the specified file format is applied to are auto saving, saving of unsaved data, manual saving, and FTP data transfer.
- When the advanced security function (/AS) is enabled, p1 is fixed to Binary.
### SRangeAI

**Measurement Range of AI Channel**

Sets the measurement range of an AI channel.

**Unused Channels**

**Syntax**

```
SRangeAI,p1,p2
```

- `p1`: Channel number
- `p2`: Input type (Skip)

**Channels Whose Input Type Is DI and No Math**

**Syntax**

```
SRangeAI,p1,p2,p3,p4,p5,p6
```

- `p1`: Channel number
- `p2`: Input type (DI)
- `p3`: Range (see “Description.”)
- `p4`: Calculation type (Off)
- `p5`: Span lower limit
- `p6`: Span upper limit

**Channels Whose Input Type Is Volt, TC, RTD, or OHM and No Calculation**

**Syntax**

```
SRangeAI,p1,p2,p3,p4,p5,p6,p7
```

- `p1`: Channel number
- `p2`: Input type (Volt, TC, RTD)
- `p3`: Range (see “Description.”)
- `p4`: Calculation type (Off)
- `p5`: Span lower limit
- `p6`: Span upper limit
- `p7`: Bias (–999999 to 999999)

**Delta Channels**

**Syntax**

```
SRangeAI,p1,p2,p3,p4,p5,p6,p7,p8
```

- `p1`: Channel number
- `p2`: Input type (Volt, TC, RTD, DI, OHM)
- `p3`: Range (see “Description.”)
- `p4`: Calculation type (Delta)
- `p5`: Span lower limit
- `p6`: Span upper limit
- `p7`: Bias (–999999 to 999999) (can be set when `p2` is not set to DI)
- `p8`: Reference channel number

**Scaling Channels**

**Syntax**

```
SRangeAI,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11
```

- `p1`: Channel number
- `p2`: Input type (Volt, TC, RTD, DI, OHM)
- `p3`: Range (see “Description.”)
- `p4`: Calculation type (Scale)
- `p5`: Span lower limit
- `p6`: Span upper limit
- `p7`: Bias (–999999 to 999999) (can be set when `p2` is not set to DI)
- `p8`: Decimal Place of mantissa (1, 2)
- `p9`: Scaling lower limit (exponential notation, 1.00E-15 to 1.00E15)
- `p10`: Scaling upper limit (exponential notation, 1.00E-15 to 1.00E15)
- `p11`: Unit (up to 6 characters, UTF-8)

**Unified Signal Input Channels (Input Type Is GS)**

**Syntax**

```
SRangeAI,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14
```

- `p1`: Channel number
- `p2`: Input type (GS)
- `p3`: Range (see “Description.”)
- `p4`: Calculation type (Scale)
- `p5`: Span lower limit
- `p6`: Span upper limit
- `p7`: Bias (–999999 to 999999)
- `p8`: Decimal Place (0 to 5)
- `p9`: Scaling lower limit
- `p10`: Scaling upper limit
- `p11`: Unit (up to 6 characters, UTF-8)
- `p12`: Low-cut function (Off, On)
- `p13`: Low-cut point (0 to 50)

**Square Root Channels**

**Syntax**

```
SRangeAI,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14
```

- `p1`: Channel number
- `p2`: Input type (Volt, GS)
- `p3`: Range (see “Description.”)
- `p4`: Calculation type (Sqrt)
- `p5`: Span lower limit
- `p6`: Span upper limit
- `p7`: Bias (–999999 to 999999)
- `p8`: Decimal Place (0 to 5)
- `p9`: Scaling lower limit
- `p10`: Scaling upper limit
- `p11`: Unit (up to 6 characters, UTF-8)
- `p12`: Low-cut function (Off, On)
- `p13`: Low-cut point (0 to 50)
- `p14`: Low-cut output (Zero, Linear)

**Log Scale (LG) Channels**

**Syntax**

```
SRangeAI,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13
```

- `p1`: Channel number
- `p2`: Input type (Volt)
- `p3`: Range (see “Description.”)
- `p4`: Calculation type (LogT1, LogT2, LogT3)
- `p5`: Span lower limit
- `p6`: Span upper limit
- `p7`: Bias (–999999 to 999999)
- `p8`: Decimal place of mantissa (1, 2)
- `p9`: Scaling lower limit (exponential notation, 1.00E-15 to 1.00E15)
- `p10`: Scaling upper limit (exponential notation, 1.00E-15 to 1.00E15)
- `p11`: Unit (up to 6 characters, UTF-8)

**Query**

```
SRangeAI[,p1]?
```

**Example**

Measure -0.5000 to 1.0000 V on channel 0002. No scaling. No bias.

```
SRangeAI,0002,Volt,2V,Off,–5000,10000,0
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
• You cannot use this command to configure settings while computation is in progress.
  • If p2=TC/RTD/DI/OHM, you cannot specify p4=Sqrt.
  • If p2=GS, you cannot specify p4=Off/Delta.
  • If p2=DI, you cannot set p7.
  • If an electro-magnetic relay type, low withstand voltage relay type or high withstand voltage type analog input module is in use, you cannot specify p2=RTD.
  • For 4-wire RTD modules, p2 cannot be set to Volt, TC, GS, or DI. (Only Skip, RTD, and OHM are selectable.)
  • p2 = OHM is valid only for 4-wire RTD modules.
  • The settable items for p3 are shown below.

<table>
<thead>
<tr>
<th>p2</th>
<th>p2=Volt</th>
<th>p2=TC</th>
<th>p2=RTD</th>
<th>p2=GS</th>
<th>p2=DI</th>
</tr>
</thead>
<tbody>
<tr>
<td>20mV</td>
<td>R</td>
<td>Pt100</td>
<td>1-5V</td>
<td>Level</td>
<td></td>
</tr>
<tr>
<td>60mV</td>
<td>S</td>
<td>Pt100-H</td>
<td>0.4-2V</td>
<td>DI</td>
<td></td>
</tr>
<tr>
<td>200mV</td>
<td>B</td>
<td>JPt100</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1V</td>
<td>K</td>
<td>JPt100-H</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2V</td>
<td>K-H</td>
<td>Cu10GE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6V</td>
<td>E</td>
<td>Cu10LN</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20V</td>
<td>J</td>
<td>Cu10WEED</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30V</td>
<td>T</td>
<td>Cu10BAILEY</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100V^</td>
<td>N</td>
<td>Cu10a392</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>Cu10a393</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L</td>
<td>Cu25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>U</td>
<td>Cu53</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PLATINEL</td>
<td>Cu100</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PR20-40</td>
<td>J263B</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WR3-25</td>
<td>Ni100SAMA</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>KpvsAu7Fe</td>
<td>Ni100DIN</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NiNiMo</td>
<td>Ni120</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WRRe26</td>
<td>Pt25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ni4</td>
<td>Pt50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>XK</td>
<td>Pt200WEED</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cu10G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cu50G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cu100G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pt46G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pt100G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pt500^</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pt1000^</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 Valid only for high-speed AI modules
2 Valid only for 4-wire RTD modules

• If p4=LogT1 on a Log scale channel, set the value in the following range.
  • p5<p6
  • p9<p10. The maximum span is 15 decades.
    If the mantissa of p9 is 1.00, the minimum span is 1 decade.
    If the mantissa of p9 is not 1.00, the minimum span is 2 decades.

• If p4=LogT2 or LogT3 on a Log scale channel, set the value in the following range.
  • p5<p6
  • p9, p10
    The maximum span is 15 decades; the minimum is 1 decade.
    If the mantissa of p9 is not 1.00, the exponent is +14 or less, and the maximum span is 14 decades.

### SRangeAlCurrent

**Measurement Range of Current Input Type AI Channel**

Sets the measurement range of an current input type AI channel.

#### Unused Channels

**Syntax**

```
SRangeAlCurrent,p1,p2
```

- **p1** Channel number
- **p2** Input type (Skip)

#### Channels Whose Input Type is Current and No Math

**Syntax**

```
SRangeAlCurrent,p1,p2,p3,p4,p5,p6,p7
```

- **p1** Channel number
- **p2** Input type (Current)
- **p3** Range (0-20mA)
- **p4** Math type (Off)
- **p5** Span lower limit
- **p6** Span upper limit
- **p7** Bias (–999999 to 999999)

#### Delta Channels

**Syntax**

```
SRangeAlCurrent,p1,p2,p3,p4,p5,p6,p7,p8
```

- **p1** Channel number
- **p2** Input type (Current)
- **p3** Range (0-20mA)
- **p4** Math type (Delta)
- **p5** Span lower limit
- **p6** Span upper limit
- **p7** Bias (–999999 to 999999)
- **p8** Reference channel number

#### Scaling Channels

**Syntax**

```
SRangeAlCurrent,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13
```

- **p1** Channel number
- **p2** Input type (Current)
- **p3** Range (0-20mA)
- **p4** Math type (Scale)
- **p5** Span lower limit
- **p6** Span upper limit
- **p7** Bias (–999999 to 999999)
- **p8** Decimal place (0 to 5)
- **p9** Scaling lower limit
- **p10** Scaling upper limit
- **p11** Unit (up to 6 characters, UTF-8)

#### Scaling Channels (General Signal 4-20 mA Input)

**Syntax**

```
SRangeAlCurrent,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13
```

- **p1** Channel number
- **p2** Input type (GS)
- **p3** Range (4-20mA)
- **p4** Math type (Scale)
2.4 Setting Commands

Square Root Channels

Syntax

SRangeAICurrent, p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12, p13, p14

p1 Channel number
p2 Input type (Current, GS)
p3 Range
0-20mA When p2 = Current
4-20mA When p2 = GS
p4 Math type (Sqrt)
p5 Span lower limit
p6 Span upper limit
p7 Bias (−999999 to 999999)
p8 Decimal place (0 to 5)
p9 Scaling lower limit
p10 Scaling upper limit
p11 Unit (up to 6 characters, UTF-8)
p12 Low-cut function (Off, On)
p13 Low-cut point (0 to 50)
p14 Low-cut output (Zero, Linear)

Query

SRangeAICurrent[, p1]?

Example

Measure 0.000 to 10.000 mA on channel 0002.
No scaling. No bias.
SRangeAICurrent, 0002, Current, 0-20mA, Off, 0, 10000, 0

Description

• You cannot use this command to configure settings while recording is in progress.
• You cannot use this command to configure settings while computation is in progress.
• If p2=GS, you cannot specify p4=Off/Delta.
• Specify p5 and p6 within the range shown in the following table.

<table>
<thead>
<tr>
<th>Range (p3)</th>
<th>Value (p5, p6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-20mA</td>
<td>0.000 to 20.000</td>
</tr>
<tr>
<td>4-20mA</td>
<td>3.200 to 20.800</td>
</tr>
</tbody>
</table>

SRangePulse

Measurement Range of Pulse Input Channel

Sets the measurement range of a pulse input channel.

Unused Channels

Syntax

SRangePulse, p1, p2

p1 Channel number
p2 Input type (Skip)

Channels Whose Input Type is PulseInput and No Math

Syntax

SRangePulse, p1, p2, p3, p4, p5, p6, p7

p1 Channel number
p2 Input type (PulseInput)
p3 Range (LevelRange, ContactRange)
LevelRange Level
ContactRange Contact
p4 Chattering filter (On, Off)
p5 Math type (Off)
p6 Span lower limit
p7 Span upper limit

Query

SRangePulse[, p1]?

Example

Measure the pulse (level) on channel 0002.
Chattering filter is on. Math is on.
SRangePulse, 0002, PulseInput, LevelRange, Level, ContactRange, Contact, On, Off, 0, 10000, 0, 200, “m3/min”

Description

• You cannot use this command to configure settings while recording is in progress.
• You cannot use this command to configure settings while computation is in progress.
### SRangeDI

**Measurement Range of DI Channel**
Sets the measurement range of a DI channel.

**Unused Channels**

Syntax: `SRangeDI,p1,p2`
- **p1**: Channel number
- **p2**: Input type (Skip)

**Channels That Are Not Delta, Scaling, Pulse Input**

Syntax: `SRangeDI,p1,p2,p3,p4,p5,p6`
- **p1**: Channel number
- **p2**: Input type (DI)
- **p3**: Fixed at "."
- **p4**: Calculation type (Off)
- **p5**: Span lower limit (0 to 1)
- **p6**: Span upper limit (0 to 1)

**Delta Channels**

Syntax: `SRangeDI,p1,p2,p3,p4,p5,p6,p7`
- **p1**: Channel number
- **p2**: Input type (DI)
- **p3**: Fixed at "."
- **p4**: Calculation type (Delta)
- **p5**: Span lower limit (0 to 1)
- **p6**: Span upper limit (0 to 1)
- **p7**: Reference channel number

**Scaling Channels**

Syntax: `SRangeDI,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10`
- **p1**: Channel number
- **p2**: Input type (DI)
- **p3**: Fixed at "."
- **p4**: Calculation type (Scale)
- **p5**: Span lower limit (0 to 1)
- **p6**: Span upper limit (0 to 1)
- **p7**: Decimal Place (0 to 5)
- **p8**: Scaling lower limit
- **p9**: Scaling upper limit
- **p10**: Unit (up to 6 characters, UTF-8)

**Pulse Input Channels**

Syntax: `SRangeDI,p1,p2,p3,p4,p5,p6`
- **p1**: Channel number
- **p2**: Input type (Pulse)
- **p3**: Fixed at "."
- **p4**: Math type (Off)
- **p5**: Span lower limit (0 to 999999)
- **p6**: Span upper limit (0 to 999999)

**Query**

```
SRangeDI[,p1]?
```

**Example**

Measure 0 to 1 on channel 0103. No scaling.
`SRangeDI,0103,DI,\~,Off,0,1`

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- `p2=Pulse` can be specified when the math function (/MT) is installed.
- If `p2=Pulse`, `p4=Delta` or `Scale` cannot be specified.
- `p2=Pulse` cannot be specified when the operation mode of the DI module is set to Remote.
- You cannot use this command to configure settings when the measurement mode is set to high speed.

### SRangeDO

**DO Channel Operation**
Sets the DO channel operation.

**Alarm Output**

Syntax: `SRangeDO,p1,p2,p3,p4,p5,p6,p7,p8,p9`
- **p1**: Channel number
- **p2**: Output type (Alarm)
- **p3**: Span lower limit (0 to 1)
- **p4**: Span upper limit (0 to 1)
- **p5**: Unit (up to 6 characters, UTF-8)
- **p6**: Energize or de-energize
  - **Energize**: Energize the relay (DO channel) during output.
  - **De_Energize**: De-energize the relay (DO channel) during output.
- **p7**: Operation
  - **And**: Operate when all set alarms are in the alarm state.
  - **Or**: Operate when any of the set alarms are in the alarm state.
- **p8**: Hold or nonhold
  - **Hold**: Hold output until an alarm ACK operation.
  - **Nonhold**: Clear output when the alarm is cleared.
- **p9**: Relay (DO channel) action on acknowledge (Normal, Reset)

**Alarm Output (Reflash)**

Syntax: `SRangeDO,p1,p2,p3,p4,p5,p6,p7,p8,p9`
- **p1**: Channel number
- **p2**: Output type (Alarm)
- **p3**: Span lower limit (0 to 1)
- **p4**: Span upper limit (0 to 1)
- **p5**: Unit (up to 6 characters, UTF-8)
- **p6**: Energize or de-energize
  - **Energize**: Energize the relay (DO channel) during output.
  - **De_Energize**: De-energize the relay (DO channel) during output.
- **p7**: Action (Reflash)
- **p8**: Reflash time (500ms, 1s, 2s)
- **p9**: Relay (DO channel) action on acknowledge

**Manual Output**
Specifies the output value.
2.4 Setting Commands

Syntax: SRangeDO, p1, p2, p3, p4, p5, p6
- p1: Channel number
- p2: Output type (Manual)
- p3: Span lower limit (0 to 1)
- p4: Span upper limit (0 to 1)
- p5: Unit (up to 6 characters, UTF-8)
- p6: Energize or de-energize
  - Energize: Energize the relay (DO channel) during output.
  - De_Energize: De-energize the relay (DO channel) during output.

Fail Output (GM10 only)
Syntax: SRangeDO, p1, p2, p3, p4, p5, p6
- p1: Channel number
- p2: Output type (Fail)
- p3: Span lower limit (0 to 1)
- p4: Span upper limit (0 to 1)
- p5: Unit (up to 6 characters, UTF-8)
- p6: Fixed to De_Energize

Query: SRangeDO[, p1]?

Example: Output an alarm on channel 0203. Set the span lower limit to 0 and span upper limit to 1.
Set the operation on ACK to Normal. Set the unit to "Unit."
SRangeDO, 0203, Alarm, 0, 1, Unit, Energize, Or, Hold, Normal

Description
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- If p2=Manual, you cannot set p7 or subsequent parameters.
- If p7=And or Or, you cannot set the reflash time.
- If individual alarm ACK is enabled (SIndivAlmACK command), p9 is fixed to Reset.

[Boxed section: SMoveAve]

Moving Average
Sets the moving average of an AI or PI channel.
Syntax: SMoveAve, p1, p2, p3
- p1: Channel number
- p2: Enable or disable (Off, On)
- p3: Number of samples
  - Modules other than high speed AI: 2 to 100 (times)
  - High speed AI: 2 to 500 (times)

Query: SMoveAve[, p1]?

Example: Set the number of moving average samples for channel 0002 to 12.
SMoveAve, 0002, On, 12
### SFFilter

**Sets the First-Order Lag Filter**
Sets the first-order lag filter of a high-speed AI channel when a high-speed AI module is installed.

**Syntax**

```
SFFilter,p1,p2,p3
```

- `p1` Channel number
- `p2` Enable or disable (Off, On)
- `p3` First-order lag coefficient (3 to 300)

**Query**

```
SFFilter[,p1]?
```

**Example**

Set the first-order lag of channel 0002 to 100.

```
SFFilter,0002,On,100
```

**Description**

- This command is valid only for channels on high-speed AI type analog input modules.

### SBurnOut

**Behavior When a Sensor Burns Out**
Sets the behavior for when a burnout occurs on an AI channel.

**Syntax**

```
SBurnOut,p1,p2
```

- `p1` Channel number
- `p2` Burnout processing (Off, Up, Down)

**Query**

```
SBurnOut[,p1]?
```

**Example**

Set the measured result to positive overflow (Up) when a burnout is detected on channel 0001.

```
SBurnOut,0001,Up
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- This command is invalid for 4-wire RTD modules.

### SRjc

**Reference Junction Compensation Method**
Sets the reference junction compensation method of an AI channel.

**Syntax**

```
SRjc,p1,p2,p3
```

- `p1` Channel number
- `p2` Mode
  - Internal: Use the internal compensation function.
  - External: Use an external compensation device.
- `p3` Compensation temperature
  - 200 to 800: 20.0 to 80.0°C
  - -40 to 1760: -40 to 1760°F
  - 2531 to 3532: 253.1 to 353.2K

**Query**

```
SRjc[,p1]?
```

**Example**

Perform reference junction compensation of channel 0003 using the internal compensation circuit.

```
SRjc,0003,Internal
```

Perform reference junction compensation of channel 0004 using an external compensation device. Set the compensation temperature to -2.3°C.

```
SRjc,0004,External,-23
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- If p2=Internal, p3 is invalid.
- This command is invalid for 4-wire RTD modules.
2.4 Setting Commands

**SAlarmIO**

**Alarm**
Sets the alarm for an AI, DI, or PI channel.

**Do Not Set Alarms**

**Syntax**

\[ \text{SAlarmIO}, \text{p1}, \text{p2}, \text{p3} \]

- **p1**: Channel number
- **p2**: Alarm number (1 to 4)
- **p3**: Alarm on or off (Off)

**Do Not Output Alarms**

**Syntax**

\[ \text{SAlarmIO}, \text{p1}, \text{p2}, \text{p3}, \text{p4}, \text{p5}, \text{p6}, \text{p7} \]

- **p1**: Channel number
- **p2**: Alarm number (1 to 4)
- **p3**: Alarm on or off (On)
- **p4**: Alarm type (H, L, DH, DL, RH, RL, TH, TL)
  - For a channel set to Log scale (/LG) (if p4 of SRangeAI is LogT1, LogT2, or LogT3), p4 is H, L, TH, or TL.
- **p5**: Value
  - For a channel set to Log scale (/LG) (if p4 of SRangeAI is LogT1, LogT2, or LogT3), specify p5 using exponential notation (e.g. 1.23E10, where the number of digits of the mantissa is as specified by p8 of the SRangeAI command).

- **p6**: Detection (Off, On)
- **p7**: Output (Off)

**Output Alarms**

**Syntax**

\[ \text{SAlarmIO}, \text{p1}, \text{p2}, \text{p3}, \text{p4}, \text{p5}, \text{p6}, \text{p7}, \text{p8} \]

- **p1**: Channel number
- **p2**: Alarm number (1 to 4)
- **p3**: Alarm on or off (On)
- **p4**: Alarm type (H, L, DH, DL, RH, RL, TH, TL)
  - For a channel set to Log scale (/LG) (if p4 of SRangeAI is LogT1, LogT2, or LogT3), p4 is H, L, TH, or TL.

- **p5**: Value
  - For a channel set to Log scale (/LG) (if p4 of SRangeAI is LogT1, LogT2, or LogT3), specify p5 using exponential notation (e.g. 1.23E10, where the number of digits of the mantissa is as specified by p8 of the SRangeAI command).

- **p6**: Detection (Off, On)
- **p7**: Output (Off)
- **p8**: Number
  - If p7=DO Relay (DO channel) number
  - If p7=SW Internal switch number (001 to 100)

**Query**

\[ \text{SAlarmIO}, [\text{p1}, \text{p2}] ? \]

**Example**

Set a high limit alarm (H) on alarm number 2 of channel 0001. Set the alarm value to 1.8000V. Use the alarm detection function. When an alarm occurs, output to the relay (DO channel) at number 0205.

\[ \text{SAlarmIO}, 0001, 2, \text{On}, \text{H}, 18000, \text{On}, \text{DO}, 0205 \]

**Description**

- You cannot set this on a “Skip” channel.
- If p3=Off, you cannot set p4 or subsequent parameters.
- If p7=Off, you cannot set p8.
- For the alarm values of p5, use the values in the following table.

<table>
<thead>
<tr>
<th>Channel Type</th>
<th>Input Type</th>
<th>Calculation Type</th>
<th>Alarm Type</th>
<th>H, L, TH, TL</th>
<th>RH, RL, DH, DL</th>
</tr>
</thead>
<tbody>
<tr>
<td>AI channel</td>
<td>Volt, GS, TC, RTD</td>
<td>Delta Scale sqrt LogT1 LogT2 LogT3</td>
<td>H, L</td>
<td>H, L</td>
<td>DH, DL</td>
</tr>
<tr>
<td>DI channel</td>
<td>Pulse</td>
<td>Same as the DI input of AI channels</td>
<td>Off</td>
<td>Off</td>
<td>0 - 999999</td>
</tr>
<tr>
<td>PI channel</td>
<td>Pulse</td>
<td>Delta Scale</td>
<td>Off</td>
<td>Off</td>
<td>0 - 999999</td>
</tr>
</tbody>
</table>

(1) Within the measurement range
(2) –5% to 105% of the scale but within –999999 to 999999 excluding the decimal point
(3) 1 digit to (measurement upper limit – measurement lower limit)
(4) 1 digit to (scale upper limit – scale lower limit) but within 1 to 999999 excluding the decimal point
(5) Within the difference measurement range
(6) Log scale range that corresponds to -5% to 105% of the span

- You cannot set DO channels or internal switches whose output type is set to Manual as output destination numbers.
- You cannot set DI channels when the measurement mode is set to high speed.

**SAImHysIO**

**Alarm Hysteresis**
Sets the alarm hysteresis for an AI, DI, or PI channel.

**Syntax**

\[ \text{SAImHysIO}, \text{p1}, \text{p2}, \text{p3} \]

- **p1**: Channel number
- **p2**: Alarm number (1 to 4)
- **p3**: Hysteresis

**Alarm Type**

| H, L, DH, DL | 0.0% to 5.0% of the span or scale width
| Linear scaling, Square root | 0 to 100000
| Log scale (/LG) | Fixed to 0

**Example**

Set a high limit alarm (H) on alarm number 2 of channel 0001. Set the alarm value to 1.8000V. Use the alarm detection function. When an alarm occurs, output to the relay (DO channel) at number 0205.
### SAlmHysIO

**Commands and Responses**

**Description**
- Hysteresis specified for delay high and low limit alarms (TH and TL) and high and low limits on rate-of-change alarms (RH and RL) do not apply.
- When the input type of a DI channel is Pulse, hysteresis is fixed at 0.
- You cannot set DI channels when the measurement mode is set to high speed.

**Example**
Set a 0.5% hysteresis on alarm 3 of channel 0002.

```
SAlmHysIO,0002,3,5
```

- Query `SAlmHysIO[,]p1[,p2]??`

### SAlmDlyIO

**Alarm Delay Time**

Sets the delay alarm time for an AI, DI, or PI channel.

**Syntax**

```
SAlmDlyIO,p1,p2,p3,p4
```

- `p1` Channel number
- `p2` Hour (0 to 24)
- `p3` Minute (0 to 59)
- `p4` Second (0 to 59)

**Query**

```
SAlmDlyIO[,]p1??
```

**Example**
Set the channel 0001 alarm delay time to 2 minutes 30 seconds.

```
SAlmDlyIO,0001,0,2,30
```

**Description**
- Set the delay time so that it is an integer multiple of the scan interval (`SScan` command).
- You cannot set DI channels when the measurement mode is set to high speed.

### STagIO

**Tag**

Sets a tag to an AI, DI, PI, AO, DO, or PID channel.

**Syntax**

```
STagIO,p1,p2,p3
```

- `p1` Channel number
- `p2` Tag (up to 32 characters, UTF-8)
- `p3` Tag number (up to 16 characters, ASCII)

**Query**

```
STagIO[,]p1??
```

**Example**
Set the channel 0001 tag to “SYSTEM1” and the tag number to “TI002.”

```
STagIO,0001,’SYSTEM1’,’TI002’
```

**Description**
- You cannot set DI channels when the measurement mode is set to high speed.

### SColorIO

**Channel Color**

Sets the color of an AI, DI, PI, AO, DO, or PID channel.

**Syntax**

```
SColorIO,p1,p2,p3,p4
```

- `p1` Channel number
- `p2` R value of RGB display colors (0 to 255, see “Description.”)
- `p3` G value of RGB display colors (0 to 255, see “Description.”)
- `p4` B value of RGB display colors (0 to 255, see “Description.”)

**Query**

```
SColorIO[,]p1??
```

**Description**
- The RGB values for different colors are indicated in the following table.

<table>
<thead>
<tr>
<th>Color</th>
<th>R</th>
<th>G</th>
<th>B</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>255</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Green</td>
<td>0</td>
<td>153</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td>Blue</td>
<td>0</td>
<td>51</td>
<td>255</td>
<td></td>
</tr>
<tr>
<td>Blue violet</td>
<td>119</td>
<td>51</td>
<td>204</td>
<td>GM10</td>
</tr>
<tr>
<td>Brown</td>
<td>153</td>
<td>51</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Orange</td>
<td>255</td>
<td>153</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td>Yellow green</td>
<td>153</td>
<td>204</td>
<td>51</td>
<td>GM10</td>
</tr>
<tr>
<td>Light blue</td>
<td>119</td>
<td>170</td>
<td>221</td>
<td>GM10</td>
</tr>
<tr>
<td>Violet</td>
<td>204</td>
<td>102</td>
<td>204</td>
<td>GM10</td>
</tr>
<tr>
<td>Gray</td>
<td>153</td>
<td>153</td>
<td>153</td>
<td></td>
</tr>
<tr>
<td>Lime</td>
<td>102</td>
<td>255</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Cyan</td>
<td>0</td>
<td>255</td>
<td>255</td>
<td></td>
</tr>
<tr>
<td>Dark blue</td>
<td>0</td>
<td>0</td>
<td>153</td>
<td></td>
</tr>
<tr>
<td>Yellow</td>
<td>255</td>
<td>255</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Light gray</td>
<td>204</td>
<td>204</td>
<td>204</td>
<td></td>
</tr>
<tr>
<td>Purple</td>
<td>153</td>
<td>0</td>
<td>153</td>
<td>GM10</td>
</tr>
<tr>
<td>Black</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Pink</td>
<td>255</td>
<td>17</td>
<td>153</td>
<td></td>
</tr>
<tr>
<td>Rosy brown</td>
<td>204</td>
<td>153</td>
<td>153</td>
<td></td>
</tr>
<tr>
<td>Pale green</td>
<td>153</td>
<td>255</td>
<td>153</td>
<td>GM10</td>
</tr>
<tr>
<td>Dark gray</td>
<td>102</td>
<td>102</td>
<td>102</td>
<td></td>
</tr>
<tr>
<td>Olive</td>
<td>153</td>
<td>153</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Dark cyan</td>
<td>0</td>
<td>153</td>
<td>153</td>
<td></td>
</tr>
<tr>
<td>Spring green</td>
<td>0</td>
<td>204</td>
<td>153</td>
<td>GM10</td>
</tr>
</tbody>
</table>

- You cannot set DI channels when the measurement mode is set to high speed.
2.4 Setting Commands

### SZoneIO

**Waveform Display Zone**
Sets the waveform display zone of an AI, DI, PI, AO, DO, or PID channel.

**Syntax**

```
SZoneIO, p1, p2, p3
```

- `p1`: Channel number
- `p2`: Zone lower limit [%] (0 to 95)
- `p3`: Zone upper limit [%] (5 to 100)

**Query**

```
SZoneIO[,p1]
```

**Example**

Set the waveform zone of channel 0001 waveform to 0% to 30%.

```
SZoneIO, 0001, 0, 30
```

**Description**

- You cannot set DI channels when the measurement mode is set to high speed.

### SScaleIO

**Scale Display [GX/GP]**
Sets the scale display of an AI, DI, PI, AO, DO, or PID channel.

**Syntax**

```
SScaleIO, p1, p2, p3
```

- `p1`: Channel number
- `p2`: Scale display position (Off, 1 to 10)
- `p3`: Number of scale divisions (4 to 12, C10)

**Query**

```
SScaleIO[,p1]
```

**Example**

Display the channel 0001 scale at display position 1. Display four equally spaced main scale marks.

```
SScaleIO, 0001, 1, 4
```

**Description**

- You cannot set DI channels when the measurement mode is set to high speed.

### SBarIO

**Bar Graph Display**
Sets the bar graph display of an AI, DI, PI, AO, DO, or PID channel.

**Syntax**

```
SBarIO, p1, p2, p3
```

- `p1`: Channel number
- `p2`: Bar display base position
  - Lower
  - Center
  - Upper
- `p3`: Number of scale divisions (4 to 12)

**Query**

```
SBarIO[,p1]
```

**Example**

Display the measured values of channel 0001 on a bar graph with the center set as the base position (Center). Display four equally spaced main scale marks.

```
SBarIO, 0001, Center, 4
```

**Description**

- You cannot set DI channels when the measurement mode is set to high speed.

### SPartialIO

**Partial Expanded Display [GX/GP]**
Sets the partial-expansion display of an AI, PI, or PID channel waveform.

**Syntax**

```
SPartialIO, p1, p2, p3, p4
```

- `p1`: Channel number
- `p2`: Partial expanded On/Off (On, Off)
- `p3`: Partial expanded boundary position [%] (1 to 99)
- `p4`: Partial expanded boundary value (span lower limit + 1 digit to span upper limit - 1 digit)

**Query**

```
SPartialIO[,p1]
```

**Example**

For channel 0001 whose measurement range is 0 to 1.0000 V, display the measured value of 0.7500 V at the 50% position.

```
SPartialIO, 0001, On, 50, 7500
```

**Description**

- You cannot set this on a “Skip” channel. p2 is fixed to Off.
- You cannot set this on a channel set to Log scale (/LG) (if p4 of SRangeAI is LogT1, LogT2, or LogT3). p2 is fixed to Off.
- If p2=Off, you cannot set p3 or subsequent parameters.
- P2=On can be specified when the difference between the span upper and lower limits is 2 digits or greater.

### SBandIO

**Color Scale Band**
Sets the color scale band of an AI or PI channel.

**Syntax**

```
SBandIO, p1, p2, p3, p4, p5, p6, p7
```

- `p1`: Channel number
- `p2`: Color scale band (Off, In, Out)
- `p3`: R value of the color scale band RGB colors (0 to 255)
- `p4`: G value of the color scale band RGB colors (0 to 255)
- `p5`: B value of the color scale band RGB colors (0 to 255)
- `p6`: Upper limit of the color scale band display (Span or scale lower limit to span or scale upper limit)
  - For a channel set to Log scale (/LG) (if p4 of SRangeAI is LogT1, LogT2, or LogT3), specify p6 using exponential notation (e.g. 1.23E10, where the number of digits of the mantissa is as specified by p8 of the SRangeAI command).
- `p7`: Lower limit of the color scale band display (Span or scale lower limit to span or scale upper limit)
  - For a channel set to Log scale (/LG) (if p4 of SRangeAI is LogT1, LogT2, or LogT3), specify p7 using exponential notation (e.g. 1.23E10, where the number of digits of the mantissa is as specified by p8 of the SRangeAI command).
### Commands and Responses

#### 2.4 Setting Commands

**SAImMarkIO**

**Alarm Mark**

Sets the display of the marker that indicates the specified alarm position of an AI, DI, or PI channel.

**Syntax**

```
SAImMarkIO,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14,p15
```

- **p1**: Channel number
- **p2**: Whether to display the alarm mark on the scale (Off, On)
- **p3**: Alarm mark type
  - Alarm: Display the default alarm mark
  - Fixed: Display the mark with the specified color
- **p4**: R value of the RGB mark colors for alarm 1 (0 to 255)
- **p5**: G value of the RGB mark colors for alarm 1 (0 to 255)
- **p6**: B value of the RGB mark colors for alarm 1 (0 to 255)
- **p7**: R value of the RGB mark colors for alarm 2 (0 to 255)
- **p8**: G value of the RGB mark colors for alarm 2 (0 to 255)
- **p9**: B value of the RGB mark colors for alarm 2 (0 to 255)
- **p10**: R value of the RGB mark colors for alarm 3 (0 to 255)
- **p11**: G value of the RGB mark colors for alarm 3 (0 to 255)
- **p12**: B value of the RGB mark colors for alarm 3 (0 to 255)
- **p13**: R value of the RGB mark colors for alarm 4 (0 to 255)
- **p14**: G value of the RGB mark colors for alarm 4 (0 to 255)
- **p15**: B value of the RGB mark colors for alarm 4 (0 to 255)

**Query**

```
SAImMarkIO[,p1]
```

**Example**

For channel 0001, set the lower limit to “OFF” and the upper limit to “ON.”

```
SAImMarkIO,0001,'OFF','ON'
```

**Description**

- For details on RGB values, see “Description” of the SColorIO command.
- You cannot set DI channels when the measurement mode is set to high speed.

### SValueIO

**Upper/Lower Limit Display Characters**

Sets the upper/lower limit display characters of AI, DI, DO, or PID channel.

**Syntax**

```
SValueIO,p1,p2,p3
```

- **p1**: Channel number
- **p2**: Lower limit display string (up to 8 characters, UTF-8)
- **p3**: Upper limit display string (up to 8 characters, UTF-8)

**Query**

```
SValueIO[,p1]
```

**Example**

For channel 0001, set a blue band in the range of -0.5000 to 1.0000.

```
SBandIO,0001,in,0,0,255,5000,10000
```

**Example**

For channel 0001, set the following parameters:

```
SBandIO,0001,In,0,0,255,5000,10000
```

**Alarm Mark**

Sets the display of the marker that indicates the specified alarm position of an AI, DI, or PI channel.

**Syntax**

```
SAImMarkIO,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14,p15
```

- **p1**: Channel number
- **p2**: Whether to display the alarm mark on the scale (Off, On)
- **p3**: Alarm mark type
  - Alarm: Display the default alarm mark
  - Fixed: Display the mark with the specified color
- **p4**: R value of the RGB mark colors for alarm 1 (0 to 255)
- **p5**: G value of the RGB mark colors for alarm 1 (0 to 255)
- **p6**: B value of the RGB mark colors for alarm 1 (0 to 255)
- **p7**: R value of the RGB mark colors for alarm 2 (0 to 255)
- **p8**: G value of the RGB mark colors for alarm 2 (0 to 255)
- **p9**: B value of the RGB mark colors for alarm 2 (0 to 255)
- **p10**: R value of the RGB mark colors for alarm 3 (0 to 255)
- **p11**: G value of the RGB mark colors for alarm 3 (0 to 255)
- **p12**: B value of the RGB mark colors for alarm 3 (0 to 255)
- **p13**: R value of the RGB mark colors for alarm 4 (0 to 255)
- **p14**: G value of the RGB mark colors for alarm 4 (0 to 255)
- **p15**: B value of the RGB mark colors for alarm 4 (0 to 255)

**Query**

```
SAImMarkIO[,p1]
```

**Example**

Display the alarm marks for alarms 1 to 4 of channel 0001 in fixed colors red, brown, orange, and yellow, respectively.

```
SAImMarkIO,0001,On,Fixed,255,0,0,165,42,42,255,165,0,255,255,0
```

**Description**

- You cannot set DI channels when the measurement mode is set to high speed.
2.4 Setting Commands

**SCalibIO**

**Calibration Correction**
Sets the calibration correction for AI channels.

**Disable Calibration Correction**

**Syntax**

SCalibIO,p1,p2

- **p1**: Channel number
- **p2**: Linearizer mode (Off)

**Use Calibration Correction (Linearizer approximation, linearizer bias)**

**Syntax**

SCalibIO,p1

- **p1**: Channel number

**Use Calibration Correction (Correction coefficient) (/AH)**

**Syntax**


- **p1**: Channel number
- **p2**: Mode
  - Correct: Correction coefficient
- **p3**: Number of correction points (2 to 12)
- **p4**: Input value of uncorrected value 1
- **p5**: Instrument correction coefficient 1
- **p6**: Sensor correction coefficient 1
- **p7**: Input value of uncorrected value 2
- **p8**: Instrument correction coefficient 2
- **p9**: Sensor correction coefficient 2
- **p10**: Input value of uncorrected value 3
- **p11**: Instrument correction coefficient 3
- **p12**: Sensor correction coefficient 3
- **p13**: Input value of uncorrected value 4
- **p14**: Instrument correction coefficient 4
- **p15**: Sensor correction coefficient 4

**Query**

SCalibIO[,p1]?

**Example**

Set three set points on channel 0001 (measurement range: 0 to 1.0000 V). Set the set points as follows: when the input value is 0 V, the output value is 0.0010 V; when the input value is 0.5000 V, the output value is 0.5020 V; when the input value is 1.0000 V, the output value is 0.9970 V.

SCalibIO,0001,Appro,3,0,10,5000,5020,10000,9970

**Description**

- If p2=Off, you cannot set p3 or subsequent parameters.
- You cannot specify set points beyond the number of points specified by p3.
- If the AI channel input type (p2 of SRangeAI) is set to Skip or DI, you cannot specify anything other than p2=Off.
### SPresetAO

**Sets the Preset Action**

Sets the preset action of an AO channel.

#### Syntax

```
SPresetAO,p1,p2,p3,p4
```

- **p1**: Channel number
- **p2**: Action at power-on (Last, Preset)
  - **Last**: Hold previous value
  - **Preset**: Output preset value
- **p3**: Action on error (Last, Preset)
  - **Last**: Hold previous value
  - **Preset**: Output preset value
- **p4**: Action on stop (Last, Preset)
  - **Last**: Hold previous value
  - **Preset**: Output preset value

#### Query

```
SPresetAO[,p1]?
```

#### Example

At power-on, replace the channel 0001 value with the preset value of 0.5.

```
SPresetAO,0001,Preset,0.5
```

#### Description

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- “Action on stop” indicates a scanning stopped, math stopped, or re-transmission off condition.

### SMathBasic

**Math Action (/MT)**

Sets the basic operation of math channels.

#### Syntax

- **GX/GP**
  
  ```
  SMathBasic,p1,p2,p3,p4,p5
  ```

- **GM**
  
  ```
  SMathBasic,p1,p2,p3,p4,p5,p6,p7
  ```

- **p1**: Indication on computation error
  - **+Over**: Display the computed value as +Over.
  - **-Over**: Display the computed value as -Over.

- **p2**: SUM and AVE computation when overflow data is detected
  - **Error**: Sets the computation result to computation error.
  - **Skip**: Discards the data that overflowed and continues the computation.
  - **Limit**: Computes by substituting upper or lower limit values in the data that overflowed.
  - For channels that do not have linear scaling specified, the upper or lower limit of the measuring range
  - For channels that have linear scaling specified, the scaling upper or lower limit

- **p3**: MAX, MIN, and P-P computation when overflow data is detected
  - **Over**: Computes using data that overflowed.
  - **Skip**: Discards the data that overflowed and continues the computation.

- **p4**: START/STOP key action
  - **(GX/GP: Off, Start/Stop, Reset+Start/Stop)**
  - **(GM: Off)**
  - **Off**: Computation does not start even when recording starts.
  - **Start/Stop**: Computation starts when recording starts.
  - **Reset+Start**: Computation resets and starts when recording starts.

- **p5**: PSUM over operation (GX/GP)
  - **Rotate**: Rotate
  - **Over**: Over

- **p6**: PSUM over operation (GM)
  - **Rotate**: Rotate
  - **Over**: Over

- **p7**: STOP key action (Off, Stop)
  - **Off**: Recording stops but not computation.
  - **Stop**: Computation stops when recording stops.

#### Query

```
SMathBasic?
```

#### Example

Set the indication on computation error to “+Over,” computation when overflow data is detected to “Skip,” and start computation when recording starts.

```
SMathBasic,+Over,Skip,Skip,Start/Stop
```

#### Description

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- p5 and p6 are invalid parameters for the GX/GP.
- When the multi batch function (/BT) is enabled, p4 is fixed to Off.
### SKConst

**Constant (/MT)**
Sets a constant for use in computations.

**Syntax**
```
SKConst,p1,p2
```
- `p1`: Constant number (1 to 100)
- `p2`: Value (–9.9999999E+29 to –1E–30, 0, 1E–30 to 9.9999999E+29, eight significant digits)

**Query**
```
SKConst[,p1]?
```

**Example**
Set constant number 12 to 1.0000E–10.
```
SKConst,12,1.0000E-10
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

### SRangeMath

**Computation Expression (/MT)**
Sets the computation expression of a math channel.

**Syntax**
```
SRangeMath,p1,p2,p3,p4,p5,p6,p7,p8
```
- `p1`: Channel number
- `p2`: Computation expression on/off (Off)
- `p3`: Math channel type (Normal)
- `p4`: Expression (up to 120 characters, ASCII)
- `p5`: Decimal Place (0 to 5)
- `p6`: Span lower limit (–9999999 to 99999999)
- `p7`: Span upper limit (–9999999 to 99999999)
- `p8`: Unit (up to 6 characters, UTF-8)

**Query**
```
SRangeMath[,p1]?
```

**Example**
Assign timer 2 to math channel 015. Set the sum scale to Off and disable reset.
```
SRangeMath,015,On,Normal,0001+0002,1,0,1000,'%'
```

**Description**
- You cannot use this command to configure settings while computation is in progress.
- You cannot set the span upper and lower limits to the same value.

### STlogMath

**TLOG (/MT)**
Sets the TLOG of a math channel.

**Syntax**
```
STlogMath,p1,p2,p3,p4,p5
```
- `p1`: Channel number
- `p2`: Timer Type
- `p3`: Timer number (1 to 12)
- `p4`: Sum scale (Off, /sec, /min, /hour)
- `p5`: Reset (On, Off)

**Query**
```
STlogMath[,p1]?
```

**Example**
```
STlogMath,015,Timer,2,Off,Off
```

**Description**
- You cannot use this command to configure settings while computation is in progress.

### SRolAveMath

**Rolling Average (/MT)**
Sets rolling average on a math channel.

**Syntax**
```
SRolAveMath,p1,p2,p3,p4
```
- `p1`: Channel number
- `p2`: Enable or disable (Off, On)
- `p3`: Sample interval (1 to 6s, 10s, 12s, 15s, 20s, 30s, 1 to 6min, 10min, 12min, 15min, 20min, 30min, 1h)
- `p4`: Number of samples (1 to 1500)

**Query**
```
SRolAveMath[,p1]?
```

**Example**
```
SRolAveMath,015,On,1min,30
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
2.4 Setting Commands

SAAlarmMath

Alarm (/MT)
Sets the alarm of a math channel.

Do Not Set Alarms
Syntax
```
SAAlarmMath, p1, p2, p3
```
p1 Channel number
p2 Alarm number (1 to 4)
p3 Alarm on or off (Off)

Do Not Output Alarms
Syntax
```
SAAlarmMath, p1, p2, p3, p4, p5, p6, p7
```
p1 Channel number
p2 Alarm number (1 to 4)
p3 Alarm on or off (Off)
p4 Alarm type (H, L, TH, TL)
p5 Alarm value (within the span range)
p6 Detection (Off, On)
p7 Output (Off)

Output Alarms
Syntax
```
SAAlarmMath, p1, p2, p3, p4, p5, p6, p7, p8
```
p1 Channel number
p2 Alarm number (1 to 4)
p3 Alarm on or off (On)
p4 Alarm type (H, L, TH, TL)
p5 Alarm value (within the span range)
p6 Detection (Off, On)
p7 Output (DO, SW)
p8 Number

If p7=DO Relay (DO channel) number
If p7=SW Internal switch number (001 to 100)

Channel Using Logarithmic Math
Syntax
```
SAAlarmMath, p1, p2, p3, p4, p5, p6, p7, p8, p9
```
p1 Channel number
p2 Alarm number (1 to 4)
p3 Alarm on or off (Off)
p4 Alarm type (Off, H, L, TH, TL)
p5 Alarm value exponent (100 to 999)
p6 Alarm value mantissa (-16 to 16)
p7 Detection (Off, On)
p8 Output (OFF, DO, SW)
p9 Number

If p8=DO Relay (DO channel) number
If p8=SW Internal switch number (001 to 100)

Query
```
SAAlarmMath, [p1], [p2]
```

Example
Set a high limit alarm (H) on alarm number 2 of math channel 015. Set the alarm value to 85.0. When an alarm occurs, output to the relay (DO channel) at number 0105.
```
SAAlarmMath, 015, 2, On, H, 850, On, DO, 0105
```

Description
- You cannot set this on a “Off” channel.
- If p3=Off, you cannot set p4 or subsequent parameters.
- If p7 (p8 for logarithmic math) = Off, you cannot set p8 (p9 for logarithmic math).
- You cannot set DO channels or internal switches whose output type is set to Manual as output destination numbers.

SAAlmHysMath

Alarm Hysteresis (/MT)
Sets the alarm hysteresis for a math channel.

Syntax
```
SAAlmHysMath, p1, p2, p3
```
p1 Channel number
p2 Alarm number (1 to 4)
p3 Hysteresis

Alarm Type Hysteresis Range
H, L 0 to 100000

Channel Using Logarithmic Math
Syntax
```
SAAlmHysMath, p1, p2, p3, p4
```
p1 Channel number
p2 Alarm number (1 to 4)
p3 Hysteresis
p4 Hysteresis exponent (100 to 999)
p5 Hysteresis mantissa (-16 to 16)

Query
```
SAAlmHysMath, [p1], [p2]
```

Example
Set a hysteresis on alarm 3 of math channel 015.
```
SAAlmHysMath, 015, 3, 10
```

Description
- Hysteresis specified for delay high and low limit alarms (TH and TL) does not apply.

SAAlmDlyMath

Alarm Delay Time (/MT)
Sets the alarm delay time for a math channel.

Syntax
```
SAAlmDlyMath, p1, p2, p3, p4
```
p1 Channel number
p2 Hour (0 to 24)
p3 Minute (0 to 59)
p4 Second (0 to 59)

Query
```
SAAlmDlyMath, [p1]
```

Example
Set the math channel 015 alarm delay time to 2 minutes 30 seconds.
```
SAAlmDlyMath, 015, 0, 2, 30
```

Description
- Set the delay time so that it is an integer multiple of the scan interval (SScan command).
2.4 Setting Commands

**STagMath**

Tag (/MT)
Sets the tag of a math channel.

**Syntax**

STagMath,p1,p2,p3

- **p1** Channel number
- **p2** Tag (up to 32 characters, UTF-8)
- **p3** Tag number (up to 16 characters, UTF-8)

**Query**

STagMath[,p1]?

**Example**

Set the math channel 015 tag to “SYSTEM1” and the tag number to “TI002.”

STagMath,015,'SYSTEM1','TI002'

**SColorMath**

Channel Color (/MT)
Sets the color of a math channel.

**Syntax**

SColorMath,p1,p2,p3,p4

- **p1** Channel number
- **p2** R value of RGB display colors (0 to 255)
- **p3** G value of RGB display colors (0 to 255)
- **p4** B value of RGB display colors (0 to 255)

**Query**

SColorMath[,p1]?

**Example**

Set the math channel 015 display color to red.

SColorMath,015,255,0,0

**Description**

- For details on RGB values, see “Description” of the SColorIO command.

**SZoneMath**

Waveform Display Zone (/MT)
Sets the waveform display zone of a math channel.

**Syntax**

SZoneMath,p1,p2,p3

- **p1** Channel number
- **p2** Zone lower limit [%] (0 to 95)
- **p3** Zone upper limit [%] (5 to 100)

**Query**

SZoneMath[,p1]?

**Example**

Set the waveform zone of math channel 015 waveform to 0% to 30%.

SZoneMath,015,0,30

**SScaleMath**

Scale Display (/MT) [GX/GP]
Sets the scale display of a math channel.

**Syntax**

SScaleMath,p1,p2,p3

- **p1** Channel number
- **p2** Scale display position (Off, 1 to 10)
- **p3** Number of scale divisions (4 to 12, C10)

**Query**

SScaleMath[,p1]?

**Example**

Display the math channel 015 scale at display position 1. Display four equally spaced main scale marks.

SScaleMath,015,1,4

**SPartialMath**

Partial Expanded Display (/MT) [GX/GP]
Sets the partial expanded display of a math channel waveform.

**Syntax**

SPartialMath,p1,p2,p3,p4

- **p1** Channel number
- **p2** Partial expanded On/Off (On, Off)
- **p3** Partial expanded boundary position [%] (1 to 99)
- **p4** Partial expanded boundary value

**Query**

SPartialMath[,p1]?

**Example**

For channel 015 whose measurement range is 0 to 1.0000 V, display the measured value of 0.7500 V at the 50% position.

SPartialMath,015,On,50,7500

**Description**

- You cannot set this on a “Off” channel. p2 is fixed to Off.
- If p2=Off, you cannot set p3 or subsequent parameters.
- P2=On can be specified when the difference between the span upper and lower limits is 2 digits or greater.
- You cannot use this command to configure settings on channels using logarithmic math.
### SBandMath

**Color Scale Band (/MT)**
Sets the color scale band of a math channel.

**Syntax**

```
SBandMath, p1, p2, p3, p4, p5, p6, p7
```

- **p1** Channel number
- **p2** Color scale band (Off, In, Out)
- **p3** R value of the color scale band RGB colors (0 to 255)
- **p4** G value of the color scale band RGB colors (0 to 255)
- **p5** B value of the color scale band RGB colors (0 to 255)
- **p6** Upper limit of the color scale band display (span lower limit to span upper limit)
- **p7** Lower limit of the color scale band display (span lower limit to span upper limit)

**Channel Using Logarithmic Math**

**Syntax**

```
SBandMath, p1, p2, p3, p4, p5, p6, p7, p8, p9
```

- **p1** Channel number
- **p2** Color scale band (Off, In, Out)
- **p3** R value of the color scale band RGB colors (0 to 255)
- **p4** G value of the color scale band RGB colors (0 to 255)
- **p5** B value of the color scale band RGB colors (0 to 255)
- **p6** Exponent of the lower limit of the color scale band display (span low limit to span high limit)
- **p7** Mantissa of the lower limit of the color scale band display (span low limit to span high limit)
- **p8** Exponent of the upper limit of the color scale band display (span low limit to span high limit)
- **p9** Mantissa of the upper limit of the color scale band display (span low limit to span high limit)

**Query**

```
SBandMath[, p1]?
```

**Example**

For math channel 015, set a blue band in the range of -0.5000 to 1.0000.

```
SBandMath, 015, In, 0, 0, 255, 5000, 10000
```

**Description**

- You cannot set this on a “Off” channel. p2 is fixed to Off.
- If p2=Off, you cannot set p3 or subsequent parameters.
- For details on RGB values, see “Description” of the **SColorIO** command.

### SAlmMarkMath

**Alarm Mark (/MT)**
Sets the display of the marker that indicates the specified alarm position of a math channel.

**Syntax**

```
SAlmMarkMath, p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12, p13, p14, p15
```

- **p1** Channel number
- **p2** Whether to display the alarm mark on the scale (Off, On)
- **p3** Alarm mark type
  - Alarm: Display the default alarm mark
  - Fixed: Display the mark with the specified color
- **p4** R value of the RGB mark colors for alarm 1 (0 to 255)
- **p5** G value of the RGB mark colors for alarm 1 (0 to 255)
- **p6** B value of the RGB mark colors for alarm 1 (0 to 255)
- **p7** R value of the RGB mark colors for alarm 2 (0 to 255)
- **p8** G value of the RGB mark colors for alarm 2 (0 to 255)
- **p9** B value of the RGB mark colors for alarm 2 (0 to 255)
- **p10** R value of the RGB mark colors for alarm 3 (0 to 255)
- **p11** G value of the RGB mark colors for alarm 3 (0 to 255)
- **p12** B value of the RGB mark colors for alarm 3 (0 to 255)
- **p13** R value of the RGB mark colors for alarm 4 (0 to 255)
- **p14** G value of the RGB mark colors for alarm 4 (0 to 255)
- **p15** B value of the RGB mark colors for alarm 4 (0 to 255)

**Query**

```
SAlmMarkMath[, p1]?
```

**Example**

Display the alarm marks for alarms 1 to 4 of math channel 015 in fixed colors red, brown, orange, and yellow, respectively.

```
SAlmMarkMath, 015, On, Fixed, 255, 0, 0, 165, 42, 255, 165, 0, 255, 255, 0
```

**Description**

- For details on RGB values, see “Description” of the **SColorIO** command.
### SRangeCom

**Measurement Range (MC)**
Sets the measurement range of a communication channel.

**Unused Channels**

**Syntax**

```
SRangeCom,p1,p2
```

- **p1**: Channel number
- **p2**: Enable or disable (Off)

**Used Channels**

**Syntax**

```
SRangeCom,p1,p2,p3,p4,p5,p6
```

- **p1**: Channel number
- **p2**: Enable or disable (On)
- **p3**: Decimal Place (0 to 5)
- **p4**: Span lower limit (-9999999 to 99999999)
- **p5**: Span upper limit (-9999999 to 99999999)
- **p6**: Unit (up to 6 characters, UTF-8)

**Query**

```
SRangeCom[,p1]?
```

**Example**

Measure 0.00 to 100.00% on communication channel 025.

```
SRangeCom,025,On,2,0,10000,'%'
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- If p2=Off, you cannot set p3 or subsequent parameters.
- You cannot set the span upper and lower limits to the same value.

---

### SWDCom

**Watchdog Timer (MC)**
Sets the watchdog timer of a communication channel.

**Channels That Do Not Use Watchdog Timers**

**Syntax**

```
SWDCom,p1,p2
```

- **p1**: Channel number
- **p2**: Watchdog timer usage (Off)

**Channels That Use Watchdog Timers**

**Syntax**

```
SWDCom,p1,p2,p3,p4
```

- **p1**: Channel number
- **p2**: Watchdog timer usage (On)
- **p3**: Watchdog timer (1 to 120) [s]
- **p4**: Value at timer expired (Preset, Last)

**Query**

```
SWDCom[,p1]?
```

**Example**

Set the watchdog timer of communication channel 025 to 60 seconds. Replace the communication channel 025 value with its preset value at watchdog timer expiration.

```
SWDCom,025,On,60,Preset
```

**Description**

- If p2=Off, you cannot set p3 or subsequent parameters.

---

### SValueCom

**Preset Operation (MC)**
Sets the preset operation of a communication channel.

**No Alarm Setting**

**Syntax**

```
SValueCom,p1,p2,p3
```

- **p1**: Channel number
- **p2**: Alarm number (1 to 4)
- **p3**: Alarm on or off (Off)

**Do Not Output Alarms**

**Syntax**

```
SValueCom,p1,p2,p3,p4,p5,p6,p7
```

- **p1**: Channel number
- **p2**: Alarm number (1 to 4)
- **p3**: Alarm on or off (On)
- **p4**: Alarm type (H, L, TH, TL)
- **p5**: Alarm value (within the span range)
- **p6**: Detection (Off, On)
- **p7**: Output (Off)

**Output Alarms**

**Syntax**

```
SValueCom,p1,p2,p3,p4,p5,p6,p7,p8
```

- **p1**: Channel number
- **p2**: Alarm number (1 to 4)
- **p3**: Alarm on or off (On)
- **p4**: Alarm type (H, L, TH, TL)
- **p5**: Alarm value (within the span range)
- **p6**: Detection (Off, On)
- **p7**: Output (Off)
- **p8**: Output to a relay (DO channel)

**Example**

At power-on, replace the communication channel 025 value with the preset value of 0.5.

```
SValueCom,025,Preset,0.5
```
### SAlarmCom

**Tag (/MC)**

Sets the tag of a communication channel.

**Syntax**

```
SAlarmCom[,p1[,p2]]
```

- **p1**: Channel number
- **p2**: Tag (up to 32 characters, UTF-8)
- **p3**: Tag number (up to 16 characters, ASCII)

**Query**

```
SAlarmCom[,p1]? P
```

**Example**

Set the communication channel 025 tag to “SYSTEM1” and the tag number to “TI002.”

```
SAlarmCom,025,'SYSTEM1','TI002'
```

### SColorCom

**Channel Color (/MC)**

Sets the color of a communication channel.

**Syntax**

```
SColorCom,p1,p2,p3,p4
```

- **p1**: Channel number
- **p2**: R value of RGB display colors (0 to 255)
- **p3**: G value of RGB display colors (0 to 255)
- **p4**: B value of RGB display colors (0 to 255)

**Query**

```
SColorCom[,p1]?
```

**Example**

Set the communication channel 025 display color to red.

```
SColorCom,025,255,0,0
```

### SZoneCom

**Waveform Display Zone (/MC)**

Sets the waveform display zone of a communication channel.

**Syntax**

```
SZoneCom,p1,p2,p3
```

- **p1**: Channel number
- **p2**: Zone lower limit [%] (0 to 95)
- **p3**: Zone upper limit [%] (5 to 100)

**Query**

```
SZoneCom[,p1]?
```

**Example**

Set the waveform zone of communication channel 025 waveform to 0% to 30%.

```
SZoneCom,025,0,30
```

### SScaleCom

**Scale Display (/MC) [GX/GP]**

Sets the scale display of a communication channel.

**Syntax**

```
SScaleCom,p1,p2,p3
```

- **p1**: Channel number
- **p2**: Scale display position (Off, 1 to 10)
- **p3**: Number of scale divisions (4 to 12, C10)

**Query**

```
SScaleCom[,p1]?
```

**Example**

Display the communication channel 025 scale at display position 1. Display four equally spaced main scale marks.

```
SScaleCom,025,1,4
```
### SBarCom

**Bar Graph Display (/MC)**

Sets the bar graph display of a communication channel.

**Syntax**

```plaintext
SBarCom,p1,p2,p3
```

- **p1**: Channel number
- **p2**: Bar display base position
  - Lower
  - Center
  - Upper
- **p3**: Number of scale divisions (4 to 12)

**Query**

```plaintext
SBarCom,[p1]?
```

**Example**

Display the values of communication channel 025 on a bar graph with the center set as the base position (Center). Display four equally spaced main scale marks.

```
SBarCom,025,Center,4
```

### SPartialCom

**Partial Expanded Display (/MC) [GX/GP]**

Sets the partial expanded display of a communication channel waveform.

**Syntax**

```plaintext
SPartialCom,p1,p2,p3,p4
```

- **p1**: Channel number
- **p2**: Partial expanded On/Off (On, Off)
- **p3**: Partial expanded boundary position [%] (1 to 99)
- **p4**: Partial expanded boundary value

**Query**

```plaintext
SPartialCom,[p1]?
```

**Example**

For channel 025 whose measurement range is 0 to 1.0000 V, display the measured value of 0.7500 V at the 50% position.

```
SPartialCom,025,On,50,7500
```

### SBandCom

**Color Scale Band (/MC)**

Sets the color scale band of a communication channel.

**Syntax**

```plaintext
SBandCom,p1,p2,p3,p4,p5,p6,p7
```

- **p1**: Channel number
- **p2**: Color scale band (Off, In, Out)
- **p3**: R value of the color scale band RGB colors (0 to 255)
- **p4**: G value of the color scale band RGB colors (0 to 255)
- **p5**: B value of the color scale band RGB colors (0 to 255)
- **p6**: Upper limit of the color scale band display (span lower limit to span upper limit)
- **p7**: Lower limit of the color scale band display (span lower limit to span upper limit)

**Query**

```plaintext
SBandCom,[p1]?
```

**Example**

For communication channel 025, set a blue band in the range of -0.5000 to 1.0000.

```
SBandCom,025,In,0,0,255,5000,10000
```

### SAlmMarkCom

**Alarm Mark (/MC)**

Sets the display of the marker that indicates the specified alarm position of a communication channel.

**Syntax**

```plaintext
SAlmMarkCom,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14,p15
```

- **p1**: Channel number
- **p2**: Whether to display the alarm mark on the scale (Off, On)
- **p3**: Alarm mark type
  - Alarm: Display the default alarm mark
  - Fixed: Display the mark with the specified color
- **p4**: R value of the RGB mark colors for alarm 1 (0 to 255)
- **p5**: G value of the RGB mark colors for alarm 1 (0 to 255)
- **p6**: B value of the RGB mark colors for alarm 1 (0 to 255)
- **p7**: R value of the RGB mark colors for alarm 2 (0 to 255)
- **p8**: G value of the RGB mark colors for alarm 2 (0 to 255)
- **p9**: B value of the RGB mark colors for alarm 2 (0 to 255)
- **p10**: R value of the RGB mark colors for alarm 3 (0 to 255)
- **p11**: G value of the RGB mark colors for alarm 3 (0 to 255)
- **p12**: B value of the RGB mark colors for alarm 3 (0 to 255)
- **p13**: R value of the RGB mark colors for alarm 4 (0 to 255)
- **p14**: G value of the RGB mark colors for alarm 4 (0 to 255)
- **p15**: B value of the RGB mark colors for alarm 4 (0 to 255)

**Query**

```plaintext
SAlmMarkCom,[p1]?
```

**Example**

Display the alarm marks for alarms 1 to 4 of communication channel 025 in fixed colors red, brown, orange, and yellow, respectively.

```
SAlmMarkCom,025,On,Fixed,255,0,0,165,42,42,255,165,0,255,255,0
```

### Description

- You cannot set this on a “Off” channel. p2 is fixed to Off.
- If p2=Off, you cannot set p3 or subsequent parameters.
- For details on RGB values, see “Description” of the SColorIO command.
### `SCalibUseCom`

**Calibration Correction Use/Not**

Sets whether to use Calibration Correction of communication channels.

**Syntax**

```
SCalibUseCom,p1,p2
```

- `p1`: Channel number
- `p2`: Use/Not
  - Off
  - Not Use
  - On
  - Use

**Query**

```
SCalibUseCom[,p1]?
```

**Example**

Set the Calibration Correction of channel number 001 to use.
```
SCalibUseCom,001,On
```

**Description**

- There is a limitation on the number of channels that `p2` can be set to On.
- If `p2=Off` in the communication channel on/off setting (SRangeCom), `p2` is fixed to Off.
- For communication channels, refer to 2.3.2 Parameter Notation and Range on page 2-12.
- You cannot use this command to configure settings while recording is in progress.

<table>
<thead>
<tr>
<th>Model</th>
<th>GX10</th>
<th>GP10</th>
<th>GX20-1</th>
<th>GP20-1</th>
<th>GM10-1</th>
<th>GX20-2</th>
<th>GP20-2</th>
<th>GM10-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of channels that can be set to On</td>
<td>50</td>
<td>150</td>
<td>300</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### `SCalibCom`

**Calibration Correction**

Sets the calibration correction for communication channels.

**Disable Calibration Correction**

**Syntax**

```
SCalibCom,p1,p2
```

- `p1`: Channel number
- `p2`: Linearizer mode (Off)

**Use Calibration Correction (Linearizer approximation, linearizer bias)**

**Syntax**

```
```

- `p1`: Channel number
- `p2`: Linearizer mode
  - Approx
  - Linearizer approximation
  - Bias
  - Linearizer bias
- `p3`: Number of set points (2 to 12)
- `p4`: Input value of set point 1
- `p5`: Output value of set point 1
- `p6`: Input value of set point 2
- `p7`: Output value of set point 2
- `p8`: Input value of set point 3
- `p9`: Output value of set point 3
- `p10`: Input value of set point 4
- `p11`: Output value of set point 4
- `p12`: Input value of set point 5
- `p13`: Output value of set point 5
- `p14`: Input value of set point 6
- `p15`: Output value of set point 6
- `p16`: Input value of set point 7
- `p17`: Output value of set point 7
- `p18`: Input value of set point 8
- `p19`: Output value of set point 8
- `p20`: Input value of set point 9
- `p21`: Output value of set point 9
- `p22`: Input value of set point 10
- `p23`: Output value of set point 10
- `p24`: Input value of set point 11
- `p25`: Output value of set point 11
- `p26`: Input value of set point 12
- `p27`: Output value of set point 12

**Use Calibration Correction (Correction coefficient) (f/AH)**

**Syntax**

```
```

- `p1`: Channel number
- `p2`: Mode
  - Correct
  - Correction coefficient
- `p3`: Number of correction points (2 to 12)
- `p4`: Input value of uncorrected value 1
- `p5`: Instrument correction coefficient 1
- `p6`: Sensor correction coefficient 1
- `p7`: Input value of uncorrected value 2
- `p8`: Instrument correction coefficient 2
- `p9`: Sensor correction coefficient 2
- `p10`: Input value of uncorrected value 3
- `p11`: Instrument correction coefficient 3
- `p12`: Sensor correction coefficient 3
- `p13`: Input value of uncorrected value 4
- `p14`: Instrument correction coefficient 4
- `p15`: Sensor correction coefficient 4
- `p16`: Input value of uncorrected value 5
- `p17`: Instrument correction coefficient 5
- `p18`: Sensor correction coefficient 5
- `p19`: Input value of uncorrected value 6
- `p20`: Instrument correction coefficient 6
- `p21`: Sensor correction coefficient 6
- `p22`: Input value of uncorrected value 7
- `p23`: Instrument correction coefficient 7
- `p24`: Sensor correction coefficient 7
- `p25`: Input value of uncorrected value 8
- `p26`: Instrument correction coefficient 8
- `p27`: Sensor correction coefficient 8
- `p28`: Input value of uncorrected value 9
- `p29`: Instrument correction coefficient 9
- `p30`: Sensor correction coefficient 9
- `p31`: Input value of uncorrected value 10
- `p32`: Instrument correction coefficient 10
- `p33`: Sensor correction coefficient 10
- `p34`: Input value of uncorrected value 11
- `p35`: Instrument correction coefficient 11
- `p36`: Sensor correction coefficient 11
- `p37`: Input value of uncorrected value 12
- `p38`: Instrument correction coefficient 12
- `p39`: Sensor correction coefficient 12

**Query**

```
SCalibIO[,p1]?
```
Example

Set three set points on communication channel 001 (measurement range: 0 to 100.0). Set the set points as follows: when the input value is 0.0, the output value is 0.1; when the input value is 50.0, the output value is 50.2; when the input value is 100.0, the output value is 99.7.

SCalibCom,001,Appro,3,0,1,50,502,1000,997

Description
• If p2=Off, you cannot set p3 or subsequent parameters.
• If calibration correction use on/off (p2 of the SCalibUseCom command) is set to Off, p2 is fixed to Off.
• You cannot specify set points beyond the number of points specified by p3.
• The correction value is not affected by the range span.
  It is valid in the range of -9999999 to 99999999.

SAlmLimit

Rate-of-Change Alarm Interval
Sets the rate-of-change interval of the rate-of-change alarm.

Syntax
SAlmLimit,p1,p2
  p1 Interval for the low limit on rate-of-change alarm
  1 to 32 Integer multiple of the scan interval
  p2 Interval for the high limit on rate-of-change alarm
  1 to 32 Integer multiple of the scan interval

Query
SAlmLimit?

Example
Set the intervals for the low limit on rate-of-change alarm and high limit on rate-of-change alarm to 10 times and 20 times the scan interval, respectively.
SAlmLimit,10,20

Description
• You cannot use this command to configure settings while recording is in progress.
• You cannot use this command to configure settings while computation is in progress.

SIndivAlmACK

Individual Alarm ACK
Enables or disables the individual alarm ACK function.

Syntax
SIndivAlmACK,p1
  p1 Enable or disable (Off, On)

Query
SIndivAlmACK?

Example
Enable the individual alarm ACK function.
SIndivAlmACK,On

Description
• You cannot use this command to configure settings while recording is in progress.
• You cannot use this command to configure settings while computation is in progress.
• If p2=Off, you cannot set p3 or subsequent parameters.
**SMatchTimer**

**Match Time Timer**
Sets a match time timer.

Do Not Use Match Time Timers

**Syntax**
```
SMatchTimer, p1, p2
```
- **p1** Match time timer number (1 to 12)
- **p2** Type (Off)

**Match Time Timer That Synchronizes Once a Year**

**Syntax**
```
SMatchTimer, p1, p2, p3, p4, p5, p6, p7
```
- **p1** Match time timer number (1 to 12)
- **p2** Type (Year)
- **p3** Start time: Month (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec)
- **p4** Start time: Day (1 to 31, depends on the month)
- **p5** Interval: Hours (HH) (00 to 23)
- **p6** Interval: Minutes (MM) (00 to 59)
- **p7** Timer action
  - Single: Single shot
  - Repeat: Repeat

**Match Time Timer That Synchronizes Once a Month**

**Syntax**
```
SMatchTimer, p1, p2, p3, p4, p5, p6
```
- **p1** Match time timer number (1 to 12)
- **p2** Type (Month)
- **p3** Start time: Day (1 to 28)
- **p4** Interval: Hours (HH) (00 to 23)
- **p5** Interval: Minutes (MM) (00 to 59)
- **p6** Timer action
  - Single: Single shot
  - Repeat: Repeat

**Match Time Timer That Synchronizes Once a Week**

**Syntax**
```
SMatchTimer, p1, p2, p3, p4, p5, p6
```
- **p1** Match time timer number (1 to 12)
- **p2** Type (Week)
- **p3** Start time: Day of week
  - Sun
  - Mon
  - Tue
  - Wed
  - Thu
  - Fri
  - Sat
- **p4** Interval: Hours (HH) (00 to 23)
- **p5** Interval: Minutes (MM) (00 to 59)
- **p6** Timer action
  - Single: Single shot
  - Repeat: Repeat

**Match Time Timer That Synchronizes Once a Day**

**Syntax**
```
SMatchTimer, p1, p2, p3, p4, p5
```
- **p1** Match time timer number (1 to 12)
- **p2** Type (Day)

- **p3** Interval: Hours (HH) (00 to 23)
- **p4** Interval: Minutes (MM) (00 to 59)
- **p5** Timer action
  - Single: Single shot
  - Repeat: Repeat

**Query**
```
SMatchTimer[, p1]?
```

**Example**
Sets match time timer number 2 to a timer that operates on 21 hours 30 minutes on April 17 every year.
```
SMatchTimer, 2, Year, Apr, 17, 21, 30, Re
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- If p2=Off, you cannot set p3 or subsequent parameters.

**SEventAct**

**Event Action**
Sets an event action.

**Syntax**
```
SEventAct, p1, p2, p3, p4, p5, p6, p7
```
- **p1** Event action number (1 to 50)
- **p2** Type (Off, On)
- **p3** Event type (see the table below)
- **p4** Source element number (see the table below)
- **p5** Event details (see the table below)
- **p6** Operation mode (see the table below)
- **p7** Action type (see the table below)
- **p8** Source element number (see the table below)
- **p9** Event detail 1 (see the table below)
- **p10** Action detail 2 (see the table below)
- **p11** Action detail 3 (see the table below)
### 2.4 Setting Commands

<table>
<thead>
<tr>
<th>p3</th>
<th>p4</th>
<th>p5</th>
<th>p6</th>
<th>Conditions</th>
<th>p7</th>
<th>p8</th>
<th>p9</th>
<th>p10</th>
<th>p11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event Type</td>
<td>Value</td>
<td>Source Element Number</td>
<td>Event details</td>
<td>Operation mode</td>
<td>Action Type</td>
<td>Value</td>
<td>Source Element Number</td>
<td>Action Detail 1</td>
<td>Action Detail 2</td>
</tr>
<tr>
<td>Internal Switch</td>
<td>SW</td>
<td>1 to 100 -</td>
<td>Rising, Falling, Both</td>
<td>Recording</td>
<td>Memory</td>
<td>-</td>
<td>Start, Stop</td>
<td>ALL, batch group number</td>
<td></td>
</tr>
<tr>
<td>Remote control input</td>
<td>DI</td>
<td>Channel number -</td>
<td>Rising, Falling, Both</td>
<td>Math (math channel)</td>
<td>Math</td>
<td>-</td>
<td>Start, Stop, Reset</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Relay (DO channel)</td>
<td>DO</td>
<td>Channel number -</td>
<td>Rising, Falling, Both</td>
<td>Display rate switch</td>
<td>RateChange</td>
<td>1, 2</td>
<td>Reset</td>
<td>-</td>
<td>ALL, batch group number</td>
</tr>
<tr>
<td>Alarm (I/O channel)</td>
<td>AlarmID</td>
<td>Channel number 1 to 4</td>
<td>Rising, Falling, Both</td>
<td>Flag</td>
<td>Flag</td>
<td>1 to 20</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Alarm (math channel)</td>
<td>AlarmMath</td>
<td>Channel number 1 to 4</td>
<td>Rising, Falling, Both</td>
<td>Manual sample</td>
<td>ManualSample</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Alarm (communication channel)</td>
<td>AlarmComm</td>
<td>Channel number 1 to 4</td>
<td>Rising, Falling, Both</td>
<td>DO output</td>
<td>DO</td>
<td>Channel Off, On</td>
<td>number</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Alarm All</td>
<td>-</td>
<td>-</td>
<td>Rising, Falling, Both</td>
<td>Output to an internal switch</td>
<td>SW</td>
<td>Channel Off, On</td>
<td>number</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Device status</td>
<td>Status</td>
<td>-</td>
<td>Rising, Falling, Both</td>
<td>Recording</td>
<td>Memory (Record)</td>
<td>-</td>
<td>Start, Stop</td>
<td>-</td>
<td>ALL, batch group number</td>
</tr>
<tr>
<td>Device status</td>
<td>Status</td>
<td>-</td>
<td>Rising, Falling, Both</td>
<td>Math (Math)</td>
<td>Math</td>
<td>-</td>
<td>Start, Stop, Reset</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Device status [GX/GP]</td>
<td>Status</td>
<td>-</td>
<td>Login (When logged in)</td>
<td>Display data</td>
<td>SaveDisplay</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>ALL, batch group number</td>
</tr>
<tr>
<td>Device status</td>
<td>Status</td>
<td>-</td>
<td>Mem/MediaErr (Memory/Media error), MeasureErr (Measurement error), CommErr (Communication error)</td>
<td>Event data save</td>
<td>SaveEvent</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>ALL, batch group number</td>
</tr>
<tr>
<td>Timer</td>
<td>Timer</td>
<td>1 to 12 -</td>
<td>Edge</td>
<td>Time adjustment</td>
<td>TimeAdjust</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Match time timer</td>
<td>MatchTimeTimer</td>
<td>1 to 12 -</td>
<td>Edge</td>
<td>Settings load</td>
<td>ConfigLoad</td>
<td>1</td>
<td>10</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>User function</td>
<td>User function</td>
<td>1 or 2 -</td>
<td>Edge</td>
<td>Settings save [GM]</td>
<td>ConfigSave</td>
<td>1</td>
<td>10</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Favorite screen display [GX/GP]</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>PlayList</td>
<td>PlayList</td>
<td>1</td>
<td>20</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Internal switch on/off</td>
<td>SWOn_Off</td>
<td>Channel number -</td>
<td>-</td>
<td>Both</td>
<td>MemoryStartStop</td>
<td>-</td>
<td>-</td>
<td>ALL, batch group number</td>
<td></td>
</tr>
<tr>
<td>Math start/stop</td>
<td>MathStartStop</td>
<td>-</td>
<td>-</td>
<td>Math (math channel)</td>
<td>Math</td>
<td>-</td>
<td>Start, Stop, Reset</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Display rate switch 1/2</td>
<td>RateChange1_2</td>
<td>-</td>
<td>-</td>
<td>Display rate switch</td>
<td>RateChange1_2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Flag On/Off</td>
<td>FlagOn_Off</td>
<td>1 to 20</td>
<td>-</td>
<td>-</td>
<td>DO On/Off</td>
<td>DOOn_Off</td>
<td>Channel number</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Internal switch on/off</td>
<td>SWOn_Off</td>
<td>Channel number -</td>
<td>-</td>
<td>Display group change</td>
<td>GroupChange</td>
<td>Display group number</td>
<td>-</td>
<td>-</td>
<td>Batch group number</td>
</tr>
<tr>
<td>Display rate switch</td>
<td>RateChange1_2</td>
<td>-</td>
<td>-</td>
<td>Relative timer reset</td>
<td>TimerReset</td>
<td>1</td>
<td>12</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Settings load</td>
<td>ConfigLoad</td>
<td>1</td>
<td>10</td>
<td>Settings save [GM]</td>
<td>ConfigSave</td>
<td>1</td>
<td>10</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Favorite screen display</td>
<td>PlayList</td>
<td>1</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Pattern file load</td>
<td>ProgPatLoad</td>
<td>1</td>
<td>10</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

1 Can be output only to DO whose type is set to Manual.
2 Can be output only to SW whose type is set to Manual.
3 Valid when the advanced security function (AS) is disabled.
4 For the setting range, see section 2.3.2, “Parameter Notation and Range.”
5 This is valid only for the program control function (PG).

**Query**

SEventAct[,p1]?

Invalid parameters are returned as blanks in queries.

**Example**

Execute memory start on the rising edge of the remote control input (channel 0101). Use event action number 2.

SEventAct,2,On,DI,0101,,Rising,Memory,,Start
**Description**
- There are limitations to event and action combinations. For details, see section 1.14 in the *Model GX10/GX20/GP10/GP20 Paperless Recorder User’s Manual* (IM 04L51B01-01EN) or section 2.15, “Configuring the Event Action Function,” in the *Data Acquisition System GM User’s Manual* (IM 04L55B01-01EN).
- Write only delimiters (commas) for irrelevant parameters (invalid even if a value is specified).
- Event type “DI” is the channel of the DI module that has been set to remote module ([SModeDI](#) command).
- You can specify p3 = Status and p6 = Both when an item that can be specified as an action (flag, DO channel, or internal switch) is valid.
- Math channel and flag are an option ([/MT](#)).
- Communication channels are an option ([/MC](#)).

### SReport

**Report Type ([/MT](#))**

Sets the type of report to create.

#### No Reports

**Syntax**

```bash
SReport,p1
```

- **p1** Type (Off)

#### Hourly and Daily Reports

**Syntax**

```bash
SReport,p1,p2
```

- **p1** Type (Hour+Day)
- **p2** Time to create reports: Hour (HH) (00 to 23)

#### Daily and Weekly Reports

**Syntax**

```bash
SReport,p1,p2,p3
```

- **p1** Type (Day+Week)
- **p2** Day to create reports (Mon, Tue, Wed, Thu, Fri, Sat, Sun)
- **p3** Time to create reports: Hour (HH) (00 to 23)

#### Daily and Monthly Reports

**Syntax**

```bash
SReport,p1,p2,p3
```

- **p1** Type (Day+Month)
- **p2** Day to create reports (1 to 28)
- **p3** Time to create reports: Hour (HH) (00 to 23)

#### Batch Reports

**Syntax**

```bash
SReport,p1,p2
```

- **p1** Type (Batch)
- **p2** Recording interval (2min, 3min, 4min, 5min, 10min, 15min, 30min, 1h)

#### Day Custom Reports

**Syntax**

```bash
SReport,p1,p2,p3,p4,p5
```

- **p1** Type (Custom)
- **p2** Recording interval (2min, 3min, 4min, 5min, 10min, 15min, 30min, 1h)
- **p3** File creation interval (4h, 6h, 8h, 12h, 24h)
- **p4** Time to create reports: Hour (HH) (00 to 23)
- **p5** Time to create reports: Minute (MM) (00 to 59)

**Query**

```bash
SReport[,p1]?
```

**Example**

Create daily reports at 09:00 every day and monthly reports at 09:00 on the first day of each month.

```bash
SReport,Day+Month,1,09
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- If p1=Off, you cannot set p2 or subsequent parameters.
### SRepData

**Report Data (/MT)**
Sets the data type and file type of reports.

**Syntax**

```
SRepData,p1,p2,p3,p4,p5,p6
```

- `p1` Data type 1 (Max, Min, Ave, Sum, Inst)
- `p2` Data type 2 (Off, Max, Min, Ave, Sum, Inst)
- `p3` Data type 3 (Off, Max, Min, Ave, Sum, Inst)
- `p4` Data type 4 (Off, Max, Min, Ave, Sum, Inst)
- `p5` Data type 5 (Off, Max, Min, Ave, Sum, Inst)
- `p6` File type
  - Combine 1 file
  - Separate Separate

**Query**

```
SRepData?
```

**Example**

Record the maximum, minimum, and average values in daily and monthly reports. Generate the daily and monthly reports in a single file.

```
SRepData,Max,Min,Ave,Off,Combine
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

### SDigitalSign

**Electronic Signature Inclusion (/MT)**
Sets whether to include an electronic signature in report template output PDF files.

**Syntax**

```
SDigitalSign,p1,p2
```

- `p1` Signature target (PDF)
- `p2` Electronic signature inclusion (Off, On)

**Query**

```
SDigitalSign[p1]?
```

**Example**

Include an electronic signature in report template output PDF files.

```
SDigitalSign,PDF,On
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

### SRepCh

**Report Channel (/MT)**
Assigns a channel to a report channel.

**Not Assign a Channel**

**Syntax**

```
SRepCh,p1,p2
```

- `p1` Report Channel Number
- `p2` Usage (Off)

**Assign a Channel**

**Syntax**

```
SRepCh,p1,p2,p3,p4
```

- `p1` Report Channel Number
- `p2` Usage
  - IO I/O channel
  - Math Math channel
  - Com Communication channel
- `p3` Channel number
- `p4` Sum scale (Off, /sec, /min, /hour, /day)

**Query**

```
SRepCh[,p1]?
```

**Example**

Assign I/O channel 0002 to report channel 1. Set the sum scale to Off.

```
SRepCh,001,IO,0002,Off
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

- If `p2=Off`, you cannot set `p3` or subsequent parameters.
- Communication channels are an option (/MC).
- If the measurement mode is dual interval, the number of channels is halved.
- If the measurement mode is set to high speed or dual interval, and the scan interval is 50 ms or less, the number of channels must be within their limits.

---

2.4 Setting Commands
### SRepBatchInfo

**Batch information output ([/MT])**

Sets the batch information output.

**Syntax**

```
SRepBatchInfo,p1
```

- **p1** Batch information output (Off, On)
  - Off  Disabled
  - On   Enabled

**Query**

```
SRepBatchInfo?
```

**Example**

- Output batch information.
  ```
  SRepBatchInfo,On
  ```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

### SLcd

**LCD [GX/GP]**

Sets the brightness and backlight saver of the LCD.

**Syntax**

```
SLcd,p1,p2,p3,p4
```

- **p1** Brightness (1 to 6)
- **p2** Backlight saver mode
  - Off  Not used
  - Dimmer Dimmer
  - TimeOff Off
- **p3** Backlight saver saver time (1min, 2min, 5min, 10min, 30min, 1h)
- **p4** Backlight saver restore
  - Key+Touch
  - Key+Touch+Alarm
  - Key, touchscreen, or alarm

**Query**

```
SLcd?
```

**Example**

- Set the LCD brightness to 3 and the screen backlight saver type to DIMMER. Set the amount time of until the GX/GP switches to saver mode to 5 minutes and the event that causes the GX/GP to return from saver mode to the pressing of a key and tapping of the touchscreen.
  ```
  SLcd,3,Dimmer,5min,Key+Touch
  ```

**Description**

- p3 and subsequent parameters are valid when p2=Off.

### SViewAngle

**View Angle [GX/GP]**

Set the view angle.

**Syntax**

```
SViewAngle,p1
```

- **p1** View Angle
  - Upper  Easy to view from above
  - Lower  Easy to view from below

**Query**

```
SViewAngle?
```

**Example**

- Set the view angle so that it is easy to view from above.
  ```
  SViewAngle,Upper
  ```

**Description**

- This command is valid for the GX10/GP10.

### SBackColor

**Screen Background Color [GX/GP]**

Sets the screen background color.

**Syntax**

```
SBackColor,p1,p2,p3
```

- **p1** R value of RGB background colors (0 to 255)
- **p2** G value of RGB background colors (0 to 255)
- **p3** B value of RGB background colors (0 to 255)

**Query**

```
SBackColor?
```

**Example**

- Set the background color to black.
  ```
  SBackColor,0,0,0
  ```

**Description**

- For details on RGB values, see “Description” of the SColorIO command.

### SGrpChange

**Automatic Group Switching Time [GX/GP]**

Sets the time for automatically switching between display groups.

**Syntax**

```
SGrpChange,p1
```

- **p1** Automatic group switching time (5s, 10s, 20s, 30s, 1min)

**Query**

```
SGrpChange?
```

**Example**

- Set the switching time to 1 minute.
  ```
  SGrpChange,1min
  ```

### SAutoJump

**Jump Default Display Operation [GX/GP]**

Sets the amount of time that must elapse until the GX/GP returns to the specified screen (standard screen) when there is no user interaction.

**Syntax**

```
SAutoJump,p1
```

- **p1** Jump default display operation (Off, 1min, 2min, 5min, 10min, 20min, 30min, 1h)

**Query**

```
SAutoJump?
```

**Example**

- Set the automatic return time to 5 minutes.
  ```
  SAutoJump,5min
  ```
## 2.4 Setting Commands

### SCaImFormat

#### Calendar Display Format [GX/GP]
Sets the calendar display format.

**Syntax**
```
SCaImFormat, p1
p1  1st weekday (Sun, Mon)
```

**Query**
```
SCaImFormat?
```

**Example**
Set the first weekday to Monday.
```
SCaImFormat, Mon
```

### SBarDirect

#### Bar Graph Display Direction [GX/GP]
Sets the bar graph display direction.

**Syntax**
```
SBarDirect, p1
p1  Direction
    Horizontal, Vertical
```

**Query**
```
SBarDirect?
```

**Example**
Display bar graphs horizontally.
```
SBarDirect, Horizontal
```

### SChgMonitor

#### Value Modification from the Monitor
Enables or disables the feature that allows values to be changed from the monitor.

**Syntax**
```
SChgMonitor, p1
p1  Disable or enable (Off, On)
```

**Query**
```
SChgMonitor?
```

**Example**
Enable the feature that allows values to be changed from the monitor.
```
SChgMonitor, On
```

### STrdWave

#### Trend Waveform Display [GX/GP]
Sets the trend waveform display mode.

**Syntax**
```
STrdWave, p1, p2
p1  Waveform display direction
    Horizontal, Vertical
p2  Trend clear
    Off, Do not clear
    On, Clear
```

**Query**
```
STrdWave?
```

**Example**
Set the trend waveform to horizontal display and clear the waveform when recording is started.
```
STrdWave, Vertical, On
```

**Description**
- When the multi batch function (/BT) is enabled, p2 is fixed to On.

### STrdScale

#### Scale [GX/GP]
Set the scale.

**Syntax**
```
STrdScale, p1, p2, p3
p1  Number of digits to display for scale values.
    Normal, Fine
p2  Current value display
    Mark, Bar
p3  Number of digits to display for channels that are added to the current value mark
    0-digit, 3-digit, 4-digit
```

**Query**
```
STrdScale?
```

**Example**
Set the number of digits to display for scale values to “Fine,” display the value indicators on a bar graph, and set the number of digits to display for channels that are added to the current value mark to 4 digits.
```
STrdScale, Fine, Bar, 4-digit
```

### STrdLine

#### Trend Line Width, Grid [GX/GP]
Sets the trend waveform line width and the grid in the display area.

**Syntax**
```
STrdLine, p1, p2
p1  Line width
    Thick, Normal, Thin
p2  Grid
    Auto, 4 to 12
```

**Query**
```
STrdLine?
```

**Example**
Set the trend waveform line width to “Thin” and the number of grid lines to 10.
```
STrdLine, Thin, 10
```
2.4 Setting Commands

### STrdRate

**Trend Interval Switching [GX/GP]**
Sets the trend interval switching.

**Syntax**

```
STrdRate, p1, p2
```

- **p1** Trend interval switching
  - Off  Not switch
  - On  Switch

- **p2** Second trend interval (5s, 10s, 15s, 30s, 1min, 2min, 5min, 10min, 15min, 20min, 30min, 1h, 2h, 4h, 10h).

**Query**

```
STrdRate?
```

**Example**

Set the second trend interval to 30 seconds.

```
STrdRate, On, 30s
```

**Description**
- You cannot set parameter p1 while recording is in progress.
- You cannot set parameter p1 while computation is in progress.
- p2 is valid only when p1=On.
- You cannot choose a second trend interval that is shorter than the scan interval.
- Trend intervals shorter than 30 s cannot be specified if an electro-magnetic relay type analog input module is in use (set up).
- When the multi batch function (/BT) is enabled, p1 is fixed to On.
- You cannot use this command to configure settings when the measurement mode is set to high speed or dual interval.

### STrdKind

**Trend Type [GX/GP]**
Sets the type of trend waveform to display.

**Syntax**

```
STrdKind, p1
```

- **p1** Type
  - Fixed to "T-Y"

**Query**

```
STrdKind?
```

**Example**

Display using rectangular coordinates.

```
STrdKind, T-Y
```

### STrdPartial

**Partial Expanded Trend Display [GX/GP]**
Enable or disable the partial expanded trend display.

**Syntax**

```
STrdPartial, p1
```

- **p1** Disable or enable (Off, On)

**Query**

```
STrdPartial?
```

**Example**

Enable the partial expanded trend display.

```
STrdPartial, On
```

### SMsgBasic

**Message Writing**
Sets the message writing operation.

**Syntax**

```
SMsgBasic, p1, p2, p3
```

- **p1** Message writing method
  - (GX/GP: Common, Separate)
  - (GM: Common)
  - Common  Write messages to all display groups.
  - Separate  Write messages to only the groups that are displayed.
- **p2** Power failure message (Off, On)
- **p3** Change message (Off, On)
  - (GX/GP: On, Off)
  - (GM: On, Off, Fixed to Off when the advanced security function (/AS) is disabled)

**Query**

```
SMsgBasic?
```

**Example**

Write messages to only the groups that are displayed. Enable the power failure message and change message.

```
SMsgBasic, Separate, On, On
```

### SGroup

**Display Group**
Sets the display group.

**Syntax**

```
SGroup, p1, p2, p3, p4
```

- **p1** Group number
- **p2** Enable or disable (Off, On)
- **p3** Group name (up to 16 characters, UTF-8)
- **p4** Channel string
  - • Specify using channel numbers. 4-digit numbers for I/O channels. Numbers that start with “A” for math channels (A015). Numbers that start with “C” for communication channels (C020). The maximum number of characters per channel is 4.
  - • Use periods to separate channel numbers (see example).

**Query**

```
SGroup[,p1]?
```

The channel string is output exactly as it is specified.

**Example**

Assign channels 0001, 0003, 0005, A001, and C023 to group 2 and name it “GROUP A.”

```
SGroup, 2, On,'GROUP A','1.3.5.A1.C23'
```

**Description**
- This command is valid only when the multi batch function (/BT) is disabled.
- For IO channels in dual interval measurement, only the channels in the scan group specified with the SDualGroup command can be selected for p4.
### STripLine

**Display Group Trip Line**
Sets a trip line for a display group.

**Syntax**
```
STripLine,p1,p2,p3,p4,p5,p6,p7,p8
```
- **p1**: Group number
- **p2**: Trip line number (1 to 4)
- **p3**: Enable or disable (Off, On)
- **p4**: Display position [%] (0 to 100)
- **p5**: R value of RGB display colors (0 to 255)
- **p6**: G value of RGB display colors (0 to 255)
- **p7**: B value of RGB display colors (0 to 255)
- **p8**: Line width
  - (GX/GP: Thin, Normal, Thick)
  - (GM: Normal)
  - Thin
  - Normal
  - Thick

**Query**
```
STripLine[,p1[,p2]]?
```

**Example**
Assign character string "MESSAGE77" to message number 77.
```
SMessage,77,'MESSAGE77'
```

**Description**
- For details on RGB values, see “Description” of the SColorIO command.
- This command is valid only when the multi batch function (/BT) is disabled.

### STimeZone

**Time Zone**
Sets the time zone.

**Syntax**
```
STimeZone,p1,p2
```
- **p1**: Time zone: Hour (-13 to 13)
- **p2**: Time zone: Minute (0 to 59)

**Query**
```
STimeZone?
```

**Example**
```
STimeZone,9,0
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

### SSclBmp

**Scale Bitmap Image Usage [GX/GP]**
Sets whether to display a bitmap scale image in the trend display of a display group.

**Syntax**
```
SSclBmp,p1,p2
```
- **p1**: Group number
- **p2**: Enable or disable (Off, On)

**Query**
```
SSclBmp[,p1]?
```

**Example**
```
SSclBmp,3,On
```

**Description**
- Specify the bitmap file to use from the front panel of the GX/GP.
- This command is valid only when the multi batch function (/BT) is disabled.

### SDateBasic

**Gradual Time Adjustment**
Sets the gradual time adjustment feature.

**Syntax**
```
SDateBasic,p1,p2
```
- **p1**: Boundary value for gradually adjusting the time (Off, 5s, 10s, 15s)
- **p2**: Action to take when the boundary value for gradually adjusting the time is exceeded.
  - NotChange
  - Change

**Query**
```
SDateBasic?
```

**Example**
```
SDateBasic,15s,NotChange
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

### SDateFormat

**Date Format**
Sets the date format.

**Syntax**
```
SDateFormat,p1,p2,p3
```
- **p1**: Date format
  - YYMMDD Year, month, day
  - MMDDYY Month, day, year
  - DDMMYY Date, month, year
- **p2**: Delimiter
  - Slash
  - Dot (period)
  - Hyphen
- **p3**: Month display
  - Digit
  - Display the month using numerals (1 to 12)

**Example**
```
SDateFormat,DDMMYY,/,Digit
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
### 2.4 Setting Commands

#### SDst

**Daylight Saving Time**
Set the daylight saving time.

**Syntax**
```
SDst, p1, p2, p3, p4, p5, p6, p7, p8, p9
```

- **p1**: Enable or disable (Use, Not)
- **p2**: Start time: Month (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec)
- **p3**: Start time: Week (1st, 2nd, 3rd, 4th, Last)
- **p4**: Start time: Weekday (Sun, Mon, Tue, Wed, Thu, Fri, Sat)
- **p5**: Start time: Hour (0 to 23)
- **p6**: End time: Month (Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec)
- **p7**: End time: Week (1st, 2nd, 3rd, 4th, Last)
- **p8**: End time: Weekday (Sun, Mon, Tue, Wed, Thu, Fri, Sat)
- **p9**: End time: Hour (0 to 23)

**Query**
SDst?

**Example**
Switch to daylight saving time at hour 0 on the first Sunday of June and switch back at hour 0 on the first Sunday of December.
```
SDst, On, Jun, 1st, Sun, 0, Dec, 1st, Sun, 0
```

#### STemp

**Temperature Unit**
Sets the temperature unit.

**Syntax**
```
STemp, p1
```

- **p1**: Temperature unit
  - C: Celsius
  - F: Fahrenheit

**Query**
STemp?

**Example**
Set the temperature unit to Celsius.
```
STemp, C
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

#### SLang

**Language**
Sets the language to use.

**Syntax**
```
SLang, p1
```

- **p1**: Language (Japanese, English, German, French, Chinese, Russian, Korean)

**Query**
SLang?

**Example**
Set the language to Japanese.
```
SLang, Japanese
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- If you change the language with this command, the recorder may restart.
2.4 Setting Commands

**SFailSts**

**Instrument Status to Output (/FL) [GX/GP]**

Sets the instrument status to output from the fail relay (DO channel).

**Syntax**

```
SFailSts, p1, p2, p3, p4, p5
```

- **p1** Memory/media status (Off, On)
- **p2** Measurement error (Off, On)
- **p3** Communication error (Off, On)
- **p4** Recording stop (Off, On)
- **p5** Alarm (Off, On)

**Query**

```
SFailSts?
```

**Example**

Output all information.

```
SFailSts, On, On, On, On, On
```

**Description**

- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

**SPrinter**

**Printer**

Sets the printer.

**Syntax**

```
SPrinter, p1, p2, p3, p4, p5, p6, p7
```

- **p1** IP address (0. 0. 0. 0 to 255. 255. 255. 255)
- **p2** Paper size (A4, A3, Letter)
- **p3** Paper orientation (Horizontal, Vertical)
- **p4** Resolution [dpi] (300, 600)
- **p5** Number of copies (1 to 10)
- **p6** Snapshot (Off, On) (GX/GP: Off, On) (GM: Off)
- **p7** Fit to page during snapshot printing (Off, On) (GX/GP: Off, On) (GM: Off)

**Query**

```
SPrinter?
```

**Example**

Set the IP address to “192.168.111.24,” the paper size to A3, the paper orientation to horizontal, the resolution to 600, the number of copies to 2, and snapshot to On. Print by fitting to page.

```
SPrinter, 192.168.111.24, A3, Horizontal, 600, 2, Off, Off
```

**SSound**

**Sound [GX/GP]**

Sets touch and warning sounds.

**Syntax**

```
SSound, p1, p2
```

- **p1** Touch sound (Off, On)
- **p2** Warning sound (Off, On)

**Query**

```
SSound?
```

**Example**

Enable touch and warning sounds.

```
SSound, On, On
```

**SInstruTag**

**Instruments Tag**

Sets tags.

**Syntax**

```
SInstruTag, p1, p2
```

- **p1** Tag (up to 32 characters, UTF-8)
- **p2** Tag number (up to 16 characters, ASCII)

**Query**

```
SInstruTag?
```

**Example**

Set the tag to assign to the GX/GP to “GX” and the tag number to “12345.”

```
SInstruTag, ‘GX’, ‘12345’
```

**SConfCmt**

**Setting File Comment**

Sets the setting file comment.

**Syntax**

```
SConfCmt, p1
```

- **p1** Setting file comment (up to 50 characters, UTF-8)

**Query**

```
SConfCmt?
```

**Example**

Set “SETTING FILE COMMENT.”

```
SConfCmt, ‘SETTING FILE COMMENT’
```
### Commands and Responses

#### 2.4 Setting Commands

---

**SUsubInput**

**USB Input Device [GX/GP]**

Specifies the USB input device.

**Syntax**

```
SUsubInput,p1
```

- **p1** USB input device type
  - Japanese_109 Japanese keyboard
  - English_104 English keyboard
  - Barcode Bar-code reader

**Query**

`SUsubInput?`

**Example**

Specify the English keyboard.

```
SUsubInput,English_104
```

**Description**

- This command is valid on models with the /UH USB interface option.
- For the communication commands that you can execute using a bar-code reader, see section 1.18.11, “Setting USB Input Devices (/UH option)” in the Model GX10/GX20/GP10/GP20 Paperless Recorder User’s Manual (IM 04L51B01-01EN).

---

**SSetComment**

**Configuration Changes Comment (/AS)**

Sets whether to enter comments when settings are changed.

**Syntax**

```
SSetComment,p1
```

- **p1** Enable/disable configuration changes comment
  - On Enter comments when settings are changed.
  - Off Do not enter comments when settings are changed.

**Query**

`SSetComment?`

**Example**

Enter comments when settings are changed.

```
SSetComment,On
```

---

**SSwitch**

**Internal Switch Operation**

Sets the internal switch operation.

**Syntax**

```
SSwitch,p1,p2,p3,p4
```

- **p1** Internal switch number (1 to 100)
- **p2** Output type
  - Alarm Output alarms
  - Manual Specify the output value
- **p3** Operation
  - And Operate when all set alarms are in the alarm state.
  - Or Operate when any of the set alarms are in the alarm state.
- **p4** Power supply
  - Last Output the previous value
  - On Output 1
  - Off Output 0

**Query**

`SSwitch[,p1]?

**Example**

Output an alarm on internal switch 3. Use “OR” logic.

```
SSwitch,3,Alarm,Or
```

**Description**

- p3 is valid when p2=Alarm.
- p4 is valid when p2 is set to Manual.

---

**SSerialBasic**

**Serial Communication Basics (/C2 or /C3)**

Sets basic serial communication parameters.

- **Not Use**
- **Normal/Bar-code**

**Syntax**

```
SSerialBasic,p1
```

- **p1** Function (Off)

**Modbus Master and Modbus Slave**

**Syntax**

```
SSerialBasic,p1,p2,p3,p4
```

- **p1** Function (Master, Slave)
- **p2** Address (1 to 247)
- **p3** Baud rate [bps] (1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200)
- **p4** Parity (Odd, Even, None)

**Query**

`SSerialBasic?`

**Example**

Set the baud rate to 9600, the data length to 8, the parity check to ODD, the stop bits to 1, the handshaking to OFF:OFF, the address to 02, and the protocol to NORMAL.

```
SSerialBasic,Normal,2,9600,Odd,1,8,Off:Off
```

**Description**

- You can set p1=Master only on recorders that have the /MC option.
- The settings specified with this command takes effect with the OSeriApply command. The recorder serial settings do not change until you send the OSeriApply command.
- For the communication commands that you can execute using a bar-code reader, see section 1.18.11, “Setting USB Input Devices (/UH option)” in the Model GX10/GX20/GP10/GP20 Paperless Recorder User’s Manual (IM 04L51B01-01EN).
### SModMaster

**Modbus Master (/C2/MC or /C3/MC)**

Sets the Modbus master operation.

**Syntax**

```plaintext
SModMaster, p1, p2, p3, p4, p5, p6
```

- **p1** Master function (Off, On)
- **p2** Read cycle (100ms, 200ms, 500ms, 1s, 2s, 5s, 10s, 20s, 30s, 1min)
- **p3** Communication timeout (100ms, 200ms, 250ms, 500ms, 1s, 2s, 5s, 10s, 1min)
- **p4** Gap between messages (Off, 5ms, 10ms, 20ms, 50ms, 100ms)
- **p5** Recovery action: retransmission (Off, 1, 2, 3, 4, 5, 10, 20)
- **p6** Recovery action: wait time (Off, 5s, 10s, 30s, 1min, 2min, 5min)

**Query**

```plaintext
SModMaster?
```

**Example**

Set the read cycle to 500ms, the communication timeout to 250ms, the gap between messages to 10ms, the retransmission to 2, and the recovery wait time to 5min.

```plaintext
SModMaster, On, 500ms, 250ms, 10ms, 2, 5min
```

### SModMCmd

**Modbus Master Transmission Command (/C2/MC or /C3/MC)**

Sets a transmit command of the Modbus master.

**Syntax**

```plaintext
SModMCmd, p1, p2, p3, p4, p5, p6, p7, p8
```

- **p1** Command number (1 to 100)
- **p2** Command type
  - Off: Disable command
  - Write: Write a value to a Modbus register of another device
  - Read: Read a value from a Modbus register of another device
- **p3** Slave number (1 to 247)
- **p4** Data type
  - BIT: Bit String data
  - INT16: 16-bit signed integer
  - UINT16: 16-bit unsigned integer
  - INT32_B: 32-bit signed integer (big endian)
  - INT32_L: 32-bit signed integer (little endian)
  - UINT32_B: 32-bit unsigned integer (big endian)
  - UINT32_L: 32-bit unsigned integer (little endian)
  - FLOAT_B: 32-bit floating point (big endian)
  - FLOAT_L: 32-bit floating point (little endian)
- **p5** Register (1 to 465535)
- **p6** Channel type
  - IO: I/O channel
  - Math: Math channel
  - Com: Communication channel
- **p7** First channel
- **p8** Last channel

**Query**

```plaintext
SModMCmd[p1]?
```

**Example**

Register the following command in command number 2: read the 32-bit signed integer data that is assigned to registers 30003 (upper 16 bits) and 30004 (lower 16 bits) in the slave device assigned to address 5 into channel C02.

```plaintext
SModMCmd, 2, Read, S, INT32_B, 30003, Com, 002
```

### SSerailAutoLOut

**Auto Logout for Serial Communication (/C2 or /C3)**

Sets the auto logout function for serial communication.

**Syntax**

```plaintext
SSerialAutoLOut, p1
```

- **p1** Auto logout function (Off, 1min, 2min, 5min, 10min)

**Query**

```plaintext
SSerialAutoLOut?
```

**Example**

Set the auto logout time for users logged in through serial communication to 1 minute.

```plaintext
SSerialAutoLOut, 1min
```

**Description**

- Auto logout is applied to users logged in through serial communication when the communication security function is set to Login (p2 of the SSecurity command) and the receiver function setting in the basic serial settings (p1 of the SSeriBasic command) is set to Normal.

### SIpAddress

**IP Address Information**

Sets the IP address information.

**Syntax**

```plaintext
SIpAddress, p1, p2, p3
```

- **p1** IP address (0.0.0.0 to 255.255.255.255)
- **p2** Subnet mask (0.0.0.0 to 255.255.255.255)
- **p3** Default gateway (0.0.0.0 to 255.255.255.255)

**Query**

```plaintext
SIpAddress?
```

**Example**

Set the IP address to 192.168.111.24, the subnet mask to 255.255.255.0, and the default gateway to 192.168.111.20.

```plaintext
SIpAddress, 192.168.111.24, 255.255.255.0, 192.168.111.20
```
### SClient

**Client Function**

Sets the client function.

**Syntax**

```plaintext
SClient, p1, p2
```

- `p1`: Client type (FTP, SMTP, SNTP, MODBUS, WATT, SLMP)
- `p2`: Client Function (Off, On)

**Query**

`SClient[p1]?
```

**Example**

Use the FTP client function.

`SClient,FTP,On`

**Description**

- Modbus client is valid on models with the /MC communication channel option.
- WATT connection client is valid on models with the WT communication (IE2) option.
- SLMP client is valid on models with the SLMP client (/E4) option.

### SClientEncrypt

**Client Communication Encryption**

Sets whether to encrypt FTP client communication and SMTP client communication.

**Syntax**

```plaintext
SClientEncrypt, p1, p2, p3
```

- `p1`: Client type (FTP, SMTP)
- `p2`: Encryption (Off, On)
- `p3`: Verification of certificate (Off, On)

**Query**

`SClientEncrypt[p1]?
```

**Example**

Encrypt FTP client communication. Check that the certificate in the recorder matches the certificate received from the server.

`SClientEncrypt,FTP,On,On`

### SDns

**DNS Information**

Sets the DNS information.

**Host (GX)**

**Syntax**

```plaintext
SDns, p1, p2, p3
```

- `p1`: Setting type (Host)
- `p2`: Host name (up to 64 characters, ASCII)
- `p3`: Domain name (up to 64 characters, ASCII)

**DNS Server**

**Syntax**

```plaintext
SDns, p1, p2, p3
```

- `p1`: Setting type (Server)
- `p2`: Primary DNS server (0.0.0.0 to 255.255.255.255)
- `p3`: Secondary DNS server (0.0.0.0 to 255.255.255.255)

**Query**

`SDns[p1]?
```

**Example**

Set the IP address of the primary DNS server to 192.168.111.1 and the IP address of the secondary DNS server to 192.168.111.10

`SDns,Server,192.168.111.1,192.168.111.10`

**Description**

- The settings specified with this command takes effect with the `OIPApply` command. The recorder IP address does not change until you send the `OIPApply` command.

### SDhcp

**DHCP Client**

Sets the DHCP client.

**Do Not Obtain the IP Address Automatically**

**Syntax**

```plaintext
SDhcp, p1
```

- `p1`: Automatic IP address assignment (Off)

**Obtain the IP Address Automatically**

**Syntax**

```plaintext
SDhcp, p1, p2, p3
```

- `p1`: Automatic IP address acquisition (On)
- `p2`: DNS information acquisition (Off, On)
- `p3`: Automatic host name registration (Off, On)

**Query**

`SDhcp?
```

**Example**

Automatically obtain the IP address and DNS information and automatically register the host name.

`SDhcp,On,On,On`

**Description**

- The settings specified with this command takes effect with the `OIPApply` command. The recorder IP address does not change until you send the `OIPApply` command.

### SFtpKind

**File to Transfer via FTP**

Sets the file to transfer via FTP.

**Syntax**

```plaintext
SFtpKind, p1, p2
```

- `p1`: Setting type

**Data**

- `(GX/GP: Data, Report, Snapshot, AlarmSummary, ManualSample, Setting)`

**Report**

- `(GM: Data, Report, AlarmSummary, ManualSample, Setting)`

**Query**

`SFtpKind[p1]?
```

**Example**

Automatically transfer display and event data files when files are generated.

`SFtpKind,Data`

**Description**

- The settings specified with this command takes effect with the `OIPApply` command. The recorder IP address does not change until you send the `OIPApply` command.
2.4 Setting Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Snapshot</td>
<td>Automatically transfer snapshot data files when files are generated.</td>
</tr>
<tr>
<td>AlarmSummary</td>
<td>Transfer alarm summaries</td>
</tr>
<tr>
<td>ManualSample</td>
<td>Automatically transfer manual sampled data files when manual sampling is executed.</td>
</tr>
<tr>
<td>Setting</td>
<td>Automatically transfer the setting file when settings are changed.</td>
</tr>
</tbody>
</table>

**Example**

For the primary server, assign the name "server1" and port number 21. Set the user name to "Administrator1," the password to "password1," and the directory to "directory1." Set PASV mode to Off.

SSftpCnct, Primary,'server1',21,'Administrator1','password1','directory1',Off

**SSftpLogin**

SMTP User Authentication

Sets the SMTP user authentication method.

**Syntax**

SSftpLogin, pl

**p1**

- **User authentication type**
  - Off: Not use authentication.
  - Auth-Smtp: Use Authentication SMTP.
  - POP3: Use POP Before SMTP (unencrypted).
  - APOP: Use POP Before SMTP (encrypted).

**Query**

SSftpLogin?

**Example**

Do not use authentication.

SSftpLogin, Off

**SSftpCnct**

SMTP Client Connection Destination Server

Sets the SMTP client connection destination server.

**Syntax**

SSftpCnct, pl, p2, p3, p4, p5

**p1**

- **Destination server type** (SMTP, POP)

**p2**

- **Server name** (up to 64 characters, ASCII)

**p3**

- **Port number** (1 to 65535)

**p4**

- **User name** (up to 32 characters, ASCII)

**p5**

- **Password** (up to 32 characters, ASCII)

**Query**

SSftpCnct[pl]?

**Example**

Connect to SMTP server "SMTPserver1." Set the port number to 25, the user name to "administrator1," and the password to "password1."

SSftpLogin, SMTP,'SMTPserver1',25,'administrator1','password1'

**SMailHead**

Mail Header

Sets the mail header including the recipient address.

**Syntax**

SMailHead, pl, p2, p3, p4

**p1**

- **Sender address** (up to 64 characters, ASCII)

**p2**

- **Recipient address 1** (up to 150 characters, ASCII)

**p3**

- **Recipient address 2** (up to 150 characters, ASCII)

**p4**

- **Character string to add to the subject** (up to 32 characters, ASCII)

**Query**

SMailHead?
2.4 Setting Commands

Example Set the sender address to "recorder1@data.com" and the recipient address to "pc1@data.com." Add "part1" to the subject.
SMailHead,'recorder1@data.com', 'pc1@data.com',,'part1'

SMailBasic

Common Section of the Mail Body
Sets the items that are common to the body of all mails.
Syntax SMailBasic, p1, p2
p1 Header string (up to 128 characters, UTF-8)
p2 Include source URL (Off, On)
Query SMailBasic?
Example Set the header to "recorder1," and include the source URL.
SMailBasic,'recorder1',On

SMail

Destination and Behavior for Each Mail Type
Sets the destination and behavior for each mail type.
Alarm Notification
Syntax SMail, p1, p2, p3, p4, p5, p6
p1 Setting type (Alarm)
p2 Recipient (Off, 1, 2, 1+2)
Off Not send
1 Send to recipient 1
2 Send to recipient 2
1+2 Send to recipient 1 and 2
p3 Inclusion of instantaneous data (Off, On)
On Include instantaneous data
p4 Alarm action
On Send mails when alarms occur
On+Off Send mails when alarms occur and when they are cleared
p5 Inclusion of tag number or channel number in subject (Off, On)
Scheduled Transmission
Syntax SMail, p1, p2, p3
p1 Setting type (Time)
p2 Recipient (Off, 1, 2, 1+2)
p3 Inclusion of instantaneous data (Off, On)
Report Notification (MT)
Syntax SMail, p1, p2
p1 Setting type (Report)
p2 Recipient (Off, 1, 2, 1+2)
Media Alarm Notification
Syntax SMailAlarm, p1, p2
p1 Setting type (Media)
p2 Recipient (Off, 1, 2, 1+2)
Power failure notification
Syntax SMail, p1, p2
p1 Setting type (Power)
p2 Recipient (Off, 1, 2, 1+2)
System Error Notification
Syntax SMail, p1, p2
p1 Setting type (System)
p2 Recipient (Off, 1, 2, 1+2)

User Lockout Notification (LAS)
Syntax SMail, p1, p2
p1 Setting type (UserLock)
p2 Recipient (Off, 1, 2, 1+2)
Query SMail[p1]?
Example Send alarm notifications to recipient 1 when alarms occur and when they are cleared. Include instantaneous data at the time of transmission, and include the tag number or channel number in the subject.
SMail,Alarm,1,On,On+Off,On

Description
• The report function is an option (MT).

SMailAlarm

Alarm Notification Mail Target Channels
Detects the alarm status of the specified channels and sends alarm notifications.
Syntax SMailAlarm, p1
p1 Channel string (up to 249 characters, up to 50 channels)
• Use channel number to specify the channels. 4-digit numbers for I/O channels. Numbers that start with "A" for math channels (A015). Numbers that start with "C" for communication channels (C020). The maximum number of characters per channel is 4.
• Use periods to separate channel numbers (see example).
• To specify all channels from the first channel to the last channel, delimit the channels with a hyphen. An error will occur if there are no valid channels in the hyphen designated channels.
Query SMailAlarm?
The channel string is output exactly as it is specified.
Example Set the target channels to channels 0001 to 0021, 0101, A025, and C003.

SMailAlarmLevel

Alarm Notification Mail Target Alarm levels
Detects the alarm status of the specified alarm levels and sends alarm notification mails.
Syntax SMailAlarmLevel, p1, p2, p3, p4
p1 Alarm level 1 (On, Off)
p2 Alarm level 2 (On, Off)
p3 Alarm level 3 (On, Off)
p4 Alarm level 4 (On, Off)
Query SMailAlarmLevel?
Example Set the target alarm levels 1 and 2 to On, 3 and 4 to Off.
SMailAlarmLevel,On,On,Off,Off

Example Set the sender address to "recorder1@data.com" and the recipient address to "pc1@data.com." Add "part1" to the subject.
SMailHead,'recorder1@data.com', 'pc1@data.com',,'part1'
### SMailAlarmDetect

**Alarm Notification Mail Target Alarm Detection Method**

Sets the alarm detection method for the alarm notification mail.

**Syntax**

```
SMailAlarmDetect, p1
```

**p1** Detection method (Ch, Level)

**Query**

```
SMailAlarmDetect?
```

**Example**

Set alarm levels to specify the target alarms. Use SMailAlarmDetect, Level

**Description**

- When **p1=Ch**, use SMailAlarm command to set the target channels. When **p1=Level**, use SMailAlarmLevel command to set the target levels.

### SMailTime

**Scheduled Transmission Times**

Sets the scheduled transmission times.

**Syntax**

```
SMailTime, p1, p2, p3, p4
```

**p1** Recipient (1 or 2)

**p2** Reference time: Hours (HH) (00 to 23)

**p3** Reference time: Minutes (MM) (00 to 59)

**p4** Interval (1h, 2h, 3h, 4h, 6h, 8h, 12h, 24h)

**Query**

```
SMailTime[, p1]?
```

**Example**

Send mail to recipient 1 every day at 08:30.

```
SMailTime, 1, 08, 30, 24
```

### SSntpCnct

**SNTP Client**

Sets the SNTP client operation and the connection destination server.

**Syntax**

```
SSntpCnct, p1, p2, p3, p4, p5, p6, p7
```

**p1** Server name (up to 64 characters, ASCII)

**p2** Port number (1 to 65535)

**p3** Reference time: Hours (HH) (00 to 23)

**p4** Reference time: Minutes (MM) (00 to 59)

**p5** Access interval (6h, 12h, 24h)

**p6** Timeout (10s, 30s, 90s)

**p7** Time adjust on start action (Off, On)

**Query**

```
SSntpCnct?
```

**Example**

Set the server name to "sntpserver1," the port number to 123, the timeout to 30s. Query the time every day at 12:00 and at memory start.

```
SSntpCnct, 'sntpserver1', 123, 12, 00, 24, 30s, On
```

### SModClient

**Modbus Client Operation (/MC)**

Sets the Modbus client operation.

**Syntax**

```
SModClient, p1, p2, p3, p4
```

**p1** Read cycle (100ms, 200ms, 500ms, 1s, 2s, 5s, 10s, 20s, 30s, 1min, 2min, 5min)

**p2** Recovery wait time (Off, 5s, 10s, 30s, 1min, 2min, 5min)

**p3** Keep connection (Off, On)

**p4** Connection timeout [s] (1 to 10)

**Query**

```
SModClient?
```

**Example**

Set the read cycle to 100ms, the recovery wait time to Off, and the connection timeout to 1 second.

```
SModClient, 100ms, off, on, 1
```

**Description**

- This command is valid on models with the /MC communication channel option.

### SModCList

**Modbus Client Connection Destination Server (/MC)**

Sets the Modbus client connection destination server.

**Syntax**

```
SModCList, p1, p2, p3
```

**p1** Registration number

**p2** Server name (up to 64 characters, ASCII)

**p3** Port number (1 to 65535)

**Query**

```
SModCList[, p1]?
```

**Example**

Assign server name "recorder1" and port number "502" to registration number 1.

```
SModClient, 1, 'recorder1', 502
```

### SModCCmd

**Modbus Client Transmission Command (/MC)**

Sets the Modbus client transmission command.

**Syntax**

```
SModCCmd, p1, p2, p3, p4, p5, p6, p7, p8
```

**p1** Command number

**p2** Command type

- Off Disable command
- Write Write a value to a Modbus register of another device.
- Read Read a value from a Modbus register of another device.

**p3** Server number (1 to 16)

**p4** Unit number (1 to 255)

**p5** Data type

- BIT Bit String data
- INT16 16-bit signed integer
- UINT16 16-bit unsigned integer
- INT32_B 32-bit signed integer (big endian)
- INT32_L 32-bit signed integer (little endian)
- UINT32_B 32-bit unsigned integer (big endian)
- UINT32_L 32-bit unsigned integer (little endian)
- FLOAT_B 32-bit floating point (big endian)
- FLOAT_L 32-bit floating point (little endian)

**p6** Register (1 to 465535)

**p7** Register (1 to 465535)

**p8** Register (1 to 465535)
2.4 Setting Commands

**SServer**

Server Function

Enables or disables the server function.

**Syntax**

SServer,p1,p2,p3

- **p1** Server type (FTP, HTTP, SNTP, MODBUS, GENE, EtherNetIP, DARWIN, OPC-UA)
- **p2** Operation (Off, On)
- **p3** Port number (1 to 65535)

**Query**

SServer[,p1]?

**Example**

Use the FTP server function.

SServer,FTP,On

**Description**

- You cannot specify a port number that is used by another function.
- p3 cannot be set to 44818, 2222, 34150, or 34151.
- p3 is invalid when p1 = DARWIN (Darwin compatible communication) or when p1 = EtherNetIP.
- The default port numbers are listed below.

<table>
<thead>
<tr>
<th>Server type (p1)</th>
<th>Default port number</th>
</tr>
</thead>
<tbody>
<tr>
<td>FTP</td>
<td>21</td>
</tr>
<tr>
<td>HTTP</td>
<td>80</td>
</tr>
<tr>
<td>SNTP</td>
<td>123</td>
</tr>
<tr>
<td>MODBUS</td>
<td>502</td>
</tr>
<tr>
<td>GENE</td>
<td>34434</td>
</tr>
</tbody>
</table>

- p1 = EtherNetIP is an option (/E1).
- p1=OPC-UA is valid on models with the OPC-UA server (/E3) option.
- The settings specified with this command takes effect with the OIPApply command.

## SServerEncrypt

Server Communication Encryption

Sets server communication encryption.

**Syntax**

SServerEncrypt,p1,p2

- **p1** Server type (FTP, HTTP)
- **p2** Encryption (Off, On)

**Query**

SServerEncrypt[,p1]?

**Example**

Encrypt FTP server communication.

SServerEncrypt,FTP,On

## SKeepAlive

Keepalive

Sets the keepalive function.

**Syntax**

SKeepAlive,p1

- **p1** Operation (Off, On)

**Query**

SKeepAlive?

**Example**

Use keepalive.

SKeepAlive,On

## STimeOut

Communication Timeout

Sets the communication timeout function.

**Syntax**

STimeOut,p1,p2

- **p1** Timeout function (Off, On)
- **p2** Timeout value [minutes] (1 to 120)

**Query**

STimeOut?

**Example**

Enable the communication timeout, and set the timeout value to 3 minutes.

STimeOut,On,3

## SFtpFormat

FTP Server Directory Output Format

Sets the FTP server directory output format.

**Syntax**

SFtpFormat,p1

- **p1** FTP server directory output format (MS-DOS, UNIX)

**Query**

SFtpFormat?

**Example**

Specify MS-DOS.

SFtpFormat,MS-DOS

## SModDelay

Modbus Server Delay Response

Sets the Modbus server delay response.

**Syntax**

SModDelay,p1

- **p1** Delay response (Off, 10ms, 20ms, 50ms)

**Query**

SModDelay?

**Example**

Specify no delay response.

SModDelay,Off
2.4 Setting Commands

**SModLimit**

Modbus Server Connection Limit
Enables or disables the Modbus server connection limit function.

**Syntax**
SModLimit,p1
  p1 Connection limit (Off, On)

**Query**
SModLimit?

**Example**
Enable connection limit.
SModLimit,On

**SModList**

IP Address to Allow Connection to Modbus Server
Sets the IP address to allow connection to Modbus server.

**Syntax**
SModList,p1,p2,p3
  p1 Registration number (1 to 10)
  p2 Enable or disable registration (Off, On)
  p3 IP address (0.0.0.0 to 255.255.255.255)

**Query**
SModList?,p1

**Example**
Register IP address "192.168.111.24" to registration number 1.
SModList,1,On,192.168.111.24

**SWattList**

WT Communication Connection Server (/E2)
Sets the WT communication connection server.

**Syntax**
SWattList,p1,p2,p3,p4
  p1 Registration number
  p2 Enable or disable (On, Off)
  p3 Server name (up to 64 characters, ASCII)
  p4 Model (WT300, WT500, WT1800)

**Query**
SWattList?,p1

**Example**
Register model WT1800 and server name "Watt01" in registration number 1.
SWattList,1,On,Watt01,WT1800

**SWattClient**

WT Communication Operation (/E2)
Sets the WT communication operation.

**Syntax**
SWattClient,p1,p2
  p1 Read cycle (500ms, 1s, 2s, 5s, 10s, 20s, 30s)
  p2 Recovery wait time (5s, 10s, 30s, 1min, 2min, 5min)

**Query**
SWattClient?

**Example**
Set the read cycle to 10 seconds and recovery wait time to 2 minutes.
SWattClient,10,2min

**SWattData**

WT Data Allocation to Communication Channel (/E2)
Allocates WT data to a communication channel.

**Syntax**
SWattData,p1,p2,p3,p4,p5,p6,p7
  p1 Allocation No
  p2 Enable or disable specification (On, Off)
  p3 Communication channel
  p4 Server registration number
  p5 Data group name (see "Description" and Appendix 6.)
  p6 Data name (see Appendix 6.)
  p7 Exponential scaling (-9 to 18), default value 0

**Query**
SWattData?,p1

**Example**
In allocation number 1, allocate the RMS voltage of element 1 of the WT1800 assigned to server registration number 2 to communication channel 003.
SWattData,1,On,003,2,Element1,URMS

**Description**
- The available data groups (p5) vary depending on the model.

<table>
<thead>
<tr>
<th>p5</th>
<th>Description</th>
<th>Supported Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Off</td>
<td>Unspecified</td>
<td>Yes Yes Yes</td>
</tr>
<tr>
<td>Element1</td>
<td>Element 1 data</td>
<td>Yes Yes Yes</td>
</tr>
<tr>
<td>Element2</td>
<td>Element 2 data</td>
<td>Yes Yes Yes</td>
</tr>
<tr>
<td>Element3</td>
<td>Element 3 data</td>
<td>Yes Yes Yes</td>
</tr>
<tr>
<td>Element4</td>
<td>Element 4 data</td>
<td>Yes — —</td>
</tr>
<tr>
<td>Element5</td>
<td>Element 5 data</td>
<td>Yes — —</td>
</tr>
<tr>
<td>Element6</td>
<td>Element 6 data</td>
<td>Yes — —</td>
</tr>
<tr>
<td>ElemHrm1</td>
<td>Element 1 harmonic data</td>
<td>Yes Yes Yes</td>
</tr>
<tr>
<td>ElemHrm2</td>
<td>Element 2 harmonic data</td>
<td>Yes Yes Yes</td>
</tr>
<tr>
<td>ElemHrm3</td>
<td>Element 3 harmonic data</td>
<td>Yes Yes Yes</td>
</tr>
<tr>
<td>ElemHrm4</td>
<td>Element 4 harmonic data</td>
<td>Yes — —</td>
</tr>
<tr>
<td>ElemHrm5</td>
<td>Element 5 harmonic data</td>
<td>Yes — —</td>
</tr>
<tr>
<td>ElemHrm6</td>
<td>Element 6 harmonic data</td>
<td>Yes — —</td>
</tr>
<tr>
<td>SigmaA</td>
<td>First wiring unit data</td>
<td>Yes Yes Yes</td>
</tr>
<tr>
<td>SigmaB</td>
<td>Second wiring unit data</td>
<td>Yes — —</td>
</tr>
<tr>
<td>SigmaC</td>
<td>Third wiring unit data</td>
<td>Yes — —</td>
</tr>
<tr>
<td>Other</td>
<td>Other types of data</td>
<td>Yes Yes Yes</td>
</tr>
<tr>
<td>DeltaA</td>
<td>First wiring unit delta math data</td>
<td>Yes — —</td>
</tr>
<tr>
<td>DeltaB</td>
<td>Second wiring unit delta math data</td>
<td>Yes — —</td>
</tr>
<tr>
<td>DeltaC</td>
<td>Third wiring unit delta math data</td>
<td>Yes — —</td>
</tr>
</tbody>
</table>
# Command and Response

## Commands and Responses

### SKdcCnct

**KDC Connection Destination (/AS)**
Sets the KDC server for the password management.

**Syntax**

```
SKdcCnct,p1,p2,p3
```

- **p1**: Connection destination (Primary, Secondary)
  - Primary: Primary server
  - Secondary: Secondary server
- **p2**: KDC server name (up to 64 characters, ASCII)
- **p3**: Port number (1 to 65535)

**Query**

```
SKdcCnct[,p1]?
```

**Example**

For the primary KDC server, assign the server name "KdcControl1" and port number 88.

```
SKdcCnct,Primary,KdcControl1,88
```

### SAuthKey

**Certification Key (/AS)**
Sets the certification key that is used during password management authentication.

**Syntax**

```
SAuthKey,p1,p2,p3,p4
```

- **p1**: Host principal (up to 20 characters, ASCII)
- **p2**: Realm name (up to 64 characters, ASCII)
- **p3**: Password (up to 20 characters, ASCII)
- **p4**: Encryption (ARC4, AES128, AES256)

**Query**

```
SAuthKey?
```

**Example**

Set the password of host principal "GX10_001" realm "REALM01" to "gDcbwT5," and the encryption (the same as the server) to AES128.

```
SAuthKey,GX10_001,REALM01,gDcbwT5,AES128
```

**Description**

- Slashes and ampersands cannot be used in p1 or p2.

### SDarwinCnvCh

**Darwin Channel Conversion (Darwin compatible communication)**
Replace Darwin channels with recorder channels.

**Syntax**

```
SDarwinCnvCh,p1
```

- **p1**: Darwin model
  - Standalone: Stand-alone type
  - Extension: Extended type

**Query**

```
SDarwinCnvCh?
```

**Example**

Replace DA100 stand-alone type channels to recorder channels.

```
SDarwinCnvCh,Standalone
```

### SDarwinPortLimit

**Port limitation of DARWIN compatible communication**
If port limitation is on, port number 34151 only allows reading of instantaneous data.

**Syntax**

```
SDarwinPortLimit,p1
```

- **p1**: Port limitation on/off (On/Off)
  - On: Instantaneous data reading only on 34151.
  - Off: No limitations on 34150 or 34151.

**Query**

```
SDarwinPortLimit?
```

**Example**

Set port number 34151 to instantaneous data reading only.

```
SDarwinPortLimit,On
```

**Description**

- The following applies when port limitation is set to On.
  - When connected via Ethernet
    Only instantaneous data reading is possible on port number 34151. Configuration, control, and operation are not possible.
  - When connected via serial communication
    Only the commands that can be used on a Ethernet connection through port 34150 are valid.

### SSLMPCList

**SLMP connection destination server (/E4)**
Sets the SLMP client connection destination server.

**Syntax**

```
SSLMPCList,p1,p2,p3
```

- **p1**: Connection destination number (1 to 16)
- **p2**: Server name (up to 64 characters, ASCII)
- **p3**: Port number (1 to 65535)

**Query**

```
SSLMPCList[,p1]?
```

**Example**

Connect to the server at connection destination number 1. Set the server name to "SMARTDAC" and the port number to 2020.

```
SSLMPCList,1,"SMARTDAC",2020
```
### SSLMPCCmd

**SLMP client transmission command (/E4)**

Sets the SLMP client transmission command.

**Syntax**

SSLMPCCmd,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13

- **p1** Command number (GX10/GP10: 1 to 50, GX20-1/GP20-1: 1 to 100, GX20-2/GP20-2/GM10-2: 1 to 200)
- **p2** Type
  - Off Disable command
  - Write Write a value to a Modbus register of another device.
  - Read Read a value from a Modbus register of another device.
- **p3** Connection destination number (1 to 16)
- **p4** See the device code table.
- **p5** Request destination station number (0 to 255)
- **p6** Request destination module I/O number (0 to 65535)
- **p7** Request destination multidrop station number (0 to 31)
- **p8** Data type
  - BIT Bit String data
  - INT16 16-bit signed integer
  - UINT16 16-bit unsigned integer
  - INT32 32-bit signed integer
  - UINT32 32-bit unsigned integer
  - FLOAT 32-bit floating point
- **p9** Head device number (0 to 16777215)
- **p10** Channel type
  - IO I/O channel
  - Math Math I/O channel (/IMT)
  - Com Communication I/O channel (/MC)
- **p11** First channel
- **p12** Last channel

**Query**

SSLMPCCmd[,p1]?

**Example**

Register: read the bit data assigned to head device number 1234 of the internal relay of the device at connection destination server number 1 from command number 1.

SSLMPCCmd,1,Read,1,M,1234,0,255,10 23,0,BIT,Com,0001,0001

---

### Device Code Table

<table>
<thead>
<tr>
<th>Device</th>
<th>Device Code (p4)</th>
<th>p5 to p8 , p10 Notation</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Special relay</td>
<td>SM</td>
<td>Hexadecimal, BIT</td>
<td></td>
</tr>
<tr>
<td>Special register</td>
<td>SD</td>
<td>Hexadecimal, INT16/UINT16/INT32/UINT32/ FLOAT</td>
<td></td>
</tr>
<tr>
<td>Input</td>
<td>X</td>
<td>Hexadecimal, BIT</td>
<td></td>
</tr>
<tr>
<td>Output</td>
<td>Y</td>
<td>Hexadecimal, BIT</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page...
2.4 Setting Commands

## Password Management (/AS)
Sets the password management.

**Syntax**
```
SKdc,p1,p2
```
- `p1`: Enable disable password management (On, Off)
- `p2`: Root user password (between 6 and 20 characters, ASCII)

**Query**
```
SKdc?
```

**Example**
Enable password management. Set the root user password to "root3210."
```
SKdc,On,root3210
```

## Password to Unlock Operation [GX/GP]
Sets the password that is used to release the operation lock.

**Syntax**
```
SOpePass,p1
```
- `p1`: Password (up to 20 characters, UTF-8)

**Query**
```
SOpePass?
```

**Example**
Set the password to "password1."
```
SOpePass,‘password1’
```

## Operation Lock Details [GX/GP]
Sets which operations to lock.

**Syntax**
```
SOpeLimit,p1,p2
```
- `p1`: Authority of user
- `p2`: Free/Lock

**Query**
```
SOpeLimit[,p1]?
```

**Example**
Prohibit operations for changing settings.
```
SOpeLimit,ChangeSet,Lock
```

**Description**
- You cannot use this command to configure settings when logged in as a user (when the user level is User).
2.4 Setting Commands

**SUser**

**User Settings**
Register users.

**Syntax**

```plaintext
SUser,p1,p2,p3,p4,p5,p6,p7
```

- **p1** User number
- **p2** User level
  - Off
  - Not Use
  - Admin
  - Administrator level
  - User
  - User level
- **p3** Login mode
  - (GX/GP: Key, Comm, Key+Comm)
  - (GM: Comm)
  - Key
    - Log in using touch operation
  - Comm
    - Log in via communication (including Web)
  - Key+Comm
    - Log in using touch operation and via communication.
- **p4** User name (up to 20 characters, ASCII)
- **p5** Password (up to 20 characters, ASCII)
- **p6** Enable or disable user limitation (Off, On)
- **p7** User limitation number (1 to 10)

**Query**

```plaintext
SUser[,p1]
```

The password of p5 are displayed using asterisks.

**Example**

Register a user-level user to user number 3. Set the user name to "user10" and the password to "pass012." Allow login only using touch operation, and specify user limitation number 5.

```plaintext
SUser,3,User,Key,'user10','pass012',On,5
```

**Description**

- If p1=1, p2 is fixed to Admin. In addition, you cannot set p3 to Comm.
- If p2=Admin, p6 and p10 are fixed to Off.
- If p2=Monitor, p6, p9, and p10 are fixed to Off.
- You cannot enter NULL or spaces in p4 or p5.
- You cannot use this command to configure settings when logged in as a user (when the user level is User).
- For the characters that you can use in the specified password (p5), see Appendix 1.
- Setting to enable password management (SKdc command)
  - If p2=Off, Admin, or User, p5 is invalid. The response to a query will be blank.
  - If p9 is fixed to Off.
- You can specify p5 only when p2=Monitor. When p2=Admin or User, you cannot specify p5 and the default password is enabled. For the default password, see section 2.3.1, “Logging In” in the Model GX10/GX20/GP10/GP20 Advanced Security Function (iAS) User’s Manual (IM 04L51B01-05EN) or section 2.3.1, “Logging In” in the Data Acquisition System GM Advanced Security Function (iAS) User’s Manual (IM 04L55B01-05EN).
- You cannot use this command to configure settings when logged in as a user (when the user level is User).

**When Using the Advanced Security Function (iAS)**

**Syntax**

```plaintext
SUser,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11
```

- **p1** User number
- **p2** User level
  - Off
  - Not use
  - Admin
    - Administrator level
  - User
    - User level
  - Monitor
    - Monitor level
- **p3** Login method
  - Key
    - Log in using touch operation
  - Comm
    - Log in via communication (including Web)
  - Key+Comm
    - Log in using touch operation and via communication.
- **p4** User name (up to 20 characters, ASCII)
- **p5** Password (between 6 and 20 characters, ASCII)
- **p6** Enable or disable user limitation (Off, On)
- **p7** User limitation number (1 to 10)
- **p8** User ID (up to 20 characters, ASCII)
  - Specify a user ID and password combination that have not been registered in the past.
- **p9** Password expiration (Off, 1Month, 3Month, 6Month)
- **p10** Enable or disable sign in property (Off, On)
- **p11** Sign in property number (1 to 8)

**Query**

```plaintext
SUser[,p1]
```

The password of p5 and user ID of p8 are displayed using asterisks.

**Example**

Register a user-level user to user number 3. Set the user name to "user10" and the password to "pass012." Allow login only using touch operation, and specify user limitation number 5.

```plaintext
SUser,3,User,Key,'user10','pass012',On,5
```

**Description**

- If p1=1, p2 is fixed to Admin. In addition, you cannot set p3 to Comm.
- If p2=Admin, p6 and p10 are fixed to Off.
- If p2=Monitor, p6, p9, and p10 are fixed to Off.
- You cannot enter NULL or spaces in p4 or p5.
- Setting to enable password management (SKdc command)
  - If p2=Off, Admin, or User, p5 is invalid. The response to a query will be blank.
  - If p9 is fixed to Off.
- You can specify p5 only when p2=Monitor. When p2=Admin or User, you cannot specify p5 and the default password is enabled. For the default password, see section 2.3.1, “Logging In” in the Model GX10/GX20/GP10/GP20 Advanced Security Function (iAS) User’s Manual (IM 04L51B01-05EN) or section 2.3.1, “Logging In” in the Data Acquisition System GM Advanced Security Function (iAS) User’s Manual (IM 04L55B01-05EN).
- For the characters that you can use in the specified password (p5), see Appendix 1.
- You cannot use this command to configure settings when logged in as a user (when the user level is User).
### SUserLimit

**Authority of User**
Sets user operation limitations.

**Syntax**
```
SUserLimit,p1,p2,p3
```
- **p1** Authority of user limitation number (1 to 10)
- **p2** Authority of user
  - Memory
  - Math
  - Data Save
  - Message
  - Batch
  - Alarm ACK
  - Comm
  - Disp Ope
  - Change Set
  - Date Set
  - File
  - System
  - Out
  - Calib Set
  - Ctrl In
  - Ctrl Out
  - Tuning
  - Local SP
  - Program
  - Calib Set
  - Free/Lock
- **p3** Free/Lock
  - Free
  - Not Lock
  - Lock

**Query**
```
SUserLimit[,p1]?
```

**Example**
Set user limitation number 1 so that changing settings is prohibited.
```
SUserLimit,1,ChangeSet,Lock
```

**Description**
- You cannot use this command to configure settings when logged in as a user (when the user level is User).

### SSginIn

**Sign In (/AS)**
Sets the sign in record for the measured data file.

**Syntax**
```
SSginIn,p1,p2,p3
```
- **p1** Sign in type (Batch, File)
- **p2** Sign in at record stop
  - (GX/GP: On, Off)
  - (GM: Fixed to Off)
- **p3** FTP transfer timing
  - (GX/GP: Data Save, Sign In)
  - (GM: Fixed to Data Save)

**Query**
```
SSginIn?
```

**Example**
When the recording of measured data of a batch process is stopped, switch to the screen for signing in.
```
SSginIn,Batch,On,Signin
```

### SSginInTitle

**Sign In Title (/AS)**
Sets the sign in title.

**Syntax**
```
SSginInTitle,p1,p2,p3
```
- **p1** Sign in title (up to 16 alphanumeric and symbol characters)
- **p2** Sign in 2 title (same as above)
- **p3** Sign in 3 title (same as above)

**Query**
```
SSginInTitle?
```

**Example**
Set the sign in 1, 2, and 3 titles to “Operator 1,” “Supervisor 1,” and “Manager 1,” respectively.
```
SSginInTitle,‘Operator 1’,‘Supervisor 1’,‘Manager 1’
```

### SSginInLimit

**Sign In Property (/AS)**
Sets the sign in property.

**Syntax**
```
SSginInLimit,p1,p2,p3,p4
```
- **p1** Sign in property number (1 to 8)
- **p2** Sign in 1 free/lock (Free, Lock)
- **p3** Sign in 2 free/lock (Free, Lock)
- **p4** Sign in 3 free/lock (Free, Lock)

**Query**
```
SSginInLimit[,p1]?
```

**Example**
Set a sign in property number 2 to allow the execution of only sign in 1.
```
SSginInLimit,1,Free,Lock,Lock
```
### SBTPassword

**Bluetooth Password (/C8) [GM]**
Sets the Bluetooth password.

**Syntax**

```
SBTPassword,p1,p2
```

- **p1** Password usage (On, Off)
- **p2** Password (up to 20 characters, ASCII)

**Query**

```
SBTPassword?
```

The password is displayed using asterisks.

**Example**

Set the password to “PaSswoRD2.”

```
SBTPassword,On,’PaSswoRD2’
```

**Description**

- You cannot use this command to configure settings when logged in as a user (when the user level is User).

---

### SWebCustomMenu

**Web Monitor Screen**
Shows or hides the categories displayed in the contents tree.

**Syntax**

```
SWebCustomMenu,p1,p2,p3,p4,p5
```

- **p1** User level (User, Monitor)
- **p2** Status display category
  - Show: On, hide: Off
- **p3** Log category
  - Show/hide (On, Off)
- **p4** System/Network information category
  - Show/hide (On, Off)
- **p5** File category
  - Show/hide (On, Off)

**Query**

```
SWebCustomMenu?
```

**Example**

Show the log category and file category for the user level.

```
SWebCustomMenu,User,Off,On,Off,On
```

**Description**

- This command can be used only when the user level is admin.
- When the advanced security function (/AS) is enabled, p1 is fixed to Monitor.

---

### SSessionSecurity

**Web Session Security Function (/AS) [GM]**
Sets the web session security function.

**Syntax**

```
SSessionSecurity,p1
```

- **p1** Session security (On, Off)

**Query**

```
SSessionSecurity?
```

**Example**

Use the session security function.

```
SSessionSecurity,On
```

**Description**

- This command can be executed only when the user is logged in as an administrator.

---

### SWebTimeOut

**Web Auto Logout (/AS) [GM]**
Sets the auto logout time for web screen.

**Syntax**

```
SWebTimeOut,p1
```

- **p1** Auto logout time (Off, 10 min, 20 min, 30 min)

**Query**

```
SWebTimeOut?
```

**Example**

Set the auto logout time to 10 minutes.

```
SWebTimeOut,10min
```

**Description**

- This command can be executed only when the user is logged in as an administrator.
**SMonitor**

**Monitor Screen Display Information [GX/GP]**
Sets the monitor screen display information.

**Syntax**

```plaintext
SMonitor, p1, p2
p1  Information type (see the table below)
p2  Status (see the table below)
```

<table>
<thead>
<tr>
<th>Information Type</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digital</td>
<td>Off, On</td>
</tr>
<tr>
<td>Message</td>
<td>Stream, List</td>
</tr>
<tr>
<td>Trend</td>
<td>Group, All</td>
</tr>
<tr>
<td>Grid</td>
<td>Off, On</td>
</tr>
<tr>
<td>Axis</td>
<td>1, 2, 3, 4, 5, 6, 7, 8</td>
</tr>
<tr>
<td>Value</td>
<td>4Value, Max, Min</td>
</tr>
<tr>
<td>Data</td>
<td>Disp, Event1</td>
</tr>
<tr>
<td>DigitalWave</td>
<td>Off, On</td>
</tr>
<tr>
<td>Alarm_Sort</td>
<td>Time, Channel, Level, Type</td>
</tr>
<tr>
<td>Alarm_Order</td>
<td>Ascending, Descending</td>
</tr>
<tr>
<td>Alarm_Time</td>
<td>Off, On</td>
</tr>
<tr>
<td>Message_Sort</td>
<td>Datetime, WriteTime, Message, Group, User</td>
</tr>
<tr>
<td>Message_Order</td>
<td>Ascending, Descending, Group, User</td>
</tr>
<tr>
<td>Memory_Data</td>
<td>Disp, Event1, Event2</td>
</tr>
<tr>
<td>Overview</td>
<td>Grouping, All</td>
</tr>
<tr>
<td>Multi_No</td>
<td>Multi panel number (1 to 20)</td>
</tr>
<tr>
<td>Custom_No</td>
<td>Customized display screen number (/CG)</td>
</tr>
<tr>
<td>DigitalPos</td>
<td>Default, Top, Bottom, Left, Right</td>
</tr>
<tr>
<td>DigitalLabel</td>
<td>Off, On</td>
</tr>
<tr>
<td>Modbus_M</td>
<td>Overview, List</td>
</tr>
<tr>
<td>Modbus_C</td>
<td>Overview, List</td>
</tr>
<tr>
<td>Watt</td>
<td>Overview, List</td>
</tr>
<tr>
<td>Switch</td>
<td>All, 1, 2, 3, 4</td>
</tr>
<tr>
<td>SLMP_C</td>
<td>Overview, List</td>
</tr>
<tr>
<td>ControlGroup</td>
<td>Controller, Faceplate</td>
</tr>
<tr>
<td>ControlAlarm_Sort</td>
<td>Time, Loop, Level, Type</td>
</tr>
<tr>
<td>ControlAlarm_Order</td>
<td>Ascending, Descending</td>
</tr>
<tr>
<td>ControlAlarm_Time</td>
<td>Off, On</td>
</tr>
</tbody>
</table>

**Example**

Set the trend display to all-channel display.

```
SMonitor, Trend, All
```

**Description**

- Custom_No is an option (/CG).
- Modbus_M and Modbus_C are an option (/MC).
- Watt is an option (/E2).
- When p1 = Switch, p2 = 3 or 4 is valid only for the GX10/GP10.

**SMultiPattern**

**Multi Panel Division [GX/GP]**
Sets the multi panel multi panel pattern.

**Syntax**

```plaintext
SMultiPattern, p1, p2, p3
p1  Registration number (1 to 20)
p2  Multi panel pattern
p3  Multi panel name (up to 16 characters, UTF-8)
```

**Example**

Set the panel of registration number 1 to “Split 2 Wide.” Set the multi panel name to “Monitor1.”

```
SMultiPattern, 1, Wide2, 'Monitor1'
```

**Description**

- This command is only valid for the GX20/GP20.
- This command can be used only when the multi batch function (/BT) is disabled.
2.4 Setting Commands

**SMultiKind**

**Multi Panel [GX/GP]**
Set the screens to display on the multi panel.

**Syntax**

```
SMultiKind, p1, p2, p3, p4
```

**p1** Registration number (1 to 20)

**p2** Screen position (1 to 6)

**p3** Screen type

- Trend
- Digital
- Bar
- Overview
- Alarm
- Message
- Memory
- Report
- Modbus-M
- Mosbus-C
- Watt
- Switch
- Action-
- Error-
- Commu-
- Ftp-
- Web-
- Mail-
- Modbus-
- Sntp-
- Dhcp-
- Network
- SLMP-C
- SLMP-Log
- Reminder
- ControlGroup
- ControlSummary
- ControlAlarmSummary
- ControlOverview

**p4** Group number

- If p3=Trend, Digital, or Bar: Display group number
- If p3=ControlGroup: Control group number

**Query**

```
SMultiKind[, p1[, p2]]?
```

**Example**

Display the bar graph of display group 8 in screen position 3 of the registration number 1 panel.

```
SMultiKind, 1, 3, Bar, 8
```

**Description**

- Custom display screen (/CG) cannot be shown in a multi panel.
- This command can be used only when the multi batch function (/BT) is disabled.
- ControlGroup, ControlSummary, ControlAlarmSummary, and ControlOverview are valid when the PID control module is installed.

**SHomeMonitor**

**Standard Screen Information [GX/GP]**
Sets the standard screen display information.

**Syntax**

```
SHomeMonitor, p1, p2
```

**p1** Information type (see the table of the SMonitor command)

**p2** Status (see the table of the SMonitor command)

**Query**

```
SHomeMonitor[, p1]?
```

**Example**

Set the trend display to all-channel display.

```
SHomeMonitor, Trend, All
```

**SHomeKind**

**Standard Screen [GX/GP]**
Set the standard screen.

**For Multi Panel**

**Syntax**

```
SHomeKind, p1, p2, p3
```

**p1** Screen type (Multi)

**p2** Multi panel number (1 to 20)

**p3** Batch group number (1 to the number used)

- p3 is valid when the multi batch function (/BT) is enabled.

**For Screens other than Multi Panel**

**Syntax**

```
SHomeKind, p1, p2, p3
```

**p1** Screen type

- Trend
- Digital
- Bar
- Overview
- Alarm
- Message
- Memory
- Report
- Modbus-M
- Mosbus-C
- Watt
- Switch
- Action-
- Error-
- Commu-
- Ftp-
- Web-
- Mail-
- Modbus-
- Sntp-
- Dhcp-
- Network
- SLMP-C
- SLMP-Log
- Reminder
- ControlGroup
- ControlSummary
- ControlAlarmSummary
- ControlOverview

**p4** Group number

- If p3=Trend, Digital, or Bar: Display group number
- If p3=ControlGroup: Control group number

**Query**

```
SMultiKind[, p1[, p2]]?
```

**Example**

Display the bar graph of display group 8 in screen position 3 of the registration number 1 panel.

```
SMultiKind, 1, 3, Bar, 8
```
### 2.4 Setting Commands

<table>
<thead>
<tr>
<th>Modbus-Log</th>
<th>Modbus log</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sntp-Log</td>
<td>SNTP log</td>
</tr>
<tr>
<td>Dhcp-Log</td>
<td>DHCP log</td>
</tr>
<tr>
<td>SLMP-Log</td>
<td>SLMP log (/E4)</td>
</tr>
<tr>
<td>Network</td>
<td>Network information</td>
</tr>
<tr>
<td>SLMP-C</td>
<td>SLMP client status (/E4)</td>
</tr>
<tr>
<td>Reminder</td>
<td>Reminder (/AH)</td>
</tr>
<tr>
<td>Setting</td>
<td>Settings</td>
</tr>
<tr>
<td>ControlGroup</td>
<td>Control group</td>
</tr>
<tr>
<td>ControlSummary</td>
<td>Control summary</td>
</tr>
<tr>
<td>ControlAlarm</td>
<td>Control alarm summary</td>
</tr>
<tr>
<td>ControlOverview</td>
<td>Control overview</td>
</tr>
<tr>
<td>SaveLoad</td>
<td>Save load</td>
</tr>
<tr>
<td>SystemInfo</td>
<td>System information</td>
</tr>
<tr>
<td>Custom</td>
<td>Customized display screen</td>
</tr>
<tr>
<td>Display</td>
<td></td>
</tr>
<tr>
<td>Batch</td>
<td>Batch overview (/BT)</td>
</tr>
<tr>
<td>Overview</td>
<td>Tuning</td>
</tr>
<tr>
<td>Tuning</td>
<td>Program selection (/PG)</td>
</tr>
<tr>
<td>ProgramSelect</td>
<td>Program operation (/PG)</td>
</tr>
<tr>
<td>ProgramRun</td>
<td></td>
</tr>
</tbody>
</table>

#### SFavoriteMonitor

**Favorite Screen Display Information [GX/GP]**  
Sets the favorite screen display information.  

**Syntax**  
SFavoriteMonitor,p1,p2,p3  
- **p1** Favorites number (1 to 20)  
- **p2** Information type (see the table of the SMonitor command)  
- **p3** Status (see the table of the SMonitor command)  

**Query**  
SFavoriteMonitor[,]?  

**Example**  
Set the trend display to all-channel display.  
SFavoriteMonitor,1,Trend,All

#### SFavoriteKind

**Favorite Screen [GX/GP]**  
Set the favorite screen.  

**For Multi Panel**  

**Syntax**  
SFavoriteKind,p1,p2,p3,p4,p5,p6  
- **p1** Favorites number (1 to 20)  
- **p2** Enable or disable (Off, On)  
- **p3** Screen type (Multi)  
- **p4** Multi panel number (1 to 20)  
- **p5** Panel name (up to 16 characters, UTF-8)  
- **p6** Batch group number (1 to the number used)  

**For Screens other than Multi Panel**  

**Syntax**  
SFavoriteKind,p1,p2,p3,p4,p5,p6  
- **p1** Favorites number (1 to 20)  
- **p2** Enable or disable (Off, On)  
- **p3** Screen type (see p1 of the SHomeKind command)  
- **p4** Display group number (when p3 is not CustomDisplay)  
  Customized display screen number (1 to 30)  
  Control group number (1 to 10)  
  Program pattern number (1 to 99)  
- **p5** Favorite screen name (up to 16 characters, UTF-8)  
- **p6** Multi batch number (All, 1 to the number used)  

**Query**  
SFavoriteKind[,]?  

**Example**  
Set the standard screen to trend of display group 1.  
SFavoriteKind,1,Trend,1

#### Description

- Report is an option (/MT).  
- Modbus-M and Modbus-C are an option (/IMC).  
- Watt is an option (/E2).  
- CustomDisplay is an option (/CG).  
- Multi is a GX20/GP20 display.  
- p3 is valid when the multi batch function (/BT) is enabled.  
- When the multi batch function (/BT) is not available, p3 is fixed to 1.  
- p1 cannot be set to BatchOverview when p3 is 1 to 12.  
- P1 cannot be set to Trend, Digital, Bar, Alarm, Message, Memory, or Multi when p3 = All.  
- p3 cannot be set to All when p1 is set to Trend, Digital, Bar, Alarm, Message, Memory, or Multi.  
- p3 cannot be set to 1 to 12 when p1 is set to BatchOverview.

- ControlGroup, ControlSummary, ControlAlarmSummary, ControlOverview, and Tuning are valid when the PID control module is installed.  
- ProgramSelect and ProgramRun are options (/PG).
Example
Register the trend display of display group 2 to favorites screen number 1. Set the screen name to “Favorite01.”
SFavoriteKind,1,On,Trend,2,’Favorite01’

Description
• When the multi batch function (/BT) is not available, p6 is fixed to 1.
• p1 cannot be set to BatchOverview when p3 is 1 to 12.
• p1 cannot be set to Trend, Digital, Bar, Alarm, Message, Memory, or Multi when p3 = All.
• ControlGroup, ControlSummary, ControlAlarmSummary, ControlOverview, and Tuning are valid when the PID control module is installed.
• ProgramSelect and ProgramRun are options (/PG).

SMltTextField
Batch Text
Sets the batch text field for multi batch.
Syntax
SMltTextField,p1,p2,p3,p4
  p1  Batch group number (1 to the number used)
  p2  Field number (1 to 24)
  p3  Title (up to 20 characters, UTF-8)
  p4  Character string (up to 30 characters, UTF-8)
Query
SMltTextField[,p1[,p2]]?
Example
For field number 3 of batch group 2, set the field title to “OPERATOR” and the character string to “RECODER1.”
SMltTextField,2,3,’OPERATOR’,’RECODER1’

Description
• This command cannot be used if the batch setting is disabled (SBatch: p1=Off).
• This command is the same as STextField when p1=1.

SMltFileHead
File Header
Sets the file header for multi batch.
Syntax
SMltFileHead,p1,p2
  p1  Batch group number (1 to the number used)
  p2  File header (up to 50 characters, UTF-8)
Query
SMltFileHead[,p1]?
Example
Set the file header of batch group 2 to “GX_DATA.”
SMltFileHead,2,’GX_DATA’

Description
• This command is the same as SFileHead when p1=1.

SMltFileName
File Naming Rule
Sets the file naming rule for saving multi batch data.
Syntax
SMltFileName,p1,p2,p3
  p1  Batch group number (1 to the number used)
  p2  File naming rule
    Date Date
    Serial Serial number
    Batch Batch name
  p3  Specified file name (up to 16 characters, ASCII)
Query
SMltFileName[,]?
Example
Set the file naming rule of batch group 2 to “Date.”
SMltFileName,2,Date,’Recorder1_data’

Description
• This command is the same as SFileName when p1=1.
• If the batch setting is disabled (SBatch: p1=Off), p2 cannot be set to Batch.

SMltGroup
Display Group
Sets the display group for multi batch.
Syntax
SMltGroup,p1,p2,p3,p4,p5
  p1  Batch group number (1 to the number used)
  p2  Display group number
  p3  Enable or disable (Off, On)
  p4  Group name (up to 16 characters, UTF-8)
  p5  Channel character string
    • Specify using channel numbers. 4-digit numbers for I/O channels. Numbers that start with “A” for math channels (A015). Numbers that start with “C” for communication channels (C020). The maximum number of characters per channel is 4.
    • Use periods to separate channel numbers (see example).
Query
SMltGroup[,p1[,p2]]?
The channel string is output exactly as it is specified.
Example
Assign channels 0001, 0003, 0005, A001, and C023 to display group 1 of batch group 3 and name it “GROUP A.”
SMltGroup,3,1,On,’GROUP A’1.3.5.A1.C23’

Description
• This command is the same as SGroup when p1=1.
### SMltTripLine

**Display Group Trip Line**
Sets the display group trip line for multi batch.

**Syntax**

\[ \text{SMltTripLine}, p1, p2, p3, p4, p5, p6, p7, p8, p9 \]

- **p1**: Batch group number (1 to the number used)
- **p2**: Display group number
- **p3**: Trip line number (1 to 4)
- **p4**: Enable or disable (Off, On)
- **p5**: Display position [%] (1 to 100)
- **p6**: R value of RGB display colors (0 to 255)
- **p7**: G value of RGB display colors (0 to 255)
- **p8**: B value of RGB display colors (0 to 255)
- **p9**: Line width
  (GX/GP: Thin, Normal, Thick)
  (GM: Normal)
  - Thin
  - Normal
  - Thick

**Query**

\[ \text{SMltTripLine}, [p1, [p2, [p3]]] \]

**Example**
Display trip line 1 using a thick line in red at the 80% position of display group 2 of batch group 3.

\[ \text{SMltTripLine}, 3, 2, 1, 80, 255, 0, 0, \text{Thin} \]

### SMltSclBmp [GX/GP]

**Scale Bitmap**
Sets the display group’s scale bitmap file for multi batch.

**Syntax**

\[ \text{SMltSclBmp}, p1, p2, p3 \]

- **p1**: Batch group number (1 to the number used)
- **p2**: Display group number
- **p3**: Enable or disable (Off, On)

**Query**

\[ \text{SMltSclBmp}, [p1, [p2]] \]

**Example**
Use a bitmap scale image on display group 2 of batch group 2.

\[ \text{SMltSclBmp}, 2, 3, \text{On} \]

### SMltMultiPattern

**Multi Panel Pattern**
Sets the multi panel pattern for multi batch.

**Syntax**

\[ \text{SMltMultiPattern}, p1, p2, p3, p4 \]

- **p1**: Batch group number (1 to the number used)
- **p2**: Registration number (1 to 20)
- **p3**: Division pattern
  - Wide2
  - Tall2
  - Wide3
  - Tall3
  - Split4
  - Even5
  - Odd5
  - Even6
  - Odd6
- **p4**: Panel name (up to 16 characters, UTF-8)

**Query**

\[ \text{SMltMultiPattern}, [p1, [p2, [p3]]] \]

**Example**
Set batch group 2. Set the panel of registration number 1 to “Split 2 Wide.” Set the panel name to “Monitor1.”

\[ \text{SMltMultiPattern}, 2, 1, \text{Wide2}, \text{Monitor1} \]

### SMltMultiKind

**Multi Panel Type**
Sets the multi panel pattern for multi batch.

**Syntax**

\[ \text{SMltMultiKind}, p1, p2, p3, p4, p5 \]

- **p1**: Batch group number (1 to the number used)
- **p2**: Registration number (1 to 20)
- **p3**: Screen position (1 to 6)
- **p4**: Screen type (see p1 of the SMultiKind command)
- **p5**: Display group number

**Query**

\[ \text{SMltMultiKind}, [p1, [p2, [p3]]] \]

**Example**
Set the panel of registration number 1 to “Split 2 Wide.” Set the panel name to “Monitor1.”

\[ \text{SMultiPattern}, 1, \text{Wide2}, \text{Monitor1} \]
2.4 Setting Commands

**SBTID**
Bluetooth Communication ID (/C8) [GM]
Sets the Bluetooth communication ID.

**Syntax**

```
SBTID,p1
```

```
p1  Local device name (GM’s Bluetooth device name)
   Up to 30 characters, ASCII
```

**Query**

```
SBTID?
```

**Example**
Set the local device name to “SMARTDAC+ GM.”

```
SBTID,'SMARTDAC+ GM'
```

**Description**

• Users logged in via USB communication can be automatically logged out.

**SBTTimeOut**
Bluetooth Communication Timeout (/C8) [GM]
Sets the Bluetooth communication timeout.

**Syntax**

```
SBTTimeOut,p1
```

```
p1  Timeout function (Off, 1min, 2min, 5min, 10min)
```

**Query**

```
SBTTimeOut?
```

**Example**
Set the Bluetooth communication timeout value to 5 minutes.

```
SBTTimeOut,5min
```

**Description**

• If the login function is in use, users that are logged in are automatically logged out when a timeout occurs.

**SUsbFunction**
USB Communication Function [GM]
Configures USB communication function settings.

**Syntax**

```
SUsbFunction,p1
```

```
p1  USB communication function On/Off (On, Off)
   On    Use
   Off   Not Use
```

**Query**

```
SUsbFunction?
```

**Example**
Use the USB communication function.

```
SUsbFunction,On
```

**Description**

• This command can be executed only when the user is logged in as an administrator.
  • p1=Monitor is valid when the advanced security function (/AS) is enabled.

**SSchedule**
Schedule Management
Configures the schedule management function.

**Syntax**

```
SSchedule,p1,p2,p3,p4,p5,p6,p7,p8,p9
```

```
p1  Schedule number
p2  Schedule management function On/Off
    On    Use
    Off   Not Use
p3  Date  Year (2001 to 2035)
p4  Date  Month (1 to 12)
p5  Date  Day (1 to 31)
```

**Example**
Set the schedule number to 1, and enable the schedule management function.

```
SSchedule,1,Use,2001,01,01
```

**Description**

• This command can be executed only when the user is logged in as an administrator.
### SDualGroup

**Scan group number of the display group**

If the measurement mode is set to dual interval, set the scan group of the display group.

**Syntax**

```
SDualGroup,p1,p2
```

- **p1**: Group number
- **p2**: Scan group number (1 or 2)

**Query**

```
SDualGroup[?,p1]
```

**Example**

Set master scan interval to scan group 2.

```
SDualGroup,5,2
```

**Description**

- You can set this command when the dual interval function is enabled.

---

### SMasterScanGrp

**Master Scan Interval**

Sets the master scan interval when the measurement mode is set to dual interval.

**Syntax**

```
SMasterScanGrp,p1
```

- **p1**: Scan group number (1 or 2)

**Query**

```
SMasterScanGrp[?,p1]
```

**Example**

Set display group 5 to scan group 2.

```
SMasterScanGrp,2
```

**Description**

- You can set this command when the dual interval function is enabled.

---

### SCtrlMode

**Control Mode**

Sets the control mode.

**Syntax**

```
SCtrlMode,p1,p2,p3,p4
```

- **p1**: Unit number
- **p2**: Module number
- **p3**: Mode (Single, Cascade, PVSwitching)
  - **Single**: Single loop control
  - **Cascade**: Cascade control
  - **PVSwitching**: PV switching
- **p4**: PV switching condition (when p3 is PVSwitching)
  - **LowRange**: Switch within range (Low side)
  - **HighRange**: Switch within range (High side)
  - **PVHigh**: Switch at PV high limit
  - **Signal**: Switch using DI

**Query**

```
SCtrlMode[?,p1,p2]
```

**Example**

Set the control mode of the PID module with module number 2, connected to the main unit, to PV switching and the switching condition to switch using DI.

```
SCtrlMode,2,1,Single,LowRange
```

---

### SScheduleText

**Schedule Management Text**

Sets the schedule management title and notification content.

**Syntax**

```
SScheduleText,p1,p2,p3,p4
```

- **p1**: Schedule number (1 to 12)
- **p2**: Title (32 characters)
- **p3**: Notification content 1 (32 characters)
- **p4**: Notification content 2 (32 characters)

**Query**

```
SScheduleText[?,p1]
```

**Example**

For schedule number 1, set the title to “Calibration correction” and notification content 1 to “Sensor correction coefficient.”

```
SScheduleText,1,"Calibration correction", "Sensor coefficient"
```

**Description**

- This command can be used on a product with the aerospace heat treatment (/AH) option.
- Operation lock ChangeSet and user privileges CalibSet limitations apply to this command.

---

**Queries**

- Operation lock ChangeSet and user privileges CalibSet limitations apply to this command.

---

**Example**

Set schedule number 1 with the date set to December 24, 2015, the notification set to 5 days before, and the renotification interval to 1 hour. Enable the notification buzzer. Show the calibration correction settings in the date setting screen.

```
SSchedule,1,On,2015,12,24,5day,1h,On,On
```

**Description**

- This command can be used on a product with the aerospace heat treatment (/AH) option.
- Operation lock ChangeSet and user privileges CalibSet limitations apply to this command.
2.4 Setting Commands

**SCtrlMode,0,2,PVSwitching,Signal**

**Description**
- This command is valid when a PID Control Module is installed.

**SCtrlScan**

**Control Period**
Sets the control period.

**Syntax**
```
SCtrlScan,p1,p2,p3
```
- `p1` Unit number
- Fixed to “-”.
- `p2` Module number
- Fixed to “-”.
- `p3` Control period (100ms, 200ms)

**Query**
```
SCtrlScan?
```

**Example**
Set the control period to 100ms.
```
SCtrlScan,-,-,100ms
```

**Description**
- This command is valid when a PID Control Module is installed.

**SCtrlAction**

**Control**
Sets the control.

**Syntax**
```
SCtrlAction,p1,p2,p3
```
- `p1` Unit number
- `p2` Module number
- `p3` Power recovery action
  - RESTART_CONT: Continue
  - RESTART_MAN: Manual
  - RESTART_AUTO: Auto
  - RESTART_RESET: Reset

**Query**
```
SCtrlAction[,p1,p2]
```

**Example**
Set the power recovery action of module number 2 connected to the main unit to manual.
```
SCtrlAction,0,2,RESTART_MAN
```

**Description**
- This command is valid when a PID Control Module is installed.
- You can set `p3` to RESTART_RESET when the program control function (/PG option) is enabled.

**SCtrlType**

**Control Type**
Sets the control type.

**Syntax**
```
SCtrlType,p1,p2
```
- `p1` Loop number
- `p2` Control Type
  - PID
  - ON/OFF

**Query**
```
SCtrlType[,p1]?
```

**Example**
Set the control type of loop number L022 to ON/OFF control.
```
SCtrlType,L022,ONOFF
```

**Description**
- This command is valid when a PID Control Module is installed.

**SCtrlLoopAction**

**Loop Control**
Sets the loop control.

**Syntax**
```
SCtrlLoopAction,p1,p2,p3,p4,p5,p6
```
- `p1` Loop number
- `p2` PID initial value
  - PIDDef_TEMP: Temperature
  - PIDDef_PRESS: Pressure flow rate
- `p3` PID selection
  - SP_SLCT: Target setpoint selection
    (when the /PG option is not installed)
  - SP_SEG_SLCT: Target setpoint selection/segment
    PID method (when the /PG option is not installed)
  - ZON_PV: Zone PID method (PV input)
  - ZON_TSP: Zone PID method (final target setpoint)
  - ZON_SP: Zone PID method (target setpoint)
  - FIX_PID: Fixed local PID selection (PID number switching with event action)
- `p4` EXPV function
  - Off: EXPV (RPV) is used.
  - On: EXPV (RPV) is not used.
- `p5` RSP function
  - Off: RSP is not used.
  - On: RSP is used.
- `p6` PID control mode
  - FollowUp: Standard PID control mode
  - Fixed-point: Fixed-point control mode

**Query**
```
SCtrlLoopAction[,p1]?
```

**Example**
Set the loop number L022 control as follows:
- PID initial value: Temperature
- PID selection: Zone PID method (PV input)
- EXPV (RPV) function: Off
- RSP function: On
- PID control mode: Standard PID control mode
```
SCtrlLoopAction,L022,PIDDef_TEMP,ZON_PV,Off,On,FollowUp
```

**Description**
- This command is valid when a PID Control Module is installed.
• The RSP function of the secondary side cannot be set to On (p5=On) in cascade mode.
• The available options for p3 (PID selection) varies depending on whether the program control function (/PG option) is available.

### 2.4 Setting Commands

#### SCtrlSPPID

**Number of SP Groups, Number of PID Groups**

Sets the number of SP groups, number of PID groups.

**Syntax**

```
SCtrlSPPID, p1, p2, p3
```

- **p1**: Loop number
- **p2**: SP group number (1 to 8)
- **p3**: PID group number (1 to 8)

**Query**

```
SCtrlSPPID[,p1]
```

**Example**

For loop number L022, set the number of SP groups to 6 and the number of PID groups to 3.

```
SCtrlSPPID,L022,6,3
```

**Description**

- **This command is valid when a PID Control Module is installed.**

#### SCtrlALNo

**Number of Control Alarms**

Sets the number of control alarms.

**Syntax**

```
SCtrlALNo, p1, p2
```

- **p1**: Loop number
- **p2**: Number of control alarms (1 to 4)

**Query**

```
SCtrlALNo[,p1]
```

**Example**

For loop number L022, set the number of control alarms to 3.

```
SCtrlALNo,L022,3
```

**Description**

- **This command is valid when a PID Control Module is installed.**

#### SCtrlAlmMode

**Alarm Mode**

Sets the alarm mode.

**Syntax**

```
SCtrlAlmMode, p1, p2
```

- **p1**: Loop number
- **p2**: Alarm mode
  - ALM_MODE_ALWAYS: Always active
  - ALM_MODE_STOP: Not active in STOP mode
  - ALM_MODE_STOP_MAN: Not active in STOP or MAN mode

**Query**

```
SCtrlAlmMode[,p1]
```

**Example**

For loop number L022, set the alarm mode to always active.

```
SCtrlAlmMode,L022,ALM_MODE_ALWAYS
```

**Description**

- **This command is valid when a PID Control Module is installed.**

#### SCtrlDIRegist

**Contact Registration**

Registers a contact.

**Syntax**

```
SCtrlDIRegist, p1, p2, p3, p4
```

- **p1**: Unit number
- **p2**: Module number
- **p3**: Action
  - A-M_LP1: AUTO/MAN Switch (A/M) LP1
  - A-M_LP2: AUTO/MAN Switch (A/M) LP2
  - R-L_LP1: REMOTE/LOCAL Switch (R/L) LP1
  - R-L_LP2: REMOTE/LOCAL Switch (R/L) LP2
  - S-R_LP1: STOP/RUN Switch (S/R) LP1
  - S-R_LP2: STOP/RUN Switch (S/R) LP2
  - CAS: Switch to Cascade
  - AUTO_LP1: Switch to AUTO (AUTO) LP1
  - AUTO_LP2: Switch to AUTO (AUTO) LP2
  - MAN_LP1: Switch to MAN (MAN) LP1
  - MAN_LP2: Switch to MAN (MAN) LP2
  - REM_LP1: Switch to REMOTE (REM) LP1
  - REM_LP2: Switch to REMOTE (REM) LP2
  - LCL_LP1: Switch to LOCAL (LCL) LP1
  - LCL_LP2: Switch to LOCAL (LCL) LP2
  - AT_LP1: Auto-tuning START/STOP Switch (AT) LP1
  - AT_LP2: Auto-tuning START/STOP Switch (AT) LP2
  - SW: PV switching (SW)
  - ACK_LP1: Alarm ACK (ACK) LP1
  - ACK_LP2: Alarm ACK (ACK) LP2
  - SPBit0_LP1 to SPBit3_LP1: Bit-0 to Bit-3 of SP Number LP1
  - PIDBit0_LP1 to PIDBit3_LP1: Bit-0 to Bit-3 of PID Number LP1
  - ALM1_L1 to ALM4_L1: Alarm status loop 1
  - ALO1_L1 to ALO4_L1: Alarm output loop 1
  - ALM1_L2 to ALM4_L2: Alarm status loop 2
  - ALO1_L2 to ALO4_L2: Alarm output loop 2

- **p4**: Terminal number
  - Off: No registration
  - DI1 to DI8: Contact input
  - DO1 to DO8: Contact output
  - ALM1_L1 to ALM4_L1: Alarm status loop 1
  - ALM1_L2: Alarm status loop 2
  - ALO1_L1: Alarm output loop 1
  - ALO1_L2: Alarm output loop 2

**Example**

For loop number L022, set the alarm mode to always active.

```
SCtrlDIRegist,L022,6,3
```
2.4 Setting Commands

**Query** `SCtrlDIRegist[,p1,p2,p3]?`

**Example** Set the AUTO/MAN Switch (A/M) LP1 of module number 2 connected to the main unit to DI1.

`SCtrlDIRegist,0,2,A-M_LP1,DI1`

**Description**
- This command is valid when a PID Control Module is installed.
- `ALM*_Lx(*=1 to 4, x=1 or 2)` indicates the alarm status. `ALM*_OUT_Lx(*=1 to 4, x=1 or 2)` indicates the alarm output status including the relay action.
- The valid range of `p4` options varies depending on the `p3` (action) setting. For details, see the table below ("Validity of `p3` settings and `p4` terminal numbers").

### Validity of `p3` settings and `p4` terminal numbers

<table>
<thead>
<tr>
<th><code>p3</code> action</th>
<th><code>p4</code> options</th>
<th>Description</th>
<th>Option</th>
<th>Off</th>
<th>D11 to D18</th>
<th>D11 to D18</th>
<th>D11 to D18</th>
<th>D11 to D18</th>
<th>Conditions in which <code>p3</code> is valid when <code>p4</code> is not Off</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTO/MAN Switch (A/M) LP1</td>
<td>A-M_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Valid when the control mode is not cascade</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTO/MAN Switch (A/M) LP2</td>
<td>A-M_LP2</td>
<td>x</td>
<td>—</td>
<td>x</td>
<td>Valid when the control mode is not PV switching</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REMOTE/LOCAL Switch (R/L) LP1</td>
<td>R-L_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Valid when the RSP function is on (see odd loops)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REMOTE/LOCAL Switch (R/L) LP2</td>
<td>R-L_LP2</td>
<td>x</td>
<td>—</td>
<td>x</td>
<td>Valid when the RSP function is on (see even loops)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STOP/RUN Switch (S/R) LP1</td>
<td>S-R_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Always valid</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STOP/RUN Switch (S/R) LP2</td>
<td>S-R_LP2</td>
<td>x</td>
<td>—</td>
<td>x</td>
<td>Valid when the control mode is not PV switching</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switch to Cascade (CAS)</td>
<td>CAS</td>
<td>x</td>
<td>—</td>
<td>x</td>
<td>Valid when the control mode is cascade</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switch to AUTO (AUTO) LP1</td>
<td>AUTO_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Valid when the control mode is not cascade</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switch to AUTO (AUTO) LP2</td>
<td>AUTO_LP2</td>
<td>x</td>
<td>—</td>
<td>x</td>
<td>Valid when the control mode is not PV switching</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switch to MAN (MAN) LP1</td>
<td>MAN_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Valid when the control mode is not cascade</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switch to LOCAL (LCL) LP1</td>
<td>LCL_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Valid when the RSP function is on (see odd loops)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Switch to LOCAL (LCL) LP2</td>
<td>LCL_LP2</td>
<td>x</td>
<td>—</td>
<td>x</td>
<td>Valid when the RSP function is on (see even loops)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Auto-tuning START/STOP Switch (AT) LP1</td>
<td>AT_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Always valid</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Auto-tuning START/STOP Switch (AT) LP2</td>
<td>AT_LP2</td>
<td>x</td>
<td>—</td>
<td>x</td>
<td>Valid when the control mode is not PV switching</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PV switching (SW)</td>
<td>SW</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Valid when the control mode is PV switching and the input switching action is contact.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alarm ACK (ACK) LP1</td>
<td>ACK_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Always valid</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alarm ACK (ACK) LP2</td>
<td>ACK_LP2</td>
<td>x</td>
<td>—</td>
<td>x</td>
<td>Valid when the control mode is not PV switching</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit-0 of SP Number LP1</td>
<td>SPBit0_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Always valid</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit-1 of SP Number LP1</td>
<td>SPBit1_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Always valid</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bit-2 of SP Number LP1</td>
<td>SPBit2_LP1</td>
<td>x</td>
<td>x</td>
<td>—</td>
<td>Always valid</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Commands and Responses

<table>
<thead>
<tr>
<th>p3 action</th>
<th>Option</th>
<th>p4 options</th>
<th>Conditions in which p3 is valid when p4 is not Off</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit-3 of SP Number LP1</td>
<td>SPBit3_ LP1</td>
<td>x x —</td>
<td>Always valid</td>
</tr>
<tr>
<td>Bit-0 of SP Number LP2</td>
<td>SPBit0_ LP2</td>
<td>x — x</td>
<td>Valid when the control mode is not PV switching</td>
</tr>
<tr>
<td>Bit-1 of SP Number LP2</td>
<td>SPBit1_ LP2</td>
<td>x — x</td>
<td>Valid when the control mode is not PV switching</td>
</tr>
<tr>
<td>Bit-2 of SP Number LP2</td>
<td>SPBit2_ LP2</td>
<td>x — x</td>
<td>Valid when the control mode is not PV switching</td>
</tr>
<tr>
<td>Bit-3 of SP Number LP2</td>
<td>SPBit3_ LP2</td>
<td>x — x</td>
<td>Valid when the control mode is not PV switching</td>
</tr>
<tr>
<td>Bit-0 of PID Number LP1</td>
<td>PIDBit0_ LP1</td>
<td>x x —</td>
<td>Valid when the PID selection is external selection (see odd loop settings)</td>
</tr>
<tr>
<td>Bit-1 of PID Number LP1</td>
<td>PIDBit1_ LP1</td>
<td>x x —</td>
<td>Valid when the PID selection is external selection (see odd loop settings)</td>
</tr>
<tr>
<td>Bit-2 of PID Number LP1</td>
<td>PIDBit2_ LP1</td>
<td>x x —</td>
<td>Valid when the PID selection is external selection (see odd loop settings)</td>
</tr>
<tr>
<td>Bit-3 of PID Number LP1</td>
<td>PIDBit3_ LP1</td>
<td>x x —</td>
<td>Valid when the PID selection is external selection (see odd loop settings)</td>
</tr>
<tr>
<td>Bit-0 of PID Number LP2</td>
<td>PIDBit0_ LP2</td>
<td>x — x</td>
<td>Valid when the PID selection is external selection (see even loop settings)</td>
</tr>
<tr>
<td>Bit-1 of PID Number LP2</td>
<td>PIDBit1_ LP2</td>
<td>x — x</td>
<td>Valid when the PID selection is external selection (see even loop settings)</td>
</tr>
</tbody>
</table>
### SCtrlRelay

**DO Terminal Action (Relay Action)**

Sets the DO terminal action (relay action) of a PID module.

When the output is “contact output within module”

**Syntax**

```
SCtrlRelay,p1,p2,p3,p4,p5,p6,p7
```

- **p1** Unit number
- **p2** Module number
- **p3** DO number (DO1 to DO8)
- **p4** Output type
  - `CtrlRelay`: Contact output within module
- **p5** Loop selection
  - `LP1`: Loop 1
  - `LP2`: Loop 2
  - `COMMON`: Common to loop 1 and loop 2
- **p6** Status (see the table below)
  - The valid range of p6 (status) varies depending on the p5 (loop) options.
- **p7** Energize/De-energize (Energize, De-Energize)

<table>
<thead>
<tr>
<th>p6 (status)</th>
<th>Content</th>
<th>p5 (loop)</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>LP1, LP2, COMMON</td>
</tr>
<tr>
<td>ALM1_OUT_L1</td>
<td>Alarm 1 status loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>ALM2_OUT_L1</td>
<td>Alarm 2 status loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>ALM3_OUT_L1</td>
<td>Alarm 3 status loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>ALM4_OUT_L1</td>
<td>Alarm 4 status loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>ALM1_L1</td>
<td>Alarm 1 loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>ALM2_L1</td>
<td>Alarm 2 loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>ALM3_L1</td>
<td>Alarm 3 loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>ALM4_L1</td>
<td>Alarm 4 loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>SR_L1</td>
<td>STOP/RUN loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>AM_L1</td>
<td>AUTO/MAN loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>RL_L1</td>
<td>REMOTE/LOCAL loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>AT_L1</td>
<td>Auto-tuning status loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>EXPV_ANAPV_L1</td>
<td>EXPV/LOCAL loop 1</td>
<td>LP1</td>
</tr>
<tr>
<td>ALM1_OUT_L2</td>
<td>Alarm 1 status loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>ALM2_OUT_L2</td>
<td>Alarm 2 status loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>ALM3_OUT_L2</td>
<td>Alarm 3 status loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>ALM4_OUT_L2</td>
<td>Alarm 4 status loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>ALM1_L2</td>
<td>Alarm 1 loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>ALM2_L2</td>
<td>Alarm 2 loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>ALM3_L2</td>
<td>Alarm 3 loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>ALM4_L2</td>
<td>Alarm 4 loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>SR_L2</td>
<td>STOP/RUN loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>AM_L2</td>
<td>AUTO/MAN loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>AUTO_L2</td>
<td>AUTO loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>MAN_L2</td>
<td>MAN loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>CAS_L2</td>
<td>Cascade loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>RL_L2</td>
<td>REMOTE/LOCAL loop 2</td>
<td>LP2</td>
</tr>
<tr>
<td>AT_L2</td>
<td>Auto-tuning status</td>
<td>LP2</td>
</tr>
<tr>
<td>EXPV_ANAPV_L2</td>
<td>EXPV/LOCAL loop 2</td>
<td>LP2</td>
</tr>
</tbody>
</table>

When the output is “Alarm” and the action is “And/Or”

**Syntax**

```
SCtrlRelay,p1,p2,p3,p4,p5,p6,p7,p8
```

- **p1** Unit number
- **p2** Module number
- **p3** DO number (DO1 to DO8)
- **p4** Output type
  - `Alarm`
- **p5** Energize/De-energize (Energize, De-Energize)
- **p6** Action (And, Or)
  - `And`: Operate when all set alarms are in the alarm state.
  - `Or`: Operate when any of the set alarms are in the alarm state.
- **p7** State
  - `Hold`
  - `Nonhold`
- **p8** Relay Action on ACK (Normal, Reset)

When the output is “Alarm” and the action is “Reflash”

**Syntax**

```
SCtrlRelay,p1,p2,p3,p4,p5,p6,p7,p8
```

- **p1** Unit number
- **p2** Module number
- **p3** DO number (DO1 to DO8)
- **p4** Output type
  - `Alarm`
- **p5** Energize or de-energize
  - `Energize`
  - `De-Energize`
- **p6** Action
  - `Reflash`
- **p7** Reflash time (500ms, 1s, 2s)
- **p8** Relay Action on ACK (Normal, Reset)

When the output is “Manual”

**Syntax**

```
SCtrlRelay,p1,p2,p3,p4,p5
```

- **p1** Unit number
- **p2** Module number
- **p3** DO number (DO1 to DO8)
- **p4** Output type
  - `Manual`
- **p5** Energize or de-energize
  - `Energize`
  - `De-Energize`

**Query**

```
SCtrlRelay[,p1,p2,p3]?
```
Example
Set DO1 of module number 2 connected to the main unit to manual and energize.
SCtrlRelay,0,2,DO1,Manual,Energize

Description
• This command is valid when a PID Control Module is installed.
• If p4=CtrlRelay, p6 alarm level is indicated as ALM* – Lx(*=1 to 8, x=1 or 2). ALM* _OUT_Lx(*=1 to 8, x=1 or 2) indicates the alarm output status including the relay action.

SCtrlRangeAI
Measurement Input Range
Sets the range of the AI terminal of a PID control module.
Input type is TC or RTD
Syntax
SCtrlRangeAI,p1,p2,p3,p4,p5,p6,p7,p8,p9
p1 Unit number
p2 Module number
p3 AI terminal number (AI1, AI2)
Input type is not TC or RTD and math type is Scaling
Syntax
SCtrlRangeAI,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13
Input type is GS or Volt and math type is square root
Syntax
SCtrlRangeAI,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13,p14,p15,p16

Query
Example
Measure -0.5000 to 1.0000 V on channel 0002.
No scaling. No bias.
SCtrlRangeAI,0002,Volt,2V,Off,–5000,10000,0

Description
• This command is valid when a PID Control Module is installed.
• You cannot use this command to configure settings while recording is in progress.
• You cannot use this command to configure settings while computation is in progress.
• If p4=TC/RTD, p6 is set to Off.
• If p4=TC/RTD/DI, p6 cannot be set to Sqrt.
• If p4=Volt/GS/DI, p6 cannot be set to Off.
• If p4=DI, you cannot set p9 (bias).
• The settable items for p5 are shown in the table below.

<table>
<thead>
<tr>
<th>settings</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>20mV R Pt100</td>
<td>1-5V Level</td>
</tr>
<tr>
<td>60mV S Pt100-H</td>
<td>0.4-2V DI</td>
</tr>
<tr>
<td>200mV J Ptf100</td>
<td></td>
</tr>
<tr>
<td>6V E Cu10LN</td>
<td></td>
</tr>
<tr>
<td>20V J Cu10WEED</td>
<td></td>
</tr>
<tr>
<td>50V T Cu10BAILEY</td>
<td></td>
</tr>
<tr>
<td>N Cu10a392</td>
<td></td>
</tr>
<tr>
<td>W Cu10a393</td>
<td></td>
</tr>
<tr>
<td>L Cu25</td>
<td></td>
</tr>
<tr>
<td>U Cu53</td>
<td></td>
</tr>
<tr>
<td>PLATINEL Cu100</td>
<td></td>
</tr>
<tr>
<td>PR20-40 J263B</td>
<td></td>
</tr>
<tr>
<td>WRe3-25 N1100SAMA</td>
<td></td>
</tr>
<tr>
<td>KpvsAu7Pe Ni1000IN</td>
<td></td>
</tr>
<tr>
<td>NiNiMo Ni120</td>
<td></td>
</tr>
<tr>
<td>WRe26 Pt25</td>
<td></td>
</tr>
<tr>
<td>Ni4 Pt50</td>
<td></td>
</tr>
<tr>
<td>XR Pt200WEED</td>
<td></td>
</tr>
<tr>
<td>Cu10G</td>
<td></td>
</tr>
<tr>
<td>Cu50G</td>
<td></td>
</tr>
<tr>
<td>Cu10GD</td>
<td></td>
</tr>
<tr>
<td>Cu50GD</td>
<td></td>
</tr>
<tr>
<td>Pt46G</td>
<td></td>
</tr>
<tr>
<td>Pt100G</td>
<td></td>
</tr>
</tbody>
</table>

• For the setting ranges of p7 (span low limit) and p8 (span high limit), see the PID Control User’s Manual (IM 04L51B01-31EN).

SCtrlBurnOut
Burnout Mode
Sets the burnout action of the AI terminal of a PID module.
Syntax
SCtrlBurnOut,p1,p2,p3,p4
p1 Unit number
p2 Module number
p3 AI terminal number (AI1, AI2)
p4 Burnout action (Off, Up, Down)
2.4 Setting Commands

**Query**  
SCtrlBurnOut[,p1,p2,p3]?

**Example**  
When a burnout is detected on AI1 of module number 2 connected to the main unit, set the terminal to UP.  
SCBurnOut,0,2,AI1,Up

**Description**
- This command is valid when a PID Control Module is installed.
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.

**SCtrlRjc**

**RJC temperature**
Sets the RJC temperature of the AI terminal of a PID control module.

**Syntax**  
SCtrlRjc,p1,p2,p3,p4,p5
- p1 Unit number
- p2 Module number
- p3 AI terminal number (AI1, AI2)
- p4 Mode (Internal, External)
  - Internal
  - External
- p5 Compensation temperature
  - Fahrenheit: -40 to 1760
  - Celsius: -200 to 800
  - Absolute temperature: 2531~3532

**Query**  
SCtrlRjc[,p1,p2,p3]?

**Example**  
Set the reference junction compensation of AI1 of module number 2 connected to the main unit to internal compensation circuit.  
SCtrlRjc,0,2,AI1,Internal

**Description**
- This command is valid when a PID Control Module is installed.
- You cannot use this command to configure settings while recording is in progress.
- You cannot use this command to configure settings while computation is in progress.
- If p4=Internal, you cannot set p5.
- Absolute temperature (K) is valid when the input type is TC and the range type is KpvsAu7Fe.

**SCtrlFilter**

**First-Order Lag Filter**
Sets the first-order lag filter of the AI terminal of the PID control module.

**Syntax**  
SCtrlFilter,p1,p2,p3,p4,p5
- p1 Unit number
- p2 Module number
- p3 AI terminal number (AI1, AI2)
- p4 Enable or disable (On, Off)
- p5 First-order lag constant (0 to 120) Unit: sec

**Query**  
SCtrlFilter[,p1,p2,p3]?

**Example**  
Set the first-order lag of AI1 of module number 2 connected to the main unit to 110 (s).  
SCtrlFilter,0,2,AI1,On,110

**Description**
- This command is valid when a PID Control Module is installed.

**SCtrlCalibAI**

**Calibration Correction**
Sets the calibration correction of the AI terminal of the PID control module.

**Disable Calibration Correction**

**Syntax**  
SCtrlCalibAI,p1,p2,p3,p4
- p1 Unit number
- p2 Module number
- p3 AI terminal number (AI1, AI2)
- p4 Linearizer mode
  - Off: Correction is not performed.

**Use Calibration Correction (Linearizer approximation, linearizer bias)**

**Syntax**  
- p1 Unit number
- p2 Module number
- p3 AI terminal number (AI1, AI2)
- p4 Linearizer mode
  - Appro: Linearizer approximation
  - Bias: Linearizer bias
- p5 Number of segmental points (2 to 12)
- p6 Input value of segmental point 1
- p7 Output value of segmental point 1
- p8 Input value of segmental point 2
- p9 Output value of segmental point 2
- p10 Input value of segmental point 3 (number of segmental points ≥ 3)
- p11 Output value of segmental point 3 (number of segmental points ≥ 3)
- p12 Input value of segmental point 4 (number of segmental points ≥ 4)
- p13 Output value of segmental point 4 (number of segmental points ≥ 4)
- p14 Input value of segmental point 5 (number of segmental points ≥ 5)
### Syntax

Use Calibration Correction (Correction coefficient)

**Syntax**

```
```

- **p1**: Unit number
- **p2**: Module number
- **p3**: AI terminal number (AI1, AI2)
- **p4**: Linearizer mode
- **p5**: Correct Correction Factor
- **p6**: Number of correction points (2 to 12)
- **p7**: Uncorrected value 1
- **p8**: Instrument correction coefficient 1
- **p9**: Sensor correction coefficient 1
- **p10**: Uncorrected value 2
- **p11**: Instrument correction coefficient 2
- **p12**: Sensor correction coefficient 2
- **p13**: Uncorrected value 3 (number of segmental points ≥ 3)
- **p14**: Instrument correction factor 3 (number of segmental points ≥ 3)
- **p15**: Sensor correction factor 3 (number of segmental points ≥ 3)
- **p16**: Uncorrected value 4 (number of segmental points ≥ 4)
- **p17**: Instrument correction factor 4 (number of segmental points ≥ 4)
- **p18**: Sensor correction factor 4 (number of segmental points ≥ 4)
- **p19**: Uncorrected value 5 (number of segmental points ≥ 5)
- **p20**: Instrument correction factor 5 (number of segmental points ≥ 5)
- **p21**: Sensor correction factor 5 (number of segmental points ≥ 5)
- **p22**: Uncorrected value 6 (number of segmental points ≥ 6)
- **p23**: Sensor correction factor 6 (number of segmental points ≥ 6)
- **p24**: Uncorrected value 7 (number of segmental points ≥ 7)
- **p25**: Instrument correction factor 7 (number of segmental points ≥ 7)
- **p26**: Sensor correction factor 7 (number of segmental points ≥ 7)
- **p27**: Uncorrected value 8 (number of segmental points ≥ 8)
- **p28**: Instrument correction factor 8 (number of segmental points ≥ 8)
- **p29**: Sensor correction factor 8 (number of segmental points ≥ 8)
- **p30**: Uncorrected value 9 (number of segmental points ≥ 9)
- **p31**: Instrument correction factor 9 (number of segmental points ≥ 9)
- **p32**: Sensor correction factor 9 (number of segmental points ≥ 9)
- **p33**: Uncorrected value 10 (number of segmental points ≥ 10)
- **p34**: Instrument correction factor 10 (number of segmental points ≥ 10)
- **p35**: Sensor correction factor 10 (number of segmental points ≥ 10)
- **p36**: Uncorrected value 11 (number of segmental points ≥ 11)
- **p37**: Instrument correction factor 11 (number of segmental points ≥ 11)
- **p38**: Sensor correction factor 11 (number of segmental points ≥ 11)
- **p39**: Uncorrected value 12 (number of segmental points ≥ 12)
- **p40**: Instrument correction factor 12 (number of segmental points ≥ 12)
- **p41**: Sensor correction factor 12 (number of segmental points ≥ 12)

### Description

- **Set three correction points on channel 0001** (measurement range: 0 to 1.0000 V). Set the correction points as follows: when the input value is 0 V, the output value is 0.0010 V; when the input value is 0.5000 V, the output value is 0.5020 V; when the input value is 1.0000 V, the output value is 0.9970 V.

```
SCtrlCalibAI[, p1, p2, p3]?
```

- **Output value of segmental point 12**
- **Input value of segmental point 12**
- **Output value of segmental point 11**
- **Input value of segmental point 11**
- **Output value of segmental point 10**
- **Input value of segmental point 10**
- **Output value of segmental point 9**
- **Input value of segmental point 9**
- **Output value of segmental point 8**
- **Input value of segmental point 8**
- **Output value of segmental point 7**
- **Input value of segmental point 7**
- **Output value of segmental point 6**
- **Input value of segmental point 6**
- **Uncorrected value 5**
- **Sensor correction factor 4**
- **Instrument correction factor 4**
- **Uncorrected value 4**
- **Sensor correction factor 3**
- **Instrument correction factor 3**
- **Uncorrected value 3**
- **Sensor correction factor 2**
- **Instrument correction factor 2**
- **Uncorrected value 2**
- **Sensor correction factor 1**
- **Instrument correction factor 1**
- **Uncorrected value 1**
- **Sensor correction factor 12**
- **Instrument correction factor 12**
- **Sensor correction factor 10**
- **Instrument correction factor 10**
- **Sensor correction factor 8**
- **Instrument correction factor 8**
- **Sensor correction factor 6**
- **Instrument correction factor 6**
- **Sensor correction factor 4**
- **Instrument correction factor 4**
- **Sensor correction factor 2**
- **Instrument correction factor 2**
- **Sensor correction factor 1**
- **Instrument correction factor 1**

- **Query Example**

```
Set three correction points on channel 0001 (measurement range: 0 to 1.0000 V). Set the correction points as follows: when the input value is 0 V, the output value is 0.0010 V; when the input value is 0.5000 V, the output value is 0.5020 V; when the input value is 1.0000 V, the output value is 0.9970 V.
```

```
SCtrlCalibAI, 0001, Appro, 3, 0, 10, 5000, 5020, 10000, 9970
```
2.4 Setting Commands

- If the AI channel input type (p4 of the SCtrlRangeAI command) is set to Skip or DI, you cannot specify anything other than p4=Off.

SCtrlRangeAO

Transmission Output
Sets the transmission output range of the AO terminal of a PID control module.

**Syntax**

\[
\text{SCtrlRangeAO}, p1, p2, p3, p4, p5, p6, p7
\]

- \( p1 \) Unit number
- \( p2 \) Module number
- \( p3 \) AO terminal number (AO1, AO2)
- \( p4 \) Transmission output (On, Off)
  - Off
  - On
- \( p5 \) Terminal number
  - OUT1
  - PV1
  - SP1
  - OUT2
  - PV2
  - SP2
- \( p6 \) Scaling low limit (–30000 to 30000)
- \( p7 \) Scaling high limit (–30000 to 30000)

**Query**

\[
\text{SCtrlRangeAO}\{,p1,p2,p3\}?
\]

**Example**

Set the output of AO1 of module number 2 connected to the main unit to PV1.

\[
\text{SCtrlRangeAO}, 0, 2, \text{AO1}, \text{On}, \text{PV1}, -30000, 30000
\]

**Description**

- This command is valid when a PID Control Module is installed.
- If p4=Off, p5 will be set in the following combinations.

<table>
<thead>
<tr>
<th>Control mode</th>
<th>( p3 = \text{AO1} )</th>
<th>( p3 = \text{AO2} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single loop or Cascade</td>
<td>( p5 = \text{OUT1} )</td>
<td>( p5 = \text{OUT2} )</td>
</tr>
<tr>
<td>PV switching</td>
<td>( p5 = \text{OUT1} )</td>
<td>( p5 = \text{OUT1} )</td>
</tr>
</tbody>
</table>

- \( p5 \) cannot be set to OUT2, PV2 or SP2 when the control mode is “PV switching”.

SCtrlSplitAO

Split Computation
Sets the split computation of the AO terminal of the PID control module.

**Syntax**

\[
\text{SCtrlSplitAO}, p1, p2, p3, p4, p5, p6, p7
\]

- \( p1 \) Unit number
- \( p2 \) Module number
- \( p3 \) AO terminal number (AO1, AO2)
- \( p4 \) Linearizer mode (Off)
  - Off
  - On
- \( p5 \) Value at the segmental point for output 0% (-1000 to 2000)
- \( p6 \) Value at the segmental point for output 100% (-1000 to 2000)

**Query**

\[
\text{SCtrlSplitAO}\{,p1,p2,p3\}?
\]

**Example**

Set the AO1 of module number 2 connected to the main unit to PV1.

\[
\text{SCtrlSplitAO}, 0, 2, \text{AO1}, \text{Off}, \text{PV1}, -30000, 30000
\]

**Description**

- This command is valid when a PID Control Module is installed.
- If p4=Off, you cannot set p5 and subsequent parameters.

SCtrlOutput

Control Output
Sets the output type of the AO terminal of a PID control module.

**Syntax**

\[
\text{SCtrlOutput}, p1, p2, p3, p4, p5, p6
\]

- \( p1 \) Unit number
- \( p2 \) Module number
- \( p3 \) AO terminal number (AO1, AO2)
- \( p4 \) Control output type
  - Current-output
  - Voltage-pulse
  - VDC-power
  - 15 VDC power supply
- \( p5 \) Cycle time (5 to 10000) (0.5 s to 1000.0 s)
- \( p6 \) Analog output type
  - 4-20mA
  - 0-20mA
  - 20-4mA
  - 20-0mA

**Query**

\[
\text{SCtrlOutput}\{,p1\}?
\]

**Example**

Set the AO1 output of module number 2 connected to the main unit to current, 800 s cycle time, and 4-20mA analog output type.

\[
\text{SCtrlOutput}, 0, 2, \text{AO1}, \text{Current-output}, 800, 4-20mA
\]

**Description**

- This command is valid when a PID Control Module is installed.
**SCtrlRangePV**

**Control Input Range**
Sets the control input range.

**Syntax**
```
SCtrlRangePV,p1,p2,p3,p4,p5
```
- `p1`: Loop number
- `p2`: PV range low limit (-30000 to 30000)
- `p3`: PV range high limit (-30000 to 30000)
- `p4`: PV range decimal place (0 to 4)
- `p5`: Unit (up to 6 characters, UTF-8)

**Query**
```
SCtrlRangePV,(p1)?
```

**Example**
For loop number L022, set the PV range to -30000 to 30000, decimal place to 2, and unit to "UniA."
```
SCtrlRangePV,L022,-30000,30000,2,'UniA'
```

**Description**
- This command is valid when a PID Control Module is installed.
- Set PV range high and low limits (p2, p3) so that p2<p3 and p3–p2≤30000 are satisfied.

---

**SCtrlPVSwitch**

**Input Switching PV for PV Switching**
Sets the input switching PV value for PV switching

**Syntax**
```
SCtrlPVSwitch,p1,p2,p3
```
- `p1`: Loop number
- `p2`: Input switching PV low limit (PV range low limit to PV range high limit)
- `p3`: Input switching PV high limit (PV range low limit to PV range high limit)

**Query**
```
SCtrlPVSwitch,(p1)?
```

**Example**
For loop number L022, set the input switching PV value to -30000 to 30000.
```
SCtrlPVSwitch,L022,-30000,30000
```

**Description**
- This command is valid when a PID Control Module is installed.
- If the PV switching condition is set to low temperature range or high temperature range, set input switching PV low limit to a value less than input switching PV high limit. (See the SCtrlMode command.)

---

**SCtrlCalc**

**EXPV/RSP Function Setting**
Sets the reference source of EXPV and RSP.

**Syntax**
```
SCtrlCalc,p1,p2,p3,p4
```
- `p1`: Loop number
- `p2`: EXPV/RSP number
- `p3`: Channel type (IO, Math, Com, Off)
- `p4`: Channel number or terminal number

**Query**
```
SCtrlOutput,(p1)?
```

**Example**
For loop number L022, set EXPV of PV1 to channel A001.
```
SCtrlCalc,L022,EXPV,Math,001
```

**Description**
- This command is valid when a PID Control Module is installed.
- p2 can be set to EXPV1 or EXPV2 when the EXPV function is enabled.
- p2 can be set to RSP when the RSP function is enabled.
- p2 can be set to EXPV2 when the EXPV function is enabled and PV switching is used.
- p3 can be set to Off when the EXPV function is enabled, and p2=EXPV or EXPV2 in PV switching.

---

**SCtrlFilterSP**

**Remote SP Filter**
Sets the remote SP filter.

**Syntax**
```
SCtrlFilterSP,p1,p2,p3
```
- `p1`: Loop number
- `p2`: Filter on/off
  - Off
  - On
- `p3`: Filter value (1 to 120)

**Query**
```
SCtrlFilterSP,(p1)?
```

**Example**
For loop number L022, set the remote SP filter to 120 s.
```
SCtrlFilterSP,L022,On,120
```

**Description**
- This command is valid when a PID Control Module is installed.
2.4 Setting Commands

**SCtrlRatioSP**
Remote SP Ratio
Sets the remote SP ratio.

**Syntax**

```
SCtrlRatioSP,p1,p2,p3
```

- **p1** Loop number
- **p2** Ratio setting on/off
  - Off
  - On
- **p3** Ratio value (1 to 9999)
  - 0.001 to 9.999
  - Fixed to three decimal places

**Query**

```
SCtrlFilterSP[,p1]?
```

**Example**

For loop number L022, set the remote SP filter to 120 s.

```
SCtrlFilterSP,L022,On,120
```

**Description**

- This command is valid when a PID Control Module is installed.

**SCtrlOutLimit**
Output Limiter Function in Manual Mode
Sets the output limiter function in manual mode

**Syntax**

```
SCtrlOutLimit,p1,p2
```

- **p1** Loop number
- **p2** Output limiter switch
  - Off Disable the output limiter in manual mode.
  - On Enable the output limiter in manual mode.

**Query**

```
SCtrlOutLimit[,p1]?
```

**Example**

For loop number L022, enable the output limiter function.

```
SCtrlOutLimit,L022,On
```

**Description**

- This command is valid when a PID Control Module is installed.

**SCtrlAlarm**
Control Alarm
Sets the control alarm type and action

**Syntax**

```
SCtrlAlarm,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,p13
```

- **p1** Loop number
- **p2** Alarm number (1 to 4)
- **p3** On/Off (Off, On)
- **p4** Type of alarm
  - PV-High PV high limit
  - PV-Low PV low limit
  - SP-High SP high limit
  - SP-Low SP low limit
  - Dev-High Deviation high limit
  - Dev-Low Deviation low limit
  - Dev-HL Deviation H/L limits
  - Dev-HL-In Deviation within H/L limits
  - OUT-High Control output high limit
  - OUT-Low Control output low limit
  - PV-Rate PV velocity
- **p5** Standby action
  - On
  - Off
- **p6** Hysteresis (0 to 30000)
- **p7** On delay timer (min) (0 to 99)
- **p8** On delay timer (sec) (0 to 59)
- **p9** Off delay timer (min) (0 to 99)
- **p10** Off delay timer (sec) (0 to 59)
- **p11** Relay action
  - off Nonhold
  - Relay1 Hold
  - Relay2 Relay action on hold & ACK Reset

**Example**
For loop number L022, set the output value for when input errors occur to control output 100%.

```
SCtrlErrPreOut,L022,OUT100%
```

**Description**

- This command is valid when a PID Control Module is installed.
2.4 Setting Commands

### SCtrlAlarm

#### SPNo Group Setting 2 (Target Setpoint Alarm)

Sets the target setpoint

**Syntax**

```
SCtrlAlarm[,p1,p2]
```

- **p1** Loop number
- **p2** Target setpoint number (1 to 8)
- **p3** Target setpoint (target setpoint low limit to target setpoint high limit)

**Example**

For loop number L022, set the target setpoint of target setpoint number 8 to -2500.5.

```
SCtrlAlarm,L022,8,-2500.5
```

**Description**

- This command is valid when a PID Control Module is installed.

#### SCtrlAlarmVal

#### SPNo Group Setting 1 (Control Alarm)

Sets the control alarm value

**Syntax**

```
SCtrlAlarmVal,p1,p2,p3,p4,p5,p6
```

- **p1** Loop number
- **p2** Target setpoint number (1 to 8)
- **p3** Target setpoint (target setpoint low limit to target setpoint high limit)
- **p4** Alarm value 1 (-30000 to 30000)
- **p5** Alarm value 2 (-30000 to 30000)
- **p6** Alarm value 3 (-30000 to 30000)

**Query**

```
SCtrlAlarmVal[,p1,p2]
```

**Example**

For loop number L022, set the alarm value of alarm number 8 of target setpoint number 5 to 2500.5.

```
SCtrlAlarmVal,L022,8,5,2500.5
```

**Description**

- This command is valid when a PID Control Module is installed.
- Alarm values p3 to p6 can be set regardless of the number of alarms.

### SCtrlSP

#### SPNo Group Setting 3 (Target Setpoint Ramp-Rate)

Sets the target setpoint ramp-rate

**Syntax**

```
SCtrlSPGradient, p1, p2,p3,p4,p5,p6
```

- **p1** Loop number
- **p2** Target setpoint number (fixed to “-“)
- **p3** Target setpoint ramp-down rate
  - **p4** Target setpoint ramp-down rate value
    - Off
    - On
  - **p5** Target setpoint ramp-up rate
    - Off
    - On
- **p6** Target setpoint ramp-up rate value
  - **p6** Target setpoint ramp-up rate value
    - Off
    - On

**Example**

For loop number L022, set the ramp-down rate to 350.4 and the ramp-up rate to 580.9.

```
SCtrlSPGradient,L022,350.4,580.9
```

**Description**

- This command is valid when a PID Control Module is installed.
### 2.4 Setting Commands

#### SCtrlPIDNo

**SPNo Group Setting 4 (PID Group Number)**

Sets the target setpoint PID group number

**Syntax**

```plaintext
SCtrlPIDNo, p1, p2, p3
```

- **p1** Loop number
- **p2** Target setpoint number (1 to 8)
- **p3** PID group number (1 to 8)

**Query**

`SCtrlPIDNo{[,p1,p2]}`

**Example**

For loop number L022, set the PID group number of target setpoint number 8 to 3.

`SCtrlPIDNo,L022,8,3`

**Description**

- This command is valid when a PID Control Module is installed.
- The maximum value of p3 is the number of SP groups set using p3 of the **SCtrlSPPID** command.

#### SCtrlRefPoint

**Zone PID Setting 1 (Reference Point)**

Sets the zone PID reference point

**Syntax**

```plaintext
SCtrlRefPoint, p1, p2, p3
```

- **p1** Loop number
- **p2** Reference point number (1 to number of PID groups – 1)
- **p3** Reference point (PV range low limit to PV range high limit)

**Query**

`SCtrlRefPoint{[,p1,p2]}`

**Example**

For loop number L022, set the reference point of reference point number 7 to -450.5.

`SCtrlRefPoint,L022,7,-4505`

**Description**

- This command is valid when a PID Control Module is installed.

#### SCtrlRHys

**Zone PID Setting 2 (Switching Hysteresis)**

Sets the zone PID switching hysteresis

**Syntax**

```plaintext
SCtrlRHys, p1, p2
```

- **p1** Loop number
- **p2** Switching hysteresis
  
  Setting range: 0% to 100% of PV range span

**Example**

For loop number L022, set the switching hysteresis to 30.8.

`SCtrlRHys,L022,308`

**Description**

- This command is valid when a PID Control Module is installed.

#### SCtrlPIDPb

**PID Parameter Setting 1 (Proportional Band)**

Sets the proportional band

**Syntax**

```plaintext
SCtrlPIDPb, p1, p2, p3
```

- **p1** Loop number
- **p2** PID group number (1 to 8)
- **p3** Proportional band P (1 to 9999)

  Setting range: 0.1 to 999.9%

**Query**

`SCtrlPIDPb{[,p1,p2]}`

**Example**

For loop number L022, set the proportional band P of PID group number 8 to 80.0%.

`SCtrlPIDPb,L022,8,800`

**Description**

- This command is valid when a PID Control Module is installed.

#### SCtrlPIDTI

**PID Parameter Setting 2 (Integration Time)**

Sets the integration time

**Syntax**

```plaintext
SCtrlPIDTI, p1, p2, p3
```

- **p1** Loop number
- **p2** PID group number (1 to 8)
- **p3** Integration time I (0 to 6000)

  Setting range: 0 (OFF) to 6000 (s)

**Query**

`SCtrlPIDTI{[,p1,p2]}`

**Example**

For loop number L022, set the integration time of PID group number 8 to 240 s.

`SCtrlPIDTI,L022,8,240`

**Description**

- This command is valid when a PID Control Module is installed.
### SCtrlPIDTD

**PID Parameter Setting 3 (Derivative Time)**
Sets the derivative time

<table>
<thead>
<tr>
<th>Syntax</th>
<th>SCtrlPIDTD,p1,p2,p3</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>Loop number</td>
</tr>
<tr>
<td>p2</td>
<td>PID group number (1 to 8)</td>
</tr>
<tr>
<td>p3</td>
<td>Derivative time D (0 to 6000)</td>
</tr>
<tr>
<td></td>
<td>Setting range: 0 (OFF) to 6000 (s)</td>
</tr>
</tbody>
</table>

**Query**
SCtrlPIDTD [,p1,p2]?

**Example**
For loop number L022, set the derivative time of PID group number 8 to 60 s.

SCtrlPIDTD,L022,8,60

**Description**
- This command is valid when a PID Control Module is installed.
- Set the control output high limit (p3) less than the control output low limit (p4).

---

### SCtrlPIDPara

**PID Parameter Setting 4 (Other Controls)**
Sets control parameters

<table>
<thead>
<tr>
<th>Syntax</th>
<th>SCtrlPIDPara,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>Loop number</td>
</tr>
<tr>
<td>p2</td>
<td>PID group number (1 to 8)</td>
</tr>
<tr>
<td>p3</td>
<td>Control output low limit (-50 to 1050)</td>
</tr>
<tr>
<td></td>
<td>Setting range: -5.0 to 105.0%</td>
</tr>
<tr>
<td>p4</td>
<td>Control output high limit (-50 to 1050)</td>
</tr>
<tr>
<td></td>
<td>Setting range: -5.0 to 105.0%</td>
</tr>
<tr>
<td>p5</td>
<td>Tight shut function</td>
</tr>
<tr>
<td></td>
<td>Off</td>
</tr>
<tr>
<td></td>
<td>On</td>
</tr>
<tr>
<td>p6</td>
<td>Manual reset (-50 to 1050)</td>
</tr>
<tr>
<td></td>
<td>Setting range: -5.0 to 105.0%</td>
</tr>
<tr>
<td>p7</td>
<td>Upper-side hysteresis</td>
</tr>
<tr>
<td></td>
<td>0% to 100% of PV range span</td>
</tr>
<tr>
<td>Example</td>
<td>0.0 to 100.0° (0 to 900) when the PV range is 10.0 to 90.0°</td>
</tr>
<tr>
<td>p7</td>
<td>Lower-side hysteresis</td>
</tr>
<tr>
<td></td>
<td>0% to 100% of PV range span</td>
</tr>
<tr>
<td>Example</td>
<td>0.0 to 100.0° (0 to 900) when the PV range is 10.0 to 90.0°</td>
</tr>
<tr>
<td>p9</td>
<td>Control direction</td>
</tr>
<tr>
<td></td>
<td>Reverse</td>
</tr>
<tr>
<td></td>
<td>Direct</td>
</tr>
<tr>
<td>p10</td>
<td>Preset output (-50 to 1050)</td>
</tr>
<tr>
<td></td>
<td>Setting range: -5.0 to 105.0%</td>
</tr>
</tbody>
</table>

**Query**
SCtrlPIDPara [,p1,p2]?

**Example**
For PID group number 8 of loop number L022, set the output limit to 10% to 80%, tight shut function to On, manual reset to 40%, hysteresis to -30.0 to 50.0, control direction to reverse, and preset output to 10%.

SCtrlPIDPara,L022,8,100,800,On,400,-300,500,Reverse,100

**Description**
- This command is valid when a PID Control Module is installed.
- Set the control output high limit (p3) less than the control output low limit (p4).

---

### SCtrlRefPb

**Reference PID Setting 1 (Proportional Band)**
Sets the proportional band

<table>
<thead>
<tr>
<th>Syntax</th>
<th>SCtrlRefPb,p1,p2</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>Loop number</td>
</tr>
<tr>
<td>p2</td>
<td>Proportional band P (1 to 9999)</td>
</tr>
<tr>
<td></td>
<td>Setting range: 0.1 to 999.9%</td>
</tr>
</tbody>
</table>

**Query**
SCtrlRefPb [,p1,p2]?

**Example**
For loop number L022, set the proportional band P of the reference PID to 80.0%.

SCtrlRefPb,L022,800

**Description**
- This command is valid when a PID Control Module is installed.
- Set the control output high limit (p3) less than the control output low limit (p4).

---

### SCtrlRefTI

**Reference PID Setting 2 (Integration Time)**
Sets the integration time

<table>
<thead>
<tr>
<th>Syntax</th>
<th>SCtrlRefTI,p1,p2</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>Loop number</td>
</tr>
<tr>
<td>p2</td>
<td>Integration time I (0 to 6000)</td>
</tr>
<tr>
<td></td>
<td>Setting range: 0 (OFF) to 6000 (s)</td>
</tr>
</tbody>
</table>

**Query**
SCtrlRefTI [,p1,p2]?

**Example**
For loop number L022, set the integration time of the reference PID to 240 s.

SCtrlRefTI,L022,240

**Description**
- This command is valid when a PID Control Module is installed.

---

### SCtrlRefTD

**Reference PID Setting 2 (Derivative Time)**
Sets the derivative time

<table>
<thead>
<tr>
<th>Syntax</th>
<th>SCtrlRefTD,p1,p2</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>Loop number</td>
</tr>
<tr>
<td>p2</td>
<td>Derivative time D (0 to 6000)</td>
</tr>
<tr>
<td></td>
<td>Setting range: 0 (OFF) to 6000 (s)</td>
</tr>
</tbody>
</table>

**Query**
SCtrlRefTD [,p1,p2]?

**Example**
For loop number L022, set the derivative time of the reference PID to 60 s.

SCtrlRefTD,L022,60

**Description**
- This command is valid when a PID Control Module is installed.
### SCtrlRefPara

**Reference PID Setting 4 (Other Controls)**

Sets control parameters

**Syntax**
```
SCtrlRefPara, p1, p2, p3, p4, p5, p6, p7, p8, p9
```

- **p1**: Loop number
- **p2**: Control output low limit (-50 to 1050)
  - Setting range: -5.0 to 105.0%
- **p3**: Control output high limit (-50 to 1050)
  - Setting range: -5.0 to 105.0%
- **p4**: Tight shut function
  - Off
  - On
- **p5**: Manual reset (-50 to 1050)
  - Setting range: -5.0 to 105.0%
- **p6**: Upper-side hysteresis
  - 0% to 100% of PV range span
- **Example**: 0.0 to 100.0° (0 to 900) when the PV range is 10.0 to 90.0°
- **p7**: Lower-side hysteresis
  - 0% to 100% of PV range span
- **Example**: 0.0 to 100.0° (0 to 900) when the PV range is 10.0 to 90.0°
- **p8**: Control direction
  - Reverse
  - Direct
- **p9**: Preset output (-50 to 1050)
  - Setting range: -5.0 to 105.0%

**Query**
```
SCtrlRefPara[,]? p1
```

**Example**
For the reference PID of loop number L022, set the output limit to 10% to 80%, tight shut function to On, manual reset to 40%, hysteresis to -30.0 to 50.0, control direction to reverse, and preset output to 10%.
```
SCtrlRefPara, L022, 100, 800, On, 400, -300, 500, Reverse, 100
```

**Description**
- This command is valid when a PID Control Module is installed.
- Set the control output high limit (p2) less than the control output low limit (p3).

### SCtrlDetail

**Control Detail Setting 1 (Tracking, Setpoint Limit, Ramp-Rate Time Unit)**

Sets the tracking, setpoint limit, and ramp-rate time unit

**Syntax**
```
SCtrlDetail, p1, p2, p3, p4, p5, p6
```

- **p1**: Loop number
- **p2**: Target setpoint tracking (Off, On)
  - Off
  - On
- **p3**: PV tracking (Off, On)
  - Off
  - On
- **p4**: Target setpoint low limit (PV range low limit to PV range high limit)
- **p5**: Target setpoint high limit (PV range low limit to PV range high limit)
- **p6**: Ramp-rate time unit (Hour, Min, Sec)
  - Hour
  - Hours
  - Min
  - Minutes
  - Sec
  - Seconds

**Query**
```
SCtrlDetail[,p1]?
```

**Example**
For loop number L022, set the target setpoint tracking to On, PV tracking to On, target setpoint limits to -300.0 to 300.0, and ramp-rate time unit to minutes.
```
SCtrlDetail, L022, On, On, -3000, 3000, Min
```

**Description**
- This command is valid when a PID Control Module is installed.
- Set the target setpoint high limit (p4) less than the target setpoint low limit (p5).

### SCtrlOutRatio

**Control Detail Setting 2 (Output Velocity Limiter)**

Sets the output velocity limiter

**Syntax**
```
SCtrlOutRatio, p1, p2, p3
```

- **p1**: Loop number
- **p2**: Output velocity limiter (Off, On)
  - Off
  - On
- **p3**: Output velocity limiter value (1 to 10000)
  - Setting range: 0.1 to 100.0%/s

**Query**
```
SCtrlOutRatio[,p1]?
```

**Example**
For loop number L022, set the output velocity limiter to 10.5 (%/s).
```
SCtrlOutRatio, L022, On, 105
```

**Description**
- This command is valid when a PID Control Module is installed.
### SCtrlAtDetail

**Control Detail Setting 3 (Auto-Tuning Details)**
Sets the auto-tuning details

**Syntax**
```
SCtrlAtDetail,p1,p2,p3,p4,p5
```
- **p1** Loop number
- **p2** Type
  - NORMAL : Normal
  - STABILITY : Stability
- **p3** Output limiter low limit (-50 to 1050)
  - Setting range: -5.0 to 105.0%
- **p4** Output limiter high limit (-50 to 1050)
  - Setting range: -5.0 to 105.0%
- **p5** Bias (-100% to 100% of the PV range span)
  - Example: -90.0° to 90.0° for 10.0° to 100.0°

**Query**
```
SCtrlAtDetail[,p1]?
```

**Example**
For loop number L022, set the auto-tuning type to Stability, limiter to -5% to 90%, and bias to 150.0.
```
SCtrlAtDetail,L022,STABILITY,-50,900,1500
```

**Description**
- This command is valid when a PID Control Module is installed.
- Set the output limiter low limit (p3) less than the output limiter high limit (p4).

### SCtrlAntiReset

**Control Detail Setting 4 (Over-Integration Suppressing Function)**
Sets the over-integration suppressing function (anti-reset windup)

**Syntax**
```
SCtrlAntiReset,p1,p2,p3
```
- **p1** Loop number
- **p2** Type
  - Auto : Auto
- **p3** Deviation band (500 to 2000)
  - Setting range: 50.0 to 200.0%

**Query**
```
SCtrlAntiReset[,p1]?
```

**Example**
For loop number L022, set the deviation band of the over-integration suppressing function to 70.0%.
```
SCtrlAntiReset,L022,Manual,700
```

**Description**
- This command is valid when a PID Control Module is installed.
- Set the output limiter low limit (p3) less than the output limiter high limit (p4).
  - If p2=Auto, p3 is fixed to 0. If p2=Manual, set p3 and subsequent parameters.

### SCtrlOvershoot

**Control Detail Setting 5 (Control Output Suppressing Function)**
Sets the control output suppressing function (overshoot-suppressing function)

**Syntax**
```
SCtrlOvershoot,p1,p2
```
- **p1** Loop number
- **p2** Control output suppressing function
  - Off : Not Use
  - Normal : Normal mode

**Query**
```
SCtrlOvershoot[,p1]?
```

**Example**
For loop number L022, set overshoot to normal mode.
```
SCtrlOvershoot,L022,Normal
```

**Description**
- This command is valid when a PID Control Module is installed.
- Set the output limiter low limit (p3) less than the output limiter high limit (p4).

### SCtrlGroupSW

**Control Group Use/Not**
Sets whether to use control groups

**Syntax**
```
SCtrlGroupSW,p1,p2
```
- **p1** Display group number
- **p2** Use/Not
  - Off : Not Use
  - On : Use

**Query**
```
SCtrlGroupSW[,p1]?
```

**Example**
Set the group of display group number 8 to not use.
```
SCtrlGroupSW,8,Off
```

**Description**
- This command is valid when a PID Control Module is installed.
- The selectable range of display group numbers (p1) varies depending on the model.

### SCtrlGroupName

**Control Group Name**
Sets the control group name

**Syntax**
```
SCtrlGroupName,p1,p2
```
- **p1** Display group number
  - GX10/GX20-1/GM10-1: 1 to 5
  - GX20-2/GM10-2: 1 to 10
- **p2** Group name (up to 16 characters, UTF-8)

**Query**
```
SCtrlGroupName[,p1]?
```

**Example**
Set the group name of display group number 8 to “Group8.”
```
SCtrlGroupName,8,'Group8'
```

**Description**
- This command is valid when a PID Control Module is installed.
- The selectable range of display group numbers (p1) varies depending on the model.
### SCtrlGroupSplit

**Control Group Divisions [GX/GP]**

Sets the number of control group divisions

**Syntax**

```
SCtrlGroupSplit, p1, p2
```

- **p1**: Display group number (1 to 10)
  - GX10/GX20-1/GM10-1: 1 to 5
  - GX20-2/GM10-2: 1 to 10
- **p2**: Number of divisions (2, 4, 6, 8)
  - The number of divisions cannot be set to 8 on the GX10.

**Query**

```
SCtrlGroupSplit[, p1]
```

**Example**

Set the number of divisions of display group number 8 to 4.

```
SCtrlGroupSplit, 8, 4
```

**Description**

- This command is valid when a PID Control Module is installed.
- The selectable range of display group numbers (p1) varies depending on the model.

### SCtrlGroup

**Control Group Assignment**

Sets loops to assign to control groups

**Syntax**

```
SCtrlGroup, p1, p2, p3, p4
```

- **p1**: Display group number
- **p2**: Setting Number
  - GX20/GM10: 1 to 8
  - GX10: 1 to 6
- **p3**: Setting On, Off
  - Off: Not set
  - On: Set
- **p4**: Type
  - INT
  - Loop
- **p5**: Loop number

**Query**

```
SCtrlGroup[, p1, p2]
```

**Example**

Assign L001, L002, L011, L012, L021, and L022 to the group with display group number 8. SCtrlGroup, 8, 1, On, INT, 001; SCtrlGroup, 8, 2, On, INT, 002; SCtrlGroup, 8, 3, On, INT, 011; SCtrlGroup, 8, 4, On, INT, 012; SCtrlGroup, 8, 5, On, INT, 021; SCtrlGroup, 8, 6, On, INT, 022;

**Description**

- This command is valid when a PID Control Module is installed.
- On the GX/GP, p3 cannot be set to On exceeding the number of divisions. (See the SCtrlGroupSplit command.)

### SCtrlTag

**Loop Tag, Tag Comment**

Sets the loop tag and tag comment

**Syntax**

```
SCtrlTag, p1, p2, p3, p4
```

- **p1**: Loop number
- **p2**: Tag (up to 32 characters, UTF-8)
- **p3**: Tag No. (up to 16 alphanumeric characters, UTF-8)
- **p4**: Type
  - INT
  - Loop

**Query**

```
SCtrlTag[, p1]
```

**Example**

For loop number L022, set the tag to “Tag L022” and tag No. to “Ctrl-L022.”

```
SCtrlTag, L022, ‘Tag L022’, ‘Ctrl-L022
```

**Description**

- This command is valid when a PID Control Module is installed.

### SCtrlDispDV

**Deviation Display Band (Control Group Display)**

Sets the deviation display band

**Syntax**

```
SCtrlDispDV, p1, p2
```

- **p1**: Loop number
- **p2**: Deviation display band (0% to 100% of PV range span)

**Example**

For loop number L022, set the deviation display band to 30.0.

```
SCtrlDispDV, L022, 300
```

**Description**

- This command is valid when a PID Control Module is installed.

### SCtrlBackColor [GX/GP]

**Background color (Control Group Display)**

Sets the background color of the control group display.

**Syntax**

```
SCtrlBackColor, p1
```

- **p1**: Background color
  - White
  - Black

**Example**

Set the background color of the control group display (Control group, Tuning, and Program) to black.

```
SCtrlBackColor, Black
```

**Description**

- This command is valid when a PID Control Module is installed.
### SCtrlOutOperate

**OUT Value Manual Output Operation Type (Control Group Display)**

Sets the OUT value manual output operation type

**Syntax**

```
SCtrlOutOperate,p1
```

- `p1` Confirmation method
  - `DIRECT` Confirm with direct operation
  - `SETENTER` Confirm with the ENTER key

**Query**

```
SCtrlOutOperate[,p1]?
```

**Example**

Confirm the OUT value with the ENTER key.

```
SCtrlOutOperate,SETENTER
```

**Description**

- This command is valid when a PID Control Module is installed.

### STagIO, SColorIO, SZoneIO, SScaleIO, SBarIO, SPartialIO, SValueIO

**Control (PID) Channel Display Settings**

The channel display setting parameters of the control PID module are shared with other channels. For details, see each of the following commands.

- **Tag and tag number**
  - `STagIO`
- **Color**
  - `SColorIO`
- **Zone low limit, zone high limit**
  - `SZoneIO`
- **Scale display position, number of scale divisions**
  - `SScaleIO`
- **Bar display position, number of bar divisions**
  - `SBarIO`
- **Partial**
  - `SPartialIO`
- **Upper and lower limit string**
  - `SValueIO`

In addition, the following table shows the association between the PID control module channel types and commands.

<table>
<thead>
<tr>
<th>Command name</th>
<th>PV</th>
<th>SP</th>
<th>OUT</th>
<th>AI</th>
<th>AO</th>
<th>DI</th>
<th>DO</th>
</tr>
</thead>
<tbody>
<tr>
<td>STagIO</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>SColorIO</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>SZoneIO</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>SScaleIO</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>SBarIO</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>SPartialIO</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>SValueIO</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
</tbody>
</table>

\(Y\): available, \(N\): not available

### SPrgColor

**Program Control Loop Color**

Sets the loop color

**Syntax**

```
SPrgColor,p1,p2,p3,p4
```

- `p1` Loop number
- `p2` R value of RGB colors (0 to 255)
- `p3` G value of RGB colors (0 to 255)
- `p4` B value of RGB colors (0 to 255)

**Query**

```
SPrgColor [,p1] ?
```

**Example**

Confirm the OUT value with the ENTER key.

```
SCtrlOutOperate,SETENTER
```

**Description**

- This command can be used when a PID control module is installed and the program control function (/PG option) is enabled.
  - For details on RGB values, see "Description" of the `SColorIO` command.

### SPrgDispDetail

**Auto Message Printout, Automatic Switch To Program Operation Display, Other Display Settings**

Sets the detail settings for auto message printout, display switch, and start of program operation to On or Off.

**Syntax**

```
SPrgDispDetail,p1,p2,p3
```

- `p1` Program Run/Reset message (Off, On)
  - Off: Messages are not displayed.
  - On: Messages are displayed.
- `p2` Automatic switch to program operation display (Off, On)
  - Off: Not switched to the program operation display
  - On: Switched to the program operation display
- `p3` Detail settings at start of program operation (Off, On)
  - Off: Detail settings are not used (default value).
  - On: Detail settings are used.

**Query**

```
SPrgDispDetail [,p1] ?
```

**Example**

Set Program Run/Reset message to On, automatic switch to program operation display to Off, and Program RUN detail settings to On.

```
SPrgDispDetail,On,Off,On
```

**Description**

- This command can be used when a PID control module is installed and the program control function (/PG option) is enabled.
  - for details on RGB values, see "Description" of the `SColorIO` command.
SCtrlEventAct

Control Event Action

Sets a control event action

**When p2 (type) is set to Off**

**Syntax**

\[ SCtrlEventAct, p1, p2 \]

- **p1** Registration number (1 to 100)
- **p2** Type (Off)

**When p2 (type) is set to DI, DO, or internal switch**

**Syntax**

\[ SCtrlEventAct, p1, p2, p3, p4, p5, p6, p7, p8, p9 \]

- **p1** Registration number (1 to 100)
- **p2** Type (DI, DO, SW)
  - **DI**
  - **DO**
  - **SW** Internal switch
- **p3** Number
  - When P2=SW: 1 to 100
  - When P2=DI or DO: 0001 to 6932
- **p4** Input/output type (In, Out)
  - In Varies depending on P2 (DI, DO, SW). See the table below.
  - Out Varies depending on P2 (DI, DO, SW). See the table below.
- **p5** Status output content or operation content
  - Varies depending on P2 (DI, DO, SW). See the table below.
- **p6** Pattern number
- **p7** Loop number

**For operation with p5=SELECT** (hold, advance, start of program operation, stop of program operation)

- **p6** Pattern type (PATTERN_NO, WCONST)
  - **PATTERN_NO** Program pattern number
  - **WCONST** Variable constant W
- **p7** Pattern number
- **p8** Loop type (LOOP_NO, WCONST)
  - **LOOP_NO** Loop number
  - **WCONST** Variable constant W
- **p9** Loop number

**When p5 is set to a value other than those above (table below)**

- **p6** Empty
- **p7** Element number corresponding to p5
- **p8** Loop number
- **p9** PV/time event number (1 to 32)

**Query**

\[ SCtrlEventAct [], p1 \]

**Example**

Set all loop control operation to stop when the internal switch number (SW1) becomes 0 to 1. Use Control event action number 1.

\[ SCtrlEventAct, 1, SW, 1, In, ALL_LP_STOP_ACT \]

**Description**

- This command can be used when a PID control module is installed and the program control function (/PG option) is enabled.
- If you need multiple channels for p2 (event type) and p5 (action), use DI/DO/SW in the same module. The table below shows the number of used channels.

<table>
<thead>
<tr>
<th>p2 Type</th>
<th>p4 Input/output type</th>
<th>p5 Status output/operation content</th>
<th>p5 Option strings</th>
<th>p7 Number of used channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>DI (PID control module or input type)</td>
<td>PROGRESET</td>
<td>PROG_RST_ACT</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>DI Number setting Bin (Patm1-2)</td>
<td>PTNNO_BIN_1-2_W</td>
<td>-</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>DI Number setting Bin (Patm1-4)</td>
<td>PTNNO_BIN_1-4_W</td>
<td>-</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>DI Number setting Bin (Patm1-8)</td>
<td>PTNNO_BIN_1-8_W</td>
<td>-</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>DI Number setting Bin (Patm1-16)</td>
<td>PTNNO_BIN_1-16_W</td>
<td>-</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>DI Number setting Bin (Patm1-32)</td>
<td>PTNNO_BIN_1-32_W</td>
<td>-</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>DI Number setting Bin (Patm1-64)</td>
<td>PTNNO_BIN_1-64_W</td>
<td>-</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>DI Number setting Bin (Patm1-99)</td>
<td>PTNNO_BIN_1-99_W</td>
<td>-</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>DI Number setting Bin (Patm1-199)</td>
<td>PTNNO_BIN_1-199_W</td>
<td>-</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>DI Number setting Bin (Patm1-199)</td>
<td>PTNNO_BIN_1-199_W</td>
<td>-</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>DI Number setting Bin (Patm1-199)</td>
<td>PTNNO_BIN_1-199_W</td>
<td>-</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

**2.4 Setting Commands**
### 2.4 Setting Commands

#### SLogicMath

**Logic Math Expression (MT)**
Sets the logic math expression

**Syntax**
```
SLogicMath,p1,p2,p3
```

- **p1** Math number
- **p2** Expression on/off
  - Off  Not Use
- **p3** Calculation expression (up to 120 alphanumeric characters, UTF-8)

**Query**
```
SLogicMath[,p1]?
```

**Example**
Output the math result of expression 0001AND0002 as 0 or 1 to DO channel number 0105. Use Logic math number 1.
```
SLogicMath,1,DO,0105,'0001AND0002'
```

**Description**
- You cannot use this command to configure settings while recording is in progress.
- p3 can be set only for DO or SW with type set to Manual.
- This command can be used when the math function (MT) is enabled.

#### SWConst

**Variable Constant (MT)**
Sets the variable constant to be used in computation

**Syntax**
```
SWConst,p1,p2
```

- **p1** Variable constant number (1 to 100)
- **p2** Value (–9.999999E+29 to 9.999999E+30, five significant digits)

**Query**
```
SWConst[,p1]?
```

**Example**
Set variable constant number 12 to 1.0000E–10.
```
SWConst,12,1.0000E-10
```

**Description**
- You can change the constant even during recording, computing, and controlling.
- This command can be used when the math function (MT) is enabled.
### 2.5 Output Commands

#### FData

**Outputs the Most Recent Channel Data**

Outputs the most recent I/O channel, math channel, and communication channel data.

**Syntax**

```
FData, p1, p2, p3
```

- **p1** Output format
  - 0 The most recent data in ASCII format
  - 1 The most recent data in binary format
- **p2** First channel
- **p3** Last channel

**Example**

Output the most recent data of channels 0001 to 0020 in ASCII format.

```
FData, 0, 0001, 0020
```

**Description**

- If you omit p2 and p3, all channels will be output.
- Channel ranges whose first channel and end channel are different channel types are interpreted as follows:

<table>
<thead>
<tr>
<th>First Channel</th>
<th>Last Channel</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>A200</td>
<td>A001 to A200, C001 to C500</td>
</tr>
<tr>
<td>A001</td>
<td>C500</td>
<td>Not allowed (will result in error)</td>
</tr>
<tr>
<td>C001</td>
<td>A200</td>
<td>Not allowed (will result in error)</td>
</tr>
<tr>
<td>A001</td>
<td>0001</td>
<td>Not allowed (will result in error)</td>
</tr>
</tbody>
</table>

- For the ASCII output format, see page 2-121.
- For the binary output format, see page 2-178.

#### FRelay

**Outputs the Most Recent Relay and Internal Switch Status**

Outputs the most recent relay (DO Channel) and internal switch status.

**Syntax**

```
FRelay, p1
```

- **p1** Output information
  - 0 The most recent relay (DO channel) status in ASCII format
  - 1 The most recent internal switch status in ASCII format

**Example**

Output the relay (DO channel) status.

```
FRelay, 0
```

**Description**

- For the output format, see page 2-122 or page 2-124.

#### FTransStatAO

**Latest Re-transmission State Output**

Outputs the latest re-transmission (AO channel) state.

**Syntax**

```
FTransStatAO, p1
```

- **p1** Fixed to 0

**Example**

Output the re-transmission state.

```
FTransStatAO, 0
```

**Description**

- For the output format, see page 2-124.

#### FFifoCur

**Outputs Channel FIFO Data**

Outputs the I/O channel, math channel, and communication channel FIFO data.

**Acquire the FIFO Data**

**Syntax**

```
FFifoCur, p1, p2, p3, p4, p5, p6, p7
```

- **p1** FIFO data output (0)
- **p2** Scan group (1 or 2)
- **p3** First channel
- **p4** Last channel
- **p5** Read start position (-1, 0 to 9999999999999)
  - -1 The most recent read position
- **p6** Read end position (-1, 0 to 9999999999999)
  - -1 The most recent read position
- **p7** Maximum number of blocks to read (1 to 9999)

**Example**

Read the measured data of channels 0001 to 0020. Set the read start position to 180 and the read end position to the most recent position. Set the maximum number of blocks to read to 9999.

```
FFifoCur, 0, 1, 0001, 0020, 180, -1, 9999
```

**Description**

- For the binary output format, see page 2-181.
  - **p2 = 2** is valid when the measurement mode is set to dual interval.

---

2-94

IM 04L51B01-17EN
### FSnap

**Snapshot [GX/GP]**
Outputs a snapshot data (screen image data) file.

**Syntax**  
`FSnap,p1`

  - `p1`: Screen image data output (GET)

**Example**  
Acquire screen image data.  
`FSnap,GET`

**Description**
- A PNG image file will be stored in the data block of the binary output file (see page 2-119).

### FUser

**Outputs the User Level**
Outputs information about the users who are currently logged in.

**Syntax**  
`FUser,p1`

  - `p1`: Information about the users who are currently logged in
    - 0: Refer to your own user information.
    - 1: Refer to information about all users who are currently logged in.
    - 2: Refer to information 2 of the user who is currently logged in.
    - 3: Refer to information 2 of all users who are currently logged in.

**Example**  
Refer to information about all users who are currently logged in.  
`FUser,1`

**Description**
- For the ASCII output format, see page 2-126.

### FAddr

**Outputs the IP Address**
Outputs the recorder IP address information.

**Syntax**  
`FAddr,p1`

  - `p1`: Address output (IP)
    - Output address information that includes the IP address, subnet mask, default gateway, and DNS server as well as the host name and domain name.

**Example**  
Output the recorder IP address information.  
`FAddr,IP`

**Description**
- For the ASCII output format, see page 2-130.

### FStat

**Outputs the Recorder Status**
Outputs the recorder status.

**Syntax**  
`FStat,p1`

  - `p1`: Status output (0)
    - 0: Status 1 to 4 output
    - 1: Status 1 to 8 output

**Example**  
Output the recorder status.  
`FStat,0`

**Description**
- For the ASCII output format, see page 2-131.

### FLog

**Outputs the Log**
Outputs the alarm summary, message summary, error log, etc.

**Syntax**  
`FLog,p1,p2,p3`

  - `p1`: Status output (0)
    - ALARM: Alarm summary
    - MSG: Message summary
    - EVENT: Event log
    - ERROR: Error log
    - DHCP: Ethernet address setting log
    - GENERAL: General communication log
    - MODBUS: Modbus log
    - FTP: FTP client log
    - SNTP: SNTP client log
    - MAIL: E-mail log
    - WEB: Web log
    - SLMP: SLMP log
    - CALARM: Control alarm summary log
    - CTRL: Control summary log
  - `p2`: Maximum log readout length
    - ALARM: 1 to 1000
    - MSG: 1 to 500
    - GENERAL: 1 to 200
    - MODBUS: 1 to 50 (1 to 200 for the GX20-2/GP20-2)
    - CALARM: 1 to 500
    - CTRL: 1 to 1000
    - Other than those above: 1 to 50
  - `p3`: Batch group number
    - All: All batch group numbers
    - 1 to the number used

**Example**  
Output 600 alarm summary entries.  
`FLog,ALARM,600`

**Description**
- For the ASCII output format, see page 2-134.
- `p3` is valid when multi batch is in use and `p1`=(alarm, msg, event). Omitting it is equivalent to specifying all batch groups.
2.5 Output Commands

FEventLog

Outputs a Detail Event Log (/AS)

Outputs an event log. You can specify the event, user, etc.

Syntax  
FEventLog, p1, p2, p3, p4, p5

<table>
<thead>
<tr>
<th>p1</th>
<th>Output format</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Same output format as Flog, EVENT (no detailed information).</td>
</tr>
<tr>
<td>1</td>
<td>Include detailed information</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p2</th>
<th>User name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up to five user names can be specified by separating each user with a colon.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p3</th>
<th>Event specification (specified with an event string)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up to five events can be specified by separating each user with a colon. Events will be searched using a prefix search.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p4</th>
<th>Maximum number of output (1 to 400)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>p5</th>
<th>Batch group number</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>All batch group numbers</td>
</tr>
<tr>
<td>1</td>
<td>to the Batch group number used</td>
</tr>
</tbody>
</table>

Example  
Output the log of up to 10 “message001” writing operations by User01.
FEventLog, 1, User01, Message001, 10

Description

• Omitting p2 is equivalent to specifying all users.
• If more than five users are specified by p2, only the first five users will be valid.
• Omitting p3 is equivalent to specifying all events.
• If more than five events are specified by p3, only the first five events will be valid.
• For the event strings of p3, see section 2.10.23 Detail Event Log Output (FEventLog) (/AS).”
• This command can be used only when the multi batch function (/BT) is enabled. Omitting p5 is equivalent to specifying all batch groups.

FMedia

Outputs External Storage Medium and Internal Memory Information

Outputs external storage medium and internal memory information.

Syntax  
FMedia, p1, p2, p3, p4

<table>
<thead>
<tr>
<th>p1</th>
<th>Output type (DIR)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>p2</th>
<th>Path name (up to 100 characters)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path name for outputting the file list</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p3</th>
<th>File list output start position (1 to 99999999)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>p4</th>
<th>File list output end position (1 to 99999999, -1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last position for outputting the file list. If you specify -1, the maximum possible number of files (as large as the recorder internal communication buffer allows) will be output.</td>
<td></td>
</tr>
</tbody>
</table>

Example  
Output all the file lists in the DRV0 directory.
FMedia, DIR, /DRV0/

Output the file lists of items 10 to 20 in the DRV0 directory.
FMedia, DIR, /DRV0/, 10, 20

Description

• Path names (p2) for the internal memory and the external media are listed below. Set the path name using a full path.
  Internal memory: /MEMO/DATA/
  SD memory card: /DRV0/
  USB flash memory: /USB0/
• If you omit p3 and p4, the maximum possible number of files (as many as the GX internal communication buffer allows) will be output.
• For the ASCII output format, see page 2-158.

Data in Files

Syntax  
FMedia, p1, p2, p3, p4

<table>
<thead>
<tr>
<th>p1</th>
<th>Output type (GET)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>p2</th>
<th>Path name (up to 100 characters)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path name of the file for outputting data</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p3</th>
<th>Data output start position (in bytes) (0 to 2147483647)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>p4</th>
<th>Data output end position (in bytes) (0 to 2147483647, -1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The last data output position. If you specify -1, the maximum file size (as large as the recorder internal communication buffer allows) will be output.</td>
<td></td>
</tr>
</tbody>
</table>

Example  
Output all the data in file xyz in the DRV0/ DATA0 directory.
FMedia, GET, /DRV0/DATA0/xyz

Description

• If you omit p3 and p4, the maximum file size (as large as the recorder internal communication buffer allows) will be output.
• The file data will be stored in the data block of the binary output file (see page 2-119).

Free Space on the External Storage Medium

Syntax  
FMedia, p1

| p1  | Output type (CHKDSK) |

Example  
Output the free space on the external storage medium.
FMedia, CHKDSK

Description

• For the ASCII output format, see page 2-158.
### FCnf

**Outputs Setting Data**

Outputs the recorder setting data.

**Syntax**  
FCnf,p1

**Operation**

<table>
<thead>
<tr>
<th>Setting category</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>Read all settings.</td>
</tr>
<tr>
<td>IO</td>
<td>Read I/O settings.</td>
</tr>
<tr>
<td>MATH</td>
<td>Read Math settings.</td>
</tr>
<tr>
<td>COMM</td>
<td>Read communication settings.</td>
</tr>
<tr>
<td>GROUP</td>
<td>Read display group settings.</td>
</tr>
<tr>
<td>IP</td>
<td>Read IP address settings.</td>
</tr>
<tr>
<td>SECURITY</td>
<td>Read security settings.</td>
</tr>
<tr>
<td>MULTIBATCH</td>
<td>Read multi batch settings.</td>
</tr>
<tr>
<td>CONTROL</td>
<td>Read control settings.</td>
</tr>
<tr>
<td>CALIB</td>
<td>Read calibration correction settings.</td>
</tr>
<tr>
<td>SERVER</td>
<td>Read Ethernet server related settings.</td>
</tr>
<tr>
<td>INSTRU</td>
<td>Read device information settings.</td>
</tr>
<tr>
<td>OTHERS</td>
<td>Read settings other than above.</td>
</tr>
</tbody>
</table>

You can specify multiple items in the list above. Separate each item with a colon (see the example).

**Example**

Read I/O and Math settings.

FCnf,IO:MATH

**Description**

- If you omit p1, all settings will be read.
- The setting data is output as the responses to the command queries. The following table lists p1 values (setting category) and the corresponding commands.

<table>
<thead>
<tr>
<th>Setting category</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>IO</td>
<td>SModeAI, SModeAICurrent, SModeDI, SModePID, SScaleOver, SB0LenAI, SB0LenAICurrent, SRangeAI, SRangeAIcurrent, SRangeDI, SRangepulse, SRangedo, SRangedO, SMoveAve, SFilt, SBurnout, SRjc, SA1arm10, SA1armDI10, STagAI, SColorAI, SZone1O, SScale1O, SBarIO, SPartial1O, SBarIO, SAl1arm10, SValueIO, SpresetA0</td>
</tr>
<tr>
<td>MATH</td>
<td>SMathBasic, SKConst, SWconst, SRangepmath, SRolAveMath, SA1armMath, SA1armMath, SA1armDI10Math, STagMath, SColorMath, SZoneMath, SScaleMath, SBarMath, SPartialMath, SBarMath, SA1armMath, SReport, SRepData, SRepTemp, SRepCh, SDigitalSign, SRepBatchInfo</td>
</tr>
</tbody>
</table>

---

**Commands and Responses**

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRangeCom, SValueCom, SWCom, SAlarmCom, SA1armHysCom, SA1armDI10Com, STagCom, SColorCom, SZoneCom, SScaleCom, SBarCom, SPartialCom, SBandCom, SA1armMarkCom, SCalibUseCom, SCalibCom</td>
</tr>
<tr>
<td>SGroup, STr1pline, SS11emP, SDualGroup</td>
</tr>
<tr>
<td>SIpAddress, SDns, SDhcp</td>
</tr>
<tr>
<td>SKdc, SSecurity, SopenPass, SopenLimit, SUSer, SUSerLimit, SSig1nIn, SSig1nInTitle, SSig1nInLimit, SBTPassword, SWebCustomMenu, SWebTimeOut, SSessionSecurity</td>
</tr>
<tr>
<td>SMT1extField, SM1tFileHead, SM1tFileName, SM1tGroup, SM1tTr1pline, SM1tS11emP</td>
</tr>
<tr>
<td>SCT1rMode, SC1tCtrlScan, SC1tAction, SCT1rType, SCT1rLoopAction, SCT1rSPFD, SCT1rALNo, SCT1rALmMode, SCT1rD1Regist, SCT1rRelay, SCT1rRangeAI, SCT1rBurnout, SCT1rRjc, SCT1rFilt, SCT1rCalibAI, SCT1rRangeAO, SCT1rS1iptA0, SCT1rOutput, SCT1rRangePV, SCT1rPVSwitch, SCT1rCalc, SCT1rFiltSP, SCT1rR1atioSP, SCT1rB1aseSP, SCT1rErrPreOut, SCT1rOutLimit, SCT1rAlarm, SCT1rAlarmVal, SCT1rSP, SCT1rSPGradient, SCT1rPIDNo, SCT1rRefPoint, SCT1rR1hy, SCT1rRefDEV, SCT1rPIDP1b, SCT1rPID1TI, SCT1rPID1T0, SCT1rPIDP1a, SCT1rRefT1, SCT1rRefT0, SCT1rRefPara, SCT1rDetail, SCT1rOutRatio, SCT1rAtDetail, SCT1rAntiReset, SCT1rOv1ershoot, SCT1rGroupSW, SCT1rGroupName, SCT1rGroupSplit, SCT1rGroup, SCT1rTag, SCT1rDispDV, SCT1rBackColor, SCT1rOutOperate, SPr1gColor, SPr1gDispDetail, SCT1rEv1entAct, SLogi1cMath, SWCon1st</td>
</tr>
</tbody>
</table>

---

For the output format, see page 2-159.
### FChInfo

**Outputs Decimal Place and Unit Information**

Outputs decimal place and unit information.

**Syntax**

\[ FChInfo, p1, p2 \]

- **p1** First channel
- **p2** Last channel

**Example**

Output the decimal place and unit information of channels 0001 to 0003.

\[ FChInfo, 0001, 0003 \]

**Description**

- If you omit \( p1 \) and \( p2 \), all channels will be output.
- For the output format, see page 2-159.

### FBTDevInfo

**Bluetooth Device Information Output [GM]**

Outputs the Bluetooth device information of the recorder.

**Syntax**

\[ FBTDevInfo, p1 \]

- **p1** Bluetooth device information output (0)

**Example**

Output the Bluetooth device information of the connected device.

\[ FBTDevInfo, 0 \]

**Description**

- \( p1 \) can be omitted.
- For the output format, see page 2-162.

### FSysConf

**Queries the System Configuration and Reconfigures Modules**

Queries the System Configuration, Reconfigures Modules, and Performs Activation.

**Query the System Configuration**

**Syntax**

\[ FSysConf \]

**Example**

Query the System Configuration.

\[ FSysConf \]

**Description**

- For the output format, see page 2-160.

**Reconfigures Modules**

Aligns the module configuration settings that are recognized by the recorder and the actual module configuration.

**Syntax**

\[ FSysConf, p1 \]

- **p1** Module reconfiguration (1)

**Example**

Reconfigure the modules.

\[ FSysConf, 1 \]

**Activate module**

Modules need to be activated when the firmware in installed modules is updated or when modules are recalibrated.

**Syntax**

\[ FSysConf, p1 \]

- **p1** Activate module (3)

**Example**

Activate modules.

\[ FSysConf, 3 \]

**Description**

- \( p1 = 3 \) is valid when the advanced security function (/AS) is enabled.

### FReminder

**Outputs Reminder Information (/AH)**

Outputs reminder information.

**Syntax**

\[ FReminder, p1 \]

- **p1** Reminder information designation
  - ALL Specifies all schedule numbers
  - 1 or 12 Schedule number

**Example**

Output the reminder information for schedule number 3.

\[ FReminder, 3 \]

**Description**

- This command is valid when the aerospace heat treatment (/AH) option is enabled.
- Omitting \( p1 \) is equivalent to specifying all registration numbers.
- For the output format, see page 2-163.

### FCtrlData

**Control Data Output**

Outputs the most recent control data.

**Syntax**

\[ FCtrlData, p1, p2, p3 \]

- **p1** Output format
  - 0 ASCII format
  - 1 Binary format
- **p2** First loop number
- **p3** Last loop number

**Example**

Output the most recent data of loop number 001.

\[ FCtrlData, L001 \]

**Description**

- If you omit \( p2 \) and \( p3 \), all loop numbers will be output.
- Only the information of detected modules will be output.
- The data time outputted is not the time of a control period but the data acquisition time by a communication command.
- For the ASCII output format, see page 2-164.
- For the Binary output format, see page 2-182.
### FCtrlNo

**SP Number and PID Number Output**

Outputs the SP number and PID number

**Syntax**

```
FCtrlNo,p1,p2
```

- `p1` First loop number
- `p2` Last loop number

**Example**

Output the SP number and PID number of loop number 001 currently in use.

```
FCtrlNo,L001
```

**Description**

- If you omit `p1` and `p2`, all loop numbers will be output.
- If you omit `p2`, `p2` is set to the same loop number as `p1`.
- Only the information of detected modules will be output.
- For the output format, see page 2-165.

---

### FCtrlMode

**Control Mode Output**

Outputs the control mode

**Syntax**

```
FCtrlMode,p1,p2
```

- `p1` First loop number
- `p2` Last loop number

**Example**

Output the control mode of loop number 001.

```
FCtrlMode,L001
```

**Description**

- If you omit `p1` and `p2`, all loop numbers will be output.
- If you omit `p2`, `p2` is set to the same loop number as `p1`.
- Only the information of detected modules will be output.
- For the output format, see page 2-166.

---

### FPrgMode (/PG)

**Program Operation Mode Output**

Outputs the program operation mode

**Syntax**

```
FPrgMode,p1
```

- `p1` Program pattern number (1 to 99)

**Example**

Output the program operation mode of program pattern number 1.

```
FPrgMode,1
```

**Description**

- This command is valid when the program control function (/PG option) is in use.
- During program operation, if the selected program pattern is not being executed, an error occurs.
- When program operation is stopped, the information at the end of operation is output only when the selected program pattern matches the program pattern that was executed last.
- For the output format, see page 2-167.

---

### FPrgPtnInfo (/PG)

**Program Pattern Information Output**

Outputs the program pattern status

**Syntax**

```
FPrgPtnInfo,p1
```

- `p1` Program pattern number (1 to 99)

**Example**

Output the program pattern information of program pattern number 1.

```
FPrgPtnInfo,1
```

**Description**

- This command is valid when the program control function (/PG option) is in use.
- During program operation, if the selected program pattern is not being executed, an error occurs.
- When program operation is stopped, the information at the end of operation is output only when the selected program pattern matches the program pattern that was executed last.
- For the output format, see page 2-168.

---

### FPrgEvent (/PG)

**PV Event and Time Event Information Output**

Outputs information about PV events and time events occurring in the program pattern

**Syntax**

```
FPrgEvent,p1
```

- `p1` Program pattern number (1 to 99)

**Example**

Output the PV events and time events of program pattern number 1.

```
FPrgEvent,1
```

**Description**

- This command is valid when the program control function (/PG option) is in use.
- During program operation, if the selected program pattern is not being executed, an error occurs.
- When program operation is stopped, "0" is output for PV events and time events.
- For the output format, see page 2-169.

---

### FPrgEnd (/PG)

**Program Control End Signal Status Output**

Outputs the program control end signal status

**Syntax**

```
FPrgEnd,p1
```

- `p1` Program pattern number (1 to 99)

**Example**

Output the program control end signal status of program pattern number 1.

```
FPrgEnd,1
```

**Description**

- This command is valid when the program control function (/PG option) is in use.
- When program operation is stopped, "0" is output when the operation is terminated by force (terminated through reset).
- For the output format, see page 2-170.
2.5 Output Commands

**FPrgPtnCur (/PG)**

**Running Program Pattern Number and Status Output**

Outputs the running program pattern number and status

**Syntax**  
FPrgPtnCur

**Example**  
Output the running program pattern number and status.  
FPrgPtnCur

**Description**

- This command is valid when the program control function (/PG option) is in use.
- When program operation is stopped, EA<crlf>EN<crlf> is returned.
- For the output format, see page 2-170.
## 2.6 Operation Commands

### OSetTime

**Sets the Time**

Sets the time.

**Syntax**

```plaintext
OSetTime, p1
```

- `p1` Time to set
  - “YYYY/MO/DD:HH:MI:SS” (the underscore denote a space), “YYYY/MO/DD”, or “HH:MI:SS.”
  - **YYYY** Year (2001 to 2035)
  - **MO** Month (01 to 12)
  - **DD** Day (01 to 31)
  - **HH** Hour (00 to 23)
  - **MI** Minute (00 to 59)
  - **SS** Second (00 to 59)

**Query**

```plaintext
OSetTime?
```

The `OSetTime` query outputs the recorder current time.

**Example**

Set the time to 23:00:00 on May 24, 2013.

```
OSetTime, 2013/05/24 23:00:00
```

### ORec

**Starts or Stops Recording**

Starts or stops recording.

**Syntax**

```plaintext
ORec, p1, p2
```

- `p1` Recording start or stop
  - 0 Start
  - 1 Stop

- `p2` Batch group number
  - **All** All batch group numbers
  - 1 to the number used

**Query**

```plaintext
ORec?, ORec?, [p1[, p2]]? (when multi batch is enabled) (p1 is any value)
```

**Example**

Start recording.

```
ORec, 0
```

**Description**

- p2 is valid when multi batch (/BT) is enabled.
- Omitting p2 is equivalent to specifying all batch group numbers.

### OAlarmAck

**Clears Alarm Output**

Clears alarm output (performs an alarm ACK).

**Syntax**

```plaintext
OAlarmAck, pl, p2
```

- `p1` Alarm output clearance (0)
  - 0 Clear the alarm output.

- `p2` Batch group number
  - **All** All batch group numbers
  - 1 to the number used

**Example**

Clear the alarm output.

```
OAlarmAck, 0
```

**Individual alarm ACK**

**Syntax**

```plaintext
OAlarmAck, pl, p2, p3
```

- `p1` Individual alarm output clearance (1)
  - 1 Separate alarm output clearance (2)

- `p2` Channel number
  - 1 Loop number

- `p3` Alarm level (1 to 4)

**Example**

Clear the alarm output of alarm 3 of channel 0001.

```
OAlarmAck, 1, 0001, 3
```

**Individual Alarm ACK (for control alarms)**

**Syntax**

```plaintext
OAlarmAck, pl, p2
```

- `p1` Separate alarm output clearance (2)

- `p2` Loop number

**Example**

Clear the alarm output of Loop L122.

```
OAlarmAck, 2, 122
```

**Description**

- If you send an individual alarm ACK command when the individual alarm ACK function is not in use, no action is taken, and a normal response is returned.

### OExecRec

**Generates a Manual Trigger, Executes Manual Sample, Takes a Snapshot, or Causes a Timeout**

Generates a manual trigger, executes manual sample, takes a snapshot, or divides the data being recorded into separate files.

**Syntax**

```plaintext
OExecRec, p1, p2
```

- `p1` Action type
  - 0 Execute manual sampling.
  - 1 Generate a manual trigger.
  - 2 Take a snapshot.
  - 3 Cause a display data timeout (divide files).
  - 4 Cause an event data timeout (divide files).

- `p2` Batch group number
  - **All** All batch group numbers
  - 1 to the number used

**Example**

Execute manual sampling.

```
OExecRec, 0
```

**When the measurement mode is set to dual interval**

- `p1` Action type
  - 1 Generate a manual trigger.

- `p2` Scan group number
### OMath

**Starts, Stops, or Resets Computation or Clears the Computation Dropout Status Display**

Starts or stops computation, resets computed values, or clears the computation dropout status display.

**Syntax**

```
OMath, p1, p2
```

- **p1** Action type (0)
  - 0: Start computation
  - 1: Stop computation
  - 2: Reset computation
  - 3: Clear the computation dropout status display

- **p2** Batch group number
  - **All**: All math channels
  - **1** to the number specified batch group used

**Example**

Start computation.

```
OMath, 0
```

**Description**

- You cannot use this command while the recorder is saving or loading setup data.
- You can specify multiple groups at once. To do so, separate display groups with a colon.

```
P4 Message string to write (up to 32 characters, UTF-8)
P5 Batch group number (1 to the number used)
```

**Example**

Write a free message "MARK" as message number 2 in display groups 3, 8, and 11.

```
OMessage, FREE, 2, 3:8:11, 'MARK'
```

**Description**

- **p5** is valid when multi batch is enabled. This cannot be omitted.

---

**OMessage**

**Message Writing**

Writes a message.

**Write a Preset Message**

**Syntax**

```
OMessage, p1, p2, p3, p4
```

- **p1** Action type (PRESET)
- **p2** Message number (1 to 100)
- **p3** Display group number
  - **ALL**: Write to all display groups
  - **1** to **60**: Write to specified groups

You can specify multiple groups at once. To do so, separate display groups with a colon.

```
P4 Batch group number (1 to the number used)
```

**Example**

Write the message in preset message number 8 to display groups 1 and 2.

```
OMessage, PRESET, 8, 1:2
```

**Description**

- **p4** is valid when multi batch is enabled. This cannot be omitted.

**Write a Free Message**

**Syntax**

```
OMessage, p1, p2, p3, p4, p5
```

- **p1** Action type (FREE)
- **p2** Message number (1 to 10)
- **p3** Display group number
  - **ALL**: Write to all display groups
  - **1** to **60**: Write to specified groups

- **p4** Message string to write (up to 32 characters, UTF-8)
- **p5** Batch group number (1 to the number used)

**Example**

Write a free message "MARK" as message number 2 in display groups 3, 8, and 11.

```
OMessage, FREE, 2, 3:8:11, 'MARK'
```

**Description**

- **p5** is valid when multi batch is enabled. This cannot be omitted.

---

**OPassword**

**Changes the Password**

Changes the password.

**Syntax**

```
OPassword, p1, p2, p3
```

- **p1** Old password (up to 20 characters, ASCII)
- **p2** New password (up to 20 characters, ASCII)
- **p3** New password (enter the same password as **p2**)

**Example**

Change the password from "PASS001" to "WORD005."

```
OPassword, 'PASS001', 'WORD005', 'WORD005'
```

**Description**

- For the characters that you can use for the password, see **Appendix 1**.

---

### OExecSNTP

**Queries the Time Using SNTP**

Queries the time using SNTP.

**Syntax**

```
OExecSNTP, p1
```

- **p1** Time query execution (0)

**Example**

Query the time using SNTP.

```
OExecSNTP, 0
```

---

### OExecRec

**All scan groups**

- **1**: Scan group 1
- **2**: Scan group 2

**Example**

When the measurement mode is dual interval, apply a trigger to scan group 2.

```
OExecRec, 1, 2
```

**Description**

- Manual trigger (p1 = 1) cannot be executed when the advanced security function (/AS) is enabled.
- If a manual sample is executed (p1 = 0) when there are no source channels for manual sampling, a file without any source channels will be created.
- **p2** is valid when multi batch is enabled and p1=3 or 4.
- **p2** is valid when the measurement mode is set to dual interval and p1=1.
• p2 is valid when multi batch is enabled and p1=2 (reset computation).
• Omitting p2 is equivalent to specifying all math channels.

**OSaveConf**

**Saves Setting Data**

Saves the recorder setting data to the recorder’s external storage medium.

**Syntax**

\[
\text{OSaveConf}, p1, p2, p3 \\
p1 \text{ File name (up to 80 characters, ASCII)} / \text{Specify the path and file name, excluding the extension.} \\
p2 \text{ Medium} / \begin{aligned} & \text{(GX/GP: SD, USB)} \\
& \text{(GM: SD)} \\
& \text{SD} \quad \text{SD memory card} \\
& \text{USB} \quad \text{USB flash memory} \\
p3 \text{ Setting file comment (up to 50 characters, UTF-8)} / \end{aligned}
\]

**Example**  
Save setting data to a file named “SETFILE1” to the SD memory card.

\[
\text{OSaveConf}, \text{’SETFILE1’}, \text{SD}
\]

**Description**

• If you omit p3, the default setting file comment will be added. You can edit the default setting file comment from the recorder front panel.

**OSaveConfAll**

**Saves Setting Data at Once**

Saves the setting data to the specified folder in the external storage medium.

**Syntax**

\[
\text{OSaveConfAll}, p1, p2 \\
p1 \text{ Folder name (up to 80 characters, ASCII)} / \text{Specify the path name+f folder name.} \\
p2 \text{ Medium} / \begin{aligned} & \text{(GX/GP: SD, USB)} \\
& \text{(GM: SD)} \\
& \text{SD} \quad \text{SD card} \\
& \text{USB} \quad \text{USB Flash Memory} \\
\end{aligned}
\]

**Example**  
Save the setting data collectively to the “CONFIG0” folder of the SD memory card.

\[
\text{OSaveConfAll}, \text{’CONFIG0’}, \text{SD}
\]

**Description**

• If you omit parameter p2, the medium is set to the SD card.
• The following items are saved. File names are indicated in parentheses.
  • Setting data file (Config.GNL or Config.GSL)
  • Scale image [GX/GP only]  
    (ScaleImageXX.png) where XX is the display group
  • Report template (Report_YY.xlsx, Report_YY.xlsm, or Report_YY.tpl)  
    YY is the report type.
  • Trusted certificate

**A “Client” folder is created in the specified folder (p1), and the data is saved there.**

**OCommCh**

**Sets a Communication Channel to a value**

Sets a communication channel to a value.

**Syntax**

\[
\text{OCommCh}, p1, p2 \\
p1 \text{ Communication channel} \\
p2 \text{ Value} / \begin{aligned} & \text{The setting range is as follows:} \\
& -9.9999999E+29 \text{ to } -1.0000000E+30, \quad 0, \\
& 1.0000000E-30 \text{ to } 9.9999999E+29 \\
& \text{The number of significant digits is 8.} \\
\end{aligned}
\]

**Query**

\[
\text{OCommCh},[p1]?
\]

**Example**  
Set communication channel C001 to 2.5350.

\[
\text{OCommCh, C001, 2.5350}
\]

**Description**

• The description of execution and response errors are not recorded in the event log.
• Custom display commands cannot be executed.

**OEmail**

**Starts or Stops the E-mail Transmission Function**

Starts or stops the e-mail transmission function.

**Syntax**

\[
\text{OEmail}, p1 \\
p1 \text{ Action type} / \begin{aligned} & 0 \text{ Start the e-mail transmission function.} \\
& 1 \text{ Stop the e-mail transmission function.} \\
\end{aligned}
\]

**Example**  
Start the e-mail transmission function.

\[
\text{OEmail, 0}
\]
### OMBRestore

**Recover Modbus Manually**
Resumes command transmission from Modbus client or Modbus master to devices in which communication errors have occurred.

**Syntax**

OMBRestore, p1

- p1 Action type
  - 0 Modbus client (Ethernet)
  - 1 Modbus master (serial)

**Example**

Manually recover the Modbus client.

OMBRestore, 0

### ORTReset

**Resets a Relative Timer**
Resets a relational timer.

**Syntax**

ORTReset, p1

- p1 Timer type
  - 0 All timers
  - 1 to 12 Timer number

Multiple selection is possible by delimiting with colons.

**Example**

Reset relative timer 2.

ORTReset, 2

Reset relative timers 4, 9, and 12.

ORTReset, 4:9:12

### OMTReset

**Resets the Match Time Timer**
Resets the match time timer.

**Syntax**

OMTReset, p1

- p1 Timer type
  - 0 All timers
  - 1 to 12 Timer number

Multiple selection is possible by delimiting with colons.

**Example**

Reset match time timer 2.

OMTReset, 2

Reset match time timers 4, 9, and 12.

OMTReset, 4:9:12

### OCmdRelay

**Outputs the DO Channel and Internal Switch Status**
Outputs the DO channel and internal switch status.

**Syntax**

OCmdRelay, p1

- p1 Specification of a setting
  - Express the setting. Set a channel status as follows: [channel number]-[status]. Use a hyphen as a separator.

- You can specify the following values for the channel number.
  - DO channel number
  - Internal switch number

- You can specify the following values for the status.
  - Off: Off status
  - On: On status

- You can specify the status of multiple channels at once. To do so, use a semicolon to separate channels as follows: [channel number]-[status]:[channel number]-[status]:... You can specify up to a total of 32 channels that consist of DO channels and internal switches.

**Example**

Set channels 0101, 0102, and 0103 to On and internal switches S001 and S002 to Off.


**Description**

- If any of the channels that you specify do not exist or are not set to manual output (SRangeDO command), the settings of all channels are canceled, and a command error results.

### OBatName

**Sets a Batch Name**
Sets a batch name.

**Syntax**

OBatName, p1

- p1 Batch group number
  - When multi batch is disabled: Always 1
  - When multi batch is enabled: 1 to the number used

- p2 Batch number (up to 32 characters, ASCII)

- p3 Lot number (0 to 99999999, up to eight digits, depending on Lot-No. digit)

**Query**

OBatName[,p1]?

**Example**

Set the batch name structure to batch number “PRESSLINE” and the lot number 007.

OBatName, 1,’PRESSLINE’,007

**Description**

- For the characters that you can use in the specified batch number (p2), see Appendix 1.
- The character limitations on the batch number (p2) are the same as those for directory names. See the explanation for the SDirectory command.
- You cannot set the batch number to a single space character. Doing so will clear the batch number.
### OBatComment

**Sets a Batch Comment**

Sets a batch comment.

**Syntax**

\[ \text{OBatComment}, p1, p2, p3 \]

- **p1**: Batch group number
  - When multi batch is disabled: Always 1
  - When multi batch is enabled: 1 to the number used
- **p2**: Comment number (1 to 3)
- **p3**: Comment string (up to 50 characters, UTF-8)

**Query**

\[ \text{OBatComment}, p1[, p2] \]

**Example**

Set comment number 2 to "THIS PRODUCT IS COMPLETED."

\[ \text{OBatComment}, 1, 2, 'THIS PRODUCT IS COMPLETED' \]

**Description**

- You cannot set the comment string to a single space character.
  - Doing so will clear the comment string.

### OBatText

**Sets a Batch Text**

Sets a batch text.

**Syntax**

\[ \text{OBatText}, p1, p2, p3, p4 \]

- **p1**: Batch group number
  - When multi batch is disabled: Always 1
  - When multi batch is enabled: 1 to the number used
- **p2**: Field number (1 to 24)
- **p3**: Field title (up to 20 characters, UTF-8)
- **p4**: Field string (up to 30 characters, UTF-8)

**Query**

\[ \text{OBatText}, p1[, p2] \]

**Example**

For field number 1, set the title to “Ope” and the character string to “GX.”

\[ \text{OBatText}, 1, 'Ope', 'GX' \]

**Description**

- You cannot set the field title or field string to a single space character.
  - Doing so will clear them.

### ODispRate

**Switches the Trend Interval [GX/GP]**

Switches between first trend interval (normal trend interval) and second trend interval.

**Syntax**

\[ \text{ODispRate}, p1 \]

- **p1**: Trend interval
  - NORMAL: First trend interval (normal trend interval)
  - SECOND: Second trend interval

**Example**

Switch from first trend interval to second trend interval.

\[ \text{ODispRate}, \text{SECOND} \]

**Description**

- Set the second trend interval with the \text{STrdRate} command.

### OLoadConf

**Loads Setting Data**

Loads a setting data file from the recorder external storage medium into the recorder.

**Syntax**

\[ \text{OLoadConf}, p1, p2, p3, p4 \]

- **p1**: File name (up to 80 characters, ASCII)
  - Specify the path and file name, excluding the extension.
- **p2**: Medium
  - (GX/GP: SD, USB)
  - (GM: SD)
  - SD: SD memory card
  - USB: USB flash memory
- **p3**: Settings to load
  - ALL: All settings
  - SECURITY: Security settings only
  - CONTROL: Control
  - IP: IP address settings only
  - OTHERS: All settings except for security and IP address settings

Multiple options can be selected for p3. To do so, separate items with a colon.

- **p4**: Setting items to be excluded from the items specified by p3=OTHERS.
  - SERVER: Server related settings
  - CALIB: Calibration correction settings
  - INSTRU: Instrument information settings

Multiple options can be selected for p4. To do so, separate items with a colon. If p3 is set to ALL, nothing is excluded.

**Example**

Load all settings from the setting file “SETTING1” on the SD memory card.

\[ \text{OLoadConf}, 'SETTING1', SD, ALL \]

Load security and IP address settings from a setting file named “SETTING1” from the SD memory card.

\[ \text{OLoadConf}, 'SETTING1', SD, SECURITY:IP \]

Load settings excluding IP address settings, server related settings, and instrument information, from a setting file named “SETTING1” from the SD memory card.

\[ \text{OLoadConf}, 'SETTING1', SD, SECURITY:OTHERS, SERVER:INSTRU \]

**Description**

- If you omit parameter p2, the medium is set to the SD memory card.
- For p3 and p4 values (setting category) and target commands, see Setting Category and Target Commands on page 2-97.
- If you omit parameter p3, all settings will be loaded.
- If you omit parameter p4, no setting will be excluded.
- If you change the language with this command, the recorder may restart.
2.6 Operation Commands

**OLoadConfAll**

Loads Setting Data at Once

Loads all settings from the specified folder of the external storage medium.

**Syntax**

```
OLoadConfAll,p1,p2
```

- **p1** Folder name (up to 80 characters)
- **p2** Medium (SD)
  - (GX/GP: SD, USB)
  - (GM: SD)
  - SD card
  - USB flash memory

**Example**

Load all settings from the "CONFIG0" folder of the SD card.

```
OLoadConfAll,'CONFIG0',SD
```

**Description**

- The following items are loaded into the GX/GP/GM.
  - File names are indicated in parentheses.
  - Setting data file (Config.GNL or Config.GSL)
  - Scale image [GX/GP only] (ScaleImageXX.png) where XX is the display group
  - Report template (Report_YY.xlsx, Report_YY.xlsm, or Report_YY.tpl)
  - YY is the report type.
  - Trusted certificate
    - The certificate file in the "Client" folder in the specified folder (p1) is loaded.
  - Custom display (GX/GP only)
    - Loads the (Setting.GCS) setting file and the settings in each folder indicating a custom display number in the specified folder (p1).
  - Program pattern file (*.GPT)
    - The program pattern file in the "ProgramPattern" folder in the specified folder (p1) is loaded.

**OSeriApply**

Applies Serial Communication Settings

Applies serial communication settings.

**Syntax**

```
OSeriApply,p1
```

- **p1** Apply the settings (0).

**Example**

Apply serial communication settings.

```
OSeriApply,0
```

**Description**

- This command applies the serial communication settings specified by the **SSerialBasic** command.
- When you send this command, the serial communication settings take effect when the recorder returns a response. After this process, the connection will be cut off.

**OIPApply**

Applies the IP Address

Applies Ethernet communication settings.

**Syntax**

```
OIPApply,p1
```

- **p1** Apply the settings (0).

**Example**

Apply the IP address settings.

```
OIPApply,0
```

**Description**

- This command applies the IP address settings specified by the **SIpAddress**, **SDhcp**, **SDns**, and **SServer** commands.
- When you send this command, the IP address settings take effect when the recorder returns a response. After this process, the connection will be cut off. This includes Ethernet connections to other devices (Modbus server, FTP server, etc.).

**OInit**

Clears Measured Data and Initializes Setting Data

Clears the measured data in internal memory. The command also initializes setting data.

**Syntax**

```
OInit,p1,p2
```

- **p1** The types of data to be initialized and cleared
  - SECURITY Security settings
  - Memory Display data, event data, manual sampled data, report data, alarm summary, message summary
  - OTHERS Settings other than those above
  - ALL All measured data and settings
- **p2** Setting items to be excluded from the items specified by p1=OTHERS.
  - IP IP address settings
  - SERVER Server related settings
  - CALIB Calibration correction settings
  - INSTRU Instrument information settings
  - You can specify multiple items at once. To do so, separate items with a colon. If p1 is set to ALL, nothing is excluded.

**Example**

Delete the measured data and summary from the internal memory.

```
OInit,MEMORY
```

```
Initialize the settings excluding IP address settings and instrument information.
```

```
OInit,MEMORY:SECURITY:OTHERS,IP:INSTRU
```

**Description**

- IP address settings are those set with the **SIpAddress**, **SDns**, **SDhcp**, and **SDhcp** commands
- For p1 and p2 values (setting category) and target commands, see Setting Category and Target Commands on page 2-97.
- If you omit parameter p2, no setting will be excluded.
### OUsbFApply

**Applies USB Communication Settings**

[GM]

Applies USB communication settings.

**Syntax**

OUsbFApply,p1

  p1 Apply the settings

**Example**

Apply the USB communication On/Off setting specified with the SUsbFunction command.

OUsbFApply,0

### OBTApply

**Applies Bluetooth Communication Settings (/C8)**

[GM]

Applies Bluetooth communication settings.

**Syntax**

OBTApply,p1

  p1 Apply the settings (0)

**Example**

Apply the Bluetooth communication On/Off setting specified with the SBluetooth command.

OBTApply,0

### OBTClearList

**Clears the Bluetooth Connection List (/C8)**

[GM]

Clears the Bluetooth connection list.

**Syntax**

OBTClearList

(No parameters)

**Example**

Clear the connected Bluetooth connection list.

OBTClearList

### OLoginAssist

**Assists Login [GX/GP]**

Assists logging in to the recorder, during bar-code input.

**Syntax**

OLoginAssist,p1,p2,p3

  p1 Input type (1, 2)

    1 User name input
    2 User name and user ID input

  p2 User name

  p3 User ID

**Example**

Log in with the user name “User01.”

OLoginAssist,1,’User01’

**Description**

- When this command is executed, the recorder shows the login screen and waits for a user password and user ID input.
- p1 = 2 is valid when the advanced security function (/AS) is enabled.
- p3 is valid when p1 = 2. However, when the user ID is not used, p3 is invalid.
- This command is valid when the serial communication function (the SSerialBasic command) is set to Barcode or the USB input device (the SUSbInput command) is set to Barcode.

### OSendValue

**Assists Touch Panel Operation Input [GX/GP]**

Assists text input during touch panel operation.

**Syntax**

OSendValue,p1,p2

  p1 Fixed to 0.
  p2 Character string (up to 64 characters, UTF-8)

**Example**

On the message settings screen, enter the message “START” (display the message settings screen and select the text box for entering the message string in advance).

OSendValue,0,’START’

**Description**

- Input into a text area that displays asterisks (*****) is not possible.
- This command is valid when the serial communication function (the SSerialBasic command) is set to Barcode or the USB input device (the SUSbInput command) is set to Barcode.

### OUserLockACK

**User Locked ACK (/AS)**

Clears the user locked display.

**Syntax**

OUserLockACK

**Example**

Clears the user locked display.

OUserLockACK

**Description**

- This command can be executed only when the user is logged in as an administrator.
- If there are no locked users, nothing will take place.

### OKeyLock

**Key Lock On/Off [GM]**

Turns key lock on or off.

**Syntax**

OKeyLock,p1

  p1 Key lock on/off (On, Off)

    On Locks the keys
    Off Releases the key lock

**Example**

Release the key lock.

OKeyLock,Off

**Description**

- Turning the key lock on will lock the START, STOP, USER1, and USER2 keys. You cannot lock the key individually.
- Only administrator level users can turn key lock on and off.
- This command is invalid when the advanced security function (/AS) is enabled and the log in via communication is enabled.
2.6 Operation Commands

**OErrorClear**
Clears the Error Display [GM]
Cuts the error display status from the 7 segment LED.

**Syntax**
```
OErrorClear,p1
```

**p1**
Error display clear type
0 Error display clear

**Example**
Clear the error display status from the 7 segment LED.
```
OErrorClear,0
```

**OSLMPRestore**
Manually Restores SLMP (/E4)
Resumes command transmission from SLMP client to devices in which communication errors have occurred.

**Syntax**
```
OSLMPRestore,p1
```

**p1**
Fixed to 0

**Example**
Manually recovery the SLMP client.
```
OSLMPRestore,0
```

**OTransChAO**
Individual Re-transmission Control
Controls the re-transmission of AO channels individually.

**Syntax**
```
OTransChAO,p1
```

**p1**
Re-transmission value specification

Express the re-transmission value.

- Express the setting. Set a channel status as follows: [channel number]-[status]. Use a hyphen as a separator.
- You can specify the following values for the status.
  - Off: Off status
  - On: On status
- You can specify the status of multiple channels at once. To do so, use a semicolon to separate channels as follows: [channel number]-[status]:[channel number]-[status]:... You can specify up to 32 channels.

**Example**
Set re-transmission of channels 0101, 0102, and 0103 to On and that of channels 201 and 202 to Off.
```
```

**Description**
- This command is valid only for channels set to re-transmission (Trans) with the SRangeAO command.
- To check the re-transmission state, use FTransStatAO.

**OTransAllAO**
Collective re-transmission control
Controls the re-transmission of AO channels collectively

**Syntax**
```
OTransAllAO,p1
```

**p1**
Re-transmission enabled or disabled.
On Re-transmission is enabled.
Off Re-transmission is disabled.

**Example**
Set re-transmission to Off.
```
OTransAllAO, Off
```

**Description**
- This command is valid only for channels set to re-transmission (Trans) with the SRangeAO command.
- To check the re-transmission state, use FTransStatAO.

**OCmdAO**
Manual output setting
Sets the manual output value of an AO channel.

**Syntax**
```
OCmdAO,p1,p2
```

**p1**
Channel number

**p2**
Manual output value

**Query**
```
OCmdAO[,p1]
```

An OCmdAO query outputs the setting.

**Example**
Set the manual output value of channel 001 to 10 mA.
```
OCmdAO,001,10000
```

**Description**
- This command is valid only for channels set to manual output (ManualAO) with the SRangeAO command.
- The output range is the span range specified with the SRangeAO command.
- For p2, enter the value excluding the decimal point. (The decimal place is fixed to 3.)
- The description of execution and response errors are not recorded in the event log.
- Custom display commands cannot be executed.

**OInitPara**
Individual Setting Parameter Initialization
Initializes setting parameters individually

**Syntax**
```
OInitPara,p1
```

**p1**
Setting parameters to initialize

- RecCh Recording channels
- Group Display groups

- You can specify multiple setting parameters at once. To do so, separate each parameter with a colon as in RecCh:Group.

**Example**
Initialize recording channels and display groups.
```
OInitPara,RecCh:Group
```

**Description**
- Recording channels are those specified by the SrecDisp, SrecEvent, or SrecManual command.
• Display groups are those specified by the SGroup, STrLine, SSclBmp, SMltGroup, SMltTripLine, SMltSclBmp, or SDualGroup commands.

**OCtlAM**

**Auto/Manual/Cascade Operation Switching**

Switches the control operation mode

**Syntax**

OCtlAM,p1,p2

- **p1** Loop number
- **p2** Auto/manual/cascade switching
  0 Auto (Auto)
  1 Manual (Man)
  2 Cascade (Cas)

**Example**

Set the operation mode of unit 1, slot 5, loop 2 to manual.

OCtlAM,L152,1

**Description**

• This command is valid when a PID Control Module is installed.
• If p2=2 (Cas), E0 is returned when cascade control is enabled and E1 when disabled.
• When the module is not installed, E1 is returned.

**OCtlAT**

**Auto-Tuning Request**

Starts or stops auto-tuning

**Syntax**

OCtlAT,p1,p2

- **p1** Loop number
- **p2** Auto-tuning start/stop
  0 Auto-tuning stop
  1 to 8 Start auto-tuning PID numbers
  9 Start auto-tuning reference deviation PID

**Example**

Start auto-tuning of PID number 5 of loop 1.

OCtlAT,L001,5

**Description**

• This command is valid when a PID Control Module is installed.
• The response when start or stop is specified with p2 is as follows according to the status at that point.
  - Change from a stopped state: E0
  - Stop from a running state: E0
  - Start from a running state: E1
• When the module is not installed, E1 is returned.

**OCtlSR**

**Operation Run/Stop Switching**

Switches between operation run (RUN) and operation stop (STOP)

**Syntax**

OCtlSR,p1,p2

- **p1** Loop number (L000 or L001 to L652)
- **p2** Operation run/stop switching
  0 Run
  1 Stop

**Example**

Stop the operation of unit 1, slot 5, loop 2.

OCtlSR,L152,1

**Description**

• This command is valid when a PID Control Module is installed.
• When p1=L000, all loops are specified.
• Operation is not possible while a program pattern is being executed. (Possible in local mode)

**OCtlSPN**

**Selects the Target Setpoint Number**

Sets the target setpoint (SP) number

**Syntax**

OCtlSPN,p1,p2

- **p1** Loop number
- **p2** Target setpoint number SP1 to SP8 (1 to 8)

**Example**

Set the target setpoint number of unit 1, slot 5, loop 2 to 5.

OCtlSPN,L152,5

**Description**

• This command is valid when a PID Control Module is installed.
• When the module is not installed, E1 is returned.

**OCtlRL**

**Remote/Local Switching**

Switches between remote and local modes

**Syntax**

OCtlRL,p1,p2

- **p1** Loop number
- **p2** Remote/Local Switching
  0 Local
  1 Remote

**Example**

Set the input of unit 1, slot 5, loop 1 to remote.

OCtlRL,L151,1

**Description**

• This command is valid when a PID Control Module is installed.
• When the module is not installed, E1 is returned.
### OCtrlMO

**Sets the Manual Output Setpoint**  
Sets the manual output setpoint  

**Syntax**  
OCtrlMO,p1,p2,p3  
  p1 Loop number  
  p2 Type  
  0 Numeric input  
  1 Shutdown (tight shut function)  
  p3 Manual output setpoint (-50 to 1050 [-5.0% to 105.0%])  
  Within the output high and low limits  

**Example**  
Set the output value of unit 1, slot 5, loop 2 to 23.4%.  
OCtrlMO,L152,0,234  
Set the output value of unit 1, slot 5, loop 2 to shutdown.  
OCtrlMO,L152,1  

**Description**  
- This command is valid when a PID Control Module is installed.  
- You cannot set p3 if p2 is set to 1.  
- When the module is not installed, E1 is returned.

### OCtrlHOLD

**Hold Program Operation (/PG)**  

**Syntax**  
OCtrlHOLD,p1  
  p1 Holding of program operation  
  0 Release hold  
  1 Hold  

**Example**  
Release the holding of program operation.  
OCtrlHOLD,0  

**Description**  
- This command is valid when the program control function (/PG option) is in use.  
- This operation is applied to the pattern number specified with OCtrlPat.  
- When program operation is stopped, E1 is returned.

### OCtrlPAT

**Pattern Number Switching (/PG)**  
Switches the program pattern number  

**Syntax**  
OCtrlPAT,p1  
  p1 Pattern number switching (1 to 99)  

**Query**  
OCtrlPat?  

**Example**  
Switch to program pattern 2.  
OCtrlPAT,2  

**Description**  
- This command is valid when the program control function (/PG option) is in use.  
- You can set p1 regardless of whether the pattern file is available.

### OCtrlADV

**Advances Program Operation (/PG)**  

**Syntax**  
OCtrlAdv,p1  
  p1 Segment advance  
  1 Fixed (Advance)  

**Example**  
Request a segment advance during program operation.  
OCtrlAdv,1  

**Description**  
- This command is valid when the program control function (/PG option) is in use.  
- This operation is applied to the pattern number specified with OCtrlPat.  
- When program operation is stopped, E1 is returned.

### OCtrlSP

**Sets the Target Setpoint (/PG)**  
Sets the target setpoint (SP).  

**Syntax**  
OCtrlSP,p1,p2  
  p1 Loop number  
  p2 Target setpoint (PV range low limit to PV range high limit)  

**Example**  
Set the target setpoint of unit 1, slot 5, loop 2 to 2.5350.  
OCtrlSP,L152,2.5350  

**Description**  
- This command is valid when the program control function (/PG option) is in use.  
- Operation is possible only when the program operation is being held.  
- If p2 is set to a value outside the range or if the program hold is released (program is running), E1 is returned.  
- This operation is applied to the pattern number specified with OCtrlPat.  
- When program operation is stopped, E1 is returned.
**OCtrlTSP**

Sets the Final Target Setpoint (/PG)

Sets the final target setpoint (TSP)

**Syntax**

```
OCtrlTSP, p1, p2
```

- **p1** Loop number
- **p2** Final target setpoint (PV range low limit to PV range high limit)

**Example**

Set the final target setpoint of unit 1, slot 5, loop 2 to 2.5350.

```
OCtrlTSP, L152, 2.5350
```

**Description**

- This command is valid when the program control function (/PG option) is in use.
- Operation is possible only when the program operation is being held.
- If p2 is set to a value outside the range or if the program hold is released (program is running), E1 is returned.
- This operation is applied to the pattern number specified with OCtrlPat.
- When program operation is stopped, E1 is returned.

**OCtrlRTIME**

Sets the Segment Remaining Time (/PG)

Sets the remaining segment time

**Syntax**

```
OCtrlRTIME, p1, p2, p3
```

- **p1** Hour (0 to 99)
- **p2** Minute (0 to 59)
- **p3** Second (0 to 59)

**Example**

Set the remaining segment time to 11 hours 05 minutes 22 seconds.

```
OCtrlRTIME, 11, 05, 22
```

**Description**

- This command is valid when the program control function (/PG option) is in use.
- Operation is possible only when the program operation is being held.
- If the program hold is released (program is running), E1 is returned.
- This operation is applied to the pattern number specified with OCtrlPat.
- When program operation is stopped, E1 is returned.

**OCtrlStSeg**

Sets the Start Segment Number (/PG)

Sets the start segment of program operation

**Syntax**

```
OCtrlStSeg, p1, p2
```

- **p1** Pattern number (1 to 99)
- **p2** Segment number (1 to the number of segments in use)

**Query**

```
OCtrlStSeg[,p1]?
```

**Example**

Set the start segment of program pattern number 2 to 3.

```
OCtrlStSeg, 2, 3
```

**Description**

- This command is valid when the program control function (/PG option) is in use.
- E1 is returned in the following cases.
  - Program pattern with the number specified by p1 is running.
  - Program pattern file for the number specified by p1 cannot be found.
  - The segment number specified by p2 is greater than the number of segments in use.
- When the program is reset, the start number returns to segment number 1.

**OCtrlDlyTime**

Sets the Starting Time of Program Operation (/PG)

Sets the delay time between the start of program operation to when the program pattern operation actually starts.

**Syntax**

```
OCtrlDlyTime, p1, p2, p3, p4
```

- **p1** Pattern number (1 to 99)
- **p2** Hour (0 to 99)
- **p3** Minute (0 to 59)
- **p4** Second (0 to 59)

**Query**

```
OCtrlDlyTime[,p1]?
```

**Example**

Set the operation start delay time of program pattern number 2 to 5 minutes 55 seconds.

```
OCtrlDlyTime, 2, 0, 5, 55
```

**Description**

- This command is valid when the program control function (/PG option) is in use.
- E1 is returned in the following cases.
  - Program pattern with the number specified by p1 is running.
  - Program pattern file for the number specified by p1 cannot be found.
- When the program is reset, the delay time is reset to 00:00:00.
### OCtrlLoadPAT

**Loads a Pattern File (/PG)**  
Loads a program pattern file  
**Syntax**  
```
OCtrlLoadPAT,p1,p2,p3
```
  - `p1` File name (up to 80 characters)
  - `p2` Medium  
    - (GX/GP: SD, USB)  
    - (GM: SD)  
    - SD SD memory card  
    - USB USB flash memory
  - `p3` Load destination  
    - Pattern file number (1 to 99)

**Example**  
Load the program pattern file "PATTERN1" from the SD memory card to pattern file number 1.  
```
OCtrlLoadPAT,’PATTERN1’,SD,1
```

**Description**  
- This command is valid when the program control function (/PG option) is in use.
- If you omit parameter p2, the medium is set to the SD card.

### OCtrlSavePAT

**Saves a Pattern File (/PG)**  
Saves a program pattern file  
**Syntax**  
```
OCtrlSavePAT,p1,p2,p3
```
  - `p1` File name (up to 80 characters)
  - `p2` Medium  
    - (GX/GP: SD, USB)  
    - (GM: SD)  
    - SD SD memory card  
    - USB USB flash memory
  - `p3` Save source  
    - Pattern file number (1 to 99)

**Example**  
Save the program pattern file of pattern file number 1 to a file named "PATTERN1" in the SD memory card.  
```
OCtrlSavePAT,’PATTERN1’,SD,1
```

**Description**  
- This command is valid when the program control function (/PG option) is in use.
- If you omit parameter p2, the medium is set to the SD card.

### OCtrlLoadPATAll

**Collectively Loads Pattern Files (/PG)**  
Collectively Loads program pattern files  
**Syntax**  
```
OCtrlLoadPATAll,p1,p2
```
  - `p1` Folder name (up to 80 characters)
  - `p2` Medium  
    - (GX/GP: SD, USB)  
    - (GM: SD)  
    - SD SD memory card  
    - USB USB flash memory

**Example**  
Collectively load program pattern files from the “Pattern” folder in the SD memory card.  
```
OCtrlLoadPATAll,Pattern,SD
```

**Description**  
- This command is valid when the program control function (/PG option) is in use.
- If you omit parameter p2, the medium is set to the SD card.
- All pattern files in the folder are loaded.
- Only the pattern files with the following fixed file names in the specified folder are loaded.  
  File name: ProgPatXX.YYY  
  XX: Pattern number (01 to 99)

### OCtrlSavePATAll

**Collectively saves Pattern Files (/PG)**  
Collectively saves program pattern files  
**Syntax**  
```
OCtrlSavePATAll,p1,p2
```
  - `p1` Folder name (up to 80 characters)
  - `p2` Medium  
    - (GX/GP: SD, USB)  
    - (GM: SD)  
    - SD SD memory card  
    - USB USB flash memory

**Example**  
Collectively save program pattern files to the “Pattern” folder in the SD memory card.  
```
OCtrlSavePATAll,Pattern,SD
```

**Description**  
- This command is valid when the program control function (/PG option) is in use.
- If you omit parameter p2, the medium is set to the SD card.
- All pattern files are saved.
- Files are saved with fixed file names in the specified folder.  
  File name: ProgPatXX.YYY  
  XX: Pattern number (01 to 99)

### OCtrlDelPAT

**Deletes a Pattern File (/PG)**  
Deletes a program pattern file  
**Syntax**  
```
OCtrlDelPAT,p1
```
  - `p1` Pattern file number (1 to 99)

**Example**  
Delete pattern file number 99.  
```
OCtrlDelPAT,99
```

**Description**  
- This command is valid when the program control function (/PG option) is in use.
## 2.7 Communication Control Commands

### CCheckSum

Sets the Checksum
Sets the presence or absence of checksum.

**Syntax**

```
CCheckSum,p1
```

- `p1` Checksum usage
  - 0 Do not compute
  - 1 Compute

**Query**

Example: Enable the checksum.
```
CCheckSum,1
```

### CSFilter

Sets the Status Filter
Sets the filter used when outputting the recorder status.

**Syntax**

```
CSFilter,p1
```

- `p1` Filter values for status information numbers 1 to 4 (0.0.0.0 to 255.255.255.255)

**Query**

Example: Set the status filter value to 255.127.63.31.
```
CSFilter,255.127.63.31
```

**Description**

- The status filter is applied to each communication connection.

### CSFilterDB

Sets the status filter (expanded)
Sets the filter used when outputting the recorder status.

**Syntax**

```
CSFilterDB,p1,p2
```

- `p1` Filter values for status information numbers 1 to 4 (0.0.0.0 to 255.255.255.255)
- `p2` Filter values for status information numbers 5 to 8 (0.0.0.0 to 255.255.255.255)

**Query**

Example: Set the status filter value of status information 1 to 4 to 255.127.63.31 and that of status information 5 to 8 to 1.2.3.4.
```
CSFilterDB,255.127.63.31,1.2.3.4
```

**Description**

- The status filter is applied to each communication connection.
- `p2` can be omitted.

### CLogin

Log in over a Communication Path
Logs in over a communication path.

**Syntax**

```
CLogin,p1,p2
```

- `p1` User name
- `p2` password

**Example**

Log in using the user name "admin" and password "password."
```
CLogin,admin,password
```

**Description**

- For the characters that you can use for the password, see Appendix 1.
- If this command is executed while logged in, the user is once logged out and then logged back in.

When Using the Advanced Security Function (AS)

**Syntax**

```
CLogin,p1,p2,p3,p4,p5
```

- `p1` User name
- `p2` User ID
- `p3` Password
- `p4` The new password when the password has expired
- `p5` The new password when the password has expired for confirmation

**Example**

Log in using the user name "admin01" and password "password01,"
```
CLogin,admin01,,password01
```

**Description**

- If `p4` and `p5` are not specified, normal login will be used.
- Even if the password has not expired, you can enter a new password in `p4` in `p5` to change the password and log in.
- If `p4` and `p5` are not the same, an error will occur.
- You cannot change to the same password (if `p3` is the same as `p4` and `p5`, an error will occur).
- If the user ID is not used, `p2` is invalid.
- When using the password management, you cannot specify `p4` and `p5`.
- For the characters that you can use for the password, see Appendix 1.
- If this command is executed while logged in, the user is once logged out and then logged back in.

### CLogout

Log Out over a Communication Path
Logs out over a communication path.

**Syntax**

```
CLogout
```

**Example**

Logs out from the recorder.
```
CLogout
```
### CBTConnect

**Starts Bluetooth Communication (/C8) [GM]**

Starts Bluetooth communication.

**Syntax**

```plaintext
CBTConnect, pl
```

*`pl` Bluetooth password of the device you want to connect to*

**Example**

Connect to the device whose Bluetooth password is "PaSswoRD2."

```plaintext
CBTConnect, 'PaSswoRD2'
```

**Description**

- This command is valid only when a Bluetooth password request has been received via Bluetooth communication. If the command is invalid, error 352, "Unknown command," will occur.

### ESC O

**Opens an Instrument : RS-422/485 Command**

Starts communication with the recorder. ESC in ASCII code is 0x1B. For details, see Appendix 1.

**Syntax**

```plaintext
ESC O pl
```

*`pl` Instrument address (01 to 99)*

**Example**

Open the instrument at address 99.

```plaintext
ESC O 99
```

**Description**

- Specify the address of the instrument that you want to communicate with.
- You can only open one instrument at any given time.
- Use a capital "O."
- For this command, use CR+LF for the terminator.
- For the responses to this command, see page 2-120.

### ESC C

**Closes an Instrument : RS-422/485 Command**

Ends communication with the recorder. ESC in ASCII code is 0x1B. For details, see Appendix 1.

**Syntax**

```plaintext
ESC C pl
```

*`pl` Instrument address (01 to 99)*

**Example**

Close the instrument at address 77.

```plaintext
ESC C 77
```

**Description**

- This command closes the connection to the instrument you are communicating with.
- Use a capital "C."
- For this command, use CR+LF for the terminator.
- For the responses to this command, see page 2-120.
2.8 Instrument Information

Output Commands

_MFG

Outputs the Instrument Manufacturer

Syntax: _MFG

Description
• For the ASCII output format, see page 2-171.

_INF

Outputs the Instrument’s Product Name

Syntax: _INF

Description
• For the ASCII output format, see page 2-171.

_COD

Outputs the Instrument's Basic Specifications

Syntax: _COD

Description
• For the ASCII output format, see page 2-172.

_VER

Outputs the Instrument’s Firmware Version Information

Syntax: _VER

Description
• For the ASCII output format, see page 2-172.

_OPT

Outputs the Instrument’s Option Installation Information

Syntax: _OPT

Description
• For the ASCII output format, see page 2-173.

_TYP

Outputs the Instrument's Temperature Unit and Daylight Saving Time Installation Information

Outputs whether the instrument's Fahrenheit temperature unit and daylight saving time setting is enabled or disabled.

Syntax: _TYP

Description
• For the ASCII output format, see page 2-174.

_ERR

Outputs the Instrument’s Error Number Information

Outputs the error description that corresponds to the error number.

Syntax: _ERR,p1,p2,...

Write the details of the negative response returned from the recorder in p1, p2, etc.

Example
Output the error description when negative response "E1,10:1:2,500:2:5" is returned.

_DESCRIPTION

For the ASCII output format, see page 2-174.

_UNS or _UNR

Outputs the Instrument’s Unit Configuration Information

Outputs the instrument's unit configuration information.

Syntax: _UNS

Outputs the status that is recognized by the device.

_UNR

Outputs the installation status.

Description
• For the ASCII output format, see page 2-175.

_MDS or _MDR)

Outputs the Instrument’s Module Configuration Information

Outputs the instrument's module configuration information.

Syntax: _MDS

Outputs the status that is recognized by the device.

_MDR

Outputs the installation status.

Description
• For the ASCII output format, see page 2-176.
2.9 Responses to Commands

This section explains the responses that recorder returns in response to commands. There are three types of responses: affirmative response, negative response, and data output response.

2.9.1 Affirmative Response (For commands other than output request commands)

If the recorder successfully completes the processing of a received command that is not an output request command, it returns an affirmative response.

Syntax

```
E0
```

“CRLF” is the terminator that the recorder uses. “CRLF” will be used in the explanation of the syntax. In the response examples, “CRLF” will be omitted.

2.9.2 Negative Response

If a command syntax error, setting error, or other error occurs, the recorder returns a negative response.

Syntax

```
E1,p,p,•••,p
```

- **p**: Error number and the position of error occurrence
  The detailed format of p is indicated below. The recorder outputs the error number, the position of the command where the error occurred, and the position of the parameter where the error occurred, each separated by a colon.
  - **en**: Error number.
  - **cp**: A value indicating the command position where the error occurred. The position is numbered in order with the first command as 1. For a single command, the recorder outputs 1.
  - **pp**: A value indicating the parameter position where the error occurred. The position is numbered in order with the first parameter in each command as 1. For errors that pertain to the entire command (for example, error in the command name), the recorder outputs 0.

If errors occur in multiple parameters, the recorder outputs numbers separated by commas in ascending order.

**Response Example 1**

If error number 3 occurs in the second parameter of a single command, the recorder outputs:

```
E1,3:1:2
```

**Response Example 2**

If error number 1 occurs in the third parameter and error number 100 occurs in the fifth parameter of a single command, the recorder outputs:

```
E1,1:1:3,100:1:5
```
Response Example 3
In a string of two commands, if error number 10 occurs in the second parameter of the first command and error number 500 occurs in the fifth parameter of the second command, the recorder outputs:

E1,10:1:2,500:2:5

Error Messages
You can use the “instrument’s error number information output command” (_ERR) to output the error message that corresponds to an error number of a negative response.

2.9.3 Data Output Response
There are two types of data output: ASCII and binary.

ASCII Output
The responses to the following commands are in ASCII.
• Queries for operation commands and setting commands
• ASCII data output requests of output commands

Syntax

```
EACRLF
ASCII string data • • • • • • • • CRLF
ASCII string data • • • • • • • • CRLF
| ASCII string data • • • • • • • • CRLF
ENCRLF
```

The recorder adds a header (EA) in front of the ASCII string output data and a footer (EN) at the end. The recorder adds the two characters CRLF to the end of headers, footers, and ASCII string data.
Binary Output

The responses to output commands consisting of binary data output requests are in binary.

Format
The following figure shows the binary output format. The recorder adds a header to the front of binary output data and a checksum at the end. The request data is entered in the data block.

**EBCR LF**
The EBCR LF block stores ASCII code “E,” ASCII code “B,” followed by “CR” “LF.” This indicates that the output data is binary.

**Data length (32 bits, big endian)**
The data length block indicates the length of “flag + reserved area 1 + reserved area 2 + header sum + data block + data sum” in bytes.

**Flag (16 bits, big endian)**
The flag block indicates information of the entire data block.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Flag Value</th>
<th>Flag Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>Always zero</td>
<td>Not used</td>
</tr>
<tr>
<td>14</td>
<td>No</td>
<td>Yes Data sum inclusion</td>
</tr>
<tr>
<td>13</td>
<td>Always zero</td>
<td>Not used</td>
</tr>
<tr>
<td>:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Intermediate data</td>
<td>If the output data is continuous data, this flag indicates whether the last value in the data block is intermediate data or last data.</td>
</tr>
<tr>
<td>0</td>
<td>Last data</td>
<td></td>
</tr>
</tbody>
</table>

**Reserved area 1 (16 bits), reserved area 2 to (16 bits)**
Not used

**Header sum (16 bits, big endian)**
The header sum block indicates the sum of “data length + flag + reserved area 1 + reserved area 2.”
2.9 Responses to Commands

Data Block
The actual output data. The format varies depending on the output content. For details, see section 2.11, "Format of the Data Block of Binary Output."

Data sum (16 bits, big endian)
The data sum block indicates the sum of the data block. Use the CCheckSum command to specify whether to include data sum. By default, check sum is set to "No." Whether data sum is included is expressed by a flag in the header block. If the data sum block is not included, the area itself will not be included. For the check sum calculation method, see Appendix 5 Check Sum Calculation Method.

2.9.4 Output in Response to RS-422/485 Commands
The table below shows the responses to the ESC O command and ESC C command. ESC in ASCII code is 0x1B. For details, see Appendix 1 ASCII Character Codes.

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Meaning</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESC O_xxCRLF</td>
<td>Opens an instrument</td>
<td>• Response from the destination instrument ESC OxxCRLF</td>
</tr>
<tr>
<td>(_: Space)</td>
<td></td>
<td>• If there is no instrument at the address specified by the command*</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• No response</td>
</tr>
<tr>
<td>ESC C_xxCRLF</td>
<td>Closes an instrument</td>
<td>• Response from the destination instrument ESC CxxCRLF</td>
</tr>
<tr>
<td>(_: Space)</td>
<td></td>
<td>• If there is no instrument at the address specified by the command*</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• No response</td>
</tr>
</tbody>
</table>

* Some possible reasons why the condition "there is no instrument at the address specified by the command" occurs are command error, the address assigned to the instrument is different, the instrument is not turned on, and the instrument is not connected through serial interface.

- "xx" in the table represents the instrument address. You can specify any address within the range of 01 to 99 and within the addresses assigned to the communication target instruments.
- You can only open one instrument at any given time.
- When you open an instrument with the ESC O command, you can send commands to it.
- Use CR+LF for the terminator.
2.10 ASCII Output Format

This section explains the ASCII output format.
- In the following format descriptions, the terminator is denoted by “<crlf>.”
- One space (ASCII code: 0x20) is denoted by an underscore (_). Consecutive spaces are denoted by alternating underscores (_) and overscores (¯).
- An I/O channel is expressed as a four-digit number (e.g., 0102), a math channel is expressed as “A” followed by a three-digit number (e.g., A015), and a communication channel is expressed as “C” followed by a three-digit number (e.g., C120).

2.10.1 Most Recent Channel Data (FData)

The output in response to the command “FData,0” is shown below.

Syntax

```
EA<crlf>
DATE_yy/mo/dd<crlf>
TIME_hh:mm:ss.mmmt<crlf>
s_ccCcCa1a2a3a4u1u2u3u4ufddddddddE-pp<crlf>
...s_ccCcCa1a2a3a4u1u2u3u4ufddddddddE-pp<crlf>
EN<crlf>
```

- **yy/mo/dd** Data time (year, month, day)
  - **yy** Year (00 to 99)
  - **mo** Month (01 to 12)
  - **dd** Day (01 to 31)

- **hh:mm:ss.mmmt** Data time (hour, minute, second, millisecond)
  - **hh** Hour (00 to 23)
  - **mm** Minute (00 to 59)
  - **ss** Second (00 to 59)
  - **mmm** Millisecond (000 to 999)
  - A period is inserted between the minute and millisecond.

- **t** Reserved (space)

- **s** Data status
  - **N** Normal
  - **D** Differential input
  - **S** Skip
  - **O** Over
  - **E** Errors
  - **B** Burnout
  - **C** Communication channel error

- **cccc** Channel number (I/O channel, math channel, communication channel)

- **a1, a2, a3, and a4** Alarm status (level 1)
  - **H** High limit alarm
  - **L** Low limit alarm
  - **h** Difference high limit alarm
  - **l** Difference low limit alarm
  - **R** High limit on rate-of-change alarm
  - **r** Low limit on rate-of-change alarm
  - **T** Delay high limit alarm
  - **t** Delay low limit alarm

- **Space** No alarm

The alarm statuses of control alarms (when a PID control module is installed) are all set to zero.

- **uuuu** Unit (fixed to 10 characters. Output flush left. Unused character positions are filled with spaces.)
2.10 ASCII Output Format

The output in response to the command “FRelay,0” is shown below.

Syntax

When no expandable I/O is connected

EA\r\nM00:aaa...\r\nM01:aaa...\r\nM02:aaa...\r\nM03:aaa...\r\nM04:aaa...\r\nM05:aaa...\r\nM06:aaa...\r\nM07:aaa...\r\nM08:aaa...\r\nM09:aaa...\r\nEN\r

When an expandable I/O or sub unit is connected

Only the information of detected units will be output.

EA\r\nUnit:nnf\r\nM00:aaa...\r\nM01:aaa...\r\nM02:aaa...\r\nM03:aaa...\r\nM04:aaa...\r\nM05:aaa...\r\nM06:aaa...\r\nM07:aaa...\r\nM08:aaa...\r\nM09:aaa...\r\nUnit:nnf\r\nM00:aaa...\r\nM01:aaa...\r\nM02:aaa...\r\nM04:aaa...\r\nM05:aaa...\r\nM06:aaa...\r\nM07:aaa...\r\nM08:aaa...\r\nM09:aaa...\r\nUnit:nnf\r\nM00:aaa...\r\nM01:aaa...\r\nM02:aaa...\r\nM04:aaa...\r\nM05:aaa...\r\nM06:aaa...\r\nM07:aaa...\r\nM08:aaa...\r\nM09:aaa...\r

2.10.2 Most Recent (DO Channel) Status (FRelay)

The output in response to the command “FRelay,0” is shown below.

Mantissa (00000000 to 99999999; 8 digits)

For erroneous data (data status is E), the mantissa is 99999999.

If the data status is O (±over), the mantissa is 99999999 (+over) or –99999999 (–over).

If the data status is B (burnout), the mantissa is 99999999 (+burnout) or –99999999 (–burnout).

Exponent (00 to 05)

On channels set to Log scale (LG), pp is a two digit integer, and the sign before pp is + or -. If the data status is E, O, or B, this value will be +99, including the sign.
nn  Unit number
f   *  Main unit
     (Space)  Expandable I/O or sub unit
aaa... Outputs the relay (DO channel) status of module numbers 00 to 09.
If the module installed in the corresponding module number is not a DO module,
a hyphen is output.
If the module installed in the corresponding module number is a DO module, “1”
or “0” is output for the number of channels in the module in ascending order by
channel number.
“1” indicates relay (DO channel) ON state, and “0” indicates relay (DO channel)
OFF state.
If the DO terminal action (relay action) of a PID control module is set to “Contact
output within module,” the DO (relay) status is fixed to OFF.
2.10.3 Internal Switch Status (FRelay)
The output in response to the command “FRelay,1” is shown below.

Syntax

EA<crlf>
S001-010:aaaaaaaaaa<crlf>
S011-020:aaaaaaaaaa<crlf>
S021-030:aaaaaaaaaa<crlf>
S031-040:aaaaaaaaaa<crlf>
S041-050:aaaaaaaaaa<crlf>
S051-060:aaaaaaaaaa<crlf>
S061-070:aaaaaaaaaa<crlf>
S071-080:aaaaaaaaaa<crlf>
S081-090:aaaaaaaaaa<crlf>
S091-100:aaaaaaaaaa<crlf>
EN<crlf>

aaa…a The most recent internal switch status is output.
The internal switch status is output 10 channels per line over 10 lines.
“1” indicates that the internal switch is ON, and “0” indicates that the internal
switch is OFF.

2.10.4 Latest re-transmission output (AO channel) state (FTransStatAO)
The output in response to the command “FTransStatAO” is shown below.

Syntax

When no expandable I/O is connected

EA<crlf>
M00:aaa…<crlf>
M01:aaa…<crlf>
M02:aaa…<crlf>
M03:aaa…<crlf>
M04:aaa…<crlf>
M05:aaa…<crlf>
M06:aaa…<crlf>
M07:aaa…<crlf>
M08:aaa…<crlf>
M09:aaa…<crlf>
EN<crlf>

When an expandable I/O or sub unit is connected

Only the information of detected units will be output.

EA<crlf>
Unit:nnf
M00:aaa…<crlf>
M01:aaa…<crlf>
M02:aaa…<crlf>
M03:aaa…<crlf>
M04:aaa…<crlf>
M05:aaa…<crlf>
M06:aaa…<crlf>
M07:aaa…<crlf>
M08:aaa…<crlf>
M09:aaa…<crlf>
:
Unit:nnf
M00:aaa…<crlf>
M01:aaa…<crlf>
M02:aaa…<crlf>
M03:aaa…<crlf>
M04:aaa…<crlf>
M05:aaa…<crlf>
M06:aaa…<crlf>
Commands and Responses

M07:aaa...<crlf>
M08:aaa...<crlf>
M09:aaa...<crlf>
EN<crlf>

nn  Unit number
f    * Main unit
     (Space) Expandable I/O or sub unit
aaa... Outputs the re-transmission (AO channel) states of module numbers 00 to 09.
If the module installed in the corresponding module number is not an AO module, a hyphen is output.
If the module installed in the corresponding module number is a AO module, "1" or "0" is output for the number of channels in the module in ascending order by channel number.
"1" indicates re-transmission (AO channel) ON state, and "0" indicates re-transmission (AO channel) OFF state.
If the channel range setting is Skip or manual output, a hyphen is output.
2.10.5 Users Who Are Currently Logged In (FUser)

The output in response to the command “FUser,0” is shown below.

Syntax

```
EA\r\n"p_l_uuuuuuuuuuuuuuuuuu_abedefghijklmnopqrstuvwxyz\r\nEN\r```

```
p

Login mode
M  Via general communication
W  Via Web (HTTP server)
F  Via FTP server
S  RS-232, RS-422/485, USB communication, or Bluetooth
D  Via front panel

l

User level
A  Administrator
U  User
M  Monitor
(only when the advanced security function (/AS) enabled)

User name (fixed to 20 characters. Unused character positions are filled with spaces.)

User level
A  Administrator
U  User
M  Monitor
(only when the advanced security function (/AS) enabled)

Authority of user
F  Free
L  Lock

a through y represent actions. p through y are output only when the advanced security function (/AS) is enabled.
a  Memory
b  Math
c  Data save
d  Message
e  Batch
f  Alarm ACK
g  Communication
h  Touch operation
i  Time set
j  Setting operation
k  External media
m  System operation
n  Output operation
p  Calibration correction setting operation
q to y  Not used (Spaces)
The output in response to the command “FUser,2” is shown below.

**Syntax**

```
EA<crlf>
p_l_uuuuuuuuuuuuuuuu_abcd_efghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWX<crlf>
EN<crlf>
```

```
Login mode
M Via general communication
W Via Web (HTTP server)
F Via FTP server
S RS-232, RS-422/485, USB communication, or Bluetooth
D Via front panel

User level
A Administrator
U User
M Monitor (only when the advanced security function (/AS) enabled)

User name (fixed to 20 characters. Unused character positions are filled with spaces.)
```

```
Authority of user
F Free
L Lock
a Memory
b Math
c Data save
d Message
e Batch
f Alarm ACK
g Communication
h Touch operation
i Time set
j Setting operation
k External media
m System operation
n Output operation
p Calibration correction
setting operation
q to y Not used (Spaces)
```

```
Remote/Local operation
A
B Control operation
C Tuning operation
D Program operation
E SP operation
```

p through y are output only when the advanced security function (/AS) is enabled.
2.10 ASCII Output Format

2.10.6 All Users Who Are Currently Logged In (FUser)

The output in response to the command “FUser,1” is shown below.

Syntax

```
EA<crlf>
p_l_uuuuuuuuuuuuuuuuuuuuuuuuuuuuuu_abcd<crlf>
p_l_uuuuuuuuuuuuuuuuuuuu_abcd<crlf>
p_l_uuuuuuuuuuuuuuuuuuuu_abcd<crlf>
p_l_uuuuuuuuuuuuuuuuuuuu_abcd<crlf>
EN<crlf>
```

```
| p | Login mode
|---|------------------
| M | Via general communication
| W | Via Web (HTTP server)
| F | Via FTP server
| S | RS-232, RS-422/485, USB communication, or Bluetooth
| D | Via front panel

| l | User level
|---|------------------
| A | Administrator
| U | User
| M | Monitor

(only when the advanced security function (/AS) enabled)

```
| uuuuuuuuuuuuuuuuuuuu | User name (fixed to 20 characters. Unused character positions are filled with spaces.)
|----------------------|---------------------------------------------------
| abcde<crlf>fgijklmnpqrstuvwxy | Authority of user

| F | Free
| L | Lock
```

a through y represent actions. p through y are output only when the advanced security function (/AS) is enabled.

```
| a | Memory
| b | Math
| c | Data save
| d | Message
| e | Batch
| f | Alarm ACK
| g | Communication
| h | Touch operation
| i | Time set
| j | Setting operation
| k | External media
| m | System operation
| n | Output operation
| p | Calibration correction setting operation
| q to y | Not used (Spaces)
```
The output in response to the command “FUser,3" is shown below.

**Syntax**

```
EA<crlf>
 p_l_uuuuuuuuuuuuuuuuuu_abcddefghijkmnpqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ<br><crlf>
 p_l_uuuuuuuuuuuuuuuuuu_abcddefghijkmnpqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ<br><crlf>
 p_l_uuuuuuuuuuuuuuuuuu_abcddefghijkmnpqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ<br><crlf>
 p_l_uuuuuuuuuuuuuuuuuu_abcddefghijkmnpqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ<br><crlf>
EN<crlf>

P Login mode
   M  Via general communication
   W  Via Web (HTTP server)
   F  Via FTP server
   S  RS-232, RS-422/485, USB communication, or Bluetooth
   D  Via front panel

L User level
   A  Administrator
   U  User
   M  Monitor
      (only when the advanced security function (/AS) enabled)

User name (fixed to 20 characters. Unused character positions are filled with spaces.)

Authority of user
   F  Free
   L  Lock
   a  Memory
   b  Math
   c  Data save
   d  Message
   e  Batch
   f  Alarm ACK
   g  Communication
   h  Touch operation
   i  Time set
   j  Setting operation
   k  External media
   m  System operation
   n  Output operation
   p  Calibration correction setting operation
   q to y  Not used (Spaces)
   A  Remote/Local operation
   B  Control operation
   C  Tuning operation
   D  Program operation
   E  SP operation
   F to N  Not used (Spaces)
   p through y are output only when the advanced security function (/AS) is enabled.
```
2.10.7 Instrument Address (FAddr)

The output in response to the command “FAddr,IP” is shown below.

Syntax

```
EA<crlf>
IP_Address_¯_¯_:xxx.xxx.xxx.xxx<crlf>
Subnet_Mask_¯_¯_:xxx.xxx.xxx.xxx<crlf>
Default_Gateway_:xxx.xxx.xxx.xxx<crlf>
Primary_DNS_¯_¯_:xxx.xxx.xxx.xxx<crlf>
Secondary_DNS_=___:xxx.xxx.xxx.xxx<crlf>
Host_ _________:yyyyyyyyyyyyyyyyyyyyy..<crlf>
Domain_ _____ ___:zzzzzzzzzzzzzzzzzzzzz..<crlf>
EN<crlf>
```

<table>
<thead>
<tr>
<th>xxx</th>
<th>IP address number (0 to 255)</th>
</tr>
</thead>
<tbody>
<tr>
<td>yyy...</td>
<td>Host name (fixed to 64 characters. Unused character positions are filled with spaces.)</td>
</tr>
<tr>
<td>zzz...</td>
<td>Domain name (fixed to 64 characters. Unused character positions are filled with spaces.)</td>
</tr>
</tbody>
</table>
2.10.8 Recorder status (FStat)

The output in response to the command “FStat,0” is shown below.

**Syntax**

EA\(\text{crlf}\)
\(\text{aaa.bbb.ccc.ddd}\text{crlf}\)
EN\(\text{crlf}\)

The output in response to the command “FStat,1” is shown below.

**Syntax**

EA\(\text{crlf}\)
\(\text{aaa.bbb.ccc.ddd.eee.fff.ggg.hhh}\text{crlf}\)
EN\(\text{crlf}\)

<table>
<thead>
<tr>
<th>aaa</th>
<th>Status information 1 (see table below)</th>
</tr>
</thead>
<tbody>
<tr>
<td>bbb</td>
<td>Status information 2 (see table below)</td>
</tr>
<tr>
<td>ccc</td>
<td>Status information 3 (see table below)</td>
</tr>
<tr>
<td>ddd</td>
<td>Status information 4 (see table below)</td>
</tr>
<tr>
<td>eee</td>
<td>Status information 5 (see table below)</td>
</tr>
<tr>
<td>fff</td>
<td>Status information 6 (see table below)</td>
</tr>
<tr>
<td>ggg</td>
<td>Status information 7 (see table below)</td>
</tr>
<tr>
<td>hhh</td>
<td>Status information 8 (see table below)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Under control</td>
<td>Set to 1 while the recorder is under control.</td>
</tr>
<tr>
<td>1</td>
<td>Memory sampling</td>
<td>Set to 1 during recording.</td>
</tr>
<tr>
<td>2</td>
<td>Computing</td>
<td>Set to 1 while computation is in progress.</td>
</tr>
<tr>
<td>3</td>
<td>Alarm activated</td>
<td>Set to 1 when an alarm is activated.</td>
</tr>
<tr>
<td>4</td>
<td>Accessing medium</td>
<td>Set to 1 while the SD medium is being accessed.</td>
</tr>
<tr>
<td>5</td>
<td>E-mail started</td>
<td>Set to 1 while the e-mail transmission has been started.</td>
</tr>
<tr>
<td>6</td>
<td>Buzzer activated</td>
<td>Set to 1 when the buzzer is activated.</td>
</tr>
<tr>
<td>7</td>
<td>Re-transmitting</td>
<td>Set to 1 while re-transmitting.</td>
</tr>
</tbody>
</table>

**Status Information 2**

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>Memory end</td>
<td>Set to 1 when the free space in the external memory is low.</td>
</tr>
<tr>
<td>3</td>
<td>Touch operation login</td>
<td>Set to 1 when a user is logged in through touch operation.</td>
</tr>
<tr>
<td>4</td>
<td>User lock out present</td>
<td>Set to 1 when a user lock out occurs, and remains at 1 until user locked ACK is issued (only when the advanced security function (/AS) enabled).</td>
</tr>
<tr>
<td>5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>Measurement error</td>
<td>Set to 1 while measurement errors are detected on an AI module or when a burnout has occurred.</td>
</tr>
<tr>
<td>7</td>
<td>Communication error</td>
<td>Set to 1 when a Modbus master, Modbus client, WT communication, or SLMP communication error has occurred.</td>
</tr>
</tbody>
</table>

Status 3 and 4 are edge operations. They are cleared when read.
### Status Information 3

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Computation dropout</td>
<td>Set to 1 when computation cannot keep up.</td>
</tr>
<tr>
<td>1</td>
<td>Decimal and unit information setting</td>
<td>Set to 1 when the decimal or unit information is changed.</td>
</tr>
<tr>
<td>2</td>
<td>Command error</td>
<td>Set to 1 when there is a command syntax error.</td>
</tr>
<tr>
<td>3</td>
<td>Execution error</td>
<td>Set to 1 when there is a command execution error.</td>
</tr>
<tr>
<td>4</td>
<td>SNTP error at startup</td>
<td>Set to 1 when SNTP time synchronization fails at startup.</td>
</tr>
<tr>
<td>5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>7</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

### Status Information 4

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>Medium access complete</td>
<td>Set to 1 when a display, event, manual-sample, report, or screen-image data file is saved to the external storage medium. Set to 1 when settings have been successfully saved or loaded.</td>
</tr>
<tr>
<td>2</td>
<td>Report generation complete</td>
<td>Set to 1 when report generation is complete.</td>
</tr>
<tr>
<td>3</td>
<td>Timeout</td>
<td>Set to 1 when a timer expires.</td>
</tr>
<tr>
<td>4</td>
<td>Saving or loading complete</td>
<td>Set to 1 when the saving or loading of setting parameters, report template, scale image, custom display settings, and trusted certificate is complete.</td>
</tr>
<tr>
<td>5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>7</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

### Status Information 5

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Batch group #1 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>1</td>
<td>Batch group #2 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>2</td>
<td>Batch group #3 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>3</td>
<td>Batch group #4 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>4</td>
<td>Batch group #5 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>5</td>
<td>Batch group #6 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>6</td>
<td>Batch group #7 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>7</td>
<td>Batch group #8 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
</tbody>
</table>

### Status Information 6

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Batch group #9 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>1</td>
<td>Batch group #10 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>2</td>
<td>Batch group #11 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>3</td>
<td>Batch group #12 memory sampling</td>
<td>Set to 1 while recording is in progress.</td>
</tr>
<tr>
<td>4</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>7</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
### Status Information 7

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>4</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>7</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

### Status Information 8

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>4</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>7</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
2.10.9 Alarm Summary (FLog)

The output in response to the command “FLog,ALARM” is shown below.

Syntax

```
EA<crlf>
yyyy/mo/dd:hh:mm:ss.ttt_kkk_cccc_lss<crlf>
...
EN<crlf>
```

```
yyyy/mo/dd:hh:mm:ss.ttt Time of alarm occurrence

yyyy Year (1900 to 2099)
mo Month (01 to 12)
dd Day (01 to 31)
hh Hour (00 to 23)
mm Minute (00 to 59)
ss Second (00 to 59)
ttt Millisecond (000 to 999)

A period is inserted between the minute and millisecond.

kkk Alarm cause

OFF Alarm release
ON_ Alarm occurrence
ACK All channel alarm ACK, Individual alarm ACK
ALL All channel alarm OFF

ccccc Channel number (set to four spaces if the alarm cause is
“ACK” or “ALL”)

l Alarm level (1 to 4)

ss Alarm type

H_ High limit alarm
h_ Difference high limit alarm
L_ Low limit alarm
l_ Difference low limit alarm
R_ High limit on rate-of-change alarm
r_ Low limit on rate-of-change alarm
T_ Delay high limit alarm
t_ Delay low limit alarm
```

If the cause of alarm is “all channel alarm ACK” or “all channel alarm OFF,” the channel number, alarm level, and alarm type will be blank.

If the cause of alarm is “individual alarm ACK,” the alarm type will be blank.
2.10.10 Message Summary (FLog)

The output in response to the command “FLog,MSG” is shown below.

Syntax

```
EA<br>
yyyy/mo/dd_HH:mm:ss_YYYY/MO/DD:HH:MM:SS_t_mmm...m_www...w<br>
EN<br>
```

**yyyy/mo/dd_HH:mm:ss**  
Time when the message was written

- **yyyy**: Year (1900 to 2099)
- **mo**: Month (01 to 12)
- **dd**: Day (01 to 31)
- **HH**: Hour (00 to 23)
- **mm**: Minute (00 to 59)
- **ss**: Second (00 to 59)

**YYYY/MO/DD:HH:MM:SS**  
Data position where message was written

- **YYYY**: Year (1900 to 2099)
- **MO**: Month (01 to 12)
- **DD**: Day (01 to 31)
- **HH**: Hour (00 to 23)
- **MM**: Minute (00 to 59)
- **SS**: Second (00 to 59)

**t**  
Message type

- **N**: Normal message
- **H**: Freehand message

**mmm...m**  
Message (fixed to 48 characters. Unused character positions are filled with spaces.)
For freehand message, the string “(image)” is output.

**zzz**  
Operation property (3 characters)

- **KEY**: Touchscreen operation, key operation
- **COM**: Ethernet communication
- **SER**: Serial communication (RS-232, RS-422/485, USB communication, or Bluetooth)
- **ACT**: Event action
- **SYS**: System
- **EXT**: Operation from an external device (e.g. Modbus)
- **WEB**: Operation from web pages (GM, only when the advanced security function (/AS) is enabled)

**ggg...g**  
Target group (multiple groups are expressed using dot delimiters) (fixed to 16 characters. Unused character positions are filled with spaces.)

- **ALL**: All display groups
- **aa.bb.cc.dd...**: Multiple display groups

**uuu...u**  
User name (fixed to 20 characters. Unused character positions are filled with spaces.)
2.10.11 Event log (FLog)

The output in response to the command "FLog,EVENT" is shown below.

Syntax

EA<crlf>
yyyy/mo/dd:hh:mm:ss_zzz_sss_uuu_u<crlf>
...  
EN<crlf>

yyyy/mo/dd:hh:mm:ss  Time of event occurrence
  yyyy  Year (1900 to 2099)
  mo    Month (01 to 12)
  dd    Day (01 to 31)
  hh    Hour (00 to 23)
  mm    Minute (00 to 59)
  ss    Second (00 to 59)
zzz  Event cause
  KEY  Touchscreen operation, key operation
  REM  Remote
  COM  Ethernet communication
  SER  Serial communication (RS-232, RS-422/485, USB communication, or Bluetooth)
  ACT  Event action
  SYS  System
  EXT  Operation from an external device (e.g. Modbus)
  WEB  Operation from web pages (GM, only when the advanced security function (/AS) is enabled)

sss...s  Event string (fixed to 16 characters. Unused character positions are filled with spaces.)
See section "2.10.23 Detail Event Log Output (FEventLog) (/AS)".

uuu...u  User name (fixed to 20 characters. Unused character positions are filled with spaces.)

2.10.12 Error Log (FLog)

The output in response to the command "FLog,ERROR" is shown below.

Syntax

EA<crlf>
yyyy/mo/dd:hh:mm:ss_nnn_uuu_u<crlf>
...  
EN<crlf>

yyyy/mo/dd:hh:mm:ss  Time of error occurrence
  yyyy  Year (1900 to 2099)
  mo    Month (01 to 12)
  dd    Day (01 to 31)
  hh    Hour (00 to 23)
  mm    Minute (00 to 59)
  ss    Second (00 to 59)

nnn  Error code (001 to 999)

uuu...u  Error message (fixed to 80 characters. Unused character positions are filled with spaces.)
### 2.10.13 Address Setting Log (FLog)

The output in response to the command “FLog,DHCP” is shown below.

**Syntax**

```
EA<crlf>
yyyy/mo/dd:hh:mm:ss_kkk...k_mmm...m<crlf>
EN<crlf>
```

- `yyyy/mo/dd:hh:mm:ss` Time of occurrence
  - `yyyy` Year (1900 to 2099)
  - `mo` Month (01 to 12)
  - `dd` Day (01 to 31)
  - `hh` Hour (00 to 23)
  - `mm` Minute (00 to 59)
  - `ss` Second (00 to 59)
- `kkk...k` Type (fixed to 15 characters. Unused character positions are filled with spaces. See table below.)
- `mmm...m` Message (fixed to 20 characters. Unused character positions are filled with spaces. See table below.)

<table>
<thead>
<tr>
<th>Type</th>
<th>Message</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINK</td>
<td>ON</td>
<td>Ethernet connection detected</td>
</tr>
<tr>
<td></td>
<td>OFF</td>
<td>Ethernet disconnection detected</td>
</tr>
<tr>
<td>SET</td>
<td>Address (e.g., 10.0.122.3)</td>
<td>IP address set</td>
</tr>
<tr>
<td>DHCP</td>
<td>OFF</td>
<td>DHCP disabled</td>
</tr>
<tr>
<td></td>
<td>ON</td>
<td>DHCP enabled</td>
</tr>
<tr>
<td></td>
<td>RENEWING</td>
<td>Acquired IP address renewing</td>
</tr>
<tr>
<td></td>
<td>RELEASING</td>
<td>Acquired IP address releasing</td>
</tr>
<tr>
<td></td>
<td>REJECTING*</td>
<td>Acquired IP address rejecting*</td>
</tr>
<tr>
<td></td>
<td>RENEWED</td>
<td>IP address renewed</td>
</tr>
<tr>
<td></td>
<td>RELEASED</td>
<td>IP address released</td>
</tr>
<tr>
<td></td>
<td>EXTENDED</td>
<td>IP address extension application complete</td>
</tr>
<tr>
<td></td>
<td>ESEND</td>
<td>DHCP message transmission failed</td>
</tr>
<tr>
<td></td>
<td>ESERVER</td>
<td>DHCP server search failed</td>
</tr>
<tr>
<td></td>
<td>ESERVFAIL</td>
<td>DHCP server response failed (reception timeout)</td>
</tr>
<tr>
<td></td>
<td>ERENEWED</td>
<td>IP address renewal failed</td>
</tr>
<tr>
<td></td>
<td>RELEASED</td>
<td>IP address release failed</td>
</tr>
<tr>
<td></td>
<td>EEXTENDED</td>
<td>IP address extension application failed</td>
</tr>
<tr>
<td></td>
<td>EEXPIRED</td>
<td>IP address lease expiration</td>
</tr>
<tr>
<td>DNS</td>
<td>UPDATED</td>
<td>DNS host name registration complete</td>
</tr>
<tr>
<td></td>
<td>REMOVED</td>
<td>DNS host name removal complete</td>
</tr>
<tr>
<td></td>
<td>EFORMERR</td>
<td>DNS message syntax error</td>
</tr>
<tr>
<td></td>
<td>ESERVFAIL</td>
<td>DNS server processing error</td>
</tr>
<tr>
<td></td>
<td>ENXDOMAIN</td>
<td>DNS server query rejected (domain does not exist)</td>
</tr>
<tr>
<td></td>
<td>EREFUSED</td>
<td>DNS server query rejected (process not allowed)</td>
</tr>
<tr>
<td></td>
<td>EYXDOMAIN</td>
<td>DNS server query rejected (record exists)</td>
</tr>
<tr>
<td></td>
<td>EYXRESET</td>
<td>DNS server query rejected (record exists)</td>
</tr>
<tr>
<td></td>
<td>ENXRESET</td>
<td>DNS server query rejected (record does not exist)</td>
</tr>
<tr>
<td></td>
<td>ENOTAUTH</td>
<td>DNS server query rejected (not authenticated)</td>
</tr>
<tr>
<td></td>
<td>ENOTZONE</td>
<td>DNS server query rejected (query error)</td>
</tr>
<tr>
<td></td>
<td>ENOTIMP</td>
<td>DNS server query rejected (The command is not implemented.)</td>
</tr>
<tr>
<td></td>
<td>ENONAME</td>
<td>Tried to register a blank host name to the DNS server</td>
</tr>
</tbody>
</table>

* If the recorder cannot accept the IP address obtained from the DHCP server, the recorder will reject the address and immediately return a response to the DHCP server.
2.10.14 General Communication Log (FLog)

The output in response to the command “FLog,General” is shown below.

**Syntax**

```
EA<br>
 yyyy/mo/dd_hh:mm:ss_nn_uuu...u_fdmmm...m<br>
```

```
EN<br>
```

```
yyyy/mo/dd_hh:mm:ss Time of command Tx/Rx
```

```
yyyy Year (1900 to 2099)
m  Month (01 to 12)
dd Day (01 to 31)
hh Hour (00 to 23)
mm Minute (00 to 59)
ss Second (00 to 59)
```

```
nn Connection ID
```

```
s0 Serial (general)
s1 Bluetooth connection
s2 USB connection
e0 Ethernet connection #0 (general)
e1 Ethernet connection #1 (general)
e2 Ethernet connection #2 (general)
e3 Ethernet connection #3 (general)
```

```
uuu...u User name (fixed to 20 characters. Unused character positions are filled with spaces.)
```

```
f Multiple command flag
```

```
s  Space
```

```
f Multiple commands
```

```
d Tx/Rx
```

```
> Tx (command: connected instrument to recorder)
< Rx (Response: recorder to connected instrument)
```

```
mmm...m Message (fixed to 40 characters. Unused character positions are filled with spaces.)
```

The recorder normally outputs the data that has been transmitted or received as-is, but it sometimes outputs special messages. Special messages are shown below.

```
(output) Data output
(Over length) Command length too long
(timed out) Timeout
(disconnected) Disconnection (occurs when an Ethernet connection is disconnected)
```
2.10.15 Modbus Communication Log (FLog)

The output in response to the command “FLog,Modbus” is shown below.

Syntax

```
EA<crlf>
yyyyy/mo/dd:hh:mm:ss_c_xxxxxx_kkk...k_nnn_d<crlf>
... EN<crlf>
```

```
yyyyy/mo/dd:hh:mm:ss_Time of error occurrence
  yyyy   Year (1900 to 2099)
  mo     Month (01 to 12)
  dd     Day (01 to 31)
  hh     Hour (00 to 23)
  mm     Minute (00 to 59)
  ss     Second (00 to 59)
  c      Communication type
  M      Modbus master
  C      Modbus client

xxxxxx_Event that occurred (fixed to 6 characters)
  ACTIVE  Activated
  READY_  Command ready state
  CLOSE_  Disconnected
  HALT_   Command halted
  _ _ _   Other than those above

kkk...k_Details (fixed to 15 characters. Unused character positions are filled with spaces. See table below.)

nnn_Command number (0 to 999)

d_Command type
  R      Read
  W      Write
  O      Write immediately (write from the custom display)
  N      Miscellaneous
```

<table>
<thead>
<tr>
<th>Detail</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>SKIP</td>
<td>Command not set.</td>
</tr>
<tr>
<td>INVALID</td>
<td>Command cannot be executed.</td>
</tr>
<tr>
<td>WAITING</td>
<td>Server/slave communication recovery wait.</td>
</tr>
<tr>
<td>CLOSED</td>
<td>Server/slave connection closed.</td>
</tr>
<tr>
<td>RESOLVING</td>
<td>Server/slave connection being established (resolving address).</td>
</tr>
<tr>
<td>CONNECTING</td>
<td>Server/slave connection being established (requesting connection).</td>
</tr>
<tr>
<td>UNREACH</td>
<td>Server/slave connection failed (peer not found).</td>
</tr>
<tr>
<td>TIMEDOUT</td>
<td>Server/slave connection failed (timeout occurred).</td>
</tr>
<tr>
<td>BROKEN</td>
<td>Response message corrupt (CRC error).</td>
</tr>
<tr>
<td>ERR.FC</td>
<td>Response message was an illegal function message.</td>
</tr>
<tr>
<td>ERR_ADDR</td>
<td>Response message was an illegal data address message.</td>
</tr>
<tr>
<td>ERR.VALUE</td>
<td>Response message was an illegal data value message.</td>
</tr>
<tr>
<td>ERR DEVICE</td>
<td>Response message was a slave device failure message</td>
</tr>
<tr>
<td>ERR.ACK</td>
<td>Response message was an acknowledge message.</td>
</tr>
<tr>
<td>ERR_BUSY</td>
<td>Response message was a slave device busy message.</td>
</tr>
<tr>
<td>ERR.NEGATIVE</td>
<td>Response message was a negative acknowledge message.</td>
</tr>
<tr>
<td>ERR_GATE_PATH</td>
<td>Response message was a gateway path unavailable message.</td>
</tr>
<tr>
<td>ERR_GATE_TARGET</td>
<td>Response message was a gateway target device failed to respond message.</td>
</tr>
<tr>
<td>BAD_SLAVE</td>
<td>The slave address of the response message is invalid (does not match the command).</td>
</tr>
<tr>
<td>BAD_FC</td>
<td>The function code of the response message is invalid (does not match the command).</td>
</tr>
<tr>
<td>Detail</td>
<td>Meaning</td>
</tr>
<tr>
<td>-------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BAD_ADDR</td>
<td>The address of the response message is invalid (does not match the command).</td>
</tr>
<tr>
<td>BAD_NUM</td>
<td>The register of the response message is invalid (does not match the command).</td>
</tr>
<tr>
<td>BAD_CNT</td>
<td>The number of registers in the response message is invalid (does not match the command).</td>
</tr>
<tr>
<td>NO_DATA</td>
<td>Data has not yet been received once.</td>
</tr>
<tr>
<td>BAD_DATA</td>
<td>Data conversion of the response message failed.</td>
</tr>
<tr>
<td>VALID</td>
<td>Data is being acquired normally.</td>
</tr>
<tr>
<td>DROP_OUT</td>
<td>Communication dropout occurred due to the inability to keep up.</td>
</tr>
<tr>
<td>STALE</td>
<td>The response from the connected device is slow relative to the read cycle.</td>
</tr>
<tr>
<td>START</td>
<td>Modbus or communication settings were changed.</td>
</tr>
<tr>
<td>STOP</td>
<td>Modbus or communication settings were changed.</td>
</tr>
</tbody>
</table>

* "_" expresses an underscore.

### 2.10.16 FTP Client Log (FLog)

The output in response to the command “FLog,FTP” is shown below.

**Syntax**

```
EA<br>
yyyymo/dd_hh:mm:ss_xxxxxxxxx_k_fff...f<br>
EN<br>
```

YYYY/mo/dd_hh:mm:ss  **Time of error occurrence**

<table>
<thead>
<tr>
<th>YYYY</th>
<th>Year (1900 to 2099)</th>
</tr>
</thead>
<tbody>
<tr>
<td>mo</td>
<td>Month (01 to 12)</td>
</tr>
<tr>
<td>dd</td>
<td>Day (01 to 31)</td>
</tr>
<tr>
<td>hh</td>
<td>Hour (00 to 23)</td>
</tr>
<tr>
<td>mm</td>
<td>Minute (00 to 59)</td>
</tr>
<tr>
<td>ss</td>
<td>Second (00 to 59)</td>
</tr>
</tbody>
</table>

xxxxxxxxxx  **Detailed code (fixed to 9 characters)**

- TCPIP_:_  Internal processing error
- HOSTADDR_  IP address not set
- HOSTNAME_  Unable to resolve server host name
- UNREACH_  Unable to connect to server
- CONNECT_  Unable to connect to data port
- SEND_:_  Transmission to data port failed
- RECV_:_  Reception from data port failed
- CMDSEND_  Error in sending command to control port
- CMDRECV_  Error in receiving command from control port
- USER_:_  Invalid user name
- PASS_:_  Invalid password
- ACCT_:_  Internal processing error
- TIMEOUT_  Response timeout
- LINK_:_  Ethernet cable not connected
- FILE_:_  File access failed
- NOFD_:_  Internal processing error
- NOID_:_  Internal processing error
- PARAM_  Internal processing error
- CERT_:_  Certificate verification error
- SSL_:_  Encryption communication error

k  **Server type (P, S)**

fff...f  **File name (fixed to 51 characters including extension. Unused character positions are filled with spaces.)**
2.10.17 SNTP (Time Adjustment) Client Log (FLog)

The output in response to the command "FLog,SNTP" is shown below.

Syntax
EA\r\nyyyy/mo/dd hh:mm:ss nnn xxxxxxxxx\r\n...\nEN\r
yyyy/mo/dd hh:mm:ss Time of error occurrence
   yyyy Year (1900 to 2099)
   mo  Month (01 to 12)
   dd  Day (01 to 31)
   hh  Hour (00 to 23)
   mm  Minute (00 to 59)
   ss  Second (00 to 59)

nnn Error code
xxxxxxxxxx Detailed code (fixed to 9 characters)
   SUCCESS_  Success
   EOVER_    Adjustment limit exceeded
   EDORMANT_ Internal processing error
   EHOSTNAME Host name lookup failed
   ETCPIP_   Internal processing error
   ESEND_    Packet transmission failed
   ETIMEDOUT Response timeout occurred
   E BROKEN_ Response packet corrupt
   ERECV_    Reception error
   EINVALID_ Internal processing error
   ENOID_    Internal processing error
2.10.18 E-Mail Client Log (FLog)

The output in response to the command "FLog,MAIL" is shown below.

Syntax
EA\crlf>
yyyy/mo/dd_hh:mm:ss_ffffff_eeeeeeeeeeee_n_uuu...u\crlf>
EN\crlf>

yyyy/mo/dd_hh:mm:ss  Time of transmission
    yyyy  Year (1900 to 2099)
    mo    Month (01 to 12)
    dd    Day (01 to 31)
    hh    Hour (00 to 23)
    mm    Minute (00 to 59)
    ss    Second (00 to 59)

ffffff  Cause (fixed to 6 characters)
    ALARM_  Alarm mail
    TIMER_  Scheduled mail
    POWER_  Power-on, power failure recovery
    Memory  Low external storage memory
    ERROR_  Error notification
    REPORT  Report file
    TEST_   Test mail
    PASSWD  User lock out

eeeeeeeeeee  Detailed error code (fixed to 12 characters)
    HOSTADDR_  IP address not set
    HOSTNAME_  Unable to resolve server host name
    TIMEOUT_   Communication with server timed out
    LINK_      Ethernet cable not connected
    UNREACH_   Unable to connect to server
    HELO_      Server rejected greeting message
    MAILFROM_  Server rejected sender
    RCPTTO_    Server rejected recipient
    DATA_      Server rejected the data transmission command
    TCPIP_     Internal processing error
    INVAL_     Internal processing error
    SMTPAUTH_  SMTP AUTH authentication failed
    ANOTSUPPORT_  Unsupported authentication method
    POP3UNREACH_  Unable to connect to POP3 server
    POP3TIMEOUT_  POP3 server connection timed out
    POP3HOSTNAME_  Unable to resolve POP3 host name
    POP3AUTH_   POP3 server authentication failed
    CERT_      Certificate verification error
    SSL_       Encryption communication error

n  recipient
    1  Recipient 1
    2  Recipient 2
    +  Recipient 1+2

uuu...u  Recipient mail address (fixed to 30 characters. Unused character positions are filled with spaces.)
The user name section of the recipient mail address (the "XXXX" section of "XXXX@abc.def.ghi") is output.
2.10.19 Web Log (FLog)

The output in response to the command “FLog,WEB” is shown below.

**Syntax**

```
EA<crlf>
yyyy/mo/dd_hh:mm:ss_xxx.xxx.xxx_mmmmmmm_uuu...u_ccc_nnn...<crlf>
EN<crlf>
```

```
yyyy/mo/dd_hh:mm:ss  Time of error occurrence
  yyyy  Year (1900 to 2099)
  mo    Month (01 to 12)
  dd    Day (01 to 31)
  hh    Hour (00 to 23)
  mm    Minute (00 to 59)
  ss    Second (00 to 59)
```

```
xxx.xxx.xxx.xxx  Source IP address
```

```
mmmmmmmm  HTTP query method
  GET    GET method
  POST   POST method
```

```
uuu...u  Access destination URL (fixed to 24 characters. Unused character positions are filled with spaces.)
ccc    HTTP response code (fixed to 32 characters. Unused character positions are filled with spaces. See table below.)
nnn...  Error message (see table below)
```

<table>
<thead>
<tr>
<th>HTTP Response Code</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>Continue</td>
</tr>
<tr>
<td>101</td>
<td>Switching Protocols</td>
</tr>
<tr>
<td>201</td>
<td>Created</td>
</tr>
<tr>
<td>202</td>
<td>Accepted</td>
</tr>
<tr>
<td>203</td>
<td>Non-Authoritative Information</td>
</tr>
<tr>
<td>204</td>
<td>No Content</td>
</tr>
<tr>
<td>205</td>
<td>Reset Content</td>
</tr>
<tr>
<td>206</td>
<td>Partial Content</td>
</tr>
<tr>
<td>400</td>
<td>Bad Request</td>
</tr>
<tr>
<td>401</td>
<td>Unauthorized</td>
</tr>
<tr>
<td>403</td>
<td>Forbidden</td>
</tr>
<tr>
<td>404</td>
<td>Not Found</td>
</tr>
<tr>
<td>405</td>
<td>Method Not Allowed</td>
</tr>
<tr>
<td>406</td>
<td>Not Acceptable</td>
</tr>
<tr>
<td>407</td>
<td>Proxy Authentication Required</td>
</tr>
<tr>
<td>408</td>
<td>Request Time-out</td>
</tr>
<tr>
<td>409</td>
<td>Conflict</td>
</tr>
<tr>
<td>410</td>
<td>Gone</td>
</tr>
<tr>
<td>411</td>
<td>Length Required</td>
</tr>
<tr>
<td>412</td>
<td>Precondition Failed</td>
</tr>
<tr>
<td>413</td>
<td>Request Entity Too Large</td>
</tr>
<tr>
<td>414</td>
<td>Request-URI Too Large</td>
</tr>
<tr>
<td>415</td>
<td>Unsupported Media Type</td>
</tr>
<tr>
<td>500</td>
<td>Internal Server Error</td>
</tr>
<tr>
<td>501</td>
<td>Not Implemented</td>
</tr>
<tr>
<td>502</td>
<td>Bad Gateway</td>
</tr>
<tr>
<td>503</td>
<td>Server Unavailable</td>
</tr>
<tr>
<td>504</td>
<td>Gateway Time-out</td>
</tr>
<tr>
<td>505</td>
<td>HTTP Version Not Supported</td>
</tr>
</tbody>
</table>
2.10 ASCII Output Format

2.10.20 SLMP Log (FLog)

The output in response to the command "FLog,SLMP" is shown below.

Syntax

EA<crlf>
yyyy/mo/dd:ss_xxxxxx_kkk...k_nnn_d<crlf>...
EN<crlf>

yyyy/mo/dd:ss Time of command Tx/Rx
yyyy Year (1900 to 2099)
mo Month (01 to 12)
dd Day (01 to 31)
hh Hour (00 to 23)
mm Minute (00 to 59)
ss Second (00 to 59)

xxxxxxx Occurred event
ACTIVE Communication has been successfully established and normal data has been acquired.

READY_ Communication has been successfully established but normal data has not been acquired.

CLOSE_ TCP connection in progress.

HALT_ Communication has failed and has entered a communication recovery wait state

Other than those above

kkk...k Detail
nnn Command number (0 to 999)
d Command type
R Read
W Write
O Immediate write
N Others

<table>
<thead>
<tr>
<th>Detail</th>
<th>Group</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>START</td>
<td>Communication status</td>
<td>SLMP was started.</td>
</tr>
<tr>
<td>STOP</td>
<td>SLMP was stopped.</td>
<td></td>
</tr>
<tr>
<td>DROPOUT</td>
<td>Command could not be processed within the specified interval.</td>
<td></td>
</tr>
<tr>
<td>SKIP</td>
<td>Command problem</td>
<td>Command is not specified.</td>
</tr>
<tr>
<td>INVALID</td>
<td>Command cannot be executed.</td>
<td></td>
</tr>
<tr>
<td>WAITING</td>
<td>Communication problem</td>
<td>Server communication recovery wait</td>
</tr>
<tr>
<td>CLOSED</td>
<td>Server connection closed</td>
<td></td>
</tr>
<tr>
<td>RESOVING</td>
<td>Server connection is being established (resolving address).</td>
<td></td>
</tr>
<tr>
<td>CONNECTING</td>
<td>Server connection is being established (requesting connection).</td>
<td></td>
</tr>
<tr>
<td>UNREACH</td>
<td>Server connection failed (peer not found).</td>
<td></td>
</tr>
<tr>
<td>TIMEOUT</td>
<td>Server connection failed (timeout occurred).</td>
<td></td>
</tr>
<tr>
<td>ERROR</td>
<td>Response problem</td>
<td>System error occurred.</td>
</tr>
<tr>
<td>BROKEN</td>
<td>Response message is corrupt.</td>
<td></td>
</tr>
<tr>
<td>BAD_HEAD</td>
<td>Response message header error</td>
<td></td>
</tr>
<tr>
<td>BAD_LEN</td>
<td>Response message size error</td>
<td></td>
</tr>
<tr>
<td>BAD_DATA</td>
<td>Response message data error</td>
<td></td>
</tr>
<tr>
<td>ERROR:□□□□</td>
<td>Error response received (4-digit error number displayed in the squares)</td>
<td></td>
</tr>
<tr>
<td>VALID</td>
<td>Data condition</td>
<td>Data is being acquired normally.</td>
</tr>
<tr>
<td>STALE</td>
<td>Data is old.</td>
<td></td>
</tr>
</tbody>
</table>
2.10.21 Control Alarm Summary (FLog)

The output in response to the command “FLog,CALARM” is shown below.

Syntax

```
EA<crlf>
yyyy/mo/dd:hh:mm:ss.kkk_cccc_lssss<crlf>
EN<crlf>
```

```
yyyy/mo/dd:hh:mm:ss.ttt Time of control alarm occurrence
```

<table>
<thead>
<tr>
<th>yyyy</th>
<th>Year (1900 to 2099)</th>
</tr>
</thead>
<tbody>
<tr>
<td>mo</td>
<td>Month (01 to 12)</td>
</tr>
<tr>
<td>hh</td>
<td>Hour (00 to 23)</td>
</tr>
<tr>
<td>mm</td>
<td>Minute (00 to 59)</td>
</tr>
<tr>
<td>ss</td>
<td>Second (00 to 59)</td>
</tr>
<tr>
<td>ttt</td>
<td>Millisecond (000 to 999)</td>
</tr>
</tbody>
</table>

A period is inserted between the minute and millisecond.

<table>
<thead>
<tr>
<th>kkk</th>
<th>Alarm cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>Alarm release</td>
</tr>
<tr>
<td>ON_</td>
<td>Alarm occurrence</td>
</tr>
<tr>
<td>ACK</td>
<td>All channel alarm ACK, individual alarm ACK</td>
</tr>
<tr>
<td>ALL</td>
<td>All channel alarm OFF</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>cccc</th>
<th>Loop number (L001 to L692)</th>
</tr>
</thead>
<tbody>
<tr>
<td>l</td>
<td>Alarm level (1 to 4)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ssss</th>
<th>Alarm type</th>
</tr>
</thead>
<tbody>
<tr>
<td>PVH_</td>
<td>PV high limit</td>
</tr>
<tr>
<td>PVL_</td>
<td>PV low limit</td>
</tr>
<tr>
<td>SPH_</td>
<td>SP high limit</td>
</tr>
<tr>
<td>SPL_</td>
<td>SP low limit</td>
</tr>
<tr>
<td>DVH_</td>
<td>Deviation high limit</td>
</tr>
<tr>
<td>DVL_</td>
<td>Deviation low limit</td>
</tr>
<tr>
<td>DVO_</td>
<td>Deviation H/L limits</td>
</tr>
<tr>
<td>DVI_</td>
<td>Deviation within H/L limits</td>
</tr>
<tr>
<td>OTH_</td>
<td>Control output high limit</td>
</tr>
<tr>
<td>OTL_</td>
<td>Control output low limit</td>
</tr>
<tr>
<td>PVR_</td>
<td>PV velocity</td>
</tr>
</tbody>
</table>

If the cause of alarm is “all channel alarm ACK” or “all channel alarm OFF,” the loop number, alarm level, and alarm type will be blank.

If the cause of alarm is “individual alarm ACK,” the loop number and alarm level will be output.
2.10  ASCII Output Format

2.10.22 Control Summary (FLog)

The output in response to the command "FLog,CTRL" is shown below.

Syntax

EA\r\nyyyy/mo/dd_hh:mm:ss_aaa...a_sss...s\r\nEN\r

yyyy/mo/dd_hh:mm:ss  Time of control occurrence
  yyyy  Year (1900 to 2099)
  mo   Month (01 to 12)
  dd   Day (01 to 31)
  hh   Hour (00 to 23)
  mm   Minute (00 to 59)
  ss   Second (00 to 59)

aaa...a  Name (up to 13 characters)

Pattern number_loop number_type

Pattern number  Up to 2 characters
Loop number    Up to 4 characters
Type           Up to 5 characters

If the maximum number of characters is not used, unused characters become spaces.

sss...s  Status (up to 12 characters)

<table>
<thead>
<tr>
<th>Name</th>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loop number</td>
<td>LOCAL</td>
<td>Control operation changed to local.</td>
</tr>
<tr>
<td></td>
<td>REMOTE</td>
<td>Control operation changed to remote.</td>
</tr>
<tr>
<td></td>
<td>PROGRAM</td>
<td>Control operation changed to program.</td>
</tr>
<tr>
<td></td>
<td>AUTO</td>
<td>Control operation changed to auto.</td>
</tr>
<tr>
<td></td>
<td>MANUAL</td>
<td>Control operation changed to manual.</td>
</tr>
<tr>
<td></td>
<td>CASCADE</td>
<td>Control operation changed to cascade.</td>
</tr>
<tr>
<td></td>
<td>RUN</td>
<td>Run control operation</td>
</tr>
<tr>
<td></td>
<td>STOP</td>
<td>Stop control operation</td>
</tr>
<tr>
<td></td>
<td>AT**_ON</td>
<td>Auto-tuning start **: PID number (value) or &quot;R&quot;</td>
</tr>
<tr>
<td></td>
<td>AT OFF</td>
<td>Auto-tuning stop</td>
</tr>
<tr>
<td>Pattern number</td>
<td>PROG RUN</td>
<td>Program operation run</td>
</tr>
<tr>
<td></td>
<td>PROG RESET</td>
<td>Program operation stop</td>
</tr>
<tr>
<td></td>
<td>HOLD ON</td>
<td>Program operation hold start</td>
</tr>
<tr>
<td></td>
<td>HOLD OFF</td>
<td>Program operation hold release</td>
</tr>
<tr>
<td></td>
<td>WAIT ON</td>
<td>Program operation wait start</td>
</tr>
<tr>
<td></td>
<td>WAIT OFF</td>
<td>Program operation wait release</td>
</tr>
<tr>
<td></td>
<td>ADVANCE</td>
<td>Program operation advance execution</td>
</tr>
<tr>
<td>Pattern number_loop number_type</td>
<td>PVE**_ON</td>
<td>PV event occurrence in program operation **: Event number (value)</td>
</tr>
<tr>
<td></td>
<td>PVE**_OFF</td>
<td>PV event release in program operation **: Event number (value)</td>
</tr>
<tr>
<td>Pattern number</td>
<td>TME**_ON</td>
<td>Time event occurrence in program operation **: Event number (value)</td>
</tr>
<tr>
<td></td>
<td>TME**_OFF</td>
<td>Time event release in program operation **: Event number (value)</td>
</tr>
</tbody>
</table>
### 2.10 ASCII Output Format

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PVH</td>
<td>PV high limit</td>
</tr>
<tr>
<td>PVL</td>
<td>PV low limit</td>
</tr>
<tr>
<td>SPH</td>
<td>SP high limit</td>
</tr>
<tr>
<td>SPL</td>
<td>SP low limit</td>
</tr>
<tr>
<td>DVH</td>
<td>Deviation high limit</td>
</tr>
<tr>
<td>DVL</td>
<td>Deviation low limit</td>
</tr>
<tr>
<td>DVO</td>
<td>Deviation H/L limits</td>
</tr>
<tr>
<td>DVI</td>
<td>Deviation within H/L limits</td>
</tr>
<tr>
<td>OTH</td>
<td>Control output high limit</td>
</tr>
<tr>
<td>OTL</td>
<td>Control output low limit</td>
</tr>
<tr>
<td>PVR</td>
<td>PV velocity</td>
</tr>
</tbody>
</table>
2.10.23 Detail Event Log Output (FEventLog) (/AS)

The output in response to the command "FEventLog" is shown below. Output is possible when the advanced security function (/AS) is enabled. Output from Web operation is possible only when the GM’s advanced security function (/AS) is enabled.

Syntax

EA<crlf>
yyyymo/dd_hh:mm:ss_zzz_sss...s_uuu...u_ddd...<crlf>
EN<crlf>

yyyymo/dd_hh:mm:ss Time of event occurrence

- yyyy: Year (1900 to 2099)
- mo: Month (01 to 12)
- dd: Day (01 to 31)
- hh: Hour (00 to 23)
- mm: Minute (00 to 59)
- ss: Second (00 to 59)

zzz Event cause

- KEY: Touchscreen operation, key operation
- REM: Remote
- COM: Ethernet communication
- SER: Serial communication (RS-232, RS-422/485, USB communication, or Bluetooth)
- ACT: Event action
- SYS: System
- EXT: Operation from an external device (e.g. Modbus)
- WEB: Operation from web pages (GM, only when the advanced security function (/AS) is enabled)

sss...s Event string (fixed to 16 characters. Unused character positions are filled with spaces. See the table below.)

uuu...u User name (fixed to 20 characters. Unused character positions are filled with spaces.)

ddd... Detailed information (see table below)

Event string, detailed information

Operations that are marked with an asterisk will be logged regardless of whether the advanced security function is enabled or disabled.

All other operations are logged only when the advanced security function (/AS) is enabled.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Event string</th>
<th>### information and detailed information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error, system notification</td>
<td>Error###</td>
<td>Error number (output in the event string)</td>
</tr>
<tr>
<td>Expiration</td>
<td>Expiring##</td>
<td>Schedule number</td>
</tr>
</tbody>
</table>

Example:

- Error occurrence: 'Check Data'

Calibration operation

A/D calibration  ExecA/DCal

- Unit:uu,Slot:ss
- uu: Unit
- ss: Slot

Module calibration  CalModule

- Unit:uu,Slot:ss,m••
- uu: Unit
- ss: Slot
- m: Module name

Example:

- Unit:00,Slot:01,GX90YA-04-C1

Login
### Operation Event string ### information and detailed information

<table>
<thead>
<tr>
<th>Operation</th>
<th>Event string</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power off*</td>
<td>POWER OFF</td>
<td>—</td>
</tr>
<tr>
<td>Power on*</td>
<td>POWER ON</td>
<td>—</td>
</tr>
<tr>
<td>Login*</td>
<td>LOGIN</td>
<td>—</td>
</tr>
<tr>
<td>Logout*</td>
<td>LOGOUT</td>
<td>—</td>
</tr>
<tr>
<td>User lock out*</td>
<td>UserLocked</td>
<td>User: UUU</td>
</tr>
</tbody>
</table>

### Control

<table>
<thead>
<tr>
<th>Mode change</th>
<th>ChgMode</th>
<th>ss***</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time change*</td>
<td>TIME CHANGE</td>
<td>—</td>
</tr>
<tr>
<td>New time*</td>
<td>NEW TIME</td>
<td>—</td>
</tr>
<tr>
<td>Start time adjustment*</td>
<td>TIME ADJ START</td>
<td>amm:ss:xxx.yyy</td>
</tr>
</tbody>
</table>

- **Difference**
  - a: Sign (- for lag, + for lead)
  - mm: Minute
  - ss: Second
  - xxx: Millisecond
  - yyy: Microsecond

**Example:**
+00:01:000.000

<table>
<thead>
<tr>
<th>Stop time adjustment*</th>
<th>TIME ADJ END</th>
<th>—</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNTP time change*</td>
<td>SNTP ADJUST</td>
<td>—</td>
</tr>
<tr>
<td>DST start*</td>
<td>DST START</td>
<td>—</td>
</tr>
<tr>
<td>DST end*</td>
<td>DST END</td>
<td>—</td>
</tr>
<tr>
<td>Password change</td>
<td>ChgPasswd</td>
<td>User: UUU</td>
</tr>
</tbody>
</table>

| User locked ACK     | UserLockedACK  | — |
| Alarm ACK           | AlarmACK      | Channel: cc***, Level: li*** |
| Message writing     | Message#####   | Front half: Message number (output in the event string) |

- Normal messages: 001 to 100
- Free messages: F01 to F10
- Freehand message: Hnd

- Latter half: Batch group number (output in the event string)
  - When multi batch is disabled: (space)
  - When multi batch is enabled: -01 to -12

**Example:**
"Message001", "MessageF01-12"

**<Detailed information>**

**Data Time:**

- Data Time: tt*** Data timestamp (only for add messages. Not output for other messages.)
- The format is the same as the time section of the FLog command output.

**Example:**
Data Time: 2012/03/13 10:25:28

**Recording start**

- Record Start###
  - When multi batch is disabled: (space)
  - When multi batch is enabled: -01 to -12 Batch group number

**Recording stop**

- Record Stop###
  - When multi batch is disabled: (space)
  - When multi batch is enabled: -01 to -12 Batch group number

**Manual sample**

- ManualSample —
### 2.10 ASCII Output Format

<table>
<thead>
<tr>
<th>Operation</th>
<th>Event string</th>
<th>### information and detailed information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Math start</td>
<td>MathStart</td>
<td></td>
</tr>
<tr>
<td>Math stop</td>
<td>MathStop</td>
<td></td>
</tr>
<tr>
<td>Math reset</td>
<td>MathRST###</td>
<td>### When multi batch is disabled: (space)</td>
</tr>
</tbody>
</table>
|                         |              | | When multi batch is enabled: (
|                         |              | space) Resetting of all math channels |
|                         |              | | -01 to-12 Resetting of math channels |
|                         |              | belonging to the specified batch group |
| Acknowledge math        | MathACK      |                                          |
| dropout                 |              |                                          |
| Mail start              | MailStart    |                                          |
| Mail stop               | MailStop     |                                          |
| Modbus manual recovery  | RefModbus    | ss*** Type [Client, Master]              |
| Manually SLMP           | RefSLMP      |                                          |
| communication recovery  |              |                                          |
| Display data saved      | SaveDisp### | ### When multi batch is disabled: (space) |
|                         |              | When multi batch is enabled: -01 to -12 Batch group number |
| Event data saved        | SaveEven### | ### When multi batch is disabled: (space) |
|                         |              | When multi batch is enabled: -01 to -12 Batch group number |
| Manual data saved       | ManualSave   | ss*** Data type [Data, Report, ManualSample, AlarmSummary] |
|                         |              | [All] for all data. [Cancel] if canceled. |
| Snapshot                | Snapshot     |                                          |
| Set batch number        | SetBatchNo### | ### When multi batch is disabled: (space) |
|                         |              | When multi batch is enabled: -01 to -12 Batch group number |
| Set lot number          | SetLotNo### | ### When multi batch is disabled: (space) |
|                         |              | When multi batch is enabled: -01 to -12 Batch group number |
| Batch text Field        | SetTextField### | ### When multi batch is disabled: (space) |
| setting                 |              | When multi batch is enabled: -01 to -12 Batch group number |
| Display update rate     | ChgRate      | ss*** Trend interval string Example: 1min/div |
| change                  |              |                                          |

Example: 1min/div
### 2.10 ASCII Output Format

<table>
<thead>
<tr>
<th>Operation</th>
<th>Event string</th>
<th>### information and detailed information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timer reset</td>
<td>TimerRST</td>
<td>Timer:ttt,ttt,ttt•••</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ttt List of timer numbers that</td>
</tr>
<tr>
<td></td>
<td></td>
<td>were reset (ALL for all timers)</td>
</tr>
<tr>
<td>Match time timer reset</td>
<td>MTimerRST</td>
<td>Timer:ttt,ttt,ttt•••</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ttt List of timer numbers that</td>
</tr>
<tr>
<td></td>
<td></td>
<td>were reset (ALL for all timers)</td>
</tr>
<tr>
<td>Communication channel write (screen operation only)</td>
<td>WriteComm</td>
<td>kk•••,CCCC=dd•••</td>
</tr>
<tr>
<td></td>
<td></td>
<td>kk••• Write type [Internal,External]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>C Communication channel</td>
</tr>
<tr>
<td></td>
<td></td>
<td>dd••• Value</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Internal,C001=1.234</td>
</tr>
<tr>
<td>DO channel write (for manual operation) (screen operation/ general communication command only)</td>
<td>WriteDO</td>
<td>CCCC=dd•••</td>
</tr>
<tr>
<td></td>
<td></td>
<td>C DO channel</td>
</tr>
<tr>
<td></td>
<td></td>
<td>dd••• Value</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0901=OFF</td>
</tr>
<tr>
<td>SW channel write (for manual operation) (screen operation/general communication command only)</td>
<td>WriteSW</td>
<td>CCCC=dd•••</td>
</tr>
<tr>
<td></td>
<td></td>
<td>C Internal switch</td>
</tr>
<tr>
<td></td>
<td></td>
<td>dd••• Value</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>S001=ON</td>
</tr>
<tr>
<td>Settings saved</td>
<td>Save#########</td>
<td>### Save type (output in the event string)</td>
</tr>
<tr>
<td></td>
<td>Report</td>
<td>Report</td>
</tr>
<tr>
<td></td>
<td>Scale</td>
<td>Scale image</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When multi batch is enabled, a hyphen followed by the specified batch group number is added.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SaveScale-02</td>
</tr>
<tr>
<td></td>
<td>Custom</td>
<td>Custom display</td>
</tr>
<tr>
<td></td>
<td>Parameter</td>
<td>Setting parameter</td>
</tr>
<tr>
<td></td>
<td>Cert</td>
<td>Certificate</td>
</tr>
<tr>
<td></td>
<td>All</td>
<td>All settings</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Settings loaded</td>
<td>Load##########</td>
<td>### Load type (output in the event string)</td>
</tr>
<tr>
<td></td>
<td>Report</td>
<td>Report</td>
</tr>
<tr>
<td></td>
<td>Scale</td>
<td>Scale image</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When multi batch is enabled, a hyphen followed by the specified batch group number is added.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LoadScale-02</td>
</tr>
<tr>
<td></td>
<td>Custom</td>
<td>Custom display</td>
</tr>
</tbody>
</table>

**<Detailed information>**

---- When ### = Report ----

cc•••,rr•••

cc••• Report format [EXCEL, PDF]

rr••• Report type [Hour, Day, Week, Month, Hour+Day, Day+Week, Day+Month, Batch, Custom]

---- When ### = Scale ----

Group:gg

gg Group number

---- When ### = Custom ----

No:nn•••

nn••• Display number (ALL for all custom display screen)
### Operation and event string

<table>
<thead>
<tr>
<th>Operation</th>
<th>Event string</th>
<th>### information and detailed information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a key</td>
<td>GeneKey###</td>
<td>Action (output in the event string)</td>
</tr>
<tr>
<td>Installation of certificate</td>
<td>InstallServCert</td>
<td>Start Start Complete Cancel</td>
</tr>
<tr>
<td>Certificate creation</td>
<td>CreateCert</td>
<td>—</td>
</tr>
<tr>
<td>Touch screen calibration reset</td>
<td>ExecTouchCal</td>
<td>—</td>
</tr>
<tr>
<td>Initialize</td>
<td>Initialize</td>
<td>ss###</td>
</tr>
<tr>
<td>Sign in</td>
<td>Sign In</td>
<td>l, ss***</td>
</tr>
<tr>
<td>Multi batch change</td>
<td>ChgMultiBatch</td>
<td>(s,num)=(b1,b2)-&gt;(a1,a2)</td>
</tr>
</tbody>
</table>

#### Detailed information

--- When ### = Report ----

- **cc*** Report format [EXCEL, PDF]
- **rr*** Report type [Hour, Day, Week, Month, Hour+Day, Day+Week, Day+Month, Batch, Custom]

--- When ### = Scale ----

- **gg** Group number

--- When ### = Custom ----

- **nn*** Display number (ALL for all custom display screen)

--- When ### = Parameter ----

- **ss*** Loaded settings [Security, IP, Other, All, w/o-SERVER, w/o-CALIB, w/o-INSTRU]
  * "w/o-" indicates that the setting is excluded.

--- When ### = Action (output in the event string) ---

- **Start**
- **Done**
- **Cancel**

--- When ### = Certificate type: Main/Middle [Main, Chained] ---

- **kk*** Purpose: SSL, PDF [COM, PDF]

--- When ### = Initialize type [Security, Other, Data, w/o-IP, w/o-SERVER, w/o-CALIB, w/o-INSTRU] ---

- **s** On/Off (before and after change) [ON, OFF]
- **num** Number of multi batches (before and after change)

--- When ### = Level (1 to 3) ---

- **l**
- **ss*** File name

--- When ### = Before change ---

- **b1,b2**
- **a1,a2**

--- When ### = After change ---

- **The following settings (those that have been changed among two settings)**

--- When ### = On/Off (before and after change) [ON, OFF] ---

- **num** Number of multi batches (before and after change)
<table>
<thead>
<tr>
<th>Operation</th>
<th>Event string</th>
<th>### information and detailed information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lock the keys</td>
<td>Keylock ON</td>
<td>—</td>
</tr>
<tr>
<td>Release the key lock</td>
<td>Keylock OFF</td>
<td>—</td>
</tr>
<tr>
<td>Turn on the Bluetooth function</td>
<td>Bluetooth ON</td>
<td>—</td>
</tr>
<tr>
<td>Turn off the Bluetooth function</td>
<td>Bluetooth OFF</td>
<td>—</td>
</tr>
<tr>
<td>Clear the Bluetooth connection list</td>
<td>BTLListClear</td>
<td>—</td>
</tr>
<tr>
<td>Fixed IP address mode</td>
<td>FixedIPMode</td>
<td>—</td>
</tr>
<tr>
<td>Saving of unsaved data</td>
<td>DiffAutoSave</td>
<td>—</td>
</tr>
<tr>
<td>AO re-transmission operation</td>
<td>AOTrans CCCC=d•••</td>
<td>AO channel (for individual channel operation), ALL (for collective channel operation)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>d Example: 0901=OFF</td>
</tr>
<tr>
<td>AO manual output operation (screen operation only)</td>
<td>AOManual CCCC=d•••</td>
<td>AO channel</td>
</tr>
<tr>
<td></td>
<td></td>
<td>d Example: 0001=1234</td>
</tr>
<tr>
<td>Individual initialization</td>
<td>IndivInit sss···</td>
<td>Initialize type [RecordCh,DisplayGroup...] (List of initialized settings)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example: RecordCh,DisplayGroup</td>
</tr>
<tr>
<td>Module</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reconfiguration Module disconnection</td>
<td>ConfigModule</td>
<td>Unit:uu,Slot:ss,mm•••,ii•••,vv•••</td>
</tr>
<tr>
<td>Module disconnection</td>
<td>RemoveModule</td>
<td></td>
</tr>
<tr>
<td>Modules installed</td>
<td>AttachModule Unit:uu,Slot:ss,mm•••,ii•••,vv•••</td>
<td></td>
</tr>
<tr>
<td>Module information</td>
<td>InfoModule Unit:uu,Slot:ss,dd•••,UU•••</td>
<td></td>
</tr>
</tbody>
</table>

Example:
(s.num)=(OFF,3)->(ON,12)
When multi batch settings are loaded, if the settings have not changed, the details are not output.
### Operation Event string ### information and detailed information

<table>
<thead>
<tr>
<th>Operation</th>
<th>Event string</th>
<th>### information and detailed information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module activation</td>
<td>ApplyModule</td>
<td>Unit:uu,Slot:ss,mm***,ii***,vv***</td>
</tr>
<tr>
<td>Module update</td>
<td>UpdateModule</td>
<td>Unit:uu,Slot:ss,mm***,ii***,vv***</td>
</tr>
</tbody>
</table>

#### Setting changes during recording

**Alarm setting change SetAlarm**

```markdown
ccc:(s,typ,val,hys,l,Otyp,Ono)=(b1,b2,b3,b4,b5,b6,b7)->(a1,a2,a3,a4,a5,a6,a7)
```

- **c**: Channel
- **l**: Level
- **b1,•••,b7**: Before change
- **a1,•••,a7**: After change

The following settings (those that have been changed among the following seven settings):

- **s**: On/Off [ON, OFF]
- **typ**: Type [H,L,R,r,h,l,T,t]
- **val**: Alarm value
- **hys**: Hysteresis
- **l**: Logging [ON, OFF]
- **Otyp**: Output type [OFF,DO,SW]
- **Ono**: Output number

**Example 1:**

0001:1:(s,typ,val,hys,l,Otyp,Ono)=(off,TH,-2.000,0.0005,DO,0001)->(off,TL,off,-2.000,0.0005,SW,001)

**Example 2:**

0002:2:(val)=(-2.000)->(-1.000)

**Alarm delay setting change SetAlmDelay**

```markdown
ccc:(hour,min,sec)=(b1,b2,b3)->(a1,a2,a3)
```

- **ccc**: Channel
- **b1,b2,b3**: Before change
- **a1,a2,a3**: After change

The following settings (those that have been changed among the following three settings):

- **hour**: Delay hour
- **min**: Delay minute
- **sec**: Delay second

**Example:**

A100:(hour,min,sec)=(00,00,00)->(01,02,03)

**Calibration correction/set point change SetCCModePnt**

```markdown
ccc:(mode,num)=(b1,b2)->(a1,a2)
```

- **ccc**: Channel
- **b1,b2**: Before change
- **a1,a2**: After change

The following settings (those that have been changed among the following two settings):

- **mode**: Mode [OFF, Bias, Appro, Corr]
- **num**: Number of set points

**Example:**

0001:(mode,num)=(OFF,3)->(Appro,12)

**Calibration correction value change SetCCValue**

```markdown
ccc:pp:(input,output)=(b1,b2)->(a1,a2)
```

- **ccc**: Channel
- **p**: Set number
- **b1,b2**: Before change
- **a1,a2**: After change
## Commands and Responses

### 2.10 ASCII Output Format

<table>
<thead>
<tr>
<th>Operation</th>
<th>Event string</th>
<th>### information and detailed information</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>The following settings (those that have been changed among the following two settings)</td>
</tr>
<tr>
<td>Save directory change</td>
<td>SetDirectory</td>
<td>input Calibration correction value output Output calibration value</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example: 0001:02:(output)=(1.234)-&gt;(2.234) (b1)-&gt;(a1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Folder name Example: (DATA0)-&gt;(DATA1)</td>
</tr>
<tr>
<td>Recipient address change</td>
<td>SetRecipient</td>
<td>Recipient:1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 Recipient number [1, 2]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example: Recipient:1</td>
</tr>
<tr>
<td>Source address change</td>
<td>SetSender</td>
<td>—</td>
</tr>
<tr>
<td>Subject change</td>
<td>SetSubject</td>
<td>—</td>
</tr>
<tr>
<td>Login change</td>
<td>SetLogin</td>
<td>User:UUU</td>
</tr>
<tr>
<td></td>
<td></td>
<td>UUU User number</td>
</tr>
<tr>
<td>Schedule setting</td>
<td>SetSchedule##</td>
<td>Schedule number (s,dd,ck,dy,bz,cc,t,1,2)=(b1,b2,b3,b4,b5,b6,b7,b8,b9)-&gt;(a1,a2,a3,a4,a5,a6,a7,a8,a9)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Before change a1,...,a9 After change</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The following settings (those that have been changed among the following eight settings)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>s On/Off</td>
</tr>
<tr>
<td></td>
<td></td>
<td>dd Date [yyyy/mo/dd]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ck Notification day [1 to 10]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cy Renotification interval [10min, 30min, 1h, 8h]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>bz Notification buzzer [ON, OFF]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>cc Load settings [ON, OFF]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>t Title</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 Notification content 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 Notification content 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>However, the title before change, notification content 1 before and after change, and notification content 2 before and after change are not output (spaces). Example: 02:(s,ck,1,2)=(OFF,3,..)-&gt;(ON,4,'abc',.)</td>
</tr>
<tr>
<td>Correction coefficient setting</td>
<td>SetCFactor</td>
<td>ccccc:pp:(uncorrected, instru,sensor)=(b1,b2,b3)-&gt;(a1,a2,a3)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c Channel</td>
</tr>
<tr>
<td></td>
<td></td>
<td>p Correction position</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b1,b2,b3 Before change</td>
</tr>
<tr>
<td></td>
<td></td>
<td>a1,a2,a3 After change</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The following settings (those that have been changed among the following two settings)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Uncorrected value (before and after change)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Instrument correction coefficient (before and after change)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sensor correction coefficient (before and after change)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example: 0001:02:(sensor)=(1.234)-&gt;(2.234)</td>
</tr>
</tbody>
</table>
### Operation Event string ### information and detailed information

<table>
<thead>
<tr>
<th>Operation</th>
<th>Event string</th>
<th>### information and detailed information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calibration correction/set point change (communication channels)</td>
<td>SetComCCMode Pnt</td>
<td>uuu:dd:cccc:ssssssssss(mode,num)=(b1,b2)-&gt;(a1,a2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>u</td>
</tr>
<tr>
<td></td>
<td></td>
<td>d</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c</td>
</tr>
<tr>
<td></td>
<td></td>
<td>s,***s</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b1,b2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>a1,a2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The following settings (those that have been changed among the following two settings)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[OFF, Bias, Appro, Corr]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>num</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>000:00:C001.: (mode,num)=(OFF,3)-&gt;(Appro,12)</td>
</tr>
<tr>
<td>Calibration correction value change (communication channels)</td>
<td>SetCom #######</td>
<td>Action (output in the event string)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>###: CCValue: Linearizer Approximation/Linearizer Bias</td>
</tr>
<tr>
<td></td>
<td></td>
<td>uuu:pp:dd:cccc:ssssssssss:(input,output)=(b1,b2)-&gt;(a1,a2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>u</td>
</tr>
<tr>
<td></td>
<td></td>
<td>p</td>
</tr>
<tr>
<td></td>
<td></td>
<td>d</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c</td>
</tr>
<tr>
<td></td>
<td></td>
<td>s,***s</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b1,b2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>a1,a2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The following settings (those that have been changed among the following two settings)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>input</td>
</tr>
<tr>
<td></td>
<td></td>
<td>output</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>000:02:00:C001.: (output)=(1.234)-&gt;(2.234)</td>
</tr>
</tbody>
</table>
### Operation | Event string | ### information and detailed information
---|---|---
Calibration correction | SetCom | Action (output in the event string)
value change | | ###: CFactor: Correction coefficient
Correction | | uuu:pp:dd.cccc.sssssssssssss
coefficient setting | | (uncorrected,instru,sensor)=(b1,b2,b3)->(a1,a2,a3)
(communication | | u: 0
channels) | | p: Correction position
d: 0
| | c: Communication channel number
| | s: Serial number: Null
| | b1,b2,b3: Before change
| | a1,a2,a3: After change

**Example:**

000:02:00:C001::(sensor)=(1.234)->(2.234)

**Change while recording is stopped**

Setting change | SetParameter | ss•••:kk•••
---|---|---
| | ss•••: Setting file name
| | kk•••: Setting change type
| | [Security,Comm,I/OCh,MathCh,CommCh,Other]
| | (list of changed settings)

**Example:**

000111_131219_095412.GSL:Security,
Comm,I/OCh,MathCh,CommCh,Other

**Schedule setting** | SetSchedule | Same as the setting changes during recording.
---|---|---
**Updating**
**Other updates** | Update### | ### Action (output in the event string)
---|---|---
| | Web | Web application
| | <Detailed information>
| | vv••• | vv... Version number
2.10.24 External Storage Medium and Internal Memory File List (FMedia)
The output in response to the command “FMedia,DIR” is shown below.

Syntax
EA\r\nyy/mm/dd hh:mm:ss_lll...l_fff...
yy/mm/dd hh:mm:ss_<DIR>_dddddddd...\r\nEN\r

yyyy/mo/dd hh:mm:ss Time of file generation
yy Year (1900 to 2099)
mm Month (01 to 12)
dd Day (01 to 31)
hh Hour (00 to 23)
mi Minute (00 to 59)
ss Second (00 to 59)
lll...l File size (fixed to 10 characters. Unused character positions are filled with spaces.)
For directories, <DIR> is output.
fff... File name

2.10.25 External Storage Medium Free Space (FMedia)
The output in response to the command “FMedia,CHKDSK” is shown below.

Syntax
EA\r
zzzzzzz_Kbytes_free\r
EN\r

zzzzzzz Free space (KB)
2.10.26 Setting Data (FCnf)

The output in response to the command “FCnf” is shown below.

Syntax
EA<crlf>
<Response to a setting query>
EN<crlf>

The setting data is output in the format of the response to a setting query.

2.10.27 Decimal Place and Unit Information (FChInfo)

The output in response to the command “FChInfo” is shown below.

Syntax
EA<crlf>
s_cccc_uuuuuuuuuu,pp<crlf>
s_cccc_uuuuuuuuuu,pp<crlf>
s_cccc_uuuuuuuuuu,pp<crlf>
EN<crlf>

\begin{itemize}
  \item \textbf{s} Data status
    \begin{itemize}
      \item N Normal
      \item D Differential input
      \item S Skip
    \end{itemize}
  \item \textbf{cccc} Channel number (I/O channel, math channel, communication channel)
  \item \textbf{uuuuuuuuu} Unit information (fixed to 10 characters. Unused character positions are filled with spaces.)
  \item \textbf{pp} Decimal place (00 to 05)
\end{itemize}

The decimal place of the mantissa on channels set to LOG scale (/LG)
2.10 ASCII Output Format

### 2.10.28 System Configuration (FSysConf)

The output in response to the command “FSysConf” is shown below.

#### Syntax

**When no expandable I/O is connected**

EA<crlf>
Unit:00
00:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
01:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
02:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
03:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
04:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
05:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
06:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
07:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
08:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
09:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
EN<crlf>

**When an expandable I/O or sub unit is connected**

EA<crlf>
U00f:cccccccccccccccc_uuuuuuuuuuuuuuuu_DEFGHIJKLMNOPQRSTUVWXYZ<crlf>
U01f:cccccccccccccccc_uuuuuuuuuuuuuuuu_DEFGHIJKLMNOPQRSTUVWXYZ<crlf>
U02f:cccccccccccccccc_uuuuuuuuuuuuuuuu_DEFGHIJKLMNOPQRSTUVWXYZ<crlf>
U03f:cccccccccccccccc_uuuuuuuuuuuuuuuu_DEFGHIJKLMNOPQRSTUVWXYZ<crlf>
U04f:cccccccccccccccc_uuuuuuuuuuuuuuuu_DEFGHIJKLMNOPQRSTUVWXYZ<crlf>
U05f:cccccccccccccccc_uuuuuuuuuuuuuuuu_DEFGHIJKLMNOPQRSTUVWXYZ<crlf>
U06f:cccccccccccccccc_uuuuuuuuuuuuuuuu_DEFGHIJKLMNOPQRSTUVWXYZ<crlf>
Unit:nn
00:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
01:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
02:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
03:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
04:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
05:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
06:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
07:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
08:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
09:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
Unit:nn
00:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
01:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
02:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
03:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
04:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
05:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
06:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
07:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
08:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
09:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
:
Unit:nn
00:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
01:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
02:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
03:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
04:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
05:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
06:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
07:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
08:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
09:cccccccccccccccc_uuuuuuuuuuuuuuuu_defghijklmnopqrs<crlf>
EN<crlf>
Output example when an expandable I/O or sub unit is connected

- The unit information area (e.g. U00) will contain the expansion module name. All seven units are output regardless of whether expansion modules are available.
- The module information area (after Unit) will contain I/O module names. Only the units that have modules installed will be output.

<table>
<thead>
<tr>
<th>EA</th>
<th>Module models that are actually installed</th>
</tr>
</thead>
<tbody>
<tr>
<td>U00*:GX20-1J</td>
<td>GX20-1J</td>
</tr>
<tr>
<td>U01 :GX90EX-02-TP1</td>
<td>GX90EX-02-TP1</td>
</tr>
<tr>
<td>U02 :--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>U03 :--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>U04 :--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>U05 :--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>U06 :--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>Unit:00</td>
<td>Module models recognized by the GX</td>
</tr>
<tr>
<td>00:GX90XA-10-U2</td>
<td>GX90XA-10-U2</td>
</tr>
<tr>
<td>01:GX90XA-10-U2</td>
<td>GX90XA-10-U2</td>
</tr>
<tr>
<td>02:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>03:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>04:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>05:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>06:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>07:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>08:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>09:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>Unit:01</td>
<td>Module status</td>
</tr>
<tr>
<td>00:GX90XA-10-U2</td>
<td>GX90XA-10-U2</td>
</tr>
<tr>
<td>01:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>02:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>03:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>04:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>05:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>06:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>07:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>08:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>09:--------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>EN</td>
<td>Displays the module model code.</td>
</tr>
<tr>
<td>cccccccccccccccccccc</td>
<td>Module not installed (16 hyphens)</td>
</tr>
<tr>
<td>uuuuuuuuuuuuuuuuuuuu</td>
<td>Displays the module model code.</td>
</tr>
<tr>
<td>defghijklmnopqrs</td>
<td>Module status</td>
</tr>
<tr>
<td>d</td>
<td>Normal</td>
</tr>
<tr>
<td>e</td>
<td>System data error</td>
</tr>
<tr>
<td>f</td>
<td>Calibration value error</td>
</tr>
<tr>
<td>g</td>
<td>Parameter error</td>
</tr>
<tr>
<td>h</td>
<td>Reserved (-)</td>
</tr>
<tr>
<td>i</td>
<td>FRAM error</td>
</tr>
<tr>
<td>j</td>
<td>Reserved (-)</td>
</tr>
<tr>
<td>k</td>
<td>Reserved (-)</td>
</tr>
<tr>
<td>l</td>
<td>A/D error</td>
</tr>
<tr>
<td>m</td>
<td>RJC error</td>
</tr>
<tr>
<td>n</td>
<td>Reserved (-)</td>
</tr>
<tr>
<td>o</td>
<td>Reserved (-)</td>
</tr>
<tr>
<td>p</td>
<td>Reserved (-)</td>
</tr>
<tr>
<td>q</td>
<td>Reserved (-)</td>
</tr>
<tr>
<td>r</td>
<td>Reserved (-)</td>
</tr>
<tr>
<td>s</td>
<td>Reserved (-)</td>
</tr>
<tr>
<td>nn</td>
<td>Unit number</td>
</tr>
</tbody>
</table>
2.10 ASCII Output Format

2.10.29 Bluetooth Device Information (FBTDevInfo)

The output in response to the command “FBTDevInfo” is shown below.

Syntax
EA\r\n(BD address),(module information)\r\nEN\r

(module information) xxxx (user-defined character string)

Before the Bluetooth function is turned on after power-on, the xx of the BD address area will be spaces, and the module information area will be empty (no characters).

Output Example
When p1 is omitted
EA\r\nB4:17:D3:AC:07:AA,Init R02.01.1(build 000)\r\nEN\r

2.10.30 Reminder Information Output (FReminder)

The output in response to the command “FReminder” is shown below.

Syntax

EA<crlf>
nn ssssss yyy/mo/dd_YYY/MO/DD_e...<crlf>
EN<crlf>

nn ssssss
Schedule number (01 to 12)

Status
None No registration
Normal Before notification
Notice Notifying
Expire After expiration

yyyy/mo/dd
YYYY/MO/DD
Previous set date

E...
Expiration date

If the status is None, the subsequent information is not output.

Output Example

EA
01_None
02_Normal_2015/02/28_2015/03/30_30
03_Notice_2015/02/25_2015/02/28_3
04_Expire_2015/02/20_2015/02/13_-7
05_Normal_2015/02/28_2015/03/30_30
06_Normal_2015/02/28_2015/03/30_30
07_Normal_2015/02/28_2015/03/30_30
08_None
09_None
10_None
11_None
12_None
EN
2.10.31 Outputs the Most Recent Control Data (FCtrlData)

The output in response to the command “FCtrlData” is shown below. Outputs the most recent control data per loop.

**Syntax**

```
EA<br>
DATE_yy/mo/dd<br>
TIME_hh:mm:ss.mmmt<br>
llll,S_fdddddddE-pp,S_fdddddddE-pp,S_fdddddddE-pp,aaaa1aaaa2aaaa3aaaa4<br>
EN<br>
```

- **yy/mo/dd**
  - **yy**: Year (00 to 99)
  - **mo**: Month (01 to 12)
  - **dd**: Day (01 to 31)

- **hh:mm:ss.mmmt**
  - **hh**: Hour (00 to 23)
  - **mm**: Minute (00 to 59)
  - **ss**: Second (00 to 59)
  - **mmm**: Millisecond (000 to 999)

- **llll**
  - Loop number

- **s**
  - Data status
    - **N**: Normal
    - **S**: Skip
    - **O**: Over
    - **E**: Errors
    - **B**: Burnout
    - **F**: No data
    - **M**: Luck of data/Module not installed

- **f**
  - Sign (+ or -)

- **ddddddd**
  - **Mantissa**: (00000000 to 99999999; 8 digits)
  - If the data status is **O** (±over) or **B** (burnout), the mantissa will be the value of -5% to 105% of the range.
  - If the data status is **E**, the mantissa is 99999999.

- **pp**
  - Exponent (00 to 04)

- **aaaa1aaaa2aaaa3aaaa4**
  - **Alarm Status**
    - **aaaa1**: Alarm status 1
    - **aaaa2**: Alarm status 2
    - **aaaa3**: Alarm status 3
    - **aaaa4**: Alarm status 4

  **aaa1, aaaa2, aaaa3, or aaaa4** is set to one of the following:
  - **PVH**: PV high limit
  - **PVL**: PV low limit
  - **SPH**: SP high limit
  - **SPL**: SP low limit
  - **DVH**: Deviation high limit
  - **DVL**: Deviation low limit
  - **DVO**: Deviation H/L limits
  - **DVI**: Deviation within H/L limits
  - **OTH**: Control output high limit
  - **OTL**: Control output low limit
  - **PVR**: PV velocity

  If an alarm has not occurred, the alarm status is set to space.

`s_fdddddddE-pp` is in order of PV, SP, and OUT.

The data time outputted is not the time of a control period but the data acquisition time by a communication command.
2.10 ASCII Output Format

2.10.32 SP Number and PID Number Output (FCtrlNo.)

The output in response to the command “FCtrlNo” is shown below.

**Syntax**

```
EA<br>
LOOP, llll<br>
SPNO, x<br>
PIDNO, y<br>
EN<br>
```

- **llll**  Loop number (L001 to L692)
- **x**  SP number (1 to 8)
- **y**  PID number (1 to 8, R)

**Output Example**

```
FCtrlNo,L001
EA
LOOP, L001<br>
SPNO, 1<br>
P1DNO, 2<br>
EN
```
2.10 ASCII Output Format

2.10.33 Control Mode Output (FCtrlMode)
The output in response to the command “FCtrlMode” is shown below.

Syntax
EA<br>llll,xxx.xxx.xxx.xxx<br>EN<br>
llll Loop number (L001 to L692)
xxx The states of bits 31 to 24, 23 to 16, 15 to 8, and 7 to 0 are indicated in order using
decimal notation.
Bit 0
0 RUN
1 STOP
Bit 1
0 LOCAL
1 REMOTE
Bit 2
Bit 3
0 AUTO
1 MANUAL
2 CASCADE
Bit 4
0 Auto-tuning off
1 Auto-tuning on
Bits 5 to 7
Not used
Bit 8
0 AI1 not used
1 AI1 used
Bit 9
0 AI2 not used
1 AI2 used
Bit 10
0 EXPV1 not used
1 EXPV1 used
Bit 11
0 EXPV2 not used
1 EXPV2 used
Bits 12 to 28
Not used
Bits 29 and 30
0 PROG
1 RESET
2 LOCAL
Bit 31
0 Program operation not available
1 Program operation available

Output Example
When program operation is in progress, auto-tuning is in progress, cascade operation is in
progress, and when in PROG, REMOTE, and RUN states
EA<br>L001,128.000.000.026<br>EN<br>
2.10.34 Program Operation Mode Output (FPrgMode)

The output in response to the command “FPrgMode” is shown below.

Syntax

```
EA<br>
pp,xxx.xxx.xxx.xxx<br>
EN<br>
```

**PP**  Program pattern number (1 to 99)

**xxx**  The states of bits 31 to 24, 23 to 16, 15 to 8, and 7 to 0 are indicated in order using decimal notation.

- Bit 0
  - 0: Program stopped
  - 1: Program running

- Bit 1
  - 0: Not holding
  - 1: Holding

- Bit 2
  - 0: Not waiting
  - 1: Waiting

- Bits 3 to 31
  - Not used

**Output Example**

When not waiting, not holding, and program is running

```
EA<br>
01,000.000.000.001<br>
EN<br>
```
2.10.35 Program Pattern Information Output (FPrgPtnInfo)

The output in response to the command “FPrgPtnInfo” is shown below.

Syntax

EA<crlf>
PTNNO,a<crlf>
SEGNO,b<crlf>
SEGUSE,c<crlf>
SEGTM, hh:mm:ss<crlf>
WAITTM, hh:mm:ss<crlf>
RPT-MODE, d<crlf>
RPT-CNT, e<crlf>
RPT-REM, f<crlf>
RPT-START, g<crlf>
RPT-END, h<crlf>
STARTTM, yy/mo/dd hh:mm:ss<crlf>
ENDTM, yy/mo/dd hh:mm:ss<crlf>
EN<crlf>

PTNNO Running pattern number a (1 to 99)
SEGNO Running segment number b (1 to 99)
SEGUSE Number of segments c of the currently selected pattern (1 to 99)
SEGTM Remaining segment time
  hh Hours (00 to 99)
  mm Minutes (00 to 59)
  ss Seconds (00 to 59)
WAITTM Elapsed time of waiting
  hh Hours (00 to 99)
  mm Minutes (00 to 59)
  ss Seconds (00 to 59)
RPT-MODE Repeat setting d of the running pattern
  0 ON
  1 OFF
  2 Repeat indefinitely
RPT-CNT Repeat count e of the running pattern (0 to 999)
RPT-REM Remaining repeat count f of the running pattern (0 to 999)
RPT-START Start segment number g of repeat operation (1 to 99)
RPT-END End segment number h of repeat operation (1 to 99)
STARTTM Program operation start time
  yy Year (00 to 99)
  mo Month (01 to 12)
  dd Day (01 to 31)
  hh Hour (00 to 99)
  mm Minute (00 to 59)
  ss Second (00 to 59)
ENDTM Program operation stop time
  yy Year (00 to 99)
  mo Month (01 to 12)
  dd Day (01 to 31)
  hh Hour (00 to 99)
  mm Minute (00 to 59)
  ss Second (00 to 59)
Output Example

EA<crlf>
PTNNO,5
SEGNO,10
SEGUSE,7
SEGTM,11:05:22
WAITTM,00:06:00
RPT-MODE,1
RPT-CNT,20
RPT-REM,2
RPT-START,3
RPT-END,6
STARTTM,17/05/03 10:00:00
ENDTM,17/05/05 10:00:00
EN<crlf>

2.10.36 PV event and time event information output (FPrgEvent)

The output in response to the command “FPrgEvent” is shown below.

Syntax

EA<crlf>
PP,xxx.xxx.xxx.xxx<crlf>
PP,yyy.yyy.yyy.yyy<crlf>
EN<crlf>

PP Program pattern number (1 to 99)

xxx The states of bits 31 to 24, 23 to 16, 15 to 8, and 7 to 0 are indicated in order using decimal notation.

Bit 0
0 PV event 1 off
1 PV event 1 on

Bit 31
0 PV event 32 off
1 PV event 32 on

yyy The states of bits 31 to 24, 23 to 16, 15 to 8, and 7 to 0 are indicated in order using decimal notation.

Bit 0
0 Time event 1 off
1 Time event 1 on

Bit 31
0 Time event 32 off
1 Time event 32 on

Output Example

FPrgEvent,1
When PV event 1 and time events 2 and 3 are occurring
EA<crlf>
01,000.000.000.001<crlf>
01,000.000.000.006<crlf>
EN<crlf>
2.10.37  Program control end signal status output (FPrgEnd)

The output in response to the command “FPrgEnd” is shown below.

Syntax

\[
\text{EA}\text{<crlf>}
\text{pp},x<\text{crlf>}
\text{EN}\text{<crlf>}
\]

\(\text{PP}\)  Program pattern number (1 to 99)
\(x\)  Program control end signal status

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5-second program control end signal is off</td>
</tr>
<tr>
<td>1</td>
<td>5-second program control end signal is on</td>
</tr>
</tbody>
</table>

Output Example

FPrgEnd,1
When the 5-second program control end signal is on
EA<\text{crlf>}
01,1<\text{crlf>}
EN<\text{crlf>}

2.10.38  Currently Running Program Pattern Number and Status Output (FPrgPtnCur)

The output in response to the command “FPrgPtnCur” is shown below.

Syntax

\[
\text{EA}\text{<crlf>}
\text{pp},xxx.xxx.xxx.xxx\text{<crlf>}
\text{oo},cccc\text{<crlf>}
\ldots\text{<crlf>}
\text{oo},ccc\text{<crlf>}
\text{EN}\text{<crlf>}
\]

\(\text{pp}\)  Program pattern number (1 to 99)
\(\text{xxx}\)  The states of bits 31 to 24, 23 to 16, 15 to 8, and 7 to 0 are indicated in order using decimal notation.
\(\text{oo}\)  Number (1 to 20)
\(\text{cccc}\)  Loop number (L001 to L692)

Output Example

Program pattern 1, not holding, not waiting, program running, loop1 and loop2 assignment
EA<\text{crlf>}
01,000.000.000.001<\text{crlf>}
01,L001
02,L011
EN<\text{crlf>}

2-170
2.10.39 Instrument Manufacturer (_MFG)

The output in response to the command "_MFG" is shown below. Outputs the instrument manufacturer.

Output Example
EA<br>
YOKOGAWA<br>
EN<br>

2.10.40 Instrument's Product Name (_INF)

The output in response to the command "_INF" is shown below.

Output Example
EA<br>
'GX20/GP20',123456789,xx-xx-xx-xx-xx-Rx.xx.xx<br>
EN<br>

'GX20/GP20'  Product name (‘GX20/GP20’, ‘GX10/GP10’, or ‘GM10’)
123456789  Product serial number
xx-xx-xx-xx-xx  MAC address (xx's are hexadecimals)
Rx.xx.xx  Firmware version
2.10.41 Instrument's Basic Specifications (_COD)

The output in response to the command "_COD" is shown below.

Output Example

```
EA\r\n'GX20',-1,J,1,M <\r\nEN\r

'GX20'                Model
-1                     Type
                     -1  100 channels
                     -2  500 channels
J                       Display language
                     J  Japanese
                     E  English
                     C  Chinese
I                       Supply voltage
                     Blank (when the product name is GX10, GX20, or GM10)
                     1  100 VAC, 240 VAC (when the product name is GP10 or GP20)
M                       Power cord
                     Blank (when the product name is GX10, GX20, or GM10)
                     M  PSE cable
                     D  UL/CSA cable
                     F  VDE cable
                     R  AS cable
                     Q  BS cable
                     H  GB cable
                     N  NBR cable
```

2.10.42 Instrument's Firmware Version Information (_VER)

The output in response to the command "_VER" is shown below.

Output Example

```
EA\r
B999999,Rx.xx.xx,'Main Program'\r
B999999,Rx.xx.xx, 'Web Program'\r
EN\r

B999999                Firmware part number (first line), Web program part number (second line)
Rx.xx.xx               Firmware version (first line), Web program version (second line)
```
2.10.43 Instrument's Option Installation Information (_OPT)

The output in response to the command "+OPT" is shown below.

**Output Example**

```
EA<br>
/C2,'RS-232'<br>
/C3,'RS-422/485'<br>
/C8,'Bluetooth'<br>
/D5,'VGA output'<br>
/FL,'Fail output (1 point) '<br>
/MT,'Mathematical function (with report function) '<br>
/MC,'Communication channel function'<br>
/P1,'24 VDC/AC power supply'<br>
/UH,'USB interface (Host 2 ports) '<br>
/AS,'Advanced security functions'<br>
/BT,'Multi-batch function'<br>
/AH,'Aerospace heat treatment'<br>
/E1,'EtherNet/IP communication'<br>
/E2,'WT connect functions'<br>
/E3,'OPC-UA server'<br>
/E4,'SLMP communication'<br>
/CN,'Custom display functions'<br>
/LS,'Log scale functions'<br>
/PG,'Program pattern'<br>
/U__0,'Model pre-installed with analog (universal) input module(s)'<br>
/CR__,'Model pre-installed with digital output module(s) and/or digital input module(s)'<br>
EN<br>
```

```
/C2   RS-232  
/C3   RS-422/485  
/C8   Bluetooth  
/D5   VGA output  
/FL   Fail output, 1 point  
/MT   Math (including the report function)  
/MC   Communication channel function  
/P1   24VDC/AC power supply  
/UH   USB interface (host 2 ports)  
/AS   Advanced security function (Part 11 compliant)  
/BT   Multi-batch function  
/AH   Aerospace heat treatment  
/E1   EtherNet/IP communication  
/E2   WT communication  
/E3   OPC-UA server  
/E4   SLMP communication  
/CN   Custom display function  
/LS   Log scale  
/PG   Program control  
/UxY  Model pre-installed with analog (universal) input modules  
     Terminal type  
     S   Screw terminal  
     C   Clamp terminal  

     Number of analog (universal) input modules installed  
     1, 2, 3, 4, 5, 6, 7, 8, 9, A (where A represents 10)  

/CRY  Model pre-installed with digital output modules and/or digital input modules  
     Number of digital output (C contact) modules installed  
     1, 2, 3, 4, 5  
     Number of digital input modules installed  
     1, 2  
```
2.10.44 Instrument's Temperature Unit and Daylight Saving Time Installation Information (_TYP)

The output in response to the command "_TYP" is shown below.

Output Example

```
EA<crlf>
DST,'Summer time/Winter time'<crlf>
DEGF,'degF'<crlf>
EN<crlf>
```

DST Daylight saving time enabled
DEGF Fahrenheit temperature unit enabled

2.10.45 Instrument's Error Number Information (_ERR)

The output in response to the command "_ERR" is shown below.

Output Example

```
EA<crlf>
10:1:2,'Dram Error'<crlf>
500:2:5,'Media Error'<crlf>
EN<crlf>
```
### 2.10.46 Instrument’s Unit Configuration Information (_UNS or _UNR)

The output in response to the command "_UNS" or "_UNR" is shown below.

#### Syntax

```
EA<crlf>
p1,p2,p3,p4,p5,p6,p7,p8,p9,p10<crlf>
...<crlf>
EN<crlf>
```

#### Output Example

```
EA<crlf>
Main,0,’GX20-1J’,1234567,xx-xx-xx-xx-xx-xx,R1.01.01,/MT /C2,0,10,--
-------------------<crlf>
Sub,1,’GX90EX-02-ET1’,1234567,xx-xx-xx-xx-xx-xx,R1.01.01,,0,6,-----
-------------------<crlf>
EN<crlf>
```

One line (p1 to p10) contains configuration information of a single unit.

<table>
<thead>
<tr>
<th>p&lt;sub&gt;n&lt;/sub&gt;</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>Main, Sub</td>
<td>Unit dependency (main or sub) information.</td>
</tr>
<tr>
<td>p2</td>
<td>0,1</td>
<td>Unit address number. The address number of the main unit is 0.</td>
</tr>
<tr>
<td>p3</td>
<td>‘GX20-1J’, ‘GX90EX-02-ET1’</td>
<td>Unit name (model name). Enclosed in single quotation marks. The main unit model or expansion module model in the expandable I/O unit or sub unit.</td>
</tr>
<tr>
<td>p4</td>
<td>1234567</td>
<td>Product serial number.</td>
</tr>
<tr>
<td>p5</td>
<td>xx-xx-xx-xx</td>
<td>MAC address.</td>
</tr>
<tr>
<td>p6</td>
<td>R1.01.01</td>
<td>Firmware version. The output format is &quot;R+version.&quot;</td>
</tr>
<tr>
<td>p7</td>
<td>/MT /C2</td>
<td>Options. Codes of installed options delimited by spaces.</td>
</tr>
<tr>
<td>p8</td>
<td>0</td>
<td>Fixed at 0.</td>
</tr>
<tr>
<td>p9</td>
<td>6, 10</td>
<td>Maximum number of installable modules. If there are not installable modules, 0 is output.</td>
</tr>
<tr>
<td>p10</td>
<td>-------------------</td>
<td>Unit status. The unit status is output in a character string. See the Expansion module status in section 2.10.24, “System Configuration (FSysConf).”</td>
</tr>
</tbody>
</table>

The main unit and expansion module information is output (indicated in green below).

**Without an expandable I/O or sub unit**

```
Main unit
|   | AI | AI |
```

**With an expandable I/O or sub unit**

```
Main unit
| AI | EX |
```

```
EX: Expansion module
<table>
<thead>
<tr>
<th>AI, DI, etc.:</th>
<th>I/O module</th>
</tr>
</thead>
</table>
```
2.10 ASCII Output Format

2.10.47 Instrument’s Module Configuration Information (_MDS or MDR)

The output in response to the command "_MDS" or "_MDR" is shown below.

Syntax

EA<crlf>
p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11<crlf>
EN<crlf>

Output Example

EA<crlf>
Main,0,1,'GX90YD-06-11',1234567,R1.01.01,,0,0,6,----------------<crlf>
Main,0,9,'GX90EX-02-ET1'1234567,R1.01.01,,0,0,0,----------------<crlf>
Sub,1,0,'GX90XA-10-U2',1234567,R1.01.01,,0,10,0,----------------<crlf>
Sub,1,1,'GX90XA-10-U2',1234567,R1.01.01,,0,10,0,----------------<crlf>
Sub,1,2,'GX90XA-10-U2',1234567,R1.02.01,,0,10,0,----------------<crlf>
Sub,2,0,'GX90XA-10-U2',1234567,R1.02.01,,0,10,0,----------------<crlf>
Sub,2,1,'GX90XD-16-11',1234567,R1.01.01,,0,16,0,----------------<crlf>
EN<crlf>

One line (p1 to p11) contains configuration information of a single module.

<table>
<thead>
<tr>
<th>p_n</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>Main, Sub</td>
<td>Unit dependency (main or sub) information. Main: Main unit (Only a single one exists in a system. GX/GP or GM main unit) Sub: Sub unit (Units other than the main unit. GX/GP expandable I/O or GM sub unit)</td>
</tr>
<tr>
<td>p2</td>
<td>0, 1, 2</td>
<td>Address number of the unit that the module is installed in. Fixed at 0.</td>
</tr>
<tr>
<td>p3</td>
<td>0, 1, 2</td>
<td>Slot number of the unit that the module is installed in (0 reference).</td>
</tr>
<tr>
<td>p4</td>
<td>'GX90YD-06-11', 'GX90EX-02-ET1', 'GX90XA-10-U2', 'GX90XD-16-11'</td>
<td>Module name (model name). Enclosed in single quotation marks. • All modules installed in the main unit • A module installed in an expandable I/O or sub unit (excluding the expansion module)</td>
</tr>
<tr>
<td>p5</td>
<td>1234567</td>
<td>Product serial number.</td>
</tr>
<tr>
<td>p6</td>
<td>R1.01.01, R1.02.01</td>
<td>Module firmware version. The output format is “R+version.”</td>
</tr>
<tr>
<td>p7</td>
<td>Space</td>
<td>Options. Codes of installed options delimited by spaces.</td>
</tr>
<tr>
<td>p8</td>
<td>0</td>
<td>Fixed at 0.</td>
</tr>
<tr>
<td>p9</td>
<td>0, 10, 8</td>
<td>Maximum number of input channels allowed on the module. If there are no inputs, 0 is output.</td>
</tr>
<tr>
<td>p10</td>
<td>0, 16</td>
<td>Maximum number of output channels allowed on the module. If there are no outputs, 0 is output.</td>
</tr>
<tr>
<td>p11</td>
<td></td>
<td>Module status. The Module status is output in a character string.</td>
</tr>
</tbody>
</table>
The I/O module information is output (indicated in green below).

**Without an expandable I/O or sub unit**

- Main unit
  - AI
  - AI

**With an expandable I/O or sub unit**

- Main unit
  - AI
  - EX

- EX
  - AI
  - AI

**Ex:** Expansion module

**AI, DI, etc.:** I/O module
2.11 Format of the Data Block of Binary Output

This section explains the data that is stored in the data block in the binary output of data output response. For the entire structure of the binary output format, see “Binary Output” on page 2-119.

2.11.1 Most Recent Channel Data (FData)

The output in response to the command “FData,1” is shown below. Outputs the most recent I/O channel, math channel, and communication channel data.

Configuration

The figure below shows the structure of the output data. Data is stored in “Block 1.”

Number of Blocks (16 bits)
Always 1.

Number of Bytes (16 bits)
Stores the number of bytes of block 1.

Block 1

The figure below shows the structure of block 1.

Data Time

<table>
<thead>
<tr>
<th>Item (Number of Bits)</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Year (8 bits)</td>
<td>0 to 99</td>
</tr>
<tr>
<td>Month (8 bits)</td>
<td>1 to 12</td>
</tr>
<tr>
<td>Day (8 bits)</td>
<td>1 to 31</td>
</tr>
<tr>
<td>Hour (8 bits)</td>
<td>1 to 23</td>
</tr>
<tr>
<td>Minute (8 bits)</td>
<td>0 to 59</td>
</tr>
<tr>
<td>Second (8 bits)</td>
<td>0 to 59</td>
</tr>
<tr>
<td>Millisecond (16 bits)</td>
<td>0 to 999</td>
</tr>
</tbody>
</table>
### Additional Information (64 bits)

Bit 0: Daylight saving time (0: standard time; 1: daylight saving time)

### Data Type (4 bits)

Indicates the data type. (1: 32 bit integer; 2: 32 bit floating point)

Data values for channels set to Log scale (/LG) are 32-bit floating-point type.

The alarm statuses of control alarms (when a PID control module is installed) are all set to zero.

### Channel Type (4 bits)

Indicates the channel type.

<table>
<thead>
<tr>
<th>Value</th>
<th>Channel Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>I/O channel</td>
</tr>
<tr>
<td>2</td>
<td>Math channel</td>
</tr>
<tr>
<td>3</td>
<td>Communication channel</td>
</tr>
</tbody>
</table>

### Status (8 bits)

Indicates the channel status.

<table>
<thead>
<tr>
<th>Value</th>
<th>Channel Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No error</td>
</tr>
<tr>
<td>1</td>
<td>Skip</td>
</tr>
<tr>
<td>2</td>
<td>+Over</td>
</tr>
<tr>
<td>3</td>
<td>-OVER</td>
</tr>
<tr>
<td>4</td>
<td>+Burnout</td>
</tr>
<tr>
<td>5</td>
<td>-Burnout</td>
</tr>
<tr>
<td>6</td>
<td>A/D error</td>
</tr>
<tr>
<td>7</td>
<td>Invalid data</td>
</tr>
<tr>
<td>16</td>
<td>Math result is NaN.</td>
</tr>
<tr>
<td>17</td>
<td>Communication error</td>
</tr>
</tbody>
</table>

### Channel Number (16 bits)

Indicates the channel number. Stored in the following manner depending on the channel type.

<table>
<thead>
<tr>
<th>Channel Type</th>
<th>Channel Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/O channel</td>
<td>6 bits 10 bits</td>
</tr>
<tr>
<td>Reserved</td>
<td>Channel number</td>
</tr>
<tr>
<td>Math channel</td>
<td>6 bits 10 bits</td>
</tr>
<tr>
<td>Reserved</td>
<td>Channel number</td>
</tr>
<tr>
<td>Communication channel</td>
<td>6 bits 10 bits</td>
</tr>
<tr>
<td>Reserved</td>
<td>Channel number</td>
</tr>
</tbody>
</table>

### Alarm (32 bits)

Indicates the alarm status.

<table>
<thead>
<tr>
<th>Alarm 1</th>
<th>Alarm 2</th>
<th>Alarm 3</th>
<th>Alarm 4</th>
</tr>
</thead>
</table>
The eight bit values of alarm 1 to alarm 4 are described in the table below.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 to 5</td>
<td>0</td>
<td>No alarm</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>High limit alarm</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Low limit alarm</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Difference high limit alarm</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Difference low limit alarm</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>High limit on rate-of-change alarm</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>Low limit on rate-of-change alarm</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>Delay high limit alarm</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>Delay low limit alarm</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>No alarm is activated.</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>An alarm is activated.</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>Alarm nonhold state</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>Alarm hold state</td>
</tr>
</tbody>
</table>
2.11.2 Channel FIFO Data (FFifoCur)

The output in response to the command “FFifoCur,0” is shown below. Outputs the I/O channel, math channel, and communication channel FIFO data.

Configuration

Data is stored in “Block 1” shown below.

<table>
<thead>
<tr>
<th>Number of blocks</th>
<th>Number of bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Block 1</td>
<td></td>
</tr>
<tr>
<td>Block 2</td>
<td></td>
</tr>
<tr>
<td>Block N-1</td>
<td></td>
</tr>
<tr>
<td>Block N</td>
<td></td>
</tr>
</tbody>
</table>

Number of Blocks (16 bits)
Number of stored blocks. Stores the number of blocks that can be output within the range specified by the read start position and end position.

Number of Bytes (16 bits)
Stores the number of bytes per block.

Block

The content of the block is the same as that of “Block 1” described in section “2.11.1 Most Recent Channel Data (FData)”.

2.11.3 FIFO Data Read Range (FFifoCur)

The output in response to the command “FFifoCur,1” is shown below. Outputs FIFO data read position information.

<table>
<thead>
<tr>
<th>8 bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional information (always 0)</td>
</tr>
<tr>
<td>The read position of the oldest data in the FIFO (1 to 99999999999)</td>
</tr>
<tr>
<td>The read position of the most recent data in the FIFO (1 to 99999999999)</td>
</tr>
</tbody>
</table>

The read position of the oldest data in the FIFO
This is the oldest data number within the readable data range.

The read position of the most recent data in the FIFO
This is the most recent data number within the readable data range.
2.11.4 The Most Recent Control Data (FCtrlData)

The output in response to the command "FCtrlData,1" is shown below. Outputs the most recent control data per loop.

The figure below shows the structure of the output data. Data is stored in “Block 1.”

<table>
<thead>
<tr>
<th>Number of blocks</th>
<th>Number of bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Block 1**

The figure below shows the structure of block 1.

<table>
<thead>
<tr>
<th>Year</th>
<th>Month</th>
<th>Day</th>
<th>Hour</th>
<th>Min.</th>
<th>Sec.</th>
<th>Millisecond</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Additional information**

<table>
<thead>
<tr>
<th>Loop number</th>
<th>Data</th>
<th>Reserved PVERR</th>
<th>SPERR</th>
<th>Decimal point</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alarm 1</td>
<td></td>
<td>Reserved</td>
<td>PV</td>
<td></td>
</tr>
<tr>
<td>Alarm 2</td>
<td></td>
<td>Reserved</td>
<td>SP</td>
<td>SPERR</td>
</tr>
<tr>
<td>Alarm 3</td>
<td></td>
<td>Reserved</td>
<td>PV</td>
<td></td>
</tr>
<tr>
<td>Alarm 4</td>
<td></td>
<td>Reserved</td>
<td>SP</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>R/S</th>
<th>A/M/C</th>
<th>L/R</th>
<th>R/P/L</th>
<th>AT</th>
<th>SPN</th>
<th>PIDN</th>
<th>PVSEL</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Reserved</th>
<th>PV</th>
<th>OUT</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Reserved</th>
</tr>
</thead>
</table>

***

<table>
<thead>
<tr>
<th>Loop number</th>
<th>Data</th>
<th>Reserved PVERR</th>
<th>SPERR</th>
<th>Decimal point</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alarm 1</td>
<td></td>
<td>Reserved</td>
<td>PV</td>
<td></td>
</tr>
<tr>
<td>Alarm 2</td>
<td></td>
<td>Reserved</td>
<td>SP</td>
<td>SPERR</td>
</tr>
<tr>
<td>Alarm 3</td>
<td></td>
<td>Reserved</td>
<td>PV</td>
<td></td>
</tr>
<tr>
<td>Alarm 4</td>
<td></td>
<td>Reserved</td>
<td>SP</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reserved</th>
<th>PV</th>
<th>OUT</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Reserved</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Reserved</th>
</tr>
</thead>
</table>
2.11 Format of the Data Block of Binary Output

Data Time

<table>
<thead>
<tr>
<th>Item (Number of Bits)</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Year (8 bits)</td>
<td>0 to 99</td>
</tr>
<tr>
<td>Month (8 bits)</td>
<td>1 to 12</td>
</tr>
<tr>
<td>Day (8 bits)</td>
<td>1 to 31</td>
</tr>
<tr>
<td>Hour (8 bits)</td>
<td>0 to 23</td>
</tr>
<tr>
<td>Minute (8 bits)</td>
<td>0 to 59</td>
</tr>
<tr>
<td>Second (8 bits)</td>
<td>0 to 59</td>
</tr>
<tr>
<td>Millisecond (16 bits)</td>
<td>0 to 999</td>
</tr>
</tbody>
</table>

Additional Information (64 bits)
Bit 0: Daylight saving time (0: standard time; 1: daylight saving time)

Loop Number (16 bits)
001 to 692

Data Information (8 bits)
Indicates the data information (or the module status.)
0: Not used, 1: No data, 2: Normal, 3: Luck of data/Module not installed.

PVERR (8 bits)
Indicates the PV error.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 to 4</td>
<td>0</td>
<td>No error</td>
</tr>
<tr>
<td>1</td>
<td>Skip</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>+OVER</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-OVER</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>+Burnout</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>-Burnout</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>A/D error</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Invalid data</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>No A/D calibration value error</td>
</tr>
<tr>
<td>1</td>
<td>A/D calibration value error</td>
<td></td>
</tr>
</tbody>
</table>

SPERR (8 bits)
Indicates the SP error.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 to 4</td>
<td>0</td>
<td>No error</td>
</tr>
<tr>
<td>1</td>
<td>Skip</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>+OVER</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-OVER</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>+Burnout</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>-Burnout</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>A/D error</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Invalid data</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>No A/D calibration value error</td>
</tr>
<tr>
<td>1</td>
<td>A/D calibration value error</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0</td>
<td>No RJC error</td>
</tr>
<tr>
<td>1</td>
<td>RJC error</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>Reserved</td>
</tr>
</tbody>
</table>
### Decimal Point Place (16 bits)
Indicates the decimal point place.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 to 3</td>
<td>0 to 4</td>
<td>PV</td>
</tr>
<tr>
<td>4 to 7</td>
<td>0 to 4</td>
<td>SP</td>
</tr>
<tr>
<td>8 to 11</td>
<td>1</td>
<td>OUT</td>
</tr>
</tbody>
</table>

### Alarm (32 bits)
Indicates the alarm status.
The eight bit values of alarm 1 to alarm 4 are described in the table below.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 to 5</td>
<td>0</td>
<td>No alarm</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>High limit alarm</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Low limit alarm</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Difference high limit alarm</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Difference low limit alarm</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>High limit on rate-of-change alarm</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>Low limit on rate-of-change alarm</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>Delay high limit alarm</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>Delay low limit alarm</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>No alarm is activated.</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>An alarm is activated.</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>Alarm nonhold state</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>Alarm hold state</td>
</tr>
</tbody>
</table>

### R/S (8 bits)
Indicates the RUN or STOP of the control operation.
0: RUN, 1: STOP

### A/M/C (8 bits)
Indicates the AUTO, MANUAL or CASCADE of the control operation.
0: AUTO, 1: MANUAL, 2: CASCADE

### L/R (8 bits)
Indicates the LOCAL or REMOTE of the control operation.
0: LOCAL, 1: REMOTE

### R/P/L (8 bits)
Indicates the RESET, PROGRAM or LOCAL of the control operation.
0: RESET, 1: PROG, 2: LOCAL

### AT (8 bits)
Indicates the On or Off of the Auto tuning.
0: OFF, 1: ON

### SPN (8 bits)
1 to 8: SP group number

### PIDN (8 bits)
1 to 8: PID group number, 9: Reference PID
PVSEL (8 bits)
Indicates the status of PV input.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>ON</td>
<td>AI1 is used for PV1.</td>
</tr>
<tr>
<td>1</td>
<td>ON</td>
<td>EXPV1 is used for PV1.</td>
</tr>
<tr>
<td>2</td>
<td>ON</td>
<td>AI2 is used for PV2.</td>
</tr>
<tr>
<td>3</td>
<td>ON</td>
<td>EXPV2 is used for PV2.</td>
</tr>
<tr>
<td>4</td>
<td>ON</td>
<td>AI2 is used for PV1 (when PV switching is selected.)</td>
</tr>
<tr>
<td>5</td>
<td>ON</td>
<td>EXPV2 is used for PV1 (when PV switching is selected.)</td>
</tr>
</tbody>
</table>

PV (32 bits)
Indicates the PV value using an integer. When the data status is Over or Burnout, the integer will be the value of -5% to 105% of the range.

SP (32 bits)
Indicates the SP value using an integer. When the data status is Over or Burnout, the integer will be the value of -5% to 105% of the range.

OUT (32 bits)
Indicates the OUT value using an integer. However, when the Tight shut function is used, the SD is -S1 and Invalid data is -100.
# Appendix 1 ASCII Character Codes

The ASCII character code table is shown below.

<table>
<thead>
<tr>
<th>Upper 4 Bits</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>SP (space)</td>
<td>0</td>
<td>@</td>
<td>P</td>
<td>'</td>
<td>p</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>!</td>
<td>1</td>
<td>A</td>
<td>Q</td>
<td>a</td>
<td>q</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>&quot;</td>
<td>2</td>
<td>B</td>
<td>R</td>
<td>b</td>
<td>r</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>#</td>
<td>3</td>
<td>C</td>
<td>S</td>
<td>c</td>
<td>s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>$</td>
<td>4</td>
<td>D</td>
<td>T</td>
<td>d</td>
<td>t</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>%</td>
<td>5</td>
<td>E</td>
<td>U</td>
<td>e</td>
<td>u</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>&amp;</td>
<td>6</td>
<td>F</td>
<td>V</td>
<td>f</td>
<td>v</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>'</td>
<td>7</td>
<td>G</td>
<td>W</td>
<td>g</td>
<td>w</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>(</td>
<td>8</td>
<td>H</td>
<td>X</td>
<td>h</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>)</td>
<td>9</td>
<td>I</td>
<td>Y</td>
<td>i</td>
<td>y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>LF (line feed)</td>
<td>*</td>
<td>:</td>
<td>J</td>
<td>Z</td>
<td>j</td>
<td>z</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>ESC</td>
<td>+</td>
<td>:</td>
<td>K</td>
<td>[</td>
<td>k</td>
<td>{</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>.</td>
<td>&lt;</td>
<td>L</td>
<td>\</td>
<td>l</td>
<td>]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>CR (return)</td>
<td>-</td>
<td>=</td>
<td>M</td>
<td>]</td>
<td>m</td>
<td>}</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td>&gt;</td>
<td>N</td>
<td>^</td>
<td>n</td>
<td>~</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td></td>
<td>?</td>
<td>O</td>
<td>_</td>
<td>o</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Characters Used in Commands

In addition to alphanumeric characters, the following characters are used: commas as delimiters, semicolons as sub delimiters, question marks as query symbols, single quotation marks to indicate user-defined character strings, and “CR” (return) “LF” (line feed) as terminators.

## Characters That Can Be Used in User-Defined Character Strings

The table below shows the characters that can be used in user-defined character strings (tags, tag No., messages, etc.).

<table>
<thead>
<tr>
<th>Item</th>
<th>Command and Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Directory name</td>
<td>p1 of the SDirectory command</td>
<td>The characters other than those in blue cells and those in thick frames can be used.</td>
</tr>
<tr>
<td>File name</td>
<td>p2 of the SFilename command</td>
<td>The characters other than those in blue cells and those in thick frames can be used.</td>
</tr>
<tr>
<td>Batch number</td>
<td>p2 of the OBatName command</td>
<td>The characters other than those in blue cells and those in thick frames can be used.</td>
</tr>
<tr>
<td>Password</td>
<td>p5 of SUser, p1, p2, and p3 of OPassword, p2 of CLogin</td>
<td>The characters other than those in blue cells and those in thick frames can be used.</td>
</tr>
<tr>
<td>Character strings that users specify other than those above</td>
<td></td>
<td>The characters other than those in blue cells can be used.</td>
</tr>
</tbody>
</table>
Appendix 2 Login Procedure

To communicate using the general communication feature, you must log in to the recorder from your PC. If you complete the procedure successfully up to “Login complete” in the following figure, you will be able to use the commands.

When Using the Login Function

For Bluetooth, the Bluetooth connection flow chart is inserted here (see appendix 7).

* “E1:402” is returned when the advanced security function (/AS) is enabled.

** E251. If the format of the CLogin command is not correct, verification is not performed, and an error code indicating the error is returned.

The following error code is returned when the advanced security function (/AS) is enabled.
251, 262, 263, 264, 265, 272, 273, 767

When the password management is in use, the following error code is returned in addition to the error code above.
004,252,261,651,657,760,761,762,763,764,765,766,768,769,770,771,772,773,774,775
When Not Using the Login Function

For Bluetooth, the Bluetooth connection flow chart is inserted here (see appendix 7).

Appendix 2  Login Procedure
Appendix 3  Output Flow Chart of External Storage Medium Files and File Lists

Example for Outputting File aaaa.dtd

The flow chart for outputting file aaaa.dtd in the DATA0 directory on the external storage medium is shown below.

START

Send command

FMedia,GET,/DRV0/DATA0/aaaa.dtd,0,-1

Receive response

Is there more data?
(1)

Obtain the number of received bytes.
(2)

If the binary header flag (bit 0) is 0, there is more data. If it is 1, data transmission is complete.

From the data length of the binary output common header, obtain the number of received bytes.

Sum the number of received bytes, compute the unread position.

Compute the unread position.
(3)

Send command

FMedia,GET,/DRV0/DATA0/aaaa.dtd,[unread position],-1

Receive response

End
Example for Outputting a File List

The flow chart for outputting the list of files in the DATA0 directory on the external storage medium is shown below.

```
START

Send command

FMedia,DIR,/DRV0/DATA0/,0,-1

Receive response

Are there more file lists? (1)

Obtain the number of received files (2)

Compute the unread position. (3)

Send command

FMedia,DIR,/DRV0/DATA0/,unread position,-1

Receive response

(1) If there are no more files, EA[crlf]EN[crlf] is output.

(2) Count the number of lines in the output file list.

(3) Sum the number of lines in the file lists to compute the unread position.
```
Appendix 4 FIFO Data Output Flow Chart

Overview of the FIFO Buffer
The recorder internal memory is equipped with a dedicated FIFO (First-In-First-Out) buffer for outputting measured data. Measured data is written to the buffer at every scan interval. The PC can continuously retrieve the most recent measured data from the FIFO buffer. The size of the internal memory allocated for the FIFO buffer varies depending on the model. The number of data entries that the FIFO buffer can store varies depending on the number of channels and scan interval. The number of data entries that the FIFO buffer can store and the data length can be determined with the following formula.

\[
\text{Data entries} = \frac{2000000}{16 + (12 \times \text{[number of channels]})} \quad \text{(fractions truncated)}
\]

\[
\text{Data length} = \text{[data entries]} \times \text{[scan interval]}
\]

Example
If there are 10 I/O channels, 10 math channels, and 10 communication channels, and the scan interval is 100 ms, the number data entries will be 5319, and the data length will be 531.9 seconds or 8.865 minutes.

Example of FIFO Buffer Operation
The following example shows the case when the scan interval is 1 second and the FIFO internal memory size is for 8 scan intervals.

Writing of Measured Data in the FIFO Buffer
Writing to the FIFO buffer takes place every scan interval. If measured data is written to block 8, the most recent value will be written to block 1 in the next scan interval, overwriting the old value. This is called FIFO wraparound.

On the other hand, the most recent write position is managed using serial numbers starting with 1. The serial number does not return to 1 even when a FIFO wraparound occurs.

Reading Measured Data
The \text{FFifoCur,0} command is used to read measured data. The read start position and read end position are specified using serial numbers. You can use the \text{FFifoCur,1} command to read the serial numbers for the positions that data can be read from.
Appendix  5  Check Sum Calculation Method

The check sum of binary data is calculated using an algorithm like the one shown below.

```c
int CalcSum(unsigned char *buf, int len)
{
    int    odd;
    unsigned long    sum;
    unsigned char    *p;

    sum = 0;
    odd = len & 1;
    len >>= 1;

    for (p = buf ; len ; len --, p += 2)
    {
        sum += (*p << 8) | *(p + 1);
    }
    if (odd) sum += (*p << 8);

    sum = (sum & 0xffff) + ((sum >> 16) & 0xffff);
    if (sum > 0xffff) sum = sum - 0xffff;

    return ((~sum) & 0xffff);
}
```
### Parameters p5 (Data Group Name) and p6 (Data Name) for SWattData Command

#### WT1800

<table>
<thead>
<tr>
<th>p5 (Data group name)</th>
<th>p6 (Data name)</th>
<th>Description</th>
<th>WT Function mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Off</td>
<td>L</td>
<td>Data assignment is disabled.</td>
<td>-</td>
</tr>
<tr>
<td>ELEMENT1 to ELEMENT5</td>
<td>URMS</td>
<td>True rms voltage</td>
<td>Urms</td>
</tr>
<tr>
<td></td>
<td>UMN</td>
<td>Rectified mean voltage calibrated to the rms value</td>
<td>Umn</td>
</tr>
<tr>
<td></td>
<td>UDC</td>
<td>Simple voltage average</td>
<td>Udc</td>
</tr>
<tr>
<td></td>
<td>IRMS</td>
<td>True rms current</td>
<td>Irms</td>
</tr>
<tr>
<td></td>
<td>IMN</td>
<td>Rectified mean current calibrated to the rms value</td>
<td>Imn</td>
</tr>
<tr>
<td></td>
<td>IDC</td>
<td>Simple current average</td>
<td>Idc</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>Active power</td>
<td>P</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>Apparent power</td>
<td>S</td>
</tr>
<tr>
<td></td>
<td>Q</td>
<td>Reactive power</td>
<td>Q</td>
</tr>
<tr>
<td></td>
<td>LAMBda</td>
<td>Power factor</td>
<td>λ</td>
</tr>
<tr>
<td></td>
<td>PHI</td>
<td>Phase difference</td>
<td>φ</td>
</tr>
<tr>
<td></td>
<td>FU</td>
<td>Voltage frequency</td>
<td>Fu</td>
</tr>
<tr>
<td></td>
<td>FI</td>
<td>Current frequency</td>
<td>fi</td>
</tr>
<tr>
<td></td>
<td>TIME</td>
<td>Integration time</td>
<td>Time</td>
</tr>
<tr>
<td></td>
<td>WH</td>
<td>Sum of watt hours</td>
<td>WP</td>
</tr>
<tr>
<td></td>
<td>WHP</td>
<td>Sum of positive P (consumed watt hours)</td>
<td>WP+</td>
</tr>
<tr>
<td></td>
<td>WHM</td>
<td>Sum of negative P (watt hours returned to the power supply)</td>
<td>WP-</td>
</tr>
<tr>
<td></td>
<td>AH</td>
<td>Sum of positive and negative ampere hours</td>
<td>q</td>
</tr>
<tr>
<td></td>
<td>AHP</td>
<td>Sum of positive I (ampere hours)</td>
<td>q+</td>
</tr>
<tr>
<td></td>
<td>AHM</td>
<td>Sum of negative I (ampere hours)</td>
<td>q-</td>
</tr>
<tr>
<td>ElemHrm1 to ElemHrm6</td>
<td>UK_1</td>
<td>RMS voltage of harmonic order 1</td>
<td>U(1)</td>
</tr>
<tr>
<td></td>
<td>UK_T</td>
<td>Rms voltage</td>
<td>U(Total)</td>
</tr>
<tr>
<td></td>
<td>IK_1</td>
<td>RMS current of harmonic order 1</td>
<td>I(1)</td>
</tr>
<tr>
<td></td>
<td>IK_T</td>
<td>Rms current</td>
<td>I(Total)</td>
</tr>
<tr>
<td></td>
<td>UTHD</td>
<td>Ratio of the total harmonic voltage to U(1) or U(Total)</td>
<td>Uthd</td>
</tr>
<tr>
<td></td>
<td>ITHD</td>
<td>Ratio of the total harmonic current to I(1) or I(Total)</td>
<td>Ithd</td>
</tr>
<tr>
<td>SigmaA to SigmaC</td>
<td>URMS</td>
<td>True rms voltage</td>
<td>Urms Σ</td>
</tr>
<tr>
<td></td>
<td>UMN</td>
<td>Rectified mean voltage calibrated to the rms value</td>
<td>Umn Σ</td>
</tr>
<tr>
<td></td>
<td>IRMS</td>
<td>True rms current</td>
<td>Irms Σ</td>
</tr>
<tr>
<td></td>
<td>IMN</td>
<td>Rectified mean current calibrated to the rms value</td>
<td>Imn Σ</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>Active power</td>
<td>P Σ</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>Apparent power</td>
<td>S Σ</td>
</tr>
<tr>
<td></td>
<td>LAMBda</td>
<td>Power factor</td>
<td>λΣ</td>
</tr>
<tr>
<td></td>
<td>PHI</td>
<td>Phase difference</td>
<td>φΣ</td>
</tr>
<tr>
<td></td>
<td>WH</td>
<td>Sum of positive and negative watt hours</td>
<td>WP Σ</td>
</tr>
<tr>
<td></td>
<td>WHP</td>
<td>Sum of positive P (consumed watt hours)</td>
<td>WP+ Σ</td>
</tr>
<tr>
<td></td>
<td>WHM</td>
<td>Sum of negative P (watt hours returned to the power supply)</td>
<td>WP- Σ</td>
</tr>
<tr>
<td></td>
<td>AH</td>
<td>Sum of positive and negative ampere hours</td>
<td>q Σ</td>
</tr>
<tr>
<td></td>
<td>AHP</td>
<td>Sum of positive I (ampere hours)</td>
<td>q+ Σ</td>
</tr>
<tr>
<td></td>
<td>AHM</td>
<td>Sum of negative I (ampere hours)</td>
<td>q- Σ</td>
</tr>
</tbody>
</table>

Continued on next page
## Appendix 6  Data Group Name and Data Name for WT Communication

<table>
<thead>
<tr>
<th>Data group name</th>
<th>Data name</th>
<th>Description</th>
<th>WT Function mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Other</td>
<td>ETA1</td>
<td>Efficiency 1</td>
<td>η 1</td>
</tr>
<tr>
<td></td>
<td>ETA2</td>
<td>Efficiency 2</td>
<td>η 2</td>
</tr>
<tr>
<td></td>
<td>ETA3</td>
<td>Efficiency 3</td>
<td>η 3</td>
</tr>
<tr>
<td></td>
<td>ETA4</td>
<td>Efficiency 4</td>
<td>η 4</td>
</tr>
<tr>
<td></td>
<td>F1</td>
<td>User-defined function 1</td>
<td>F1</td>
</tr>
<tr>
<td></td>
<td>F2</td>
<td>User-defined function 2</td>
<td>F2</td>
</tr>
<tr>
<td></td>
<td>F3</td>
<td>User-defined function 3</td>
<td>F3</td>
</tr>
<tr>
<td></td>
<td>F4</td>
<td>User-defined function 4</td>
<td>F4</td>
</tr>
<tr>
<td></td>
<td>F5</td>
<td>User-defined function 5</td>
<td>F5</td>
</tr>
<tr>
<td></td>
<td>F6</td>
<td>User-defined function 6</td>
<td>F6</td>
</tr>
<tr>
<td></td>
<td>F7</td>
<td>User-defined function 7</td>
<td>F7</td>
</tr>
<tr>
<td></td>
<td>F8</td>
<td>User-defined function 8</td>
<td>F8</td>
</tr>
<tr>
<td></td>
<td>F9</td>
<td>User-defined function 9</td>
<td>F9</td>
</tr>
<tr>
<td></td>
<td>F10</td>
<td>User-defined function 10</td>
<td>F10</td>
</tr>
<tr>
<td></td>
<td>F11</td>
<td>User-defined function 11</td>
<td>F11</td>
</tr>
<tr>
<td></td>
<td>F12</td>
<td>User-defined function 12</td>
<td>F12</td>
</tr>
<tr>
<td></td>
<td>F13</td>
<td>User-defined function 13</td>
<td>F13</td>
</tr>
<tr>
<td></td>
<td>F14</td>
<td>User-defined function 14</td>
<td>F14</td>
</tr>
<tr>
<td></td>
<td>F15</td>
<td>User-defined function 15</td>
<td>F15</td>
</tr>
<tr>
<td></td>
<td>F16</td>
<td>User-defined function 16</td>
<td>F16</td>
</tr>
<tr>
<td></td>
<td>F17</td>
<td>User-defined function 17</td>
<td>F17</td>
</tr>
<tr>
<td></td>
<td>F18</td>
<td>User-defined function 18</td>
<td>F18</td>
</tr>
<tr>
<td>DeltaA to DeltaC</td>
<td>DU1</td>
<td>Delta computation voltage 1</td>
<td>Δ U1</td>
</tr>
<tr>
<td></td>
<td>DU2</td>
<td>Delta computation voltage 2</td>
<td>Δ U2</td>
</tr>
<tr>
<td></td>
<td>DU3</td>
<td>Delta computation voltage 3</td>
<td>Δ U3</td>
</tr>
<tr>
<td></td>
<td>DUS</td>
<td>Delta computation wiring voltage</td>
<td>Δ U Σ</td>
</tr>
<tr>
<td></td>
<td>DI</td>
<td>Delta computation current</td>
<td>Δ I</td>
</tr>
<tr>
<td></td>
<td>DP1</td>
<td>Delta computation power 1</td>
<td>Δ P1</td>
</tr>
<tr>
<td></td>
<td>DP2</td>
<td>Delta computation power 2</td>
<td>Δ P2</td>
</tr>
<tr>
<td></td>
<td>DP3</td>
<td>Delta computation power 3</td>
<td>Δ P3</td>
</tr>
<tr>
<td></td>
<td>DPS</td>
<td>Delta computation wiring power</td>
<td>Δ P Σ</td>
</tr>
<tr>
<td>Motor</td>
<td>SPEED</td>
<td>Motor rotating speed</td>
<td>Speed</td>
</tr>
<tr>
<td></td>
<td>TORQUE</td>
<td>Motor torque</td>
<td>Torque</td>
</tr>
<tr>
<td></td>
<td>SYNCSP</td>
<td>Synchronous speed</td>
<td>SyncSp</td>
</tr>
<tr>
<td></td>
<td>SLIP</td>
<td>Slip (%)</td>
<td>Slip</td>
</tr>
<tr>
<td></td>
<td>PM</td>
<td>Mechanical output of the motor (mechanical power)</td>
<td>Pm</td>
</tr>
<tr>
<td>Aux</td>
<td>Aux1</td>
<td>Mechanical output of the motor</td>
<td>Pm</td>
</tr>
<tr>
<td></td>
<td>Aux2</td>
<td>Auxiliary input 1</td>
<td>Aux1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Auxiliary input 2</td>
<td>Aux2</td>
</tr>
</tbody>
</table>

1  Will become ΣA, ΣB, or ΣC depending on the WT1800 wiring type.
## WT500

<table>
<thead>
<tr>
<th>Data group name</th>
<th>Data name</th>
<th>Description</th>
<th>WT Function mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Off</td>
<td>-</td>
<td>Data assignment is disabled.</td>
<td>–</td>
</tr>
<tr>
<td>ELEMENT to ELEMENT3</td>
<td>URMS</td>
<td>True rms voltage</td>
<td>U rms</td>
</tr>
<tr>
<td></td>
<td>UMN</td>
<td>Rectified mean voltage calibrated to the rms value</td>
<td>U mn</td>
</tr>
<tr>
<td></td>
<td>UDC</td>
<td>Simple voltage average</td>
<td>U dc</td>
</tr>
<tr>
<td></td>
<td>URMN</td>
<td>Rectified mean voltage</td>
<td>U rmn</td>
</tr>
<tr>
<td></td>
<td>UAC</td>
<td>AC component</td>
<td>U ac</td>
</tr>
<tr>
<td></td>
<td>IMN</td>
<td>Rectified mean current calibrated to the rms value</td>
<td>I mn</td>
</tr>
<tr>
<td></td>
<td>IDC</td>
<td>Simple current average</td>
<td>I dc</td>
</tr>
<tr>
<td></td>
<td>IRMN</td>
<td>Rectified mean current</td>
<td>I rmn</td>
</tr>
<tr>
<td></td>
<td>IAC</td>
<td>AC component</td>
<td>I ac</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>Active power</td>
<td>P</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>Apparent power</td>
<td>S</td>
</tr>
<tr>
<td></td>
<td>Q</td>
<td>Reactive power</td>
<td>Q</td>
</tr>
<tr>
<td></td>
<td>LAMBda</td>
<td>Power factor</td>
<td>λ</td>
</tr>
<tr>
<td></td>
<td>PHI</td>
<td>Phase difference</td>
<td>φ</td>
</tr>
<tr>
<td></td>
<td>FU</td>
<td>Voltage frequency</td>
<td>f U</td>
</tr>
<tr>
<td></td>
<td>FI</td>
<td>Current frequency</td>
<td>f I</td>
</tr>
<tr>
<td></td>
<td>UPP</td>
<td>Maximum voltage</td>
<td>U+ pk</td>
</tr>
<tr>
<td></td>
<td>UMP</td>
<td>Minimum voltage</td>
<td>U- pk</td>
</tr>
<tr>
<td></td>
<td>IPP</td>
<td>Maximum current</td>
<td>I+ pk</td>
</tr>
<tr>
<td></td>
<td>IMP</td>
<td>Minimum current</td>
<td>I- pk</td>
</tr>
<tr>
<td></td>
<td>CFU</td>
<td>Voltage crest factor</td>
<td>CFU</td>
</tr>
<tr>
<td></td>
<td>CFI</td>
<td>Current crest factor</td>
<td>CFI</td>
</tr>
<tr>
<td>TIME</td>
<td>Integration time</td>
<td>Time</td>
<td></td>
</tr>
<tr>
<td>WH</td>
<td>Sum of positive and negative watt hours</td>
<td>WP</td>
<td></td>
</tr>
<tr>
<td>WHP</td>
<td>Sum of positive P (consumed watt hours)</td>
<td>WP+</td>
<td></td>
</tr>
<tr>
<td>WHM</td>
<td>Sum of negative P (watt hours returned to the power supply)</td>
<td>WP-</td>
<td></td>
</tr>
<tr>
<td>AH</td>
<td>Sum of positive and negative ampere hours</td>
<td>q</td>
<td></td>
</tr>
<tr>
<td>AHP</td>
<td>Sum of positive I (ampere hours)</td>
<td>q+</td>
<td></td>
</tr>
<tr>
<td>AHM</td>
<td>Sum of negative I (ampere hours)</td>
<td>q-</td>
<td></td>
</tr>
<tr>
<td>WS</td>
<td>Volt-ampere hours</td>
<td>WS</td>
<td></td>
</tr>
<tr>
<td>WQ</td>
<td>Var hours</td>
<td>WQ</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ElemHrm1 to ElemHrm3</th>
<th>UK_0</th>
<th>Rms voltage of harmonic order 0</th>
<th>U(0)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>UK_1</td>
<td>Rms voltage of harmonic order 1</td>
<td>U(1)</td>
</tr>
<tr>
<td></td>
<td>UK_T</td>
<td>Rms voltage</td>
<td>U(Total)</td>
</tr>
<tr>
<td></td>
<td>IK_0</td>
<td>Rms current of harmonic order 0</td>
<td>I(0)</td>
</tr>
<tr>
<td></td>
<td>IK_1</td>
<td>Rms current of harmonic order 1</td>
<td>I(1)</td>
</tr>
<tr>
<td></td>
<td>IK_T</td>
<td>Rms current</td>
<td>I(Total)</td>
</tr>
<tr>
<td></td>
<td>PK_0</td>
<td>Active power of harmonic order 0</td>
<td>P(0)</td>
</tr>
<tr>
<td></td>
<td>PK_1</td>
<td>Active power of harmonic order 1</td>
<td>P(1)</td>
</tr>
<tr>
<td></td>
<td>PK_T</td>
<td>Active power</td>
<td>P(Total)</td>
</tr>
<tr>
<td></td>
<td>SK_0</td>
<td>Apparent power of harmonic order 0</td>
<td>S(0)</td>
</tr>
<tr>
<td></td>
<td>SK_1</td>
<td>Apparent power of harmonic order 1</td>
<td>S(1)</td>
</tr>
<tr>
<td></td>
<td>SK_T</td>
<td>Total apparent power</td>
<td>S(Total)</td>
</tr>
<tr>
<td></td>
<td>QK_0</td>
<td>Reactive power of harmonic order 0</td>
<td>Q(0)</td>
</tr>
<tr>
<td></td>
<td>QK_1</td>
<td>Reactive power of harmonic order 1</td>
<td>Q(1)</td>
</tr>
<tr>
<td></td>
<td>QK_T</td>
<td>Total reactive power</td>
<td>Q(Total)</td>
</tr>
<tr>
<td></td>
<td>LAMBDA0</td>
<td>Power factor of harmonic order 0</td>
<td>λ (0)</td>
</tr>
<tr>
<td></td>
<td>LAMBDA1</td>
<td>Power factor of harmonic order 1</td>
<td>λ (1)</td>
</tr>
<tr>
<td></td>
<td>LAMBDAT</td>
<td>Total power factor</td>
<td>λ (Total)</td>
</tr>
<tr>
<td></td>
<td>PHI1</td>
<td>Phase difference between the voltage and current of harmonic order 1</td>
<td>φ (1)</td>
</tr>
<tr>
<td></td>
<td>PHI_T</td>
<td>Total phase difference</td>
<td>φ (Total)</td>
</tr>
</tbody>
</table>

Continued on next page
## Appendix 6  Data Group Name and Data Name for WT Communication

<table>
<thead>
<tr>
<th>Data group name</th>
<th>Data name</th>
<th>Description</th>
<th>WT Function mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>ElemHrm1 to ElemHrm3</td>
<td>PHIuK3</td>
<td>Phase difference between harmonic voltage U(3) and the fundamental signal U(1).</td>
<td>φ U(3)</td>
</tr>
<tr>
<td></td>
<td>PHIiK3</td>
<td>Phase difference between harmonic current I(3) and the fundamental signal I(1).</td>
<td>φ I(3)</td>
</tr>
<tr>
<td></td>
<td>UTHD</td>
<td>Ratio of the total harmonic voltage to U(1) or U(Total)</td>
<td>Uthd</td>
</tr>
<tr>
<td></td>
<td>ITHD</td>
<td>Ratio of the total harmonic current to I(1) or I(Total)</td>
<td>Ithd</td>
</tr>
<tr>
<td></td>
<td>PTHD</td>
<td>Ratio of the total harmonic active power to P(1) or P(Total)</td>
<td>Pthd</td>
</tr>
</tbody>
</table>

### SigmaA
- URMS: True rms voltage \( U_{rms} \)
- UMN: Rectified mean voltage calibrated to the rms value \( U_{mn} \)
- UDC: Simple voltage average \( U_{dc} \)
- URMN: Rectified mean voltage \( U_{rmn} \)
- UAC: AC component \( U_{ac} \)
- IRMS: True rms current \( I_{rms} \)
- IMN: Rectified mean current calibrated to the rms value \( I_{mn} \)
- IDC: Simple current average \( I_{dc} \)
- IMRN: Rectified mean current \( I_{rmn} \)
- IAC: AC component \( I_{ac} \)
- P: Active power \( P \)
- S: Apparent power \( S \)
- Q: Reactive power \( Q \)
- LAMBDa: Power factor \( \lambda \)
- PHI: Phase difference \( \phi \)
- WH: Sum of positive and negative watt hours \( WP \)
- WHP: Sum of positive P (consumed watt hours) \( WP^+ \)
- WHM: Sum of negative P (watt hours returned to the power supply) \( WP^- \)
- AH: Sum of positive and negative ampere hours \( q_+ \)
- AHP: Sum of positive I (ampere hours) \( q_+ \)
- AHM: Sum of negative I (ampere hours) \( q_- \)
- WS: Integrated value of SΣ \( WS \)
- WQ: Integrated value of QΣ \( WQ \)

### Other
- ETA1: Efficiency 1 \( \eta_1 \)
- ETA2: Efficiency 2 \( \eta_2 \)
- F1: User-defined function 1 \( F_1 \)
- F2: User-defined function 2 \( F_2 \)
- F3: User-defined function 3 \( F_3 \)
- F4: User-defined function 4 \( F_4 \)
- F5: User-defined function 5 \( F_5 \)
- F6: User-defined function 6 \( F_6 \)
- F7: User-defined function 7 \( F_7 \)
- F8: User-defined function 8 \( F_8 \)

### Delta
- DELTA1: Delta computation 1 \( \Delta F_1 \)
- DELTA2: Delta computation 2 \( \Delta F_2 \)
- DELTA3: Delta computation 3 \( \Delta F_3 \)
- DELTA4: Delta computation 4 \( \Delta F_4 \)

### Phase
- PHI_UU2: The phase difference between the fundamental voltage of element 1, U1(1), and the fundamental voltage of element 2, U2(1) \( \phi U1-U2 \)
- PHI_UU3: The phase difference between the fundamental voltage of element 1, U1(1), and the fundamental voltage of element 3, U3(1) \( \phi U1-U3 \)
- PHI_UU1: The phase difference between the fundamental voltage of element 1, U1(1), and the fundamental current of element 1, I1(1) \( \phi U1-I1 \)

Continued on next page
### Appendix 6  Data Group Name and Data Name for WT Communication

#### WT300

<table>
<thead>
<tr>
<th>Data group name</th>
<th>Data name</th>
<th>Description</th>
<th>WT Function mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Off</td>
<td>-</td>
<td>Data assignment is disabled.</td>
<td>–</td>
</tr>
<tr>
<td>ELEMENT1 to ELEMENT3</td>
<td>U</td>
<td>voltage</td>
<td>U</td>
</tr>
<tr>
<td></td>
<td>I</td>
<td>current</td>
<td>I</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>active power</td>
<td>P</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>apparent power</td>
<td>S</td>
</tr>
<tr>
<td></td>
<td>Q</td>
<td>reactive power</td>
<td>Q</td>
</tr>
<tr>
<td></td>
<td>LAMBda</td>
<td>power factor</td>
<td>λ</td>
</tr>
<tr>
<td></td>
<td>PHI</td>
<td>phase difference</td>
<td>φ</td>
</tr>
<tr>
<td></td>
<td>FU</td>
<td>voltage frequency</td>
<td>fU</td>
</tr>
<tr>
<td></td>
<td>FI</td>
<td>current frequency</td>
<td>fI</td>
</tr>
<tr>
<td></td>
<td>UPPPeak</td>
<td>Maximum voltage</td>
<td>U+pk</td>
</tr>
<tr>
<td></td>
<td>UMPPeak</td>
<td>Minimum voltage</td>
<td>U-pk</td>
</tr>
<tr>
<td></td>
<td>IPPPeak</td>
<td>Maximum current</td>
<td>I+pk</td>
</tr>
<tr>
<td></td>
<td>IMPPeak</td>
<td>Minimum current</td>
<td>I-pk</td>
</tr>
<tr>
<td></td>
<td>PPPPeak</td>
<td>Maximum active power</td>
<td>P+pk</td>
</tr>
<tr>
<td></td>
<td>PMPeak</td>
<td>Minimum active power</td>
<td>P-pk</td>
</tr>
<tr>
<td>TIME1</td>
<td>Integration time</td>
<td>Time</td>
<td></td>
</tr>
<tr>
<td>WH</td>
<td>sum of watt hours</td>
<td>WP</td>
<td></td>
</tr>
<tr>
<td>WHP</td>
<td>Sum of positive P (consumed watt hours)</td>
<td>WP+</td>
<td></td>
</tr>
<tr>
<td>WHM</td>
<td>Sum of negative P (watt hours returned to the power supply)</td>
<td>WP-</td>
<td></td>
</tr>
<tr>
<td>AH</td>
<td>Sum of positive and negative ampere hours</td>
<td>q</td>
<td></td>
</tr>
<tr>
<td>AHP</td>
<td>Sum of positive I (ampere hours)</td>
<td>q+</td>
<td></td>
</tr>
<tr>
<td>AHM</td>
<td>Sum of negative I (ampere hours)</td>
<td>q-</td>
<td></td>
</tr>
<tr>
<td>ElemHrm1 to ElemHrm3</td>
<td>UK_1</td>
<td>RMS voltage of harmonic order 1</td>
<td>U(1)</td>
</tr>
<tr>
<td></td>
<td>UK_T</td>
<td>Rms voltage</td>
<td>U(Total)</td>
</tr>
<tr>
<td></td>
<td>UK_1</td>
<td>RMS current of harmonic order 1</td>
<td>I(1)</td>
</tr>
<tr>
<td></td>
<td>UK_T</td>
<td>Rms current</td>
<td>I(Total)</td>
</tr>
<tr>
<td></td>
<td>PK_1</td>
<td>Active power of harmonic order 1</td>
<td>P(1)</td>
</tr>
<tr>
<td></td>
<td>PK_T</td>
<td>Active power</td>
<td>P(Total)</td>
</tr>
<tr>
<td></td>
<td>LAMBDA1</td>
<td>Power factor of harmonic order 1</td>
<td>λ (1)</td>
</tr>
<tr>
<td>PHIK_1</td>
<td>Phase difference between the voltage and current of harmonic order 1</td>
<td>ϕ (1)</td>
<td></td>
</tr>
<tr>
<td>PHIK3</td>
<td>Phase difference between harmonic voltage U(3) and the fundamental signal U(1).</td>
<td>ϕ U(3)</td>
<td></td>
</tr>
<tr>
<td>PHIk3</td>
<td>Phase difference between harmonic current I(3) and the fundamental signal I(1).</td>
<td>ϕ I(3)</td>
<td></td>
</tr>
<tr>
<td>UTHD</td>
<td>Ratio of the total harmonic voltage to U(1) or U(Total)</td>
<td>Uthd</td>
<td></td>
</tr>
<tr>
<td>ITHD</td>
<td>Ratio of the total harmonic current to I(1) or I(Total)</td>
<td>Ithd</td>
<td></td>
</tr>
<tr>
<td>Uhdf_1</td>
<td>relative harmonic content of harmonic voltage of order 1</td>
<td>Uhdf(1)</td>
<td></td>
</tr>
<tr>
<td>ihdf_1</td>
<td>relative harmonic content of harmonic current of order 1</td>
<td>ihdf(1)</td>
<td></td>
</tr>
<tr>
<td>Phdf_1</td>
<td>relative harmonic content of harmonic power of order 1</td>
<td>Phdf(1)</td>
<td></td>
</tr>
<tr>
<td>FPLL2</td>
<td>Current frequency or voltage frequency of PLL source</td>
<td>FPLL</td>
<td></td>
</tr>
</tbody>
</table>

Continued on next page
### Appendix 6  Data Group Name and Data Name for WT Communication

<table>
<thead>
<tr>
<th>Data group name</th>
<th>Data name</th>
<th>Description</th>
<th>WT Function mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>SigmaA</td>
<td>U</td>
<td>voltage</td>
<td>U Σ</td>
</tr>
<tr>
<td></td>
<td>I</td>
<td>current</td>
<td>I Σ</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>active power</td>
<td>P Σ</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>apparent power</td>
<td>S Σ</td>
</tr>
<tr>
<td></td>
<td>Q</td>
<td>reactive power</td>
<td>Q Σ</td>
</tr>
<tr>
<td></td>
<td>LAMBda</td>
<td>power factor</td>
<td>λΣ</td>
</tr>
<tr>
<td></td>
<td>PHI</td>
<td>phase difference</td>
<td>φΣ</td>
</tr>
<tr>
<td></td>
<td>WH</td>
<td>Sum of positive and negative watt hours</td>
<td>WP Σ</td>
</tr>
<tr>
<td></td>
<td>WHP</td>
<td>Sum of positive P (consumed watt hours)</td>
<td>WP+ Σ</td>
</tr>
<tr>
<td></td>
<td>WHM</td>
<td>Sum of negative P (watt hours returned to the power supply)</td>
<td>WP- Σ</td>
</tr>
<tr>
<td></td>
<td>AH</td>
<td>Sum of positive and negative ampere hours</td>
<td>q Σ</td>
</tr>
<tr>
<td></td>
<td>AHP</td>
<td>Sum of positive I (ampere hours)</td>
<td>q+ Σ</td>
</tr>
<tr>
<td></td>
<td>AHM</td>
<td>Sum of negative I (ampere hours)</td>
<td>q- Σ</td>
</tr>
<tr>
<td>Other</td>
<td>MATH</td>
<td>Computed value, such as efficiency</td>
<td>Math</td>
</tr>
</tbody>
</table>

1. "TIME" is valid only when the data group is "ELEMENT1."
2. "PFLL" is valid only when the data group is "ElemHrm1."
Appendix 7  Bluetooth Communication Connection Flow Chart

To compose the complete Bluetooth communication connection flow chart, in appendix 2, insert the following flow chart after “Connect” in the flow chart shown under “When Using the Login Function” when the communication login function is in use or “When Not Using the Login Function” when the function is not in use.

1 A Bluetooth password is required when the first terminal tries to establish a connection when the password usage is enabled.
2 If no input is received within 2 minutes of a password input request (E403), Bluetooth communication will be disconnected.
3 If an error occurs during the CBTConnect command check (the number of parameters, whether the command is a query, etc.), the flow chart sequence follows the same path as when the password verification fails.
4 The above sequence between “Addition start” and “Addition end” is not recorded in the general log.