WR1000 SERIES

THERMAL ARRAYCORDER

Command Reference Manual

MANUAL NO. WR1000-UM-352

GRAPHTEC

CONTENTS

1. SUMMARY OF PROGRAMMING 1-1

1.1 Overview of Basic Program Formats 1-2
1.2 Format of the Program Message 1-3
1.3 Format of the Program Message Unit 1-4
1.4 Format of the Return Message 1-5
1.5 Format of the Return Message Unit 1-6
1.6 The Block Data Format 1-7
1.7 Precautions on Issuing a Query (for the Recorder's Settings) 1-8
1.8 Precautions on the Transfer of Messages 1-8
1.9 Precautions on the IFC Timing When Using the GP-IB Interface 1-8

2. PROCEDURES FOR WRITING COMMANDS 2-1

2.1 Writing Commands 2-2
2.1.1 Types of Command Headers 2-2
2.1.2 Writing Program Messages 2-2
2.1.3 Rules for Interpreting Command Headers 2-3
2.1.4 Rules for Interpreting Program Data 2-4
2.2 Replies to Queries 2-5
2.2.1 Higher-Order Queries 2-5
2.2.2 Output Format for the Return Message 2-5
2.3 The Handling of Program Data 2-6
2.3.1 Numeric Data (Decimal Numerals) 2-6
2.3.2 Dimensional Numeric Data 2-6
2.3.3 Alphabetic (Mnemonic) Data 2-7
2.3.4 Logical-Value Data 2-7
2.3.5 Character-String Data 2-7
2.3.6 Registers 2-8
2.3.7 Block Data 2-8

3. COMMAND DESCRIPTION BY FUNCTIONAL GROUP 3-1

3.1 Input Amp Settings (AMP Group) 3-2
3.1.1 Summary of AMP Group Commands 3-3
3.1.2 Functions and Settings by Amp Type 3-5
3.2 Logic Amp Settings (LOGIC Group) 3-6
3.2.1 Summary of LOGIC Group Commands 3-6
3.3 Settings Related to Measurement and Control (MEASURE Group) 3-7
3.3.1 Summary of MEASURE Group Commands 3-7
3.4 Memory-Related Settings (MEMORY Group) 3-8
3.4.1 Summary of MEMORY Group Commands 3-9
3.4.2 Program Data Settings for MEMORY Group Settings 3-12
3.4.2.1 Settings for Saving Data to Disk 3-12
3.4.2.2 Settings and Replies Related to the Sampling Interval 3-13
3.5 REPLAY Settings (REPLAY Group) 3-14
3.5.1 Summary of REPLAY Group Commands 3-15
3.6 Trigger-Related Settings (TRIGGER Group) 3-19
3.6.1 Summary of TRIGGER Group 1/2 Commands 3-19
3.6.2 Summary of TRIGGER Group 2/2 Commands 3-21

3.7 Settings Related to Recording (RECORD Group) 3-26
3.7.1 Summary of RECORD Group Commands 3-26
3.7.2 Settings and Replies Related to the Chart Speed 3-28
3.8 Settings Related to Logging (LOGGING Group) 3-29
3.8.1 Summary of LOGGING Group Commands 3-29
3.9 Annotation Settings (ANNOTATION Group) 3-30
3.9.1 Summary of ANNOTATION Group Commands 3-30
3.9.2 Supplementary Information on Annotation Printing 3-32
3.9.2.1 Setting the Annotation Functions 3-32
3.9.2.2 Printing Control Via the Interface 3-33
3.10 System Settings (SYSTEM Group) 3-34
3.10.1 Summary of SYSTEM Group Commands 3-34
3.11 Printer Settings (PRINTER Group) 3-35
3.11.1 Summary of PRINTER Group Commands 3-35
3.12 Common Command Settings (COMMON Group) 3-36
3.12.1 Summary of COMMON Group Commands 3-36
3.13 Interface Status Control (STATus Group) 3-39
3.13.1 Summary of STATUS Group Commands 3-39

4. OUTPUTTING THE RECORDER'S STATUS INFORMATION 4-1
4.1 The Status Byte Register 4-2
4.2 The Status Byte 4-4
4.3 The Standard Event Register 4-5
4.4 The Extended Event Register 4-6
4.4.1 Description of the Extended Event Register's Bits 4-7
4.4.2 Sample Operations 4-8
4.5 The Error Queue 4-11
4.6 The Output Queue 4-12

5. DATA OUTPUT FORMATS 5-1
5.1 Procedure for Outputting Data 5-2
5.2 Specifying the Source Setting for Output Data 5-2
5.3 Output Formats 5-3

APPENDIX A. SPECIFICATIONS OF THE RS-232C CONNECTOR A-1
A.1 Compatible Interface Cable A-2
A.2 Pin Assignment of the RS-232C Connector A-2

APPENDIX B. ERROR CODES B-1
B.1 ERROR CODES B-2

CHAPTER 1

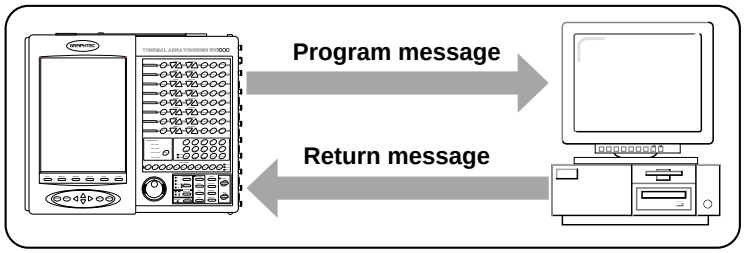
SUMMARY OF PROGRAMMING

This chapter provides a summary of programming for controlling the WR1000 from an external device, such as a personal computer.

1.1 Overview of Basic Program Formats
1.2 Format of the Program Message
1.3 Format of the Program Message Unit
1.4 Format of the Return Message
1.5 Format of the Return Message Unit
1.6 The Block Data Format
1.7 Precautions on Issuing an Inquiry (for the Recorder's Settings)
1.8 Precautions on the Transfer of Messages
1.9 Precautions on the IFC Timing When Using the GP-IB Interface

1.1 Overview of Basic Program Formats

The transfer of data between the controlling device and the recorder is conducted in “message” units. A message sent from the controlling device is defined as a “program message,” whereas a message output by the recorder in response is defined as a “return message.”
When the controlling device sends to the recorder a program message containing commands that each query a return message in response, the recorder sends back return messages to the controlling device upon having received that program message from the controlling device.
In response to a program message that contains multiple queries, the recorder always outputs one return message corresponding to each query.



Sample Query

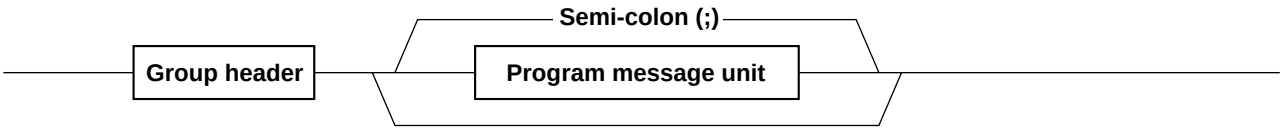
:AMP:CH1:INPUT?;RANGE?<PMT>

Sample Return Message

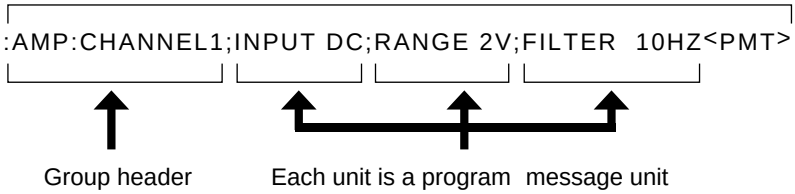
:AMP:CH1:INPUT DC;RANGE 5V<RMT>

1.2 Format of the Program Message

A program message is a collection of program message units which each form a single command and are delimited by a semi-colon (;). One program message combines multiple program message units and is terminated by a PMT (Program Message Terminator). Note that the recorder executes commands based on the sequence in which it has received program message units.



Sample Program Message



NOTE

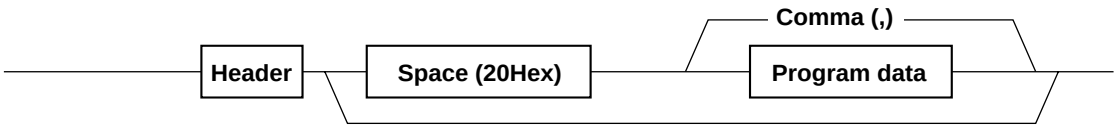
<PMT> : Abbreviation for Program Message Terminator
Character codes and control codes that are valid for use as the <PMT> are listed below.

IF type	Setting	<PMT>
RS-232C	Conforms to the New Line Code setting	
	New Line Code = CR + LF	CR + LF(0D 0A)$_{16}$
	New Line Code = CR	CR(0D)$_{16}$
	New Line Code = LF	LF(0A)$_{16}$
GP-IB	Terminator = EOI only	EOI
	Terminator = EOI + New Line Code	EOI + specified New Line Code setting
		EOI + CR + LF*
		EOI + CR
		EOI + LF
	Terminator = New Line Code	Specified New Line Code setting
		CR + LF
		CR
		LF

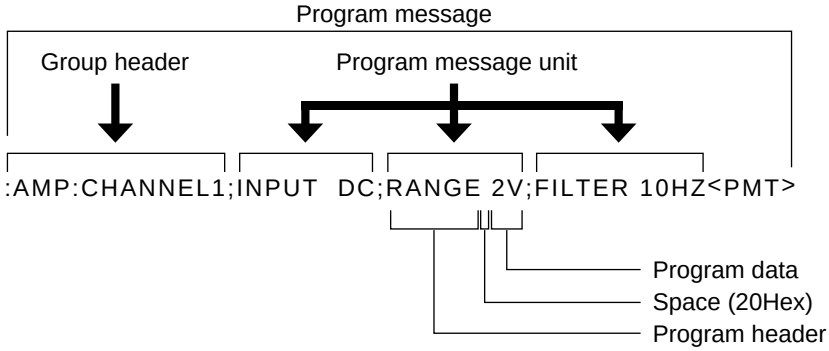
* EOI occurs concurrently with LF

1.3 Format of the Program Message Unit

The format of the program message unit is explained in detail within the command summary for each functional group.
One program message begins with a header that is uniquely defined for each functional group, followed by program message units that each consist of a program header and program data.



Sample Program Message



If Program Data Will Be Added

When constructing a program message unit (a command), program data (a parametric setting) is sometimes added. In this case, always construct the program message unit by inserting a space code (20Hex) before the program data.
Moreover, when a program message unit contains multiple sets of program data, separate each set of program data by inserting a comma (2CHex) as the delimiter. In this case, there is no need to precede each set of program data by a space (20Hex).

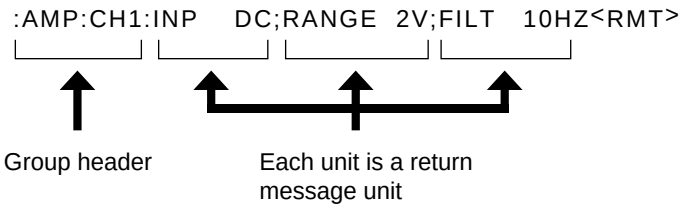
1.4 Format of the Return Message

When the controlling device sends to the recorder a program message containing “query” commands that each query a return message in response, the recorder sends back return messages to the controlling device upon having received that program message from the controlling device.
The return messages are sent to the controlling device according to the sequence in which the recorder receives them.

Sample Query

AMP:CHANNEL1:INPUT?;RANGE?;FILTER?<PMT>

Sample Return Message



NOTE

<RMT> : Abbreviation for Return Message Terminator
Character codes and control codes that are valid for use as the <RMT> are listed below.

IF type	Setting	<RMT>
RS-232C	Conforms to the New Line Code setting	
	New Line Code = CR + LF	CR + LF(0D 0A)$_{16}$
	New Line Code = CR	CR(0D)$_{16}$
	New Line Code = LF	LF(0A)$_{16}$
GP-IB	Terminator = EOI only	EOI
	Terminator = EOI + New Line Code	EOI + specified New Line Code setting
		EOI + CR + LF*
		EOI + CR
		EOI + LF
	Terminator = New Line Code	Specified New Line Code setting
		CR + LF
		CR
		LF

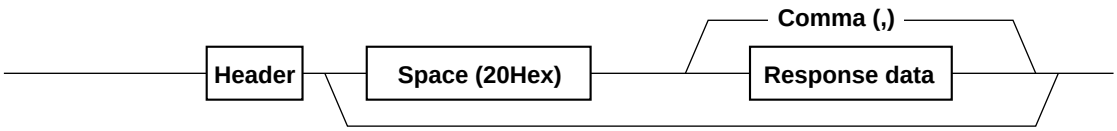
* EOI occurs concurrently with LF

NOTE

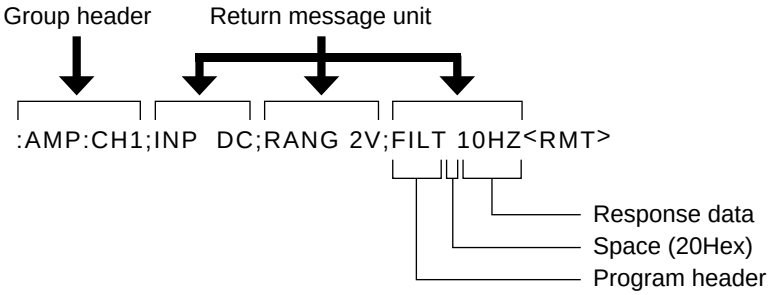
For this recorder’s GP-IB interface, no delimiter is specified for input data alone, and the EOI setting is always given top priority as the RMT code.

1.5 Format of the Return Message Unit

The format of the return message unit is explained in detail within the command summary for each functional group.
One return message begins with a header that is uniquely defined for each functional group, followed by return message units that each consist of a program header and response data.



Sample Return Message



If Response Data Will Be Added

A return message unit may be constructed to include response data (the status of parametric settings). In this case, always construct the return message unit by inserting a space code (20Hex) before the response data. Moreover, when a return message unit contains multiple sets of response data, separate each set of response data by inserting a comma (2CHex) as the delimiter. In this case, there is no need to precede each set of response data by a space (20Hex).

1.6 The Block Data Format

Memory waveform data and other data to be output using the binary format are output using the block data format which includes a specification of the total volume of data to be output.

Example: 120,000 bytes of block data

1	2	3	4	5	6	7	8	9	10	11	12
#	6	1	2	0	0	0	0	data	data	data	data

1st digit	Begins with the # symbol which signifies the output of block data
2nd digit	Indicates the number of digits used to express the total data volume which is specified from the third-digit position onward. With the WR1000, the number of digits specified here varies with the amount of data to be output.
3rd digit onward	Indicates the total volume of data as the number of data bytes, using the number of digits specified by the numeral in the second-digit position.
data	Data in binary format

NOTE

- The volume of data is expressed as the total number of data bytes. The portion of the format which indicates the number of data bytes (from the third-digit position onward) is written using ASCII character codes.
- Because every unit containing binary-format data will include eight bits ranging from 00H to FFH, this command can only be executed if the I/F setting conforms to the following conditions.

IF type	Binary data can be sent	Binary data cannot be sent
RS-232C	When the Flow Cont setting is None or when the Flow Cont setting is RS-CS and the Length setting is 8bit	When the Flow Cont setting is Xon/Xoff or the Length setting is 7bit
GP-IB	When the EOI setting is On (When the Terminator setting is EOI only or EOI + New Line Code)	When the EOI setting is Off

* The recommended Flow Cont setting for the RS-232C interface is RS-CS. The recommended Terminator setting for the GP-IB interface is EOI only. When using the block data format with the GP-IB interface, the character code(s) specified as the New Line Code setting will not be output as a delimiter.

1.7 Precautions on Issuing a Query (for the Recorder's Settings)

When a program message contains multiple queries, the recorder sends back return messages based on the sequence in which it has received the queries. Though the majority of queries are each answered by a single return message unit, there are also a few queries that are each answered by multiple return message units. As a result, even if the controlling device consecutively sends X number of queries, it may not always receive X number of return message units in response. To reliably obtain the parametric settings of the recorder, include only one query per program message in order to receive each return message unit corresponding to that query with certainty.

1.8 Precautions on the Transfer of Messages

(1) When a program message contains queries (to the recorder), be sure that the controlling device receives all of the recorder's return messages after completely transmitting the program message.
If all of the return messages are not received, an error occurs. In addition, any return message that is not received by the controlling device will be discarded by the recorder.

(2) When a program message is sent that contains multiple program message units, an error occurs if that program message contains formatting mistakes or unrecognizable program data.
If that program message contains a valid program message unit, only that program message unit may be acknowledged. Due to the relationship with the program message units that precede or follow a valid program message unit, however, the recorder will not always process every valid program message unit.

1.9 Precautions on the IFC Timing When Using the GP-IB Interface

When using the GP-IB Interface, specify 1 ms or longer for the IFC interval.

NOTE

The IFC interval can be specified from 100 μs, but if processing is not completed within the specified interval errors can occur if commands are sent consecutively.

CHAPTER 2

PROCEDURES FOR WRITING COMMANDS

The commands that can be used with this recorder are explained in detail in Chapter 3, “COMMAND DESCRIPTION BY FUNCTIONAL GROUP,” which contains command summaries and flowcharts.
This chapter provides an overview on writing three types of commands: common commands, compound commands, and control-only commands.

2.1 Writing Commands
2.1.1 Types of Command Headers
2.1.2 Writing Program Messages
2.1.3 Rules for Interpreting Command Headers
2.1.4 Rules for Interpreting Program Data
2.2 Replies to Queries
2.2.1 Higher-Order Queries
2.2.2 Output Format for the Return Message
2.3 The Handling of Program Data
2.3.1 Numeric Data (Decimal Numerals)
2.3.2 Dimensional Numeric Data
2.3.3 Alphabetic (Mnemonic) Data
2.3.4 Logical-Value Data
2.3.5 Character-String Data
2.3.6 Registers
2.3.7 Block Data

2.1 Writing Commands

2.1.1 Types of Command Headers

(1) Header for Common Commands

According to the IEEE488.2-1987 standard which stipulates the format of common commands, the command header format for a common command consists of a leading asterisk (* or 2AHex).

(2) Header for Compound Commands

All commands that are not common commands are classified by functional group and begin with a group header. The group header is always preceded by a colon (: or 3AHex).
Commands are also hierarchically arranged within each function group. To move down one rank in such a hierarchy, it is possible to write a command header preceded by a colon (: or 3AHex) and then to continue writing multiple commands (For details, see the description of settings for each functional group within Chapter 3, “COMMAND DESCRIPTION BY FUNCTIONAL GROUP.”)

(3) Header for Control-Only Commands

Within the functional groups, there are command headers that do not possess a functionally independent rank. Such commands as the Start Measurement, Stop Measurement, and Event Marker functions fall into this category.

Sample Statements :MEASURE:START
:MEASURE:STOP
:MEASURE:EVENT

2.1.2 Writing Program Messages

A program message is defined as a collection of program message units (commands). The method of writing programs offers a certain degree of freedom. If you write a program according to the programming rules, you will be able to consecutively issue commands.
This subsection describes the programming method based on examples of several different situations.

(1) Writing consecutive commands from the same functional group

To assemble a program message, select the target functional group using its functional group command header and then compose the desired commands.

Sample statement **:**AMP:CHANNEL1:INPUT DC;RANGE 2V;FILTER OFF<PMT>

Explanation
- The “:AMP” group header selects the AMP Group settings.
- The “:CHANNEL1” command header selects the channel number.
- “:INPUT DC” is a command on the next lower level. As a result, the RANGE and FILTER commands follow consecutively and are delimited by a semicolon because they are on the same level as the INPUT command.

(2) Writing consecutive commands from different functional groups

To assemble a program message, select the target functional group using its functional group command header and then compose the desired commands, and also add a semi-colon after the final command in that functional group. Next, assemble the next program message by selecting another function group using a different functional group command header and then compose the desired commands.

Sample statement **:**AMP:CHANNEL1:INPUT DC;RANGE 2V;FILTER OFF;:MEASURE:START<PMT>
Explanation
- This message specifies amp settings for Channel 1, followed by the Start Measurement command in the MEASURE Group.
- As illustrated by "FILTER OFF;", the final AMP Group command is followed by a semi-colon as the delimiter, after which the command level of the MEASURE Group is declared.

(3) Combining functional group settings and common commands

To transmit functional group settings followed by common commands, declare the presence of the common command by adding an asterisk (*) as the common command's header. No colon (:) is required for the common command's header.

Sample statement :MEASURE:STOP*CLS<PMT>

(4) Delimiting multiple program messages using the <PMT>

The <PMT> indicates the end of a single program message. Consequently, subsequent commands can only be acknowledged within a new program message. Always begin the transmission of a program message from the functional group header to select the functional group.

Sample statement **:**AMP:CHANNEL1:INPUT DC<PMT>:AMP:CHANNEL1:RANGE 2V<PMT>

2.1.3 Rules for Interpreting Command Headers

A command header received by the recorder is interpreted according to the following rules.

(1) A command header is recognized without distinguishing between the upper- and lower-case letters of mnemonic (alphabetic) formulas.

MEASURE → measure

(2) It is permissible to leave out those letters which appear as lower-case letters in the flowchart and command summary within each subsection of Chapter 3, "COMMAND DESCRIPTION BY FUNCTIONAL GROUP."

MEASure → MEAS → MEASU → MEASUR

2.1.4 Rules for Interpreting Program Data

Program data to be received by the recorder can be in one of two formats: (1) a format containing numeric-value settings that have been internally predetermined by the recorder (such as the input range or filter constant), and (2) a format that enables the recorder to freely accept numeric-value settings, provided such numeric values are each within their stipulated parametric range.

The recorder interprets program data according to the following rules.

(1) When using the format containing predetermined numeric-value settings such as the input range, if the corresponding value does not exist within the recorder, an error occurs and the recorder does not execute the specified processing.

Example :AMP:CHANNEL1:RANGE 20mV<PMT>
Program data

Correct example : 20mV or 0.02V or 20E-03V
Incorrect example : 25mV and other values that do not correspond to one of the recorder's internally existing ranges

For more information, see Section 2.3, "The Handling of Program Data," and each subsection of Chapter 3, "COMMAND DESCRIPTION BY FUNCTIONAL GROUP."

(2) When using the format that enables the recorder to freely accept numeric-value settings, provided such numeric values are each within their stipulated parametric range, if an out-of-range value is received, an error occurs and the recorder does not execute the specified processing.

2.2 Replies to Queries

When the controlling device issues a query consisting of a command that ends with a question mark (?), the recorder outputs a return message in response.

2.2.1 Higher-Order Queries

When a query is issued using a higher-order command in a functional group, the recorder outputs a return message that includes the status of lower-order settings.

The settings received in that return message can be used as a program message. By sending such settings to the recorder, it is possible to return the recorder's settings to their previous status resulting from the receipt of the higher-order query.

Depending on the functional group, however, functional settings which are not currently in use may not be included in the return message, making it impossible to completely restore the settings in such case.

2.2.2 Output Format for the Return Message

When using commands that permit a return message to be directly used as a program message, the return message in response to a query consists of a command header followed by program data.

Sample query :AMP:CHANNEL1:INPUT?<PMT>
Sample response :AMP:CH1:INP DC<RMT>

NOTE : In the return message, an abbreviated form of the command header is output.

When using commands that do not permit a return message to be directly used as a program message, the return message in response to a query may consist solely of program data in some cases.

Sample query *IDN?<PMT>
Sample response GRAPHTEC,WR1000,0,1.00
Explanation A common command is used to query the model ID. The recorder sends back the manufacturer's name, the model number, and the firmware's version number.

2.3 The Handling of Program Data

The program data consists of a command header, a space (20Hex), and settings consisting of alphabetic characters or numeric values.
The recorder determines the parameter according to the command header and then uses the program data to rewrite that parameter's setting. The following subsections describe the different categories of program data. For details about the formatting rules for numeric values and the permissible ranges of parametric settings, either refer to the "WR1000 USER'S MANUAL" or see the pertinent subsection within Chapter 3, "COMMAND DESCRIPTION BY FUNCTIONAL GROUP."

2.3.1 Numeric Data (Decimal Numerals)

Decimal values are written in the NR format stipulated in the ANSI X3.42-1975 standard.

The recorder uses two types of formats for numeric data: <NR1> and <NRf>. For information about which commands use <NR1> or <NRf>, see the pertinent subsection within Chapter 3, "COMMAND DESCRIPTION BY FUNCTIONAL GROUP."

Sumbol	Description
<NR1>	Integers (1,100, -200, +150, etc.)
<NRf>	Integers, fixed decimal points (10.00, -1.5, +10.000, etc.), and floating decimal points (3.21E00, 7E-03, 1.5E03, etc.) are acceptable.

NOTE

Regarding the handling of program data, there is a certain degree of freedom when writing data to be issued by the controlling device (program messages), but the return messages are sent back using a fixed format determined by the query command. For more details, see each subsection within Chapter 3, "COMMAND DESCRIPTION BY FUNCTIONAL GROUP."

2.3.2 Dimensional Numeric Data

Settings related to the analog input, such as <Voltage> as well as time-related settings such as the sampling interval are representative examples of program data containing dimensional numeric values.
Such numeric values are expressed using <NRf> followed by the unit. For more information about commands that use <Voltage>, see the flowchart in each subsection of Chapter 3, "COMMAND DESCRIPTION BY FUNCTIONAL GROUP."

Symbol	Description
<Voltage>	20 mV → 2.0E-02 V → 0.02 V
<Time>	20 μs → 2.0E-06 s → 0.02 ms

(1) Permissible factors consist of the three types below.

Symbol	Meaning	Factor
K (k)	kilo	10^{3}
M (m)	milli	10^{-3}
U (u)	micro	10^{-6}

(2) Permissible units consist of the two types below.

Symbol	Meaning	Description
V (v)	volts	Voltage
S (s)	seconds	Time (in seconds)

NOTE

Regarding the handling of program data, there is a certain degree of freedom when writing data to be issued by the controlling device (program messages), but the return messages are sent back using a fixed format determined by the query command. For more details, see each subsection within Chapter 3, "COMMAND DESCRIPTION BY FUNCTIONAL GROUP."

2.3.3 Alphabetic (Mnemonic) Data

Alphabetic data refers to mnemonic data that is stipulated by the recorder, whereby the available settings are expressed by specific alphabetic data.
In the case of the INPUT (input coupling) command, for example, such settings as OFF, DC, AC, and GND each correspond to alphabetic data. No abbreviated form exists for any of the alphabetic data.

2.3.4 Logical-Value Data

Logic-value data is program data used for selecting whether to enable or disable a function by On or Off settings, respectively.

2.3.5 Character-String Data

In contrast with alphabetic data which consists of mnemonic data that is stipulated by the recorder, character-string data consists of user-defined alphabetic data.
Character-string data consists of the two types described below. Each set of character-string data is expressed by enclosing the character string which forms the program data by a pair of either double quotation marks (" or 22Hex) or single quotation marks (' or 2cHex).
When a double or single quotation mark is required within a character string comprising program data, send two of them consecutively.

Regular character-string data : " 123ABC"
Recognized character string : 123ABC
Exceptional character-string data : " 123"" ABC"
Recognized character string : 123" ABC

(1) Character-string data that has a free format and requires no internal interpretation by the recorder, such as an annotation character string.

(2) Character-string data that has a format that is internally stipulated by the recorder, such as a character string that will substitute for a variable in the numeric formula in a MATH Group command.

(3) Character-string data that is generally subject to format restrictions, such as a file name.

For more information about the character-string data described in (2) and (3) above, refer to the "WR1000 USER'S MANUAL."

2.3.6 Registers

A register contains binary or hexadecimal data wherein each bit has a specific meaning, similar to a status report. Such data are basically written as indicated in the following table. Each bit is weighted and the sum of all valid bits is expressed as a decimal value.

Program data & response data

Format	Sample data representation
<NR1>	128

<NR1>: A decimal value representing the sum of all weighted “1” bits

Bit no.	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Weight	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}

2.3.7 Block Data

Memory waveform data and other data to be output in binary format is output using the block data format which includes a specification of the total volume of data to be output.

Example: 120,000 bytes of block data

1	2	3	4	5	6	7	8	9	10	11	12
#	6	1	2	0	0	0	0	data	data	data	data

1st digit	Begins with the # symbol which signifies the output of block data
2nd digit	Indicates the number of digits used to express the total data volume which is specified from the third-digit position onward. With the WR1000, the number of digits specified here varies with the amount of data to be output.
3rd digit onward	Indicates the total volume of data as the number of data bytes, using the number of digits specified by the numeral in the second-digit position.
data	Data in binary format

NOTE

- The volume of data is expressed as the total number of data bytes. The portion of the format which indicates the number of data bytes (from the third-digit position onward) is written using ASCII character codes.
- Because every unit containing binary-format data will include eight bits ranging from 00H to FFH, this command can only be executed if the I/F setting conforms to the following conditions.

IF type	Binary data can be sent	Binary data cannot be sent
RS-232C	When the Flow Cont setting is None or when the Flow Cont setting is RS-CS and the Length setting is 8bit	When the Flow Cont setting is Xon/Xoff or the Length setting is 7bit
GP-IB	When the EOI setting is On (When the Terminator setting is EOI only or EOI + New Line Code)	When the EOI setting is Off

* The recommended Flow Cont setting for the RS-232C interface is RS-CS. The recommended Terminator setting for the GP-IB interface is EOI only. When using the block data format with the GP-IB interface, the character code(s) specified as the New Line Code setting will not be output as a delimiter.

CHAPTER 3

COMMAND DESCRIPTION BY FUNCTIONAL GROUP

This chapter provides a description of each command. The commands have been grouped according to their respective functions.

3.1 Input Amp Settings (AMP Group)
3.1.1 Summary of AMP Group Commands
3.1.2 Functions and Settings by Amp Type
3.2 Logic Amp Settings (LOGIC Group)
3.2.1 Summary of LOGIC Group Commands
3.3 Settings Related to Measurement and Control (MEASURE Group)
3.3.1 Summary of MEASURE Group Commands
3.4 Memory-Related Settings (MEMORY Group)
3.4.1 Summary of MEMORY Group Commands
3.4.2 Program Data Settings for MEMORY Group Settings
3.4.2.1 Settings for Saving Data to Disk
3.4.2.2 Settings and Replies Related to the Sampling Interval
3.5 REPLAY Settings (REPLAY Group)
3.5.1 Summary of REPLAY Group Commands
3.6 Trigger-Related Settings (TRIGGER Group)
3.6.1 Summary of TRIGGER Group 1/2 Commands
3.6.2 Summary of TRIGGER Group 2/2 Commands
3.7 Settings Related to Recording (RECORD Group)
3.7.1 Summary of RECORD Group Commands
3.7.2 Settings and Replies Related to the Chart Speed
3.8 Settings Related to Logging (LOGGING Group)
3.8.1 Summary of LOGGING Group Commands
3.9 Annotation Settings (ANNOTATION Group)
3.9.1 Summary of ANNOTATION Group Commands
3.9.2 Supplementary Information on Annotation Printing
3.9.2.1 Setting the Annotation Functions
3.9.2.2 Printing Control Via the Interface
3.10 System Settings (SYSTEM Group)
3.10.1 Summary of SYSTEM Group Commands
3.11 Printer Settings (PRINTER Group
3.11.1 Summary of PRINTER Group Commands
3.12 Common Command Settings (COMMON Group)
3.12.1 Summary of COMMON Group Commands
3.13 Interface Status Control (STATus Group)
3.13.1 Summary of STATUS Group Commands

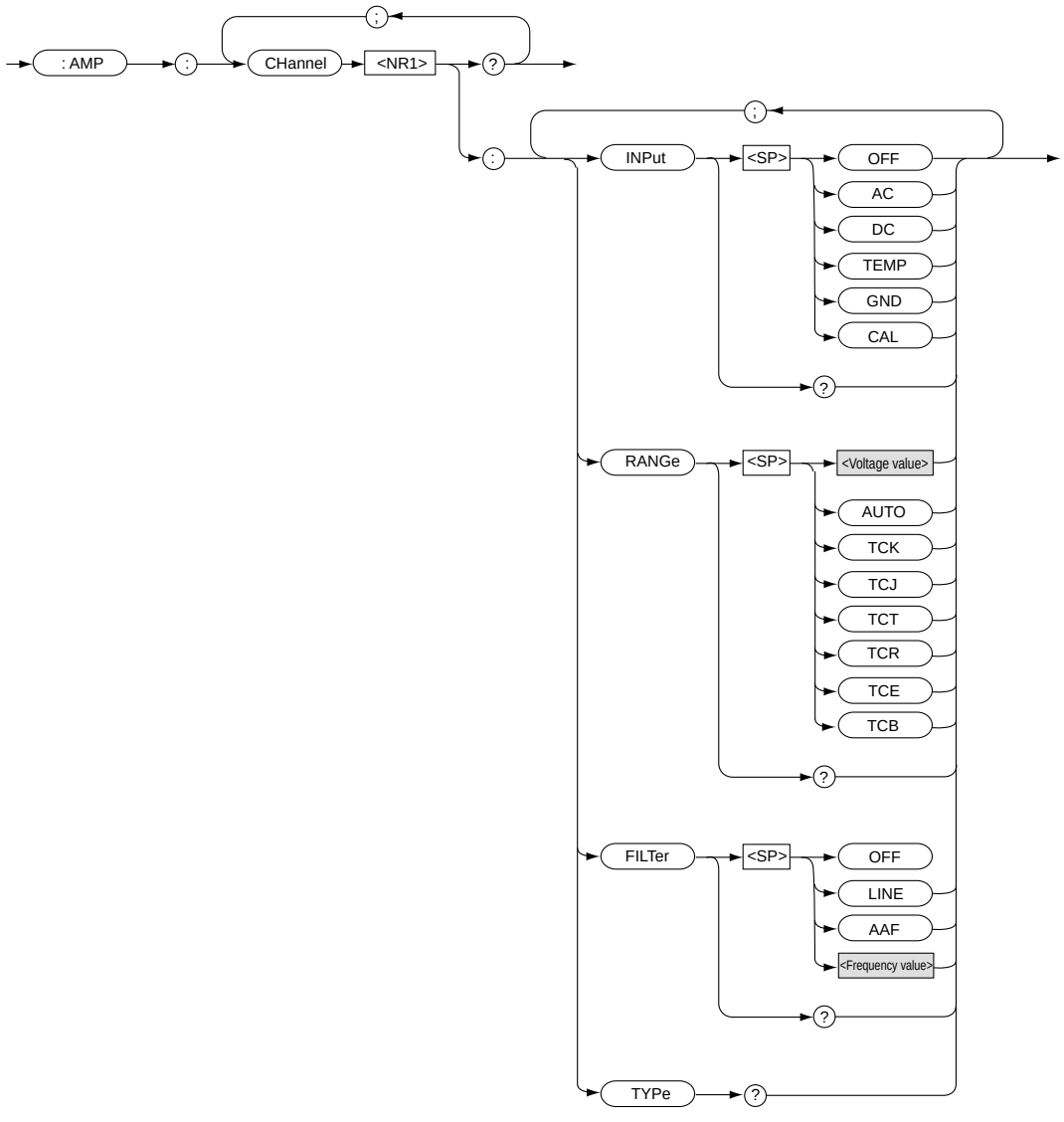
3.1 Input Amp Settings (AMP Group)

The AMP Group is the command group related to the input range for each channel.
They can be used to set or query the settings of the input amps. Depending on the amp type, the AMP Group is subject to restrictions on its commands and parameters. (for details, see Subsection 3.1.2, “Functions and Settings by Amp Type”).

NOTE

For information about the logic amp, see Section 3.2, “Logic Amp Settings (LOGIC Group).”

Tree Structure of AMP Group Commands



3.1.1 Summary of AMP Group Commands

Command : :AMP:CHannel<NR1>?
Function : Requests all input-related settings of the specified channel.
<NR1> : The channel no. Any integer between 1 to 32 is valid.
Sample query : :AMP:CHANNEL1? or :AMP:CHANNEL1?;CHANNEL2?
Sample response : :AMP:CH1:INP DC;RANG 2V;FILT OFF;TYP V

NOTE

The response below is returned if the specified channel is not equipped with an amp.
:AMP:CH1:INP NONE;RANG NONE;FILT NONE;TYP NONE

Command : :AMP:CHannel<NR1>:INPut
Function : Specifies or queries the input coupling status of the specified channel
<NR1> : The channel no. Any integer between 1 to 32 is valid.
Sample setting : :AMP:CHANNEL1:INPUT DC
Sample query : :AMP:CHANNEL1:INPUT?
Sample response : :AMP:CH1:INP DC

NOTE

- The response below is returned if the specified channel is not equipped with an amp.
  :AMP:CH1:INP NONE
- For a description of the available INPUT settings, see Subsection 3.1.2, “AMP Group Functions and Settings.”

Command : :AMP:CHannel<NR1>:RANGe
Function : Specifies or queries the measurement range setting of the specified channel.
<NR1> : The channel no. Any integer between 1 to 32 is valid.
Sample setting : :AMP:CHANNEL1:RANGE 2V
Sample query : :AMP:CHANNEL1:RANGE?
Sample response : :AMP:CH1:RANG 2V

NOTE

- The response below is returned if the specified channel is not equipped with an amp.
  :AMP:CH1:RANG NONE
- For a description of the available RANGE settings, see Subsection 3.1.2, “AMP Group Functions and Settings.”

Command : :AMP:CHannel<NR1>:FILTer
Function : Specifies or queries the filter setting of the specified channel.
<NR1> : The channel no. Any integer between 1 to 32 is valid.
Sample setting : :AMP:CHANNEL1:FILTER OFF
Sample query : :AMP:CHANNEL1:FILTER?
Sample response : :AMP:CH1:FILT OFF

NOTE

- The response below is returned if the specified channel is not equipped with an amp.
  :AMP:CH1:FILT NONE
- For a description of the available FILTER settings, see Subsection 3.1.2, “AMP Group Functions and Settings.”

Command : :AMP:CHannel<NR1>:TYPe?
Function : Requests the type of amp installed on the specified channel.
<NR1> : The channel no. Any integer between 1 to 32 is valid.
Sample query : :AMP:CHANNEL1:TYPE?
Sample response : :AMP:CH1:TYP V

NOTE

- The response below is returned if the specified channel is not equipped with an amp.
  :AMP:CH1:TYP NONE
- The recorder outputs response data conforming to the amp type, as listed in the following table.

Amp type	Response data
V (Voltage) amp	V
M (Voltage & Temperature) amp	M

NOTE

For amp units other than the V, M, and Logic amps, please refer to each individual Amp User Guide for descriptions of the interface commands.

3.1.2 Functions and Settings by Amp Type

The available settings for the V (Voltage) amp and multi-input M (voltage & temperature) amp are listed in the following table.

Note that the settings below cannot be set for the V amp:

- INPUT : TEMP
- RANGE : 20mV, TC-K, TC-J, TC-T, TC-R, TC-E, TC-B

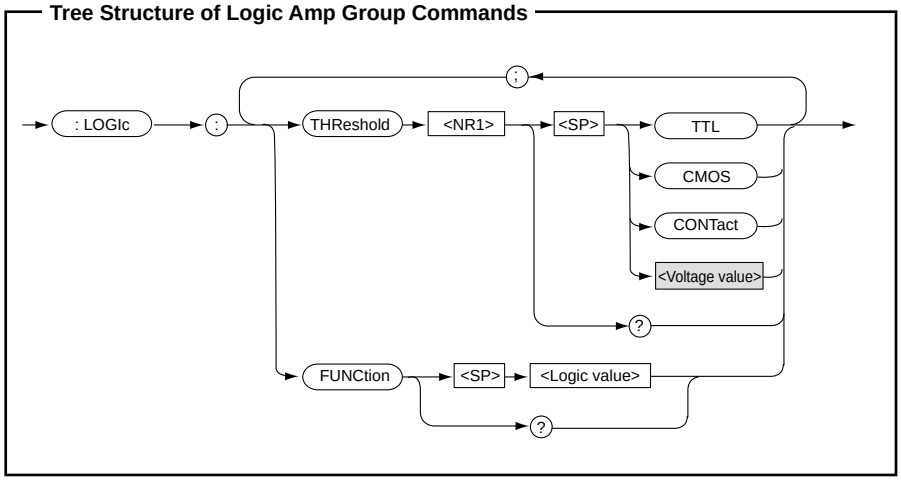
INPUT	INPUT response format	RANGE	RANGE response format	FILTER	FILTER response format
OFF	OFF	20 mV*	20 mV	OFF	OFF
AC	AC	50 mV	50 mV	LINE	LINE
DC	DC	100 mV	100 mV	5 Hz	5 Hz
GND	GND	200 mV	200 mV	10 Hz	10 Hz
CAL	CAL	500 mV	500 mV	30 Hz	30 Hz
TEMP*	TEMP	1 V	1 V	50 Hz	50 Hz
		2 V	2 V	500 Hz	500 Hz
		5 V	5 V	5 kHz	5 kHz
		10 V	10 V		
		20 V	20 V		
		50 V	50 V		
		100 V	100 V		
		200 V	200 V		
		500 V	500 V		
		AUTO	Selected voltage range		
		TC-K*	TCK		
		TC-J*	TCJ		
		TC-T*	TCT		
		TC-R*	TCR		
		TC-E*	TCE		
		TC-B*	TCB		

* Can only be specified for the M amp.

3.2 Logic Amp Settings (LOGIC Group)

The LOGIC Group is the Command group related to the logic amp settings.
LOGIC Group Commands can be used to set or request the settings of a logic amp.

Tree Structure of Logic Amp Group Commands



3.2.1 Summary of LOGIC Group Commands

Command : :LOGIc:THReshold<NR1>
Function : Specifies or requests the input settings for a logic group.
<NR1> : Logic group no. 1, 2, 3, and 4 respectively correspond to A, B, C, and D.
Sample setting : :LOGIC:THRESHOLD1 TTL
Sample query : :LOGIC:THRESHOLD1?
Sample response : :LOGI:THR1 CMOS

Sample application: Setting :LOGIC:THRESHOLD1 TTL;THRESHOLD2 CMOS
Query :LOGIC:THRESHOLD1?;THRESHOLD2?

NOTE

The Threshold level setting is specified after the space which follows the Logic Group No. The Threshold level can be specified one of three settings below.

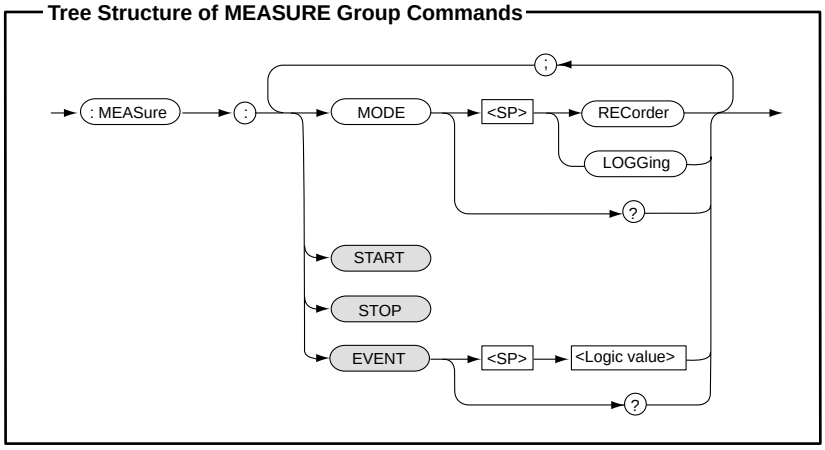
- TTL = +1.4 V
- CMOS = +2.5 V
- CONTACT = +5 V

Command : :LOGIc:FUNCtion
Function : Selects or requests the On/Off status of the logic amp.
Sample setting : :LOGIC:FUNCTION ON
Sample query : :LOGIC:FUNCTION?
Sample response : :LOGIC:FUNC ON

3.3 Settings Related to Measurement and Control (MEASURE Group)

The MEASURE Group is the Command group related to the recorder's measurement conditions and control-related settings.
MEASURE Group Commands can be used to set or request settings related to measurement and control.

Tree Structure of MEASURE Group Commands



3.3.1 Summary of MEASURE Group Commands

Command : :MEASure:MODE
Function : Selects or requests the recorder's measurement mode.
Sample setting : :MEASURE:MODE RECORDER
Sample query : :MEASURE:MODE?
Sample response : :MEAS:MODE REC

Command : :MEASure:START
Function : Instructs the recorder to start measurement.
Sample setting : :MEASURE:START

Command : :MEASure:STOP
Function : Instructs the recorder to stop measurement.
Sample setting : :MEASURE:STOP

Command : MEASure:EVENT
Function : Duplicates the function of the EVENT key. Select ON to continue the generation of events, or select OFF to discontinue the generation of events. Whenever this Command is input, the recorder is internally informed of the event status.
Sample setting : :MEASURE:EVENT ON
Sample query : :MEASURE:EVENT?
Sample response : :MEAS:EVENT OFF

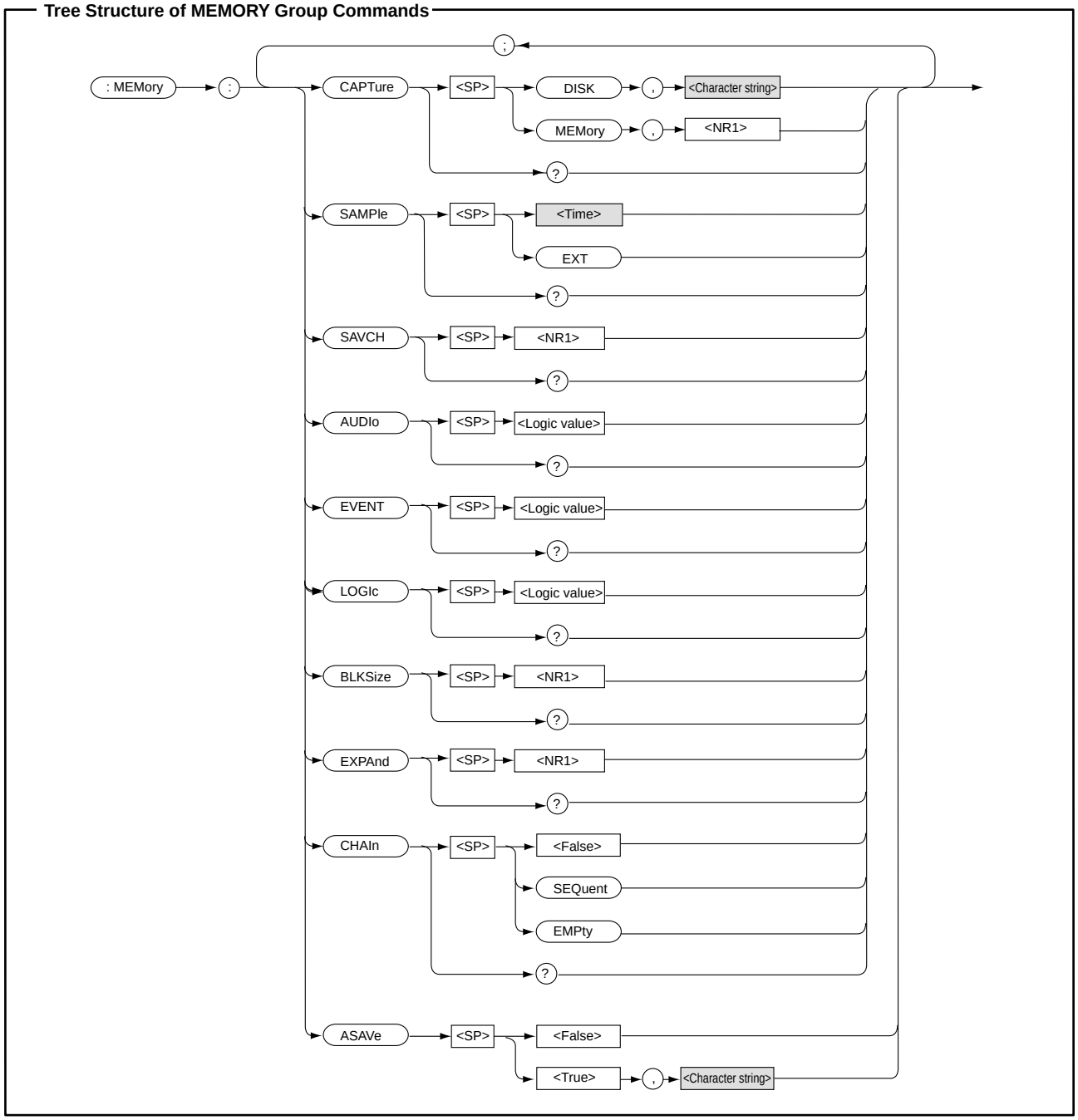
NOTE

This command is only valid during a Direct Save operation. Even if ON was selected during a Direct Save operation, it will automatically be set to OFF when measurement has been completed or when the recorder is switched to LOCAL mode.

3.4 Memory-Related Settings (MEMORY Group)

The MEMORY Group is the Command group related to the recorder's memory settings.
MEMORY Group Commands can be used to set or request memory-related settings. For information about setting procedures and restrictions, refer to the "WR1000 USER'S MANUAL."

Tree Structure of MEMORY Group Commands



3.4.1 Summary of MEMORY Group Commands

Command : :MEMory:CAPTure
Function : Specifies or requests the destination memory device
Sample setting : :MEMORY:CAPTURE MEMORY,1
Sample query : :MEMORY:CAPTURE?
Sample response : :MEM:CAPT MEM,1

NOTE

When the Save To setting is Memory, a Memory Block No. must also be specified to allocate an area of the recorder's internal RAM as the destination for capturing data.
When the Save To setting is Disk, the destination file name must also be specified. For details on how to designate a file name, see Subsection 3.4.2.1, "Settings for Saving Data to Disk."

Command : MEMory:SAMPLe
Function : Specifies or requests the sampling interval for saving data.
Sample setting : :MEMORY:SAMPLE 10us
Sample query : :MEMORY:SAMPLE?
Sample response : :MEM:SAMPL 10us

NOTE

For a list of permissible sampling interval settings, see Subsection 3.4.2.2, "Settings and Replies Related to the Sampling Interval."

Command : MEMory:SAVCH <NR1>
Function : Specifies or requests the number of channels involved in the saving of channel data to disk. <NR1> represents the number of channels, which can be specified as any integer from 1 to 32.
Sample setting : :MEMORY:SAVCH 8
Sample query : :MEMORY:SAVCH?
Sample response : :MEM:SAVCH 16

Command : :MEMory:AUDIo
Function : Specifies or requests the On/Off status of the Audio function for memory-related operations.
Sample setting : :MEMORY:AUDIO ON
Sample query : :MEMORY:AUDIO?
Sample response : :MEM:AUDI OFF

NOTE

The response below is returned if the audio input card is not installed.
:MEM:AUDI NONE

Command : :MEMory:EVENT
Function : Specifies or requests the On/Off status for saving event data to disk.
Sample setting : :MEMORY:EVENT ON
Sample query : :MEMORY:EVENT?
Sample response : :MEM:EVENT OFF

Command : :MEMory:LOGIc
Function : Specifies or requests the On/Off status for saving logic data to disk.
Sample setting : :MEMORY:LOGIC ON
Sample query : :MEMORY:LOGIC?
Sample response : :MEM:LOGI OFF

NOTE
The response below is returned if the logic amp is not installed.
:MEM:LOG NONE

Command : :MEMory:BLKSize
Function : Specifies or requests the number of block divisions in the recorder's internal RAM.
Sample setting : :MEMORY:BLKSIZE 32
Sample query : :MEMORY:BLKSIZE?
Sample response : :MEM:BLKS 128

NOTE
The Block Size setting (number of memory blocks) can be set to 1, 2, 4, 8, 16, 32, 64, or 128.

Command : :MEMory:EXPAnd
Function : Specifies or requests the expansion factor for the recorder's internal RAM.
Sample setting : :MEMORY:EXPAND 2
Sample query : :MEMORY:EXPAND?
Sample response : :MEM:EXPA 2

NOTE
The expansion factor can be set to 1, 2, 4, or 8.

Command : :MEMory:CHAIn
Function : Specifies or requests the Memory Chain setting which determines the sequence for saving data to memory blocks.
Sample setting : :MEMORY:CHAIN OFF
Sample query : :MEMORY:CHAIN?
Sample response : :MEM:CHAI OFF

NOTE
The Memory Chain setting can be set to OFF, SEQuent, or EMPty.

Command : :MEMory:ASAVe
Function : Specifies or requests the On/Off status of the Auto Save function for saving data automatically.
Sample setting : :MEMORY:ASAVE ON "C:\GRAPHTEC. GBD"
Sample query : :MEMORY:ASAVE?
Sample response 1 : :MEM:ASAV OFF
Sample response 2: :MEM:ASAV ON,"C:\GRAPHTEC. GBD"

NOTE

- This command specifies whether or not to save memory data automatically.
- The file format is designated by the file name extension. However, if the auto file name function is used, the file name extension cannot be specified by a command. The file format setting specified at the recorder is used.

  GBD: Binary format. **TXT:** text format
- The ASAVE parameter can be set to ON or OFF. When set to ON, the destination file name must also be specified. For details on how to designate a file name, see Subsection 3.4.2.1, “Settings for Saving Data to Disk”.
- The response when the auto file name function is used will be similar to the following:

  :MEM:ASAV ON, **“C:\GRAPHTEC\ WR1000\”**

3.4.2 Program Data Settings for MEMORY Group Settings

3.4.2.1 Settings for Saving Data to Disk

To designate a file name as program data, specify the drive letter and file name as indicated below.

Example: “A:\GRAPHTEC DATA-1.DAT”

Since this program data consists of a character string, it must be enclosed by double quotation marks.

Drive letter	Description
A	Floppy disk drive
C	Hard disk drive 1
D	Hard disk drive 2
E	Hard disk drive 3
F	Hard disk drive 4
G	Hard disk drive 5
H onward	External device (a hard disk drive, Zip drive, or MO drive)

Due to the actual installed status of internal and external memory devices, the drive letters indicated in the table above may not always be available.

NOTE

When using the auto file name function, you can specify a folder name which ends in “\”. At this time, the last folder in the string need not be an existing folder, but the one preceding it must be.

Example: The command for saving measured data to an automatically designated file name in the WR1000 folder within the GRAPHTEC folder in the C drive is:

:MEM:CAPT DISK,“C:\GRAPHTEC\WR1000\”

GRAPHTEC must be an existing folder name
WR1000 need not be an existing folder name

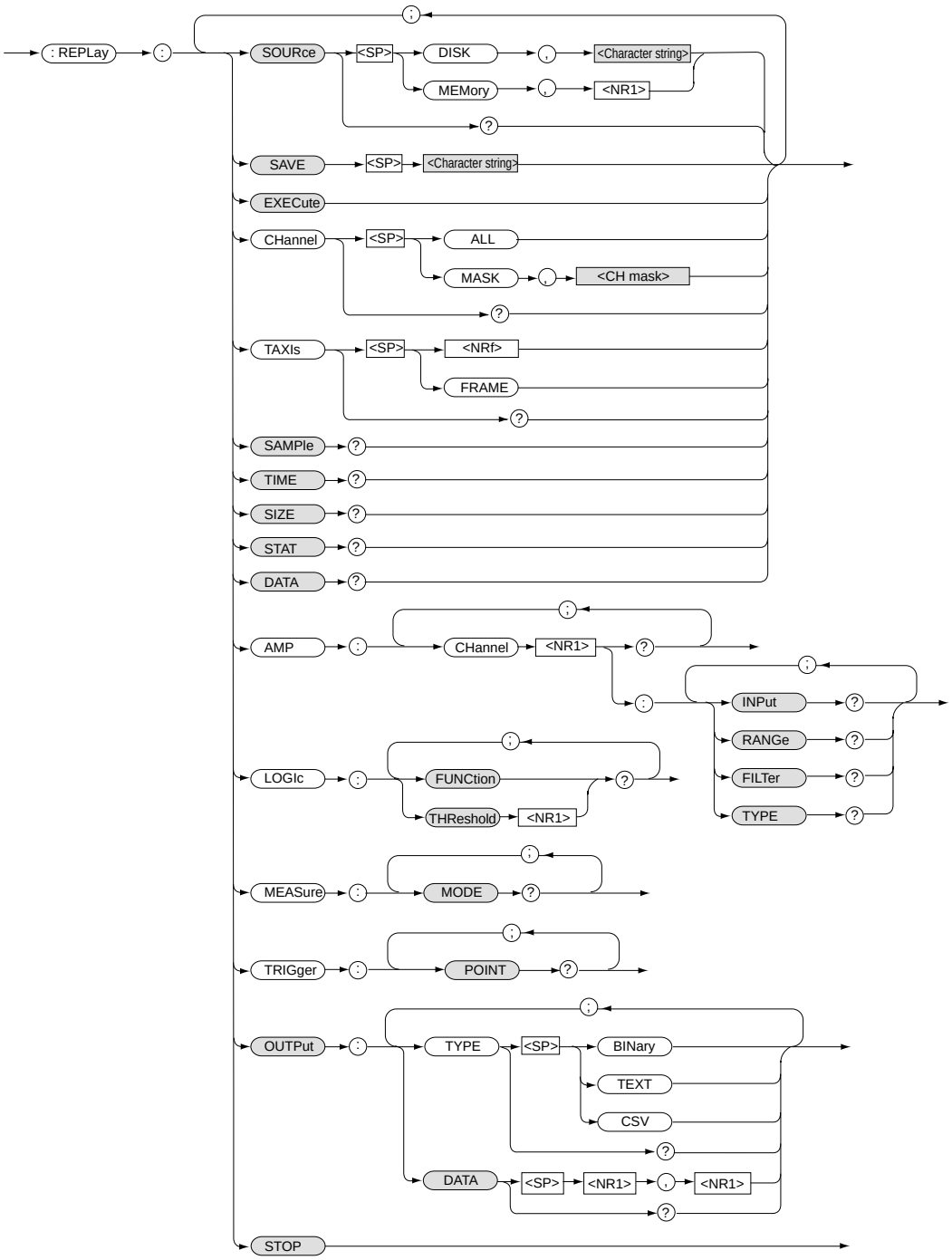
3.4.2.2 Settings and Replies Related to the Sampling Interval

Program data in program message		Response data in return message	Program data in program message		Response data in return message
1 µs	1E-06s	1 µs	1 ms	1E-03s	1 ms
2 µs	2E-06s	2 µs	2 ms	2E-03s	2 ms
4 µs	4E-06s	4 µs	4 ms	4E-03s	4 ms
5 µs	5E-06s	5 µs	8 ms	8E-03s	8 ms
8 µs	8E-06s	8 µs	10 ms	1E-02s	10 ms
10 µs	1E-05s	10 µs	20 ms	2E-02s	20 ms
20 µs	2E-05s	20 µs	40 ms	4E-02s	40 ms
40 µs	4E-05s	40 µs	50 ms	5E-02s	50 ms
50 µs	5E-05s	50 µs	80 ms	8E-02s	80 ms
80 µs	8E-05s	80 µs	100 ms	1E-01s	100 ms
100 µs	1E-04s	100 µs	200 ms	2E-01s	200 ms
200 µs	2E-04s	200 µs	400 ms	4E-01s	400 ms
400 µs	4E-04s	400 µs	800 ms	8E-01s	800 ms
800 µs	8E-04s	800 µs	1 s		1 s
			2 s		2 s
			4 s		4 s
			5 s		5 s
			EXT		EXT

3.5 REPLAY Settings (REPLAY Group)

The REPLAY Group is the Command group related to the output of data captured in the recorder's internal RAM.
REPLAY Group Commands can be used to set or request settings related to Memory Out operations. For information about setting procedures and restrictions, refer to the "WR1000 USER'S MANUAL."

Tree Structure of REPlay Group Commands



3.5.1 Summary of REPLAY Group Commands

Command : :REPLay:SOURce
Function : Specifies or requests the source location for loading data to be replayed for display.
Sample setting : :REPLAY:SOURCE MEMORY,1
Sample query : :REPLAY:SOURCE?
Sample response 1: :REPL:SOUR DISK,“C:\GRAPHTEC.DAT”
Sample response 2: :REPL:SOUR MEM,1

NOTE

- In sample response 1, the designated drive letter and file name are returned. In sample response 2, the Memory Block No. is returned. When the Replay Source setting is Disk, the following type of response may also be returned if a save or replay operation is not performed after the disk has been inserted in the drive:
  :REPL:SOUR DISK,“NONE”
- When the Replay Source setting is Disk, the source file name must also be designated. For details on how to designate the file name, see Subsection 3.4.2.1, “Settings for Saving Data to Disk.”

Command : :REPLay:SAVE
Function : Specifies the source memory data as a Memory Block No. and specifies its output destination as a file name.
Sample setting : :REPLAY:SAVE“A:\GRAPHTEC.GBD”

NOTE

- For details on how to designate a file name, see Subsection 3.4.2.1, “Settings for Saving Data to Disk.”
- This is an executable Command, so be careful when composing its program message.

Command : :REPLay:EXECute
Function : Executes the output of memory data.
Sample setting : :REPLAY:EXECUTE

NOTE

- This Command has the same effect as pressing the recorder’s MEM.OUT key.
- This is an executable Command, so be careful when composing its program message.

Command : :REPLay:CHannel
Function : Specifies or requests the channels for which to perform a memory output operation.
Sample setting : :REPLAY:CHANNEL ALL
Sample query : :REPLAY:CHANNEL?
Sample response : :REPL:CH ALL
Sample application: REPLAY:CHANNEL MASK,CH6,CH7,CH8

NOTE

In the sample application above, replay is disabled for the data of channels 6, 7, and 8, but enabled for the remaining channels. Any amp not equipped with an amp will be internally ignored by the recorder. The REPL:CH MASK command will clear any previous setting that has disabled one or more channel numbers for replay prior to execution of the REPL:CH command.

Command : :REPLay:TAXIs
Function : Specifies or requests the magnification factor of the time axis for recording memory data as a Memory Output operation.
Sample setting : :REPLAY:TAXIS 0.125
Sample query : :REPLAY:TAXIS?
Sample response : :REPL:TAXI FRAME

NOTE
The method for printing memory data can be set to FRAME or TAXIS (Time Axis). When TAXIS is selected, the magnification rate of the time axis (NRf) can be set to 10, 8, 5, 4, 2, 1, 0.5, 0.25, 0.2, 0.125, or 0.1.

Command : :REPLay:SAMPle?
Function : Requests the sampling interval that was used when capturing the data to be replayed.
Sample query : REPLAY:SAMPLE?
Sample response : REPL:SAMP 10Es-03s

Command : :REPLay:TIME?
Function : Requests the starting time, ending time, and trigger time of the data to be replayed.
Sample query : :REPLAY:TIME?
Sample response : :REPL:TIME “1999-10-10 12:00:00”,“1999-10-10 12:00:15”,“1999-10-10 12:00:01”

Command : :REPLay:SIZE?
Function : Requests the number of data points in the data to be replayed. The response indicates the number of data points per channel.
Sample query : :REPLAY:SIZE?
Sample response : :REPL:SIZE 2000

Command : :REPLay:STAT?
Function : Requests the present/absent status of data in the Memory Block to be replayed.
Sample query : :REPLAY:STAT?
Sample response : :REPL:STAT NONE

NOTE
FULL and NONE respectively indicate the presence and absence of the pertinent data.

Command : :REPLay:DATA?
Function : Requests the sequence for data output.
Sample query : :REPLAY:DATA?
Sample response : :REPL:DATA CH1,CH2,CH3,CH4,LOGI,EVENT

NOTE
The sequence given in the above response corresponds to the actual data and sequence output in response to the :REPLay:OUTPut:DATA command on page 3-18.

Command : :REPLay:AMP:CHannel<NR1>?]
Function : Requests the amp conditions that were used by the amp specified by channel number when capturing the data to be replayed.
Sample query : :REPLAY:AMP:CHANNEL1?
Sample response : :REPL:AMP:CH1:INP DC,RANGE 2V,FILT OFF,TYP V

Command : **:REPLay:AMP:CHannel<NR1>:INPut?**
Function : Requests the Range setting, that was used by the amp specified by channel number, when capturing the data to be replayed.
Sample query : :REPLAY:AMP:CHANNEL1:INPUT?
Sample response : :REPL:AMP:CH1:INP DC

Command : **:REPLay:AMP:CHannel<NR1>:RANGe?**
Function : Requests the Range setting, that was used by the amp specified by channel number, when capturing the data to be replayed.
Sample query : :REPLAY:AMP:CHANNEL1:RANGE?
Sample response : :REPL:AMP:CH1:RANG 2V

Command : **:REPLay:AMP:CHannel<NR1>:FILTer?**
Function : Requests the Filter setting that was used by the amp specified by channel number when capturing the data to be replayed.
Sample query : :REPLAY:AMP:CHANNEL1:FILTER?
Sample response : :REPL:AMP:CH1:FILT OFF

Command : **:REPLay:AMP:CHannel<NR1>:TYPe?**
Function : Requests the type of the amp, specified by channel number, used when capturing the data to be replayed.
Sample query : :REPLAY:AMP:CHANNEL1:TYPE?
Sample response : :REPL:AMP:CH1:TYP V

Command : **:REPLay:LOGic:FUNCtion?**
Function : Requests the present/absent status of Logic data within the data to be replayed.
Sample query : :REPLAY:LOGIC:FUNCTION?
Sample response : :REPL:LOG:FUNC ON

Command : **:REPLay:LOGic:THReshold<NR1>?**
Function : Requests the Threshold level of the Logic data to be replayed. <NR1> represents a Logic Group No.
Sample query : :REPLAY:LOGIC:THRESHOLD1?
Sample response : :REPL:LOG:THR1 TTL

Command : **:REPLay:MEASure:MODE?**
Function : Requests the measurement mode which was used when capturing the data to be replayed.
Sample query : :REPLAY:MEASURE:MODE?
Sample response : :REPL:MEAS:MODE REC

Command : **:REPLay:TRIGger:POINT?**
Function : Requests the position of the trigger point within the data to be replayed. The response indicates the trigger point's position as an ordinal number relative to the first point.
Sample query : :REPLAY:TRIGGER:POINT?
Sample response : :REPL:TRIG:POINT 125

Command : **:REPLay:OUTPut:TYPe**
Function : Specifies the output format for data output via an interface.
Sample setting : :REPLAY:OUTPUT:TYPE TEXT
Sample response : :REPL:OUTP:TYP BIN

NOTE

The output format can be set to one of three formats: BINary, TEXT, or CSV.

Command : :REPLay:OUTPut:DATA<NR1>,<NR1>
Function : Specifies the starting point for data output via an interface (as the number of points from the first point) and the total number of data points to be output.
Sample setting : :REPLAY:OUTPUT:DATA 0,1000

Command : :REPLay:OUTPut:DATA
Function : Executes the output of data.
Sample setting : :REPLAY:OUTPUT:DATA

NOTE

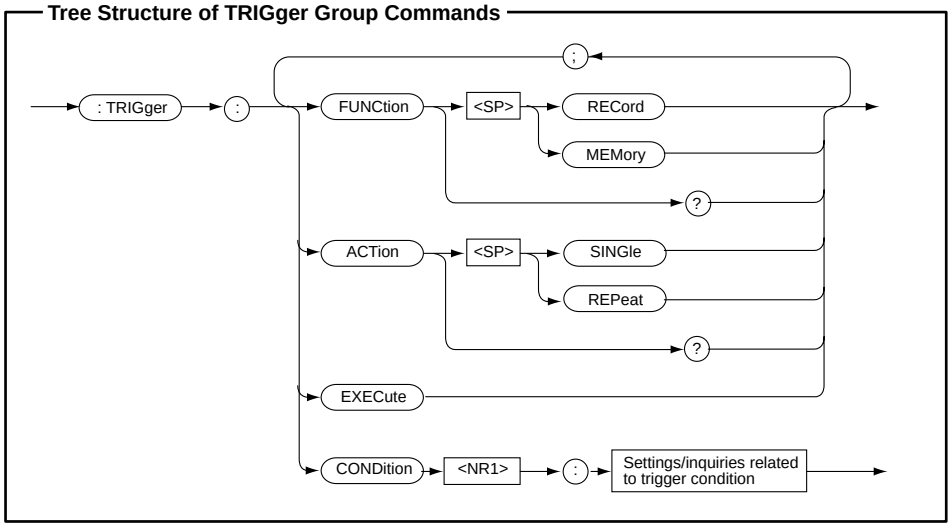
This is an executable command, so be careful when composing its program message.

Command : :REPLay:STOP
Function : Specifies return to a measurement mode (such as RECORDER mode).
Sample setting : :REPLAY:STOP

3.6 Trigger-Related Settings (TRIGGER Group)

The TRIGGER Group is the Command group related to the recorder's trigger settings.
TRIGGER Group Commands can be used to set or request trigger-related settings. For information about the setting procedures and restrictions, refer to the "WR1000 USER'S MANUAL."

3.6.1 Summary of TRIGGER Group 1/2 Commands



Command : :TRIGger:FUNCtion
Function : Enables the trigger function and selects its setting.
Sample setting : :TRIGGER:FUNCTION MEMORY
Sample query : :TRIGGER:FUNCTION?
Sample response : :TRIG:FUNC REC

NOTE

RECord or MEMory can be selected as the program data for the Trigger Function setting. RECord controls the start/stop of recording, whereas MEMory controls the start/stop of the waveform recording of memory data.

Command : :TRIGger:ACTion
Function : Selects either single or repeated trigger action.
Sample setting : :TRIGGER:ACTION SINGLE
Sample query : :TRIGGER:ACTION?
Sample response : :TRIG:ACT REP

NOTE

SINGLe or REPeat can be selected as the program data for the Trigger Action setting.

Command : :TRIGger:EXECute
Function : Has the same effect as pressing the recorder's TRIGGER key.
Sample setting : TRIGGER:EXECUTE

NOTE

When the Trigger setting is Off and the Trigger Source setting is Manual, this Command activates the trigger.

Command : :TRIGger:CONDition<NR1>
Function : Specifies the trigger source for Trigger Condition 1 (the start trigger) or Trigger Condition 2 (the stop trigger).
<NR1> indicates the Start or Stop Condition. Input 0 to select the Start Condition, or input 1 to select the Stop Condition.
Sample setting : :TRIGGER:CONDITION 1:SOURCE OFF

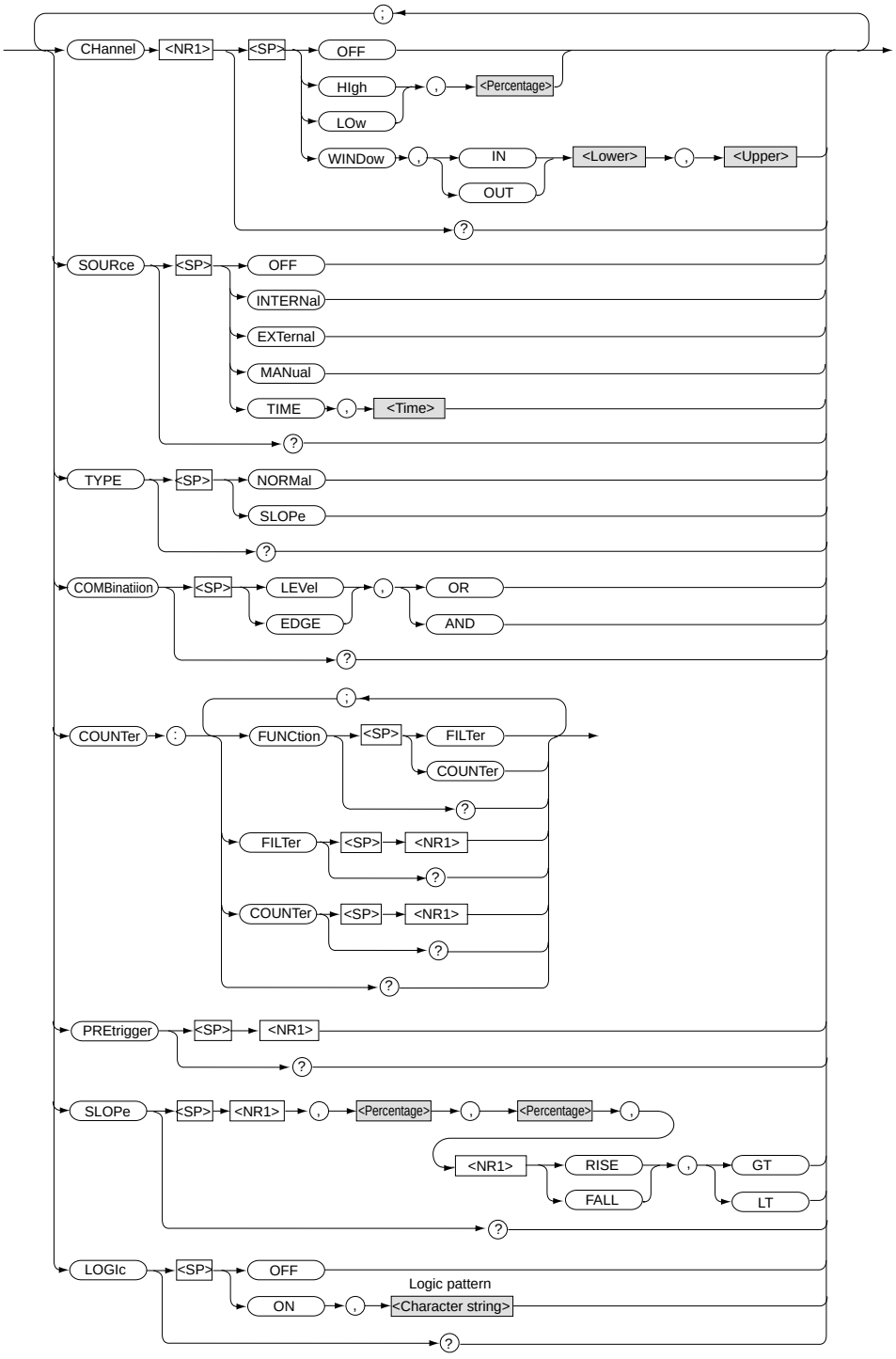
NOTE

The parameter settings that follow CONDITION<NR1> consist of multiple branches and auxiliary parameters.
Possible settings for Trigger Conditions 1 and 2 are numerous and also have subordinate Commands. For details, see Subsection 3.6.2, "Summary of TRIGGER Group 2/2 Commands."

3.6.2 Summary of TRIGGER Group 2/2 Commands

This subsection provides a detailed description of the subordinate Commands under the Command for specifying Trigger Conditions 1 and 2 (the start and stop triggers).

Tree Structure of TRIGGER Group Commands



Command : :TRIGger:CONDition<NR1>:CHannel<NR1>
Function : Selects or requests the trigger mode of the specified trigger channel used by the specified trigger Condition (1 or 2). CONDition<NR1> represents the Start Condition (0) or Stop Condition (1). CHannel<NR1> represents a trigger channel number, which can be specified as any integer from 1 to 32.
Sample setting : :TRIGGER:CONDITION 1:CHANNEL1 OFF
Sample query : :TRIGGER:CONDITION 1:CHANNEL1?
Sample response : :TRIG:COND1:CH1 OFF

NOTE

The program data which can be used as this Command's settings are listed below.

Program data	Specifiable range
CHannel<NR1>	<NR1> 1 to 32
OFF HIgh LOw WINDow	OFF Disabled as a trigger channel HIGH Waveform data exceeds the specified value LOW Waveform data goes below the specified value WINDOW,IN, Waveform data goes inside the upper- and lower-limit values WINDOW,OUT, Waveform data goes outside the upper- and lower-limit values
Percentage	±100% Example: 5V RANGE -5V → -100% +5V → +100%
<NRf>	Within the current RANGE setting Example: 5V RANGE → -5V to +5V

Command : :TRIGger:CONDition<NR1>:SOURce
Function : Selects or requests the trigger source for use with Trigger Condition 1 or 2.
Sample setting : :TRIGGER:CONDITION 1:SOURCE OFF
Sample query : :TRIGGER:CONDITION 1:SOURCE?
Sample response : :TRIG:COND1:SOUR OFF

NOTE

- OFF, INTERNal, EXTernal, MANual, or TIME can be selected as the program data for this Command's setting.
- If TIME is selected, specify the TIME setting as a decimal value between 1 and 32000 in <NR1> format. The value will be internally processed by the recorder as a time interval in second units.

Function setting	Description
OFF	Disables the trigger function.
INTERNal	Activates the trigger according to the recorder's internal settings.
EXTernal	Activates the trigger according to an external contact point or the TTL level.
MANual	Activates the trigger when the TRIGGER key is pressed or the recorder receives a :TRIGGER:EXECUTE Command.
TIME	Stops measurement at the specified time (only valid for the Stop Trigger). :TRIGGER:CONDITION1:SOURCE TIME,60

Command : :TRIGger:CONDition<NR1>:TYPE
Function : Selects or requests the Trigger Type to be used by the Start or Stop Condition.
Sample setting : :TRIGGER:CONDITION 1:TYPE NORMAL
Sample query : :TRIGGER:CONDITION 1:TYPE?
Sample response : :TRIG:COND1:TYPE NORM

NOTE

NORMal or SLOPe can be selected as the program data for the TYPE setting. The NORMAL setting enables the **:TRIGger:CONDition<NR1>:CHannel<NR1>** Command. If SLOPe is selected, the activation of the trigger is determined by the change in waveform data. Use the separate Slope Command to define the range within which waveform data changes over time and a specific channel.

Command : :TRIGger:CONDition<NR1>:COMBination
Function : Selects or requests the Trigger Combination setting of the channel(s) specified as the trigger channel(s) for Trigger Condition 1 or 2.
Sample setting : :TRIGGER:CONDITION 1:COMBINATION LEVEL,OR
Sample query : :TRIGGER:CONDITION 1:COMBINATION?
Sample response : :TRIG:COND1:COMB LEV,OR

NOTE

LEVel or EDGE can be selected as the program data for the COMBINATION setting. This Command must be specified even if only one trigger channel is specified by the **:TRIGger:CONDition<NR1>:CHannel<NR1>** Command. If only one trigger channel is specified, however, only that single trigger is valid, regardless of the AND or OR setting.

Command : :TRIGger:CONDition<NR1>:COUNTer:?
Function : Requests the currently selected settings of the Trigger Count function.
Sample query : :TRIGGER:CONDITION1:COUNTER:?
Sample response : :TRIG:COND1:COUNT:FUNC FILT;FILT 2;COUNT 4

NOTE

Because the Trigger Counter and Trigger Filter settings are separately registered in the recorder's internal memory, the two settings are separately returned in response to this command.

Command : :TRIGger:CONDition<NR1>:COUNTer:FUNCtion?
Function : Specifies or requests the Trigger Counter setting.
Sample setting : :TRIGGER:CONDITION1:COUNTER:FUNCTION COUNTER
Sample query : :TRIGGER:CONDITION1:COUNTER:FUNCTION?
Sample response : :TRIG:COND1:COUNT:FUNC FILT

Command : :TRIGger:CONDition<NR1>:COUNTer:FILTer<NR1>
Function : Specifies or requests the setting of the Trigger Counter's Filter function.
Sample setting : :TRIGGER:CONDITION1:COUNTER:FILTER 100
Sample query : :TRIGGER:CONDITION1:COUNTER:FILTER?
Sample response : :TRIG:COND1:COUNT:FILT 100

NOTE

This command's FILTer parameter can be specified as a value from 1 to 255.

Command : :TRIGger:CONDition<NR1>:COUNTer:COUNTer<NR1>
Function : Specifies or requests the setting of the Trigger Counter's Counter function.
Sample setting : :TRIGGER:CONDITION1:COUNTER:COUNTER 100
Sample query : :TRIGGER:CONDITION1:COUNTER:COUNTER?
Sample response : :TRIG:COND1:COUNT:COUNT 100

NOTE

This command's COUNTer parameter can be specified as a value from 1 to 255.

Command : :TRIGger:CONDition<NR1>:PREtrigger<NR1>
Function : Specifies or requests the Delay setting (which can only be set for the Start Condition).
Sample setting : :TRIGGER:CONDITION0:PRETRIGGER 0
Sample query : :TRIGGER:CONDITION0:PRETRIGGER?
Sample response : :TRIG:COND0:PRE 0

NOTE

- This command's PRETRIGGER parameter can be specified as a value from 0 to 100.
- This command is ignored if the Save To setting is Disk (when data will be saved to a hard disk or MO disk). In this case, the PRETRIGGER setting will always be regarded as being zero.

Command : :TRIGger:CONDition<NR1>:SLOPe<NR1>
Function : Specifies or requests the Slope Trigger.
Sample setting : :TRIGGER:CONDITION1:SLOPE 1,10,20,100,RISE,GT
Sample query : :TRIGGER:CONDITION1:SLOPE?
Sample response : :TRIG:COND1:SLOP 1,10,20,100,FALL,LT

NOTE

The following table lists the program data which can be used with this command, in an order from the Lower to the Upper limit.

Program data	Description
<NR1>	Specifies a channel number from 1 to 32.
<Percentage>	Specifices the Lower dV value (0 to 100%).
<Percentage>	Specifies the Upper dV value (0 to 100%).
<NR1>	Specifies the dT setting (1 to 255)
RISE	Specifies a rising slope for trigger activation.
FALL	Specifies a falling slope for trigger activation.
GT	The trigger is activated when the time interval during which the waveform passes from the Lower value to the Upper value is longer than the dT setting.
LT	The trigger is activated when the time interval during which the waveform passes from the Lower value to the Upper value is shorter than the dT setting.

Command : :TRIGger:CONDition<NR1>:LOGIc
Function : Specifies or requests the Logic Trigger condition for the Start or Stop Condition.
Sample setting : :TRIGGER:CONDITION1:LOGIC ON,"HHLLxxxxHHLLHHLL"
Sample query : :TRIGGER:CONDITION:LOGIC?
Sample response : :TRIG:COND1:LOGI ON,"HHLLxxxxHHLLHHLL"

NOTE

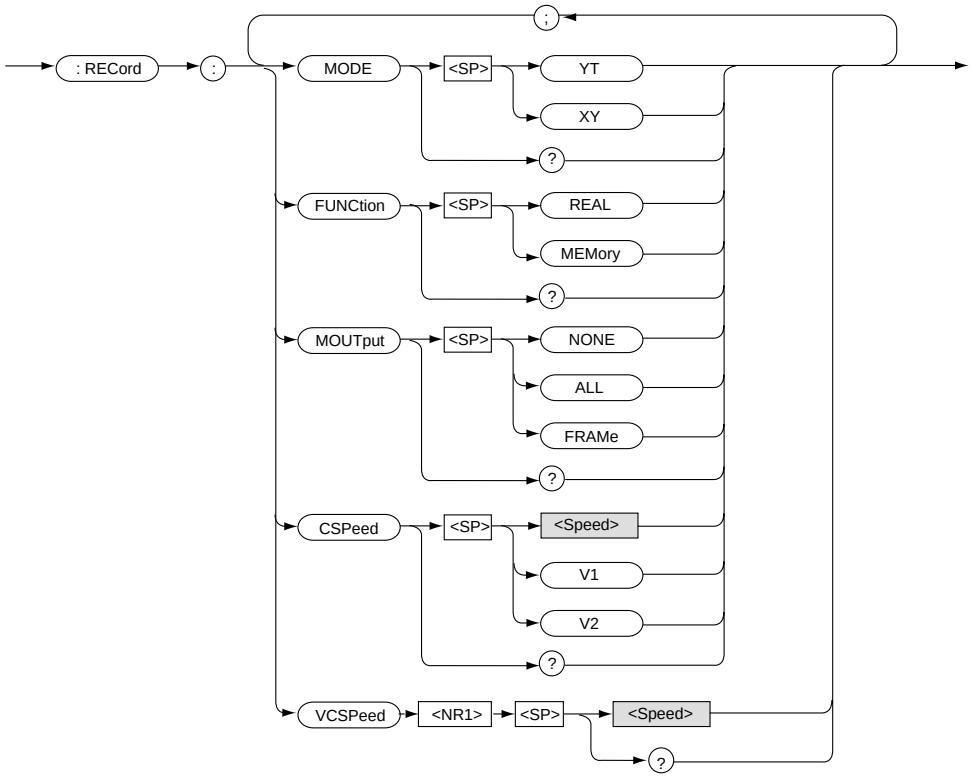
The following table lists the program data which can be used with this command.

Program data	Permissible settings (Description)
OFF ON	OFF : The Logic trigger is disabled. ON : The Logic trigger is enabled.
Character string	The character string is arranged by Logic Group No. as follows. A B C D 1234 1234 1234 1234 This parameter is fixed to a 16-character setting and any "x" symbol in the character string represents an invalid bit.

3.7 Settings Related to Recording (RECORD Group)

The RECORD Group is the Command group related to the recorder's recording settings.
RECORD Group Commands can be used to set or request recording-related settings. For information about the setting procedure and restrictions for recording formats, refer to the "WR1000 USER'S MANUAL."

Tree Structure of RECORD Group Commands



3.7.1 Summary of RECORD Group Commands

Command : :RECord:MODE
Function : Selects or requests the display and recording mode for measured data (YT or XY).
Sample setting : :RECORD:MODE YT
Sample query : :RECORD:MODE?
Sample response : :REC:MODE YT
Sample application: Setting :RECORD:MODE XY,1,2,3,4,5,6,7,8,9

NOTE

The sample application above sets the MODE setting to XY. The channel specified in the initial program data is regarded as the X axis, whereas the subsequent channels are regarded as Y1, Y2, Y3 through to Y8.

Command : :RECord:FUNCtion
Function : Selects whether to perform realtime (REC) or memory output (MEM) of the recorded data or requests the current FUNCTION setting.
Sample setting : :RECORD:FUNCTION RECORD
Sample query : :RECORD:FUNCTION?
Sample response : :REC:FUNC REC

Command : :RECord:MOUTput
Function : Selects or requests the range for recording memory data.
Sample setting : :RECORD:MOUTPUT ALL
Sample query : :RECORD:MOUTPUT?
Sample response : :REC:MOUT ALL

NOTE

NONE (recording only), ALL (output of all data), or FRAME (output of a single frame) can be selected as the program data for this Command's setting.

Command : :RECord:CSPeed
Function : Selects or requests the chart speed for direct recording.
Sample setting : :RECORD:CSPEED 10mm_s
Sample query : :RECORD:CSPEED?
Sample response : :REC:CSP 10MM_s

NOTE

As the program data for the CSPEED setting, you can select one of three types of chart speeds: (1) a chart speed listed in Subsection 3.7.2, "Settings and Replies Related to the Chart Speed," (2) V_one (V1), or (3) V_two (V2). The V_one or V_two setting enables a chart speed that has been pre-defined by the user.

Command : :RECord:VCSPeed<NR1>
Function : Selects or requests the user-defined chart speed of V1 or V2 for use during direct recording.
For <NR1>, input 1 to select the V1 setting or input 2 to select the V2 setting.
Sample setting : :RECORD:VCSPEED1 4mm_s
Sample query : :RECORD:VCSPEED1?
Sample response : :REC:VCSP 4MM_s

3.7.2 Settings and Replies Related to the Chart Speed

Settings (Program data)	Replies (Response data)
1 mm_s	1 MM_S
2 mm_s	2 MM_S
5 mm_s	5 MM_S
10 mm_s	10 MM_S
20 mm_s	20 MM_S
50 mm_s	50 MM_S
100 mm_s	100 MM_S
200 mm_s	200 MM_S
1 mm_min	1 MM_M
2 mm_min	2 MM_M
5 mm_min	5 MM_M
10 mm_min	10 MM_M
20 mm_min	20 MM_M
50 mm_min	50 MM_M
100 mm_min	100 MM_M
200 mm_min	200 MM_M
1 mm_h	1 MM_H
2 mm_h	2 MM_H
5 mm_h	5 MM_H
10 mm_h	10 MM_H
20 mm_h	20 MM_H
50 mm_h	50 MM_H
100 mm_h	100 MM_H
200 mm_h	200 MM_H
V_one	V_one
V_two	V_two
EXT	EXT

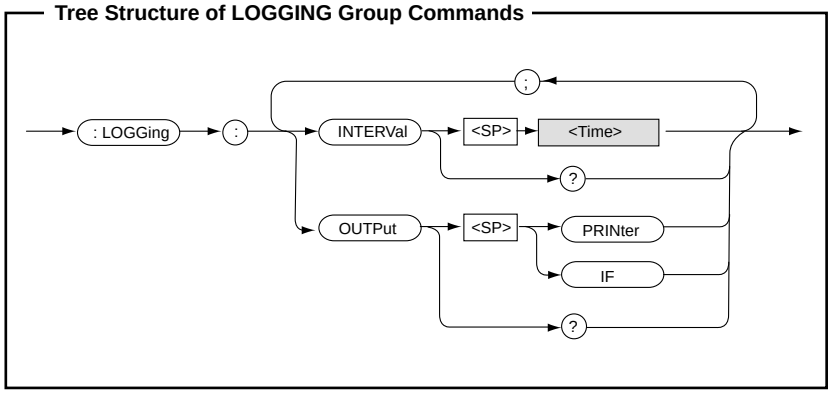
NOTE

For details on the permissible settings for V_one and V_two, refer to the description of the V1 and V2 settings in the WR1000 USER'S MANUAL. The chart speed can be specified as mm_s, mm_min, or mm_h.

3.8 Settings Related to Logging (LOGGING Group)

The LOGGING Group is the Command group related to the recorder's logging settings.
LOGGING Group Commands can be used to set or request settings related to the logging format. For information about the setting procedures and restrictions regarding the logging format, refer to the "WR1000 USER'S MANUAL."

Tree Structure of LOGGING Group Commands



3.8.1 Summary of LOGGING Group Commands

Command : :LOGGing:INTERVal
Function : Specifies or requests the interval for logging recording.
Sample setting : :LOGGING:INTERVAL 1s
Sample query : :LOGGING:INTERVAL?
Sample response : :LOGG:INTERV 1s

NOTE
Any integral value from 1s to 3600s can be specified as the program data for this Command's setting.

Command : :LOGGing:OUTPut
Function : Selects or requests the output destination of logging data (as output from the recorder's printer or its interface).
Sample setting : :LOGGING:OUTPUT PRINTER
Sample query : :LOGGING:OUTPUT?
Sample response : :LOGG:OUTP IF

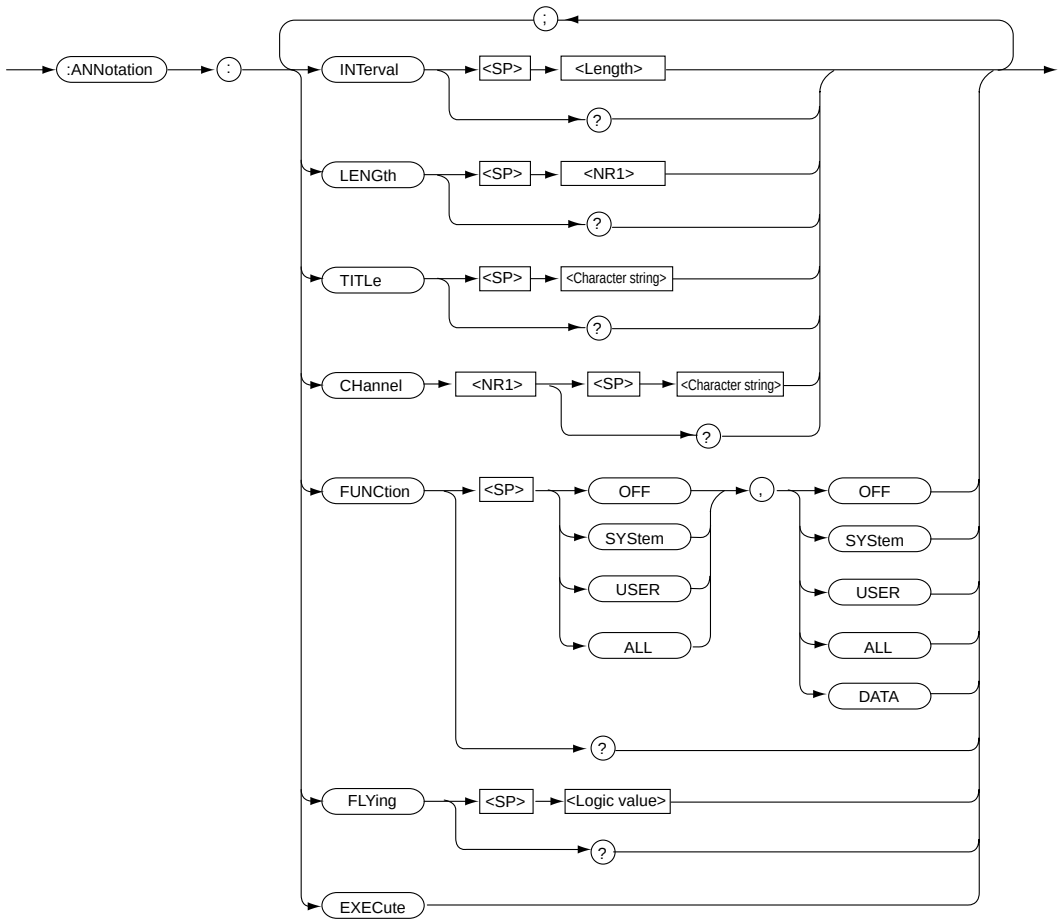
NOTE
Interface output cannot be specified by a command. Make this setting at the recorder.

3.9 Annotation Settings (ANNOTATION Group)

The ANNOTATION Group is the Command group related to the recorder's Annotation settings.

ANNOTATION Group Commands can be used to set or request settings related to annotation. For information about the setting procedures and restrictions regarding annotation, refer to the "WR1000 USER'S MANUAL."

Tree Sturecture of ANNOTATION Group Commands



3.9.1 Summary of ANNOTATION Group Commands

Command : :ANNotation:INTerval
Function : Specifies or requests the interval for repeatedly printing annotation.
Sample setting : :ANNOTATION:INTERVAL 30
Sample query : :ANNOTATION:INTERVAL?
Sample response : :ANN:INT 30

NOTE

As the program data, this Command can be set to 10, 20, 30, 40, 50, 60, 70, 80, 90, or 100. The unit is internally fixed to "cm" at the recorder.

Command : :ANNotation:LENGth
Function : Specifies or requests the maximum length of the annotation character string.
Sample setting : :ANNOTATION:LENGTH 63
Sample query : :ANNOTATION:LENGTH?
Sample response : :ANN:LENG 63

NOTE

Any integer from 10 to 63 can be specified as the program data for this Command's setting.

Command : :ANNotation:TITLe
Function : Specifies or requests the character string for use in Title Annotation, provided that the Title Annotation setting is either User or System & User.
Sample setting : :ANNOTATION:TITLE "WR1000 produced by Graphtec Corp."
Sample query : :ANNOTATION:TITLE?
Sample response : :ANN:TITL "WR1000 produced by Graphtec Corp."

NOTE

The number of characters which can be input is determined by the character string length specified by the **:ANNotation:LENGth** Command and by the **ANNotation:FUNCtion** Command.

Command : :ANNotation:CHannel
Function : Specifies or requests the user-defined character string for use in Channel Annotation for the specified channel.
Sample setting : :ANNOTATION:CHANNEL1 "Channel-1 Temp."
Sample query : :ANNOTATION:CHANNEL1?
Sample response : :ANN:CH1 "Channel-1 Temp."

NOTE

The number of characters which can be input is determined by the character string length specified by the **:ANNotation:LENGth** Command and by the **ANNotation:FUNCtion** Command.

Command : :ANNotation:FUNCtion
Function : Specifies or requests the information to be printed during Title Annotation and Channel Annotation.
Sample setting : :ANNOTATION:FUNCTION SYStem,SYStem
Sample query : :ANNOTATION:FUNCTION?
Sample response : :ANN:FUNC SYS,SYS

NOTE

For information about the functions which can be selected as the program data, see Subsection 3.9.2, "Supplementary Information on Annotation Printing."

Command : :ANNotation:FLYing
Function : Specifies or requests the On/Off status for controlling the printing of Title Annotation and Channel Annotation via the interface.
Sample setting : :ANNOTATION:FLYING OFF
Sample query : :ANNOTATION:FLYING?
Sample response : :ANN:FLY OFF

NOTE

- The printing of Title or Channel Annotation can be controlled in one of two ways.
  OFF : Printing is conducted according to the Annotation setting.
  ON : Printing is controlled via the interface. In this case, annotation is printed by issuing the **:ANNOTATION:EXECUTE** command.
- For more information about this command, see Subsection 3.9.2, "Supplementary Information on Annotation Printing."

Command : :ANNotation:EXECute
Function : Inputs the character strings to be printed as the Title Annotation and Channel Annotation, then executes Annotation Printing.
Sample setting : :ANNOTATION:EXECUTE

NOTE

For more information about this Command, see Subsection 3.9.2.2, "Printing Control Via the Interface."

3.9.2 Supplementary Information on Annotation Printing

3.9.2.1 Setting the Annotation Functions

The tables below describe the available :ANNotation:FUNCtion settings.

Title Annotation	
FUNCtion setting	**Contents printed**
OFF	Disables Title Annotation.
SYStem	Prints the date, time, and chart speed.
USER	Prints the user-defined character string.
ALL	Prints the date, time, chart speed, and user-defined character string.

Channel Annotation	
FUNCtion setting	**Contents printed**
OFF	Disables Channel Annotation.
AMP	Prints the amp settings.
USER	Prints the user-defined character string.
ALL	Prints the amp settings and user-defined character string.
DATA	Prints the measured data.

NOTE

- If the FUNCtion setting is ALL for both Title Annotation and Channel Annotation, the maximum length of each user-defined character string is 32 characters.
- When the character string length has been limited by the **:ANNotation:LENGth** Command, SYSTEM annotation printing takes priority so the restriction on the character string length first affects the user-defined character strings.

3.9.2.2 Printing Control Via the Interface

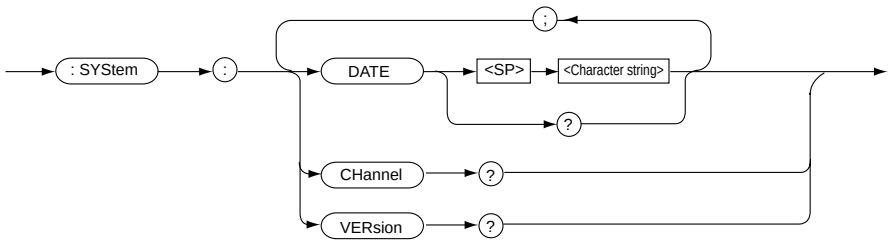
When the :ANNotation:FLYing command is set to ON, printing can be controlled via the interface. In such case, the recorder operates exactly as if all of its internal print modes are in Off status.
Input one user-defined character string each for Title Annotation and Channel Annotation via the interface, then issue the **:ANNotation:EXECute** Command to execute printing. Note that the setting of the **:ANNotation:LENGth** Command determines the maximum length of each character string.
While the chart speed is being controlled at a low speed, please note that character strings cannot be input via the interface for use as Title Annotation and Channel Annotation.
Control the transfer and printing of Annotation character strings as described in Subsection 4.3, “The Extended Event Register.”

3.10 System Settings (SYSTEM Group)

The SYSTEM Group is the Command group related to the recorder's system settings.

SYSTEM Group Commands can be used to set or request system settings. For information about the setting procedures and restrictions regarding system settings, refer to the "WR1000 USER'S MANUAL."

Tree Structure of SYSTEM Group Commands



3.10.1 Summary of SYSTEM Group Commands

Command : :SYStem:DATE
Function : Specifies or requests the current date and time of the recorder's internal clock function.
Sample setting : :SYSTEM:DATE 1999/08/27 12:14:15
Sample query : :SYSTEM:DATE?
Sample response : :SYS:DATE 1999/08/27 12:34:45

NOTE

- The date and time cannot be changed during measurement.
- As the program data, the year is specified as four digits, whereas the month, day, hour, minute, and second are specified as two digits each. The time is specified using the 24-hour format.

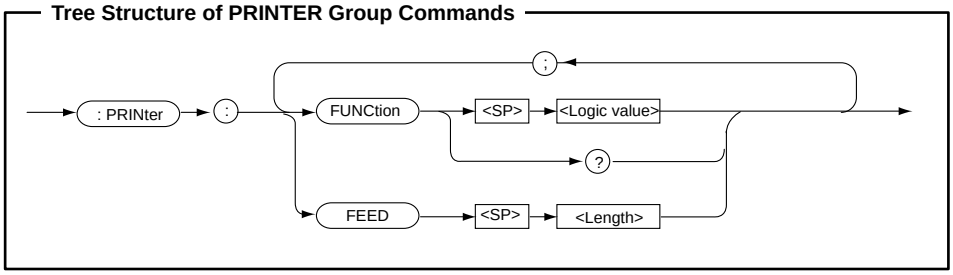
Command : :SYStem:CHannel?
Function : Requests the channel configuration of the recorder.
Sample query : :SYSTEM:CHANNEL?
Sample response : :SYS:CH 8

Command : :SYStem:VERsion?
Function : Requests the version number of the recorder's firmware.
Sample query : :SYSTEM:VERSION?
Sample response : :SYS:VER VER1.00,1.00,1.00,1.00

3.11 Printer Settings (PRINTER Group)

The PRINTER Group is the Command group related to the recorder's printing functions.
PRINTER Group Commands can be used to set or request settings related to printing. For information about printing-related procedures and restrictions, refer to the "WR1000 USER'S MANUAL."

Tree Structure of PRINTER Group Commands



3.11.1 Summary of PRINTER Group Commands

NOTE

Printer operation cannot be specified when the recorder is in Paper End status or is performing a measurement operation. Confirm the recorder's status with the Extended Event Register before executing these commands.

Command : :PRINter:FUNCtion
Function : Specifies or requests the On/Off status of the recorder's Printer function.
Sample setting : :PRINTER:FUNCTION ON
Sample query : :PRINTER:FUNCTION?
Sample response : :PRIN:FUNC ON

Command : :PRINter:FEED
Function : Specifies the distance for feeding the printer then feeds the chart paper the specified length.
Sample setting : PRINTER:FEED 200

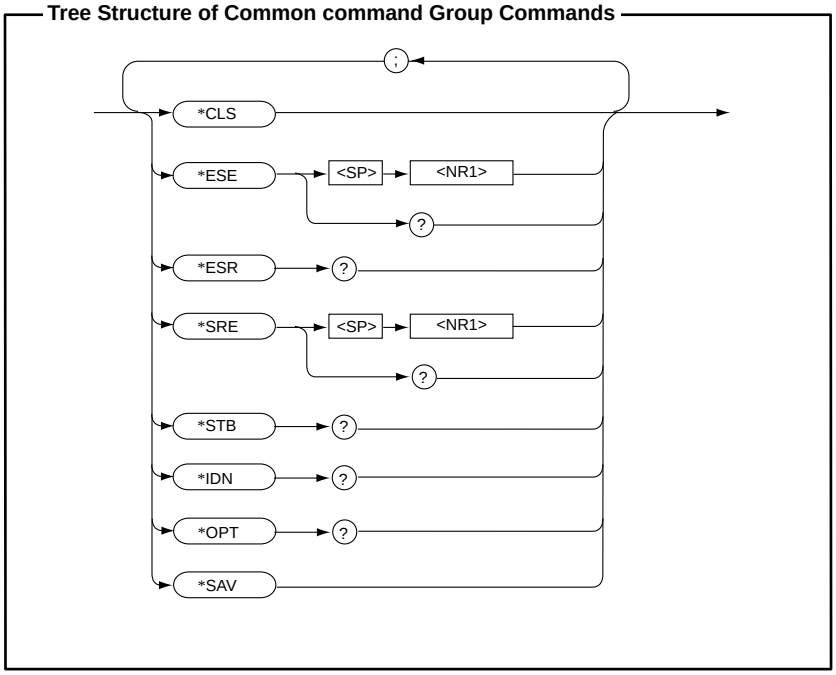
NOTE

Any integer can be specified as the program data, but the unit is internally fixed to "mm" at the recorder.
This Feed Command is ignored while the recorder is measuring data.
This is an executable command, so be careful when composing its program message.

3.12 Common Command Settings (COMMON Group)

The COMMON Group is the Command group related to the checking and setting of the recorder's operating status.
COMMON Group Commands have been created with reference to the common Command group stipulated in the IEEE488.2-1987 standard.
Use of these Commands will provide you with an accurate grasp of the recorder's internal processing status and error status, enabling you to synchronize the recorder with the controlling device and perform other operations.
While reading the description of the COMMON Group Commands, also see Chapter 4, "OUTPUTTING THE RECORDER'S STATUS INFORMATION."

Tree Structure of Common command Group Commands



3.12.1 Summary of COMMON Group Commands

Command : *CLS
Function : Clears the standard event register, extended event register, and error queue.
Sample setting : *CLS

Command : *ESE <NR1>
Function : Specifies or requests the standard event enable register.
Sample setting : *ESE 255
Sample query : *ESE?
Sample response : *ESE 255

NOTE

- This standard event enable register setting lets you mask the bits of the standard event register. When this Command is used to write a zero to the bit to be masked, it becomes subject to a logical AND operation with the standard event register's corresponding bit. As a result, the masked bit of the standard event register cannot enter the enable query status expressed by "1".
- When 247 is specified as the program data, Bit 3 is masked so that the status byte register's Bit 5 (ESB) will not change to "1" even if the recorder enters a DDE (Device Error) state.
- The default setting of 255 enables all bits. Moreover, the standard event enable register's setting will not be cleared even after being read out in response to the ***ESE** query.

Command : *ESR?
Function : Requests the value of the standard event register.
Sample query : *ESR?
Sample response : *ESR 16

NOTE

- Upon receipt of a request for the standard event register's value, the recorder outputs the result of a logical AND operation between the standard event enable register setting and the standard event register.
  When this standard event enable register setting is 255 so that all bits are enabled, an Execution Error may occur in some cases if the response data to the ESR query is 16.
- The standard event register's value will be cleared after being read out in response to the ***ESR?** query.

Command : *SRE <NR1>
Function : Specifies or requests the value of the service request enable register.
Sample setting : *SRE 255
Sample query : *SRE?
Sample response : *SRE 255

NOTE

- This service request enable register setting lets you mask the bits of the status byte register. When this Command is used to write a zero to the bit to be masked, it becomes subject to a logical AND operation with the status byte register's corresponding bit. As a result, the status byte register's bit that has been masked by the *SRE Command cannot enter the enable query status expressed by "1". Also note that Bit 6 cannot be set to zero.
- When 251 is specified as the program data, Bit 2 is masked so that the status byte register's Bit 2 (ESB) will not change to "1" even if the recorder enters the EAV (Error Available) state.
- The default setting of 255 enables all bits. Moreover, the service request enable register's setting will not be cleared even after being read out in response to the ***SRE** query.

Command : *STB?
Function : Requests the value of the status byte register.
Sample query : *STB?
Sample response : *STB 72

NOTE

- Upon receipt of a request for the status byte's settings, the recorder outputs the result of a logical AND operation between the service request enable register setting and the status byte register. When this status byte register setting is 255 so that all bits are enabled, if Bit 3 (EES: Extend Event Summary bit) changes to 1, the recorder outputs 1 for Bit 3 and for Bit 6 (Master Summary bit) and outputs 72 as the response data.
- The status byte register's setting will not be cleared even after being read out in response to the ***STB?** query.

Command : *IDN?
Function : Requests the recorder's model ID.
Sample query : *IDN?
Sample response : *IDN Graphtec,WR1000,0.1.11

NOTE

As response data, the recorder outputs the manufacturer's name, the model number, and the firmware's version number.

Command : *OPT
Function : Requests which options are installed.
Sample query : *OPT?
Sample response : *OPT LOGIC, AUDIO

NOTE

- The options are displayed as follows:
  - Logic amp : **LOGIC**
  - Extension memory : **EXTMEM**
  - Sound card : **AUDIO**
  - GP-IB interface : **GPIB**
  - SCSI interface : **SCSI**
  - External SCSI interface : **EXTSCSI**
- If no options are installed, the response is:
  ***OPT NONE**

Command : *SAV
Function : Registers the recorder's currently selected settings.
Sample setting : *SAV

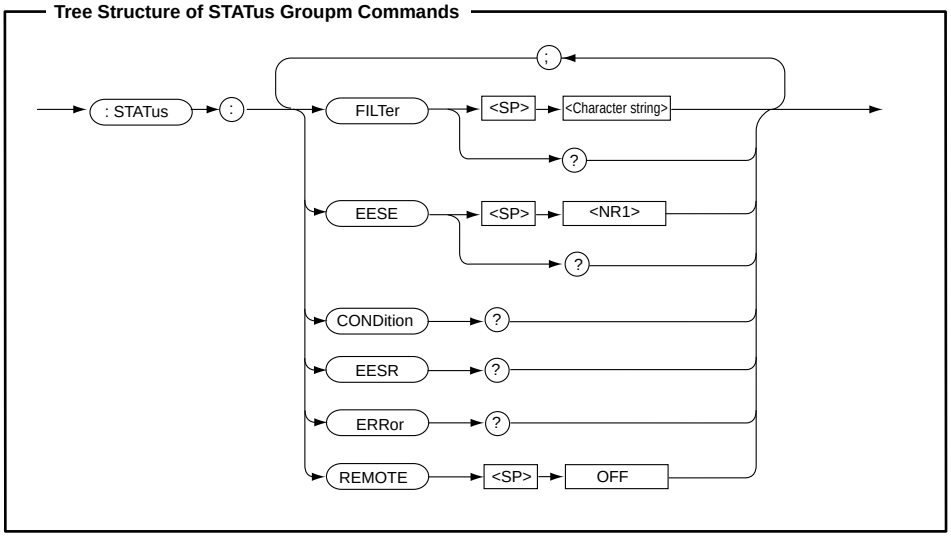
NOTE

- Whenever a setting is changed by command input via the interface, the new setting will not be registered in the recorder's internal memory as in the case of changing a setting by operating the recorder's panel key(s).
- To retain a setting that has been specified via the interface so that it remains in the recorder's internal memory even while the recorder is turned off, you will need to execute this command.
- There is a limit to the number of times the recorder's non-volatile memory (where the recorder's settings are registered) can be rewritten. If you plan to use the ***SAV** command within programmed data, take care to reduce the rewriting of the non-volatile memory to the lowest possible frequency.

3.13 Interface Status Control (STATus Group)

The IF (Interface Status Control) Group is the Command group used to read out the extended event register's status for conducting a detailed check of the recorder's status. IF Group Commands are closely connected to the status byte register which is first described in Section 3.12, "Common Command Settings (COMMON Group)."
Use of these Commands to read out the extended event register's status will provide you with a grasp of the recorder's internal processing status and error status, enabling you to synchronize the recorder with the controlling device and perform other operations.
While reading the description of the IF Group Commands, also see Chapter 4, "OUTPUTTING THE RECORDER'S STATUS INFORMATION."

Tree Structure of STATus Groupm Commands



3.13.1 Summary of STATUS Group Commands

Command : :STATus:FILTer
Function : Selects the type of status change in the device condition register's value and then rewrites the extended event register's bits whenever the selected type of status change occurs.
Sample setting : :STATUS:FILTER RISE
Sample query : :STATUS:FILTER?
Sample response : :STAT:FILT FALL

NOTE

RISE, FALL, BOTH, or NEVer can be selected as the program data for this Command's setting. Each option is described below.

Program data	Description of the status change filter
RISE	When a bit in the device condition register changes from zero to 1, the extended event register's corresponding bit is set to 1.
FALL	When a bit in the device condition register changes from 1 to zero, the extended event register's corresponding bit is set to 1.
BOTH	When a bit in the device condition register changes from zero to 1 or visa versa, the extended event register's corresponding bit is set to 1.
NEVer	The extended event register's bits are fixed to zero.

Command : :STATus:EESE <NR1>
Function : Specifies or requests the value of the extended event enable register.
Sample setting : :STATUS:EESE 65535
Sample query : :STATUS:EESE?
Sample response : :STAT:EESE 65535

NOTE

- This extended event enable register setting lets you mask the bits of the extended event register. When this Command is used to write a zero to the bit to be masked, it becomes subject to a logical AND operation with the extended event register's corresponding bit. As a result, the masked bit of the extended event register cannot enter the enable query status expressed by "1".
- When 65534 is specified as the program data, Bit 0 is masked so that the status byte register's Bit 0 (RUN) will not change to "1" even if the recorder enters a RUN state.
- The default setting of 65535 enables all bits. Moreover, the extended event enable register's setting will not be cleared even after being read out in response to the **:STATus:EESE?** query.

Command : :STATus:CONDition?
Function : Requests the value of the device condition register.
Sample query : :STATUS:CONDITION?
Sample response : :STAT:COND 1

NOTE

- Upon receipt of this request for the device condition register's value, the recorder outputs its current status. A return message of **:STAT:COND 1** signifies that the recorder is in the REC state (recording data).
- The device condition register's value will not be cleared even after being read out in response to the **:STATUS:CONDITION?** query.

Command : :STATus:EESR?
Function : Requests the value of the extended event register.
Sample query : :STATUS:EESR?
Sample response : :STAT:EESR 1

NOTE

- When this extended event register setting is 65535 to enable all bits, a return message of **:STAT:EESR1** signifies that the recorder is in the RUN status but has undergone some change. The recognized change states are determined by the setting of the status change filter.
- The extended event register's value will be cleared after being read out in response to the **:STATUS:EESR?** query. It will also be cleared after being read out in response to the *CLR query, which is a COMMON Group Command.

Command : :STATus:ERRor?
Function : Requests the error data in the error queue.
Sample query : :STATUS:ERRor?
Sample response : :STAT:ERR 18,1,1

NOTE

- The presence or absence of data in the error queue can be confirmed at the status byte register's Bit 2 (EAV).
- For a description of the error codes and error states, see Appendix B, "ERROR CODES."
- The response below is returned if no errors have occurred:
  :STAT:ERR NONE

Command : **:STATus:REMote**

Function : Switches the WR1000 to LOCAL status. When using the GP-IB interface, use the GTL (Go To Local) command instead.

Sample setting : :STATUS:REMOTE OFF

CHAPTER 4

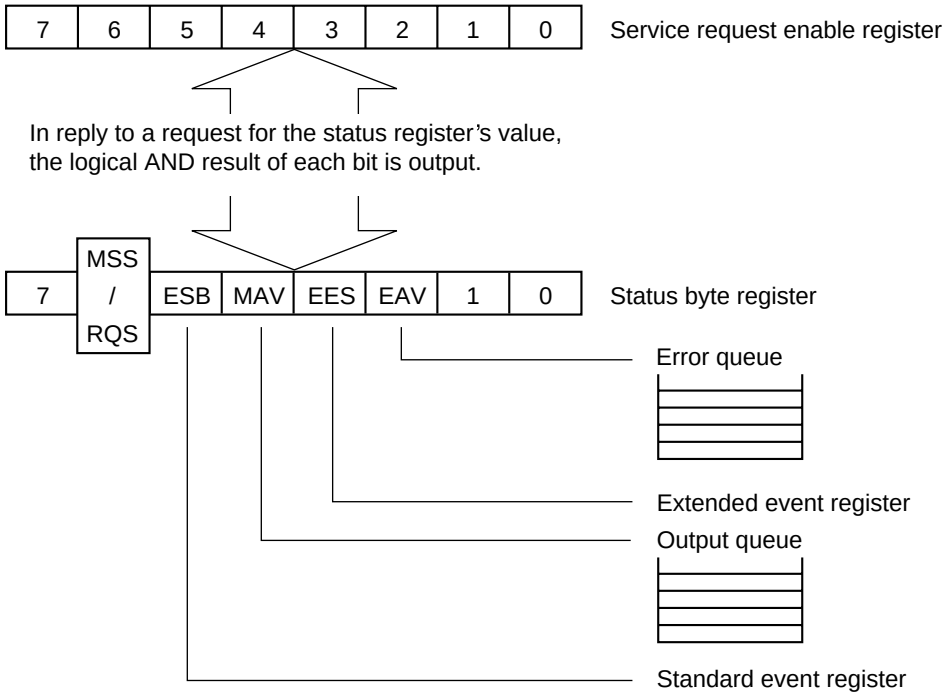
OUTPUTTING THE RECORDER'S STATUS INFORMATION

This chapter provides a description of the recorder's status information output.

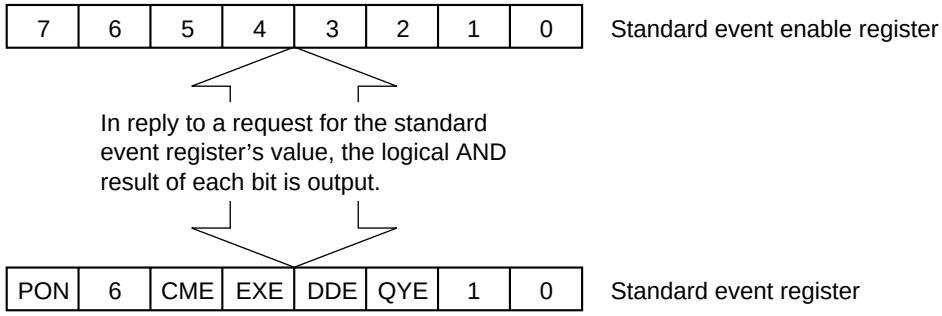
4.1 The Status Byte Register
4.2 The Status Byte
4.3 The Standard Event Register
4.4 The Extended Event Register
4.5 The Error Queue
4.6 The Output Queue

4.1 The Status Byte Register

Status output fulfills the role of reporting the recorder's operating status to the controlling device. Status output is basically constructed with reference to the format stipulated in the IEEE-488.2-1987 standard.
The status byte register represents the recorder's overall status and several of its bits may also include a register.



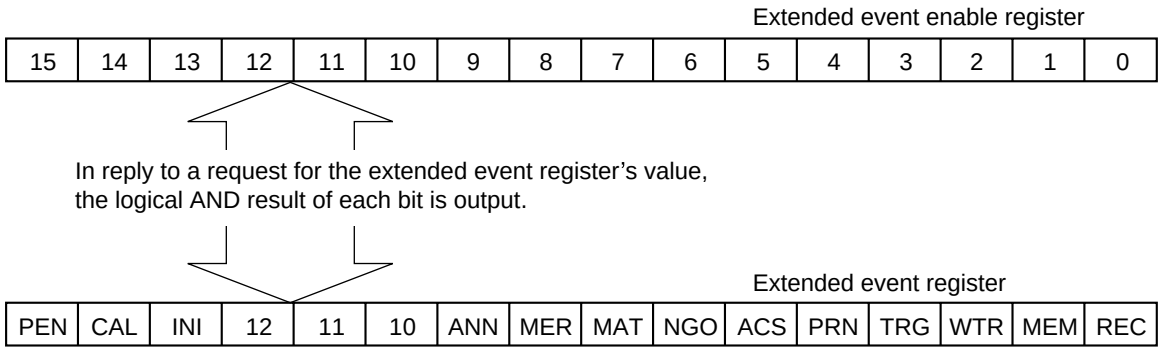
(1) Standard Event Register



NOTE

Bit 5 (ESB) of the status byte changes to 1 when the enable query status expressed by 1 is assumed by one or more bits of the standard event register. Of course, the bits of the standard event register can be masked by using the standard event enable register.

(2) Extended Event Register



NOTE

Bit 3 (EES) of the status byte changes to 1 when the enable query status expressed by 1 is assumed by one or more bits of the extended event register. Of course, the bits of the extended event register can be masked by using the extended event enable register.

4.2 The Status Byte

You can check the status byte register's values by issuing the *STB Command which is a COMMON Group Command. It is also possible to mask the bits of the service request enable register using the *SRE Command.
When the *SRE Command is issued to write a zero to the service request enable register's bit to be masked, it becomes subject to a logical AND operation with the status byte's corresponding bit. As a result, the masked bit of the status byte register cannot enter the enable query status expressed by "1".
The default setting of this recorder's service request enable register is 255, which enables all bits.

The following describes the meaning of each of the status byte register's bits (Bits 0 to 7).

Status byte register

(1) Bits 0, 1, and 7
Not used (fixed to zero)

(2) Bit 2: EAV (Error Available)
Whenever an error occurs, Bit 2 is set to 1.

(3) Bit 3: EES (Extend Event Summary Bit)
When a bit set to 1 exists as a result of a logical AND operation between the extended event register's bits and the extended event enable register's bits, Bit 3 is set to 1. That is, this bit indicates whether or not an event has occurred inside the recorder.

(4) Bit 4: MAV (Message Available)
When the recorder has received an output request from the controlling device and is ready to output response data, Bit 4 is set to 1.

(5) Bit 5: ESB (Event Summary Bit)
When a bit set to 1 exists as a result of a logical AND operation between the standard event register's bits and the standard event enable register's bits, Bit 5 is set to 1.

(6) Bit 6: RQS (ReQuest Service)/MSS (Master Status Summary bit)
Whenever Bit 2, 3, 4, or 5 of the status byte register assume enable query status, Bit 6 is set to 1. That is, this bit indicates whether or not the recorder is requesting service to the controlling device. In this status, the recorder issues a service request via the GP-IB interface.
When Bits 2, 3, 4, and 5 leave enable query status, Bit 6 is reset to zero. At the GP-IB interface, the service request is cancelled by serial polling. Serial polling does not clear Bit 6, however, which only becomes zero when Bits 2, 3, 4, and 5 are no longer in enable query status.

Condition for clearing the status byte register

The recorder is turned off then back on again. The status byte register's default setting is zero.

Condition for clearing the service request enable register

The recorder is turned off then back on again. The service request enable register's default setting is 255.

4.3 The Standard Event Register

You can simultaneously check and clear the standard event register's values by issuing the *ESR? query which is a COMMON Group Command. It is also possible to mask the bits of the standard event enable register using the *ESE Command.
When the *ESE Command is issued to write a zero to the standard event enable register's bit to be masked, therefore, it becomes subject to a logical AND operation with the standard event register's corresponding bit. As a result, the masked bit of the standard event register cannot enter the enable query status expressed by "1".
The default setting of this recorder's standard event enable register is 255, which enables all bits.

The following describes the meaning of each of the standard event register's bits (Bits 0 to 7).

PON	6	CME	EXE	DDE	QYE	1	0

Standard event register

(1) Bits 0, 1, and 6
Not used (fixed to zero)

(2) Bit 2: QYE (Query Error)
When the recorder has received a query Command but the output queue is empty of data or has lost data due to overflowing, Bit 2 is set to 1.

(3) Bit 3: DDE (Device Error)
When the recorder becomes incapable of Command execution due to an internal cause, Bit 3 is set to 1.

(4) Bit 4: EXE (Execution Error)
When the recorder receives a correctly written Command but cannot execute it due to the recorder's current status, Bit 4 is set to 1. This error also occurs when program data is outside of its stipulated numeric range or is otherwise incorrectly specified.

(5) Bit 5: CME (Command Error)
When the recorder receives an illegal Command, Bit 5 is set to 1.

(6) Bit 7: PON (Power ON)
Whenever the recorder is turned on, Bit 7 is set to 1.

Conditions for clearing the standard event register

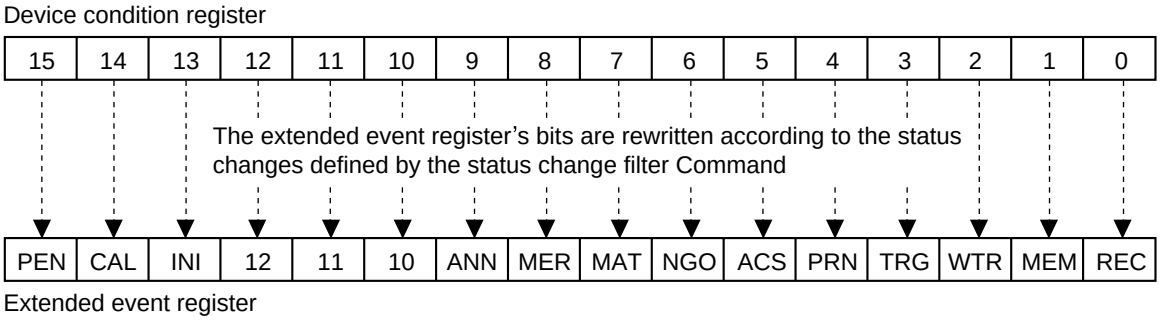
The standard event register will be cleared in any of the cases below.
(1) After the recorder responds to the *ESR query (a COMMON Group Command) which requests the standard event register's value, all bits of the standard event register are cleared and become zero.
(2) When the recorder receives the *CLS Command, which is a COMMON Group Command; or
(3) When the recorder is turned off then back on again.

Condition for clearing the standard event enable register

The recorder is turned off then back then back on again. The standard event enable register's default setting is 255.

4.4 The Extended Event Register

Depending on the nature of the query, the query placed by several of the bits of the device condition register changes over an extremely short period, making it difficult for the controlling device to obtain an accurate grasp of the recorder's status in certain cases.
To avoid this situation, as described in the following, the extended event register has a dual-register structure, consisting of (1) the device condition register which directly reflects the recorder's status and (2) the extended event register which reflects the status changes determined by the status change filter's setting.



The status change filter's setting is specified by the :STATus:FILTer Command.
The :STATus:FILTer Command's program data can be set to RISE, FALL, BOTH, or NEVer.

Sample Command: :STATus:FILTer RISE	
Program data	**Description of the status change filter**
RISE	When a bit in the device condition register changes from zero to 1, the extended event register's corresponding bit is set to 1.
FALL	When a bit in the device condition register changes from 1 to zero, the extended event register's corresponding bit is set to 1.
BOTH	When a bit in the device condition register changes from zero to 1 or vice versa, the extended event register's corresponding bit is set to 1.
NEVer	The extended event register's bits are fixed to zero.

The device condition register's status can be requested by using the :STATus:CONDition? Command. The extended event register's status can be requested by using the :STATus:EESR? Command. Furthermore, the bits of the extended event enable register can be masked by using the :STATus:EESE Command.
When the :STATus:EESE Command is issued to write a zero to the extended event enable register's bit to be masked, therefore, it becomes subject to a logical AND operation with the extended event register's corresponding bit. As a result, the masked bit of the extended event register cannot enter the enable query status expressed by "1".
The default setting of this recorder's extended event enable register is 65535, which enables all 16 bits.

Conditions for clearing the extended event register

The extended event register will be cleared in any of the cases below.

(1) After the recorder responds to the :STATus:EESR? query which requests the extended event register's value, all bits of the extended event register are cleared and become zero.
(2) When the recorder receives the *CLS Command, which is a COMMON Group Command; or
(3) When the recorder is turned off then back on again.

Condition for clearing the extended event enable register

The recorder is turned off then back then back on again. The extended event enable register's default setting is 65535.

Condition for clearing the device condition register

The recorder is turned off then back then back on again.

4.4.1 Description of the Extended Event Register's Bits

The following describes the meaning of each of the extended event register's bits (Bits 0 to 15).

PEN	CAL	INI	12	11	10	ANN	MER	MAT	NGO	ACS	PRN	TRG	WTR	MEM	REC

Bit	Symbol	Description
0	REC	Becomes 1 during a recording operation.
1	MEM	Becomes 1 during the input or output of memory data.
2	WTR	Becomes 1 while the trigger is armed.
3	TRG	Becomes 1 when the trigger has been activated.
4	PRN	Becomes 1 while the printer is in operation.
5	ACS	Becomes 1 while a disk is being accessed.
6	NGO	Becomes 1 during a waveform judgement operation.
7	MAT	Becomes 1 during execution of a mathematic or FFT operation.
8	MER	Becomes 1 when a calculation error occurs.
9	ANN	Becomes 1 during annotation printing.
10	Undefined	
11	Undefined	
12	Undefined	
13	INI	Becomes 1 while the recorder is being initialized.
14	CAL	Becomes 1 during automatic calibration of the amp settings for such functions as Auto Range, Auto Balance, and so on.
15	PEN	Becomes 1 when the recorder's chart paper supply has run out or while its printhead is raised.

NOTE

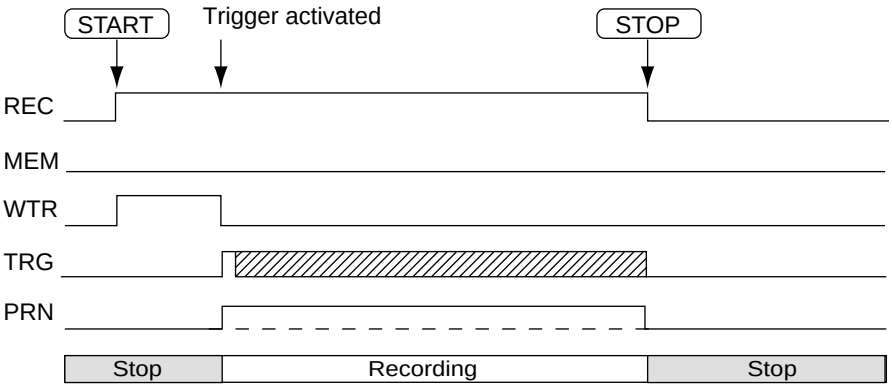
The trigger status is indicated by the status of Bit 2 (WTR) and Bit 3 (TRG).

TRG	WTR	Status
0	0	The trigger is not armed
0	1	The trigger is armed and awaiting activation
1	0	The trigger has been activated
1	1	No status

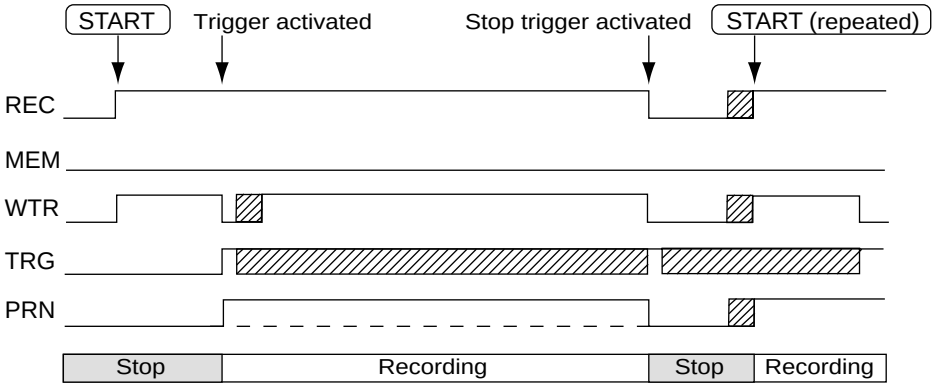
4.4.2 Sample Operations

Examples in RECORDER mode when the Trigger Function setting is Record

- The Trigger Function setting is Record, the Start Condition setting is On, and the Stop Condition setting is Off:

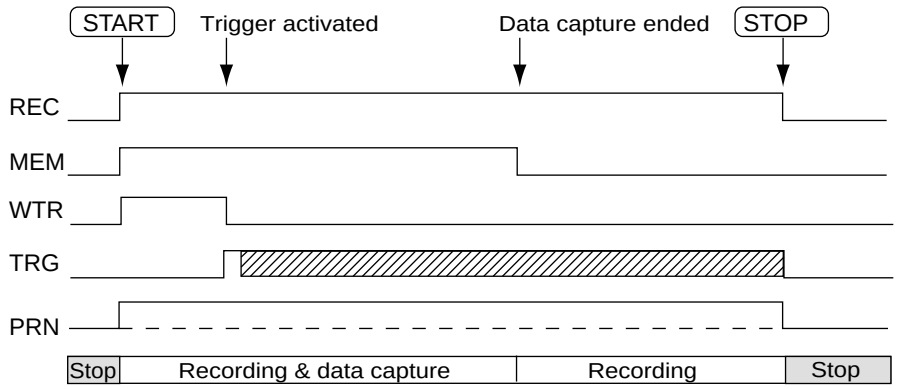


- The Trigger Function setting is Record and both the Start Condition and Stop Condition settings are On:

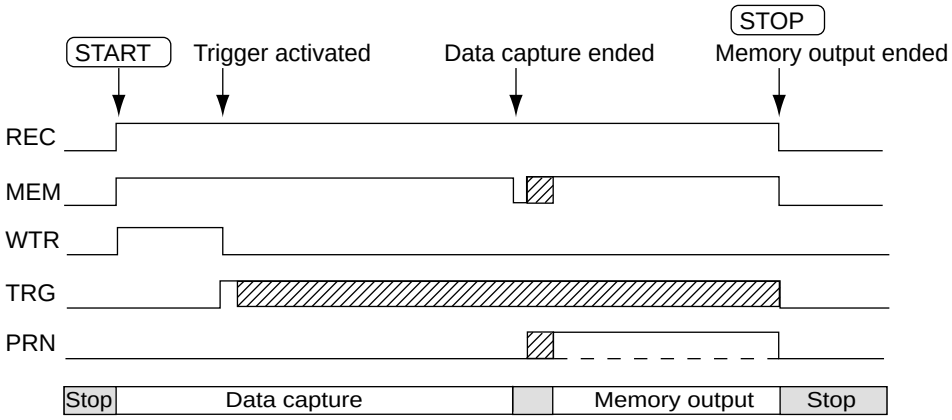


Examples in RECORDER mode when the Trigger Function setting is Memory

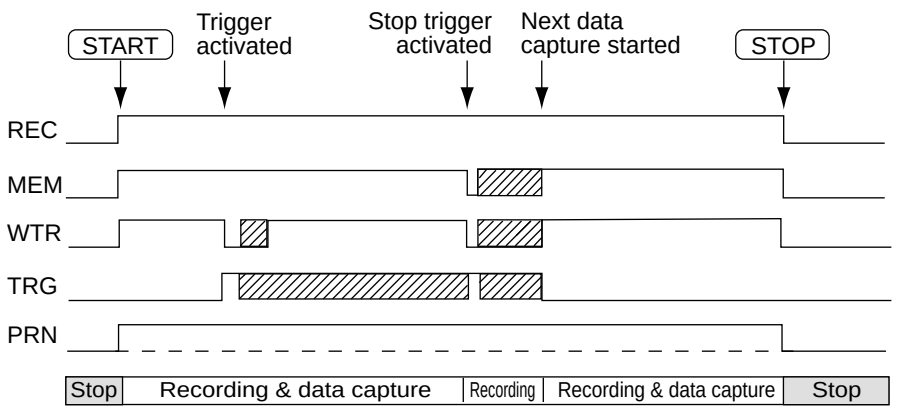
- The Trigger Function setting is Memory, the Start Condition setting is On, the Stop Condition setting is Off, and the Record Function setting is Direct:



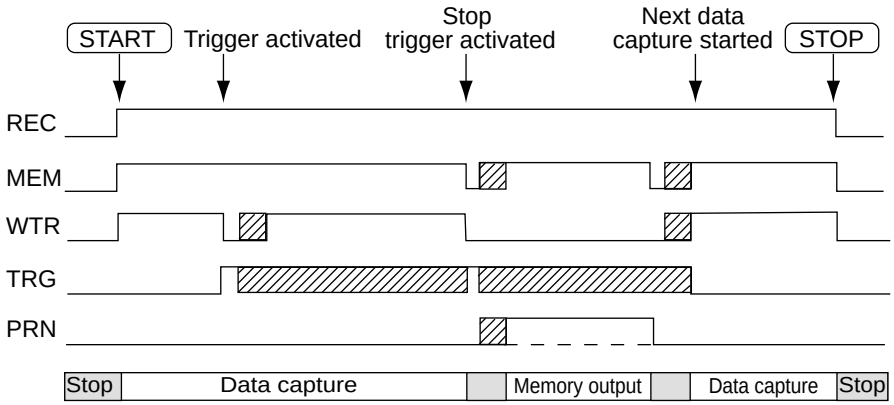
- The Trigger Function setting is Memory, the Start Condition setting is On, the Stop Condition setting is Off, and the Record Function setting is Memory:



- The Trigger Function setting is Memory, both the Start Condition and Stop Condition settings are On, and the Record Function setting is Direct:



- The Trigger Function setting is Memory, both the Start Condition and Stop Condition settings are On, and the Record Function setting is Memory:



4.5 The Error Queue

Whenever an error occurs, that error's number and message are stored in the error queue. As long as that error's data is not read out, it is left in the error queue. When the error queue overflows, the Queue Overflow error's data is written to the error queue.
An error status can be read out by issuing the **:STATus:ERRor?** query.
Error data in the error queue is output according to FIFO (First-In-First-Out) control whereby the oldest error data is output first, after which the output error data is erased from the error queue.
For a description of the error numbers and error states, refer to the "WR1000 USER'S MANUAL."

Conditions for clearing the error queue

(1) When the recorder receives the *CLS Command, which is a COMMON Group Command; or
(2) When the recorder is turned off then back then back on again.

NOTE

The presence or absence of error data in the error queue can be checked at Bit 2 (EAV) of the status byte register.

4.6 The Output Queue

The output queue stores the data to be output by the recorder in response to a query. A query may consist of a request for the recorder's current settings, a request to read out waveform data, and so on. In addition, a single query may require the return of numerous sets of response data.
Data in the output queue is output according to FIFO (First-In-First-Out) control whereby the oldest data is output first. The data output by the recorder in response to the controlling device's query is then erased from the output queue.

Conditions for clearing the output queue

(1) When the recorder receives a new program message;
(2) When the recorder receives a Device Clear Command via the GP-IB interface; or
(3) When the recorder is turned off then back then back on again.

NOTE

The presence or absence of data in the output queue can be checked at Bit 4 (MAV) of the status byte register.

CHAPTER 5

DATA OUTPUT FORMATS

This chapter provides a description of the formats used for outputting data.

5.1 Procedure for Outputting Data
5.2 Specifying the Source Setting for Output Data
5.3 Output Formats

5.1 Procedure for Outputting Data

To output data, issue the following commands in the sequence below.

(1) To specify all channels for replay: **:REPLay:CHannel ALL**

(2) To specify the number of the Memory Block for output: **:REPLay:SOURce MEMory,1<PMT>**

(3) To specify the output data format: **:REPLay:OUTPut:TYPe BINary|TEXT|CSV<PMT>**

(4) To specify the starting and end points of the data to be output: **:REPLay:OUTPut:DATA 1,1000<PMT>**

(5) To execute the output of data: **:REPLay:OUTPut:DATA?<PMT>**

5.2 Specifying the Source Setting for Output Data

This section explains how to specify the destination for saving the data to be output and how to check the contents of the target destination.

:REPLay:SOURce MEMory,<NR1>

Specifies the number of the Memory Block as the source of data output.

:REPLay:SOURce DISK,<file name>

Specifies a file name when the target data is to be output from a disk.

:REPlay:SOURce?

Requests the Memory Block No. or file name that has been specified by the Source setting as the source of data output.

Sample Responses:

REPL:SOUR:MEM, 12 : Memory Block No. 12 has been specified as the source
REPL:SOUR:DISK, “C:\GRAPHTEC.DAT”: A file has been specified as the source

NOTE

The data files which can be loaded from a disk are those with the .DAT or .GBD extension to their file names.

5.3 Output Formats

■ Data Output Using the Binary Format

To output data using the binary format, issue the following command.

:REPLay:OUTPut:TYPe BINary

Data which has been output in binary format will be organized into block data as illustrated below.

Overall Format:

Block data header
1st data point
2nd data point
3rd data point
~
nth data point

nth Data Point:

Data of CH1 (higher-order byte)
Data of CH1 (lower-order byte)
Data of CH2 (higher-order byte)
Data of CH2 (lower-order byte)
~
Data of CHxx (higher-order byte)
Data of CHxx (lower-order byte)
Logic data (higher-order byte)
Logic data (lower-order byte)
Event data (higher-order byte)
Event data (lower-order byte)

NOTE

- Each datum consists of a 16-bit integer with a plus or minus sign, and its higher-order byte is always output first.
- Any channel data, logic data, and/or event data that has not been captured in memory will not be output.
- To learn the contents of channel data, logic data, and/or event data that has been captured in memory, issue the :REPlay:DATA? command.

■ Data Output Using the Text Format

To output data in text format that uses commas as delimiters, issue the following command.

:REPLay:OUTPut:TYPe TEXT

Data that is output in text format becomes text data that is delimited by commas. The end of a record is indicated by a new line code which conforms to your New Line Code setting at the Customize Settings window (accessed by the SYSTEM key).
With this format, however, the overall text will be regarded as binary data and thus output using block data format.

■ Data Output Using the CSV Format

To output data in its original form while using commas as delimiters, issue the following command.

:REPLay:OUTPut:TYPe CSV

APPENDIX A

SPECIFICATIONS OF THE RS-232C CONNECTOR

This appendix shows the pin assignment of the connector for the RS-232C interface cable.

A.1 Compatible Interface Cable
A.2 Pin Assignment of the RS-232C Connector

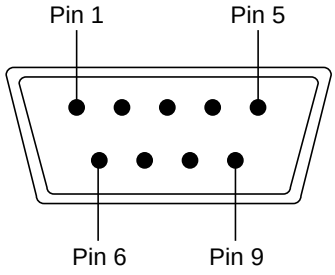
A.1 Compatible Interface Cable

To control this recorder via the RS-232C interface, connect the recorder and a personal computer using a commercially available RS-232C serial interface cable described below.

- Cross-linked cable with a D-SUB nine-pin (female) connector for connection to the recorder

A.2 Pin Assignment of the RS-232C Connector

To create a cable for use with the RS-232C interface, ensure that the pin assignment of the connector to be connected to the recorder conforms to the following diagram and table.



Pin no.	Symbol	Description
1	DCD	Carrier Detection
2	RXD	Receive Data
3	TXD	Transmit Data
4	DTR	Data Terminal Ready
5	GND	Ground level for the signal lines
6	DSR	Data Set Ready
7	RTS	Request-To-Send
8	CTS	Clear-To-Send
9	RI	Ring Indicator

APPENDIX B

ERROR CODES

This chapter provides a description of the error codes.

B.1 ERROR CODES

To confirm a status error, use the command below.

Sample inquiry : :STATUS:ERRor
Output format : :STAT:ERR<Error Code>, <Group No.>, <Error position>

The Error Codes

The error code indicates the type of error that has occurred. When a command error with an error code from 16 to 21 occurs, the recorder immediately stops interpreting the command concerned.

Error code	Description of error
1	A parameter has been incorrectly specified.
2	The setting cannot be changed at this time.
3	The specified function does not exist.
4	The specified command conflicts with another setting.
16	The command has been incorrectly structured.
17	An invalid Channel No. has been specified.
18	The program header has been incorrectly specified.
19	The specified command has no inquiry function.
20	The specified command has only an inquiry function.
21	An illegal parameter has been specified.
32	The error queue has overflowed.

The Group No.

When a single command has been specified that is associated with multiple command groups, the Group No. indicates the command group at which the error occurred. When an error occurs at a command that is associated with a single command group, the Group No. is 1.

The Error Position

The Error Position indicates the command at which an error occurred within a single command group. When an error occurs at a single command, the Error Position is 1.

Sample output

Sample inquiry : :AMP:CH1:FLT 50Hz (The “FILT” parameter has been incorrectly specified as “FLT”)
Sample response : :STAT:ERR 18,1,1