ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *Anritsu MW9060A* at our website: *Click HERE*

MW9060A

OPTICAL TIME DOMAIN REFLECTOMETER

OPERATION MANUAL (GPIB)

FIRST EDITION

MEASURING INSTRUMENTS DIVISION
ANRITSU CORPORATION

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

MW9060A

OPTICAL TIME DOMAIN
REFLECTOMETER

OPERATION MANUAL
(GPIB)

JUNE 1993 (FIRST EDITION)

Copyright © 1993 by ANRITSU CORPORATION

All rights reserved. No part of this manual may be reproduced without the prior written permission of ANRITSU CORPORATION.

The contents of this manual may be changed without prior notice.

Printed in Japan

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

WARNING

NO OPERATOR SERVICEABLE PARTS INSIDE.

REFER SERVICING TO QUALIFIED PERSONNEL.

CAUTION

FOR CONTINUED FIRE PROTECTION REPLACE

ONLY WITH SPECIFIED TYPE AND RATED FUSE.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(Blank)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ SAFETY CONSIDERATIONS:

Anritsu uses the following labels to identify safety precautions which should be followed to prevent personal injury or product damage. Please familiarize yourself with them before operating this product.

Labels used in this manual:

WARNING Indicates that the procedure could result in personal injury if not correctly performed. Do not proceed before you fully understand the explanation given with this symbol and meet the required conditions.

CAUTION Indicates that the operating procedure could result in damage to the product if not correctly performed. Do not proceed before you fully understand the explanation given with this symbol and meet the required conditions

Note: Indicates that information helpful in understanding the operation of the product is about to be presented.

Labels or symbols used on/in the product:



This international caution symbol indicates that the operator should refer to the operation manual before beginning a procedure.



This symbol indicates an earth (ground) terminal. The product should be grounded via the earth terminal if a three prong power cord is not used.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

CERTIFICATION

ANRITSU CORPORATION certifies that this instrument has been thoroughly tested and inspected, and found to meet published specifications prior to shipping.
Anritsu further certifies that its calibration measurements are based on the Japanese Electrotechnical Laboratory and Radio Research Laboratory standards.

WARRANTY

All parts of this product are warranted by Anritsu Corporation of Japan against defects in material or workmanship for a period of one year from the date of delivery.
In the event of a defect occurring during the warranty period, Anritsu Corporation will repair or replace this product within a reasonable period of time after notification, free-of-charge, provided that: it is returned to Anritsu; has not been misused; has not been damaged by an act of God; and that the user has followed the instructions in the operation manual.

Any unauthorized modification, repair, or attempt to repair, will render this warranty void.

This warranty is effective only for the original purchaser of this product and is not transferable if it is resold.

ALL OTHER EXPRESSED WARRANTIES ARE DISCLAIMED AND ALL IMPLIED WARRANTIES FOR THIS PRODUCT, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED IN DURATION TO A PERIOD OF ONE YEAR FROM THE DATE OF DELIVERY. IN NO EVENT SHALL ANRITSU CORPORATION BE LIABLE TO THE CUSTOMER FOR ANY DAMAGES, INCLUDING LOST PROFITS, OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES, ARISING OUT OF THE USE OR INABILITY TO USE THIS PRODUCT.

All requests for repair or replacement under this warranty must be made as soon as possible after the defect has been noticed and must be directed to Anritsu Corporation or its representative in your area.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Note 1:

1. The instrument is operable on a nominal voltage of 100 to 127 Vac or 200 to 250 Vac by changing the voltage-rating selection switch under the top cover.

   The voltage and current ratings are indicated on the rear panel when the instrument is shipped from the factory.

   To operate on the other voltage, change the selection switch. The plate on the rear panel indicating the voltage and current ratings should be changed to the appropriate one. Order the plate from ANRITSU CORPORATION if needed.

2. In this manual, the power supply voltage and current ratings are represented by **Vac and ***A, respectively.

3. The relationship between power supply voltage and current rating is shown below.

**Vac	***A
85 to 132 V	5A
170 to 250 V	5A

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

TABLE OF CONTENTS

SECTION 1 OVERVIEW 1-1

1.1 Outline 1-1

SECTION 2 INTERFACE FUNCTIONS AND DEVICE MESSAGE LIST 2-1

2.1 GPIB Interface Function 2-1

2.2 Device Message List 2-2

2.2.1 Suffix Code 2-2

2.2.2 Status Messages 2-3

2.2.3 Device Messages 2-8

2.2.4 Bus Commands 2-19

2.3 Parameters 2-20

2.4 Rules and Definition of Notation 2-22

2.5 Protocol 2-23

SECTION 3 PREPARING FOR USE OF GPIB 3-1

3.1 Connecting Devices with GPIB Cable 3-1

3.2 Confirming and Setting Addresses 3-2

SECTION 4 INITIALIZATION 4-1

4.1 Initializing Bus with IFC Message 4-1

4.2 Initializing Message Exchange with DCL and SDC Bus Commands 4-2

4.3 Initializing Equipment with *RST Command 4-3

4.4 Initializing Equipment with INI Command 4-4

4.5 Equipment State immediately after Power-on 4-5

4.6 Activating Equipment 4-6

SECTION 5 LISTENER INPUT FORMAT 5-1

5.1 Program Message Notation 5-2

5.1.1 Separator, Terminator and Space Preceding Header 5-2

5.1.2 General Format of Program Command Messages 5-4

5.1.3 General Format of Query Messages 5-5

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2 Functional Elements of Program Messages 5-6
5.2.1 <TERMINATED PROGRAM MESSAGE> 5-6
5.2.2 <PROGRAM MESSAGE TERMINATOR> 5-7
5.2.3 <white space> 5-8
5.2.4 <PROGRAM MESSAGE> 5-8
5.2.5 <PROGRAM MESSAGE UNIT SEPARATOR> 5-9
5.2.6 <PROGRAM MESSAGE UNIT> 5-9
5.2.7 <COMMAND MESSAGE UNIT> and <QUERY MESSAGE UNIT> 5-10
5.2.8 <COMMAND PROGRAM HEADER> 5-11
5.2.9 <QUERY PROGRAM HEADER> 5-14
5.2.10 <PROGRAM HEADER SEPARATOR> 5-15
5.2.11 <PROGRAM DATA SEPARATOR> 5-15

5.3 Program Data Format 5-16
5.3.1 <CHARACTER PROGRAM DATA> 5-17
5.3.2 <DECIMAL NUMERIC PROGRAM DATA> 5-18
5.3.3 <SUFFIX PROGRAM DATA> 5-23
5.3.4 <NON-DECIMAL NUMERIC PROGRAM DATA> 5-26
5.3.5 <STRING PROGRAM DATA> 5-27
5.3.6 <ARBITRARY BLOCK PROGRAM DATA> 5-28
5.3.7 <EXPRESSION PROGRAM DATA> 5-31

SECTION 6 TALKER OUTPUT FORMAT 6-1

6.1 Differences in Format Syntax between Program and Response Messages 6-2

6.2 Functional Elements of Response Messages 6-3
6.2.1 <TERMINATED RESPONSE MESSAGE> 6-3
6.2.2 <RESPONSE MESSAGE TERMINATOR> 6-3
6.2.3 <RESPONSE MESSAGE> 6-4
6.2.4 <RESPONSE MESSAGE UNIT SEPARATOR> 6-4
6.2.5 <RESPONSE MESSAGE UNIT> 6-5
6.2.6 <RESPONSE HEADER SEPARATOR> 6-5
6.2.7 <RESPONSE DATA SEPARATOR> 6-5
6.2.8 <RESPONSE HEADER> 6-6
6.2.9 <RESPONSE DATA> 6-9

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 7 COMMON COMMANDS 7-1

[1] *CLS (Clear Status) 7-2

[2] *ESE/*ESE? (Event Status Enable) 7-3

[3] *ESR? (Event Status Register) 7-5

[4] *IDN? (Identification Number) 7-6

[5] *OPC/*OPC? (Operation Complete) 7-7

[6] *RST (Reset) 7-9

[7] *SRE/*SRE? (Service Request Enable) 7-12

[8] *STB? (Read Status Byte) 7-14

[9] *TST? (Self-Test) 7-15

[10] *TRG? (Trigger) 7-16

[11] *WAI (Wait to Complete) 7-17

SECTION 8 STATUS BYTE AND SYNCHRONIZATION TECHNIQUE 8-1

8.1 Standard Status Model of IEEE488.2 8-1

8.2 Status Byte (STB) Register 8-3

8.2.1 ESB and MAV Summary Messages 8-3

8.2.2 Device Specific Summary Messages 8-4

8.2.3 Reading and Clearing STB Register 8-5

8.3 Enabling SRQ 8-7

8.4 Standard Event Status Register 8-9

8.4.1 Bit Definition of Standard Event Status Register 8-9

8.4.2 Details of Query Errors 8-10

8.4.3 Reading, Writing, and Clearing Standard Event Status Register 8-11

8.4.4 Reading, Writing, and Clearing Standard Event Status Enable Register 8-11

8.5 Extended Event Status Register 8-12

8.5.1 Bit Definition of Error Event Status Register 8-12

8.5.2 Bit Definition of Termination Event Status Register 8-14

8.5.3 Reading, Writing, and Clearing Extended Event Status Register 8-16

8.5.4 Reading, Writing, and Clearing Extended Event Status Enable Register 8-16

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.6 Queue Model 8-17

8.7 Technique to Synchronize Device and Controller 8-19
8.7.1 Forced Sequential Execution 8-19
8.7.2 Wait for Device's Output Queue Response 8-20
8.7.3 Waiting for Service Request 8-21

SECTION 9 DETAILS OF DEVICE MESSAGES 9-1

[1] APR/APR? (Approximate Method) 9-2
[2] ATA/ATA? (Auto Attenuator) 9-3
[3] ATT/ATT? (Attenuator) 9-4
[4] AUT? (Auto Measurement Data) 9-5
[5] AUTD/AUTD? (Auto Data Select) 9-7
[6] AVG/AVG? (Average) 9-8
[7] AVL/AVL? (Average Limit) 9-9
[8] BSL/BSL? (Return-Loss Backscattered Level) 9-11
[9] CIT/CIT? (Copy Item) 9-12
[10] CON? (Connector) 9-13
[11] CMP/CMP? (Compare Recall ON/OFF) 9-14
[12] CPY/CPY? (Copy) 9-15
[13] DAT? (Data) 9-17
[14] DATE/DATE? (Date) 9-20
[15] DLEVT/DLEVT? (Delete Events) 9-21
[16] DPEVT/DPEVT? (Display Events) 9-22
[17] DSR/DSR? (Distance Range) 9-23
[18] DZL? (Spline & Dead Zone Loss Measurement Data) 9-24
[19] ESE2/ESE2? (Termination Event Status Enable) 9-25
[20] ESE3/ESE3? (Error Event Status Enable) 9-27
[21] ESR2? (Termination Event Status Register) 9-29
[22] ESR3?(Error Event Status Register) 9-31
[23] ETBL/ETBL? (Dislplay Events Table) 9-32
[24] EVT/EVT? (Events) 9-33
[25] EVTD? (Events Data) 9-34
[26] EVTS/EVTS? (Events Select) 9-36

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[27] FDL (File Delete) 9-37
[28] FEED (Feed) 9-38
[29] FMT (Format) 9-39
[30] FNC/FNC? (Function) 9-40
[31] FTR/FTR? (Full Trace) 9-41
[32] HSC/HSC? (Horizontal Scale) 9-42
[33] HSF/HSF? (Horizontal Shift) 9-43
[34] IFS/IFS? (Interface Setting) 9-44
[35] IEVT/IEVT? (Insert Events) 9-46
[36] INI (Initialize) 9-47
[37] IOR/IOR? (IOR) 9-48
[38] ITEM/ITEM? (ITEM) 9-49
[39] LD/LD? (Laser) 9-50
[40] LOS? (Loss Measurement Data) 9-51
[41] MED/MED? (Media) 9-52
[42] MKP/MKP? (Marker Position) 9-53
[43] MKS/MKS? (Marker Select) 9-56
[44] MSP/MSP? (Mask Position) 9-58
[45] NMSK/NMSK? (Near End Mask) 9-60
[46] PLG? (Plug-in Unit) 9-61
[47] PLS/PLS? (Pulse Width) 9-62
[48] PRO/PRO? (Procedure) 9-63
[49] PWR/PWR? (LD Power) 9-64
[50] RCL/RCL? (Recall) 9-65
[51] RDA/RDA? (Result Display Area) 9-66
[52] RDY? (Ready) 9-67
[53] RMEM/RMEM? (Regist Memory Number) 9-68
[54] RNG/RNG? (Save Range) 9-69
[55] RTL? (Return-Loss Measurement Data) 9-71
[56] RTP/RTP? (Return-Loss Parameter) 9-72
[57] SAV (Save) 9-74
[58] SET (Setting) 9-75

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[59] SMP? (Sampling Start/End/Resolution) 9-76
[60] SMT/SMT? (Smoothing) 9-77
[61] SMTL/SMTL? (Smoothing Level) 9-78
[62] SPL? (Splice Loss Measurement Data) 9-79
[63] SWP/SWP? (Sweep Mode) 9-80
[64] TGT/TGT? (Target) 9-81
[65] THR/THR? (Threshold) 9-82
[66] TIM/TIM? (Time Out) 9-83
[67] TIME/TIME? (Time) 9-84
[68] TIT/TIT? (Title) 9-85
[69] TRM/TRM? (Terminator) 9-87
[70] UNL/UNL? (Unit Of Length) 9-88
[71] VSC/VSC? (Vertical Scale) 9-89
[72] VSF/VSF? (Vertical Shift) 9-90
[73] WLS/WLS? (Wavelength Select : λ Select) 9-91
[74] ZADJ/ZADJ? (Horizontal Zero Adjust) 9-93

SECTION 10 PROGRAM EXAMPLES 10-1

10. 1 Precautions on Creating GPIB Control Programs 10-1

10.2 Basic Programming 10-2
10.2.1 Measuring Loss between Two Arbitrary Points 10-2
10.2.2 Measuring Loss during Sweep 10-3
10.2.3 Measuring Splice Loss 10-5
10.2.4 Reading Waveform Data (ASCII Format) 10-6
10.2.5 Reading Waveform Data (Binary Format) 10-7
10.2.6 Wait for Average Termination 10-8

APPENDIX A COMPARISON OF MESSAGES BETWEEN MW9060A AND MW910C A-1

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(Blank)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 1
OVERVIEW

1.1 Outline

The MW9060A Optical Time Domain Reflectometer enables measurement automation by building a system bus in combination with an external controller using the GPIB interface bus (IEEE std 488.2-1987).

The GPIB is a standard digital interface bus for programmable measurement equipment established in 1975 by IEEE (Institute of Electric and Electronics Engineers). Its original version was announced in 1975 as IEEE std 488 (hereafter called IEEE488). The IEEE std 488-1978 presented several problems because it only stipulated the hardware specifications of the bus. To solve the problems, IEEE std 728-1982 (hereafter called IEEE728) was additionally formulated in 1982 to stipulate the software specifications of the connected devices. The IEEE std 728-1982, however, did not fully support a software-sharing concept based on the part of users. To overcome this shortcoming, a modified version of IEEE-728 was introduced in 1987 under the name of IEEE std 488.2-1987 (hereafter called IEEE488.2). This latest version is enhanced in the standardization of the message exchange protocol, message data code, device input/output format, and common commands. As a result of this development, the first version IEEE488 is renamed the IEEE std 488.1-1987 (hereafter called IEEE488.1).

Table 1-1 summarizes the historical transition of the GPIB standards described above.

Table 1-1 Historical Transition of GPIB Standards

New Standard	Old standard	Subject of standardization	Remarks
IEEE488.1	IEEE488	Hardware	Contents are the same.
IEEE488.2	IEEE728	Software	—

The equipment that conforms to IEEE488.2 must also conform to IEEE488.1. On the contrary, not all equipment based on IEEE488.1 is guaranteed of conformance to IEEE488.2.

The MW9060A Optical Time Domain Reflectometer has the following GPIB functions:

1. Control of each function
2. Read of measurement conditions
3. Interrupt function and serial polling operation

Sample programs in this manual are produced assuming the use of the Packet V Personal Technical Computer (Anritsu) as the controller and assuming that the MW9060A GPIB address is 1.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(Blank)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 2
INTERFACE FUNCTIONS AND DEVICE MESSAGE LIST

2.1 GPIB Interface Function

Among the GPIB interface functions stipulated in IEEE488.1, IEEE488.2 prescribes the absolute minimum subset required for each measurement instrument to enable system configuration with any particular measurement instrument without problems.

The MW9060A Optical Time Domain Reflectometer has the interface functions shown in Table 2-1.

Table 2-1 GPIB Interface Functions

No.	Symbol	Function
1	S H 1	All source handshake functions provided.
2	A H 1	All accept handshake functions provided.
3	T 6	Basic talker function provided. Talk-only function not provided. MLA-based talker release function provided.
4	L 4	Basic listener function provided. Listener-only function not provided. MTA-based listener release function provided.
5	S R 1	All service request and status byte functions provided.
6	R L 1	All remote/local functions provided.
7	P P 0	Parallel poll function not provided.
8	D C 1	All device clear functions provided.
9	D T 1	All device trigger functions provided.
10	C 0	Controller function not provided.
11	E 2	Tri-state

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.2 Device Message List

Device messages are sent and received between the controller and device (measurement instrument) via GPIB. These messages include program messages and response messages.

Consisting of ASCII code, program messages are transferred from the controller to the device. These program messages come in two types: command messages (control messages) and query messages (inquiry messages). Command messages are used to control the device and consist of the common commands stipulated in IEEE488.2 and dedicated command messages for the MW9060A Optical Time Domain Reflectometer. Query messages are used to read device status, and also consist of IEEE-based common commands and dedicated query messages for the MW9060A Optical Time Domain Reflectometer.

Response messages are transferred from the device to the controller. More specifically, these messages are the ones that are returned from the device to the controller when query messages are received from the controller.

2.2.1 Suffix Code

The program or response messages sent and received between the controller and device can have a suffix (unit) added to them.

Note, however, that the MW9060A Optical Time Domain Reflectometer does not use this suffix. With the MW9060A, all numeric values are expressed in fixed units. Table 2-2 lists the numeric units used in the MW9060A.

Table 2-2 Numeric Units Handled by MW9060A

Item	Unit
Length (distance)	1 m; 1 cm for binary notation (IOR is handled as 1.5 on all occurrences.)
Pulse width	1μs
Wavelength	1μm
Level (waveform data, attenuator)	0.001dB for binary notation 1 dB for others

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.2.2 Status Messages

Device status is indicated by using a status byte register. The controller determines device status and establishes synchronization by reading values from this status byte register. Each bit of the status byte register functions as a summary bit for the register structure present behind it (i.e., each bit indicates a specific event inherent in it among various events.).

Figure 2-1 shows the configuration of status byte.

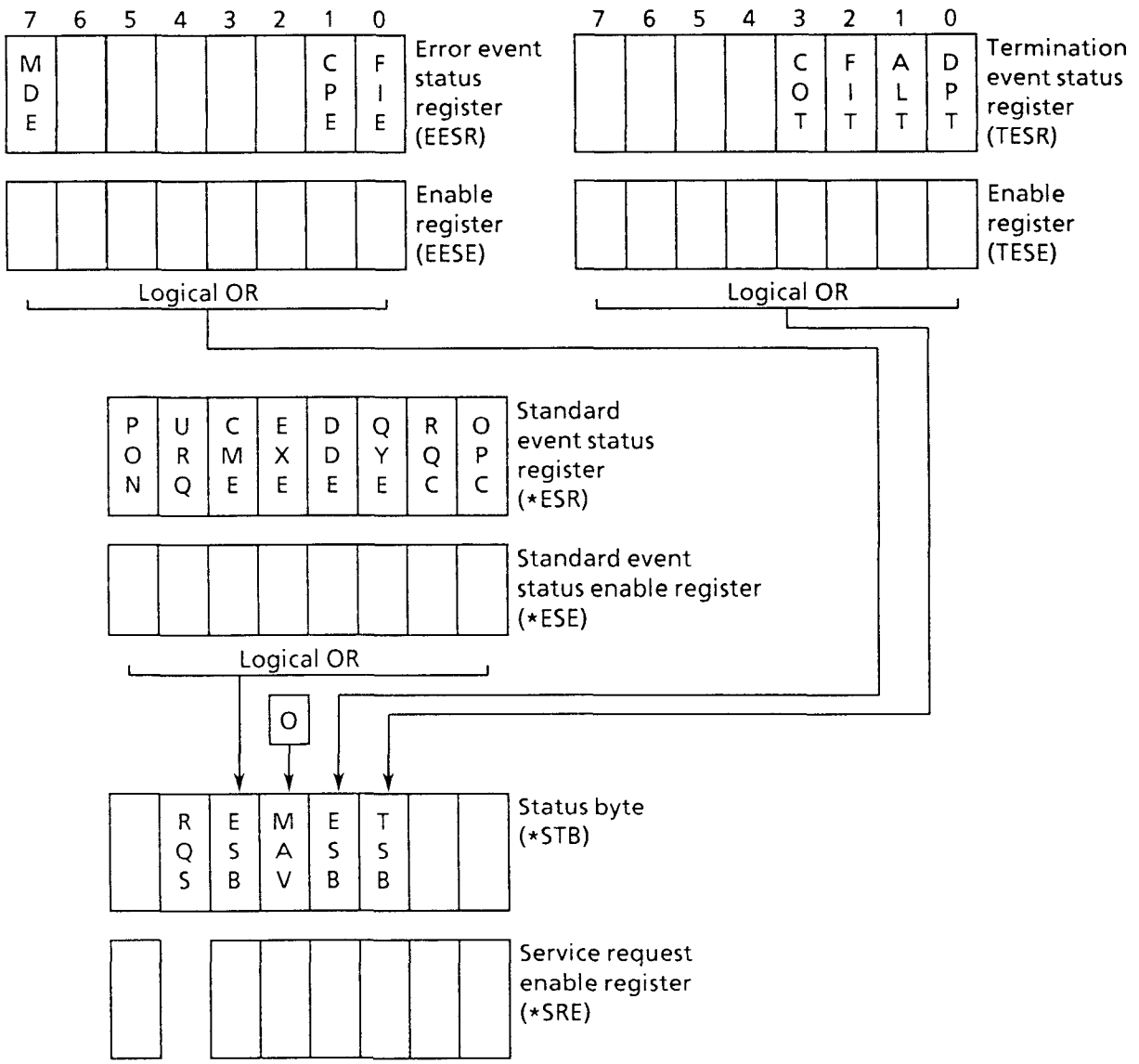


Fig. 2-1 Configuration of Status Byte

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-3 Bit Definition of Status Byte

Bit	Event name	Description
7	——	Always 0 for the MW9060A.
6	R Q S (Request Service)	Indicates whether the device is requesting service. This bit is returned during serial poll. RQS is reset to 0 after being read by the controller through serial poll.
5	E S B (Event Status Bit)	Indicates whether any condition set in the standard event status register is enabled.
4	M A V (Message Available)	Indicates whether there is response in the output queue.
3	E R S B (Error Summary Bit)	Indicates that an error event inherent in the MW9060A has occurred.
2	T S B (Termination Summary Bit)	Indicates an event that has been terminated.
1	——	Always 0 for the MW9060A.
0	——	Always 0 for the MW9060A.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-4 Bit Definition of Standard Event Status Register

Bit	Event name	Description
7	P O N (Power on)	Indicates that the power is on.
6	U R Q (User Request)	Requests local control (rtl). This bit is generated regardless of the device's remote/local mode. This bit is always 0 for the MW9060A.
5	C M E (Command Error)	Indicates a syntax-error program message, mis-spelled command, or reception of Group Execution Trigger command (GET) during program message execution.
4	E X E (Execution Error)	Indicates that a program message is received which, although syntax is correct, cannot be executed because parameters are out of range.
3	D D E (Device Dependent Error)	Indicates that an error inherent in the device has occurred.
2	Q Y E (Query Error)	Indicates non-conformance such as reading data while no data is present in the output queue and losing data in the output queue for some reason (e.g., overflow) to the query protocol.
1	R Q C (Request Control)	Indicates that the device is requesting control right of the bus to become the active controller itself. This bit is always 0 for the MW9060A.
0	O P C (Operation Complete)	Indicates whether the device has completed all pending operations. This bit only responds to the *OPC command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-5 Bit Definition of Error Event Status Register

Bit	Event name	Description
7	M D E (Measurement Data Error)	0 : No error detected. 1 : Set when DAT?, LOS?, SPL?, or AUT?, DZL? or EVTD? is received while no waveform data is present, for example, immediately after power-on or immediately after average reset.
6		0 : Not used
5		0 : Not used
4		0 : Not used
3		0 : Not used
2	PUE (Plug-in Unit error)	0 : No error 1 : Plug-in Unit READ error that occurs when the return-loss parameter of the plug-in unit is abnormal at power on.
1	C P E (Copy Error)	0 : No error detected. 1 : Indicates all errors that occur when COPY is executed while paper is used up or the target device is not connected.
0	F I E (File Error)	0 : No error detected. 1 : Indicates all errors that occur when FILE is executed while the format is incorrect, no media is set into place, or the PMC battery has run down.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-6 Bit Definition of Termination Event Status Register

Bit	Event name	Description
7	——	0 : Not used
6	——	0 : Not used
5	SMTT (Smoothing Termination)	0 : No reports 1 : Indicates that SMOOTHING operation is terminated. When an error occurs, the error event status register is set to prevent this bit from being affected.
4	TRG (*TRG Command Termination)	0 : No reports 1 : Indicates that *TRG operation is terminated. When an error occurs, the error event status register is set to prevent this bit from being affected.
3	C O T (Copy Termination)	0 : No reports 1 : Indicates that COPY operation is terminated. When an error occurs, the error event status register is set to prevent this bit from being affected.
2	F I T (File Termination)	0 : No reports 1 : Indicates that FILE operation is terminated. When an error occurs, the error event status register is set to prevent this bit from being affected.
1	A L T (Average Limit Termination)	0 : No reports 1 : Set when averaging the average limit set values is terminated.
0	D P T (Data Prepare Termination)	0 : No reports 1 : Indicates that preparation for measurement results is completed. This bit is set when waveform is prepared by executing laser-on with no waveform or by recalling a waveform.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.2.3 Device Messages

Tables 2-7 and 2-8 list the common commands available with and the commands inherent in the MW9060A Optical Time Domain Reflectometer, as well as query and response messages.

Table 2-7 Common Commands

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Clear Status	✻CLS	—	—	
2	Event Status Enable	✻ESE␣ <Register value>	✻ESE?	<Register value>	
3	Event Status Register	—	✻ESR?	<Register value>	
4	Identification Number	—	✻IDN?	ANRITSU, MW9060 A,0,<Equipment version>	
5	Operation Complete	✻OPC	✻OPC?	1	
6	Reset	✻RST	—	—	
7	Service Request Enable	✻SRE␣ <Register value>	✻SRE?	<Register value>	
8	Read Status Byte Query	—	✻STB?	<Register value>	
9	Trigger	✻TRG	—	—	
10	Self-Test Query	—	✻TST?	0	
11	Wait to Complete	✻WAI	—	—	
12	Pass Control Back	✻PCB␣ <Primary address> ␣<Secondary address>	—	—	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-8 Inherent Commands

Control commands (measurement control)

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Laser	LD␣{0|1}	L D ?	LD␣{0|1}	1 : ON 0 : OFF
2	Average	AVG␣{0|1}	A V G ?	AVG␣{0|1}	1 : ON 0 : OFF
3	Marker Select	MKS␣<Marker No.>	M K S ?	MKS␣<Marker No.>	<Marker No.> (i) For LOSS 0 : * marker 1 : x_1 marker (ii) For SPLICE 0 : * marker 1 : x_1 marker 2 : x_2 marker 3 : x_3 marker 4 : x_4 marker (iii) For AUTO 1 ~ 3 max. (iv) For R.LOSS 0 : * marker 1 : x_1 marker (v) For SPL & DZL 0 : * marker 1 : x_1 marker 2 : x_2 marker 3 : x_3 marker 4 : x_4 marker 6 : x_6 marker 7 : x_7 marker (vi) For EVENTS 0 : * marker 1 : x_1 marker 2 : x_2 marker 3 : x_3 marker 4 : x_4 marker 5 : x_5 marker
4	Marker Position	MKP␣<Marker No.><Distance>	M K P ? ␣<Marker No.> M K P ?	MKP␣<Distance> MKP␣<Number of markers><Distance>, ···, ···, <Distance>	<Marker No.> is the same as Marker Select.
5	Vertical Shift	VSF␣<Vertical axis shift value>	V S F ?	VSF␣<Vertical axis shift value>	
6	Horizontal Shift	HSF␣<Horizontal	H S F ?	HSF␣<Horizontal axis shift value>	
7	Vertical Scale	VSC␣<Vertical axis scale value>	V S C ?	VSC␣<Vertical axis scale value>	
8	Horizontal Scale	HSC␣<Horizontal axis scale value>	H S C ?	HSC␣<Horizontal axis scale value>	
9	Function	FNC␣<Measurement item>	F N C ?	FNC␣<Measurement item>	<Measurement item> 0 : LOSS 1 : SPLICE 2 : AUTO 3 : RETURN LOSS 4 : SPL & DZL

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-8 Inherent Commands (Continued)

Control commands (measurement control)

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
10	Sampling Start/End/Res.	——	SMP?	SMP␣<SS>,<SE>, <SR>	<SS> Sampling start <SE> Sampling end <SR> Sampling resolution
11	Ready	——	RDY?	RDY␣{0|1}	1: Ready 0: Not ready
12	Connector	——	CON?	CON␣{0|1}	1: Connected 0: Unconnected
13	Wavelength Select (λ Select)	WLS␣ <Wavelength>	WLS? [␣0] WLS? ␣1	WLS␣<Wavelength> WLS␣<No. of wavelengths>, <Wavelength>, ---------- ,<Wavelength>	

Control commands (CONDITION)

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Distance Range	DSR␣<Distance range>	DSR?	DSR␣<Distance range>	
2	Pulse Width	PLS␣<Pulse width>	PLS?	PLS␣<Pulse width>	
3	Auto. Attenuator	ATA␣{0|1}	ATA?	ATA␣{0|1}	1: ON (AUTO) 0: OFF(MANUAL)
4	Attenuator	ATT␣<Attenuator value>	ATT?	ATT␣<Attenuator value>	
5	Mask Position	MSP␣<Mask No.> <Distance>	MSP? ␣<Mask No.> MSP?	MSP␣<Distance> MSP␣<Number of masks>,<Distance>, ···,···,<Distance>	
6	NMSK	NMSK␣<Deviation>	NMSK?	NMSK␣<Deviation>	
7	LD Output Power	PWR␣<Output power>	PWR?	PWR␣<Output power>	Output power:0 to 127

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-8 Inherent Commands (Continued)

Control commands (MEASURE)

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Approximate Method (Linear approximation)	APR␣<Linear approximation>	A P R ?	APR␣<Linear approximation>	<Linear approximation> 0 : 2PA 1 : LSA (ALL) 2 : LSA (DISP)
2	Threshold	THR␣<Threshold value>	T H R ?	THR␣<Threshold value>	
3	Average Limit	AVL␣<NT>,<NS> AVL␣<NT>,<TS>	A V L ?	AVL␣<NT>,<NS> <TS>,<NTC>	<NT> Count or time 0 : Count 1 : Time <NS> Set value of count <TS> Set value of time <NTC> Counter value of count or time
4	IOR	IOR␣<IOR value>	I O R ?	IOR␣<IOR value>	
5	Return Loss Parameter	RTP␣<Set value 1>, ·····,<Set value 4>	R T P ?	RTP␣<Set value 1>, ····, <Set value 4>	Set value 1: RSL Set value 2: N1 Set value 3: N2 Set value 4: Ne
6	Return Loss Parameter Back Scattered Level	BSL␣<Set value>	B S L ?	BSL␣<BSL value>	
7	Auto Data Select	AUTD␣{0|1}	A U T D ?	AUTD␣{0|1}	0 : DISP 1 : ALL

Control commands (DISPLAY)

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Title	TIT␣<Title> TIT	T I T ?	TIT␣<Title>	
2	Unit of Length	UNL␣<Unit>	U N L ?	UNL␣<Unit>	<Unit> 0 : Meter 1 : Foot 2 : Mile
3	Smoothing	SMT␣<0|1>	S M T ?	SMT␣{0|1}	0 : OFF 1 : ON
4	Smoothing Level	SMTL␣<Smoothing level>	S M T L ?	SMTL␣<Smoothing level>	<Smoothing level> 1 to 6
5	Full Trace	FTR␣{0|1}	F T R ?	FTR␣{0|1}	0 : OFF 1 : ON
6	Horizontal Zero Adjust	ZADJ␣<Distance>	Z A D J ?	ZADJ␣<Distance>	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-8 Inherent Commands (Continued)

Control commands (FILE)

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Media	MED␣<Media>	MED?	MED␣<Media>	<Media> 0: INT MEMORY 1: INT PMC 2: EXT PMC1 3: EXT PMC2 4: EXT FDD 5: INT FDD
2	Save	SAV␣<Memory No.> SAV␣<File name>	—	—	
3	Save Range	RNG␣{1|2}	RNG?	RNG␣{1|2}, <SVS>, <SVE>, <SVR>	1: All measured 2: Disp range <SVS> Save start <SVE> Save end <SVR> Save resolution
4	Recall	RCL␣<Memory No.> RCL␣<File name>	RCL?	RCL␣<Memory No.> RCL␣<File name>	
5	File Delete	FDL␣<Memory No.> FDL␣<File name>	—	—	
6	Format	FMT	—	—	
7	Item	ITEM␣<File type>	ITEM?	ITEM␣<File type>	<File type> 0: Waveform 1: Waveform and Event table 2: Event table 3: Procedure

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-8 Inherent Commands (Continued)

Control commands (COPY)

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Copy	CPY␣{0|1},<screen No.>	C P Y ?	CPY␣{0|1},<screen No.>	1 : ON 0 : OFF <Screen No.> 0 : Measurement screen 1 : Event table 2 : Return loss parameter 3 : Recall file table 4 : Compare setting 5 : Interface 6 : Procedure
2	Target	TGT␣<Target>	T G T ?	TGT␣<Target>	<Target> 0 : Group 1 1 : Group 2 50 : Group 3 51 : Group 4 52 : Data Storage Unit MC8104A
3	Copy item	CIT␣<Print item>	C I T ?	CIT␣<Print item>	<Print item> 0 : All items 1 : Wave form only
4	Feed	FEED␣<No. of line>			

Control commands (COMPARE)

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Compare Recall ON/OFF	CMP␣{0|1}	C M P ?	CMP␣{0|1}	1 : ON 0 : OFF
2	Setting ON/OFF	SET␣{0|1}			1 : ON 0 : OFF

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-8 Inherent Commands (Continued)

Control commands (SYSTEM) ␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Date	DATE␣<Year>,<Month>,<Day>	D A T E ?	DATE␣<Year>,<Month>,<Day>	
2	Time	TIME␣<Hour>,<Minute>	T I M E ?	TIME␣<Hour>,<Minute>	
3	Interface Setting	IFS␣<Slot>,<Set item>,<Set value>	I F S ? ␣<Slot>,<Set item>	IFS␣<Slot>,<Set item>,<Set value>	
4	Result Display Area	RDA␣{0 | 1}	R D A ?	RDA␣{0 | 1}	0 : Scale 1 : Table

Control commands (SWEEP MODE) ␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Sweep Mode	SWP␣{0 | 1}	S W P ?	SWP␣{0 | 1}	0 : Normal mode 1 : Fast mode

Control commands (Other OTDR control) ␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Initialize	INI	—	—	
2	Plug-in unit	—	P L G ?	PLG␣<Plug-in unit model name>	
3	Terminator	TRM␣{0 | 1}	T R M ?	TRM␣{0 | 1}	0 : LF 1 : CR,LF
4	Time out	TIM␣<Timeout time>	T I M ?	TIM␣<Timeout time>	
5	Error Event Status Enable Register	ESE3␣<Register value>	E S E 3 ?	ESE3␣<Register value>	
6	Error Event Status Register	—	E S R 3 ?	ESR3␣<Register value>	
7	Termination Event Status Enable Register	ESE2␣<Register value>	E S E 2 ?	ESE2␣<Register value>	
8	Termination Event Status Register	—	E S R 2 ?	ESR2␣<Register value>	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-8 Inherent Commands (Continued)

Control commands (EVENTS) ␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Events	EVT␣{0|1}	E V T ?	EVT␣{0|1}	0 : OFF 1 : ON
2	Display Events Table	ETBL␣{0|1}	E T B L ?	ETBL␣{0|1}	0 : OFF 1 : ON
3	Insert Events	IEVT␣<Event position>,<X1 marker position>,<X2 marker position>,<X3 marker position>,<X4 marker position>,<X5 marker position>	I E V T ?	IEVT␣<Event position>,<X1 marker position>,<X2 marker position>,<X3 marker position>,<X4 marker position>,<X5 marker position>	
4	Delete Events	DLEVT␣<Event No.>	—	—	
5	Display Events	DPEVT␣<Event No.>	D P E V T ?	DPEVT␣<Event No.>	
6	Events Select	EVTS␣<Event No.>	E V T S ?	EVTS␣<Event No.>	

Control commands (PROCEDURE) ␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Procedure	PRO␣{0|1}	P R O ?	PRO␣{0|1}	0 : OFF 1 : ON

Control commands (PROCEDURE REGIST) ␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Regist Memory Number	RMEM␣<Memory No.>	R M E M ?	RMEM␣<Memory No.>	<Memory No.> 1~6
2	Wait	WAIT␣{0|1},<Set value>	W A I T ?	WAIT␣{0|1},<Set value>,<Elapsed value>	0 : Number of times 1 : Time
3	Pause	PAUSE␣{0|1}	P A U S E ?	PAUSE␣{0|1}	0 : ON 1 : OFF

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-8 Inherent Commands (Continued)

Measurement result request commands

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
1	Loss Measurement Data	——	L O S ?	LOS␣ <Loss between 2 points>,<Distance between 2 points>, <Transmission loss>	
2	Splice Loss Measurement Data	——	S P L ?	SPL␣ <Splice loss>	
3	Auto Measurement Data	——	A U T ? AUT?␣ <Faulty point No.>	AUT?␣ <Number of faulty points> AUT␣ <N> <SPLICE>,<LOSS_ L>,<LENG_L>, <TRLOSS_L>, <LOSS_R>,<LENG_ R>,<TRLOSS_R>	<N> Faulty point No. <SPLICE> Splice loss at faulty point <LOSS_L> Fiber loss (Left side of faulty point) <LENG_L> Fiber length (Left side of faulty point) <TRLOSS_L> Transmission loss (Left side of faulty point) <LOSS_R> Fiber loss (Right side of faulty point) <LENG_R> Fiber length (Right side of faulty point) <TRLOSS_R> Transmission loss (Right side of faulty point)
4	Return Loss Measurement Data	——	R T L ?	RTL␣ <Return loss value>	
5	Data	——	DAT?␣ <START>, <RES>, <NUMBER> [,<TYPE>]	<START>, <RES>, <NUMBERn>, 0,<1st data>,……,<n'th data>	<START> Data start distance <RES> Data resolution <NUMBER> Number of data entries <TYPE> 0: ASCII 1: Binary

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-8 Inherent Commands (Continued)

Measurement result request commands

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
6	Splice & Dead Zone Loss Measurement Data	——	D Z L ?	DZL␣ <SPLICE>, <LOSS_L>,<LENG_L>,<TRLOSS_L>,<LOSS_R>,<LENG_R>, <TRLOSS_R>	<SPLICE> Splice loss at faulty point <LOSS_L> Fiber loss (Left side of faulty point) <LENG_L> Fiber length (Left side of faulty point) <TRLOSS_L> Transmission loss (Left side of faulty point) <LOSS_R> Fiber loss (Right side of faulty point) <LENG_R> Fiber length (Right side of faulty point) <TRLOSS_R> Transmission loss (Right side of faulty point)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table 2-8 Inherent Commands (Continued)

Measurement result request commands

␣ denotes a space.

No.	Control item	Command	Query	Response message	Remarks
7	Events Data	——	E V T D ?	EVTD␣ <Number of events>	
			EVTD?␣ <Event No.>	EVTD␣ <N>, <SPLICE>,<TYPE>, <RLOSS>,<LOSS-L>,<LENG_L>, <TRLOSS_L>,<LOSS_R>,<LENG_R>,<TRLOSS_R>	<N> Event No. <SPLICE> Event splice loss <TYPE> 0 : Nonreflective 1 : Reflective 2 : Saturated <RLOSS> Return loss <LOSS_L> Fiber loss (Left side of Event) <LENG_L> Fiber length (Left side of Event) <TRLOSS_L> Transmission loss (Left side of Event) <LOSS_R> Fiber loss (Right side of Event) <LENG_R> Fiber length (Right side of Event) <TRLOSS_R> Transmission loss (Right side of Event)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.2.4 Bus Commands

The MW9060A has the following bus command functions:

1. Interface clear (IFC)

The interface clear command (IFC) stops all bus functions. It clears the address specification of all listeners and talkers and disables serial poll on all devices before returning control to the system controller.

2. Device clear (DCL, SDC)

Device clear (DCL) and selective device clear (SDC) initialize message exchange between devices on the GPIB bus.

Message-exchange initialization is done to prepare a device for new commands to be sent from the controller when the message exchange-related internal parts of the device are not in an appropriate state for control by the controller (e.g., as a result of execution of some other program) although equipment settings need not be changed. The MW9060A clears the input and output buffers upon reception of device clear (DCL) or selective device clear (SDC) and all pending commands are thereby cleared.

3. Group Execution Trigger (GET)

The MW9060A turns laser and average on when Group Execution Trigger (GET) is received.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.3 Parameters

The MW9060A checks the validity of input parameters in two ways. It first checks the allowable range of parameters; then, it checks whether the input parameter is appropriate as the parameter of each header message.

When an error is found by one of these checks, the following action is taken:

1. When the allowable range of parameters is exceeded

The command error (bit 5) of the standard event status register is set.

2. When inappropriate as the parameter of each header message

The execution error (bit 4) of the standard event status register is set.

The following shows decision criterion for each.

1. Input range

The allowable range of parameters received by the MW9060A are defined as shown below. When a parameter value exceeds the range of these values, an error is assumed.

Integer	: −2147483648	to	2147483647
Real number	: −2147483648	to	2147483647
Exponent	: −9.9E+37	to	9.9E+37

2. Determination of qualification for parameter to each header message

Before determining whether the parameter is appropriate for a header message, its value is rounded off first. The following shows how values are rounded off and how they are determined to be an error or not.

<Rounding-off of continuous values>

If the parameter is a continuous value representing, for example, distance or IOR; a number in the place one digit below the minimum effective digit is rounded to the nearest whole number.

For parameters (where resolution is not constant as in the case of vertical-axis or horizontal-axis shift because it is affected by other conditions such as the CRT screen resolution), the value is rounded to the nearest value with the current-minimum-resolution increment.

Also, if the value is subject to a range of values, its rounded value is checked to see if it within the range of values. When within the range, the value is handled as normal.

Example : The range of IOR is 1.400 000 to 1.699 999.

Therefore, value 1.399 999 5 is normal because it is rounded to 1.400 000. Value 1.699 999 49 is also normal because it is rounded to 1.699 999.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

<Rounding-off of discrete values>

If a parameter consists of a discrete value to represent an ON or OFF state as in the case of 1/0 or pulse width, a number in the place one digit below the minimum effective digit is rounded off to the nearest whole number.

If the rounded value is not a reasonable value, an error results.

Example : Assume a pulse width of 50 ns. A pulse width parameter in messages is expressed as 1 = 1 ns.

Therefore, value 50.4 is normal because it is rounded to 50, indicating a pulse width of 50 ns. Value 50.5 results in an error because it is rounded to 51.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 Rules and Definition of Notation

The following rules are used in this manual to describe remote operation.

< >	Characters enclosed with these brackets indicate a word or phrase representing a program code parameter or bus command.
:: =	This means "defined by xxx". For example, A:: = B means that A can be expressed by B in any statement which contains A.
|	Means "or" and selects one element from a list. For example, A | B means A or B, but not both.
[ ]	These brackets mean that an item enclosed with them is omittable.
{ }	When multiple items are enclosed with these brackets, one element is selected from them.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.5 Protocol

The MW9060A clears the input buffer and output queue when the power is turned on or the device clear command is received.

When the MW9060A operates as a device, the MW9060A and controller communicate by exchanging complete <program message> and <response message>. This means that the controller must always terminate <program message> before it can attempt to read a response. The MW9060A terminates <response message> at any time other than outputting hardcopy.

When a query message is sent from the controller, the next message that passes the bus is always a <response message>. The controller must read the <response message> for the query message before sending another <program message> to the MW9060A.

The controller can send two or more queries in one query message. This is called a "compound query" transfer. Multiple queries in one query message are divided by semicolons. Response for each query is also divided by a semicolon.

Commands are executed in order they are received. The same applies when Group Execution Trigger command (GET) is received. Group Execution Trigger command (GET) cannot be sent in a <program message>.

[Exceptions to the protocol]

(1) When addressed for the talker while there is no information to send

When the MW9060A has no information to send and is addressed for the talker before receiving a query, a query error results and no information is output to the bus. When the MW9060A cannot send any information because the requested query cannot be executed due to error for some reason, the MW9060A simply waits for the next message from the controller without indicating query error to it.

(2) When addressed for the talker while listener is nonexistent on the bus

When the MW9060A is addressed for the talker while the listener is nonexistent on the bus, the MW9060A waits until there is a listener or for control from the controller.

(3) Command error

When the MW9060A discovers a syntax error or undefined command header, it sends a command error to the controller.

- The message is nullified.
- The error command bit of the status byte is set to 1.
- An error message is displayed after ringing a buzzer.
- A service request is generated.

(4) Execution error

When the parameter is out of the range or the requested command or query cannot be executed with the current set value, an execution error is reported.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(5) Device-dependent error

When the MW9060A cannot execute a command for reasons inherent to the device, it sends a device-dependent error. Note, however, that the present MW9060A does not have conditions under which device-dependent errors are generated.

(6) Query error

When the appropriate query-read protocol is not followed, a query error is reported. This includes the interrupt and non-termination conditions described below.

(7) Non-termination condition

When the controller attempts to read a <response message> before terminating a <program message>, a query error is generated.

(8) Interrupt condition

When the controller attempts to send another <program message> before reading all <response messages> created by a query message, a query error is generated. In this case, the responses that have not been read are discarded. The <program message> on the interrupt side is not affected.

(9) Buffer deadlock

When both input buffer and output queue are full, a deadlock results. This state tends to occur when a very long <program message> (containing a query that creates a large amount of response data) is sent to the MW9060A. The MW9060A can receive no more of data, and the controller cannot read response data until <program message> is completely transferred. When this state is encountered, the deadlock is released by clearing the output queue and discarding all responses till the end of that <program message>. Also, the query error bit is set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 3
PREPARING FOR USE OF GPIB

Remote control for the instruments connected via the GPIB system interface is exercised by specifying the address set for each instrument. This section describes how to connect the GPIB cables and how to set addresses as a preparatory step before the GPIB can be used.

3.1 Connecting Devices with GPIB Cable

Before remote control can be applied, connect the GPIB cable first. The GPIB connector is GPIB 1 type connector mounted as standard on the rear panel. Note that the GPIB 2 mounted as an option does not operate as a device.

Before connecting the cable, turn off the power to all devices connected to the bus line.

Also turn off the power while the option GPIB board is being inserted.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2 Confirming and Setting Addresses

Each device on the GPIB has its own address. These devices are specified for the listener or talker by MLA (My Listen Address) or MTA (My Talk Address) from the controller.

Sets the MW9060A GPIB address by key operation, as described below. When settings are completed, it immediately starts operation as a device with that address. Once its address is set, the address is retained by a backup battery so that the setting does not change regardless of whether the power is on or off.

To confirm the currently set GPIB address, follow the procedure described below.

① Press [PRIOR] two times to bring the soft key hierarchy to the first layer (1/3).

② Press soft key [etc.] twice to bring the soft key hierarchy to the first layer (3/3).

③ Press soft key [SYSTEM] to bring the soft key hierarchy to the second layer SYSTEM.

④ Press soft key [INTERFACE] to bring the soft key hierarchy to the third layer INTERFACE. When this is done, the CRT screen displays the interface setting screen shown below. In this display example, the MW9060A's GPIB address is 1.

```
                              GPIB 1          GPIB 2
-------------------------------------------------------
  INTERFACE                   GPIB            GPIB
-------------------------------------------------------
  - GPIB -

  ADDRESS                     01              02
  CONTROLLER/DEVICE           DEVICE          CONTROLLER

  - PERIPHERAL ADDRESS -

  DATA STORAGE UNIT           --              19
  PRINTER                     --              16
  PLOTTER                     --              15


    Input values by turning the rotary knob and then
  press the [SET] key.
```

To change the GPIB address, turn the rotary knob on this interface screen while the item ADDRESS on GPIB 1 is reversed. When the desired address is reached, press soft key [SET].

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 4
INITIALIZATION

4.1 Initializing Bus with IFC Message

For the MW9060A Optical Time Domain Reflectometer to be controlled via GPIB, the GPIB bus must be set to the designated state by initializing it before starting control. To do this, insert the interface clear command at the beginning of the program.

For details on interface clear, refer to the item for bus commands.

Example :

```
IFC @1
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4.2 Initializing Message Exchange with DCL and SDC Bus Commands

When controlling the MW9060A Optical Time Domain Reflectometer via GPIB, its parts related to GPIB message exchange must also be initialized. To do this, execute device clear or selective device clear before starting GPIB control.

For details on device clear (DCL) and selective device clear (SDC), refer to the paragraph 2.2.4 for bus commands.

Example :

```
DCL @1
DCL @1Ø1
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4.3 Initializing Equipment with ✳RST Command

The ✳RST command is one of the common commands under IEEE488.2. This command initializes each parameter of the MW9060A Optical Time Domain Reflectometer itself.

For the contents of initialization, refer to the section 7 for ✳RST among common commands.

Example :

```
WRITE @101: "*RST"
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4.4 Initializing Equipment with INI Command

The INI command corresponds to the [Initialize] key of the MW9060A. Its contents of processing are the same as those of *RST command.

For the contents of initialization, refer to the item of the [Initialize] key in the main frame operation manual.

Example :

```
WRITE @101: "INI"
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4.5 Equipment State immediately after Power-on

The state of the MW9060A Optical Time Domain Reflectometer (when its power is turned on) is the same as when the power was turned off last.

However, when the power is turned on after replacing a plug-in unit by turning the power off, each parameter is initialized. The initialized items and initial values are the same as those of the *RST command message. Refer to the detailed description on section 7 for the *RST command message.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4.6 Activating Equipment

To activating the instruments connected to the GPIB bus simultaneously; use a bus command, Group Execution Trigger (GET).

When Group Execution Trigger (GET) is received, the MW9060A starts measurement after turning laser and average on.

Example :

```
TRG @1
```

To start measurement with an MW9060A's device message, use the laser-on command.

Example :

```
WRITE @101: "LD 1"
```

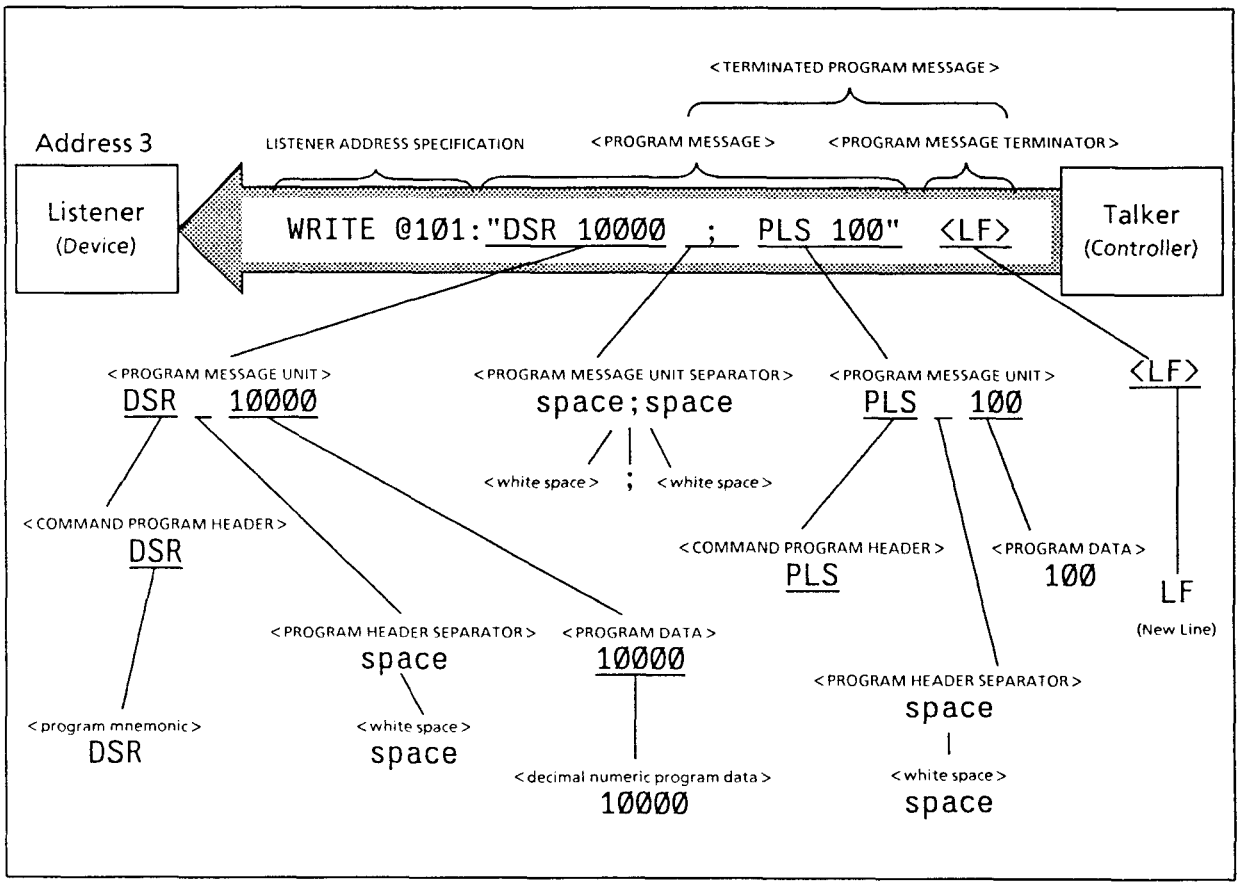

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 5
LISTENER INPUT FORMAT

Data messages come in two types: program messages and response messages. This section describes the format of the program messages received by the listener.

Program messages consist of a sequence of program message units, with each unit comprised of program command or program query.

The diagram below shows one program message being transmitted from controller to device. It consists of two program message units intended to set the distance range to 10 Km and the pulse width to 100 ns, i.e., DSR 10000 and PLS 100, which are connected via a program message unit separator.



The program message format consists of a sequence of "functional elements" which are minimum dividable units of function. The upper-case alphabetic characters enclosed with brackets < > in the above diagram are examples of functional elements. Functional elements may further be divided into what are called "encoded elements." The lower-case alphabetic characters enclosed with brackets < > in the above diagram are examples of encoded elements.

A diagram indicating selection of functional elements in a specific path is called a functional syntax diagram. Similarly, a diagram indicating selection of encoded elements in a specific path is called

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

an encoded syntax diagram. The following pages describe the program message format using these functional and encoded syntax diagrams.

Encoded elements represent the encoded element of the actual bus necessary to send functional element data bytes to the device. Upon reception of functional element data bytes, the listener checks whether each individual element conforms correctly to the rules of encoding procedure. If any element does not conform to the rules, the listener generates a command error without interpreting it as a functional element.

5.1 Program Message Notation

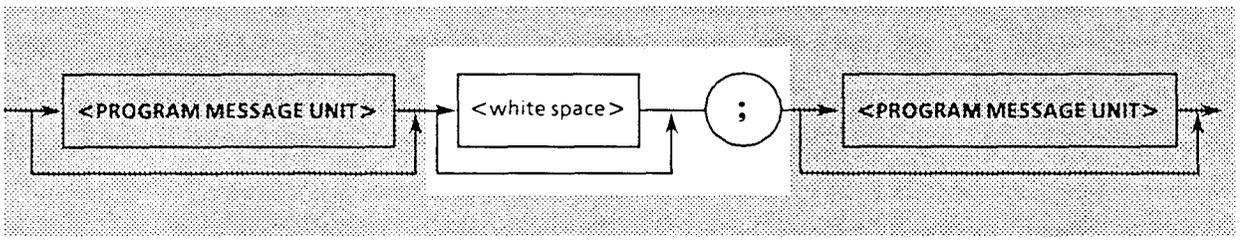
The following shows the formats of program-message functional elements and program data. (Compound and common commands are omitted.)

5.1.1 Separator, Terminator and Space Preceding Header

(1) PROGRAM MESSAGE UNIT SEPARATOR

Multiple program message units are linked using "zero or more spaces + semicolon."

Example 1 : General format linking two program message units

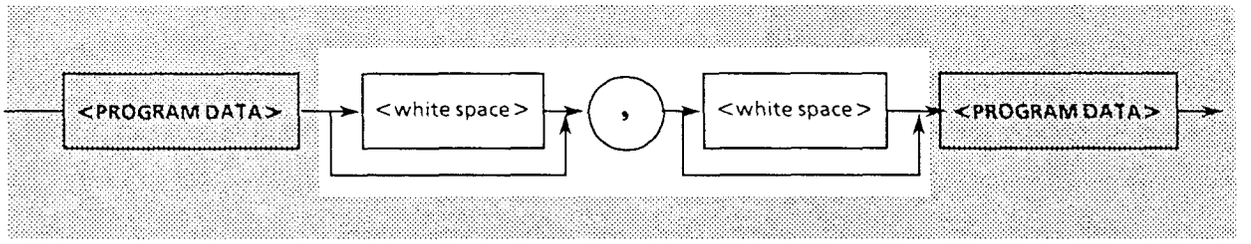


Example 2 : One space + semicolon AVG␣1 ;LD␣1

(2) PROGRAM DATA SEPARATOR

Multiple items of program data are divided by "zero or more spaces + comma + zero or more spaces."

Example 1 : General format dividing two items of program data



Example 2 : Comma only TIME 10,15

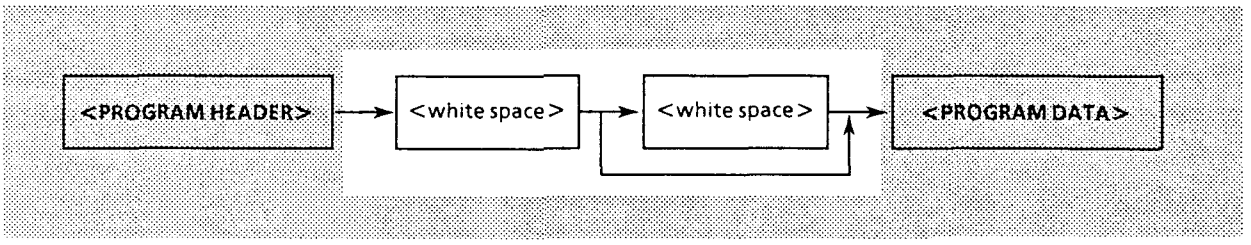
Example 3 : Comma + one space TIME 10, 15

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(3) PROGRAM HEADER SEPARATOR

The interval between the program header and program data is divided by "one space + zero or more spaces."

Example 1 : General format with single command program header

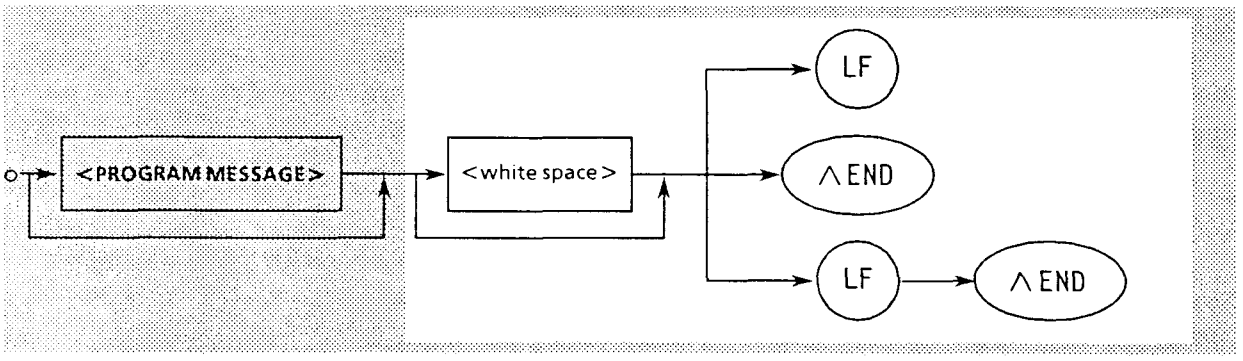


Example 2 : One space FNC Ø

(4) PROGRAM MESSAGE TERMINATOR

A program message is suffixed by "zero or more spaces + any of LF, EOI, or LF + EOI."

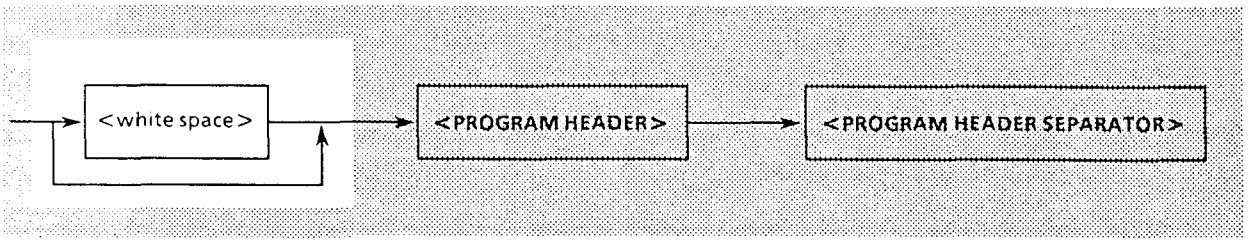
<General format>



(5) SPACE PRECEDING THE HEADER

The program header may be preceded by "zero or more spaces."

<General format>

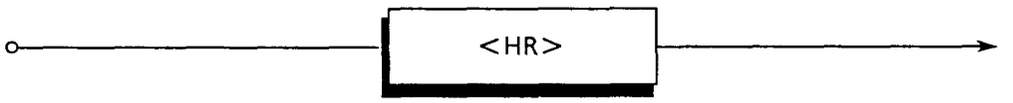


Example : One space before the second program header SPF DSR 25ØØØ ; PLS 1ØØ

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.1.2 General Format of Program Command Messages

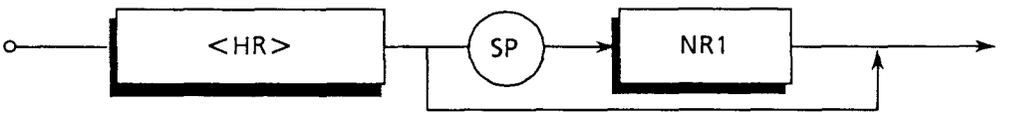
(1) Message not accompanied by any data



HR : COMMAND PROGRAM HEADER

Example : INI Initialization

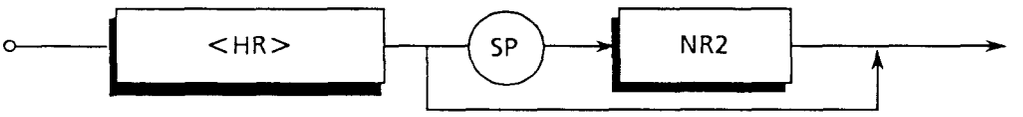
(2) Message accompanied by integer data



NR1 : Integer

Example : DSR␣50000 Sets distance range

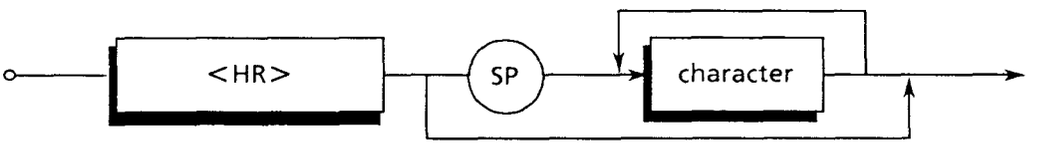
(3) Message accompanied by real number



NR2 : Real number

Example : IOR␣1,500 Sets IOR

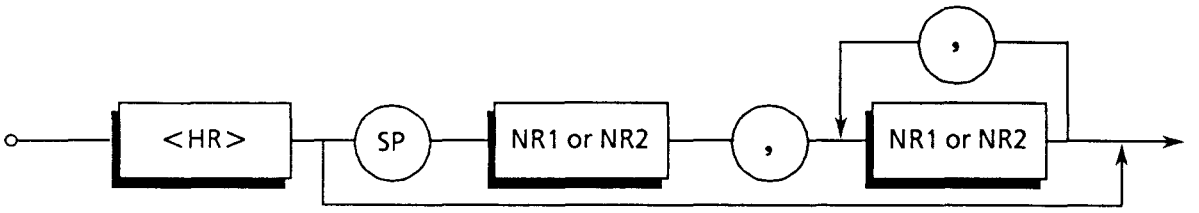
(4) Message accompanied by fixed or arbitrary character string data



Example : RCL␣TESTDATA.DAT Recalls memory data.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(5) Message accompanied by multiple program data

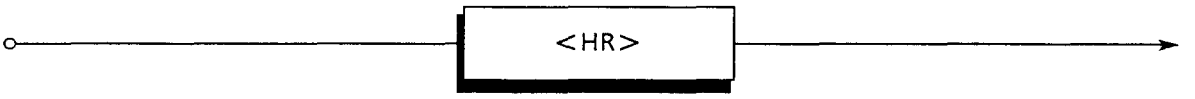


Example : DATE␣93,10,10 Sets date to October 10, 1993.

5.1.3 General Format of Query Messages

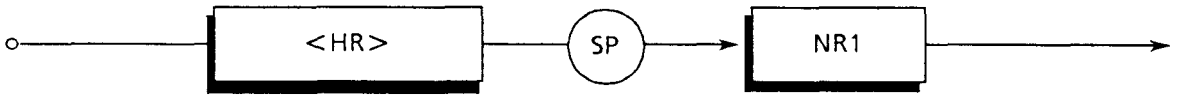
The query program header is represented by suffixing the command program header with "?."

(1) Message not accompanied by query data



Example : LOS? Requests measured results of transmission loss.

(2) Message accompanied by query data



Example : MSP?␣1 Requests set distance of mask 1.

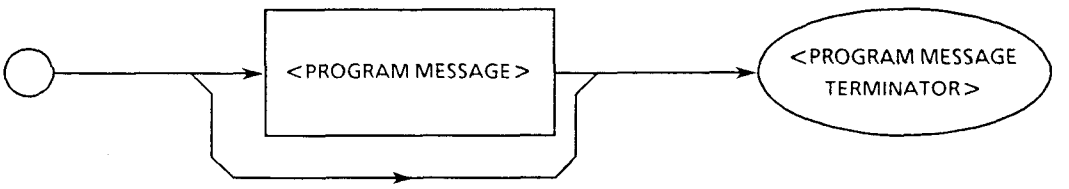
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2 Functional Elements of Program Messages

The device accepts a program message after detecting the terminator placed at the end of the program message. The following describes each functional element of this program message.

5.2.1 <TERMINATED PROGRAM MESSAGE>

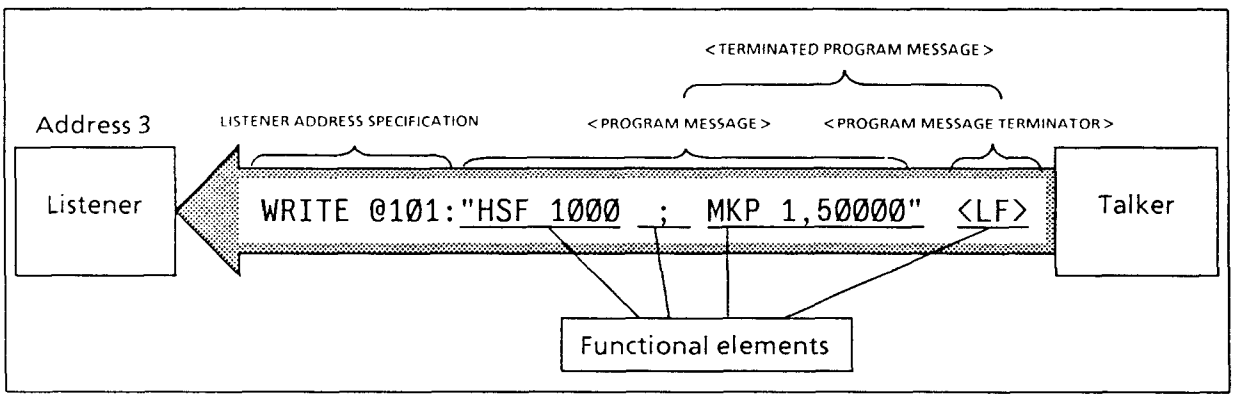
<TERMINATED PROGRAM MESSAGE> is defined as follows:



<TERMINATED PROGRAM MESSAGE> is a data message that satisfies all functional elements necessary for transmission from the controller to the listener device.

<PROGRAM MESSAGE TERMINATOR> is added after <PROGRAM MESSAGE> to complete transfer of <PROGRAM MESSAGE>.

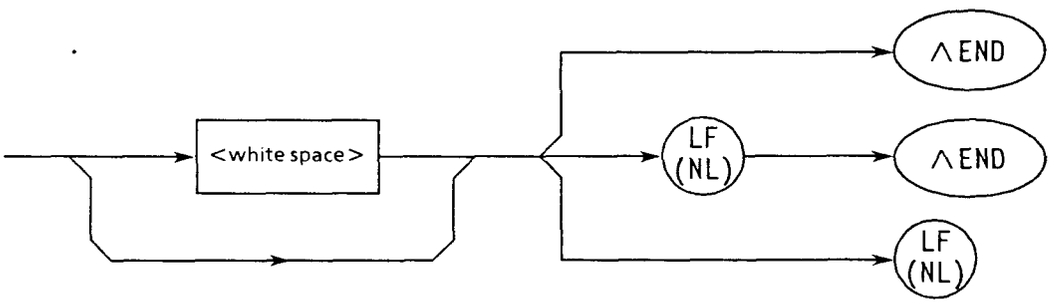
Example: <TERMINATED PROGRAM MESSAGE> to send two commands with a WRITE statement.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.2 <PROGRAM MESSAGE TERMINATOR>

<PROGRAM MESSAGE TERMINATOR> is defined as follows:

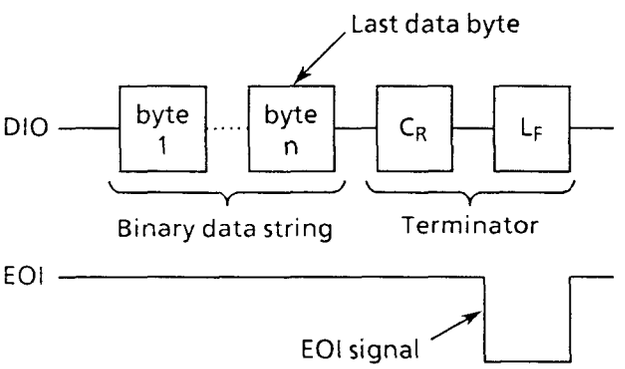


<PROGRAM MESSAGE TERMINATOR> terminates the sequence of one or more <PROGRAM MESSAGE UNIT> elements of a certain length.

LF Defined as a single ASCII code byte 0A (10 in decimal). It is ASCII control code LF (Line Feed), used to return the carriage and bring the print position to the beginning of the next line. Therefore, it is also called NL (New Line). When sending <PROGRAM MESSAGE> with a WRITE@ statement using PACKET V, there is no need to include a description to generate the CR-LF code when creating program because the WRITE@ statement automatically transmits the CR-LF code. In this case, when you want to generate LF code only, execute the following statement at the beginning of the program.

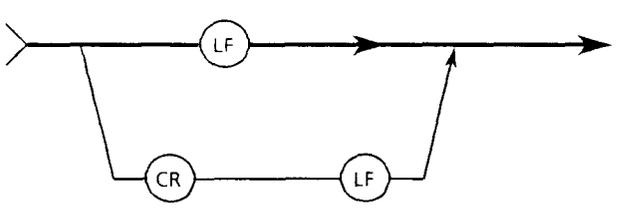
```
TERM IS CHR$(10)
```

END The EOI signal can be generated by asserting the EOI line TRUE (LOW level). (The EOI line is one of the GPIB management buses.)



One of the statements of PACKET V to control the EOI line is a EOI ON/OFF statement. The default is the same state as EOI OFF is executed and the EOI line is not controlled. By executing EOI ON beforehand, it is possible to send the EOI signal simultaneously with terminator LF when sending the last data byte of the WRITE@ statement.

<PROGRAM MESSAGE> may be terminated by the EOI signal alone without sending LF.



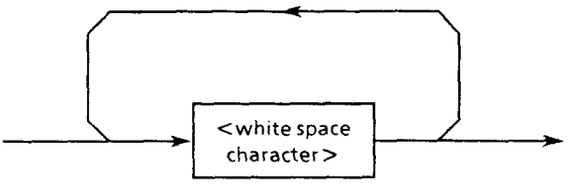
NOTE

The CR code returns the print position to the beginning of the same line, but is generally ignored on the listener side. However, because many products available on the market use the CR-LF code, most controllers are designed to output the LF code following the CR code.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.3 <white space>

<white space> is defined as follows:

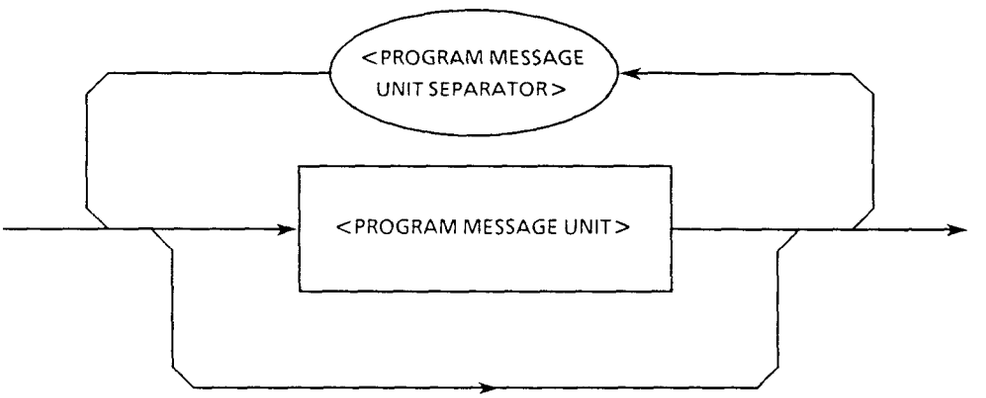


<white space> is defined as a single ASCII code byte in the range of ASCII code bytes 00 to 09, 0B to 20 (0 to 9, 11 to 32 in decimal).

Except the LF, although the range includes ASCII control and space signals, the device simply processes them as spaces or skips them over without interpreting them as ASCII control signals.

5.2.4 <PROGRAM MESSAGE>

<PROGRAM MESSAGE> is defined as follows:



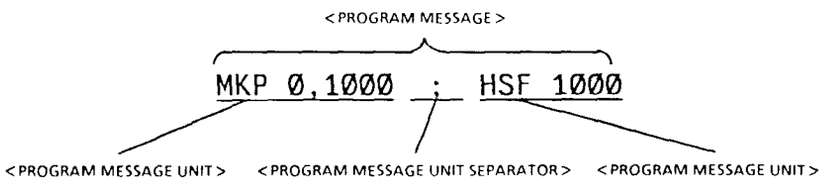
<PROGRAM MESSAGE> is a sequence of zero or one <PROGRAM MESSAGE UNIT> element or more <PROGRAM MESSAGE UNIT> elements. <PROGRAM MESSAGE UNIT> elements represent programming commands or data sent from controller to device. The <PROGRAM MESSAGE UNIT SEPARATOR> element is used as a separator to divide multiple <PROGRAM MESSAGE UNITs>.

Example 1 : Program message to set ✻marker to 1 km.

```
MKP Ø,1ØØØ
```

Example 2 : Program message to set horizontal shift to 1 km following the above setting of the example 1.

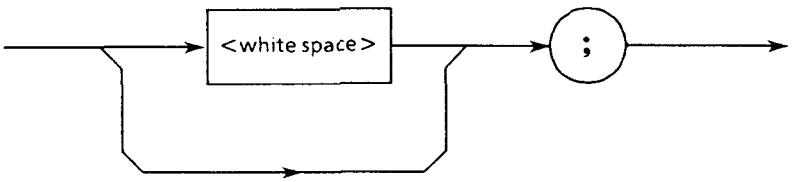
```
HSF 1ØØØ
```



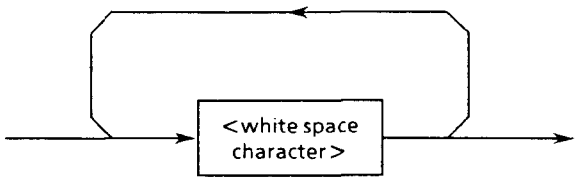
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.5 <PROGRAM MESSAGE UNIT SEPARATOR>

<PROGRAM MESSAGE UNIT SEPARATOR> is defined as follows:



<white space> is defined as follows:

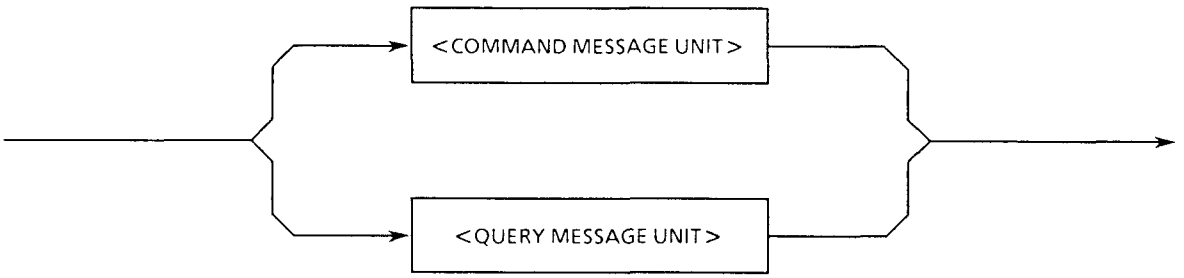


<PROGRAM MESSAGE UNIT SEPARATOR> divides the sequence of <PROGRAM MESSAGE UNIT> elements into each element of <PROGRAM MESSAGE>.

The device interprets the semicolon ";" as the separator of <PROGRAM MESSAGE UNIT>. Therefore, <white space characters> before and after the semicolon are skipped over. However, <white space characters> are effective in that they make the program easy to read. Note that if a semicolon is followed by a <white space>, it is the <white space> for the following program header.

5.2.6 <PROGRAM MESSAGE UNIT>

<PROGRAM MESSAGE UNIT> is defined as follows:



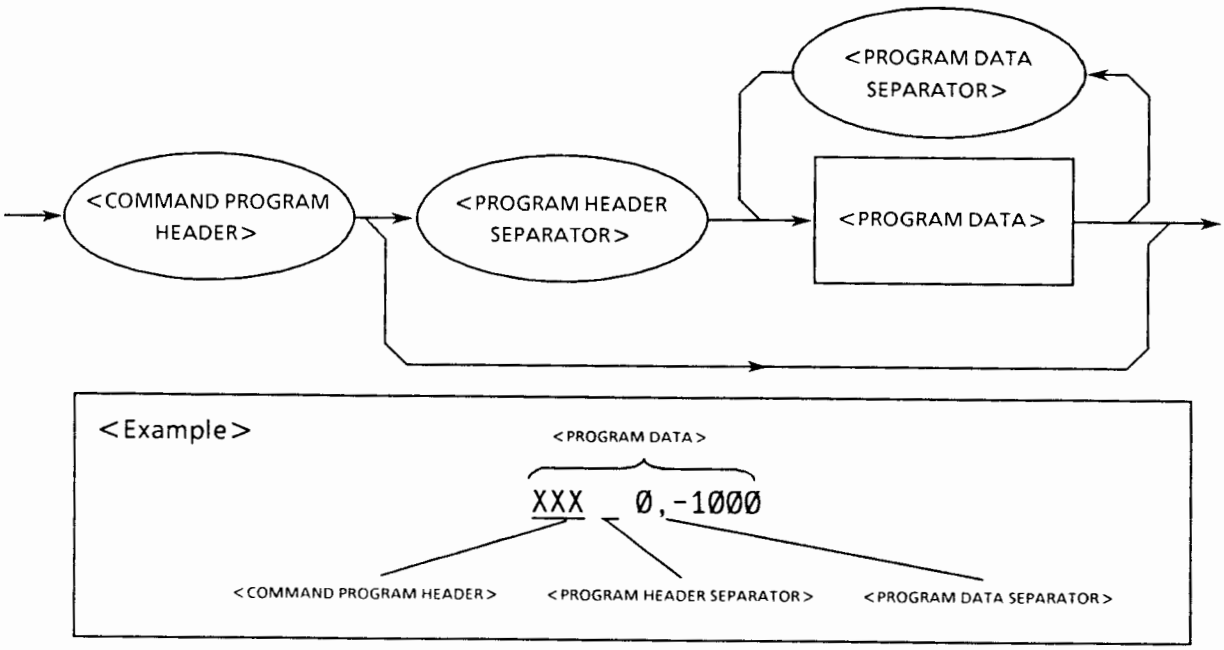
<PROGRAM MESSAGE UNIT> consists of a single command message <COMMAND MESSAGE UNIT> or a single query message <QUERY MESSAGE UNIT> received by the device.

<COMMAND MESSAGE UNIT> and <QUERY MESSAGE UNIT> are described in detail in paragraph 5.2.7.

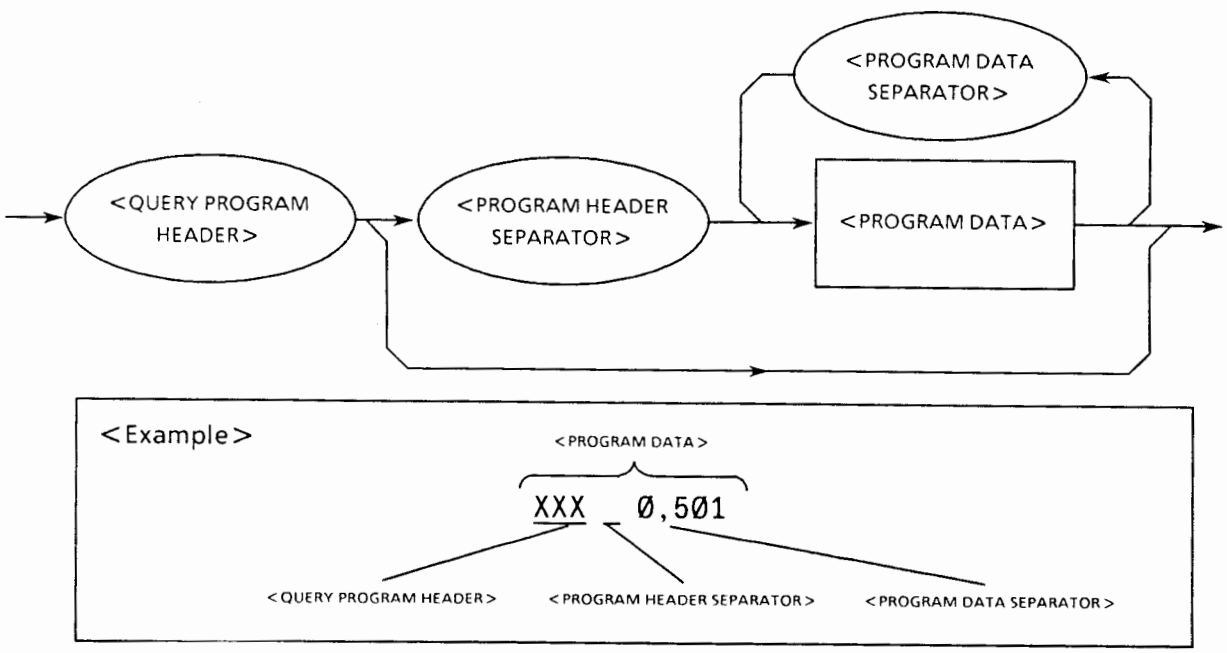
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.7 <COMMAND MESSAGE UNIT> and <QUERY MESSAGE UNIT>

1. <COMMAND MESSAGE UNIT> is defined as follows:



2. <QUERY MESSAGE UNIT> is defined as follows:



If <COMMAND MESSAGE UNIT> or <QUERY MESSAGE UNIT> is followed by program data after its program header, there is always one space between them as a separator. The program

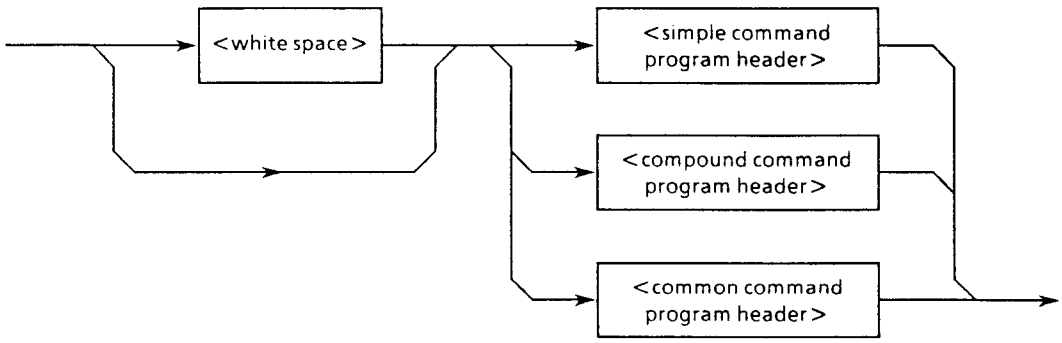
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

header tells the use, function, and operation of program data. If program data is not included, the use, function, and operation to be executed in the device are indicated by the header alone.

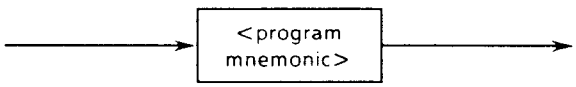
<COMMAND PROGRAM HEADER> in the program header is a command to control the device from the controller. <QUERY PROGRAM HEADER> is a query command sent from the controller to the device in advance for response messages from the device to be received by the controller. This header is unique in that it is always suffixed by "?" as a query indicator.

5.2.8 <COMMAND PROGRAM HEADER>

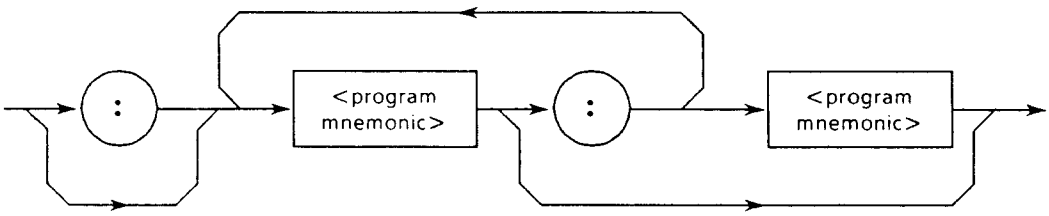
<COMMAND PROGRAM HEADER> is defined as follows:
Each header may be preceded by <white space>.



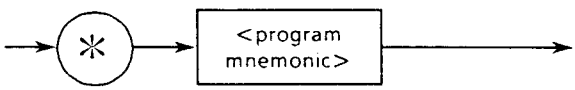
1. <simple command program header> is defined as follows:



2. <compound command program header> is defined as follows:

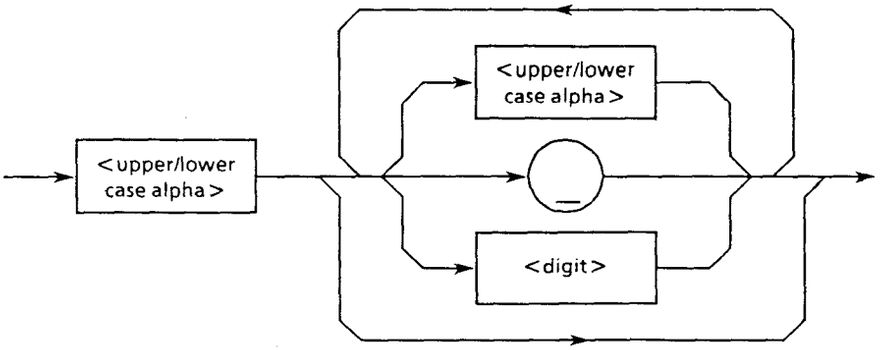


3. <common command program header> is defined as follows:



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

4. <program mnemonic> is defined as follows:



■ <COMMAND PROGRAM HEADER>

Indicates the use, function, and operation of the program data executed by the device. If program data is not included; the use, function, and operation to be executed in the device are indicated by the header alone.

<program mnemonic> indicates these meanings with ASCII code characters. The following describes the definition of mnemonic and the definitions in paragraphs 1, 2, and 3 above.

■ <program mnemonic>

The mnemonic always begins with an upper-case or lower-case alphabet. This is followed by a combination of upper-case alphabets (A to Z), lower-case alphabets (a to z), under line (_), and numbers (0 to 9). Mnemonic can be up to 12 characters long, but it normally consists of three to four upper-case alphabets. No space is included between characters.

- <upper/lower case alpha> Defined as a single ASCII code byte in the range of ASCII code bytes 41 to 5A and 61 to 7A (65 to 90 and 97 to 122 in decimal = uppercase alphabets A to Z and lower-case alphabets a to z). The header may be accepted by the device regardless of whether it is sent in upper-case or lower-case characters.
- <digit> Defined as a single ASCII code byte in the range of ASCII code bytes 30 to 39 (48 to 57 in decimal = numeric values 0 to 9).
- (_) Indicates ASCII code byte 5F (95 in decimal = underline), and is defined as a single ASCII code byte.

■ <simple command program header>

The definition of <program mnemonic> described above is applied directly as is. For example, the MW9060A uses "INI" as a "mnemonic to mean initialization," and it becomes a "simple command program header" to mean the execution of initialization without program data. "DSR" is a "mnemonic to mean distance range," and it becomes a "simple command program header" to execute the distance range setting only when it is accompanied by program data indicating the distance range.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ <compound command program header>

<compound command program header> is a command program header to execute compound functions. <program mnemonic> is always preceded by a colon ":" as the separator of <compound command program header>. If only one of this header is used, the following ":" can be omitted.

Although the MW9060A does not support this compound command program header now, this header is described here assuming expansion in the future.

- Function — This header is used for complex devices to theoretically configure a device command set by assigning it with compound functions.

Example 1 : When using all device commands of other equipment MSXXXX (provisional name) in the MW9060A.

```
:MSXXXX
```

Example 2 : When using the WXYZ command among the device commands of other equipment MSXXXX (provisional name) in the MW9060A.

```
MSXXXX:WXYZ or :MSXXXX:WXYZ
```

Example 3 : A white male rabbit living in a forest (FOREST) is named WHITE. A white female rabbit living in a forest (GROVE) is also named WHITE. If WHITE only is used as a command, it does not distinguish between these two rabbits. Therefore

```
FOREST:WHITE or : FOREST:WHITE..... Indicates the male rabbit.
GROVE:WHITE or  : GROVE:WHITE..... Indicates the female rabbit.
```

■ <common command program header>

<common command program header> always has its <program mnemonic> preceded by an asterisk " *". This command is named "common" because it is one of the common commands applied in common to all other IEEE 488.2 compatible measurement instruments connected via the bus.

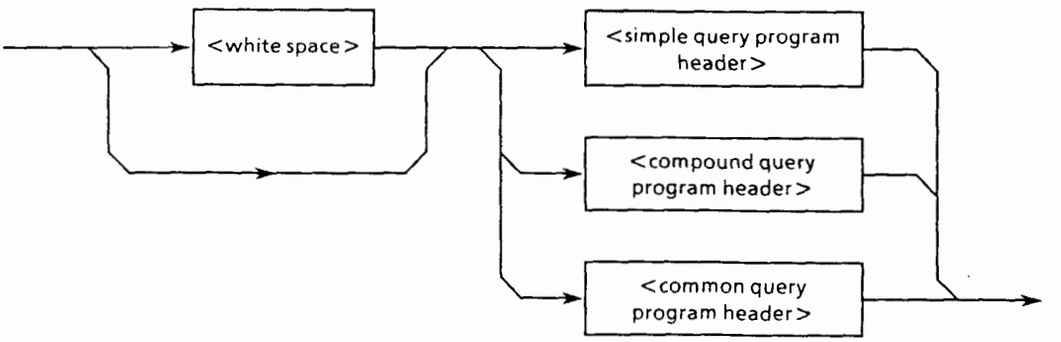
Example: When making the end of operation (in the device of address 8 connected to the GPIB interface of select code 1) idle, and resetting each device to its designated inherent state.

```
WRITE @108:"*RST"..... The above is executed by *RST, a IEEE
                       488.2 common command, enclosed with
                       quotations " ".
```

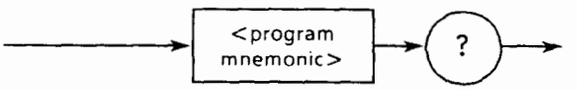

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.9 <QUERY PROGRAM HEADER>

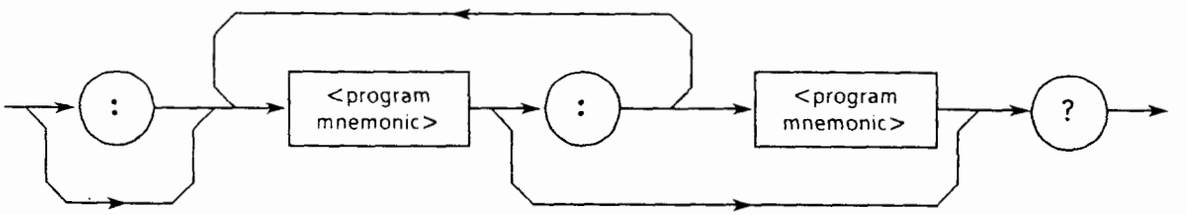
<QUERY PROGRAM HEADER> is defined as follows:
Each header may be preceded by <white space>.



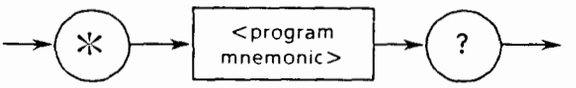
1. <simple query program header> is defined as follows:



2. <compound query program header> is defined as follows:



3. <common query program header> is defined as follows:



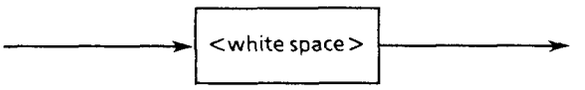
■ <QUERY PROGRAM HEADER>

<QUERY PROGRAM HEADER> is a query command sent from the controller to the device in advance for response messages from the device to be received by the controller. This header is unique in that it is always suffixed by "?" as a query indicator.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.2.10 <PROGRAM HEADER SEPARATOR>

<PROGRAM HEADER SEPARATOR> is defined as follows:



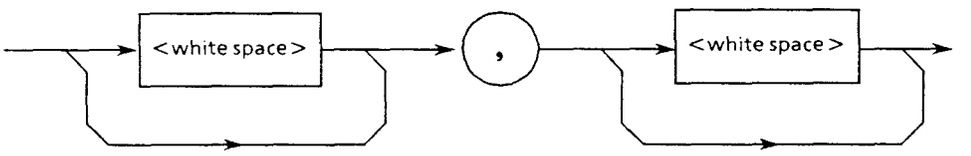
<PROGRAM HEADER SEPARATOR> is used as a separator between <COMMAND PROGRAM HEADER> or <QUERY PROGRAM HEADER> and <PROGRAM DATA>.

If there are multiple <white space> between the program header and program data, the first <white space> is interpreted as the separator, with all other <white space> skipped over. However, <white space>, are effective in that they make the program easy to read.

Thus, there is always one header separator between the header and data, indicating the end of the program header as well as the beginning of program data.

5.2.11 <PROGRAM DATA SEPARATOR>

<PROGRAM DATA SEPARATOR> is defined as follows:

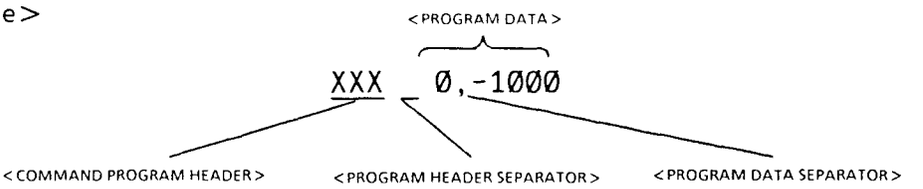


When <COMMAND PROGRAM HEADER> or <QUERY PROGRAM HEADER> has multiple parameters, <PROGRAM DATA SEPARATOR> is used to divide them.

When using this data separator, always use a comma, but need not necessarily use <white space>.

<white space> before or after the comma are skipped over. However, <white space> are effective in that they make the program easy to read.

<Example>

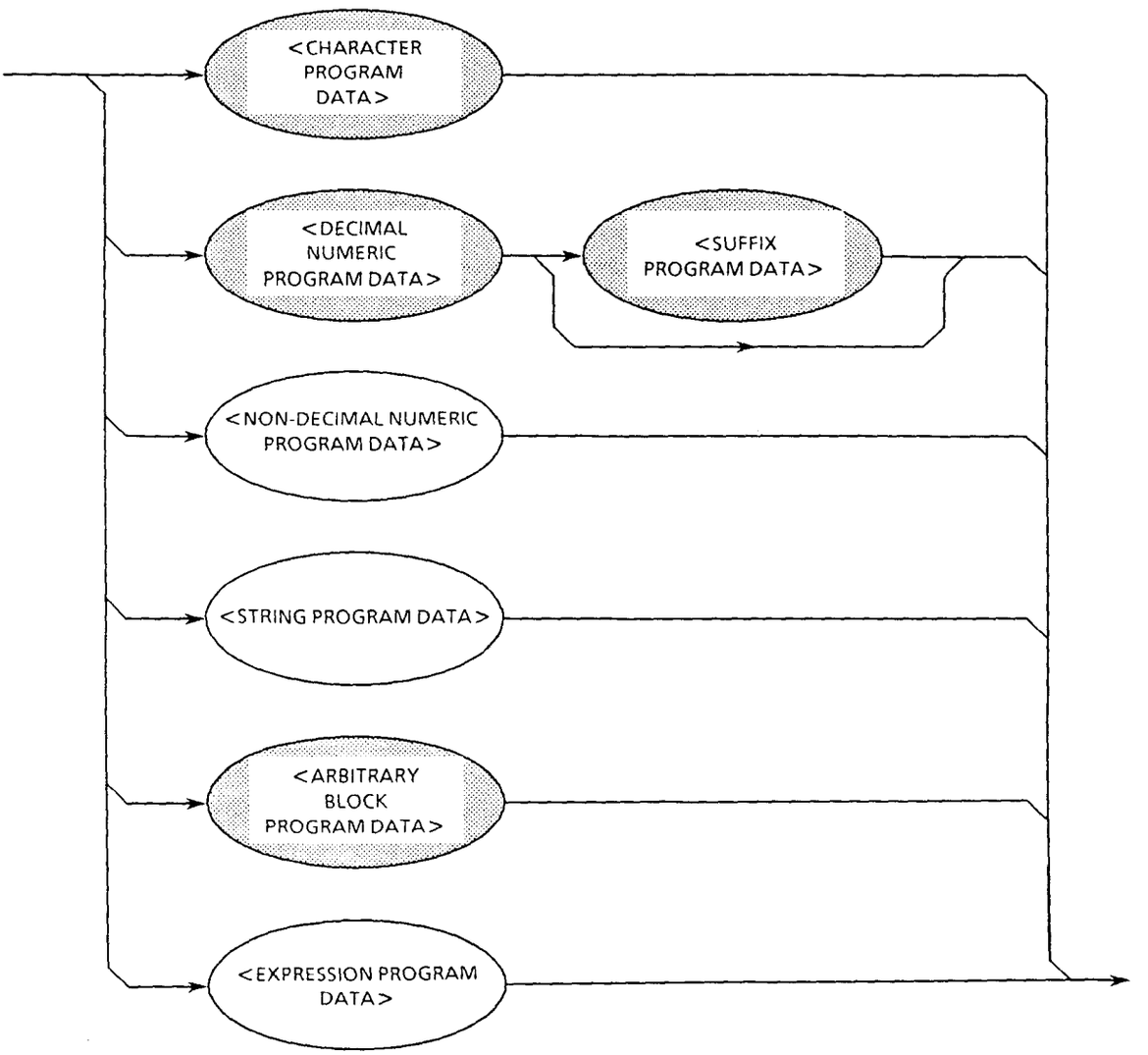


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3 Program Data Format

The following describes the format of <PROGRAM DATA> indicated in function syntax diagrams.

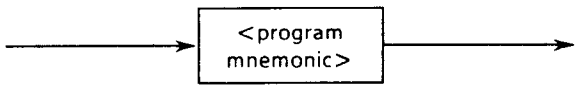
<PROGRAM DATA> functional elements are used to transmit various types of parameters related to the program header. The diagram below shows the types of these program data. Among these types of program data, the MW9060A accepts those which are indicated on white background in the shaded ovals. The program data not used with the MW9060A shown here are for reference purposes.



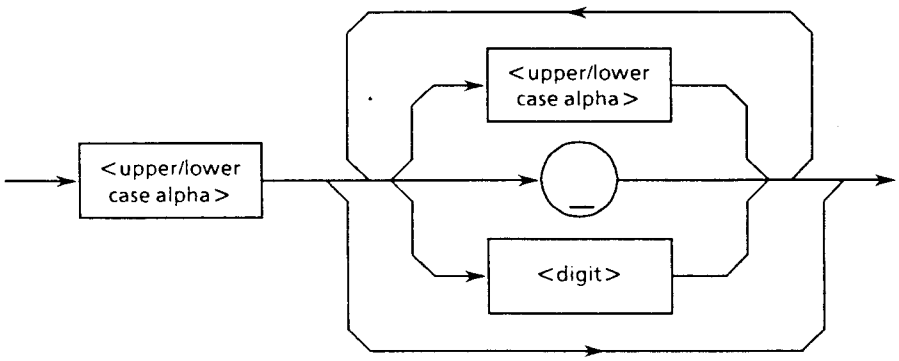
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.1 <CHARACTER PROGRAM DATA>

<CHARACTER PROGRAM DATA> elements are used for remote control by transmitting abbreviated alphabetic or alphanumeric data, and is defined as follows:



The contents of character data are the same as those of program mnemonic. Although control data is normally comprised mainly of numeric data, control can also be exercised using this character program data. Shown below is its encoded syntax diagram.



The data always begins with an upper-case or lower-case alphabet. This is followed by a combination of upper-case alphabets (A to Z), lower-case alphabets (a to z), underline (_), and numbers (0 to 9). This combination of alphanumeric characters are used as symbols as for mnemonic. The data can be up to 12 characters long, with no space included between characters.

- <upper/lower case alpha> Defined as a single ASCII code byte in the range of ASCII code bytes 41 to 5A and 61 to 7A (65 to 90 and 97 to 122 in decimal = upper-case alphabets A to Z and lower-case alphabets a to z).
  The data may be accepted by the device regardless of whether it is sent in upper-case or lower-case characters.
- <digit> Defined as a single ASCII code byte in the range of ASCII code bytes 30 to 39 (48 to 57 in decimal = numeric values 0 to 9)
- (_) Indicates ASCII code byte 5F (95 in decimal = underline), and is defined as a single ASCII code byte.

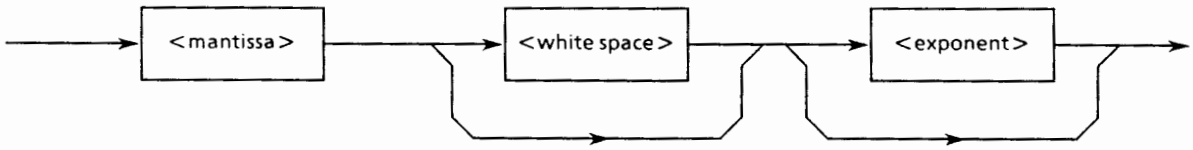
Thus, <CHARACTER PROGRAM DATA> is a program data intended to send alphanumeric symbols of a relatively short mnemonic type.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

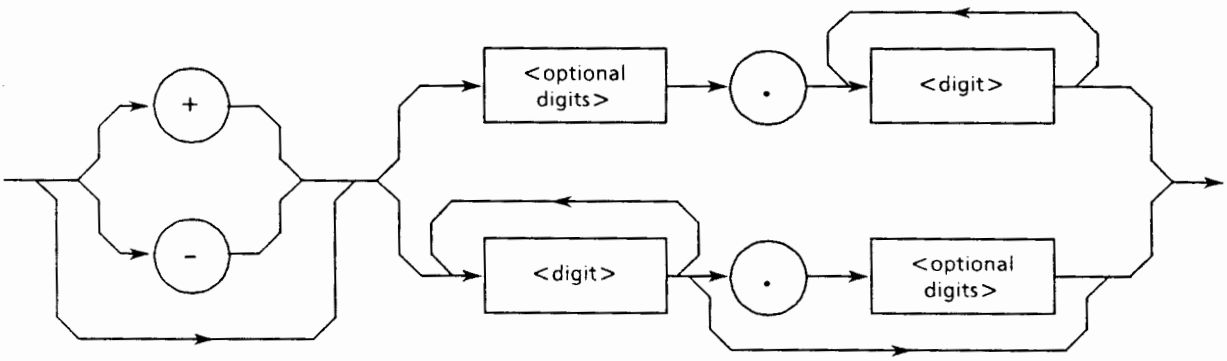
5.3.2 <DECIMAL NUMERIC PROGRAM DATA>

<DECIMAL NUMERIC PROGRAM DATA> is a program data to transmit numeric constants expressed by decimal notation. Decimal numbers may be expressed as an integer type, fixed-point type, or floating-point type.

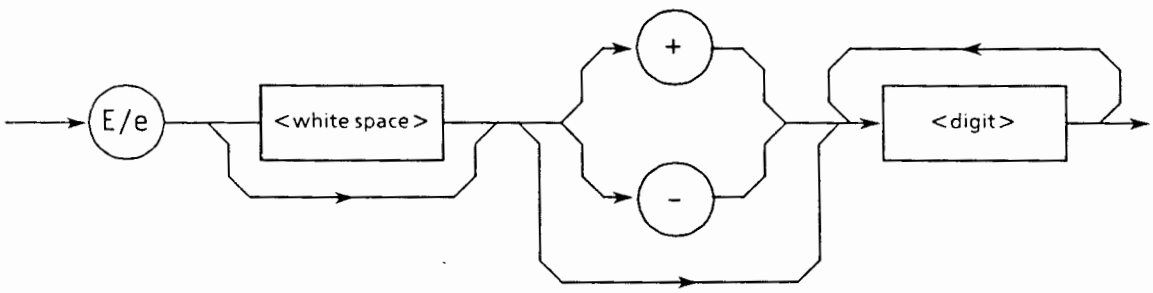
These three types of numeric notation are used to enable NRF (flexible numeric representation) of program data consisting of decimal numbers, and are defined as shown by the encoded syntax diagram below.



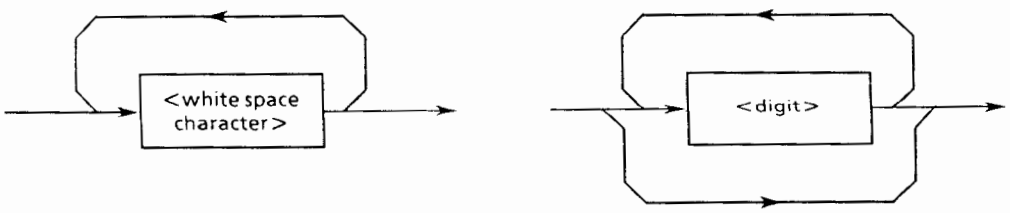
<mantissa> is defined as follows:



<exponent> is defined as follows:



<white space> and <optional digit> are defined as follows:



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

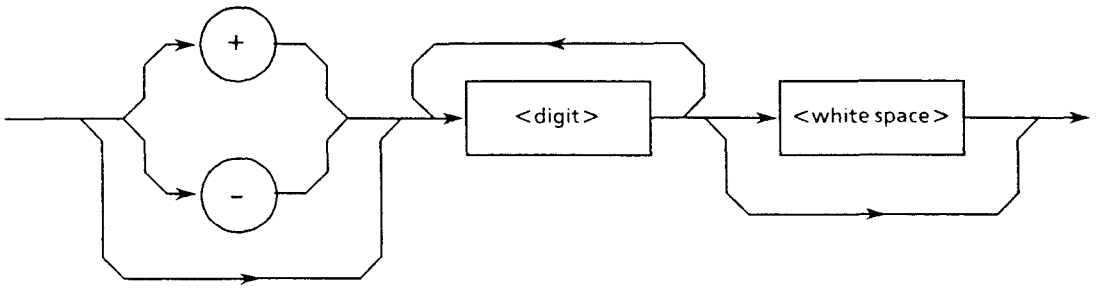
The encoded syntax diagram of decimal number program data described above will now be explained separately for the integer, fixed-point, and floating-point types when transmitting these program data.

Note that in any type of transmission, the following processes are performed.

- Rounding-off of numeric elements — When the device receives a <DECIMAL NUMERIC PROGRAM DATA> element consisting of more digits than can be handled internally, it rounds off the element to the nearest whole number while ignoring its sign (±).
- Off-range data — If the value of a <DECIMAL NUMERIC PROGRAM DATA> element is out of the allowed range, an execution error is reported.

(1) Integer type – NR1 transmission

The example below transmits integer (NR1), that is, a decimal number without containing any decimal point or exponential expression.

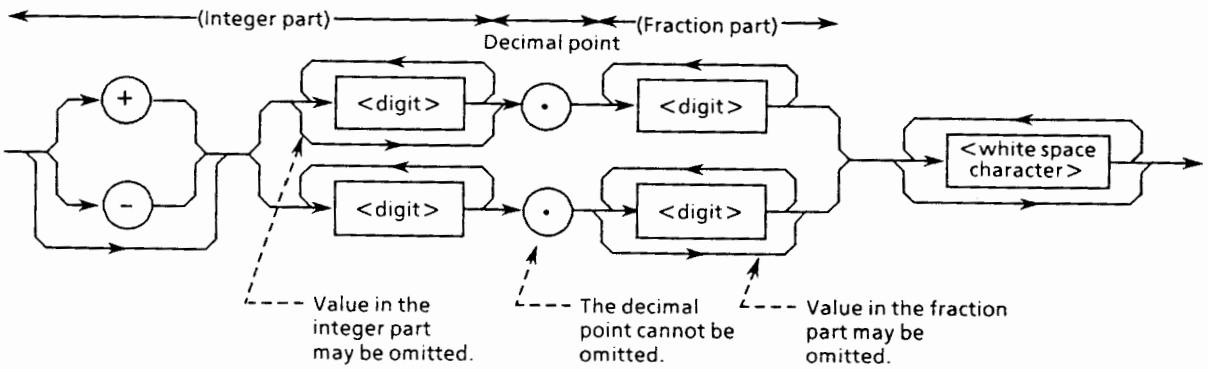


- Zero (0) can be inserted at the beginning.
  Example: 005, + 000045
- No space can be inserted between sign (+ or –) and the number.
  Example: +5, + ␣5 (invalid)
- Space (s) can be inserted after the number.
  Example: +5␣␣␣
- Sign "+" may be or need not be added.
  Example: +5, 5
- A comma cannot be used to divide digits.
  Example: 1,234,567 (invalid)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(2) Fixed-point type – NR2 transmission

The example below transmits real number (NR2), that is, a decimal number containing digits below the decimal point, except an integer or exponential expression. The syntax diagram consists of (integer part), decimal point, and (fraction part).

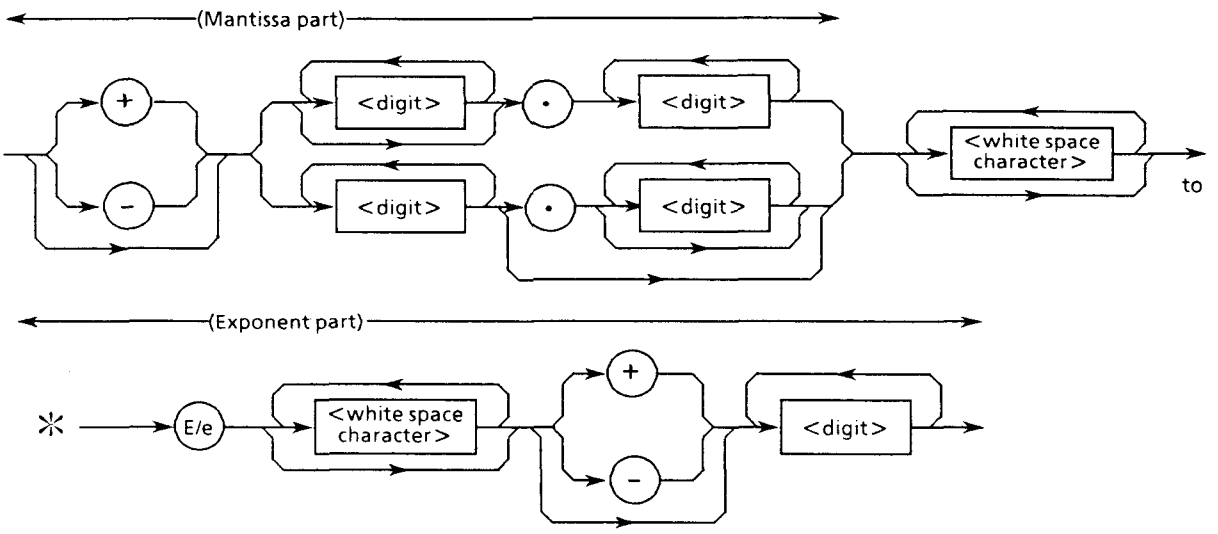


- The numeric representation of the integer type is applied to the (integer part).
- No space can be inserted between the number and decimal point.
  Example: +753␣.123 (invalid)
- Space (s) can be inserted after the number in the (fraction part).
  Example: +753.123␣␣␣␣
- The decimal point need not always be preceded by a number.
  Example: .05
- A sign can be placed before the decimal point.
  Example: +.05, –.05
- A number can be ended with the decimal point.
  Example: 12.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(3) Floating-point type – NR3 transmission

The example below transmits real number (NR3) expressed by a floating-point format, that is, a decimal number containing digits of exponential expression. The syntax diagram consists of (mantissa part) and (exponent part). The mantissa part is expressed by the integer or fixed-point format to indicate the precision of the value. The exponent part begins with E, followed by a value indicating the power of 10.

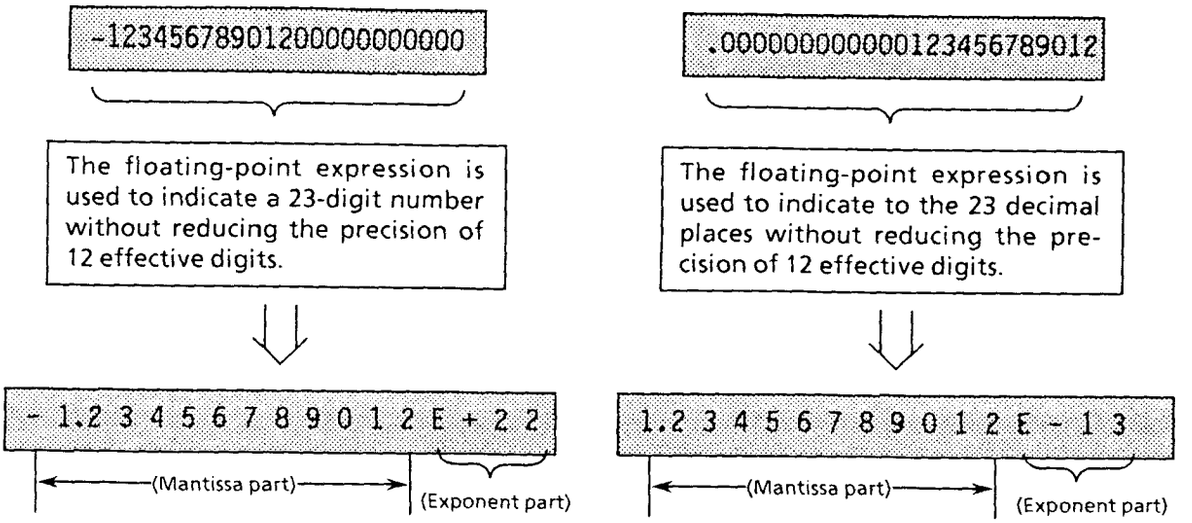


- E denotes the power of 10, and indicates the beginning of the exponent part.
- E may be entered in either the upper or lower case.
  Example: 1.234E+12, 1.234e+12
- A space can be placed before and after or either before or after E (or e).
  Example: 1.234␣E␣+12
- Preceding zeros must be omitted in the exponential expression.
  Example: 1.234E+4, 1.234E+04 (invalid)
- If the sign is +, the sign + in both the mantissa and exponent parts can be omitted.
  Example: +1.234E+4, 1.234E4
- Numbers in the mantissa part cannot be omitted.
  Example: −1E2, −E2 (invalid), −.E2 (invalid)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ Floating-point notation

The following describes range restrictions, etc. applicable to the mantissa and exponent parts using a number consisting of 12 effective digits as an example.



- Sign of the value — If the value is negative, the negative sign (−) is placed before the effective number at the left of the mantissa part. If the value is positive, the positive sign (+) may be omitted.

- Mantissa part — Indicates the effective number in the effective digit part, that is, the precision of the value.
  The value in the mantissa part does not exceed 255 without leading 0s. If this range is exceeded, a command error results. If, when expressing a value by the integer or fixed-point notation, the precision of the value cannot be expressed within its effective digits, the mantissa part makes it possible.

- Exponent part — Indicates the number of shiftings of unit position. The number of the exponent is in the range of −32000 to +32000. If this range is exceeded, a command error results. Leading zeros are omitted. The sign may be either + or −. The sign + may be omitted.

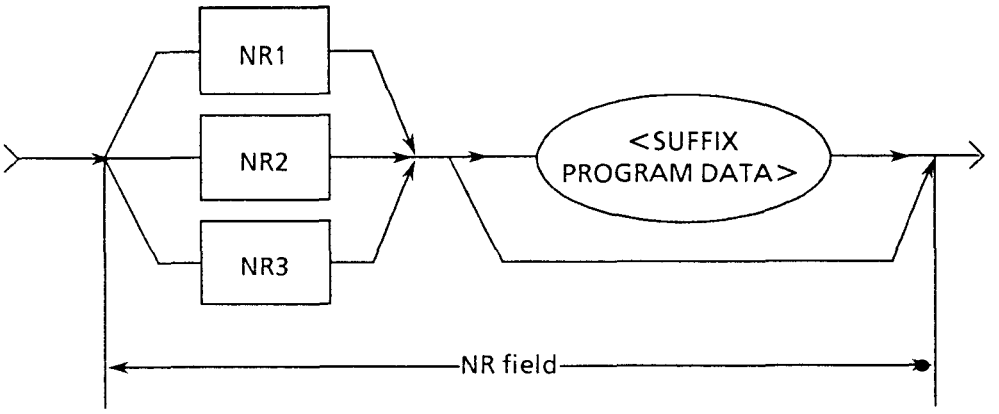
  + sign ..... E is followed by positive sign (+), with (exponent value + 1) indicating the number of digits of the value.

  − sign ..... E is followed by negative sign (−), indicating the decimal place at which the effective number begins.

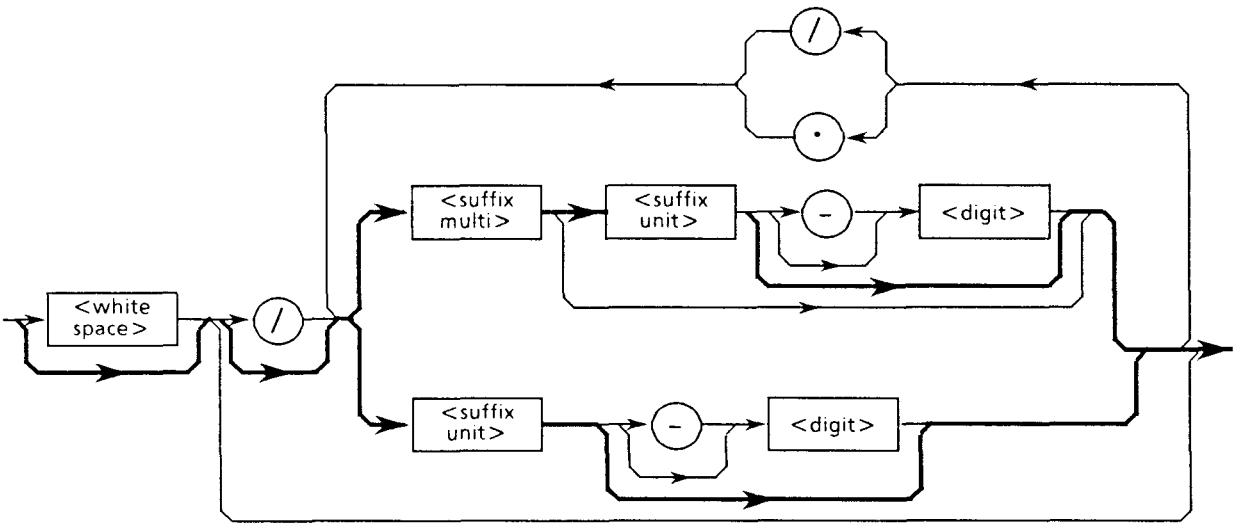
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.3 <SUFFIX PROGRAM DATA>

<SUFFIX PROGRAM DATA> is used following the <DECIMAL NUMERIC PROGRAM DATA> (integer type NR1, fixed-point type NR2, or floating-point type NR3) described above, allowing each type of numeric representation to be suffixed as necessary.



The suffix is added for decimal number program data when the unit of measurement is required for that data. It is used independently as suffix unit or as a combination of suffix unit and suffix multipliers. Its syntax diagram is as shown below. The generally used routes are indicated by thick lines.



- A suffix multiplier is expressed with upper-case or lower-case alphabets. For example, 1E3HZ is expressed as 1 KHZ where 1E3 = 1 K.
- A suffix unit is expressed with upper-case or lower-case alphabets.
- <SUFFIX PROGRAM DATA> cannot be preceded by letter E because it may be confused with the E that is used in floating-point numeric representation.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The following shows suffix multipliers and suffix units. (These are not used on the MW9060A.)

(1) Suffix multipliers

Table 5-1 Suffix Multipliers

Multiplier	Mnemonic	Name
1 E 1 8	EX	E X A
1 E 1 5	PE	P E T A
1 E 1 2	T	T E R A
1 E 9	G	G I G A
1 E 6	MA (Note)	M E G A
1 E 3	K	K I L O
1 E – 3	M (Note)	M I L L I
1 E – 6	U	M I C R O
1 E – 9	N	N A N O
1 E – 1 2	P	P I C O
1 E – 1 5	F	F E M T O
1 E – 1 8	A	A T T O

Note: According to conventional tradition, 10^6 of HZ is represented by MHZ (megahertz) and 10^6 of OHM is represented by MOHM (megohm). These are listed in the table of suffix units in Table 5-2, and not included in the above table of suffix multipliers.

(2) Relative units (dB)

- dB value for 1 μV ..... DBUV
- dB value for 1 μW ..... DBUW
- dB value for 1 mW ..... DBMW

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(3) Suffix units

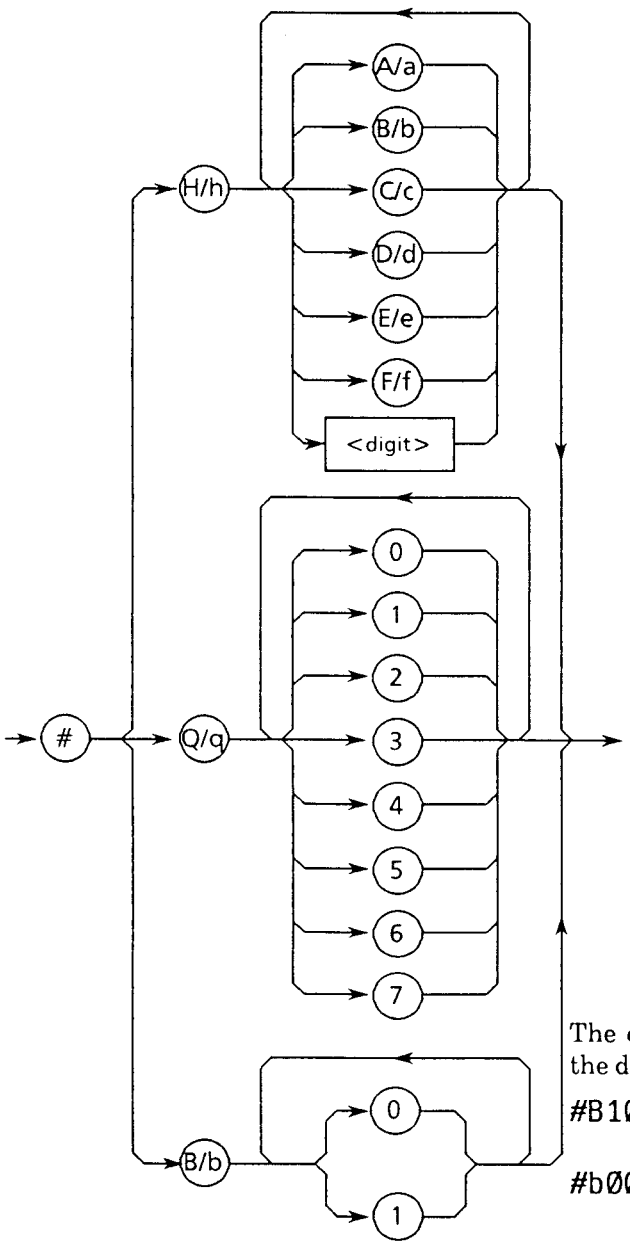
Table 5-2 Suffix Units

Item	Recommended unit mnemonic	Quasi used unit mnemonic	Name
Current	A		Ampere
Atmospheric pressure	ATM		Atmosphere
Charge	C		Coulomb
Luminous intensity	CD		Candela
Decibel	DB		Decibel
Power	DBM		Decibel milliwatt
Capacitance	F		Farad
Mass		G	Gram
Inductance	H		Henry
Frequency (hertz)	HZ		Hertz
Mercury column	INHG		Inches of mercury
Joule	J		Joule
Temperature	K		Degree Kelvin
		CEL	Degree Celsius
		FAR	Degree Fahrenheit
Volume	L		Liter
Luminous intensity	LM		Lumen
Luminous intensity	LX		Lux
Length (meter)	M		Meter
		FT	Feet
		IN	Inch
Frequency (1E6Hz)		MHZ	Megahertz
Resistance		MOHM	Megohm
Force	N		Newton
Resistance	OHM		Ohm
Pressure	PAL		Pascal
Ratio (percent)	PCT		Percent
Angle (radian)	RAD		Radian
Angle (degree)		DEG	Degree
		MNT	Minute (of arc)
Time (second)	S	SEC	Second
Conductance	SIE		Siemens
Automatic speed	T		Tesla
Pressure	TORR		Torr
Voltage	V		Volt
Power (watt)	W		Watt
Hourly speed	WB		Weber
Luminous intensity	LM		Lumen

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.4 <NON-DECIMAL NUMERIC PROGRAM DATA>

<NON-DECIMAL NUMERIC PROGRAM DATA> is non-decimal numeric program data used to transmit hexadecimal, octal, and binary numeric data. Non-decimal data always begins with the # mark. It is defined as the encoded syntax diagram shown on the left below. However, the MW9060A handles only the decimal number. Transmitting row of character string other than those specified results in a command error.



The characters following #H or #h are accepted by the device as an unsigned hexadecimal. Figures in ( ) indicate corresponding decimal numbers.

```
#Habc123Ø        (11,256,099D)
#hAbc123
#H2DC3           (11.715D)
#h2dc3
#H83Ø1           (33,537D)
#h83Ø1
```

The characters following #Q or #q are accepted by the device as an unsigned octal.

```
#Q37             (31D)
#q37
#Q267Ø3          (11.715D)
#q267Ø3
#q267Ø3
```

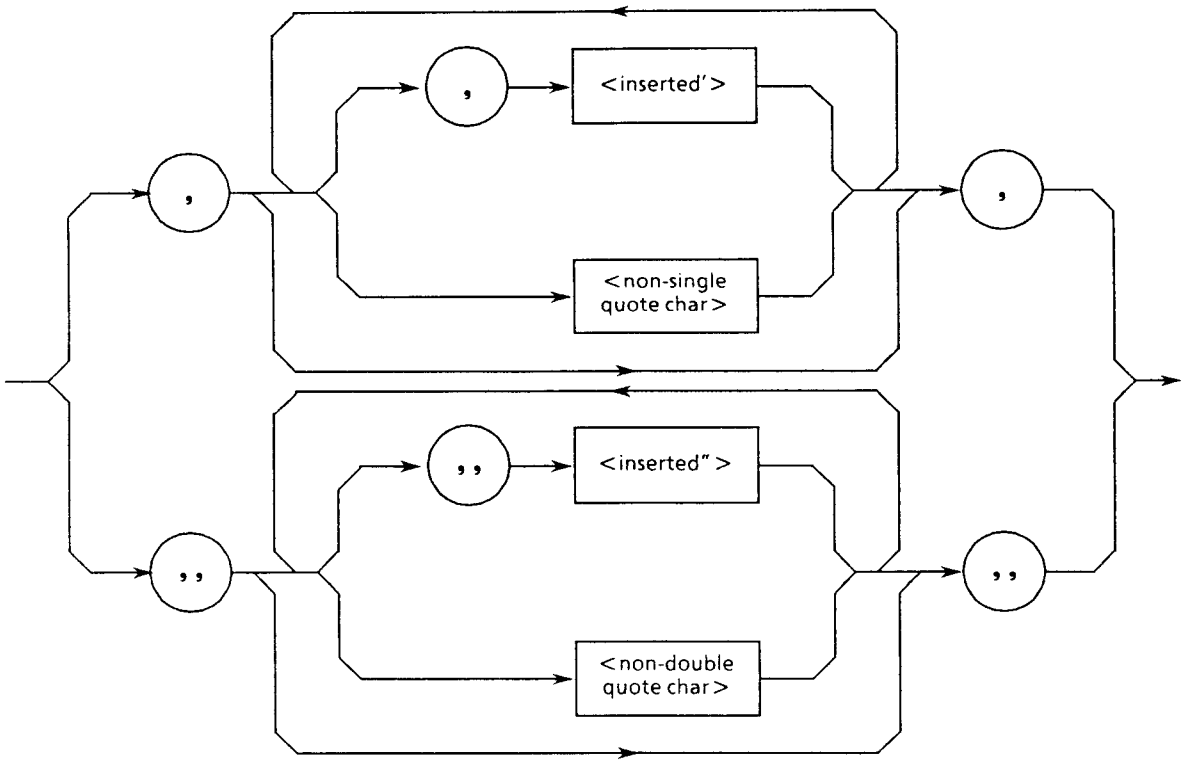
The characters following #B or #b are accepted by the device as an unsigned binary.

```
#B1Ø1Ø1Ø111100ØØØ1ØØ1ØØØ11
                          (11,256,099D)
#bØØ1Ø11Ø111ØØØØ11        (11,715D)
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.5 <STRING PROGRAM DATA>

<STRING PROGRAM DATA> is the program data exclusively used for character strings. All of ASCII 7-bit codes can be used. However, single and double quotations used in a character string must always come as a pair of the same type, one immediately following the other.



- Both ends of a character string must be enclosed with single or double quotations regardless of whether quotations are used within the character string. For example,

```
It's a nice day →  "It's a nice day."
                →  'It's a nice day.'
```

- When both ends of a character string are enclosed with single quotations, the single quotations used within the character string must be repeated (i.e., ''). Other characters, including double quotations, may be described as are. For example,

```
"I shouted, 'Shame'."  →  '"I shouted, ''Shame''."'
```

- When both ends of a character string are enclosed with double quotations, the double quotations used within the character string must be repeated (i.e., ""). Other characters, including single quotations, may be described as are. For example,

```
"I shouted, 'Shame'."  →  """I shouted, 'Shame'."""
```

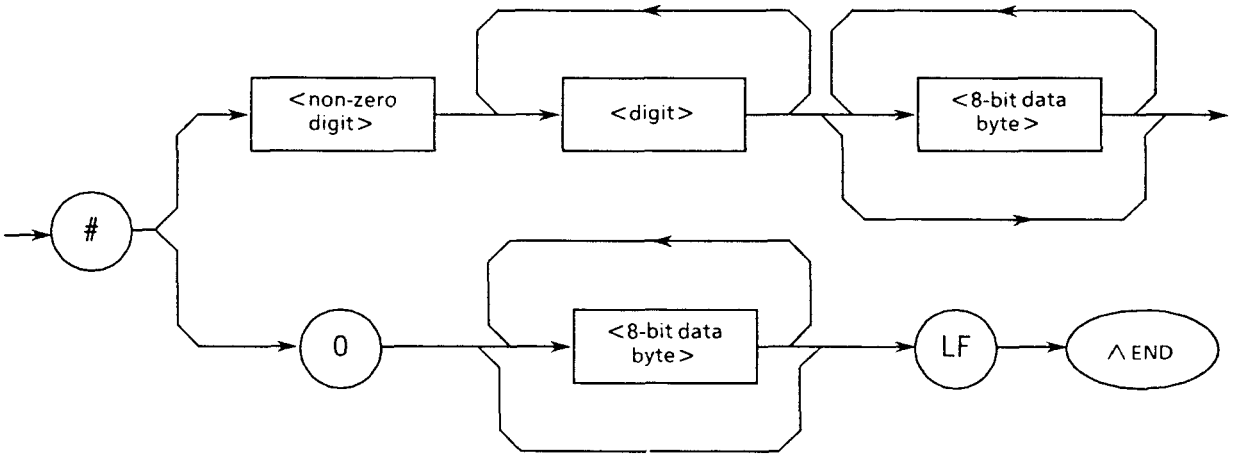
- <inserted'> is a single ASCII symbol of ASCII code byte 27 (39 in decimal = symbol '); <inserted"> is a single ASCII symbol of ASCII code byte 22 (34 in decimal = symbol "). <non-single quote char> and <non-double quote char> are respectively single ASCII symbols other than single and double quotations.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

5.3.6 <ARBITRARY BLOCK PROGRAM DATA>

<ARBITRARY BLOCK PROGRAM DATA> is a non-decimal program data that begins with the # mark. It directly transmits binary data using 1 byte = 8 bits as a minimum block without changing the data format. The differences with the <NON-DECIMAL NUMERIC PROGRAM DATA> described earlier are:

- It handles both character string data and numeric data without being limited to numeric data.
- It has information indicating the number of data bytes, etc. sent, between # and first data.



Thus, this non-decimal data allows the data bytes to be transferred to be specified as necessary, and is defined as follows:

- <digit> Is a number in the range of ASCII code bytes 30 to 39 (48 to 57 in decimal = numbers 0 to 9) and defined as a single ASCII code byte.
- <non-zero digit> Is a number in the range of ASCII code bytes 31 to 39 (49 to 57 in decimal = numbers 1 to 9) and defined as a single ASCII code byte.
- <8-bit data byte> Is an 8-bit byte in the range of 00 to FF (0 to 255 in decimal).

(1) When the number of data bytes to be sent is known

This applies to the upper-right route in the above syntax diagram. The number of bytes in <8-bit data byte> to be transferred is specified at the position of <digit> in the diagram, that is, immediately before the beginning of writing data. Then, the number of digits in the specified number of bytes is written between # and <digit>, that is, at the position of <non-zero digit>. For example, 4-byte data bytes (DAB) to be sent may be described as follows:

Specify 4 at the position of <digit> because 4 bytes are sent.

```
#14<DAB><DAB><DAB><DAB>
```

The value of <non-zero digit> is 1 because 4 at the position of <digit> on the right is one digit.

Specify 4 at the position of <digit> because 4 bytes are sent. May also be specified with leading zeros.

```
#3004<DAB>DAB><DAB><DAB>
```

The value of <non-zero digit> is 3 because 4 at the position of <digit> on the right is three digits (i.e., 004).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(2) When the number of data bytes to be sent is unknown

This applies to the lower-right route shown in the syntax diagram in the preceding page. Place #0 before the first data. The last data is terminated with LF^END.

```
#Ø<DAB>DAB><DAB><DAB><DAB>LF^END
```

LF^END is such that when the statement shown below is executed beforehand in the beginning part of the program, the EOI signal is sent out as the END signal simultaneously with terminator LF when the last data byte is transmitted.

- For LF `TERM IS CHR$(1Ø)`
- For END `EOI ON`

(3) Handling of integer-precision binary data

Integer-precision binary data is used as transfer data of the <ARBITRARY BLOCK> format for both program data and response data. Its specification is as shown below.
Negative numbers are processed as a two's-complement form.

Number of bytes transferred	1, 2, 4, or 8 bytes
Sequence of transfer	Transferred sequentially beginning with the most significant bit.
Signed binary code	• LSB ..... Right-aligned from the right edge • MSB ..... Sign bit • When the data length is smaller than the field, the remaining space in the field is filled with MSB.
Unsigned binary code	• LSB ..... Right-aligned from the right edge • MSB ..... Not a sign bit • Unused high-order bits are filled with 0.

The following shows the range of typical 1 byte (8 bits) and 2 bytes (16 bits) integer data with and without sings.

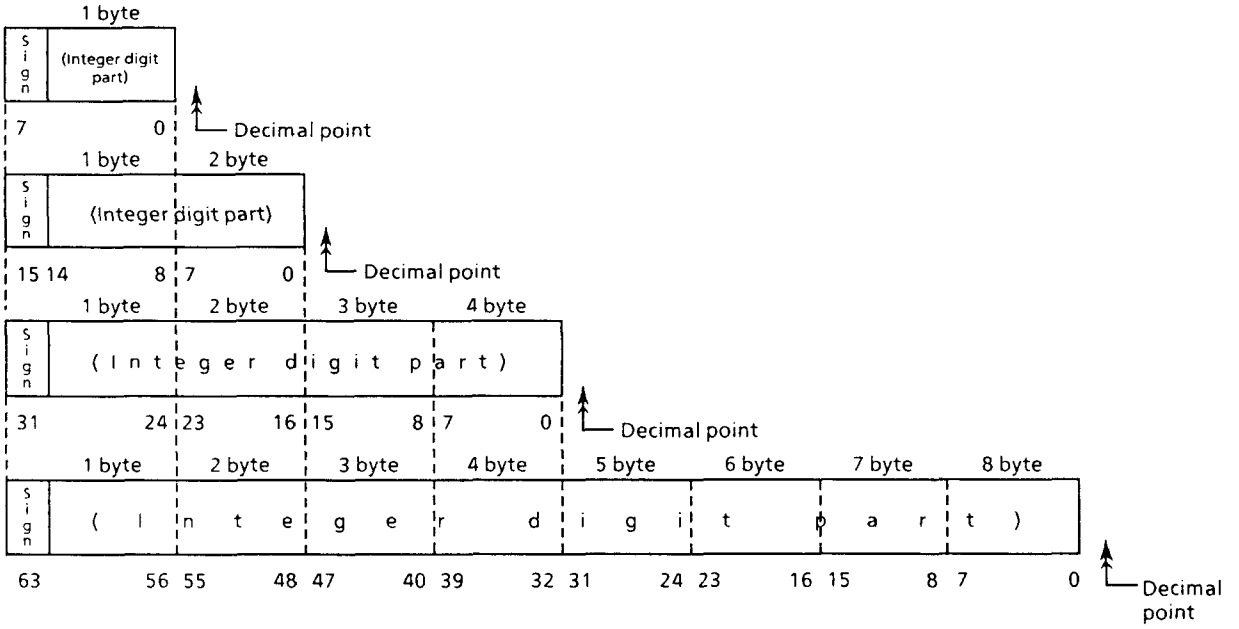
8-Bit Binary	With Sign	No Sign
10000000	-128	128
10000001	-127	129
10000010	-126	130
11111101	-3	253
11111110	-2	254
11111111	-1	255
00000000	0	0
00000001	1	1
00000010	2	2
00000011	3	3
01111101	125	125
01111110	126	126
01111111	127	127

16-Bit Binary	With Sign	No Sign
1000000000000000	-32768	32768
1000000000000001	-32767	32769
1000000000000010	-33766	32770
1111111111111101	-3	65533
1111111111111110	-2	65534
1111111111111111	-1	65535
0000000000000000	0	0
0000000000000001	1	1
0000000000000010	2	2
0000000000000011	3	3
0111111111111101	32765	32765
0111111111111110	32766	32766
0111111111111111	32767	32767

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The following shows the internal representation of signed 1, 2, 4, and 8-byte integer data. The sign bit is such that when 0, it indicates positive data; when 1, it indicates negative data.

Because the decimal point is placed at a fixed position on the right of the LSB bit, this type of representation is referred to as fixed-point binary. Because the position of the decimal point is fixed as such, when data contains numbers below the decimal point, these numbers are discarded and only the integer data is stored in the integer digit part. For unsigned data, all bits are set in the integer digit part.



(4) Handling of floating-point binary data

Floating-point binary data is used as transfer data of the <ARBITRARY BLOCK> format for both program data and response data. Note that floating-point binary data is not used in the devices manufactured by Anritsu. The following describes general specifications of this type of data for your reference.

A numeric value of this format must be comprised of the following three fields:

① Sign field (sign bit)

② Exponent field (exponent bit)

③ Mantissa field (mantissa bit)

The values handled here are the numeric data that contains fractions, with its numeric precision available as single or double precision. The following shows the field structure and transfer sequence of this type of numeric data. The symbols in the diagram below are: S = sign bit, EM = MSB exponent bit, EL = LSB exponent bit, FM = MSB mantissa bit, and FL = LSB mantissa bit.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

<table>
<tr><th>Precision</th><th>Number of transferred bytes</th><th>Field structure and transfer sequence</th></tr>
<tr><td>Single-precision</td><td>4 bytes
(32 bits)</td><td>
<table>
<tr><th>Transfer byte</th><th colspan="8">DIO line</th></tr>
<tr><td></td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td></tr>
<tr><td>1st byte</td><td>S</td><td>EM</td><td>E</td><td>E</td><td>E</td><td>E</td><td>E</td><td>E</td></tr>
<tr><td>2nd byte</td><td>EL</td><td>FM</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr>
<tr><td>3rd byte</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr>
<tr><td>4th byte</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>FL</td></tr>
</table>
① Sign bit : 1 bit
② Exponent bit: 8 bits (+127 to −126)
③ Mantissa bit: 23 bits
</td></tr>
<tr><td>Double-precision</td><td>8 bytes
(64 bits)</td><td>
<table>
<tr><th>Transfer byte</th><th colspan="8">DIO line</th></tr>
<tr><td></td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td></tr>
<tr><td>1st byte</td><td>S</td><td>EM</td><td>E</td><td>E</td><td>E</td><td>E</td><td>E</td><td>E</td></tr>
<tr><td>2nd byte</td><td>E</td><td>E</td><td>E</td><td>EL</td><td>FM</td><td>F</td><td>F</td><td>F</td></tr>
<tr><td>3rd to 7th byte</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr>
<tr><td>8th byte</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>FL</td></tr>
</table>
① Sign bit : 1 bit
② Exponent bit: 11 bits (+1023 to −1022)
③ Mantissa bit: 52 bits
</td></tr>
</table>

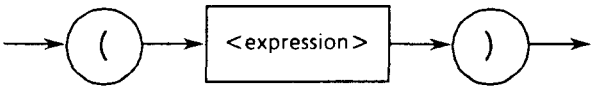
5.3.7 <EXPRESSION PROGRAM DATA>

The <EXPRESSION PROGRAM DATA> element is used to send an expression for calculating scaler, vector, matrix, or string value to the device so that the device can calculate the expression and obtain a value in place of the controller.

Note: The MW9060A does not have the function of <expression>, described below. When calculation of an expression is required, therefore, the value of that expression is obtained by the controller in advance and the calculated numeric data is transferred to the device as program data.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

<EXPRESSION PROGRAM DATA> is defined as follows:



- <expression> <expression> uses ASCII characters in the range of ASCII code bytes 20 to 7E (32 to 126 in decimal) in sequence. However, the six characters enclosed with [ ] below are not used: [", #, ', (,), ;]. That is, a double quotation, number symbol, single quotation, left-side parenthesis, right-side parenthesis, and semicolon.

When <expression> is given as a + b + c for example, the above syntax diagram becomes as follows:

(a+b+c)

These expressions can be transferred to the device using any program data format described hitherto, except the indefinite format of <ARBITRARY BLOCK PROGRAM DATA>. When <expression> is received, the device executes processing to calculate the expression and obtain its value.

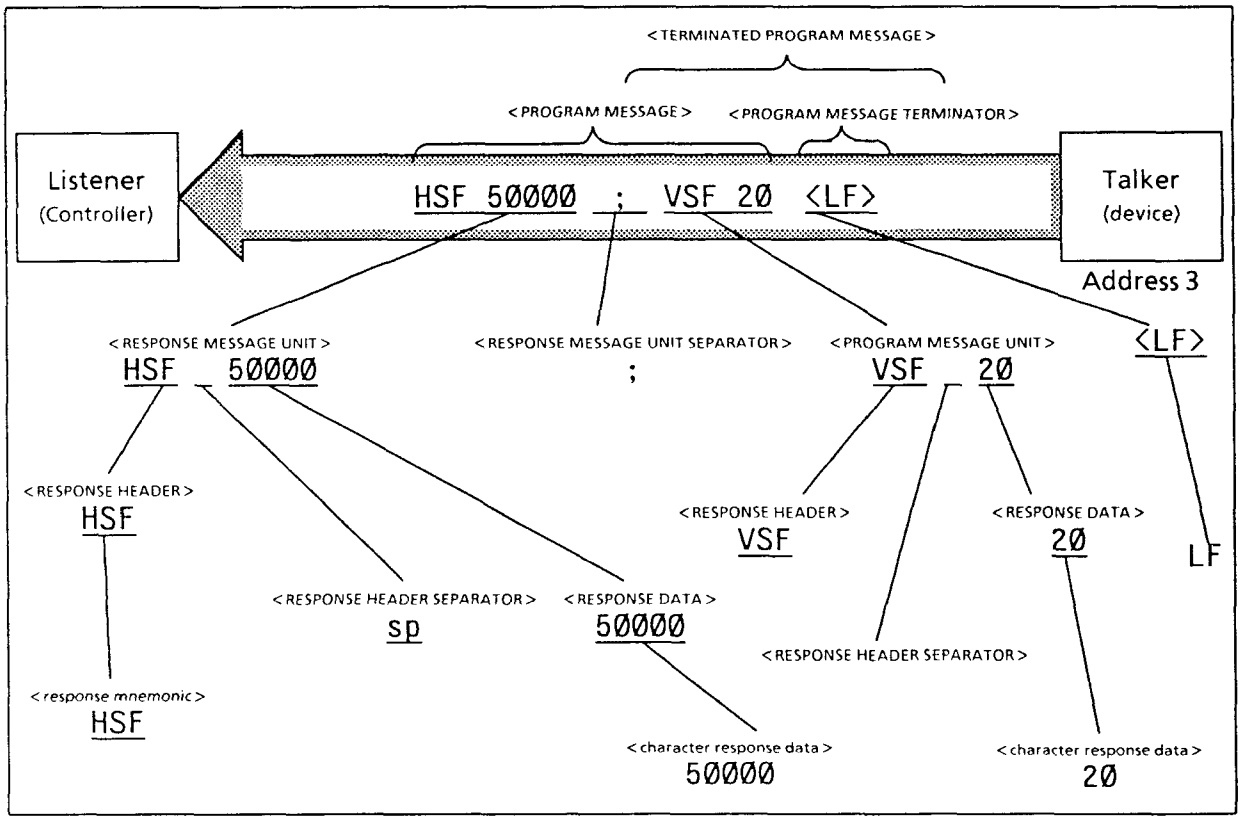
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 6
TALKER OUTPUT FORMAT

Data messages come in two types: program messages and response messages. This section describes the format of response messages sent from the talker device to the controller.

Response messages include such messages as measurement results, setting conditions, and status information. These response messages may be returned with or without the header.

The diagram below shows response messages sent from device to controller as ASCII character strings with header in return for holizontal-shift query message HSF? and vertical-shift query message VSF?



The program below shows the operation part of the above.

```
100 WRITE @101:"HSF? ; VSF?"
110 READ @101:A$!
```

← When the terminator LF is detected, response messages HSF 50000;VSF 20 are read into A$.

The response message format is configured by a sequence of functional elements (where function is divided into units of minimum level of function) as in the case of program messages. The upper-case alphabet characters enclosed with brackets < > in the above diagram are the examples of functional elements. A functional element may further be divided into what is called encoded elements. The lower-case alphabet characters enclosed with brackets < > in the above diagram

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

are the examples of encoded elements. Consequently, the syntax diagram notation is the same for both talker and listener.

The following pages describe the response message format centering on the differences from the program message format.

6.1 Differences in Format Syntax between Program and Response Messages

The greatest differences in format syntax between program and response messages are as follows:

- Program message format — Flexible program creation is intended to enable the device to easily accept program messages from the controller. Consequently, although there may be some difference in message description between program messages, these program messages can perform the same function. For example, <white space> can be inserted in separators or terminators as many as desired, it is possible to create easy-to-read programs.
- Response message format — Output messages are sent out by following the strict rules of syntax to enable the controller to easily accept response messages from the device. Therefore, contrary to the above, the response message syntax allows only one notation for one function.

The table below summarizes the differences between the program message and response message formats. Note that "zero or one or more spaces" described in the table indicates <white space>.

Characteristics	Program message syntax	Response message syntax
Program description	(Flexible)	(Strict)
Alphabetic characters	Both upper-case and lower-case characters can be used for the same effect.	Upper-case characters only.
Before and after E in NR3 exponent part	Zero or more spaces + E/e + zero or more spaces.	Upper-case character E only.
+sign in NR3 exponent part	May be omitted.	Cannot be omitted.
<white space>	Two or more spaces can be placed before and after separator as well as before the terminator.	Not used.
Message unit	① Header with program data ② Header without program data	① Data with header ② Data without header
Unit separator	Zero or more spaces + semicolon	Semicolon only.
Space placed before header	Zero or more spaces + header	Header only.
Header separator	Header + one or more spaces	Header + one #20※
Data separator	Zero or more spaces + comma + zero or more spaces	Comma only.
Terminator	Zero or more spaces + any of LF, EOI, or LF + EOI	LF + EOI

※ ASCII code byte 20 (32 in decimal = ASCII character space: sp)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

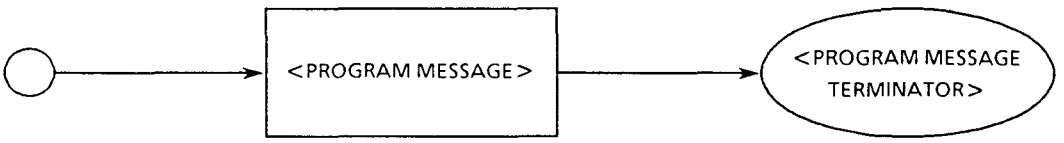
6.2 Functional Elements of Response Messages

A response message output by the talker is accepted by the controller as it is terminated by the LF∧END signal. The following describes each functional element of this response message.

The rules of syntax diagram notation are the same as for program messages. For details, refer to Section 5. Also note that the description of functional and encoded elements here is omitted wherever they are the same as those of program messages. For this part of description, refer to Section 5 as necessary.

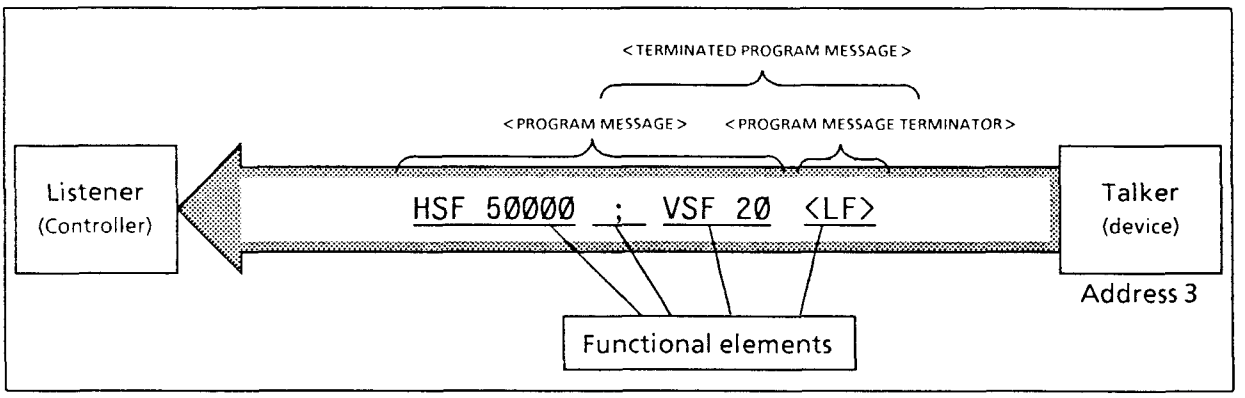
6.2.1 <TERMINATED RESPONSE MESSAGE>

<TERMINATED RESPONSE MESSAGE> is defined as follows:



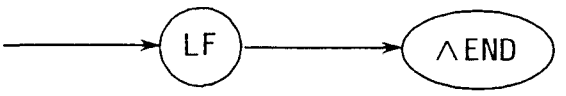
<TERMINATED RESPONSE MESSAGE> is a data message that satisfies all functional elements required for transmission from the talker device to the controller. <RESPONSE MESSAGE> is appended with <RESPONSE MESSAGE TERMINATOR> at its end to complete the transfer of <RESPONSE MESSAGE>.

Example : <TERMINATED RESPONSE MESSAGE> linking two message units



6.2.2 <RESPONSE MESSAGE TERMINATOR>

<RESPONSE MESSAGE TERMINATOR> is defined as follows:



<RESPONSE MESSAGE TERMINATOR> is placed next to the last <RESPONSE MESSAGE UNIT> to terminate the sequence of one or more <RESPONSE MESSAGE UNIT> elements of a certain length.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

LF^END is such that when the statement shown below is executed beforehand in the beginning part of the program, the EOI signal is sent out as the END signal simultaneously with terminator LF when the last data byte is transmitted.

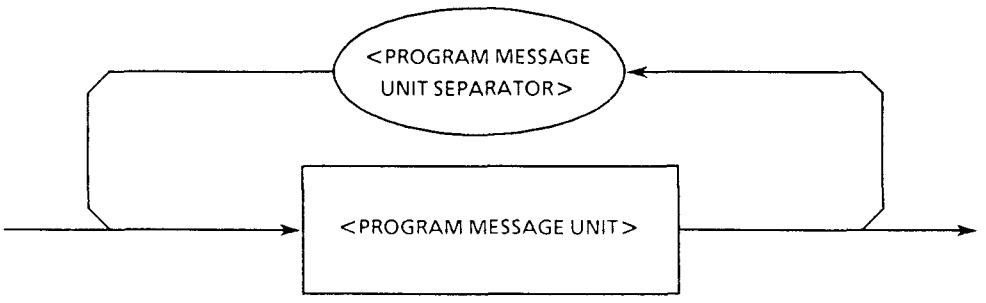
- For LF `TERM IS CHR$(10)`
- For END `EOI ON`

Example : Reading the currently set horizontal shift.

```
10 TERM IS CHR$(10)! ..... LF is assigned as terminator code.
20 EOI ON! ..... The EOI signal is output that asserts the EOI line TRUE
                 when sending the last data byte.
30 WRITE @101:"HSF?"! ..... Query for reading the horizontal shift.
40 READ @101:A$! ..... Response data read is terminated by EOI signal.
```

6.2.3 <RESPONSE MESSAGE>

<RESPONSE MESSAGE> is defined as follows:

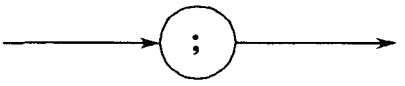


<RESPONSE MESSAGE> is a sequence of one or more <RESPONSE MESSAGE UNIT> elements.

The <RESPONSE MESSAGE UNIT > element means a single message sent from device to controller. The <RESPONSE MESSAGE UNIT SEPARATOR> element is used as a separator to divide multiple <RESPONSE MESSAGE UNITs>.

6.2.4 <RESPONSE MESSAGE UNIT SEPARATOR>

<RESPONSE MESSAGE UNIT SEPARATOR> is defined as follows:

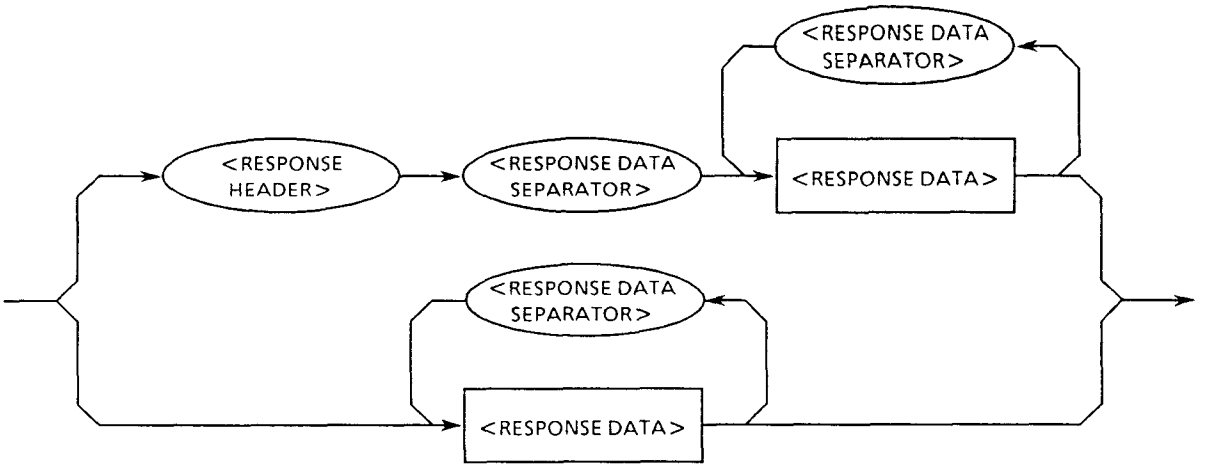


<RESPONSE MESSAGE UNIT SEPARATOR> is used when outputting a sequence of multiple <RESPONSE MESSAGE UNIT> elements as a single <RESPONSE MESSAGE> by dividing the <RESPONSE MESSAGE UNIT> elements with <UNIT SEPARATOR> semicolon ";."

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6.2.5 <RESPONSE MESSAGE UNIT>

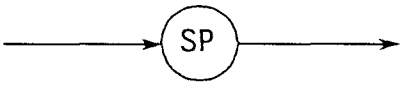
<RESPONSE MESSAGE UNIT> is defined as follows:



<RESPONSE MESSAGE UNIT> has two types of syntaxes. One is a response message unit with the header, which returns the exact results of processing done for the information set by a program message. The other is a response message unit without the header, which returns only the measurement results data.

6.2.6 <RESPONSE HEADER SEPARATOR>

<RESPONSE HEADER SEPARATOR> is defined as follows:



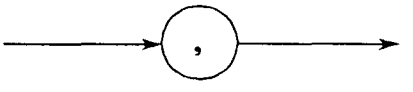
<RESPONSE HEADER SEPARATOR> separates <RESPONSE HEADER> and <RESPONSE DATA> by inserting one space immediately after <RESPONSE HEADER>.

Space SP is ASCII code byte 20 (32 in decimal).

Consequently, response messages with the header always contain one space between the header and data as a response header separator. It indicates the end of the response header as well as the beginning of response data.

6.2.7 <RESPONSE DATA SEPARATOR>

<RESPONSE DATA SEPARATOR> is defined as follows:



<RESPONSE DATA SEPARATOR> is used when outputting multiple <RESPONSE DATAs>, to divide them by inserting it between one and following data.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

6.2.8 <RESPONSE HEADER>

<RESPONSE HEADER> is the same in format representation as <COMMAND PROGRAM HEADER> described in paragraph 5.2.8 except the following three points.

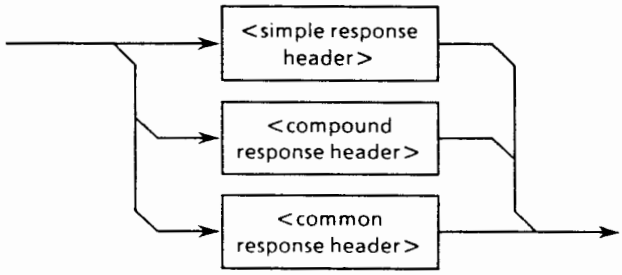
1. The characters usable in <response mnemonic> are stipulated so that for alphabetic characters, only upper-case characters can be used. Others are the same as with <program mnemonic>.
2. Although a space can be inserted before the program header, the response header cannot be preceded by a space.

The next page collectively shows <RESPONSE HEADER> from <simple response header> through <response mnemonic>.

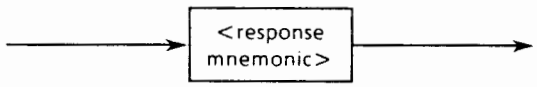
Note: For usable characters, <response mnemonic> is the same as with <program mnemonic> except that only upper-case alphabets can be used in <response mnemonic>. For others, refer to paragraph 5.2.8.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

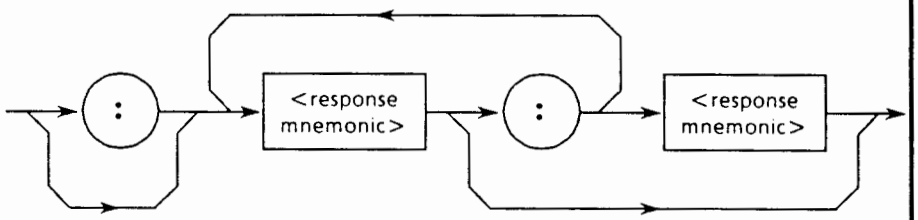
Element	Function
RESPONSE HEADER	The header represents the function of response data. Its meaning is indicated by <response mnemonic> which consists of up to 12 characters (upper-case alphabets, numbers, or underlines, or a combination of the foregoing) beginning with an upper-case alphabet.



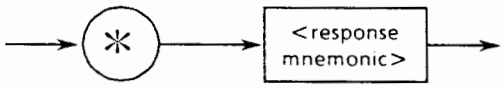
1. <simple response header> is defined as follows:



2. <compound response header> is defined as follows:

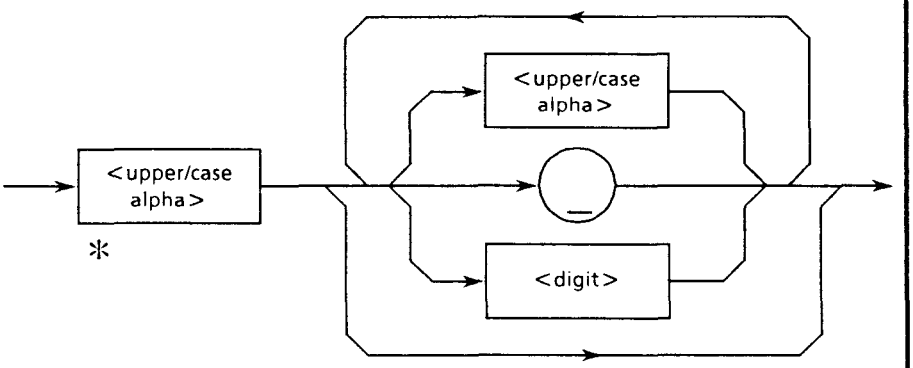


3. <common response header> is defined as follows:



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

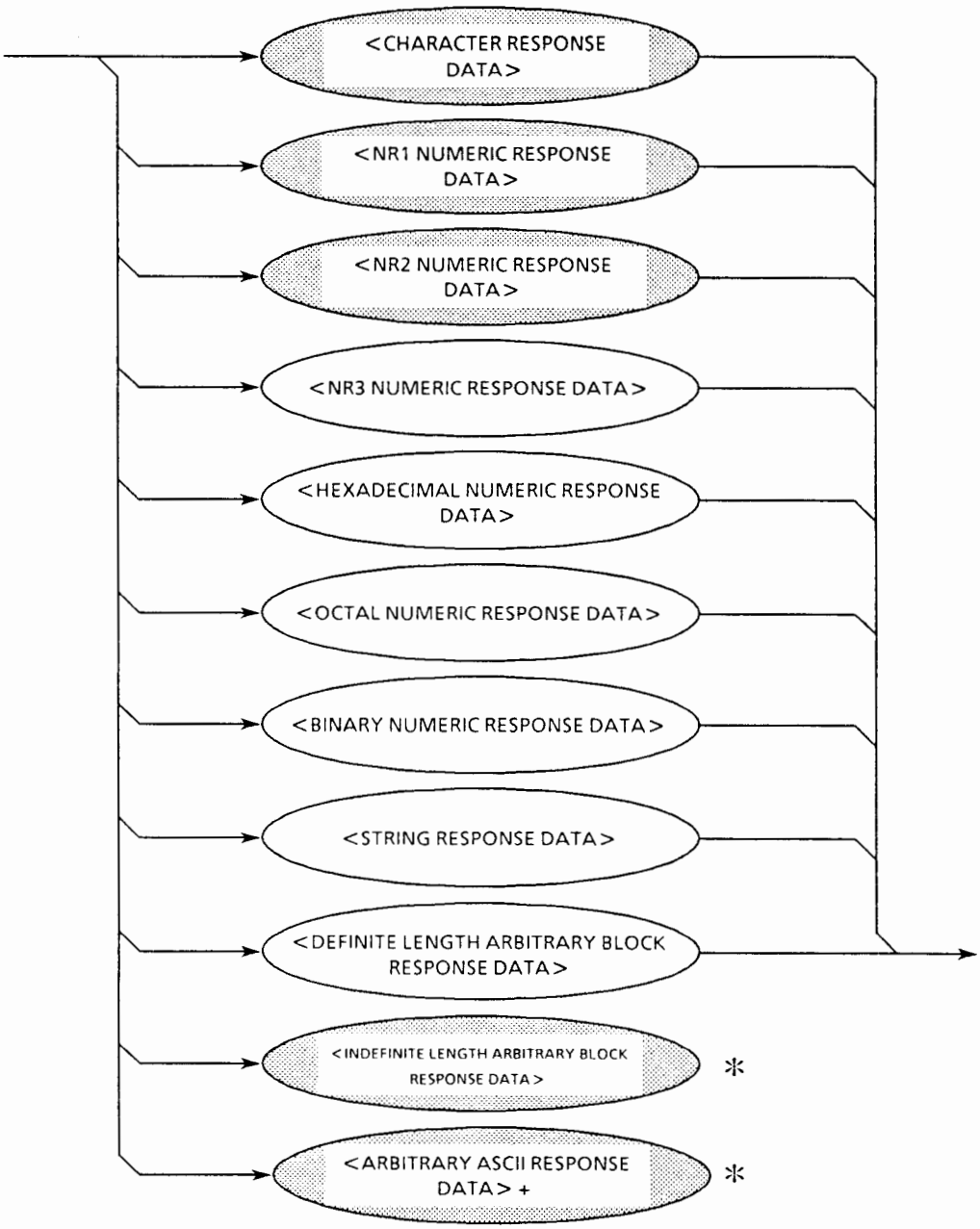
(Continued)

Element	Function
RESPONSE HEADER (Cont.)	4. <response mnemonic> is defined as follows:  * <upper-case alpha> ASCII code bytes 41 to 5A (65 to 90 in decimal = upper-case alphabets A to Z)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

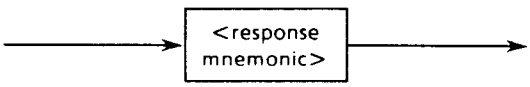
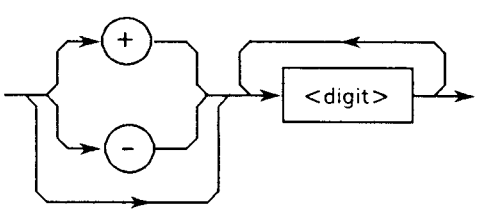
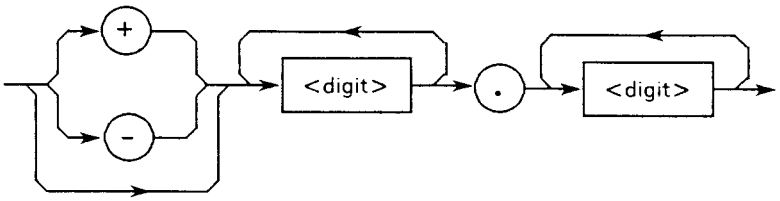
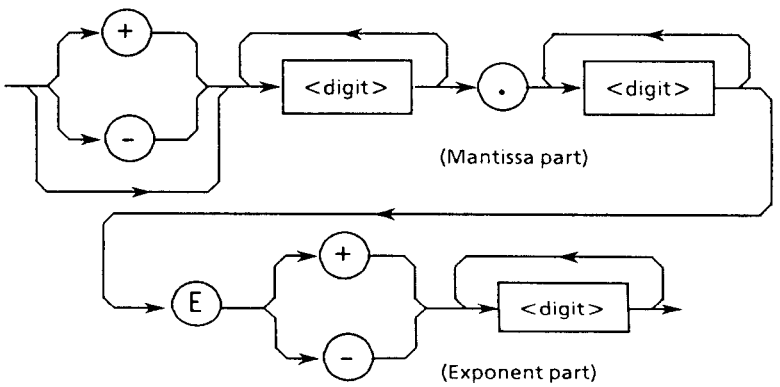
6.2.9 <RESPONSE DATA>

<RESPONSE DATA> comes in 11 types, of which the response data types enclosed in shaded ovals can be sent to the controller with the MW9060A. Which response data is returned to the controller is determined by the query message sent to the device.



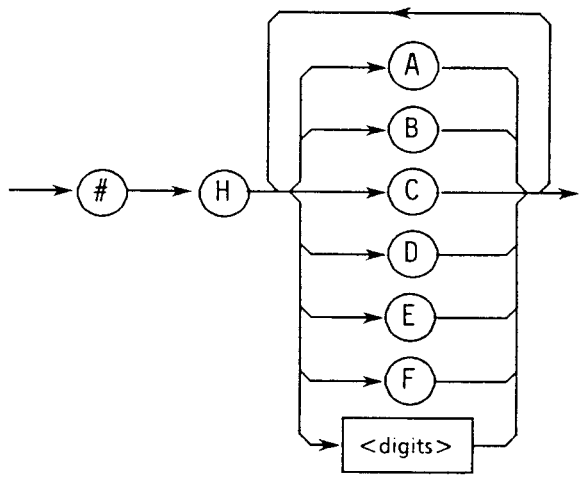
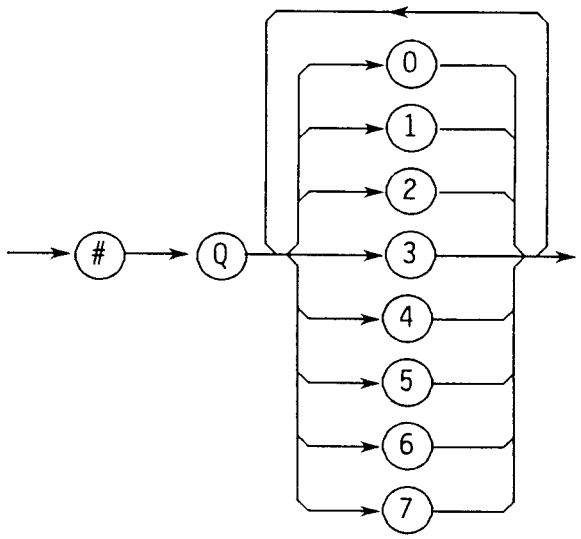
* <INDEFINITE LENGTH ARBITRARY BLOCK RESPONSE DATA> and <ARBITRARY ASCII RESPONSE DATA> are terminated by LF^END immediately after their own last data byte.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

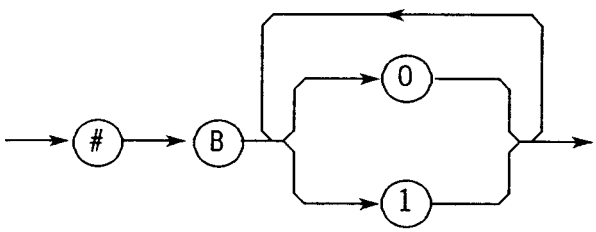
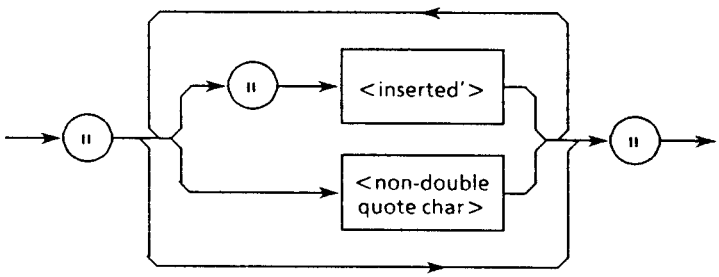
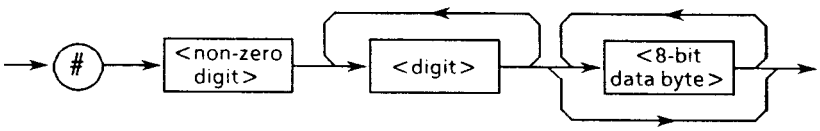
Element	Function
(1) CHARACTER RESPONSE DATA <Example> SAVE␣TRACE1	Data consisting of the same character string as in <response mnemonic>. Therefore, its character string always begins with an upper-case alphabet and is 12 characters long at maximum. Do not use numeric parameters because they are inappropriate. 
(2) NR1 NUMERIC RESPONSE DATA <Example> 123 +123 −1234	Integer data, that is, a decimal number of integer without the decimal point or exponential representation. 
(3) NR2 NUMERIC RESPONSE DATA <Example> 12.3 +12.34 −12.345	Fixed-point data, that is, a decimal number other than integer and exponential representation. 
(4) NR3 NUMERIC RESPONSE DATA <Example> 1.23E+4 +12.34E−5 −12.345E+6 • Lower-case alphabets cannot be used for E. • No spaces can be inserted before or after E. • "+" in the exponent part cannot be omitted. • "+" in the mantissa part may be omitted.	Floating-point data, that is, a decimal number with digits of exponential representation. 

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(Continued)

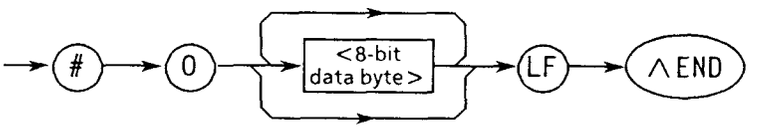
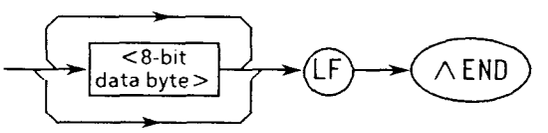
Element	Function
(5) HEXADECIMAL NUMERIC RESPONSE DATA <Example> #HABC123 #H2DC3 #H8301	Hexadecimal numeric data 
(6) OCTAL NUMERIC RESPONSE DATA <Example> #Q37 #Q26703 #Q30562	Octal numeric data 

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Element	Function
(7) BINARY NUMERIC RESPONSE DATA <Example> #B011101 #B1011 #B1011	Binary numeric data 
(8) STRING RESPONSE DATA <Example> "This is a text" "Say, ""Hello""."	All of ASCII 7-bit code can be used. Both ends of the character string are always enclosed with double quotations. Double quotations within the character string are repeated in succession with the same type of quotation on each occurrence. Because CR, LF, and space can be used, this type of data is suited for outputting text to a printer or CRT. 
(9) DEFINITE LENGTH ARBITRARY BLOCK RESPONSE DATA <Example> Transferring 11256099 in 4 bytes ↓ #1400ABC123	8-bit binary block data of fixed length. This type of data is suited for transferring large amounts of data, 8-bit extended ASCII code, or non-display data. 

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(Continued)

Element	Function
(10)INDEFINITE LENGTH ARBITRARY BLOCK RESPONSE DATA <Example> Transferring −250, −50, 120, ··· in indefinite length ↓ #0FF06FFCE0078	8-bit binary block data of indefinite length. For this reason, the first data is preceded by #0. Also, the last data is terminated by LF∧END. 
(11)ARBITRARY ASCII RESPONSE DATA <Example-1> <ASCII Byte> <ASCII Byte> LF∧END <Example-2> LF∧END	This type of data consists of ASCII data bytes, except LF, which are transmitted without dividing each byte. Therefore, the last data is terminated by LF∧END. 

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(Blank)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 7
COMMON COMMANDS

This section describes the common commands set forth in IEEE488.2.

The common commands control such basic functions as identifying and resetting the equipment, as well as how status is read and cleared and how commands and queries are received and processed by the MW9040B Optical Time Domain Reflectometer.

Each status register has a status enable register associated with it. Desired status information can be selected by masking the status bits with the appropriate pattern. The masked status bits are not used in status summary information reports.

For details on how to read status registers and how to use the status information obtained from the MW9040B Optical Time Domain Reflectometer, refer to Section 8 in this manual.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[1] ✻CLS (Clear Status)

■ Syntax

Command syntax : ✻CLS

■ Example

```
WRITE @101:✻CLS
```

■ Description

The ✻CLS command clears all event status registers, queues, and data structure including a device definition error queue and status byte. When the ✻CLS command follows immediately after <program message terminator>, the output queue and MAV (Message Available) bit are cleared.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[2] ＊ESE/＊ESE? (Event Status Enable)

■ Syntax

Command syntax : ＊ESE <register value>

Query syntax : ＊ESE?

where

<Register Value> ::=indicated by a number from 0 to 255 (integer).

■ Example

- To set bits 4 and 5 of the standard event status enable register to "1"

```
WRITE @101:"*ESE 48"
```

- To read the settings of the standard event status enable register

```
10 DIM ESE$*20
20 WRITE @101:"*ESE?"
30 READ @101:ESE$
40 PRINT ESE$
50 END
```

■ Description

The ＊ESE command sets the bits of the standard event status enable register. The standard event status enable register masks the bits of the standard event status register. When enabling the particular bit of the standard event status register, set it to "1"; when disabling the particular bit of the standard event status register, set it to "0". For details on the standard event status enable register, see the table below.

The ＊ESE? query returns the current contents of the enable register.

Standard Event Status Enable Register

Bit	Weight	Bit name	Enable conditions
7	1 2 8	PON	Power on
6	6 4	URQ	User request
5	3 2	CME	Command error
4	1 6	EXE	Execution error
3	8	DDE	Device dependent error
2	4	QYE	Query error
1	2	RQC	Bus control request
0	1	OPC	Operation completed

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ Response message format

Return format : <register value> <LF | CR,LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[3] ✻ESR? (Event Status Register)

■ Syntax

Query syntax : ✻ESR?

■ Example

∘ To read the N value of the standard event status register

```
10 DIM SESR$*20
20 WRITE @101:"*ESR?"
30 READ @101:SESR$
40 PRINT SESR$
50 END
```

■ Description

The ✻ESR? query returns the contents of the standard event status register. The standard event status register is cleared when it is read by the controller.

Standard Event Status Register

Bit	Weight	Bit name	Conditions
7	1 2 8	PON	0 = Register read – Not startup mode 1 = Startup
6	6 4	URQ	0 = User request – Not used (always 0)
5	3 2	CME	0 = No command error detected 1 = Command error detected
4	1 6	EXE	0 = No execution error detected 1 = Execution error detected
3	8	DDE	0 = No device-dependent error detected 1 = Device-dependent error detected
2	4	QYE	0 = No query error detected 1 = Query error detected
1	2	RQC	0 = Bus control – right request – Not used (always 0)
0	1	OPC	0 = Operation not terminated 1 = Operation terminated (completed)

■ Response message format

Return format : <register value> <LF ∣ CR,LF>

where

<register value> ::= Indicated by a number from 0 to 255 (integer)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[4] ＊IDN? (Identification Number)

■ Syntax

Query syntax : ＊IDN?

■ Example

◦ To read the character string for identification

```
10 DIM IDN$*50
20 WRITE @101:"*IDN?"
30 READ @101:IDN$
40 PRINT IDN$
50 END
```

■ Description

When the ＊IDN? query is issued, the character string that allows an external controller to identify the equipment is returned. This character string consists of <manufacturer's name>, <equipment name>, <manufacturing No.>, and <equipment version>. For <manufacturing No.>, the MW9060A always returns 0 because it does not have the corresponding <manufacturing No.>.

■ Response message format

Return format : ANRITSU, MW9060A, 0, <equipment version><LF | CR,LF>

where

<equipment version> ::= A 4-digit integer indicating the current equipment version.

Item	Character string	Number of characters
Manufacturer's name	ANRITSU	7 characters
Equipment name	MW9060A	7 characters
Manufacturing No.	0	1 characters
Equipment version	0001	4 characters

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[5] ＊OPC/＊OPC? (Operation Complete)

■ Syntax

Command syntax : ＊OPC

Query syntax : ＊OPC?

■ Example

- To specify that the operation complete bit of the standard event status register be set when pending processing of an overlap command is completed

```
WRITE @101:"*OPC"
```

- To wait until pending processing of an overlap command is completed

```
10 DIM OPC$*20
20 WRITE @101:"*OPC?"
30 READ @101:OPC$
40 PRINT OPC$
50 END
```

■ Description

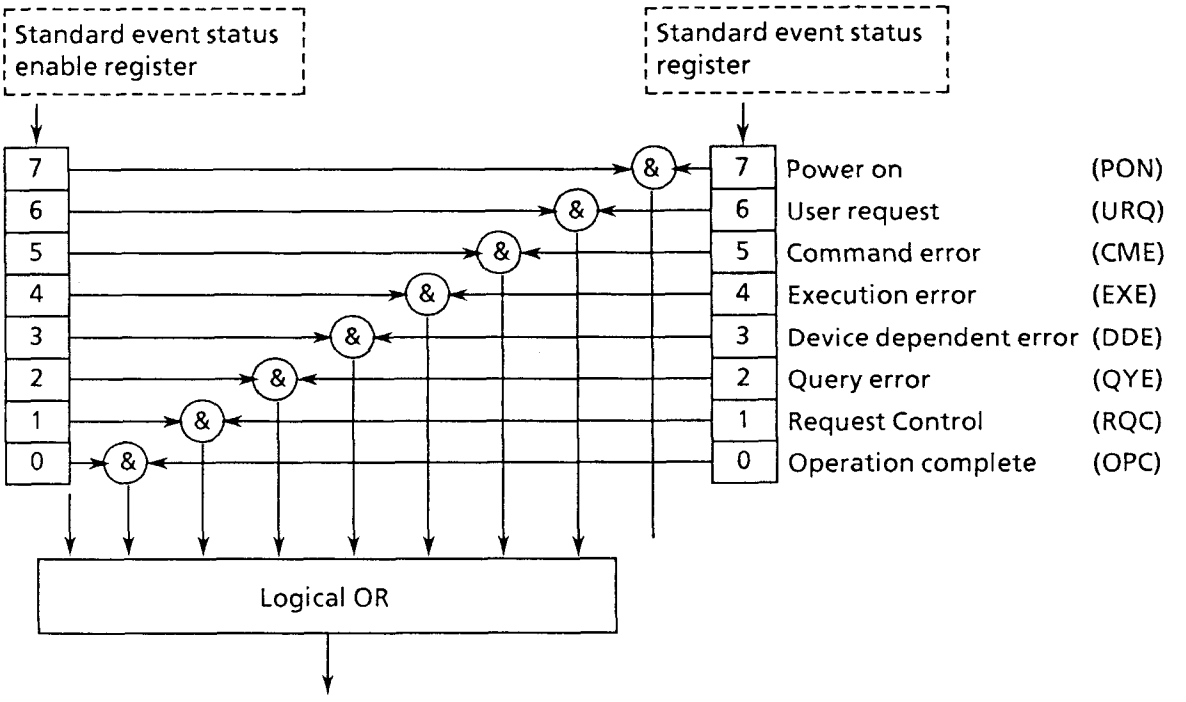
＊OPC and ＊OPC? do not have any effect with the MW9060A.

The ＊OPC command sets the operation complete bit of the standard event status register when all processing is terminated for a command that has been accepted but remained incomplete of processing.

Such a processing-incomplete state occurs for overlap commands. An overlap command means a command that executes the next command even when device operation initiated by that command is in progress. The MW9060A does not have such overlap commands.

The ＊OPC? query returns ASCII character "1" when all pending processing in device operation is completed.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com



■ Response message format

Return format : 1<LF | CR,LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[6] ＊RST (Reset)

■ Syntax

Command syntax : ＊RST

■ Example

° To initialize

```
WRITE @101:"*RST"
```

■ Description

The *RST command resets various settings of the MW9060A to their initial states. This command initializes many more settings than in the case of initialization described earlier in Section 4.

The following lists the items initialized by this command.

① The device's inherent functions or states are set to certain known states regardless of how they have been set until now. The contents of settings are shown in the tables in the next pages.

② The macro defined by the *DDT command is set to a state defined for the device.

③ Macro operations are disabled and the MW9060A is placed in the macro command reject mode. Also, the macro definition is returned to the state indicated by the equipment designer.

④ The device is placed in the OCIS state (Operation Complete Command Idle State). As a result, the operation complete bit cannot be set to the SESR (Standard Event Status Register).

⑤ The device is placed in the OQIS state (Operation Complete Query Idle State). As a result, operation complete bit 1 cannot be set to the output queue.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Initial Values Set by ✳RST Command

No.	Item	Initial value
1	Soft key hierarchy	First hierarchical layer L1: (1/2) (This state is recovered when returned to the Local state.)
2	Laser	Off
3	Hold	Off
4	Average	Off
5	Horizontal axis scale	Initial value of plug-in unit data
6	Vertical axis scale	5 dB/div
7	Horizontal axis shift	0 km
8	Vertical axis shift	10 dB
9	Function	LOSS function
10	AUTO	Off
11	Marker Selected MARKER ✳ MARKER X1 MARKER X2 MARKER X3 MARKER X4 MARKER	 Marker Marker initial state Marker initial state Marker initial state Marker initial state Marker initial state
12	Distance range	Initial value of plug-in unit data
13	Pulse width	Initial value of plug-in unit data
14	Attenuator	Initial value of plug-in unit data
15	Auto attenuator	On
16	Selected mask No.	No. 1
	Mask No. 1 distance	0 km
	Mask No. 2 distance	0 km
	Mask No. 3 distance	0 km
	Mask No. 4 distance	0 km
	Mask No. 5 distance	0 km
17	Near-end mask width	NEAR END mode:Off

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Initial Values Set by ✳RST Command (Continued)

No.	Item	Initial value
18	Setting value of near-end mask width	NEAR END mode:Off
19	λ-SELECT	Initial value of plug-in unit data for switchable plug-in unit; otherwise, not initialized.
20	Linear approximation	2PA
21	Average limit count/time	Count
	Time	12 hours (43,200 seconds)
	Count	50,000 times
22	Rotary knob	Marker shift mode
23	Threshold	1.0 dB
24	WAVE COMPARE	Off
25	D/A converter for LD current (power)	0 for D/A value
26	Panel lamp	COARSE: Off
27	Smoothing	Off
28	Smoothing level	Level 1
29	Full Trace	Off

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[7] ✳SRE/✳SRE? (Service Request Enable)

■ Syntax

Command syntax : ✳SRE␣<register value>

Query syntax : ✳SRE?

where

<Register Value>::=Indicated by an integer number from 0 to 255

■ Example

- To set bits 4 and 5 of the service request enable register to "1"

```
WRITE @101:"*SRE 48"
```

- To read the settings of the standard event status enable register

```
10 DIM SRE$*20
20 WRITE @101:"*SRE?"
30 READ @101:SRE$
40 PRINT SRE$
50 END
```

■ Description

The ✳SRE command sets the designated bits of the service request enable register. The service request enable register, in turn, masks the designated bits of the status byte register. To enable the service request register, set the corresponding bit to "1"; to disable the service request register, set the corresponding bit to "0". For details on the service request enable register, see the table below.

The ✳SRE? query returns the current value of the enable register.

MW9060A Service Request Enable Register

Bit	Weight	Bit name	Enable conditions
7	128		Not used
6	64	MSS	Master summary status
5	32	ESB	Event status
4	16	MAV	Message available
3	8	ESB	Error summary
2	4	TSB	Termination (Completed) summary
1	2		Not used
0	1		Not used

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ **Response message format**

Return format : <register value> <LF | CR,LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[8] ＊STB? (Read Status Byte)

■ Syntax

Query syntax : ＊STB?

■ Example

◦ To read the value of the status byte

```
10 DIM STB$*20
20 WRITE @101:"*STB?"
30 READ @101:STB$
40 PRINT STB$
50 END
```

■ Description

The ＊STB? query returns the current value of the status byte. The MSS (master summary status) bit and RQS (request service) bit are reported to bit 6. MSS indicates whether the device has at least one cause of request for service.

MW9060A Status Byte Register

Bit	Weight	Bit name	Conditions
7	1 2 8		0 Not used
6	6 4	MSS	0 = Service not requested 1 = Service being requested
5	3 2	ESB	0 = No event status occurs 1 = Enabled event status occurred
4	1 6	MAV	0 = Output message not prepared 1 = Output message prepared
3	8	ERSB	0 = MW9060A-specific error not detected 1 = MW9060A-specific error detected
2	4	TSB	0 = No termination report 1 = Termination report present
1	2		0 Not used
0	1		0 Not used

■ Response message format

Return format : <register value> <LF ! CR,LF>

where

<Register Value> ::= Indicated by an integer number from 0 to 255

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[9] ＊TST? (Self-Test)

■ Syntax

Query syntax : ＊TST?

■ Example

○ To read the results of self-test

```
10 DIM TEST$*20
20 WRITE @101:"*TST?"
30 READ @101:TEST$
40 PRINT TEST$
50 END
```

■ Description

The ＊TST? query is not supported for the MW9060A. Therefore, it always returns "0".

■ Response message format

Return format : <result> <LF | CR,LF>

where

<result> ::= Always value 0 (integer) is returned.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[10] ✳TRG? (Trigger)

■ Syntax

Command Syntax : ✳TRG

■ Example

- To use the *TRG command to start measurement

```
WRITE @101:"*TRG"
```

■ Description

The ✳TRG command has the same function as a bus command, Group Execution Trigger (GET). For this command, the subject of execution varies depending on the availability of the *DDT command. The MW9060A does not support the *DDT command. Therefore, when the *TRG command is received, the MW9060A operates as follows:

- Laser ON
- Average ON

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[11] ＊WAI (Wait to Complete)

■ Syntax

Command Syntax : ＊WAI

■ Example

◦ To keep the device waiting until the operation of an overlap command is completed

```
WRITE @1Ø1:"*WAI"
```

■ Description

The ＊WAI command does not have any effect with the MW9060A. Although this command is accepted by the MW9060A, nothing is affected by it.

The ＊WAI command keeps the device waiting until the overlap command completes execution of all operations before executing the next command or query. An overlap command means a command that executes the next command even when device operation initiated by that command is in progress. The MW9060A does not have such overlap commands.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(Blank)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 8
STATUS BYTE AND SYNCHRONIZATION TECHNIQUE

8.1 Standard Status Model of IEEE488.2

The diagram below shows the standard model of the status data structure stipulated in IEEE 488.2.

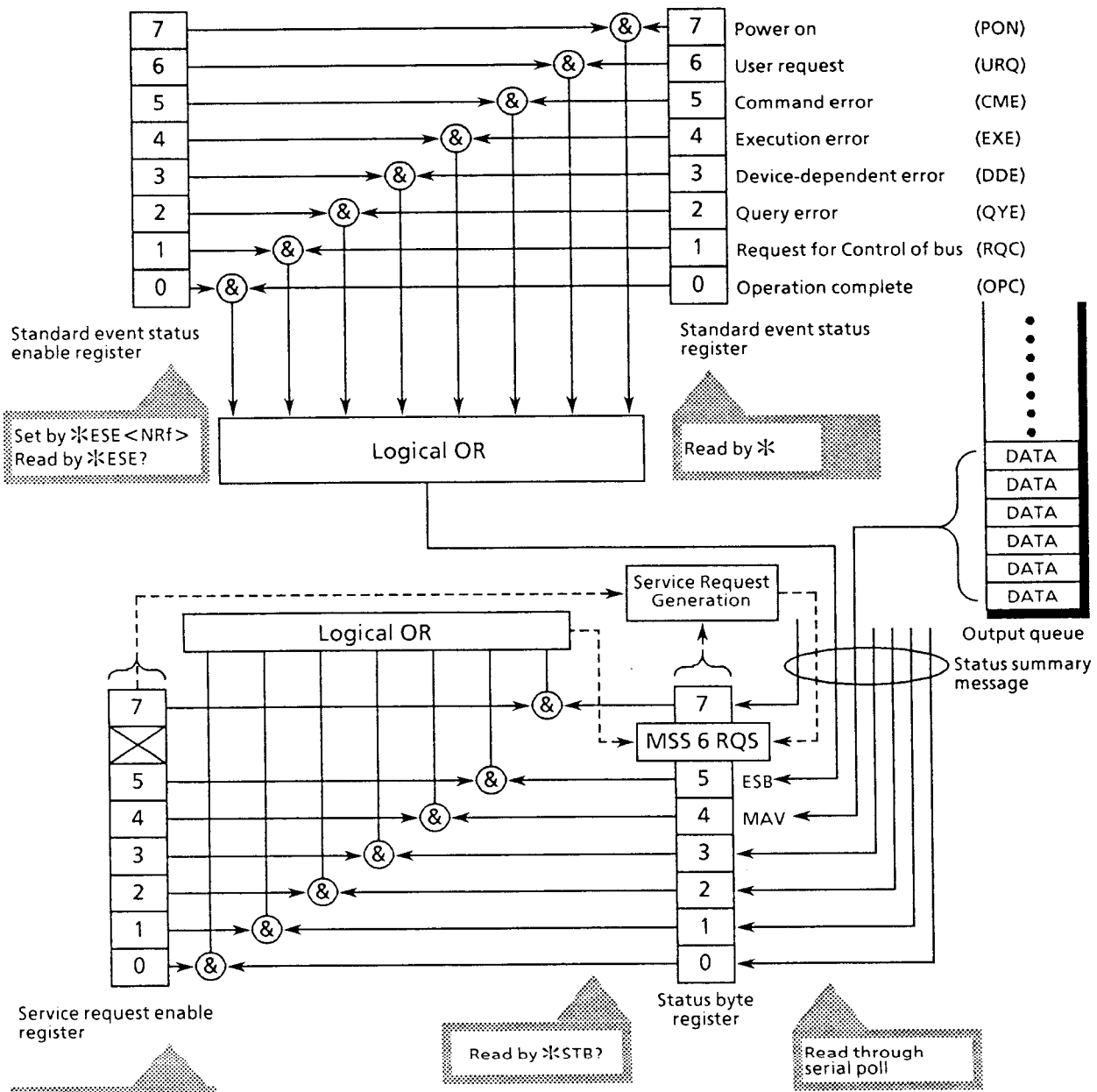


Diagram of Standard Status Model

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The status model uses the IEEE 488.1 status byte. This status byte consists of seven summary message bits supplied from status data structure. To generate these summary message bits, the status data structure is configured by two models: a register model and a queue model.

Register model	Queue model
A register model is comprised of one pair of registers to record events encountered by the device and its conditions. Specifically, this pair consists of an event status register and event status enable register. When the AND of the two is not 0, the corresponding status bit is set to 1; otherwise, the bit is 0. Then, when the logical OR of each of these bits results in 1, the corresponding summary message bit is set to 1; otherwise, the summary message bit is 0.	Called a queue model, this queue sequentially records waiting status value or information. The queue structure is such that only when there is data in the queue, the corresponding bit is set to 1; when the queue is empty, the bit is 0.

Based on the register and queue models described above, the standard model of the IEEE 488.2 status data structure consists of two register models and one queue model.

① Standard event status register and standard event status enable register

② Status byte register and service request enable register

③ Output queue

Standard Event Status Register	Status Byte Register	Output Queue
This register has the structure of the register model described above. Among the events encountered by the device, it covers eight events ①Power on, ②User request, ③Command error, ④Execution error, ⑤Device-dependent error, ⑥Query error, ⑦Request for control of bus, and ⑧Operation complete, and sets these event bits in the standard event status register. The logical OR output bit is summarized and represented by bit 5 (DI06) of the status byte register as Event Status Bit (ESB) summary message.	The status byte register allows the RQS bit and the seven summary message bits from the status data structure to be set. It is used in combination with the service request enable register, and when OR of the two is not 0, SRQ is turned on. To indicate this, bit 6 (DIO7) of the status byte register is system-reserved as the RQS bit. This bit indicates to external controller that a service is requested. This mechanism of SRQ conforms to the IEEE 488.1 standard.	This queue has the structure of the queue model described above. When data is present in the output buffer, this is summarized and indicated by bit 4 (DIO5) of the status byte register as a Message Available (MAV) summary message.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.2 Status Byte (STB) Register

The STB register consists of the device's STB and RQS (or MSS) messages. Under IEEE 488.1, although the method to report the STB and RQS messages is defined, the protocol to set and clear the message bits and the meaning of STB are not defined. IEEE 488.2 defines the device's status summary messages and the Master Summary Status (MSS) that is sent to bit 6 along with STB in response to the *STB? common query.

8.2.1 ESB and MAV Summary Messages

The following describes the ESB and MAV summary messages.

(1) ESB summary message

The ESB (Event Summary Bit) is a message defined under IEEE 488.2 and represented by bit 5 of the STB register. The state of this bit indicates whether at least one or more event defined under IEEE 488.2 have occurred while the service request enable register is set to allow occurrence of events after the standard event status register is last read or cleared. The ESB summary message bit is turned TRUE when any one of the events entered in the standard event status register is set TRUE while settings are made to allow occurrence of events. Conversely, the ESB summary bit is turned FALSE where none of the entered events occurs even while settings are made to allow occurrence of events.

(2) MAV summary message

The MAV (Message Available) summary bit is a message defined under IEEE 488.2 and represented by bit 4 of the STB register. The state of this bit indicates whether the output queue is empty. The MAV summary message bit is set to 1 (TRUE) when the device is ready to accept response-message send-requests from the controller, or reset to 0 (FALSE) when the output queue is empty. This message is used to synchronize information exchange with the controller. For example, it may be used to have the controller wait until MAV becomes TRUE after transmitting a query command to the device. While waiting for a response from the device, the controller can process other jobs. On the other hand, if the controller begins reading the output queue without checking MAV first, all system bus operations must be kept waiting until the device responds.

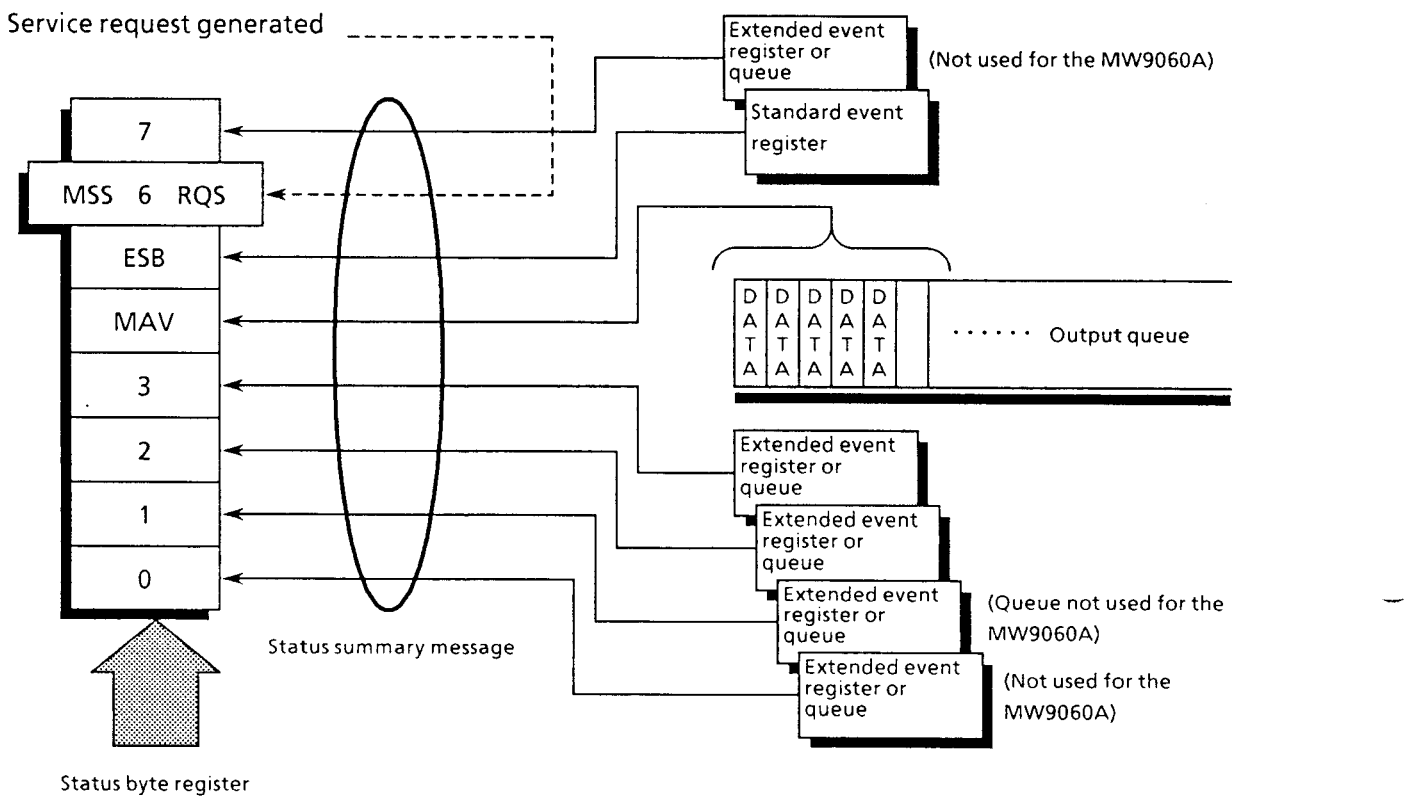
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.2.2 Device Specific Summary Messages

IEEE 488.2 does not specify whether bit 7 (DIO8), bit 3 (DIO4) to bit 0 (DIO1) of the status byte register may be used as status register summary bits or as the bits to inform of the presence of data in queue. These bits can be used as device-specific summary messages.

The device-specific summary messages have the respective status data structures of the register and queue models. That is to say, these status data structures may be a pair of registers to report events and status in parallel or one queue to sequentially report status and information. The summary bit indicates the summarized current state of the corresponding status data structure. In the case of a register model, the summary message becomes TRUE when there is an event that is set to enable one or more occurrences of TRUE; in the case of a queue model, it becomes TRUE when the queue is not empty.

Because the MW9060A uses two bits (bits 2 and 3) as status register summary bits while leaving bits 7, 1, and 0 unused as shown below, there is a total of four kinds of register models (two kinds of extended) and one kind of queue model, i.e., an output queue, without extension.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.2.3 Reading and Clearing STB Register

The contents of the STB register are read through serial poll or using the *STB? Common query. In either case, the STB message of IEEE 488.1 is read, but the value sent to bit 6 varies with the method of read.

The contents of the STB register can be cleared by the *CLS common command.

(1) Reading through serial poll

When serially polled under IEEE 488.1, the device must return a 7-bit status byte and the RQS message bit under IEEE 488.1. According to IEEE 488. 1, the RQS message indicates whether the device is sending SRQ TRUE. The value of the status byte is not changed by serial poll. Immediately after polling, the device must set the rsv message FALSE. Therefore, the RQS message is FALSE when the device is polled again before a new cause of service request occurs.

(2) Reading with *STB? common query

The *STB? common query demands the device to send the contents of the STB register and one <NR1 NUMERIC RESPONSE DATA> from the MSS (Master Summary Status) summary message. The response value represents the sum of the binary-weighted value of the STB register and the value of the MSS summary message. Bits 0 to 5 and 7 of the STB register are respectively weighted to 1, 2, 4, 8, 16, 32 and 128, and the MSS bit is weighted to 64. Thus, the response to *STB? is the same as that for serial poll except that the bit 6 represents the MSS summary message in place of the RQS message.

(3) Definition of MSS (Master Summary Status)

Indicates that the device has at least one cause of service request. In the device's response to the *STB? query, the MSS message is indicated by bit 6, but not generated as a response to serial poll. It is not a part of the status byte stipulated by IEEE 488.1, either. MSS is configured as an overall OR of a combination of STB register and SRQ enable (SRE) register bits. Concretely, MSS can be defined as follows:

(STB Register bit 0 AND SRE Register bit 0)
OR
(STB Register bit 1 AND SRE Register bit 1)
OR
:
:
(STB Register bit 5 AND SRE Register bit 5)
OR
(STB Register bit 7 AND SRE Register bit 7)

In MSS definition, the states of bit 6 in both STB and SRQ enable registers are ignored. Therefore, when calculating the MSS value, the status byte can be handled as an 8-bit value where bit 6 is always 0.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

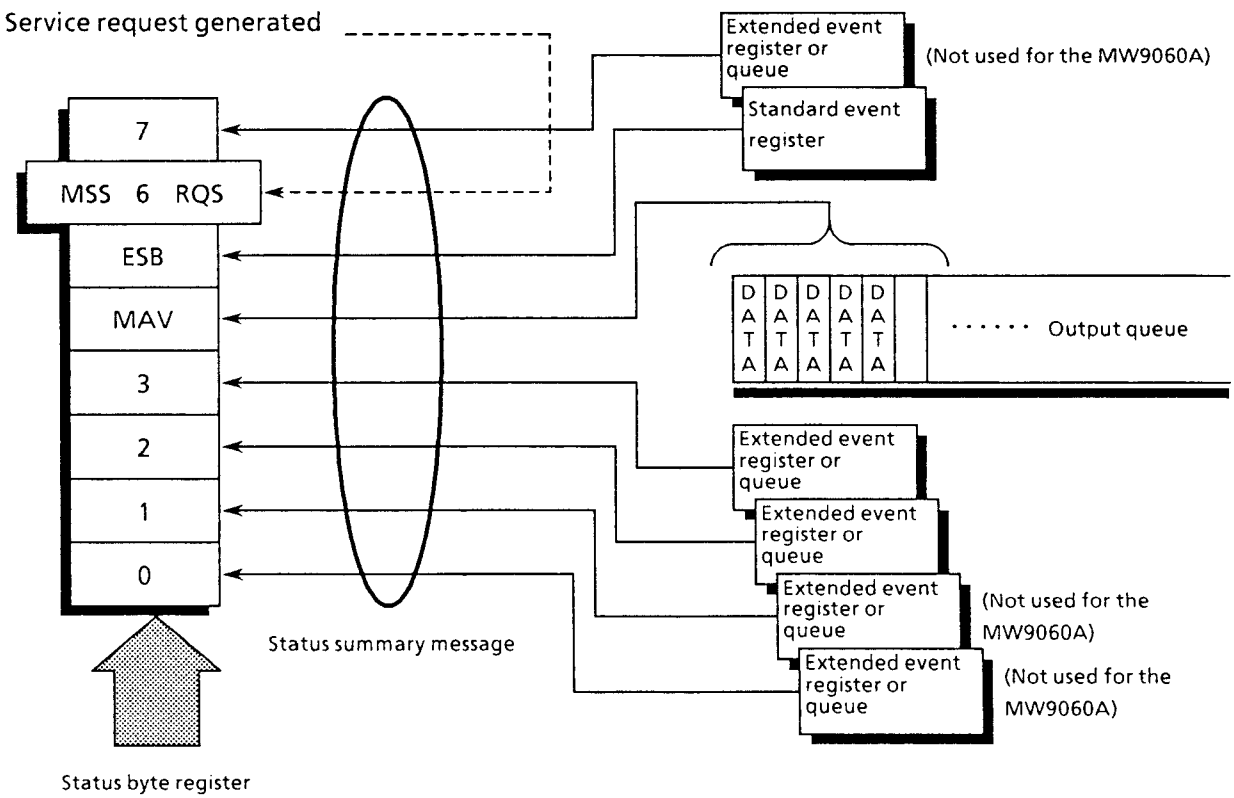
(4) Clearing STB register with ✳CLS common command

The ✳CLS common command clears all status data structures (that is, status event registers and queues) except output queues and their MAV summary messages. Accordingly, it also clears the summary messages corresponding to each of these registers and queues.

Output queues and their MAV summary messages are also cleared in the following case:

```
10 WRITE @103 : "IOR 1.5058 ; HSF 1000"
20 WRITE @103 : "*CLS ; IOR?"
30 END
```

In other words, when the ✳CLS common command is sent after <PROGRAM MESSAGE TERMINATOR> or before <QUERY MESSAGE UNIT> element, all status bytes are cleared. In this way, all pending messages in an output queue (messages not read) are cleared and the MAV message is set FALSE. When responding to ✳STB?, the MSS message is also set FALSE. Note that the settings of each enable register are unaffected by ✳CLS.

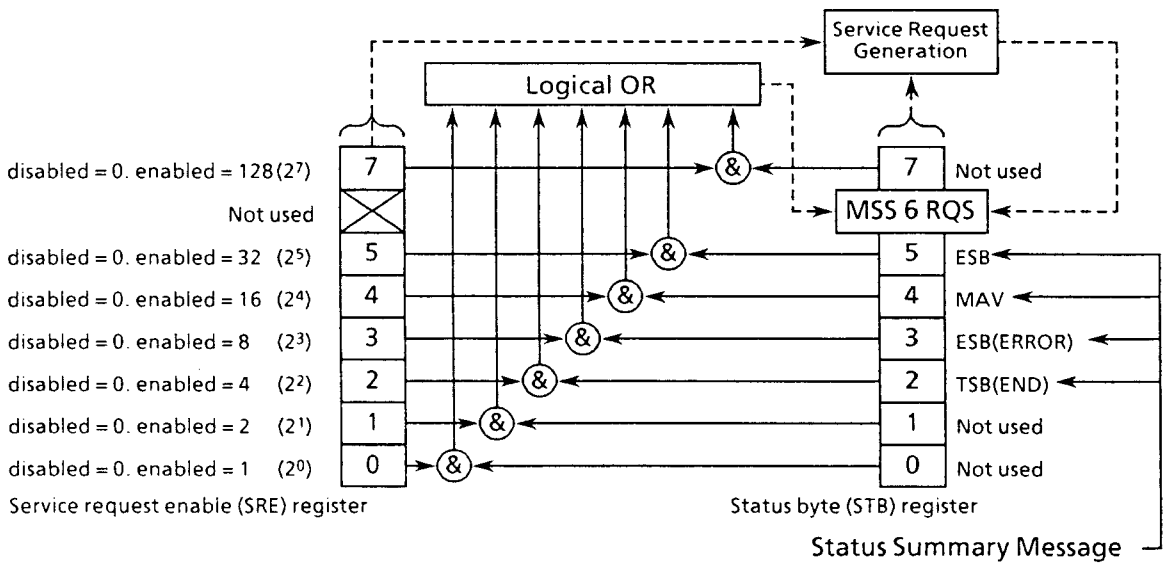


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.3 Enabling SRQ

Enabling SRQ makes it possible to select which summary message in the STB register to be enabled or disabled for service requests. The service request enable (SRE) register shown below is used to select such summary messages.

The bits in the service request enable register correspond to the bits in the status byte register one to one. When a bit in the status byte register corresponding to the enabled bit of the service request enable register is set to 1, the device sets the RQS bit to 1 to request service to the controller. For example, if bit 4 of the service request enable register is enabled, service will be requested to the controller each time the MAV bit is set to 1 when there is data in the output queue.



(1) Reading SRE register

The contents of the SRE register can be read using the *SRE? common query. The response message to this query is <NR1 NUMERIC RESPONSE DATA> = integer of 0 to 255, representing the sum of the weighted values of each service request enable register bit digit value. Bits 0 to 5 and 7 of the service request enable register are respectively weighted to 1, 2, 4, 8, 16, 32, and 128. The unused bit 6 must always be 0.

(2) Updating SRE register

The SRE register is written using the *SRE common command. The *SRE common command is followed by a <DECIMAL NUMERIC PROGRAM DATA> element. <DECIMAL NUMERIC PROGRAM DATA> is rounded to the nearest integer number, expressed in binary with a base of 2, and represents the sum of the weighted values of each SRE register bit digit value. This bit value indicates an enabled state when it is 1 or a disabled state when 0. The value of bit 6 must always be ignored.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(3) Clearing SRE register

The SRE register can be cleared using the *SRE common command or turning the power on again.

When using the *SRE common command, the SRE register can be cleared by resetting the value of <DECIMAL NUMERIC PROGRAM DATA> element to 0. When the register is cleared, the rsv local message can be inhibited from being generated by status information, so that service requests will no longer be generated.

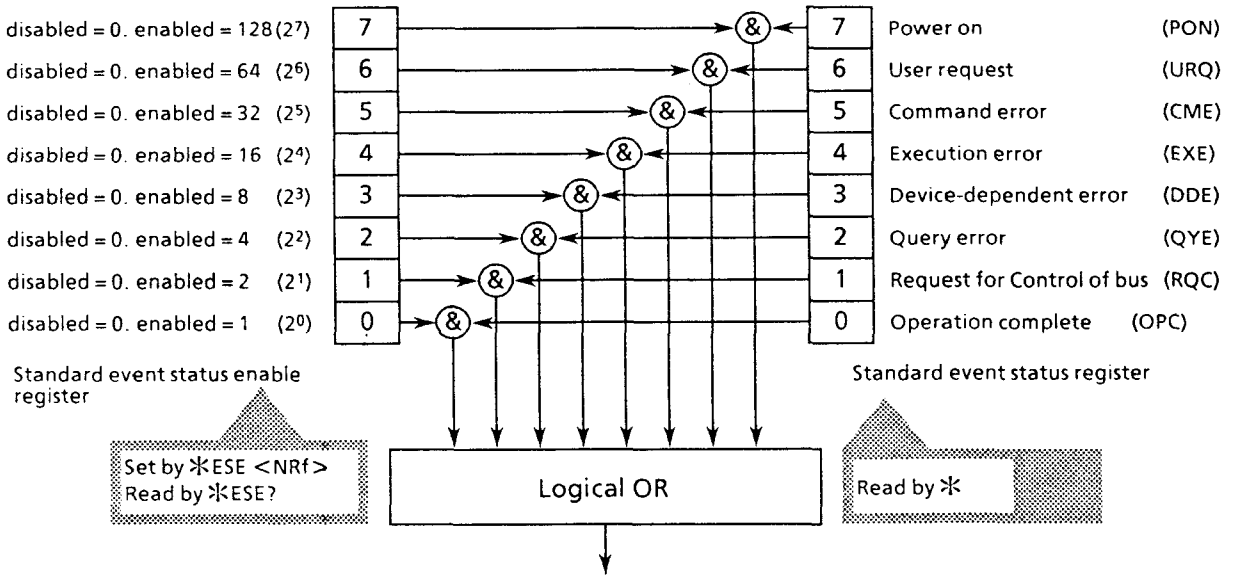
When powering on again, if the power-on status clear flag is TRUE and clearing is not blocked because of no *PSC command, the SRE register is cleared when the power is turned on again. The *PSC command is used to set the power-on-status clear flag.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.4 Standard Event Status Register

8.4.1 Bit Definition of Standard Event Status Register

The standard event status register must be included in all devices as long as they are IEEE 488.2 compatible. The diagram below shows the operation of the standard event status register model. The register model operation itself is the same as described hitherto, so the description here is made of the IEEE 488. 2 definition about the meaning of each bit in the standard event status register.



Bit	Event name	Description
7	PON (Power on)	The power is turned on.
6	URQ (User Request)	Local control (RTL) is requested. This bit is generated regardless of whether the device is in remote or local mode. This is always 0 for the MW9060A because not used.
5	CME (Command Error)	An illegal program message, misspelled command, or GET command in program message is received.
4	EXE (Execution Error)	A legal but unexecutable program message is received.
3	DDE (Device-dependent Error)	An error occurred due to causes other than CME, EXE, or QYE.
2	QYE (Query Error)	An attempt is made to read data from the output queue while no data is present in the output queue. Or, data in the output queue is lost for overflow or some other reason.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(Continued)

Bit	Event name	Description
1	RQC (Request Control)	Requesting to become the active controller itself. This bit is always 0 for the MW9060A because not used.
0	OPC (Operation Compete)	The device has completed the specified operation that was pending and is ready to accept new commands. This bit responds only to the *OPC command, and sets the operation complete bit.

8.4.2 Details of Query Errors

No.	Item	Description
1	Incomplete program message	If the device receives MTA from the controller before receiving the program message terminator during program message reception, the device abandons the incomplete program message input to it up to that time and waits for the next program message. When abandoning an incomplete program message, the device clears the input/output buffers, tells a query error to the status report part, and sets bit 2 of the standard event status register for query error.
2	Interruption of response message output	If the device receives MLA from the controller before finishing transferring the response message terminator during response message transmission, the device automatically interrupts response message output and waits for the next program message. When interrupting response message output, the device clears the output buffers, tells a query error to the status report part, and sets bit 2 of the standard event status register for query error.
3	When next program message is sent without reading response message	If the device could not output a response message because the controller transmitted the next program message immediately after sending a program message containing a query message, the device abandons the response message and waits for the next program message. It tells a query error to the status report part as in 2 above.
4	Output queue overflow	When a program message containing a number of query messages is executed, response messages may be generated beyond the capacity of the output queue (256 bytes). If still another query message is input and a response message for it must be output when the output queue is already full, the output queue may overflow. When the output queue overflows, the device clears the output queue and resets the response message generating part. Also, it sets bit 2 of the standard event status register for query error in the status report part.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.4.3 Reading, Writing, and Clearing Standard Event Status Register

Read	The register is destructively read by the *ESR? common query. That is, the bits in the register are cleared after read. The response message is <NR1> obtained by converting a binary-weighted value of event bit into a decimal number.
Write	The register cannot be written from external devices except clearing it.
Clear	The register is only cleared in the following cases : ① When the *CLS command is received. ② When the power is turned on while the power-on-status clear flag is TRUE. While executing the power-on sequence, the device first clears the standard event status register and then records events occurring during this sequence (e.g., setting the PON event bit). ③ When an event is read for the *ESR? query command.

8.4.4 Reading, Writing, and Clearing Standard Event Status Enable Register

Read	The register is non-destructively read by the *ESE? common query. That is, the bits in the register are not cleared after read. The response message is returned with <NR1> obtained by converting a binary-weighted value into a decimal number.
Write	The register is written by the *ESE common command. Because bits 0 to 7 of the register are respectively weighted to 1, 2, 4, 8, 16, 32, 64, and 128, the written data is sent by <DECIMAL NUMERIC PROGRAM DATA>, the sum of bit digit values selected from the bits.
Clear	The register is only cleared in the following cases : ① When the *ESE common command with data value of 0 is received. ② When the power is turned on while the power-on-status clear flag is TRUE or while the *PSC common command is not available. The standard event status enable register is not affected by the following : ① Change in IEEE 488.1 device clear function status ② Reception of *RST common command ③ Reception of *CLS common command

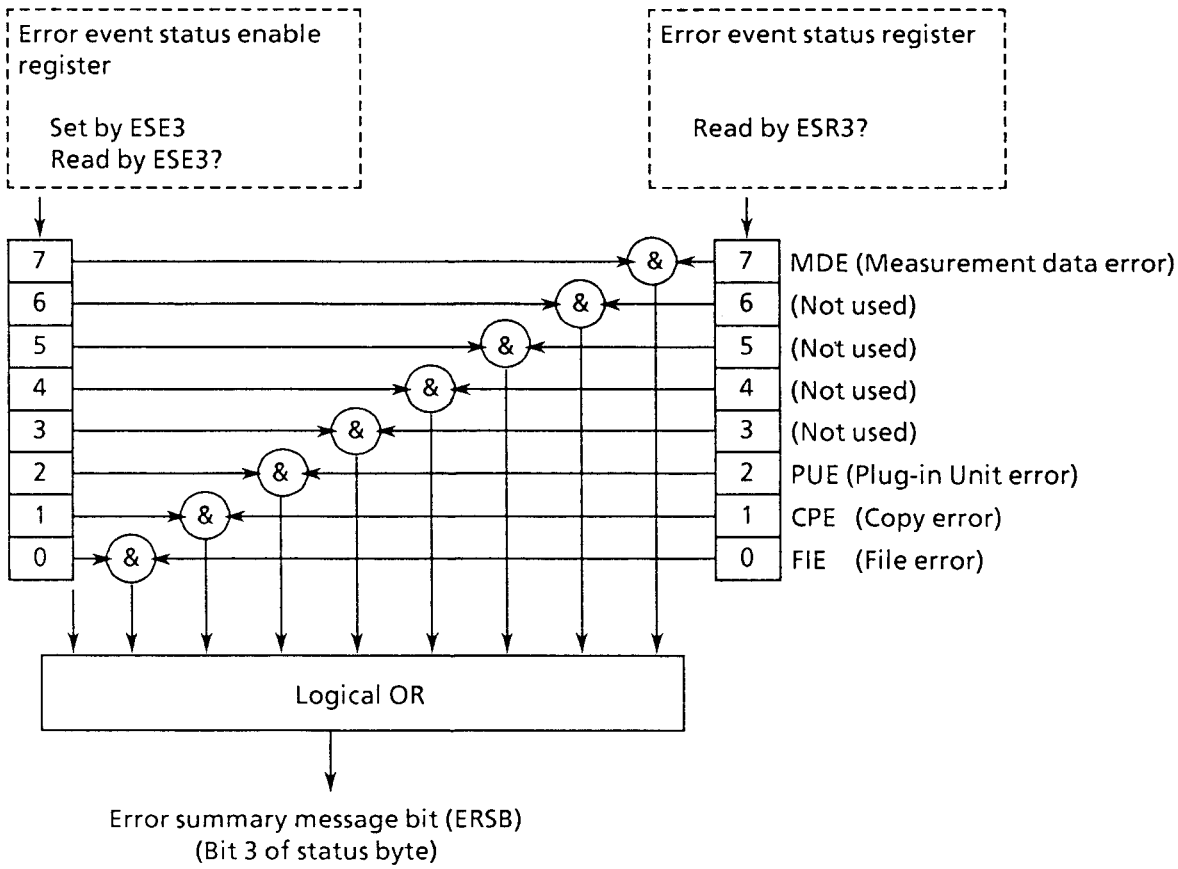
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.5 Extended Event Status Register

The MW9060A is provided with two registers (error event status register and termination event status register) for the extended event status register.

8.5.1 Bit Definition of Error Event Status Register

This register indicates errors specific to the MW9060A. This error event status register is cleared by a read with ESR3?, followed by clearing of bit 3 in the status byte.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

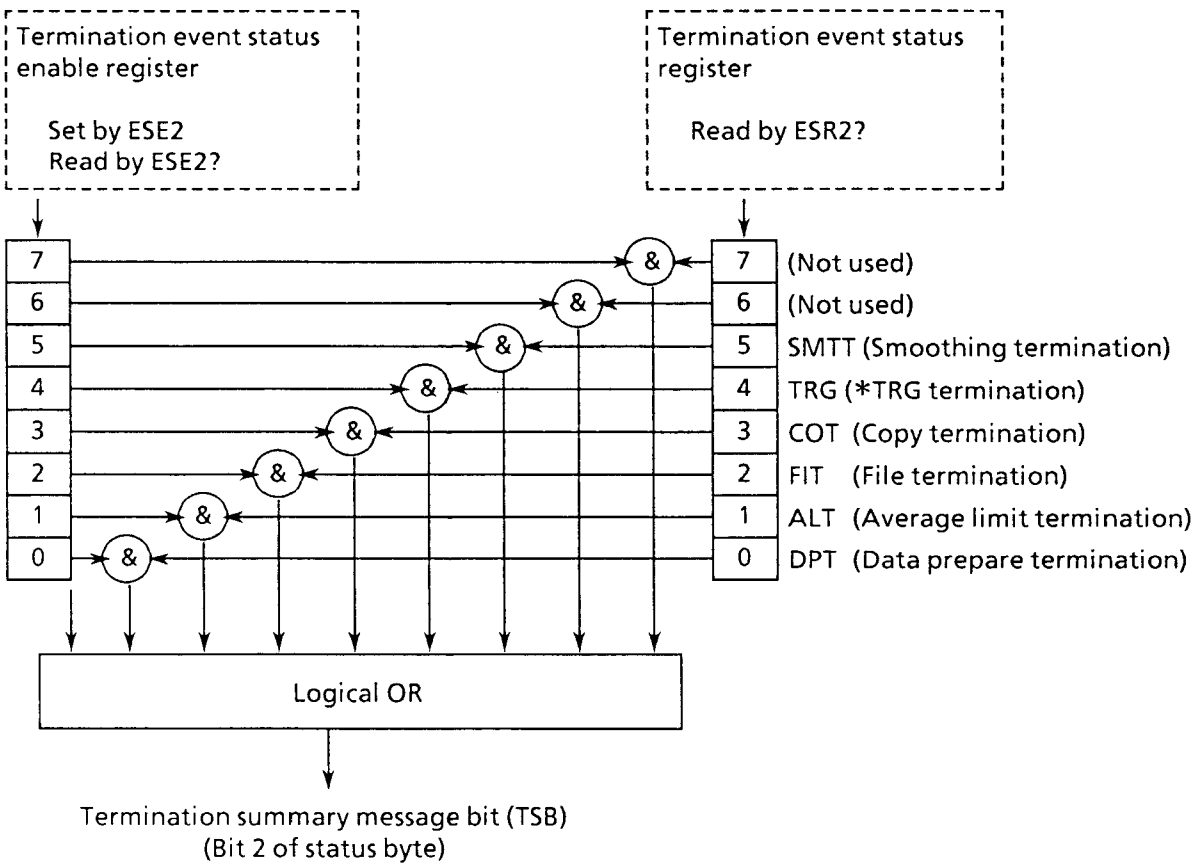
The following shows the definition of each event in the error event status register.

Bit	Event name	Description
7	MDE (Measurement data error)	0 : No error 1 : Set when DAT?, LOS?, SPL? AUT?, DZL?, or EVTD? is received while no waveform data is present, for example, immediately after power-on or immediately after average reset.
6		0 : Not used
5		0 : Not used
4		0 : Not used
3		0 : Not used
2	PUE (Plug-in Unit error)	0 : No error 1 : Plug-in Unit READ error that occurs when the return-loss parameter of the plug-in unit is abnormal at power on.
1	CPE (Copy error)	0 : No error 1 : Indicates all errors that occur when COPY is executed while paper is used up or the target device is not connected.
0	FIE (File error)	0 : No error 1 : Indicates all errors that occur when FILE is executed while the format is incorrect, no media is set into place, or the PMC battery has run down.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.5.2 Bit Definition of Termination Event Status Register

This register indicates termination of measurement or other operation. This termination event status register is cleared by a read with ESR2?, followed by clearing of bit 2 in the status byte.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

The following shows the definition of each event in the termination event status register.

Bit	Event name	Description
7		0 : Not used
6		0 : Not used
5	SMTT (Smoothing Termination)	0 : No reports 1 : Indicates that SMOOTHING operation is terminated. When an error occurs, the error event status register is set to prevent this bit from being affected.
4	TRG (*TRG Command Termination)	0 : No reports 1 : Indicates that *TRG operation is terminated. When an error occurs, the error event status register is set to prevent this bit from being affected.
3	COT (Copy termination)	0 : No reports 1 : Indicates that COPY operation is terminated. When an error occurs, the error event status register is set to prevent this bit from being affected.
2	FIT (File termination)	0 : Not reports 1 : Indicates that FILE operation is terminated. When an error occurs, the error event status register is set to prevent this bit from being affected.
1	ALT (Average limit termination)	0 : No reports 1 : Set when averaging the average limit set values is terminated.
0	DPT (Data prepare termination)	0 : No reports 1 : Indicates that preparation for measurement results is completed. This bit is set when waveform is prepared by executing laser-on or recall after a waveform-absent state.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.5.3 Reading, Writing, and Clearing Extended Event Status Registers

Read	The register is destructively read by a query. That is, the register is cleared after read. The END and ERROR event status registers are respectively read by the ESR2? and ESR3? queries. The read value is <NR1> obtained by converting the sum of binary-weighted values of event bits into a decimal number.
Write	The register cannot be written from external devices except clearing it.
Clear	The register is cleared in the following cases : ① When the *CLS command is received. ② When the power is turned on while the power-on-status clear flag is TRUE. ③ When an event is read for a query command.

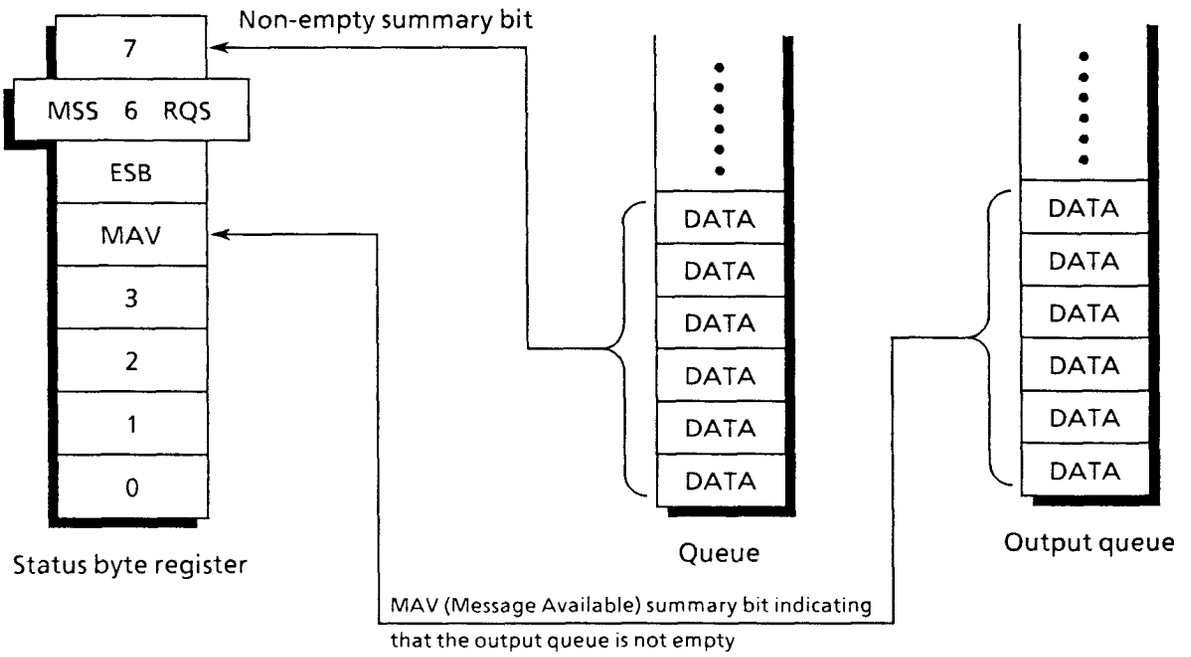
8.5.4 Reading, Writing, and Clearing Extended Event Status Enable Register

Read	The register is non-destructively read by a query. That is, the register is not cleared after read. The END and ERROR event status registers are respectively read by the ESE2? and ESE3? queries. The read value is returned with <NR1> obtained by converting the sum of binary-weighted values of event bits into a decimal number.
Write	The END and ERROR event status registers are respectively written by the ESE2 and ESE3 program commands. Because bits 0 to 7 of the register are respectively weighted to 1, 2, 4, 8, 16, 32, 64, and 128, the write data is sent by <DECIMAL NUMERIC PROGRAM DATA>, the sum of bit digit values selected from the bits.
Clear	The register is cleared in the following cases : ① When the ESE2 or ESE3 program command with data value of 0 is received for the END or ERROR event status register. ② When the power is turned on while the power-on-status clear flag is TRUE or while the *PSC command is not available. The extended event status enable register is not affected by the following : ① Change in IEEE 488.1 device clear function status ② Reception of *RST common command ③ Reception of *CLS common command

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.6 Queue Model

A queue model of the status data structure is shown on the right-hand side of the diagram below. A queue provides a means of reporting sequential status and other information with a data structure including a sequentially arranged information list. The presence of such information in the queue is indicated in a summary message. The contents of the queue are read through handshake when the device is in a talker active state (TACS).



A queue that outputs the MAV summary message, one of summary messages, to bit 4 of the status byte register is called the "output queue," and must always be included. A queue that can output the MAV summary message to any one of bits 0 to 3 or 7 of the status byte register is called simply a "queue," and is an optional facility. Because summary messages from register models can also be connected to bits 0 to 3 or 7 of the status byte register, the types of summary messages vary with each device.

For the Anritsu devices, bit 7 of the status byte register is assigned for use as the summary message from a "queue," but when the "output queue" can suffice, a "queue" is not used. Therefore, bit 7 of the status byte register is left unused.

The next page compares the "output queue" and a general queue.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Comparison between output Queue and General Queue

Item	Output queue	General queue
Data input/output type	FIFO (first-in first-out)	Need not always be the FIFO (first-in first-out)
Read	Can only be read through the protocol defined in Section 6. The type of response message unit read is determined by the query.	Read by a device-dependent query command. The response message unit read must be of the same type.
Write	<PROGRAM MESSAGE> element is not directly written to the output queue. Messages can only be transferred to and from the system interface through the protocol defined in Section 6.	<PROGRAM MESSAGE> element is not directly written to this queue. Indicates encoded device information.
Summary message	Becomes TRUE (1) when the output queue is not empty or FALSE (0) when the output queue is empty. The MAV summary message is used to synchronize information exchange between device and controller.	Becomes TRUE (1) when this queue is not empty or FALSE (0) when the queue is empty.
Clear	This queue is cleared in the following cases : ① When all items in the queue are read ② When the DCL bus command is received for message exchange initialization ③ When PON becomes TRUE at power-on ④ When operation is unterminated or interrupted	This queue is cleared in the following cases : ① When all items in the queue are read ② When the *CLS command is received ③ When cleared by other device-dependent means

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.7 Technique to Synchronize Device and Controller

There are three methods to synchronize operation between the device and controller.

① Forced sequential execution ..... (by the *WAI command)

② Wait for device's output queue response ..... (by the *OPC? query)

③ Wait for service request ..... (by the *OPC command/*OPC? query)

8.7.1 Forced Sequential Execution

The device-dependent commands may be classified into two types : sequential commands and overlap commands.

- Sequential command ..... A command or query sent from the controller which does not start executing the next command while the device is executing something.
- Overlap command ..... A command or query sent from the controller which starts executing the next command even when the device is executing something.

Forced sequential execution is a synchronization technique that forces a command which otherwise functions as an overlap command to operate sequentially, so that when one processing is completed, the next processing is initiated. The *WAI command is used for this synchronization technique. Note, however, that because all commands available with the MW9060A are sequential commands, results are the same regardless of whether or not the *WAI command is executed.

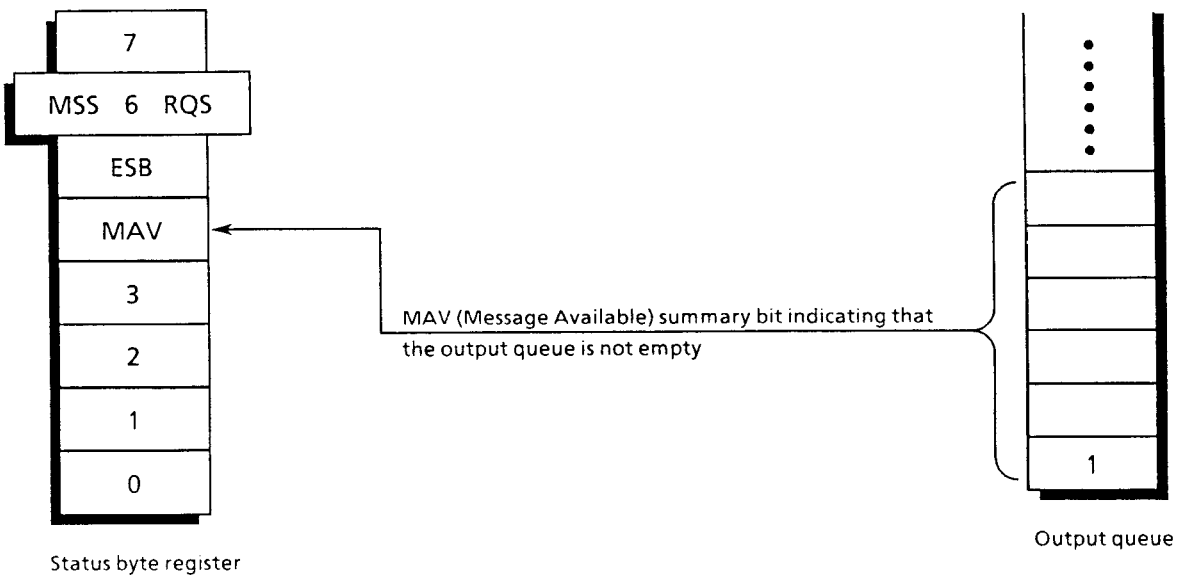
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.7.2 Wait for Device's Output Queue Response

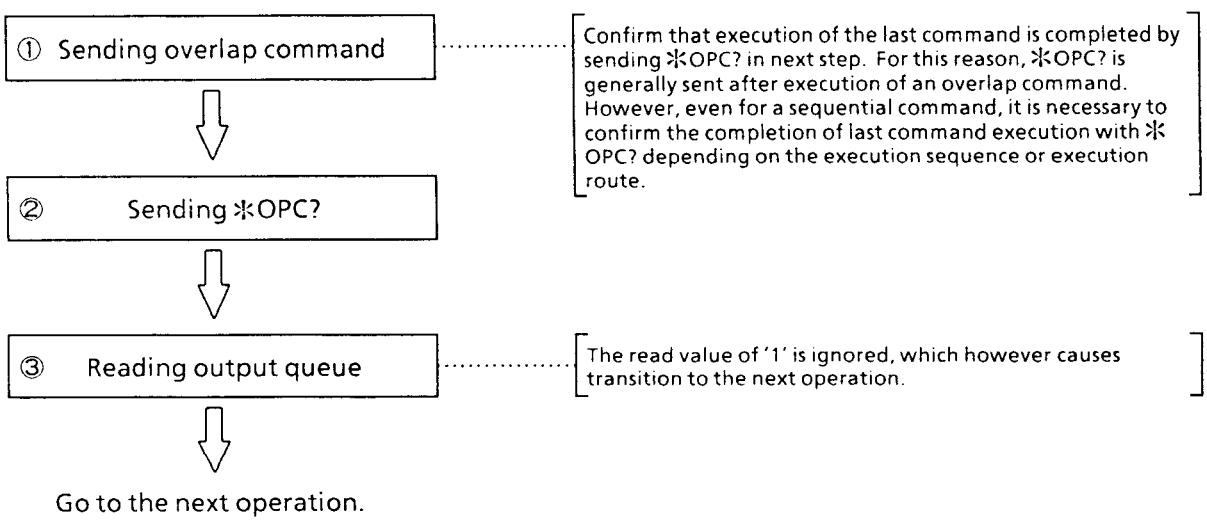
When the ✻OPC? query is executed, '1' is set in the output queue to generate the MAV summary message when the device completes all specified operations that have been pending.

Synchronization by a wait for output queue response is a method to synchronize operation between the device and controller by reading data 1 in the output queue or MAV summary message bit thus generated.

The method using the MAV summary message bit belongs to the method of synchronizing by a "Wait for service request," so details on it are described in paragraph 8.7.3. Description here is made of the method of synchronization by "reading the output queue."



<Output queue read sequence>

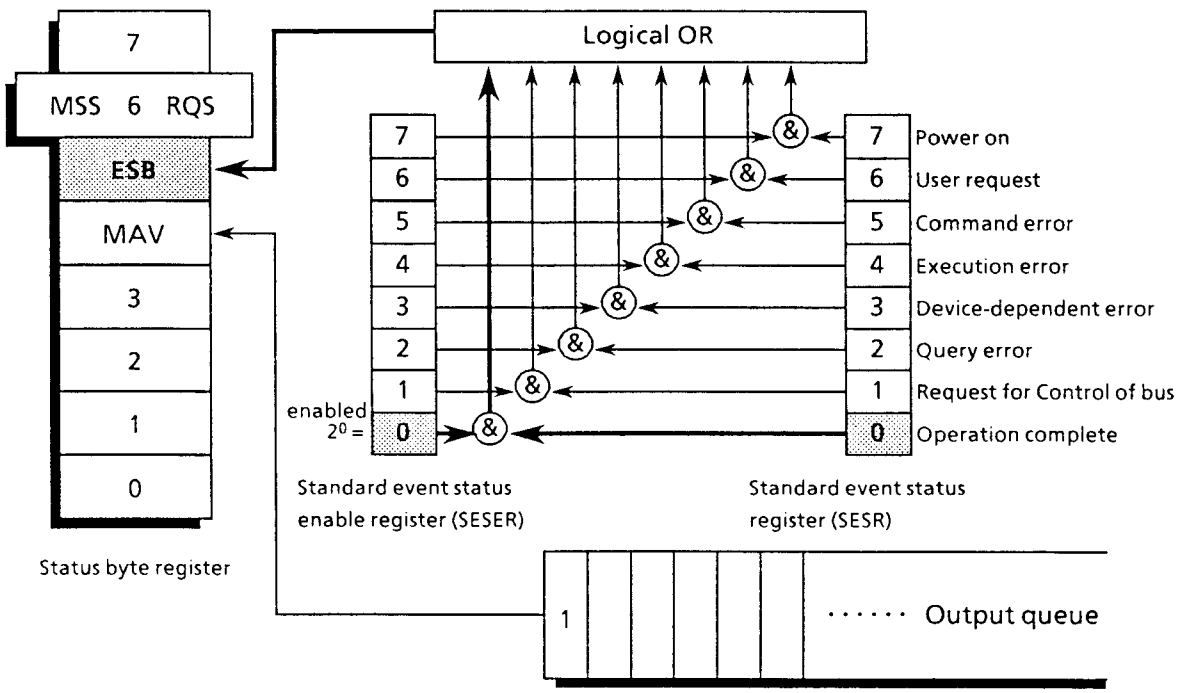


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

8.7.3 Waiting for Service Request

Wait for service request means that when the controller is interrupted by the SRQ signal from the device while executing normal processing, it temporarily stops that processing and begins another processing corresponding to the conditions of the device's status message.

In usual interruption handling, the device sends a request to the controller at any time regardless of what processing the controller is doing. In device and controller synchronization techniques, however, the controller sends the *OPC command or *OPC? query to the device to confirm whether the device operation is terminated before responding to the request. While waiting for the SRQ signal generated by an operation complete event, the controller proceeds with some other processing and, when the operation complete event is detected, begins the designated processing. This is a technique of synchronization by a wait for service request.



■ <Synchronization method by *OPC command>

① Enable the 2^0 bit (1) of the standard event status enable register. — WRITE @1Ø1:"*ESE 1"

② Enable the 2^5 bit (32) of the service request enable register. — WRITE @1Ø1:"*SRE 32"

③ Cause the device to execute the designated operation. — WRITE @1Ø1:<PROGRAM MESSAGE>

④ Execute the *OPC command (next is also executed because of an overlap command). — WRITE @1Ø1:"*OPC"

⑤ Wait for SRQ interrupt (ESB summary message). ···· Value of status byte is $2^6 + 2^5 = 96$.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ <Synchronization method by ✻OPC?query>

① Enable the 2^4 bit (16) of the service request enable register. — `WRITE @101:"✻SRE 16"`

⇩

② Cause the device to execute the designated operation. — `WRITE @101:<PROGRAM MESSAGE>`

⇩

③ Send the ✻OPC? query (waiting until operation ② is completed.) — `WRITE @101:"✻OPC?"`

⇩

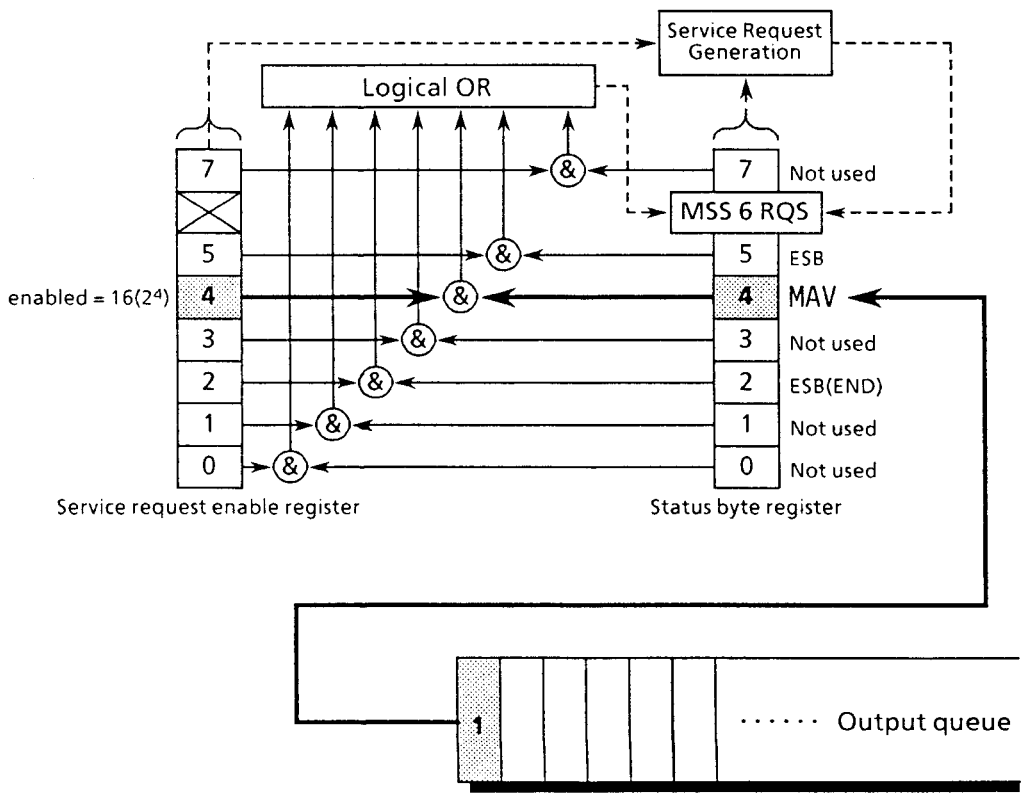
④ Read ASCII character '1' from output queue and abandon it.

⇩

⑤ Wait for SRQ interrupt (MAV summary message). ···· Value of status byte is $2^6 + 2^4 = 80$.

⇩

Go to the next operation.



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 9
DETAILS OF DEVICE MESSAGES

Common item

If a message that does not conform to the syntax or a message with a parameter value outside the specified range is received, an error occurs.

For example, a command error when the syntax is violated or an execution error when a parameter value is outside the specified range.

For the basic errors common to all messages, see Section 8 "Status Byte and Synchronization Technique".

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[1] APR/APR? (Approximate Method)

■ Syntax

Command syntax : APR␣ <linear approximation>

Query syntax : APR?

where

<linear approximation> ::= Number corresponding to the linear approximation method

0 : 2PA

1 : LSA (ALL)

2 : LSA (DISP)

■ Example

° To set linear approximation to 2PA (2-point approximation)

```
WRITE @101:"APR 0"
```

° To read the settings of linear approximation

```
10 DIM APPROX$*20
20 WRITE @101:"APR?"
30 READ @101:APPROX$
40 PRINT APPROX$
50 END
```

■ Description

The APR command changes the linear approximation method.

The query returns the current linear approximation method.

■ Response message format

Return format : APR␣ {0 | 1 | 2} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[2] ATA/ATA? (Auto Attenuator)

■ Syntax

Command syntax : ATA␣ {1 | 0}

Query syntax : ATA?

where

1 : ON (AUTO)

0 : OFF (MANUAL)

■ Example

- To set auto attenuator ON

```
WRITE @101:"ATA 1"
```

- To read the settings of auto attenuator

```
10 DIM AUTO_ATT$*20
20 WRITE @101:"ATA?"
30 READ @101:AUTO_ATT$
40 PRINT AUTO_ATT$
50 END
```

■ Description

The ATA command specifies whether the automatic adjustment (auto attenuator) of attenuator value is valid or not.

The query returns the current state of auto attenuator.

■ Response message format

Return format : ATA␣ {1 | 0} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[3] ATT/ATT? (Attenuator)

■ Syntax

Command syntax : ATT␣ <attenuator value>

Query syntax : ATT?

where

<attenuator value> :: = Attenuator value (expressed by an integer, in units of dB)

■ Example

- To set the attenuator to 25 dB

```
WRITE @101:"ATT 25"
```

- To read the settings of the attenuator

```
10 DIM ATT$*20
20 WRITE @101:"ATT?"
30 READ @101:ATT$
40 PRINT ATT$
50 END
```

■ Description

The ATT command changes the attenuator value. When the auto attenuator is on, it sets the specified attenuator value after turning it off.

The attenuator value is handled as a numeric value where 1 represents 1 dB. Note, however, that usable attenuator values differ with the plug-in unit used.

The query returns the current attenuator value. The attenuator value specified by the auto attenuator can be read by the query. The state of auto attenuator then set is unaffected by the query.

■ Response message format

Return format : ATT␣ <attenuator value> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[4] AUT? (Auto Measurement Data)

■ Syntax

Query syntax	:	AUT?	Reads the number of faulty points
Query syntax	:	AUT?␣<faulty point No.>	Reads the measurement results of the faulty point

■ Example

∘ To read the number of faulty points detected during automatic measurement

```
10 DIM AUT$*20
20 WRITE @101:"AUT?"
30 READ @101:AUT$
40 PRINT AUT$
50 END
```

∘ To read the measurement results of the first faulty point

```
10 DIM AUT_MEAS$*100
20 WRITE @101:"AUT? 1"
30 READ @101:AUT_MEAS$
40 PRINT AUT_MEAS$
50 END
```

■ Description

The AUT? query reads the number of faulty points detected by an automatic faulty point detector and measurement results for the specified faulty point. The <fiber length> indicates a value where IOR = 1.5 with length expressed in units of meter. When no faulty points are found during a read, 0 is returned as the number of faulty points.

When a faulty point is assigned a No. as parameter, the measured value for that faulty point is returned.

When laser is on, auto measurement results are updated every sweep. Therefore, after changing the marker selection, wait until one sweep is made before reading measurement results with the AUT? query. When laser is off, measurement results are updated each time the marker selection is changed. In this case, therefore, you can read measurement results immediately after changing the marker selection.

When the AUT? query is received while measurement item is not AUTO, an execution error results.

■ Response message format

① When reading the number of faulty points

Return format : AUT␣<total number of faulty points><LF I CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

where

<total number of faulty point> ::= Indicated by a number 1 to 5. 0 is returned when no faulty points are detected.

② When reading the measurement results of faulty point

Return Format : AUT␣ <faulty point No.>,<splice loss at faulty point>,
<fiber loss (L)>, <fiber length (L)>,
<transmission loss (L)>,<fiber loss (R)>,
<fiber length (R)>,<transmission loss (R)>,
<LF | CR,LF>

where

<faulty point No.> ::= Indicated by a number 1 to 5. 0 is error. Specifying a value greater than the total number of faulty points also results in an error.

<splice loss at faulty point> ::= Measured value of splice loss at faulty point. When the loss is unmeasurable (＊＊.＊＊＊ is displayed on the CRT screen) as in the case where the faulty point is located at the far end of the fiber, 900.000 is returned.

<fiber loss> ::= Loss in fiber connected to the right or left side of the faulty point. When the fiber loss cannot be measured because the faulty point No. is 1 or equal to the total number of faulty points, 900.000 is returned.

<fiber length> ::= Length of fiber connected to the right or left side of the faulty point. When unmeasurable, 000000.00 is returned.

<transmission loss> ::= Transmission loss in fiber connected to the right or left side of the faulty point. When unmeasurable (＊＊.＊＊＊ is displayed on the CRT screen), 900.000 is returned.

Note: (L) and (R) respectively denote the left side and right side of the screen.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[5] AUTD/AUTD? (Auto Data Select)

■ Syntax

Command syntax : AUTD␣ {1 | 0}

Query syntax : AUTD?

where

1 : All
0 : DISP

■ Example

° To select all the measurement data for the data used in auto search

```
WRITE @101:"AUTD  1"
```

° To check the data used in auto search

```
10 DIM AUTDT_SL$*20
20 WRITE @101:"AUTD?"
30 READ @101:AUTDT_SL$
40 PRINT AUTDT_SL$
50 END
```

■ Description

The AUTD command selects whether the data used in auto search consists of all the measurement data or display data.

The query returns the currently selected data.

■ Response message format

Return format : AUTD␣ {1 | 0} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[6] AVG/AVG? (Average)

■ Syntax

Command syntax : AVG␣ {1 | 0}

Query syntax : AVG?

where

1 : ON
0 : OFF

■ Example

- To set average ON

```
WRITE @101:"AVG  1"
```

- To read the settings of average

```
10 DIM AVERAGE$*20
20 WRITE @101:"AVG?"
30 READ @101:AVERAGE$
40 PRINT AVERAGE$
50 END
```

■ Description

The AVG command switches on or off measurement-waveform average processing.

Use the [AVG LIMIT] key (GPIB command, AVL) to set the average limit. Averaging is started when average is turned on and, when averaging reaches the average limit, laser is automatically turned off. The query returns the settings of average.

■ Response message format

Return format : AVG␣ {1 | 0} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[7] AVL/AVL? (Average Limit)

■ Syntax

Command syntax : AVL␣ <count or time>, <set value>

Query syntax : AVL?

where

<count or time> :: =A value corresponding to the averaging count or time.

0 : Count

1 : Time

<set value> :: =Set value of the selected item

Count : 1 =1 time

Time : 1 =1 second

■ Example

- To set the averaging count to 100 times

```
WRITE @101:"AVL 0,100"
```

- To read the settings of average limit

```
10 DIM AVG_LIMIT$*50
20 WRITE @101:"AVG?"
30 READ @101:AVG_LIMIT$
40 PRINT AVG_LIMIT$
50 END
```

■ Description

The AVL command changes the set value of average limit. When the settings of a limit item (time or count as selected) are changed, average is reset. The set value of an unchanged item (time or count) is not affected by this command.

When averaging reaches the set value, the AVL bit (bit 1) of the termination event status register is set.

The query returns the currently set value of the average limit. The current averaging count/averaging time is also returned.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ **Response message format**

Return format : AVL ␣ <count or time>,<set value of count>,<set value of time>,
<counter value of count or time>,<LF | CR, LF>

where

<counter value of count or time> ::= Current averaging count/time indicated by the counter

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[8] BSL/BSL? (Return-Loss Backscattered Level)

■ Syntax

Command syntax : BSL␣ <Backscattered level setting value>

Query syntax : BSL?

where

<Backscattered level setting value> ::= Backscattered level expressed by a real number with the units of −1 dB.

■ Example

- To set the backscattered level to −30 dB

```
WRITE @101:"BSL 30"
```

- To read the settings of backscattered level

```
10 DIM SCATTER$*20
20 WRITE @101:"BSL?"
30 READ @101:SCATTER$
40 PRINT SCATTER$
50 END
```

■ Description

The BSL command sets the backscattered level for return-loss measurement.

The backscattered level set value is the value with the units of −1 dB.

■ Response message format

Return format : BSL␣ <backscattered level setting value> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[9] CIT/CIT? (Copy Item)

■ Syntax

Command syntax : CIT␣ {0 | 1}

Query syntax : CIT?

where

0 : All items
1 : Waveform only

■ Example

∘ To set COPY items to waveform only

```
WRITE @101:"CIT 1"
```

∘ To read the settings of COPY items

```
10 DIM COPY_ITEM$*20
20 WRITE @101:"CIT?"
30 READ @101:COPY_ITEM$
40 PRINT COPY_ITEM$
50 END
```

■ Description

The CIT command selects the items for hard copy to be produced.

The query returns information of which items are selected.

■ Response message format

Return format : CIT␣ {0 | 1} <LF | CR,LF>

■ Errors

① When the CIT command is received while no plug-in unit is fitted in place, an execution error results.

② When the CIT command is not accompanied by an argument, a command error results. In this case, the state of COPY item is not affected.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[10] CON? (Connector)

■ Syntax

Query syntax : CON?

■ Example

◦ To read information on whether fiber is connected to the output connector

```
10 DIM CON$*20
20 WRITE @101:"CON?"
30 READ @101:CON$
40 PRINT CON$
50 END
```

■ Description

The CON? query returns information on whether fiber is currently connected to the output connector. When the plug-in unit is not fitted in place, 0 is always returned.

■ Response message format

Return format : CON␣ {1 | 0} <LF | CR,LF>

where

1 : Fiber is connected to the output connector

0 : Fiber is not connected to the output connector

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[11] CMP/CMP? (Compare Recall ON/OFF)

■ Syntax

Command Syntax : CMP␣{0 | 1}

Query syntax : CMP?

where

0 : Waveform compare recall-ON

1 : Waveform compare recall-OFF

■ Example

° To compare waveform after reading TRACE 1 file in INT PMC

```
10 WRITE @101:"MED 1"
20 WRITE @101:"CMP 1"
30 WRITE @101:"RCL TRACE 1"
40 END
```

° To read information on whether waveform-compare-recall is ON or OFF

```
10 DIM CMP$*20
20 WRITE @101:"CMP?"
30 READ @101:CMP$
40 PRINT CMP$
50 END
```

■ Description

The CMP command performs waveform comparison. After compare-recall is turned ON, execute the RCL command to become the waveform-comparison state. Even if in waveform comparison state at local mode, execute the CMP command to turn the compare-recall ON and then execute the RCL command.

Query returns the state of waveform-comparison recall.

■ Response message format

Return format : CMP␣{0 | 1} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[12] CPY/CPY? (Copy)

■ Syntax

Command Syntax : CPY␣ {1 | 0}, <Screen No.>

Query syntax : CPY?

where

<Screen No.> ::= Specifies the screen to make a hard copy.

0 : Measurement screen
1 : Event table
2 : Return loss parameter
3 : Recall file list
4 : Compare setting
5 : Interface
6 : Procedure

■ Example

- To start generating hard copy of measurement screen

```
WRITE @101:"CPY 1,0"
```

- To read information on whether hard copy is being produced

```
10 DIM CPY$*20
20 WRITE @101:"CPY?"
30 READ @101:CPY$
40 PRINT CPY$
50 END
```

■ Description

The CPY command outputs data to produce CRT-screen hard copy from an external interface.

The query returns information on whether hard copy data is currently output.

When the target is set to the internal printer, a response is not returned until copying ends. Accordingly, always 0 is returned.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ Response message format

Return format : CPY␣{1 | 0}, <Screen No.>,<LF | CR, LF>

where

1 : Being output

0 : Not output

■ Errors

① When the CPY command is received while no plug-in unit is fitted in place, an execution error results.

② When the CPY command is not accompanied by an argument, a command error results. In this case, the copy state is not affected.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[13] DAT? (Data)

■ Syntax

Query syntax : DAT?␣ <start distance of data>,<data interval>,
<data quantity n> [,<data type>]

where

<start distance of data> ::= Distance of first data to be read. Specify a distance that fits data resolution within the sampling range. Specify the distance in units of meter using a value expressed as 1 = 1m. IOR is 1.5.

<data interval> ::= Interval of data to be read. Specify a distance that is an integer multiple of the sampling resolution. Specify the distance in units of meter using a value expressed as 1 = 1m. IOR is 1.5.

<data quantity n> ::= Number of data entries to be read. Specify an integer number so that data at the n'th entry is not larger than the sampling end.

<data type> ::= Specify the return format with a value indicating whether it is the binary or ASCII format. When this specification is omitted, 0 is assumed.

0 : ASCII format
1 : Binary format

■ Example

- When reading 200 points beginning from a 10 km point with 10 m data resolution (interval)

```
10 DIM DATA (1000)
20 WRITE @101:"DAT? 10000,10,200"
30 TERM IS ","
40 READ @101:START_DISTANCE
50 READ @101:DATA_RES
60 READ @101:NUMBER
70 READ @101:DUMMY
80 FOR N=1 TO NUMBER-1
90 READ @101:DATA(N)
100 PRINT DATA(N)
110 NEXT N
120 TERM IS CHR$(10)
130 READ @101:DATA(NUMBER)
140 PRINT DATA(NUMBER)
150 END
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ Description

The DAT? query reads measured waveforms by specifying the start distance of data to be read, data interval (resolution), and data quantity. Waveforms are values in the range of 0 to 50 dB. <start distance of data> and <data interval> are values where IOR = 1.5 and length indicated in units of meter.

The return format of waveform data does not have a header message.

When two waveforms are displayed on the screen in a compare state, information on the selected waveform is output.

If a distance outside the sampling range or if a distance which cannot be displayed with the data resolution is specified for <start distance of data>, or if a distance which is not an integer multiple of the sampling resolution specified for <data interval>, an execution error occurs.

■ Response message format

Return format : <start distance of data>, <data interval>, <data quantity n>, <0>, <data at 1st entry>, <data at 2nd entry> ....<data at n'th entry> <LF | CR, LF>

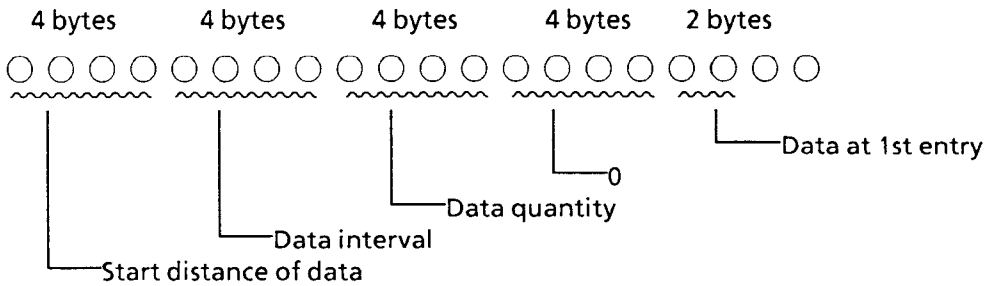
where

<start distance of data>	::=	Indicates the distance of <data at 1st entry>. When in ASCII format, it is indicated by 1 = 1 m. In binary format, it is indicated by 1 = 1 cm. IOR is 1.5. In binary format, it consists of 4 bytes.
<data interval>	::=	Interval between data entries. When in ASCII format, it is indicated by 1 = 1 m. In binary format, it is indicated by 1 = 1 cm. IOR is 1.5. In binary format, it consists of 4 bytes.
<data quantity n>	::=	Number of waveform data entries. This number does not include <start distance of data>, <data interval>, <data quantity n>, and <0>.
<0>	::=	Always 0.
<data at n'th entry>	::=	Indicates the waveform level. When in ASCII format, it is indicated by 1 = 1 dB. In binary format, it is indicated by 1 = 0.001 dB. In binary format, it consists of 2 bytes.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

<<binary format>>

Binary-format data are not separated with comma.



<start distance of data>, <data interval>, <data quantity n>, and <0> are expressed by 32 bits (4 bytes). 32 bits in one data entry are divided into four, and data is sent in units of 8 bits beginning with the MSB byte. The data of <start distance of data> and <data interval> are multiplied by 100.

One data entry for the waveform is expressed by 16 bits (2 bytes). 16 bits in one data entry are divided into two (high order 8 bits and low order 8 bits), and data is sent in units of 8 bits beginning with the MSB byte. The waveform data is multiplied by 1000.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[14] DATE/DATE? (Date)

■ Syntax

Command syntax : DATE␣ <year>, <month>, <day>

Query syntax : DATE?

where

<year> ::= Indicates a calendar year specified by a value from 0 to 99. The internal clock automatically processes leap years.

<month> ::= Indicates a month specified by a value from 1 to 12.

<day> ::= Indicates a day specified by a value from 1 to 31.

Note: Enter the correct date;taking 30-day month, 31-day month, and leap year into consideration.

■ Example

° To set the date to September 11, 1993

```
WRITE @101:"DATE 93,9,11"
```

° To read the date

```
10 DIM DATE$*20
20 WRITE @101:"DATE?"
30 READ @101:DATE$
40 PRINT DATE$
50 END
```

■ Description

The DATE command sets the date of the internal clock.

The query returns the date set in the internal clock.

■ Response message format

Return format : DATE␣ <year>, <month>, <day> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[15] DLEVT/DLEVT? (Delete Events)

■ Syntax

Command syntax : DLEVT␣ <event No.>

where

<event No.> ::= Integer value of 1 or more corresponding to the event.

■ Example

° To delete event No. 6

```
WRITE @101:"DLEVT 6"
```

■ Description

The DLEVT command deletes an event. If the event function is off or if an illegal event number is sent, an execution error occurs.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[16] DPEVT/DPEVT? (Display Events)

■ Syntax

Command syntax : DPEVT␣ <event No.>

Query syntax : DPEVT?

where

<event No.> ::= Integer value of 1 or more corresponding to the event number

■ Example

- To specify event No. 9

```
WRITE @101:"DPEVT 9"
```

- To read the specified event No.

```
10 DIM DPEVT$*20
20 WRITE @101:"DPEVT?"
30 READ @101:DPEVT$
40 PRINT DPEVT$
50 END
```

■ Description

The DPEVT command displays the waveform optimally scaled by the specified event number. If the event function is off or an illegal event number was specified by the DPEVT command, an execution error occurs.

The query returns the specified event number.

■ Response message format

Return format : DPEVT␣ <event No.><LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[17] DSR/DSR? (Distance Range)

■ Syntax

Command syntax : DSR␣ <distance range>

Query syntax : DSR?

where

<distance range> ::= Indicates distance range. Specify it using a value where distance is expressed in units of meter by an integer. IOR is 1.5.

■ Example

- To set the distance range to 25 km

```
WRITE @101:"DSR 25000"
```

- To read the settings of distance range

```
10 DIM DISTANCE$*20
20 WRITE @101:"DSR?"
30 READ @101:DISTANCE$
40 PRINT DISTANCE$
50 END
```

■ Description

The DSR command changes distance range. Use a value where IOR = 1.5, with length expressed in units of meter.

The value that can be assigned for distance range is discrete. Also, the usable distance range differs with the plug-in unit used.

The query returns the currently set distance range.

■ Response message format

Return format : DSR␣ <distance range> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[18] DZL? (Splice & Dead Zone Loss Measurement Data)

■ Syntax

Query syntax : DZL?

■ Example

◦ To read the measurement data

```
10 DIM DZ_DATA$*20
20 WRITE @101:"DZL?"
30 READ @101:DZ_DATA$
40 PRINT DZ_DATA$
50 END
```

■ Description

The DZL? query returns the SPL&DZL measurement result. If this query is sent when the measurement item is not SPL&DZL, an execution error occurs.

■ Response message format

Return format : DZL␣<faulty point splice loss>, <fiber loss (L)>, <fiber length (L)>, <transmission loss (L)>, <fiber loss (R)>, <fiber length (R)>, <transmission loss (R), <LF | CR, LF>

where

<faulty point splice loss> ::= Faulty point splice loss. If measurement is impossible when the faulty point is at the far end of the fiber, etc., 900.000 is returned.

<fiber loss> ::= Loss of fiber spliced at left and right sides of the faulty point. When measurement is impossible, 900.000 is returned.

<fiber length> ::= Length of fiber spliced at left and right sides of faulty point. When measurement is impossible, 900.000 is returned.

<transmission loss> ::= Transmission loss of fiber spliced at left and right sides of faulty point. When measurement is impossible, 900.000 is returned.

* (L) or (R) indicates the left or right side of the screen.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[19] ESE2/ESE2? (Termination Event Status Enable)

■ Syntax

Command syntax : ESE2␣ <register value>

Query syntax : ESE2?

where

<register value> ::= Indicated by an integer value from 0 to 255.

■ Example

- To disable all bits in the termination event status enable register

```
WRITE @101:"ESE2 0"
```

- To read the settings of the termination event status enable register

```
10 DIM ESE2$*20
20 WRITE @101:"ESE2?"
30 READ @101:ESE2$
40 PRINT ESE2$
50 END
```

■ Description

The ESE2 command sets (or resets) the bits of the termination event status enable register, one of the MW9060A's extended event status enable registers.

When an event is reported by the termination even status register, the termination event status enable register enables or disables the interrupt bit corresponding to the reported event. For details of conditions on which bits are masked, see the table below.

The ESE2? query returns the current contents of the termination event status enable register.

Table Termination Event Status Enable Register

Bit	Weight	Bit name	Enable conditions
7	1 2 8		Not used
6	6 4		Not used
5	3 2		Not used
4	1 6		Not used
3	8	CPT	End of COPY operation
2	4	FIT	End of FILE operation
1	2	ALT	End of averaging
0	1	DPT	End of measurement result preparation

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ **Response message format**

Return format : ESE2␣ <register value> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[20] ESE3/ESE3? (Error Event Status Enable)

■ Syntax

Command syntax : ESE3␣ <register value>

Query syntax : ESE3?

where

<register value> ::= Indicated by an integer number from 0 to 255.

■ Example

- To disable all bits of the error event status enable register

```
WRITE @101:"ESE3 0"
```

- To read the settings of the error event status enable register

```
10 DIM ESE3$*20
20 WRITE @101:"ESE3?"
30 READ @101:ESE3$
40 PRINT ESE3$
50 END
```

■ Description

The ESE3 command sets (or resets) the bits of the error event status enable register which is one of extended event status enable registers.

When a given error event in the error event status register is reported, the error event status enable register enables or disables an interrupt for that error. For details on how each bit is masked, see the table below.

The ESE3? query returns the current contents of the enable register.

Error Event Status Enable Register

Bit	Weight	Bit name	Enable conditions
7	1 2 8	MDE	Measurement result read error
6	6 4		Not used
5	3 2		Not used
4	1 6		Not used
3	8		Not used
2	4		Not used
1	2	CPE	COPY operation error
0	1	FIE	FILE operation error

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ **Response message format**

Return format : ESE3␣ <register value> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[21] ESR2? (Termination Event Status Register)

■ Syntax

Query syntax : ESR2?

■ Example

◦ To read the termination event status register

```
10 DIM ESR2$*20
20 WRITE @101:"ESR2?"
30 READ @101:ESR2$
40 PRINT ESR2$
50 END
```

■ Description

The ESR2? query returns the contents of the termination event status register which is one of extended event status registers. The termination event status register is cleared by a read, followed by clearing of bit 2 in the status byte.

Termination Event Status Register

Termination Event Status Register

(0 = False = Low, 1 = True = High)

Bit	Weight	Bit name	Conditions
7	1 2 8		0: Not used
6	6 4		0: Not used
5	3 2	SMTT	0: No termination event 1: Smoothing terminated
4	1 6	TRG	0: No termination event 1: *TRG operation terminated
3	8	CPT	0: No termination event 1: COPY operation terminated
2	4	FIT	0: No termination event 1: FILE operation terminated
1	2	ALT	0: No termination event 1: Averaging terminated
0	1	OPT	0: No termination event 1: Preparation for measurement results terminated

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ Response message format

Return format : ESR2␣ <register value> <LF | CR, LF>

where

<register value> ::= Indicated by an integer number from 0 to 255

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[22] ESR3? (Error Event Status Register)

■ Syntax

Query syntax : ESR3?

■ Example

° To read the error event status register

```
10 DIM ESR3$*20
20 WRITE @101:"ESR3?"
30 READ @101:ESR3$
40 PRINT ESR3$
50 END
```

■ Description

The ESR3? query returns the contents of the error event status register which is one of extended event status registers. The error event status register is cleared by a read, followed by clearing of bit 3 in the status byte.

Error Event Status Register

(0 = False = Low, 1 = True = High)

Bit	Weight	Bit name	Conditions
7	128	MDE	0 : No error 1 : Measurement result read error
6	64		0 : Not used
5	32		0 : Not used
4	16		0 : Not used
3	8		0 : No error
2	4	PUE	0 : No error 1 : Plug-in Unit read error
1	2	CPE	0 : No error 1 : COPY operation error
0	1	FIE	0 : No error 1 : FILE operation error

■ Response message format

Return format : ESR3␣ <register value> <LF | CR, LF>

where

<register value> ::= Indicated by an integer number from 0 to 255

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[23] ETBL/ETBL? (Display Events Table)

■ Syntax

Command syntax : ETBL␣ {1 | 0}

Query syntax : ETBL?

where

0 : OFF
1 : ON

■ Example

- To turn on the event table display

```
WRITE @101:"ETBL 1"
```

- To read the event table display status

```
10 DIM EV_TBL$*20
20 WRITE @101:"ETBL?"
30 READ @101:EV_TBL$
40 PRINT EV_TBL$
50 END
```

■ Description

The ETBL command displays the results of measurement by event function execution in a table. If the ETBL command is sent when the event function is off, an execution error occurs.

The query returns the event table display status.

■ Response message format

Return format : ETBL␣ {1 | 0} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[24] EVT/EVT? (Events)

■ Syntax

Command syntax : EVT␣ {1 | 0}

Query syntax : EVT?

where

0 : OFF
1 : ON

■ Example

° To turn on an event

```
WRITE @101:"EVT 1"
```

° To read the setting status of an event

```
10 DIM EVT$*20
20 WRITE @101:"EVT?"
30 READ @101:EVT$
40 PRINT EVT$
50 END
```

■ Description

The EVT command turns an event on and off.

The query returns the event status.

■ Response message format

Return format : EVT␣ {1 | 0} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[25] EVTD? (Events Data)

■ Syntax

Query syntax : EVTD? [␣<event No.>]

where

<event No.> ::= Indicated by an integer value of 1 or more corresponding to the event.

■ Example

∘ To read the total number of events

```
10 DIM N_EVT$*20
20 WRITE @101:"EVTD?"
30 READ @101:N_EVT$
40 PRINT N_EVT$
50 END
```

∘ To read the measured value of event No. 7

```
10 DIM DT_EVT$*20
20 WRITE @101:"EVTD? 7"
30 READ @101:DT_EVT$
40 PRINT DT_EVT$
50 END
```

■ Description

The EVTD? query returns the total number of events or the measured value of an event. If the EVTD? query is not accompanied by an argument, the total number of events is returned. If the EVTD? query is sent with <event No.> specified, the measured result corresponding to that <event No.> is returned. If the measurement item is not EVENTS, an execution error occurs.

■ Response message format

(1) Total number of events read

Return format : EVTD␣ <total number of events><LF | CR, LF>

where

<total number of events> ::= Indicated by a number from 1 to 100. If the faulty point cannot be detected, 0 is returned.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(2) Event point measured result read

Return format : EVTD ␣ <event No.>, <event splice loss>, <reflective>, <return loss>, <fiber loss (L)>, <fiber length (L)>, <transmission loss (L)>, <fiber loss (R)>, <fiber length (R)>, <transmission loss (R)>, <LF | CR, LF>

Where,

<event No.> ::= Specified event No. is returned.

<event splice loss> ::= Measured splice loss of event. If measurement is impossible when the event is at the far end of the fiber, etc., returns 900.000.

<reflective> ::= Returns the type of event.

0 : Nonreflective
1 : Reflective
2 : Saturative

<return loss> ::= Measured return loss of event. If the event is reflective or the measured result is an error, returns 900.000.

<fiber loss> ::= Loss of fiber spliced at left and right sides of event. When fiber loss cannot be measured because <event No.> is 1 or is not equal to the total number of events, returns 900.000.

<fiber length> ::= Length of fiber spliced at left and right sides of event. When measurement is impossible, returns 900.000.

<transmission loss> ::= Transmission loss of fiber spliced at left and right sides of event. When measurement is impossible, returns 900.000.

* (L) or (R) indicates the left or right side of the screen.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[26] EVTS/EVTS? (Events Select)

■ Syntax

Command syntax : EVTS␣ <event No.>

Query syntax : EVTS?

where

<event No.> ::= Indicated by an integer value of 1 or more corresponding to the event.

■ Example

- To specify event No. 3

```
WRITE @101:"EVTS 3"
```

- To read the specified event number

```
10 DIM EVTS$*20
20 WRITE @101:"EVTS?"
30 READ @101:EVTS$
40 PRINT EVTS$
50 END
```

■ Description

The EVTS command selects an event. If the event function is off or the EVTS command specifies an illegal event number, an execution error occurs.

The query returns the specified event number.

■ Response message format

Return format : EVTS␣ <event No.><LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[27] FDL (File Delete)

■ Syntax

Command syntax : FDL␣{<memory No.> | <file name>}

where

<memory No.> ::=Memory No. of INT MEMORY indicated by a number 1 to 32

<file name> ::=MS-DOS format file name

■ Example

- To erase a file named "TRACE2" stored in INT PMC

```
10 WRITE @101:"MED 1"
20 WRITE @101:"FDL TRACE 2"
30 END
```

- To memory No. 10 in INT MEMORY

```
10 WRITE @101:"MED 0"
20 WRITE @101:"FDL 10"
30 END
```

■ Description

The FDL command erases specified measurement data stored in the specified media.

Note: MS-DOS is a registered trademark of Microsoft Corporation.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[28] FEED (Feed)

■ Syntax

Command syntax : FEED␣ <number of lines>

where

<number of lines> ::= Number of paper feed lines indicated by an integer value from 1 to 50.

■ Example

- To line feed the paper approximately 13 mm

```
WRITE @101:"FEED 13"
```

■ Description

The FEED command feeds the built-in printer paper. If this command is sent when the target is not the built-in printer, an execution error occurs.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[29] FMT (Format)

■ Syntax

Command syntax : FMT

■ Example

- To format INT PMC

```
10 WRITE @101:"MED 1"
20 WRITE @101:"FMT"
30 END
```

- To erase all contents stored in INT MEMORY

```
10 WRITE @101:"MED 0"
20 WRITE @101:"FMT"
30 END
```

■ Description

The FMT command formats the specified media. When the media is formatted, all data stored on it is erased.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[30] FNC/FNC? (Function)

■ Syntax

Command syntax : FNC␣ <measurement item>

Query syntax : FNC?

where

<measurement item> ::= Number corresponding measurement items

0 : LOSS
1 : SPLICE
2 : AUTO
3 : RETURN LOSS
4 : SPL & DZL

■ Example

- To set the measurement item to LOSS

```
WRITE @101:"FNC 0"
```

- To read the settings of a measurement item

```
10 DIM FUNCTION$*20
20 WRITE @101:"FNC?"
30 READ @101:FUNCTION$
40 PRINT FUNCTION$
50 END
```

■ Description

The FNC command changes the measurement items (LOSS/SPLICE/AUTO/RETURN LOSS/SPL & DZL).

The query returns the current measurement item.

■ Response message format

Return format : FNC␣ <measurement item> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[31] FTR/FTR? (Full Trace)

■ Syntax

Command syntax : FTR␣ {1 | 0}

Query syntax : FTR?

where

1 : ON
0 : OFF

■ Example

- To set the Full-Trace function to ON

```
WRITE @101:"FTR 1"
```

- To read the settings of the Full-Trace mode

```
10 DIM FULL$*20
20 WRITE @101:"FTR?"
30 READ @101:FULL$
40 PRINT FULL$
50 END
```

■ Description

The FTR command turns the Full-Trace mode ON/OFF. The following three conditions are required to start the Full-Trace function.

① AVERAGE : ON

② LASER : ON

③ ATT : AUTO

The query returns the current setting conditions of the Full-Trace mode.

■ Response message format

Return format : FTR␣ {0 | 1} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[32] HSC/HSC? (Horizontal Scale)

■ Syntax

Command syntax : HSC␣ <horizontal scale value>

Query syntax : HSC?

where

<horizontal scale value> ::= Distance expressed in units of meter and specified by an integer

■ Example

- To set 250 m/div as horizontal scale

```
WRITE @101:"HSC 250"
```

- To read the setting of horizontal scale

```
10 DIM H_SCALE$*20
20 WRITE @101:"HSC?"
30 READ @101:H_SCALE$
40 PRINT H_SCALE$
50 END
```

■ Description

The HSC command changes the horizontal scale value (amount of one scale division (m/div) in horizontal direction relative to waveform screen).

The query returns the horizontal scale value. The value that can be assigned to the horizontal scale is discrete. It is expressed by a value where IOR = 1.5 with length indicated in units of meter. Also, the usable horizontal scale differs with the distance range.

■ Response message format

Return format : HSC␣ <horizontal scale value> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[33] HSF/HSF? (Horizontal Shift)

■ Syntax

Command syntax : HSF␣ <horizontal shift value>

Query syntax : HSF?

where

<horizontal shift value> ::= Distance expressed in units of meter and specified by an integer

■ Example

- To set horizontal shift to 100 m

```
WRITE @101:"HSF 100"
```

- To read the set value of horizontal shift

```
10 DIM H_SHIFT$*[20]
20 WRITE @101:"HSF?"
30 READ @101:H_SHIFT$
40 PRINT H_SHIFT$
50 END
```

■ Description

The HSF command changes the horizontal shift value (amount of movement in horizontal direction relative to waveform screen). When the rotary knob input item is other than horizontal shift, it is changed to horizontal shift. When an error occurs, the rotary knob input item does not change.

The query returns the horizontal shift value. It is expressed by a value where IOR = 1.5 with length indicated in units of meter.

■ Response message format

Return format : HSF␣ <horizontal shift value> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[34] IFS/IFS? (Interface Setting)

■ Syntax

Command syntax : IFS␣ <GPIB No.>, <set item>, <set value>

Query syntax : IFS? <GPIB No.>, <set item>

where

<GPIB No.> ::= Value corresponding to the GPIB No.

0 : GPIB 1
1 : GPIB 2

<set item> ::= Value corresponding to the set item

0 : GPIB address of controller (0 to 30)
1 : Printer output address (0 to 30)
2 : Plotter output address (0 to 30)
3 : MC8104A Data Storage Unit output address (0 to 30)

Note: Numbers in ( ) indicate the range of <set value> of each item.

<set value> ::= Set value of each set item

■ Example

◦ To set 10 as the plotter output address

```
WRITE @101:"IFS 1,2,10"
```

◦ To read the output address of the MC8104A Data Storage Unit

```
10 DIM IFS$*20
20 WRITE @101:"IFS? 1,3"
30 READ @101:IFS$
40 PRINT IFS$
50 END
```

■ Description

The IFS command sets the interface.

When the following messages are received, an execution error occurs:

① Message specifying GPIB 2 when GPIB 2 is not installed.

② Message with <GPIB 1 address> and <printer output address>, <plotter output address>, and <MC8104A data storage unit output address> set to the same address.

③ Message with <GPIB 2 address> and <printer output address>, <plotter output address>, and <MC8104A data storage unit output address> set to the same address.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ **Response message format**

Return format : IFS␣ <GPIB No.>, <set item>, <set value> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[35] IEVT/IEVT? (Insert Events)

■ Syntax

Command syntax : IEVT␣ <event position>

Query syntax : IEVT?

where

<event position> ::= Distance at which the event is to be placed specified in units of meters. Here, IOR is 1.5.

■ Example

° To set event position to 19km

```
WRITE @101:"IEVT 19000"
```

° To read the setting set status

```
10 DIM IEVT$*100
20 WRITE @101:"IEVT?"
30 READ @101:IEVT$
40 PRINT IEVT$
50 END
```

■ Description

The IEVT command inserts an event. If the IEVT command is sent when the event function is off, an execution error occurs.

The query returns the inserted event information.

■ Response message format

Return format : IEVT ␣ <event position>, <X1 marker position>, ···, <X5 marker position><LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[36] INI (Initialize)

■ Syntax

Command syntax : INI

■ Example

- To initialize

```
WRITE @1Ø1:"INI"
```

■ Description

The INI command initializes the MW9060A. This command performs the same functions as the [Initialize] key except that it is executed in the remote mode.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[37] IOR/IOR? (IOR)

■ Syntax

Command syntax : IOR␣ <IOR>

Query syntax : IOR?

where

<IOR> ::= Index of refraction. The setting range is 1.400 000 to 1.699 999.

■ Example

- To set 1.45 as IOR

```
WRITE @101:"IOR 1.45"
```

- To read the settings of IOR

```
10 DIM IOR$*20
20 WRITE @101:"IOR?"
30 READ @101:IOR$
40 PRINT IOR$
50 END
```

■ Description

The IOR command changes the setting of refractive index.

Changing the IOR setting by the IOR command only results in changing the value indicated on the CRT screen. In GPIB, calculation is always made assuming a IOR value of 1.5. The value (where IOR = 1.5) can be converted to a value based on the IOR (that you set) using the following equation

$$\text{<desired distance>} = \text{<distance read via GPIB>} \times \frac{1.5}{\text{Set value of IOR}}$$

The query returns the currently set value of refractive index.

■ Response message format

Return format : IOR␣ <IOR> <LF ! CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[38] ITEM/ITEM? (Item)

■ Syntax

Command syntax : ITEM␣ <file type>

Query syntax : ITEM?

where

<file type> ::= Number corresponding to file type.

0 : Waveform
1 : Waveform and event table
2 : Event table
3 : Procedure

■ Example

- To set the file type to event table

```
WRITE @101:"ITEM 2"
```

- To read the file type setting status

```
10 DIM FL_TYPE$*20
20 WRITE @101:"ITEM?"
30 READ @101:FL_TYPE$
40 PRINT FL_TYPE$
50 END
```

■ Description

The ITEM command specifies the file type when writing and reading the media.

The query returns the current file type.

■ Response message format

Return format : ITEM␣ <type of file><LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[39] LD/LD? (Laser)

■ Syntax

Command syntax : LD␣ {1 | 0}

Query syntax : LD?

where

1 : ON
0 : OFF

■ Example

- To turn laser on

```
WRITE @101:"LD 1"
```

- To read the settings of laser output

```
10 DIM LD$*20
20 WRITE @101:"LD?"
30 READ @101:LD$
40 PRINT LD$
50 END
```

■ Description

The LD command switches laser output on and off.

Laser output is automatically turned off when fiber is removed from the output connector or laser temperature control gets out of range.

If an LD ON command is sent when the laser output cannot be turned on (fiber not connected, READY lamp not lit, etc.), an execution error occurs.

The query returns the current state of laser output.

■ Response message format

Return format : LD␣ {1 | 0} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[40] LOS? (Loss Measurement Data)

■ Syntax

Query syntax : LOS?

■ Example

- To read the LOSS measurement results

```
10 DIM LOSS$*50
20 WRITE @101:"LOS?"
30 READ @101:LOSS$
40 PRINT LOSS$
50 END
```

■ Description

The LOS? query reads the measured values of loss between two points, distance between two points, and transmission loss. If the LOS? query is sent when the measurement item is not LOSS, an execution error occurs. The length (used in distance between two points and in transmission loss) is expressed in units of meter. When laser is on, the measurement results of loss between two points and transmission loss are updated every sweep. Therefore, when reading measurement results with the LOS? query, wait until one sweep is made after moving the marker.

When laser is off, measurement results are updated each time the marker is moved. Therefore, measurement results can be read immediately after moving the marker.

■ Response message format

Return format : LOS␣ <loss between two points>, <distance between markers>, <transmission loss> <LF | CR, LF>

where

<loss between two points> ::= Difference in waveform levels between two markers set during LOSS measurement. When unmeasurable (**.**** is displayed on the CRT screen.), 900.000 is returned.

<distance between markers> ::= Distance between two markers set during LOSS measurement

<transmission loss> ::= <loss between 2 points> divided by <distance between markers>
When unmeasurable (**.**** is displayed on the CRT screen.), 900.000 is returned.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[41] MED/MED? (Media)

■ Syntax

Command syntax : MED␣ <media>

Query syntax : MED?

where

<media> ::= A number corresponding to media.

0 : INT MEMORY
1 : INT PMC
2 : EXT PMC1
3 : EXT PMC2
4 : EXT FDD
5 : INT FDD

■ Example

- To set the media to INT PMC

```
WRITE @101:"MED 1"
```

- To read the settings of media

```
10 DIM MEDIA$*20
20 WRITE @101:"MED?"
30 READ @101:MEDIA$
40 PRINT MEDIA$
50 END
```

■ Description

The MED command changes the settings of media (storage media).

The query returns the current settings of media.

■ Response message format

Return format : MED␣ <media><LF | CR,LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[42] MKP/MKP? (Marker Position)

■ Syntax

Command syntax : MKP␣<marker No.>,<distance>

(1) When specifying marker

Query syntax : MKP?␣<marker No.>

(2) When not specifying marker

Query syntax : MKP?

---

where

<marker No.> ::= Marker No. is defined depending on measurement item as follows:

(a) For LOSS measurement
- 0 : * marker
- 1 : X1 marker

(b) For SPLICE measurement
- 0 : * marker
- 1 : X1 marker (X marker at the leftmost position)
- 2 : X2 marker (X marker at the second position from the left)
- 3 : X3 marker (X marker at the second position from the right)
- 4 : X4 marker (X marker at the rightmost position)

(c) For AUTO measurement
- 1 : Fault point at the first position from the left
- 2 : Fault point at the second position from the left
- 3 : Fault point at the third position from the left
- 4 : Fault point at the fourth position from the left
- 5 : Fault point at the fifth position from the left

(d) For R.LOSS measurement
- 0 : * marker
- 1 : X1 marker

(e) For SPL & DZL measurement
- 0 : * marker
- 1 : X1 marker (X marker at the second position from the left)
- 2 : X2 marker (X marker at the third position from the left)
- 3 : X3 marker (X marker at the third position from the right)
- 4 : X4 marker (X marker at the second position from the left)
- 6 : X6 marker (X marker at the leftmost position)
- 7 : X7 marker (X marker at the rightmost position)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(f) For EVENTS

0 : * marker
1 : X1 marker (X marker at the leftmost position)
2 : X2 marker (X marker at the second position from the left)
3 : X3 marker (X marker at the second position from the right)
4 : X4 marker (X marker at the rightmost position)
5 : X5 marker (X marker at the third position from the right)

<distance> ::= Specify the distance using a value where it is expressed in units of meter as 1 = 1 m. IOR is 1.5.

■ Example

- To set X1 marker to 2 km

```
WRITE @101:"MKP 1,2000"
```

- To read the distance where ✳ marker is set

```
10 DIM POSITION$*20
20 WRITE @101:"MKP? 0"
30 READ @101:POSITION$
40 PRINT POSITION$
50 END
```

■ Description

The MKP command moves the marker position. The range of marker movement is within the sampling range as in local mode. If movement beyond the range is specified, the limit value that can then be assumed is set. If the input item of the rotary knob was not MARKER, it is changed to MARKER. When an error occurs, the rotary knob input item does not change. If the MKP command is sent when the measurement item is AUTO, an execution error occurs.

If a distance is specified that may cause the marker to leave an adjacent marker behind, the adjacent marker is also moved.

When moved, a marker is assumed to have been selected and becomes a selected marker. Consequently, the cursor line moves onto this marker.

Since the ✳ and X1 to X4 markers move simultaneously, set the ✳ marker at first before setting the X1 to X4 markers.

The query returns the marker position in terms of distance.

The distance is indicated in units of meter by a value where IOR = 1.5.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ **Response message format**

(1) When specifying a marker

Return format : MKP␣ <distance> <LF | CR, LF>

Note: If the measurement item is AUTO, Markers cannot be moved.

(2) When not specifying a marker

Return format : MKP␣ <number of markers n>, <distance of No. 1 marker> ...
<distance of No. n marker> <LF | CR, LF>

Note: If there is no fault point when the measurement item is AUTO, <number of markers> is set to 0 and no parameters are appended after it. If there is any fault point, <number of markers> is changed to the number of fault points and the fault point distance is returned as a marker distance.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[43] MKS/MKS? (Marker Select)

Syntax

Command syntax : MKS␣ <marker No.>

Query syntax : MKS?

where

<marker No.> ::= Marker No. is defined depending on measurement item as follows:

(a) For LOSS measurement
- 0 : * marker
- 1 : X1 marker

(b) For SPLICE measurement
- 0 : * marker
- 1 : X1 marker (X marker at the leftmost position)
- 2 : X2 marker (X marker at the second position from the left)
- 3 : X3 marker (X marker at the second position from the right)
- 4 : X4 marker (X marker at the rightmost position)

(c) For AUTO measurement
- 1 : Fault point at the first position from the left
- 2 : Fault point at the second position from the left
- 3 : Fault point at the third position from the left
- 4 : Fault point at the fourth position from the left
- 5 : Fault point at the fifth position from the left

(d) For R.LOSS measurement
- 0 : * marker
- 1 : X1 marker

(e) For SPL & DZL measurement
- 0 : * marker
- 1 : X1 marker (X marker at the second position from the left)
- 2 : X2 marker (X marker at the third position from the left)
- 3 : X3 marker (X marker at the third position from the right)
- 4 : X4 marker (X marker at the second position from the left)
- 6 : X6 marker (X marker at the leftmost position)
- 7 : X7 marker (X marker at the rightmost position)

(f) For EVENTS
- 0 : * marker
- 1 : X1 marker (X marker at the leftmost position)
- 2 : X2 marker (X marker at the second position from the left)
- 3 : X3 marker (X marker at the second position from the right)
- 4 : X4 marker (X marker at the rightmost position)
- 5 : X5 marker (X marker at the third position from the right)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ Example

- To select the X4 marker

```
WRITE @101:"MKS 4"
```

- To read the selected marker No.

```
10 DIM SELECT$*20
20 WRITE @101:"MKS?"
30 READ @101:SELECT$
40 PRINT SELECT$
50 END
```

■ Description

The MKS command selects a marker. The cursor line is displayed on the selected marker. When the rotary knob input item is not marker, it changes to marker. The rotary knob input item also changes to marker when an error occurs.

The query returns information on which marker is selected.

If the fault point cannot be detected when the measurement item is AUTO, 0 is returned as <number of markers> to the MKS? query.

■ Response message format

Return format : MKS␣ <marker No.><LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[44] MSP/MSP? (Mask Position)

■ Syntax

Command syntax : MSP␣ <mask No.>, <distance>

(1) When specifying mask

Query syntax : MSP?␣ <mask No.>

(2) When not specifying mask

Query syntax : MSP?

where

<mask No.> ::= A number (0 to 4) corresponding to the mask

<distance> ::= Distance is specified in units of meter by a integer

■ Example

° To set mask 0 to 15 km

```
WRITE @101:"MSP 0,15000"
```

° To read the set position of mask 2

```
10 DIM MASK_POSITION$*100
20 WRITE @101:"MSP? 2"
30 READ @101:MASK_POSITION$
40 PRINT MASK_POSITION$
50 END
```

■ Description

The MSP command changes a mask position.

If a distance is 0 km, the mask is turned off.

Masks cannot be set within an adjacent distance of 1 m from a previously set marker.

If not set like this, an execution error occurs.

The query returns the current mask position. For masks not set, 0 km is returned.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ Response message format

(1) When specifying a mask

Return format : MSP␣ <distance> <LF | CR, LF>

(2) When not specifying a mask

Return format : MSP␣ <number of masks n>, <distance of No. 1 mask>,...,
<distance of No. n mask> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[45] NMSK/NMSK? (Near End Mask)

■ Syntax

Command syntax : NMSK␣<deviation>

Query syntax : NMSK?

where

<deviation> ::= An integer (−10 to 500) in units of meter, which is a deviated distance from the specific distance of the plug-in unit used.

Note: The specific distance is the near-end mask width specific to each plug-in unit. The distance is initialized by the *RST or INI command.

■ Example

- To widen the near-end mask width by 10 m

```
10 WRITE @101:"NMSK 10"
```

- To read the set deviation

```
10 DIM NEAR_MASK$*20
20 WRITE @101:"NMSK?"
30 READ @101:NEAR_MASK$
40 PRINT NEAR_MASK$
50 END
```

■ Description

The NMSK command changes the deviation value of the near-end mask width.
If an argument is 0 m, the NEAR END mask mode is turned off.
If the NMSK command is sent when a unit that does not use an EO switch is installed, and if the near-end mask width setting (inherent value + changed value) is a negative number, etc., an execution error occurs.
If the NMSK command is received while no plug-in unit is fitted into position, an error results.
These commands are effective only to the plug-in unit which uses a E/O switch as an optical switch.
If the NMSK command is received while any other type than the effective plug-in unit is fitted into position, an error results.
The query returns the current deviation (difference between the set near-end mask width and the specific value).

■ Response message format

Return format : NMSK␣ <deviation> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[46] PLG? (Plug-in Unit)

■ Syntax

Query syntax : PLG?

■ Example

° To read the type of the currently fitted plug-in unit

```
10 DIM PLUGIN$*20
20 WRITE @101:"PLG?"
30 READ @101:PLUGIN$
40 PRINT PLUGIN$
50 END
```

■ Description

The PLG? query returns the type of the current plug-in unit.

■ Response message format

Return format : PLG␣ <name of plug-in unit> <LF | CR, LF>

where

<name of plug-in unit> ::= Name of plug-in unit is defined by 16-character ASCII character string.

When no plug-in unit is fitted, the following character string is returned.

"NO␣UNIT␣␣␣␣␣␣␣␣␣␣"

Also, if the fitted plug-in unit is not a plug-in unit for the MW9060A, the following character string is returned.

"UNDEFINED␣UNIT␣␣"

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[47] PLS/PLS? (Pulse Width)

■ Syntax

Command syntax : PLS␣ <pulse width>

Query syntax : PLS?

where

<pulse width> ::= Value of pulse width (n units of ns, integer)

■ Example

- To set 1 μs as pulse width

```
WRITE @101:"PLS 1000"
```

- To read the settings of pulse width

```
10 DIM PULSE$*20
20 WRITE @101:"PLS?"
30 READ @101:PULSE$
40 PRINT PULSE$
50 END
```

■ Description

The PLS command changes the pulse width.

The pulse width is set by an integer in units of ns. The usable pulse width depends on the plug-in unit.

The query returns the current pulse width.

■ Response message format

Return format : PLS␣ <pulse width> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[48] PRO/PRO? (Procedure)

■ Syntax

Command syntax : PRO␣ {1 | 0}

Query syntax : PRO?

where

1 : ON
0 : OFF

■ Example

- To turn on the procedure function

```
WRITE @101:"PRO 1"
```

- To read the procedure function status

```
10 DIM PROCDR$*20
20 WRITE @101:"PRO?"
30 READ @101:PROCDR$
40 PRINT PROCDR$
50 END
```

■ Description

The PRO command turns the procedure function on and off.

The query returns the procedure function status.

If this command is sent while a procedure is executing, procedure execution is interrupted.

■ Response message format

Return format : PRO␣ {1 | 0} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[49] PWR/PWR? (LD Power)

■ Syntax

Command syntax : PWR␣ <LD output power>

Query syntax : PWR?

where

<LD output power> ::= A value expressed by a number from 0 to 127

■ Example

- To set the LD output power to 123

```
WRITE @101:"PWR 123"
```

- To read the LD output power

```
10 DIM LD_POWER$*20
20 WRITE @101:"PWR?"
30 READ @101:LD_POWER$
40 PRINT LD_POWER$
50 END
```

■ Description

The PWR command changes the optical output power of the OTDR.

If the PWR command is sent when a unit whose output power cannot be changed is installed, an execution error occurs.

The query returns the current LD output power.

■ Response message format

Return format : PWR␣ <LD output power> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[50] RCL/RCL? (Recall)

■ Syntax

Command syntax : RCL␣ {<memory No.> | <file name>}

Query syntax : RCL?

where

<memory No.> ::= Memory No. of INT MEMORY expressed by a number from 1 to 32

<file name> ::= MS-DOS format file name

■ Example

- To recall a file named "TRACE1"

```
WRITE @101:"RCL TRACE1"
```

- To read the recalled file name

```
10 DIM RECALL$*20
20 WRITE @101:"RCL?"
30 READ @101:RECALL$
40 PRINT RECALL$
50 END
```

■ Description

The RCL command reads the measurement data stored in a media, and displays it on the screen. Select the media by the MED command before executing the RCL command. In the waveform-compare-recall-ON state with the CMP command, execute the RCL command to compare waveforms.

Query returns the current-readout measurement-data file name. If data is not read out, 0 is returned.

If the measurement data read by RCL command is measured by a unit different from the installed unit or the unit was set to the waveform compare mode by local operation, an execution error occurs.

■ Response message format

Return format : RCL␣ {<memory No.> | <file name>} <LF | CR, LF>

Note: MS-DOS is a registered trade mark of Microsoft Corporation.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[51] RDA/RDA? (Result Display Area)

■ Syntax

Command syntax : RDA␣ {1 | 0}

Query syntax : RDA?

where

0 : Scale Area
1 : Table

■ Example

- To display the measured result in table form

```
WRITE @101:"RDA 1"
```

- To read the measured result display format

```
10 DIM DPAREA$*20
20 WRITE @101:"RDA?"
30 READ @101:DPAREA$
40 PRINT DPAREA$
50 END
```

■ Description

The RDA command specifies the measured result display format. There are two measured result display formats: display in table format at bottom left of scale (Table) and display to inside of scale (Scale Area).

The query returns the current display format.

■ Response message format

Return format : RDA␣ {1 | 0} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[52] RDY? (Ready)

■ Syntax

Query syntax : RDY?

■ Example

° To read information on whether the laser temperature control is within the range

```
10 DIM READY$*20
20 WRITE @101:"RDY?"
30 READ @101:READY$
40 PRINT READY$
50 END
```

■ Description

The RDY? query returns information on whether the laser-temperature control is currently possible. Laser output does not turn ON when the control is not possible. When no plug-in unit is fitted in place, 0 is always returned.

■ Response message format

Return format : RDY␣ {1 | 0} <LF | CR, LF>

where

1 : Temperature control with in range (possible) (Ready)

0 : Temperature control out of range (impossible) (Not ready)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[53] RMEM/RMEM? (Regist Memory Number)

■ Syntax

Command syntax : RMEM␣ <memory No.>

Query syntax : RMEM?

where

<memory No.> ::= Number corresponding to function key assigned operation procedure

1 : F1
2 : F2
3 : F3
4 : F4
5 : F5
6 : F6

■ Example

◦ To assign operation procedure to function key 5

```
WRITE @101:"RMEM 5"
```

◦ To read assigned function key

```
10 DIM RMEMNO$*20
20 WRITE @101:"RMEM?"
30 READ @101:RMEMNO$
40 PRINT RMEMNO$
50 END
```

■ Description

The RMEM command assigns an operation procedure to a function key by procedure register function.

The query returns the number corresponding to the assigned function key.

■ Response message format

Return format : RMEM␣ <memory No.> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[54] RNG/RNG? (Save Range)

■ Syntax

Command syntax : RNG␣ {1 | 2}

Query syntax : RNG?

where

1 : All measured range
2 : Displayed range only

■ Example

° To set all measured range as the range to save data

```
WRITE @101:"RNG 1"
```

° To read the settings of save range

```
10 DIM RANGE$*50
20 WRITE @101:"RNG?"
30 READ @101:RANGE$
40 PRINT RANGE$
50 END
```

■ Description

The RNG command sets the range of data to be saved (save range). When this command is not accompanied by an argument, the save range then set is unaffected.

The query returns the current set value of save range.

When the media is INT MEMORY, all the measurement range can not be specified. If specified, an execution error results.

■ Response message format

Return format : RNG ␣ {1 | 2} <save start>, <save end>, <save resolution> <LF | CR, LF>

where

<save start> ::= Distance of data at the beginning of the data to be saved. Expressed by a value where IOR = 1.5 with length indicated in units of meter.

<save end> ::= Distance of data at the end of the data to be saved. Expressed by a value where IOR = 1.5 with length indicated in units of meter.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

<save resolution> ::= Data resolution of the data to be saved. Expressed by a value where IOR = 1.5 with length indicated in units of meter.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[55] RTL? (Return-Loss Measurement Data)

■ Syntax

Query syntax : RTL?

■ Example

° To read the return-loss measurement results

```
10 DIM LOSS*20
20 WRITE @101:"RTL?"
30 READ @101:LOSS$
40 PRINT LOSS$
50 END
```

■ Description

The RTL? query reads the measured results of return loss.

When the measurement item is not RETURN LOSS, an execution error results.

■ Response message format

Return format : RTL␣ <return loss> <LF | CR, LF>

where

<return loss> ::= Return-loss measurement value expressed by a real number with the units of 1 dB.
When unmeasurement (**.*** is displayed on the CRT screen), 900.000 is returned.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[56] RTP/RTP? (Return-Loss Parameter)

■ Syntax

Command syntax : RTP␣ <set value 1>, ···, <set value 4>

Query syntax : RTP?

where

<Set value X> ::= A value corresponding to the parameter.
In the setting order, the value means as follows.

		Range	Unit
Set value 1 :	RSL value	0.005 to 9.999	dB/km
Set value 2 :	N1 value	1.0000 to 1.9999	None
Set value 3 :	N2 value	1.0000 to 1.9999	None
Set value 4 :	Ne value	1.0000 to 1.9999	None

RSL : Rayleigh Scattering Loss
N1 : Refractive index of the optical fiber's core
N2 : Refractive index of the optical fiber's clad
Ne : Group refractive index of the optical fiber

■ Example

- To set RSL: 0.35 dB/km, N1: 1.4666, N2: 1.4616, Ne: 1.4650

```
WRITE @101:"RTP 0.35, 1.4666, 1.4616, 1.4650"
```

- To read the parameter of return-loss measurement

```
10 DIM PARAMETER$*30
20 WRITE @101:"RTP?"
30 READ @101:PARAMETER$
40 PRINT PARAMETER$
50 END
```

■ Description

The RTP command sets the parameter of return-loss measurement.

The query returns the parameter of return-loss measurement.

When *.**** is displayed, the preveous value (before setting) is returned.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ **Response message format**

Return format : RTP␣ <set value 1>, ···, <set value 4> <LF | CR, LF>

Set value 1 ::= RSL
Set value 2 ::= N1
Set value 3 ::= N2
Set value 4 ::= Ne

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[57] SAV (Save)

■ Syntax

Command syntax : SAV␣ {<memory No.> | <file name>}

where

<memory No.> ::= Memory No. of INT MEMORY expressed by a number from 1 to 32

<file name> ::= MS-DOS format file name

■ Example

- To save data in the file name of "TRACE1" in INT PMC

```
10 WRITE @101:"MED 1"
20 WRITE @101:"SAV TRACE 1"
30 END
```

- To save data in INT MEMORY No. 7

```
10 WRITE @101:"MED 0"
20 WRITE @101:"SAV 7"
30 END
```

■ Description

The SAV command saves (stores) measured data in the specified media.

When the waveform data to be saved is less than five points, it cannot be saved. (In this case, an execution error occurs.)

Note: MS-DOS is a registered trade mark of Microsoft Corporation.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[58] SET (Setting)

■ Syntax

Command syntax : SET␣ {1 | 0}

where

0 : OFF
1 : ON

■ Example

- To turn setting on

```
WRITE @101:"SET 1"
```

- To read the setting set status

```
10 DIM STTING$*20
20 WRITE @101:"SET?"
30 READ @101:STTING$
40 PRINT STTING$
50 END
```

■ Description

The SET command displays the measurement conditions of two waveforms when the compare function is on. If the SET command is sent when the compare function is off, an execution error occurs.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[59] SMP? (Sampling Start/End/Resolution)

■ Syntax

Query syntax : SMP?

■ Example

° To read the sampling range

```
10 DIM SAMPLING$*50
20 WRITE @101:"SMP?"
30 READ @101:SAMPLING$
40 PRINT SAMPLING$
50 END
```

■ Description

The SMP? query returns the currently set values of sampling start, end, and resolution.

■ Response message format

Return format : SMP ␣ <sampling start>,<sampling end>,<sampling resolution> <LF | CR, LF>

where

<sampling start> ::= Distance of data at the beginning of the waveform data being sampled. Expressed by a value where IOR = 1.5 with length indicated in units of meter.

<sampling end> ::= Distance of data at the end of the waveform data being sampled. Expressed by a value where IOR = 1.5 with length indicated in units of meter.

<sampling resolution> ::= Resolution of the waveform data being sampled. Expressed by a value where IOR = 1.5 with length indicated in units of meter.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[60] SMT/SMT? (Smoothing)

■ Syntax

Command syntax : SMT␣{1 | 0}

Query syntax : SMT?

where

1 : ON
0 : OFF

■ Example

- To set the smoothing function to ON

```
WRITE @101:"SMT 1"
```

- To read the settings of the smoothing mode

```
10 DIM SMOOTH$*20
20 WRITE @101:"SMT?"
30 READ @101:SMOOTH$
40 PRINT SMOOTH$
50 END
```

■ Description

The SMT command reduces noise on the measured waveform by a arithmetic processing to display the S/N-improved waveform.

Recall waveforms cannot be smoothed.

During the processing, the commands except the SMT 0 (Smoothing OFF) and INI (initialization) cannot be received.

The query returns the current smoothing setting conditions.

■ Response message format

Return format : SMT␣{0 | 1} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[61] SMTL/SMTL? (Smoothing Level)

■ Syntax

Command syntax : SMTL␣ <Smoothing level>

Query syntax : SMTL?

where

<Smoothing level> ::= Number corresponding to smoothing level expressed by a number from 1 to 6.

■ Example

° To set the smoothing level to 3

```
WRITE @101:"SMTL 3"
```

° To read the setting of the smoothing level

```
10 DIM LEVEL$*20
20 WRITE @101:"SMTL?"
30 READ @101:LEVEL$
40 PRINT LEVEL$
50 END
```

■ Description

The SMTL command specifies the smoothing level.

When the smoothing processing is completed and then the smoothing level is changed in the smoothing mode, the smoothing processing restarts with the changed smoothing level.

The query returns the current smoothing-level setting condition.

■ Response message format

Return format : SMTL␣ <Smoothing level> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[62] SPL? (Splice Loss Measurement Data)

■ Syntax

Query syntax : SPL?

■ Example

- To read the SPLICE measurement results

```
10 DIM SPLICE$*20
20 WRITE @101:"SPL?"
30 READ @101:SPLICE$
40 PRINT SPLICE$
50 END
```

■ Description

The SPL? query reads the measured values of splice loss measurement. When the measurement item is not SPLICE, an execution error results.

When laser is on, the measurement results of splice loss are updated every sweep. Therefore, when reading the measurement results with the SPL? query, wait until one sweep is made after moving the marker.

When laser is off, the measurement results are updated each time the marker is moved. Therefore, you can read the measurement results immediately after moving the marker.

■ Response message format

Return format : SPL␣ <splice loss> <LF | CR, LF>

where

<splice loss> ::= Measured value of splice loss. When unmeasurable (* *. * * * is displayed on the CRT screen.), 900.000 is returned.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[63] SWP/SWP? (Sweep Mode)

■ Syntax

Command syntax : SWP␣ {1 | 0}

Query syntax : SWP?

where

1 : FAST mode
0 : NORMAL mode

■ Example

° To set the sweep speed to FAST mode

```
WRITE @101:"SWP 1"
```

° To read the setting status of sweep speed

```
10 DIM SWEEP$*20
20 WRITE @101:"SWP?"
30 READ @101:SWEEP$
40 PRINT SWEEP$
50 END
```

■ Description

The SWP command sets the sweep speed.

The query returns the setting state of the sweep speed.

■ Response message format

Return format : SWP␣ {1 | 0} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[64] TGT/TGT? (Target)

■ Syntax

Command syntax : TGT␣ <target>

Query syntax : TGT?

where

<target> ::= A value corresponding to each printer or plotter

0 : HP-GL (PL 1)

1 : GP-GL (PL 2)

50 : VP800, FP-850, CTM-800 (Epson, Japan) (PR 1)

51 : HP2225 (PR 2)

52 : MC8104 Data Storage Unit (DSU)

53 : Internal

■ Example

- To set the internal printer as the target

  (equipment to produce hard copy)

```
WRITE @101:"TGT 53"
```

- To read the settings of target

```
10 DIM TARGET$*20
20 WRITE @101:"TGT?"
30 READ @101:TARGET$
40 PRINT TARGET$
50 END
```

■ Description

The TGT command specifies the printer or plotter (i.e., target) to which data is output to produce hard copy.

The query returns information on which target is selected.

■ Response message format

Return format : TGT␣ <target> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[65] THR/THR? (Threshold)

■ Syntax

Command syntax : THR␣ <threshold value>

Query syntax : THR?

where

<threshold value> ::= Threshold value expressed in units of dB by a discrete number among 0.05, 0.10, 0.30, 1.00, 3.00, and 5.00

■ Example

- To set 1.0 dB as the threshold value

```
WRITE @101:"THR 1.0"
```

- To read the threshold settings

```
10 DIM THRESHOLD$*20
20 WRITE @101:"THR?"
30 READ @101:THRESHOLD$
40 PRINT THRESHOLD$
50 END
```

■ Description

The THR command sets the criterion (threshold value) to determine faulty points when making AUTO measurement. This setting is valid even when the measurement item is not AUTO.

The query returns the current threshold value.

■ Response message format

Return format : THR␣ <threshold value> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[66] TIM/TIM? (Time Out)

■ Syntax

Command syntax : TIM␣ <time-out time>

Query syntax : TIM?

where

<time-out time> ::= Set in units of seconds. The setting range is from 1 to 999 seconds.

■ Example

- To set 5 seconds as the time-out time

```
WRITE @101:"TIM 5"
```

- To read the settings of time-out time

```
10 DIM TIMEOUT$*20
20 WRITE @101:"TIM?"
30 READ @101:TIMEOUT$
40 PRINT TIMEOUT$
50 END
```

■ Description

The TIM command sets the time-out time.

The query returns the currently set time-out time. The time-out time is defaulted to 20 seconds.

■ Response message format

Return format : TIM␣ <time-out time> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[67] TIME/TIME? (Time)

■ Syntax

Command syntax : TIME␣ <hour>, <minute>

Query syntax : TIME?

where

<hour> ::= Hour data of the time. Specified by a number from 0 to 23 using the 24-hour system.

<minute> ::= Minute data of the time. Specified by a number from 0 to 59.

■ Example

- To set the built-in clock to 3:25 p.m.

```
WRITE @101:"TIME 15,25"
```

- To read the time.

```
10 DIM TIME$*20
20 WRITE @101:"TIME?"
30 READ @101:TIME$
40 PRINT TIME$
50 END
```

■ Description

The TIME command sets the time of the built-in clock.

The query returns the time.

■ Response message format

Return format : TIME␣ <hour>, <minute> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[68] TIT/TIT? (Title)

■ Syntax

Command syntax : TIT␣ <title>

Query syntax : TIT?

where

<title> ::= Character string with equal to or less then 40 characters. The same characters as those of the local state are used except single quotation mark (' ') and a sharp mark (#).

■ Example

- To set title "TOKYO_OSAKA"

```
WRITE @101:"TIT TOKYO_OSAKA"
```

- To delete the title

```
WRITE @101:"TIT"
```

- To read the title

```
10 DIM TITLE$*50
20 WRITE @101:"TIT?"
30 READ @101:TITLE$
40 PRINT TITLE$
50 END
```

■ Description

The TIT command changes the title. When the command is not accompanied by an argument, it deletes the title. When the title thus set consists of less than 40 characters, the title is filled up from the top and the remainder is filled with blank spaces. Conversely, if the title consists of more than 40 characters, the characters in up to the 40th place are set as valid and the rest is ignored.

If any invalid character is included in the parameter, an execution error results. Invalid characters included in the ignored portion of the title (in places over the 40th character) do not have effect.

The query returns the current title. If the title has been deleted, a response consisting of 40 blank spaces is returned.

The TIT command does not discriminate between upper-case and lower-case characters. The title is set entirely with upper-case characters even when received in lower-case characters. Conversely, the title set in lower-case characters is output in upper-case characters.

Spaces in front of the title are ignored.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ **Response message format**

Return format : TIT␣ <title> <LF | CR, LF>

Note: The syntax of the TIT command does not conform to the IEEE 488.2 standard.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[69] TRM/TRM? (Terminator)

■ Syntax

Command syntax : TRM␣ {1 | 0}

Query syntax : TRM?

where

0 : LF
1 : CR, LF

■ Example

° To set "LF" as the terminator

```
WRITE @101:"TRM 0"
```

° To read the terminator settings

```
10 DIM TERMINATOR$*20
20 WRITE @101:"TRM?"
30 READ @101:TERMINATOR$
40 PRINT TERMINATOR$
50 END
```

■ Description

The TRM command specifies the query terminator marker.

The query returns information on which terminator is currently selected.

The terminator is defaulted to "LF".

■ Response message format

Return format : TRM␣ {1 | 0} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[70] UNL/UNL? (Unit Of Length)

■ Syntax

Command syntax : UNL␣ {0 | 1 | 2}

Query syntax : UNL?

where

0 : Meter
1 : Foot
2 : Mile

■ Example

° To set MILE as the unit of distance displayed on screen

```
WRITE @101:"UNL 2"
```

° To read the settings for the unit of distance displayed on screen

```
10 DIM UNL$*20
20 WRITE @101:"UNL?"
30 READ @101:UNL$
40 PRINT UNL$
50 END
```

■ Description

The UNL command changes the unit of length shown on screen. It does not affect the unit of wavelength (μm), however. The unit of length handled by GPIB is always METER and is not affected by the UNL command. The MW9060A uses the following equations to convert FOOT and MILE to and from METER.

1 mile = 1609.3440 meters

1 foot = 0.3048 meter

The query returns the current unit of length.

■ Response message format

Return format : UNL␣ {0 | 1 | 2} <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[71] VSC/VSC? (Vertical Scale)

■ Syntax

Command syntax : VSC␣ <vertical scale value>

Query syntax : VSC?

where

<vertical scale value> ::= Value of vertical scale expressed in units of dB by a discrete number among 0.1, 0.25, 0.5, 1.0, 2.5, and 5.0.

■ Example

° To set 1 dB/div as the vertical scale

```
WRITE @101:"VSC 1"
```

° To read the set value of vertical scale

```
10 DIM V_SCALE$*20
20 WRITE @101:"VSC?"
30 READ @101:V_SCALE$
40 PRINT V_SCALE$
50 END
```

■ Description

The VSC command changes the vertical scale value (amount of one scale division (dB/div) in the vertical direction relative to waveform screen).

The query returns the vertical scale value.

■ Response message format

Return format : VSC␣ <vertical scale value> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[72] VSF/VSF? (Vertical Shift)

■ Syntax

Command syntax : VSF␣ <vertical shift value>

Query syntax : VSF?

where

<vertical shift value> ::= Amount of vertical shift

■ Example

° To set 5 dB as the value of vertical shift

```
WRITE @101:"VSF 5"
```

° To read the set value of vertical shift

```
10 DIM V_SHIFT$*20
20 WRITE @101:"VSF?"
30 READ @101:V_SHIFT$
40 PRINT V_SHIFT$
50 END
```

■ Description

The VSF command changes the vertical shift value (amount of movement in the vertical direction relative to waveform screen). The range of vertical shift values is from 0.000 to 50.000 dB. However, the upper limit is a value obtained by subracting (vertical scale × 8) from 50 dB. When the rotary knob input item is other than vertical shift, it is changed to vertical shift. When an error occurs, the rotary knob input item does not change.

The query returns the vertical shift value.

■ Response message format

Return format : VSF␣ <vertical shift value> <LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[73] WLS/WLS? (Wavelength Select : λ Select)

■ Syntax

Command syntax : WLS␣ <wavelength>

Query syntax : WLS? {[0] | 1}

where

<wavelength> ::= Wavelength expressed by a value in units of μm

0 : Requests the current wavelength. If omitted, 0 is assumed.

1 : Requests all wavelengths of plug-in unit.

■ Example

° To set the wavelength to 1.55 μm for a plug-in unit which has wavelengths of 1.31 μm and 1.55 μm

```
WRITE @101:"WLS 1.55"
```

° To read the currently used wavelength

```
10 DIM WAVELENGTH$*20
20 WRITE @101:"WLS?"
30 READ @101:WAVELENGTH$
40 PRINT WAVELENGTH$
50 END
```

° To read the wavelengths that the currently used plug-in unit has

```
10 DIM WAVELENGTH$*50
20 WRITE @101:"WLS? 1"
30 READ @101:WAVELENGTH$
40 PRINT WAVELENGTH$
50 END
```

■ Description

The WLS command selects between the wavelengths of the plug-in unit when the fitted unit is a switchable unit (which incorporates two wavelengths).

The query returns the wavelength value. If the parameter is 0 or omitted, the current wavelength is returned. If the parameter is 1, the wavelengths that the currently used plug-in unit has are returned.

The wavelength value is not affected by the UNL or IOR command.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ Response message format

(1) When current wavelength is returned

Return format : WLS␣ <wavelength> <LF | CR, LF>

(2) When plug-in unit's wavelengths are returned

Return format : WLS␣ <number of wavelength> ,<wavelength> ,...,
<wavelength>,<LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

[74] ZADJ/ZADJ? (Horizontal Zero Adjust)

■ Syntax

Command syntax : ZADJ␣<distance>

Query syntax : ZADJ?

where

<distance> ::= Distance is specified temporarily as 0Km. Distance units are meters; specify the value using 1 = 1m.

■ Example

- To set 100m to 0Km

```
WRITE @101:"ZADJ 100"
```

- To read the 0Km position

```
10 DIM VRZERO$*20
20 WRITE @101:"ZADJ?"
30 READ @101:VRZERO$
40 PRINT VRZERO$
50 END
```

■ Description

The ZADJ command changes the 0Km position of the horizontal scale. The setting range is within the sampling range.

The setting resolution is the minimum resolution of the installed unit.

The query returns the current 0Km position distance.

■ Response message format

Return format : ZADJ␣ <distance><LF | CR, LF>

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

(Blank)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

SECTION 10
PROGRAM EXAMPLES

10. 1 Precautions on Creating GPIB Control Programs

Observe the following precautions when creating GPIB control programs:

1. Initialize the device.

The devices used for measurement are not necessarily in an appropriate state when you actually begin measurement because their settings may have been changed by panel operation on the device itself or execution of other programs. Therefore, be sure to initialize each device before using it. For the MW9060A, execute the following commands to initialize it.

① Interface clear (IFC)

② Device clear (DCL) or selective device clear (SDC)

③ INI or *RST

2. Place the device in a local lockout state.

When the [LOCAL] key is accidentally pressed while in a ordinary remote mode, the device is set into the local mode. If a panel key is pressed in this state, the device's automatic measurement may not function correctly and the reliability of measured data is lost. Therefore, place the device in a local lockout state to prevent it from being accidentally set into the local mode.

3. Always read the response after sending a query message.

After sending a query message to the device, always read the response for it before sending any other command.

Otherwise, the response message may be lost.

4. Avoid protocol exception processing.

The above case 3. is one of protocol exception processing. Other exception processing cases are described in the last part of Section 6 of this manual. Read this part carefully and avoid causing exception processing. Also, insert exception handling routines in your program to avoid stopping program execution for exception handling.

5. Confirm the interface functions of each device.

If the device does not have the required subset, it cannot execute the program. Therefore, be sure to confirm the subset of each device before executing the program.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

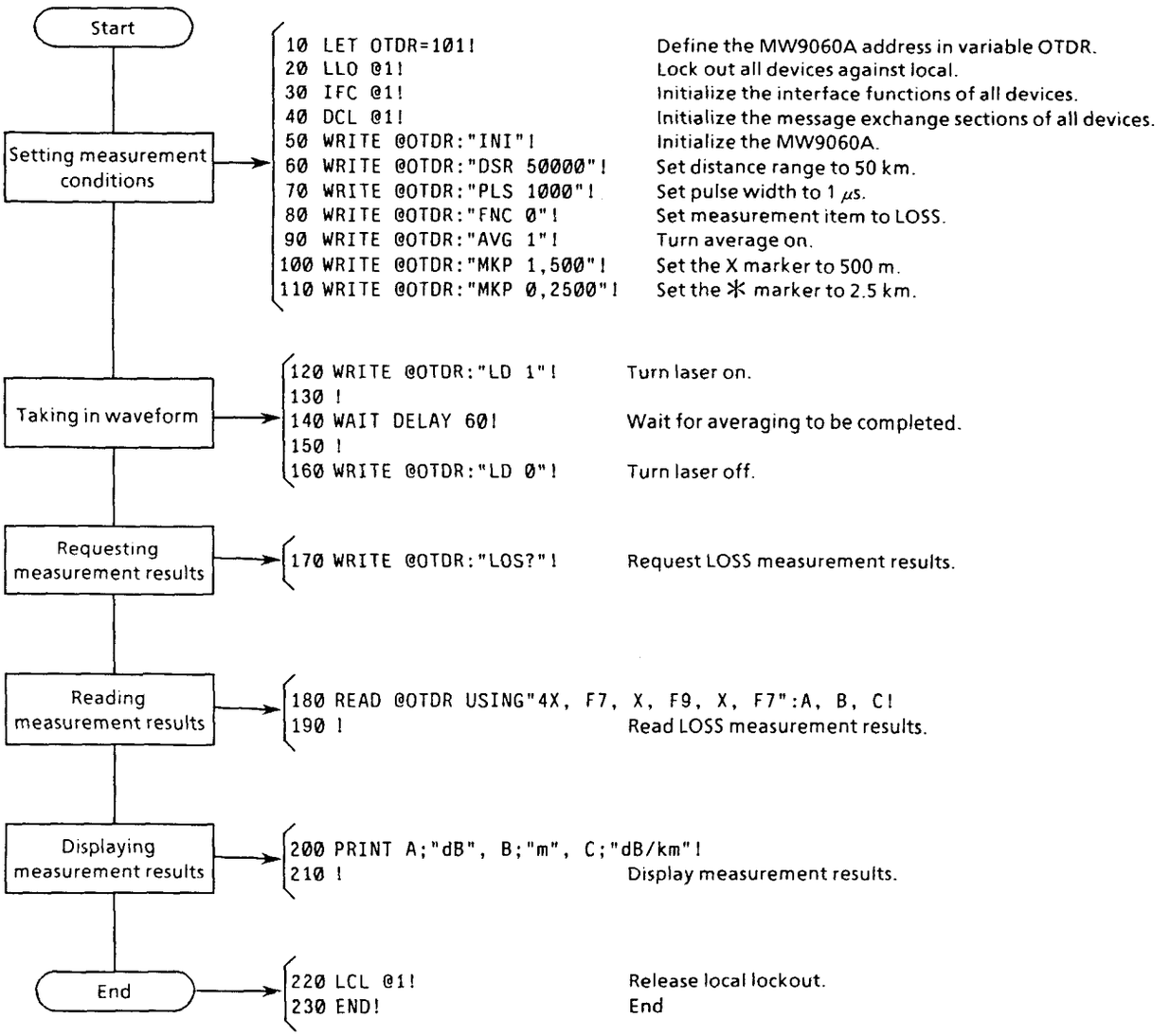
10.2 Basic Programming

10.2.1 Measuring Loss between Two Arbitrary Points

■ **Description**

Measure loss (LOSS) between two points.

■ **Flowchart and sample program**



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

10.2.2 Measuring Loss during Sweep

■ Description

When performing waveform sweep while laser is on, measurement results are not immediately updated by moving the marker. Measurement results are updated every waveform sweep end. Therefore, when laser is on, confirm that waveform sweep is completed after moving the marker before reading measurement results.

Here, as a method to confirm that waveform sweep is completed, we check the DPT bit (bit 0) of the termination event status register which is one of extended event status registers.

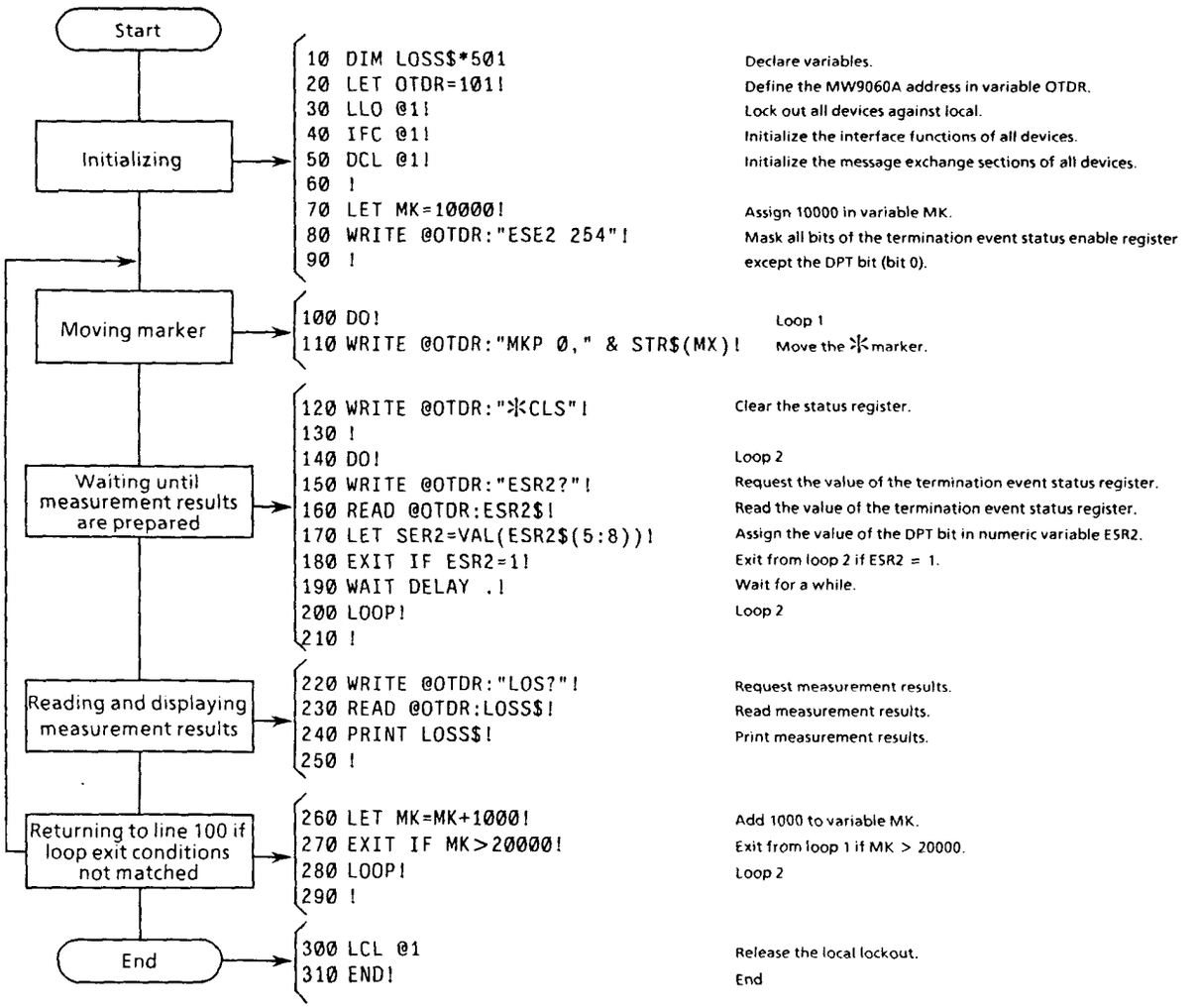
The wait on line 190 is provided to prevent the internal measurement processing of the MW9060A from being stopped by continuous execution of line 150 and line 160.

The sample program below measures loss while moving *marker from 10 km to 20 km in 1-km spacing with laser-on.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

■ Flowchart and sample program

It is assumed here that the device is in a sweep state with laser on.




```
10  DIM LOSS$*501                    Declare variables.
20  LET OTDR=101!                    Define the MW9060A address in variable OTDR.
30  LLO @1!                          Lock out all devices against local.
40  IFC @1!                          Initialize the interface functions of all devices.
50  DCL @1!                          Initialize the message exchange sections of all devices.
60  !
70  LET MK=10000!                    Assign 10000 in variable MK.
80  WRITE @OTDR:"ESE2 254"!          Mask all bits of the termination event status enable register
90  !                                except the DPT bit (bit 0).
100 DO!                              Loop 1
110 WRITE @OTDR:"MKP 0," & STR$(MX)! Move the ✳ marker.
120 WRITE @OTDR:"✳CLS"!              Clear the status register.
130 !
140 DO!                              Loop 2
150 WRITE @OTDR:"ESR2?"!             Request the value of the termination event status register.
160 READ @OTDR:ESR2$!                Read the value of the termination event status register.
170 LET SER2=VAL(ESR2$(5:8))!        Assign the value of the DPT bit in numeric variable ESR2.
180 EXIT IF ESR2=1!                  Exit from loop 2 if ESR2 = 1.
190 WAIT DELAY .1                    Wait for a while.
200 LOOP!                            Loop 2
210 !
220 WRITE @OTDR:"LOS?"!              Request measurement results.
230 READ @OTDR:LOSS$!                Read measurement results.
240 PRINT LOSS$!                     Print measurement results.
250 !
260 LET MK=MK+1000!                  Add 1000 to variable MK.
270 EXIT IF MK>20000!                Exit from loop 1 if MK > 20000.
280 LOOP!                            Loop 2
290 !
300 LCL @1                           Release the local lockout.
310 END!                             End
```

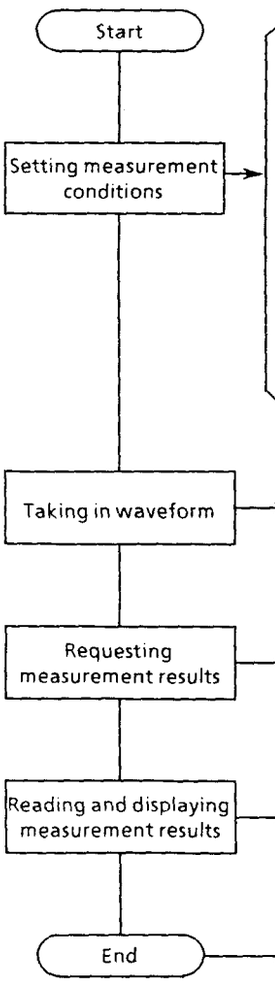

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

10.2.3 Measuring Splice Loss

■ Description

Measure splice loss (SPLICE).

■ Flowchart and sample program




```
10 LET OTDR=101!                    Define the MW9060A address in variable OTDR.
20 LLO @1!                          Lock out all devices against local.
30 IFC @1!                          Initialize the interface functions of all devices.
40 DCL @1!                          Initialize the message exchange sections of all devices.
50 WRITE @OTDR:"INI"!               Initialize the MW9060A.
60 WRITE @OTDR:"DSR 50000"!         Set distance range to 50 km.
70 WRITE @OTDR:"PLS 1000"!          Set pulse width to 1 μs.
80 WRITE @OTDR:"FNC 1"!             Set measurement item to SPLICE.
90 WRITE @OTDR:"AVG 1"!             Turn average on.
100 WRITE @OTDR:"MKP 0,2500"!       Set the ✳ marker to 2.5 km.
110 WRITE @OTDR:"MKP 1,500"!        Set the X1 marker to 500 m.
120 WRITE @OTDR:"MKP 2,2000"!       Set the X2 marker to 2 km.
130 WRITE @OTDR:"MKP 3,3000"!       Set the X3 marker to 3 km.
140 WRITE @OTDR:"MKP 4,4500"!       Set the X4 marker to 4.5 km.

150 WRITE @OTDR:"LD 1"!             Turn laser on.
160 !
170 WAIT DELAY 60!                  Wait for averaging to be completed..
180 !
190 WRITE @OTDR:"LD 0"!             Turn laser off.

200 WRITE @OTDR:"SPL?"!             Request SPLICE measurement results.

210 READ @OTDR:USING "4X, F7":SPLICE!   Read SPLICE measurement results.
220 !
230 PRINT SPLICE;"dB"!                  Display measurement results.
240 !

250 LCL @1!                         Release local lockout.
260 END!                            End
```

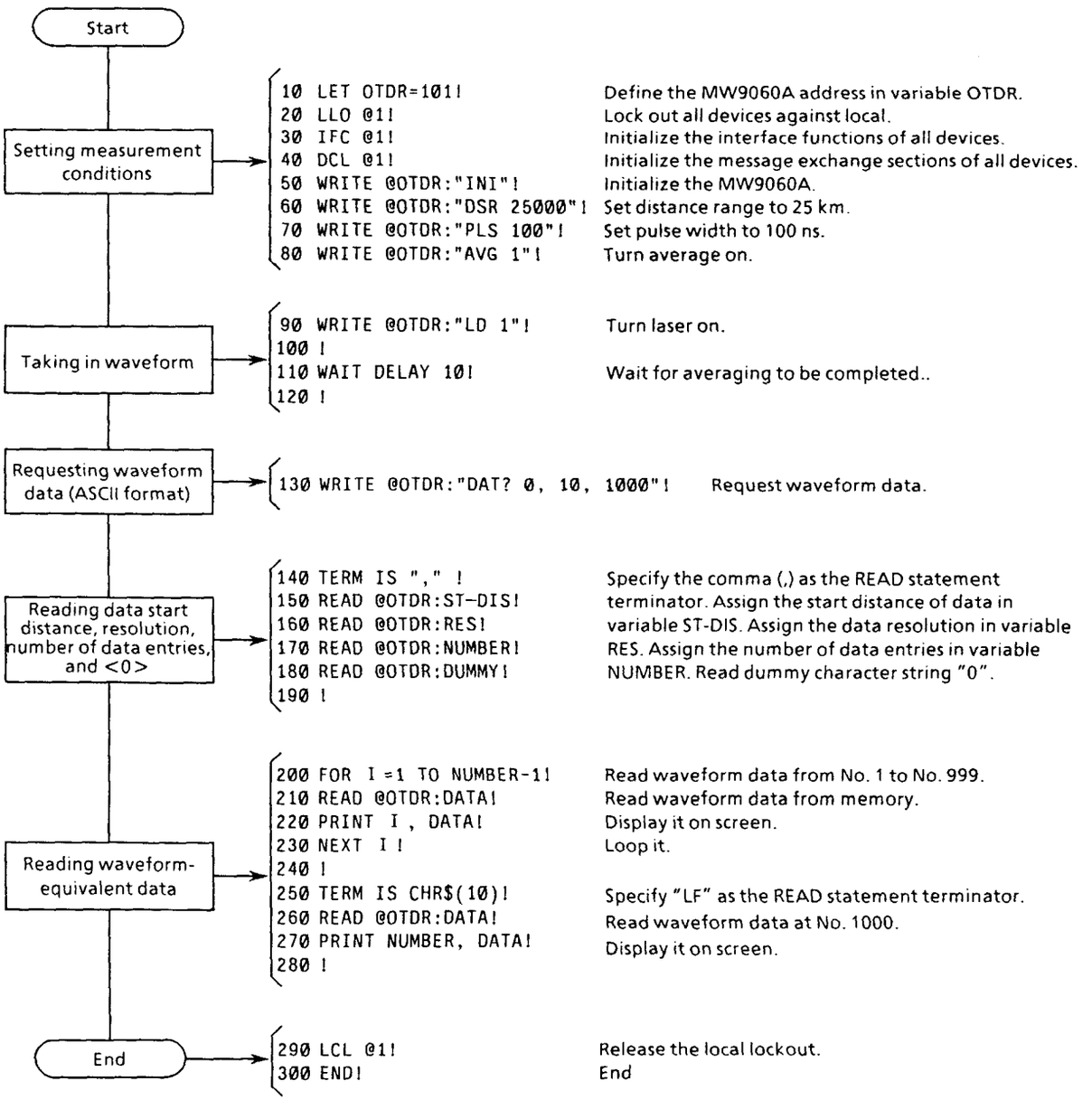

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

10.2.4 Reading Waveform Data (ASCII Format)

■ Description

Read 1000-point waveform data beginning from 0 km with a 10 m resolution.

■ Flowchart and sample program



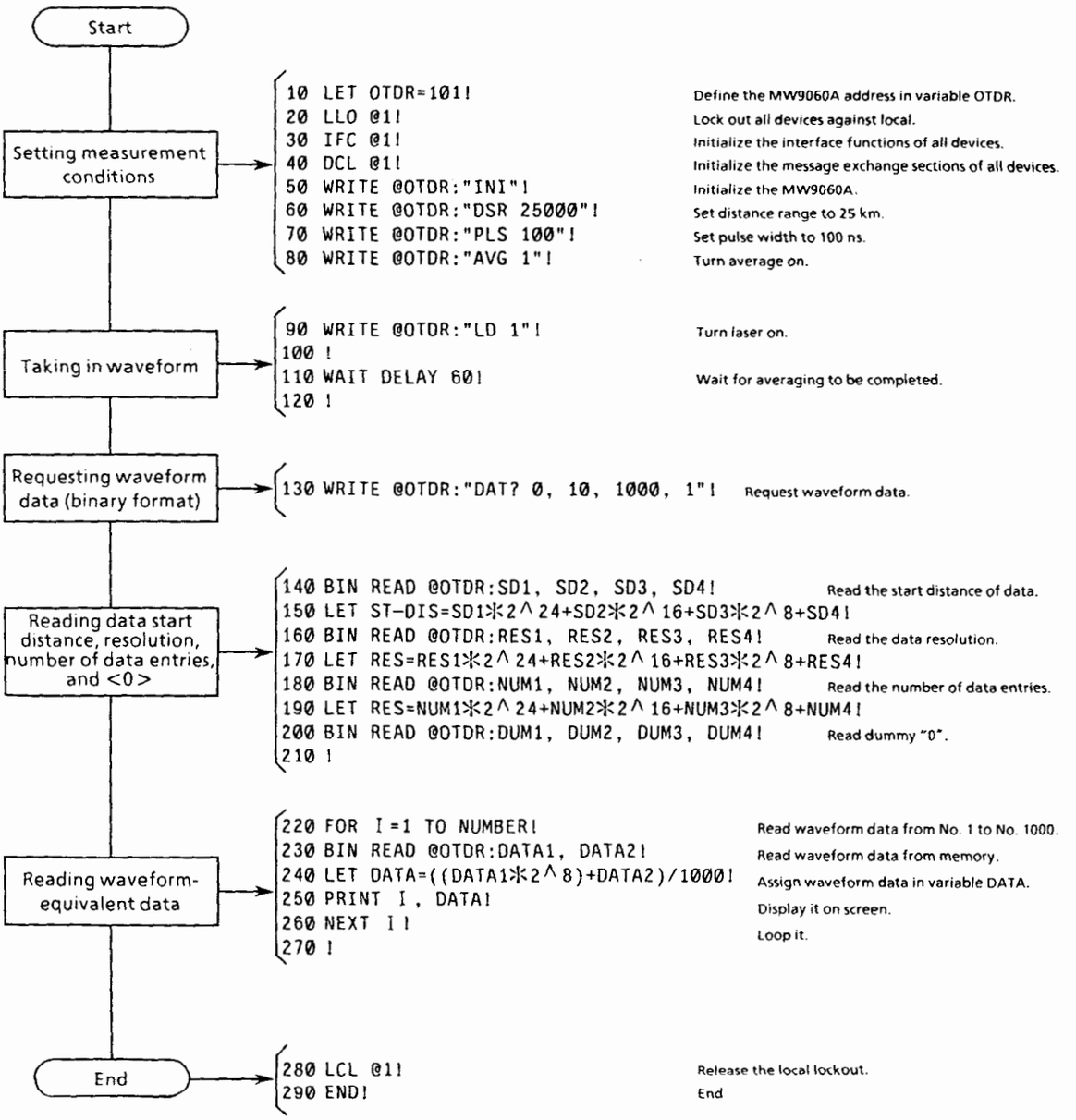
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

10.2.5 Reading Waveform Data (Binary Format)

■ Description

Data in the binary format can be read with a shorter transfer time than in the ASCII format. Read 1000-point waveform data beginning from 0 km with a 10 m resolution.

■ Flowchart and sample program



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

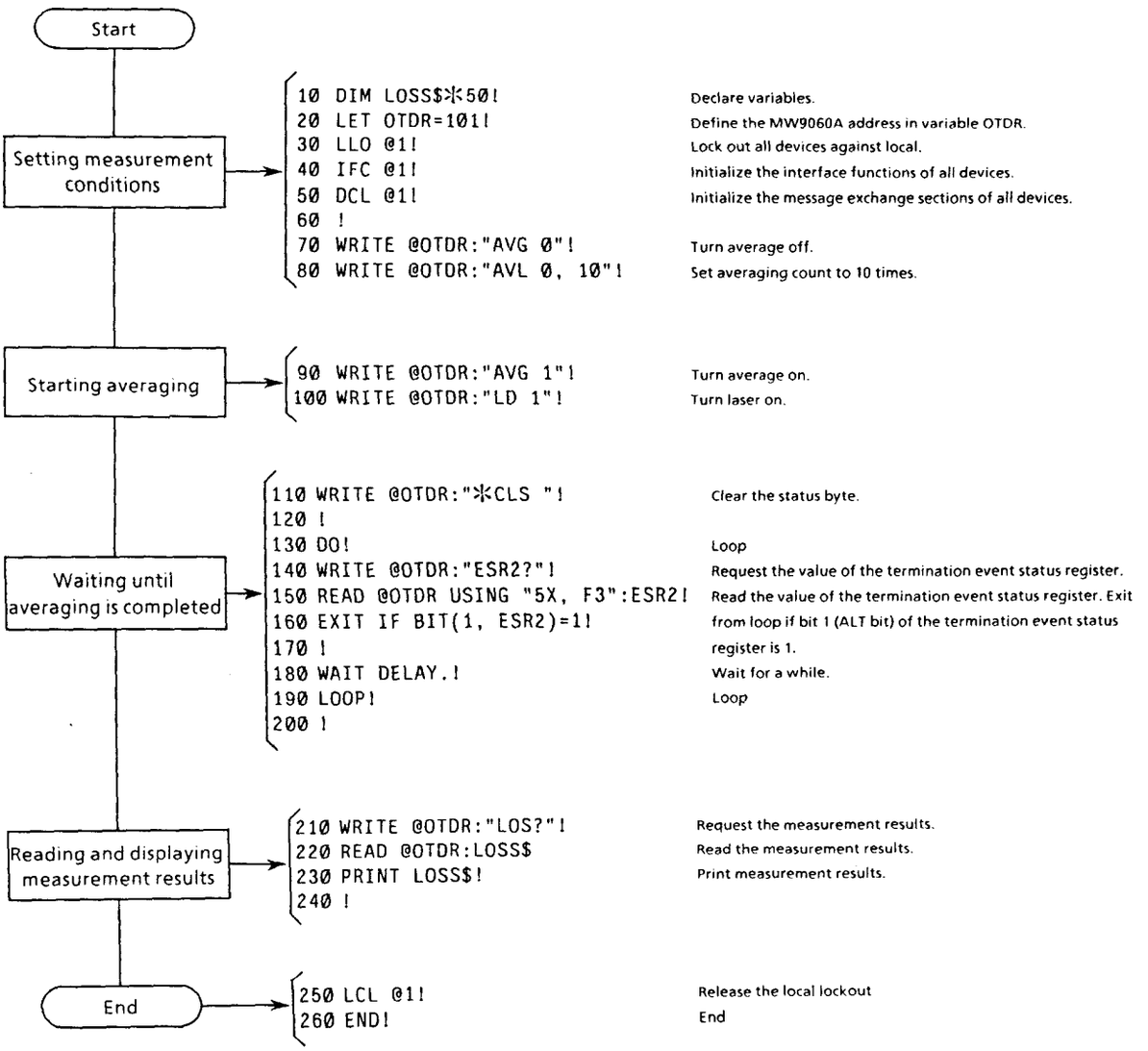
10.2.6 Wait for Average Termination

■ Description

This program specifies an averaging count (number of samples to be averaged) and reads measurement results when the specified averaging count is reached. In this example, whether the specified averaging count is reached is determined by checking the ALT (Average Limit Termination) bit of the termination event status register which is one of extended event status registers.

The wait on line 180 is provided to prevent the internal measurement processing of the MW9060A from being stopped by continuous execution of line 140 and line 150.

■ Flowchart and sample program



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

APPENDIX A
COMPARISON OF MESSAGES BETWEEN MW9060A AND MW910C

Table A-1 compares the GPIB messages used by the MW9060A and MW910C. Table A-2 compares the measurement result read commands between the MW9060A and MW910C. For details on messages used by the MW910C, refer to its user's manual.

Table A-1 GPIB Message Comparison

No.	Function		Message		Remarks
			MW910C	MW9060A	
1	Distance range	Command	DSRm	DSR␣<distance range>	
		Query	DSRR	DSR?	
		Response	m	DSR␣<distance range>	
2	Horizontal scale	Command	HSCm	HSC␣<horizontal scale	
		Query	HSCR	HSC?	
		Response	m	HSC␣<horizontal scale	
3	Pulse width	Command	PLSm	PLS␣<pulse width>	
		Query	PLSR	PLS?	
		Response	m	PLS␣<pulse width>	
4	Horizontal shift (1)	Command	HSFℓ	HSF␣<horizontal shift value>	
		Query	HSFR	HSF?	
		Response	ℓ	HSF␣<horizontal shift value>	
5	Horizontal shift (2)	Command	HIFℓ	———	
		Query	HIFR	———	
		Response	ℓ	———	
6	MASK (1)	Command	MSKm : ℓ	MSP␣<mask No.>, <distance>	
		Query	MSKmR	MSP?␣<mask No.>	
		Response	ℓ	MSP␣<distance>	
7	MASK (2)	Command	MIKmℓ	———	
		Query	MIKmR	———	
		Response	ℓ	———	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table A-1 GPIB Message Comparison (Continued)

No.	Function		Message		Remarks
			MW910C	MW9060A	
8	Average	Command	AVRn	AVG␣{0|1}	
		Query	AVRR	AVG?	
		Response	n	AVG␣{0|1}	
9	Function	Command	FNCn	FNC␣<measurement item>	
		Query	FNCR	FNC?	
		Response	n	FNC␣<measurement item>	
10	Approx method	Command	APRn	APR␣<linear approximation>	
		Query	APRR	APR?	
		Response	n	APR␣<linear approximation>	
11	Hold	Command	HLDn	————	
		Query	HLDR	————	
		Response	n	————	
12	Marker select	Command	MKSm	MKS␣<marker No.>	
		Query	MKSR	MKS?	
		Response	m	MKS␣<marker No.>	
13	Marker initialize	Command	MKIl	————	
		Query	————	————	
		Response	————	————	
14	Marker shift (1)	Command	MSmℓ	MKP␣ <marker No.>, <distance>	
		Query	MSmR	MKP?<marker No.>	
		Response	ℓ	MKP␣<distance>	
15	Marker shift (2)	Command	MImℓ	————	
		Query	MImR	————	
		Response	ℓ	————	
16	Vertical scale	Command	VSCm	VSC␣<vertical scale value>	
		Query	VSCR	VSC?	
		Response	m	VSC␣<vertical scale value>	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table A-1 GPIB Message Comparison (Continued)

No.	Function		Message		Remarks
			MW910C	MW9060A	
17	Vertical shift	Command	VSFℓ	VSF␣<vertical shift value>	
		Query	VSFR	VSF?	
		Response	ℓ	VSF␣<vertical shift value>	
18	Laser	Command	LDRn	LD␣{0|1}	
		Query	LDRR	LD?	
		Response	n	LD␣{0|1}	
19	Initialize	Command	IST1	INI	
		Query	———	———	
		Response	———	———	
20	Attenuator	Command	ATTℓ	ATT␣<attenuator value>	
		Query	ATTR	ATT?	
		Response	ℓ	ATT␣<attenuator value>	
21	Attenuator auto	Command	ATAn	ATA␣{0|1}	
		Query	ATAR	ATA?	
		Response	n	ATA␣{0|1}	
22	Averaging number	Command	AVHℓ	AVL␣<NT>, <NS>	
		Query	———	AVL?	
		Response	———	AVL␣<NT>, <NS>, <TS>, <NTC>	
23	Mask clear	Command	MSC1	———	
		Query	———	———	
		Response	———	———	
24	IOR	Command	IORℓ	IOR␣<IOR value>	
		Query	IORR	IOR?	
		Response	ℓ	IOR␣<IOR value>	
25	Memory save	Command	MESm	SAV␣<memory No.>	
		Query	———	———	
		Response	———	———	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table A-1 GPIB Message Comparison (Continued)

No.	Function		Message		Remarks
			MW910C	MW9060A	
26	Memory recall	Command	MERm	RCL␣<memory No.>	
		Query	———	RCL?	
		Response	———	RCL␣<memory No.>	
27	Title	Command	TTL displayed character	TIT␣<title>	
		Query	TTLR	TIT?	
		Response	Displayed character	TIT␣<title>	
28	Date/Time	Command	TDEm0-m1-m2-m3-m4	DATE␣ <year>, <month>, <day> TIME␣ <hour>, <minute>	
		Query	TDER	DATE? TIME?	
		Response	m0-m1-m2-m3-m4	DATE␣ <year>, <month>, <day> TIME␣ <hour>, <minute>	
29	Meter/Feet	Command	MFTm	UNL␣<unit>	
		Query	MFTR	UNL?	
		Response	m	UNL␣<unit>	
30	Trace	Command	TRCn	———	
		Query	TRCR	———	
		Response	n	———	
31	LED Light source	Command	LEDn	———	
		Query	LEDR	———	
		Response	n	———	
32	Print	Command	PRIR	CPY␣{0|1}	
		Query	———	CPY?	
		Response	———	CPY␣{0|1}	
33	Feed	Command	FEDℓ	———	
		Query	———	———	
		Response	———	———	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table A-1 GPIB Message Comparison (Continued)

No.	Function		Message		Remarks
			MW910C	MW9060A	
34	Plug-in unit	Command	————	————	
		Query	UNTR	PLG?	
		Response	Type of unit	PLG␣<name of plug-in unit>	
35	Auto/Manual	Command	AUTn	————	
		Query	AUTR	————	
		Response	n	————	
36	λ-Select	Command	WAVn	WLS␣<wavelength>	
		Query	WAVR	WLS?[␣0]	
		Response	n	WLS␣<wavelength>	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table A-2 GPIB Message (Measurement Result Read) Comparison

No.	Function		Message		Remarks
			MW910C	MW9060A	
1	Loss data	Command	———	———	
		Query	LSDR	LOS? SPL?	
		Response	ℓ1, ℓ2, ℓ3, ℓ4	LOS␣ <loss between 2 points>, <distance between 2 points>, <transmission loss> SPL␣ <splice loss>	
2	Measuring data (1)	Command	———	———	
		Query	DPDRℓ	DAT?␣<START>?<RES>, <NUMBER> [,<TYPE>]	
		Response	ℓ1, ℓ2	<START>, <RES>, <NUMBER>, 0, <1st data, ····, <n'th data>	
3	Measuring data (2)	Command	———	———	
		Query	DPIRℓ	———	
		Response	ℓ1, ℓ2	———	
4	Display data	Command	———	———	
		Query	DADR	———	
		Response	x, y	———	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP