

SCPI Programming Manual

MG369xC Series Synthesized Signal Generators

The Anritsu logo is located in the bottom right corner of the page. It consists of the word "Anritsu" in a bold, sans-serif font. The letter "A" is stylized with a diagonal slash through it.

Table of Contents

Chapter 1—General GPIB Information

1-1	Introduction	1-1
1-2	Scope of Manual	1-1
1-3	Related Manuals	1-1
1-4	Chapter Introduction	1-2
1-5	IEEE-488 Interface Bus Description	1-2
	Functional Elements	1-4
	Bus Structure	1-4
	Data Bus Description	1-5
	Data Byte Transfer Control Bus Description	1-5
	General Interface Management Bus Description	1-6
	Device Interface Function Capability	1-6
	Message Types	1-7
1-6	MG369xC GPIB Operation	1-9
	Setting GPIB Operating Parameters	1-9
	Selecting the Interface Language	1-9
	Response to GPIB Interface Function Messages	1-9
1-7	Remote Programming Setup and Interface	1-10
	Interface Language	1-10
	VISA (Virtual Instrument System Architecture)	1-10
	Interface Connectors	1-10
1-8	GPIB Interconnection and Setup	1-11
	GPIB Interconnection	1-11
	GPIB Cable Length Restrictions	1-11
	Setting the GPIB Address	1-11
	Selecting the Line Terminator	1-12
1-9	Ethernet LAN TCP/IP Requirements and Setup	1-13
	TCP/IP General Requirements and Settings	1-13
	Ethernet Setup - Static IP Configuration	1-14
	Ethernet Setup - DHCP with AutoIP Configuration	1-17
	Testing the TCP/IP Connection Over LAN	1-19
	Example PC Control of MG369xC via LAN	1-20
	Ethernet Control Example in C# Programming using NI VISA	1-22
	Resetting Ethernet IP Address to Factory Default	1-24

Table of Contents (Continued)

Chapter 2—Programming with SCPI Commands

2-1	Introduction	2-1
2-2	Introduction to SCPI Programming	2-1
	SCPI Command Types	2-1
	Common Commands	2-1
	Required and Optional SCPI Commands	2-2
	Query Commands	2-2
	Command Names	2-2
	Hierarchical Command Structure	2-3
	Data Parameters	2-4
	Unit Suffixes	2-4
2-3	Notational Conventions	2-5
	General Notations	2-5
	Parameter Notations	2-5
	Notational Examples	2-6
2-4	SCPI Interface Language Selection	2-6
	Front Panel Selection	2-6
	Remote Selection	2-6
2-5	Status System Programming	2-7
	Status Group Registers	2-7
	Status Group Reporting	2-9
2-6	Trigger System Programming	2-12
	Trigger System Operation	2-12

Chapter 3—Programming Commands

3-1	Introduction	3-1
3-2	Common Commands	3-1
	IEEE 488.2 Mandated Commands	3-1
	Optional Common Commands	3-2
3-3	Subsystem Commands	3-3
3-4	:ABORt Subsystem	3-4
3-5	:CONTRol Subsystem	3-5
3-6	:DIAGnostic Subsystem	3-7
3-7	:DISPlay Subsystem	3-8
3-8	:INITiate Subsystem	3-9
3-9	:OUTPut Subsystem	3-10

Table of Contents (Continued)

3-10	[:SOURce] Subsystems	3-12
	[:SOURce]:AM Commands	3-16
	[:SOURce]:CORRection Commands	3-19
	[:SOURce]:FM Commands	3-19
	[:SOURce]:FREQuency Commands	3-23
	[:SOURce]:LIST Commands	3-29
	[:SOURce]:MARKer Commands	3-34
	[:SOURce]:PM Commands	3-37
	[:SOURce]:POWer Commands	3-40
	[:SOURce]:PULM Commands	3-49
	[:SOURce]:PULSe Commands	3-51
	[:SOURce]:ROSCillator Commands	3-55
	[:SOURce]:SCAN Commands	3-56
	[:SOURce]:SWEep Commands	3-56
3-11	:STATus Subsystem	3-65
3-12	:SYSTem Subsystem	3-70
3-13	:TRIGger Subsystem	3-72
3-14	:TSWEEP Command	3-74
3-15	:UNIT Subsystem	3-75

Chapter 4—Error Messages

4-1	Introduction	4-1
4-2	Error Query	4-1
4-3	Error Queue	4-1
4-4	Error Codes	4-2
4-5	No Error	4-2
4-6	Command Errors	4-3
4-7	Execution Errors	4-6
4-8	Device-Specific Errors	4-10
4-9	Query Errors	4-11
4-10	Parser Errors	4-12
4-11	Self-Test Errors	4-13

Appendix A—Overall Command Tree

A-1	Introduction	A-1
-----	--------------------	-----

Appendix B—SCPI Conformance Information

B-1	Introduction	B-1
-----	--------------------	-----

Index

Chapter 1 — General GPIB Information

1-1 Introduction

This manual provides information for remote operation of the Series MG369xC Synthesized High Performance Signal Generator using commands sent from an external controller via the IEEE-488 General Purpose Interface Bus (GPIB).

1-2 Scope of Manual

This manual includes the following:

- A general description of the GPIB and the bus data transfer and control functions
- A listing of the IEEE-488 Interface Function Messages recognized by the signal generator with a description of its response
- A complete listing and description of all the Standard Commands for Programmable Instruments (SCPI) commands that can be used to control signal generator operation with examples of command usage
- Error messages related to the MG369xC operation

This manual is intended to be used in conjunction with the Series MG369xC Synthesized Signal Generator Operation Manual, P/N: 10370-10373. Refer to that manual for general information about the MG369xC, including equipment set up and front panel (manual mode) operating instructions.

Electronic Manual

This manual is available on CD ROM as an Adobe Acrobat Portable Document Format (.pdf) file. The file can be viewed using Acrobat Reader, a free program that is also included on the CD ROM. The file is “linked” such that the viewer can choose a topic to view from the displayed “bookmark” list and “jump” to the manual page on which the topic resides. The text can also be word-searched. Contact Anritsu Customer Service for price and availability.

1-3 Related Manuals

This is one of a four manual set that consists of an operation manual, a GPIB programming manual, a SCPI programming manual, and a maintenance manual.

GPIB Programming Manual

The *Series MG369xC Synthesized Signal Generator GPIB Programming Manual* provides information for remote operation of the signal generator with product specific commands sent from an external controller via the IEEE 488 General Purpose Interface Bus (GPIB). It contains a general description of the GPIB and bus data transfer and control functions, a complete listing and description of all MG369xC GPIB product specific commands, and several programming examples. The Anritsu part number for the GPIB programming manual is 10370-10374.

Operation Manual

The *Series MG369xC Synthesized Signal Generator Operation Manual* provides general information, installation, and operating information for the Anritsu series MG369xC synthesized signal generator. The Anritsu part number for the operation manual is 10370-10373.

Maintenance Manual

The Series MG369xC Synthesized Signal Generator Maintenance Manual supplies service information for all models in the MG369xC series. The service information includes functional circuit descriptions, block diagrams, performance verification tests, calibration procedures, troubleshooting data, and assembly and component removal/replacement procedures. The Anritsu part number for the maintenance manual is 10370-10376.

1-4 Chapter Introduction

This chapter provides a general description of the GPIB and the bus data transfer and control functions. It also contains a listing of the MG369xC's GPIB interface function subset capability and response to IEEE-488 interface function messages.

The GPIB information presented in this chapter is general in nature. For complete and specific information, refer to the following documents: ANSI/IEEE Std 488.1-1987 *IEEE Standard Digital Interface for Programmable Instrumentation* and ANSI/IEEE Std 488.2-1987 *IEEE Standard Codes, Formats, Protocols and Common Commands*. These documents precisely define the total specification of the mechanical and electrical interface, and of the data transfer and control protocols.

1-5 IEEE-488 Interface Bus Description

The IEEE-488 General Purpose Interface Bus (GPIB) is an instrumentation interface for integrating instruments, computers, printers, plotters, and other measurement devices into systems. The GPIB uses 16 signal lines to effect transfer of information between all devices connected to the bus.

The following requirements and restrictions apply to the GPIB:

- No more than 15 devices can be interconnected by one contiguous bus; however, an instrumentation system may contain more than one interface bus.
- The maximum total cumulative cable length for one interface bus may not exceed twice the number of devices connected (in meters), or 20 meters—whichever is less.
- A maximum data rate of 1 Mb/s across the interface on any signal line.
- Each device on the interface bus must have a unique address, ranging from 00 to 30.

The devices on the GPIB are connected in parallel, as shown in [Figure 1-1 on page 1-3](#). The interface consists of 16 signal lines and 8 ground lines in a shielded cable. Eight of the signal lines are the data lines, DIO 1 thru DIO 8. These data lines carry messages (data and commands), one byte at a time, among the GPIB devices. Three of the remaining lines are the handshake lines that control the transfer of message bytes between devices. The five remaining signal lines are referred to as interface management lines.

The following paragraphs provide an overview of the GPIB including a description of the functional elements, bus structure, bus data transfer process, interface management bus, device interface function requirements, and message types.

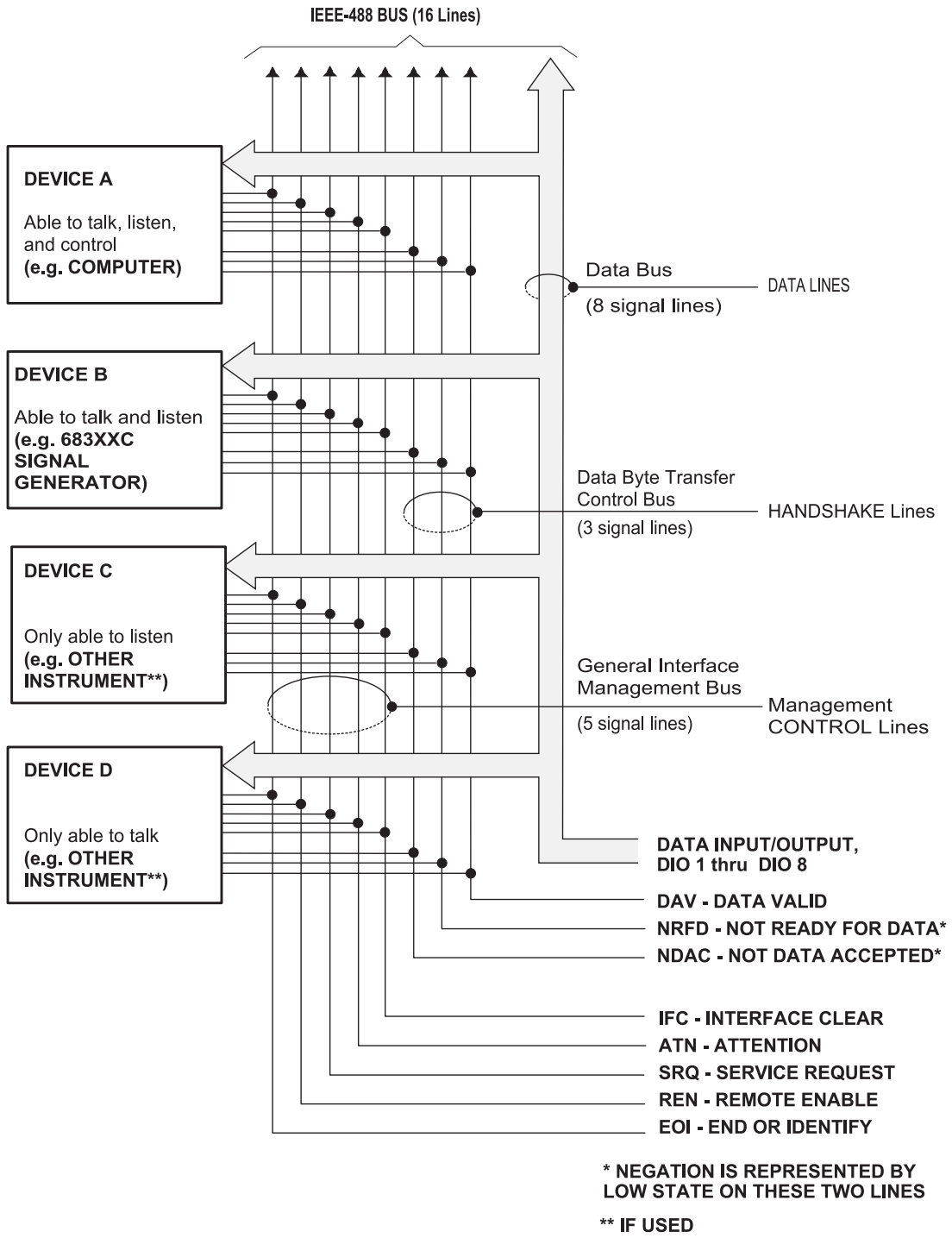


Figure 1-1. Interface Connections and GPIB Bus Structure

Functional Elements

Effective communications between devices on the GPIB requires three functional elements; a *talker*, a *listener*, and a *controller*. Each device on the GPIB is categorized as one of these elements depending on its current interface function and capabilities.

Talker

A talker is a device capable of sending device-dependent data to another device on the bus when addressed to talk. Only one GPIB device at a time can be an active talker.

Listener

A listener is a device capable of receiving device-dependent data from another device on the bus when addressed to listen. Any number of GPIB devices can be listeners simultaneously.

Controller

A controller is a device, usually a computer, capable of managing the operation of the GPIB. Only one GPIB device at a time can be an active controller. The active controller manages the transfer of device-dependent data between GPIB devices by designating who will talk and who will listen.

System Controller

The system controller is the device that always retains ultimate control of the GPIB. When the system is first powered-up, the system controller is the active controller and manages the GPIB. The system controller can pass control to a device, making it the new active controller. The new active controller, in turn, may pass control on to yet another device. Even if it is not the active controller, the system controller maintains control of the Interface Clear (IFC) and Remote Enable (REN) interface management lines and can thus take control of the GPIB at anytime.

Bus Structure

The GPIB uses 16 signal lines to carry data and commands between the devices connected to the bus. The interface signal lines are organized into three functional groups.

- Data Bus (8 lines)
- Data Byte Transfer Control Bus (3 lines)
- General Interface Management Bus (5 lines)

The signal lines in each of the three groups are designated according to function. [Table 1-1](#) lists these designations.

Table 1-1. Interface Bus Signal Line Designations

Bus Type	Signal Line Name	Function
Data Bus	DIO1–DIO8	Data Input/Output, 1 thru 8
Data Byte Transfer Control Bus	DAV	Data Available
	NRFD	Not Ready For Data
	NDAC	Not Data Accepted
General Interface Management Bus	ATN	Attention
	IFC	Interface Clear
	SRQ	Service Request
	REN	Remote Enable
	EOI	End Or Identify

Data Bus Description

The data bus is the conduit for the transfer of data and commands between the devices on the GPIB. It contains eight bi-directional, active-low signal lines—DIO 1 thru DIO 8. Data and commands are transferred over the data bus in byte-serial, bit-parallel form. This means that one byte of data (eight bits) is transferred over the bus at a time. DIO 1 represents the least-significant bit (LSB) in this byte and DIO 8 represents the most-significant bit (MSB). Bytes of data are normally formatted in seven-bit ASCII (American Standard Code for Information Interchange) code. The eighth (parity) bit is not used.

Each byte placed on the data bus represents either a command or a data byte. If the Attention (ATN) interface management line is TRUE while the data is transferred, then the data bus is carrying a bus command which is to be received by every GPIB device. If ATN is FALSE, then a data byte is being transferred and only the active listeners will receive that byte.

Data Byte Transfer Control Bus Description

Control of the transfer of each byte of data on the data bus is accomplished by a technique called the “three-wire handshake”, which involves the three signal lines of the Data Byte Transfer Control Bus. This technique forces data transfers at the speed of the slowest listener, which ensures data integrity in multiple listener transfers. One line (DAV) is controlled by the talker, while the other two (NRFD and NDAC) are wired-OR lines shared by all active listeners. The handshake lines, like the other GPIB lines, are active low. The technique is described briefly in the following paragraphs and is depicted in [Figure 1-2](#). For further information, refer to ANSI/IEEE Std 488.1.

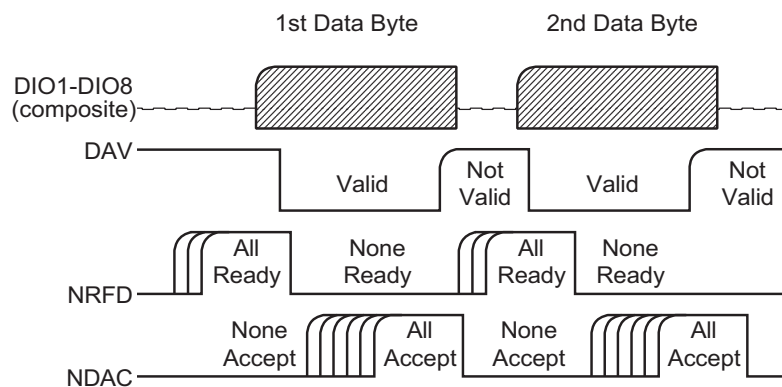


Figure 1-2. Typical GPIB Handshake Operation

DAV (Data Valid)

This line is controlled by the active talker. Before sending any data, the talker verifies that NDAC is TRUE (active low) which indicates that all listeners have accepted the previous data byte. The talker then places a byte on the data lines and waits until NRFD is FALSE (high) which indicates that all addressed listeners are ready to accept the information. When both NRFD and NDAC are in the proper state, the talker sets the DAV line TRUE (active low) to indicate that the data on the bus is valid (stable).

NRFD (Not Ready For Data)

This line is used by the listeners to inform the talker when they are ready to accept new data. The talker must wait for each listener to set the NRFD line FALSE (high) which they will do at their own rate. This assures that all devices that are to accept the data are ready to receive it.

NDAC (Not Data Accepted)

This line is also controlled by the listeners and is used to inform the talker that each device addressed to listen has accepted the data. Each device releases NDAC at its own rate, but NDAC will not go FALSE (high) until the slowest listener has accepted the data byte.

General Interface Management Bus Description

The general interface management bus is a group of five signal lines used to manage the flow of information across the GPIB. A description of the function of each of the individual control lines is provided below.

ATN (Attention)

The active controller uses the ATN line to define whether the information on the data bus is a command or is data. When ATN is TRUE (low), the bus is in the command mode and the data lines carry bus commands. When ATN is FALSE (high), the bus is in the data mode and the data lines carry device-dependent instructions or data.

EOI (End or Identify)

The EOI line is used to indicate the last byte of a multibyte data transfer. The talker sets the EOI line TRUE during the last data byte.

The active controller also uses the EOI line in conjunction with the ATN line to initiate a parallel poll sequence.

IFC (Interface Clear)

Only the system controller uses this line. When IFC is TRUE (low), all devices on the bus are placed in a known, quiescent state (unaddressed to talk, unaddressed to listen, and service request idle).

REN (Remote Enable)

Only the system controller uses this line. When REN is set TRUE (low), the bus is in the remote mode and devices are addressed either to listen or to talk. When the bus is in remote and a device is addressed, it receives instructions from the GPIB rather than from its front panel. When REN is set FALSE (high), the bus and all devices return to local operation.

SRQ (Service Request)

The SRQ line is set TRUE (low) by any device requesting service by the active controller.

Device Interface Function Capability

An interface function is the GPIB system element which provides the basic operational facility through which a device can receive, process, and send messages. Each specific interface function may only send or receive a limited set of messages within particular classes of messages. As a result, a set of interface functions is necessary to achieve complete communications among devices on the GPIB. ANSI/IEEE Std 488.1 defines each of the interface functions along with its specific protocol.

ANSI/IEEE Std 488.2 specifies the minimum set of IEEE 488.1 interface capabilities that each GPIB device must have. This minimum set of interface functions assures that the device is able to send and receive data, request service, and respond to a device clear message. [Table 1-2](#) lists the interface function capability of the series MG369xC signal generator.

Table 1-2. MG369xC Interface Function Capability

Function Identifier	Function	MG369xC Capability
AH1	Acceptor Handshake	Complete Capability
SH1	Source Handshake	Complete Capability
T6	Talker	No Talk Only (TON)
L4	Listener	No Listen Only (LON)
SR1	Service Request	Complete Capability
RL1	Remote/Local	Complete Capability
PP1	Parallel Poll	Complete Capability
DC1	Device Clear	Complete Capability
DT1	Device Trigger	Complete Capability
C0, C1, C2, C3, C28	Controller Capability Options	C0, No Capability
		C1, System Controller
		C2, Send IFC and Take Charge
		C3, Send REN
		C28, Send IF Messages
E2	Tri-State Drivers	Three-state bus drivers

Message Types

There are three types of information transmitted over the GPIB—interface function messages, device-specific commands, and data and instrument status messages.

Interface Function Messages

The controller manages the flow of information on the GPIB using interface function messages, usually called *commands* or *command messages*. Interface function messages perform such functions as initializing the bus, addressing and unaddressing devices, and setting device modes for remote or local operation.

There are two types of commands—multiline and uniline. Multiline commands are bytes sent by the active controller over the data bus (DIO1–DIO8) with ATN set TRUE. Uniline commands are signals carried by the individual interface management lines.

The user generally has control over these commands; however, the extent of user control depends on the implementation and varies with the specific GPIB interface hardware and software used with the external controller.

Device-Specific Commands

These commands are keywords or mnemonic codes sent by the external controller to control the setup and operation of the addressed device or instrument. The commands are normally unique to a particular instrument or class of instruments and are described in its documentation.

Device-specific commands are transmitted over the data bus of the GPIB to the device in the form of ASCII strings containing one or more keywords or codes. They are decoded by the device's *internal controller* and cause the various instrument functions to be performed.

Data and Instrument Status Messages

These messages are sent by the device to the external controller via the GPIB. They contain measurement results, instrument status, or data files that the device transmits over the data bus in response to specific requests from the external controller. The contents of these messages are instrument specific and may be in the form of ASCII strings or binary data.

In some cases data messages will be transmitted from the external controller to the device. For example, messages to load calibration data.

An SRQ (service request) is an interface function message sent *from the device* to the external controller to request service from the controller, usually due to some predetermined status condition or error. To send this message, the device sets the SRQ line of the General Interface Management Bus true, then sends a status byte on the data bus lines.

An SRQ interface function message is also sent by the device in response to a serial poll message from the controller, or upon receiving an Output Status Byte(s) command from the controller. The protocols associated with the SRQ functions are defined in the ANSI/IEEE Std 488.2 document.

The manner in which interface function messages and device-specific commands are invoked in programs is implementation specific for the GPIB interface used with the external controller. Even though both message types are represented by mnemonics, they are implemented and used in different ways.

Normally, the interface function messages are sent automatically by the GPIB driver software in response to invocation of a software function. For example, to send the IFC (Interface Clear) interface function message, one would call the `ibsic` function of the National Instruments software driver. On the other hand, the command `*RST` (Reset) is sent in a command string to the addressed device. In the case of the National Instruments example, this would be done by using the `ibwrt` function call.

1-6 MG369xC GPIB Operation

All Series MG369xC Synthesized Signal Generator functions, settings, and operating modes (except for power on/standby) are controllable using commands sent from an external controller via the GPIB. When in the remote (GPIB) mode, the signal generator functions as both a listener and a talker. The GPIB interface function capability of the MG369xC is listed in [Table 1-2 on page 1-7](#).

Setting GPIB Operating Parameters

The MG369xC leaves the factory with the GPIB address value set to 5 and the data delimiting terminator set to carriage return and line feed (CR/LF). A different address value can be entered from the front panel using the Configure GPIB menu. Using this same menu, the data delimiting terminator can be changed to carriage return (CR) only. Refer to the Series MG369xC Synthesized Signal Generator Operation Manual for the procedure.

Selecting the Interface Language

Series MG369xC Synthesized Signal Generators with the SCPI option can be remotely operated using one of two external interface languages—Native or SCPI. The Native interface language uses a set of MG369xC GPIB product specific commands to control the instrument; the SCPI interface language uses a set of the Standard Commands for Programmable Instruments (SCPI) commands to control the unit. Selecting which of these external interface languages is to be used can be made from the front panel using the Configure GPIB menu. Refer to [Section 2-4 “SCPI Interface Language Selection” on page 2-6](#).

Response to GPIB Interface Function Messages

[Table 1-3](#) lists the GPIB interface function messages that the MG369xC will recognize and respond to. With the exception of the Device Clear and Selected Device Clear messages, these messages affect only the operation of the MG369xC GPIB interface. The signal generator’s response for each message is indicated.

Interface function messages are transmitted on the GPIB data lines and interface management lines as either unaddressed or addressed commands. The manner in which these messages are invoked in programs is implementation dependent. For programming information, refer to the documentation included with the GPIB interface used for the external controller.

Table 1-3. MG369xC Response to GPIB Interface Function Messages

Interface Function Message	Addressed Command	MG369xC Response
Device Clear (DCL)	No	Resets the MG369xC to its default state. (Equivalent to sending the *RST command.)
Selected Device Clear (SDC)	Yes	
Go To Local (GTL)	Yes	Returns the MG369xC to local (front panel) control.
Group Execute Trigger (GET)	Yes	Executes a string of commands, if programmed.
Interface Clear (IFC)	No	Stops the MG369xC GPIB interface from listening or talking. (The front panel controls are not cleared.)
Local Lockout (LLO)	No	Disables the front panel menu RETURN TO LOCAL soft key.
Remote Enable (REN)	No	Places the MG369xC under remote (GPIB) control when it has been addressed to listen.
Serial-Poll Enable (SPE)	No	Outputs the serial-poll status byte.
Serial-Poll Disable (SPD)	No	Disables the serial-poll function.
Parallel-Poll Configure (PPC)	Yes	Responds to a parallel-poll message (PPOLL) by setting assigned data bus line to the logical state (1,0) that indicates its correct SRQ status.
Parallel-Poll Unconfigure (PPU)	No	Disables the parallel-poll function.

1-7 Remote Programming Setup and Interface

The MG369xC provides automated microwave signal generation via Ethernet (VXI-11 over TCP/IP) or the GPIB. The following paragraphs provide general information about interface language, utilizing VISA, interface connections, and cable requirements.

Refer to [Section 1-8](#) for GPIB setup instructions.

Refer to [Section 1-9](#) for Ethernet setup instructions.

Interface Language

The series MG369xC synthesized signal generators can be remotely operated via the Ethernet or GPIB using an external interface language – Native or SCPI. Refer to [“Related Manuals” on page 1-1](#).

VISA (Virtual Instrument System Architecture)

VISA is an I/O software standard for communicating with test instruments, like the MG369xC, over any of the bus architectures which the MG369xC supports. Anritsu recommends NI-VISA, the National Instruments implementation of the VISA I/O standard.

Note	Anritsu recommends that you use the full version of NI-VISA (not just the runtime) for the best support of the latest .NET Framework, USB, and TCP/IP updates. Check with National Instruments for details.
-------------	---

The MG369xC LabVIEW driver uses VISA to communicate. VISA uses connection strings to set up communication with the MG369xC over various protocols. Some examples include:

```
//VXI-11 Connection string
"TCPIP0::192.168.1.7::inst0::INSTR";
//GPIB Connection string
"GPIB0::6::INSTR";
```

To communicate with the MG369xC, note either the IP address or GPIB address of the signal generator and then set up a resource using the appropriate connection string.

Note	When using VISA, change the connection string to access any of the communication protocols noted. The remaining code can remain intact. VXI-11 is used over TCP/IP, since it implements the IEEE 488.2 standard and all status checking. For this reason, the remainder of information contained in this manual will use GPIB as default remote connectivity selection – but it will be generally applicable with Ethernet connectivity as well.
-------------	--

Interface Connectors

When using Ethernet connectivity, the interface between the signal generator and other devices is through an RJ45 connector and a standard CAT-7 Ethernet cable. For instruments on the GPIB, connectivity is established via a 24-wire interface cable. This cable uses connector shells having two connector faces. These double-faced connectors allow for the parallel connection of two or more cables to a single device.

1-8 GPIB Interconnection and Setup

GPIB Interconnection

The only interconnection required for GPIB operation is between the signal generator and the controller. This interconnection is via a standard GPIB cable. The Anritsu part number for such a cable is 2100-1, -2, or -4 (1, 2, or 4 meters in length).

GPIB Cable Length Restrictions

The GPIB can accommodate up to 15 instruments at any one time. To achieve design performance on the bus, proper timing and voltage level relationships must be maintained. If either the cable length between separate instruments or the cumulative cable length between all instruments is too long, the data and control lines cannot be driven properly and the system may fail to perform. Cable length restrictions are as follows:

- No more than 15 instruments may be installed on the bus.
- Total cumulative cable length (in meters) may not exceed two times the number of bus instruments or 20 meters—whichever is less.

Note For low EMI applications, the GPIB cable should be a fully shielded type with well-grounded metal-shell connectors.

Setting the GPIB Address

The default GPIB address is five. If a different GPIB address is desired, it can be set from the front panel using the Configure GPIB menu.

To change the GPIB address:

1. Press the front panel main menu key labeled **System**.

The System menu is displayed.

2. Press the Config soft key.

The System Configuration menu ([Figure 1-3](#)) is displayed.

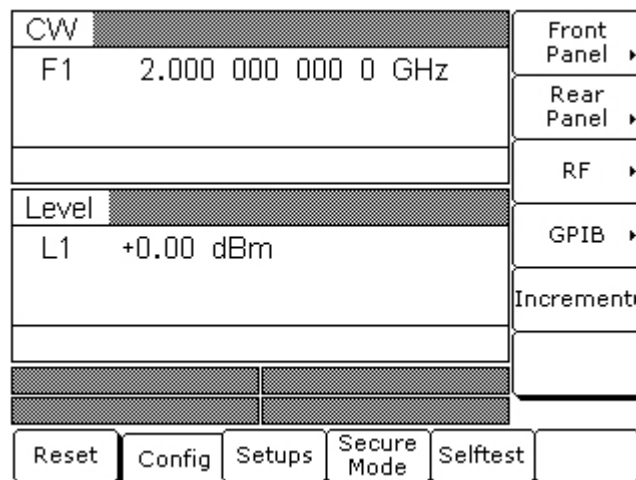


Figure 1-3. System Configuration Menu

- To access the Configure GPIB menu from this menu, press the GPIB > soft key.

The Configure GPIB menu (Figure 1-4) is displayed.

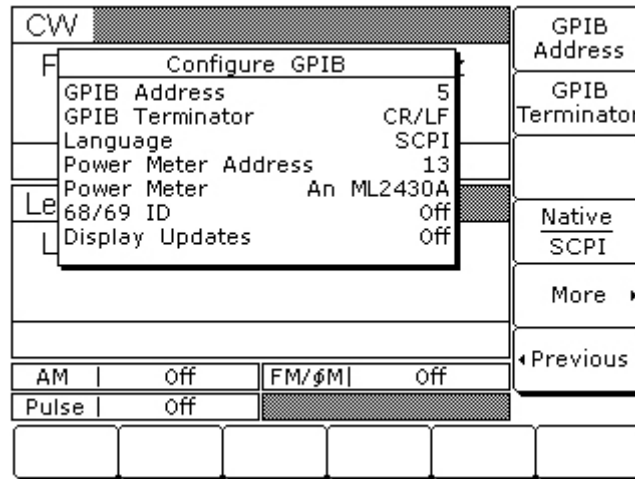


Figure 1-4. GPIB Configuration Menu

- Press the menu soft key GPIB Address [ADD] to change the current GPIB address of the signal generator.
- Enter a new address using the cursor control keys or the data entry keypad and the terminator soft key [ADR].

The new GPIB address will now appear on the display. The entry must be between 1 and 30 to be recognized as a valid GPIB address.

Selecting the Line Terminator

Data is delimited on the GPIB by either the carriage return (CR) ASCII character or both the carriage return and line feed (CR/LF) ASCII characters. Which character is used depends upon the requirements of the system controller. Most modern controllers can use either CR or CR/LF, while many older controllers require one or the other. Consult the controller's manual for its particular requirements.

- From the Configure GPIB menu display, you can select which GPIB terminator to use by pressing the menu soft key GPIB Terminator.

This menu soft key toggles the GPIB terminator between CR and CR/LF. The current selection appears on the display.

1-9 Ethernet LAN TCP/IP Requirements and Setup

The MG3690C supports Ethernet 10/100 BASE-T. The instrument is connected directly to the LAN via the rear panel RJ-45 Ethernet Port using a standard CAT-7 Ethernet cable.

Information about Ethernet LAN general requirements, configuration, use, and restoring defaults are discussed in the following sections. Ethernet control programming is discussed with an example in the MG369xC GPIB and SCPI programming manuals.

- [“TCP/IP General Requirements and Settings”](#)
- [“Ethernet Setup - Static IP Configuration”](#)
- [“Ethernet Setup - DHCP with AutoIP Configuration”](#)
- [“Example PC Control of MG369xC via LAN”](#)
- [“Resetting Ethernet IP Address to Factory Default”](#)

Note	This section is provided for general information about directly manually configuring an Ethernet connection. Consult the user local network administrator for the exact requirements and settings that are required for the network installation.
-------------	---

TCP/IP General Requirements and Settings

Transmission Control Protocol/Internet Protocol (TCP/IP) is a network protocol. In the Windows XP (or newer) operating system, TCP/IP is automatically installed and in most cases, installation, configuration, and communication are transparent to the user.

In a TCP/IP network, the user must provide IP addresses and other information to clients. Clients may also require a naming service or a method for name resolution. The TCP/IP protocol setup requires the following information:

IP Address: Every device in a TCP/IP network requires an IP address that consists of four groups of numbers, each between 0 and 255, separated by periods. For example: 128.111.122.42 is a valid IP address.

Subnet Mask: The subnet mask is the network identification (ID) address from the portion that is the station ID address, and assists in routing information on the network.

Default Gateway: A TCP/IP network may have a gateway to communicate beyond the LAN identified by the network ID. A gateway is a computer or electronic device that is connected to two different networks and can move TCP/IP data from one network to the other. A single LAN that is not connected to other LANs requires a default gateway setting of 0.0.0.0. The default gateway setting for the MG3690C series signal generators is 0.0.0.0. If the network has a gateway, then the default gateway would be set to the appropriate value of the gateway.

Hardware Address (MAC Address): A media access control (MAC) address is a unique 48-bit hexadecimal value that identifies the network interface card internal to the signal generator to the rest of the network. Every network card has a unique MAC address permanently stored into its memory.

Network Interface Setup: TCP/IP connectivity requires setting up the parameters described at the beginning of this section. The user may need to contact the network administrator or refer to the network documentation for further assistance. The following procedure is a general overview of how to set up a general LAN connection on both the signal generator and the remote machine. The actual menus and sequence may vary.

Auto Configuration

The MG3690C can be configured to automatically acquire its TCP/IP settings from an existing Ethernet network that provides Dynamic Host Configuration Protocol (DHCP).

Ethernet Setup - Static IP Configuration

The MG369xC ships with the Ethernet/GPIB Controller set to a default static IP address of 192.168.0.254. If you want to set the address to something other than the default, you can use the Static Configuration or DHCP with AutoIP configuration. This section provides steps for static configuration. Check with your network administrator for any limitations on setting up a Static IP. Refer to “[Ethernet Setup - DHCP with AutoIP Configuration](#)” on page 1-17 for configuring DHCP. For best results when configuring the Ethernet connection, Anritsu recommends that you first connect the MG369xC directly to a PC using a crossover cable and then set the TCP/IP settings.

To set up using STATIC TCP/IP configuration:

1. Connect a PC to the MG369xC using a CAT-7 Ethernet crossover cable.
2. Set the PC network adapter settings to match the subnet and subnet mask of the instrument, but with a different static PC IP address than the instrument. The instrument factory default settings are:

Static IP Address: 192.168.0.254

Gateway: 0.0.0.0

Subnet Mask: 255.255.255.0

If the network settings are not known, refer to “[Resetting Ethernet IP Address to Factory Default](#)” on page 1-24.

3. Open a web browser on the PC. Enter the instrument IP address and press **Enter**. The browser will connect to the Ethernet/GPIB Controller and display the MG369x Welcome Page as shown in [Figure 1-5](#).

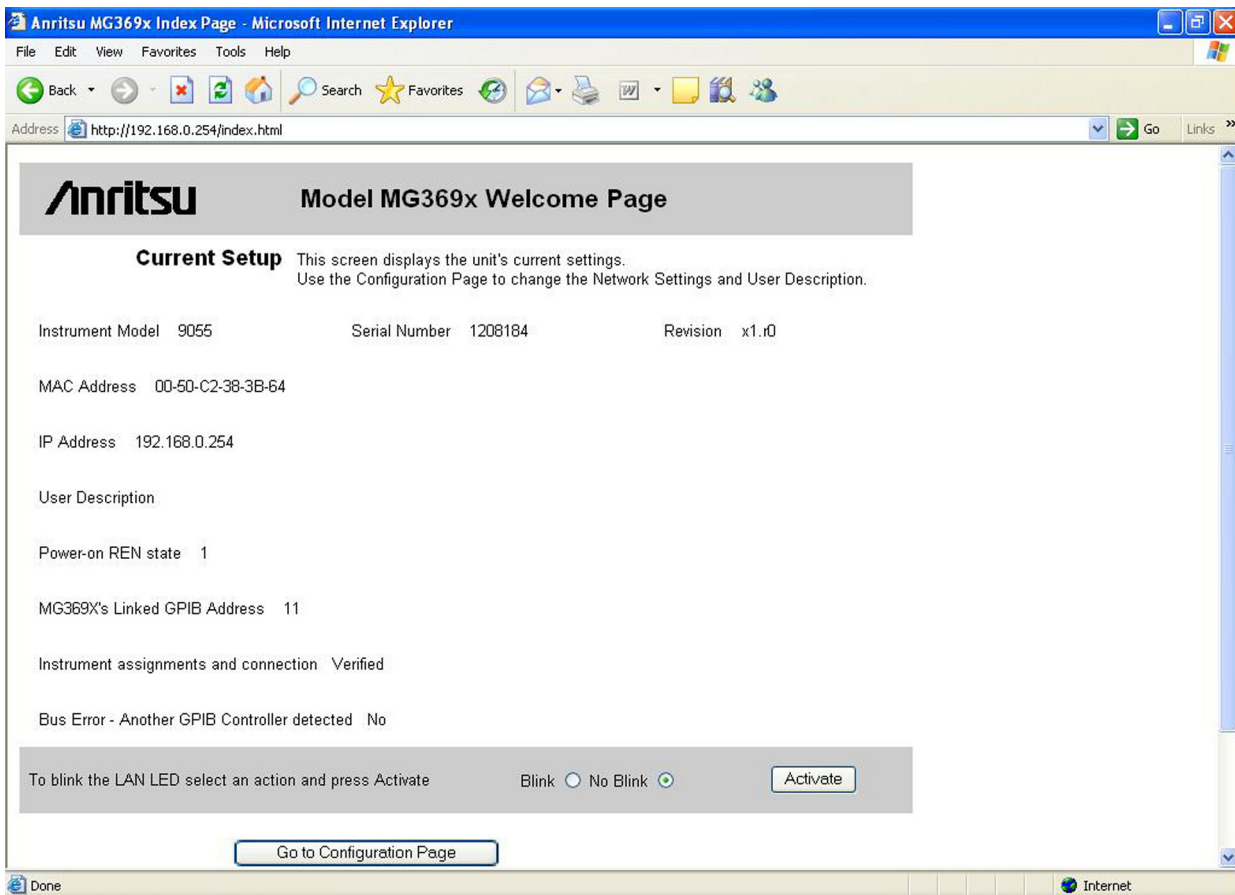


Figure 1-5. MG369xC Ethernet Device Welcome/Current Settings Page

4. Click the Go to Configuration Page button. The MG369x TCP/IP configuration page appears as shown in [Figure 1-6](#) on page 1-15.

5. In the IP Address field, enter the IP address as indicated by your network administrator.
6. Enter the correct subnet mask and gateway for your network, if applicable.
7. Click the Update Flash button to complete the configuration.

The TCP/IP Confirmation Page appears as shown in [Figure 1-7 on page 1-16](#) (refer to “TCP/IP General Requirements and Settings” on [page 1-13](#) for information on the below settings).

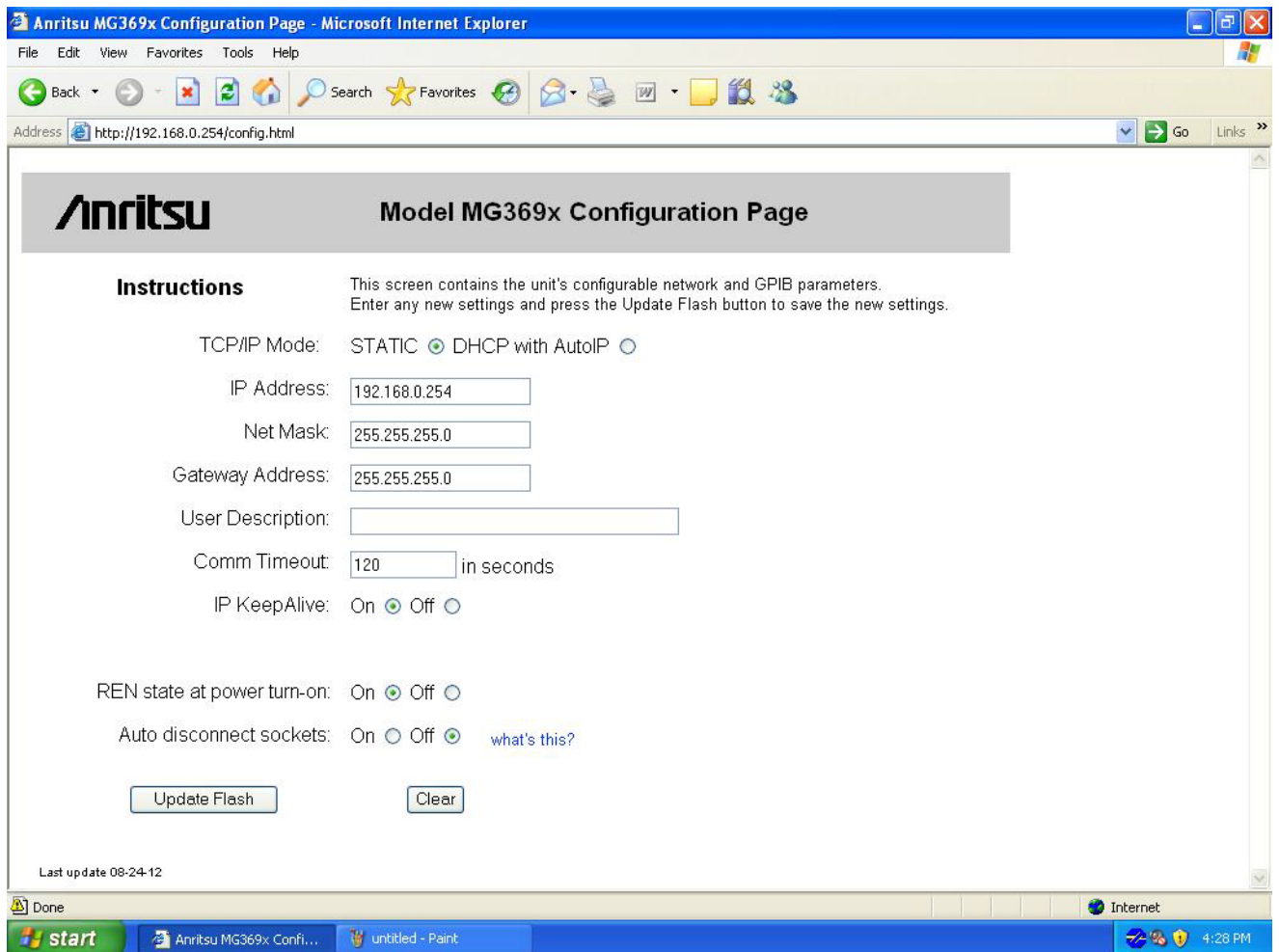


Figure 1-6. MG369xC Ethernet Device Configuration Page - Static TCP/IP Mode

8. Click Reboot button for the changes to take effect or power cycle the unit later to activate the changes.

Note When rebooting, it can take three to five minutes for the changes to take effect.

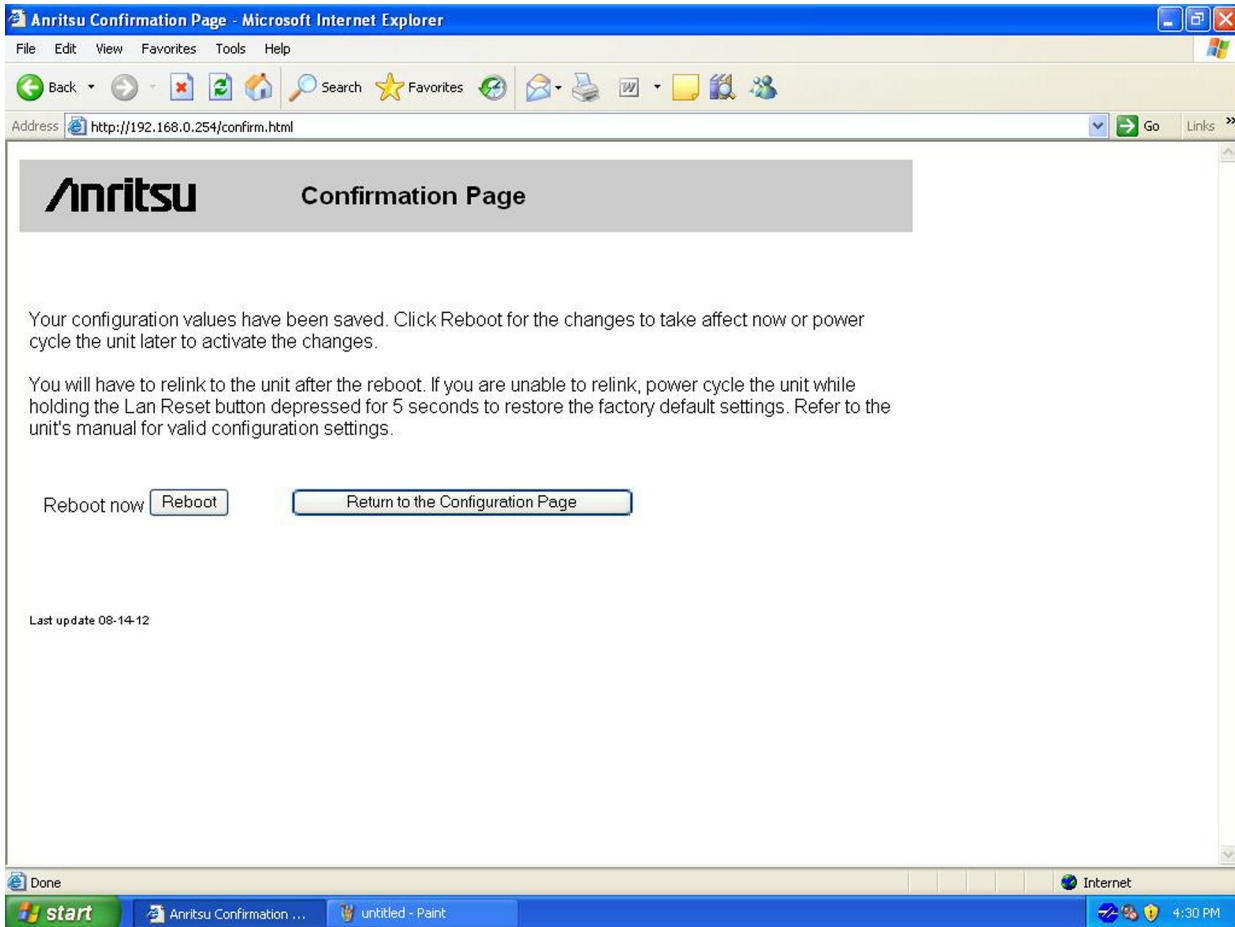


Figure 1-7. Ethernet Configuration Confirmation Page

9. Make sure the PC network settings are changed to connect to the newly entered IP address on the instrument.

10. To validate the TCP/IP connection, at the PC browser, enter the IP address of the MG369xC.

The Model MG369xC Welcome Page should again appear, this time with the new IP address in the Current Setup summary

Testing the TCP/IP Connection Over LAN

1. Connect the MG369xC Ethernet port to the LAN.

2. Using a PC that is also connected to the LAN, enter the instrument IP address into an open web browser (the default instrument IP is 192.168.0.254).

The Model MG369xC Welcome Page should again appear, with the new IP address in the Current Setup summary.

Ethernet Setup - DHCP with AutoIP Configuration

The MG369xC ships with the Ethernet/GPIB Controller set to a default static IP address of 192.168.0.254. If you want to set the address to something other than the default, you can use this procedure to set DHCP with AutoIP. To change the static IP address, refer to “[Ethernet Setup - Static IP Configuration](#)” on page 1-14. For best results when configuring the Ethernet connection, Anritsu recommends that you first connect the MG369xC directly to a PC using a crossover cable and then set the TCP/IP settings.

To set up using DHCP with AutoIP TCP/IP configuration:

1. Connect a PC to the MG369xC using a CAT-7 Ethernet crossover cable.
2. Set the PC network adapter settings to match the subnet and subnet mask of the instrument, but with a different static PC IP address than the instrument. The instrument factory default settings are:

Static IP Address: 192.168.0.254

Gateway: 0.0.0.0

Subnet Mask: 255.255.255.0

If the network settings are not known, refer to “[Resetting Ethernet IP Address to Factory Default](#)” on page 1-24.

3. Open a web browser on the PC. Enter the instrument IP address and press **Enter**. The browser will connect to the Ethernet/GPIB Controller and display the MG369x Welcome Page as shown in [Figure 1-5](#).

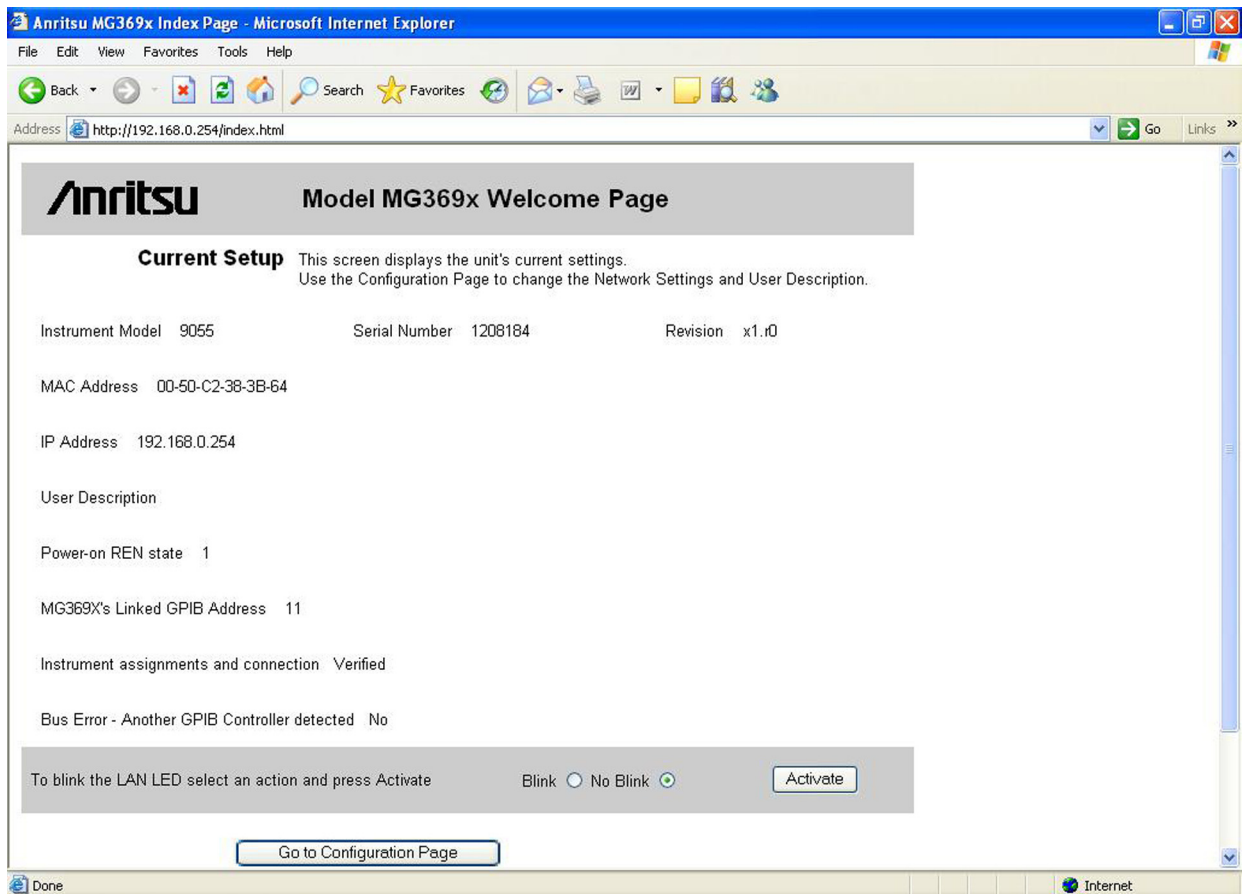


Figure 1-8. MG369xC Ethernet Device Welcome/Current Settings Page

4. Click the Go to Configuration Page button. The MG369x TCP/IP configuration page appears as shown in [Figure 1-6](#) on page 1-15.

5. Select DHCP with AutoIP.

The configuration page is altered for DHCP as shown in [Figure 1-9](#).

6. Click the Update Flash button to complete the configuration. The TCP/IP Confirmation Page appears as shown in [Figure 1-10 on page 1-19](#) (refer to “TCP/IP General Requirements and Settings” on page 1-13 for information on the below settings).

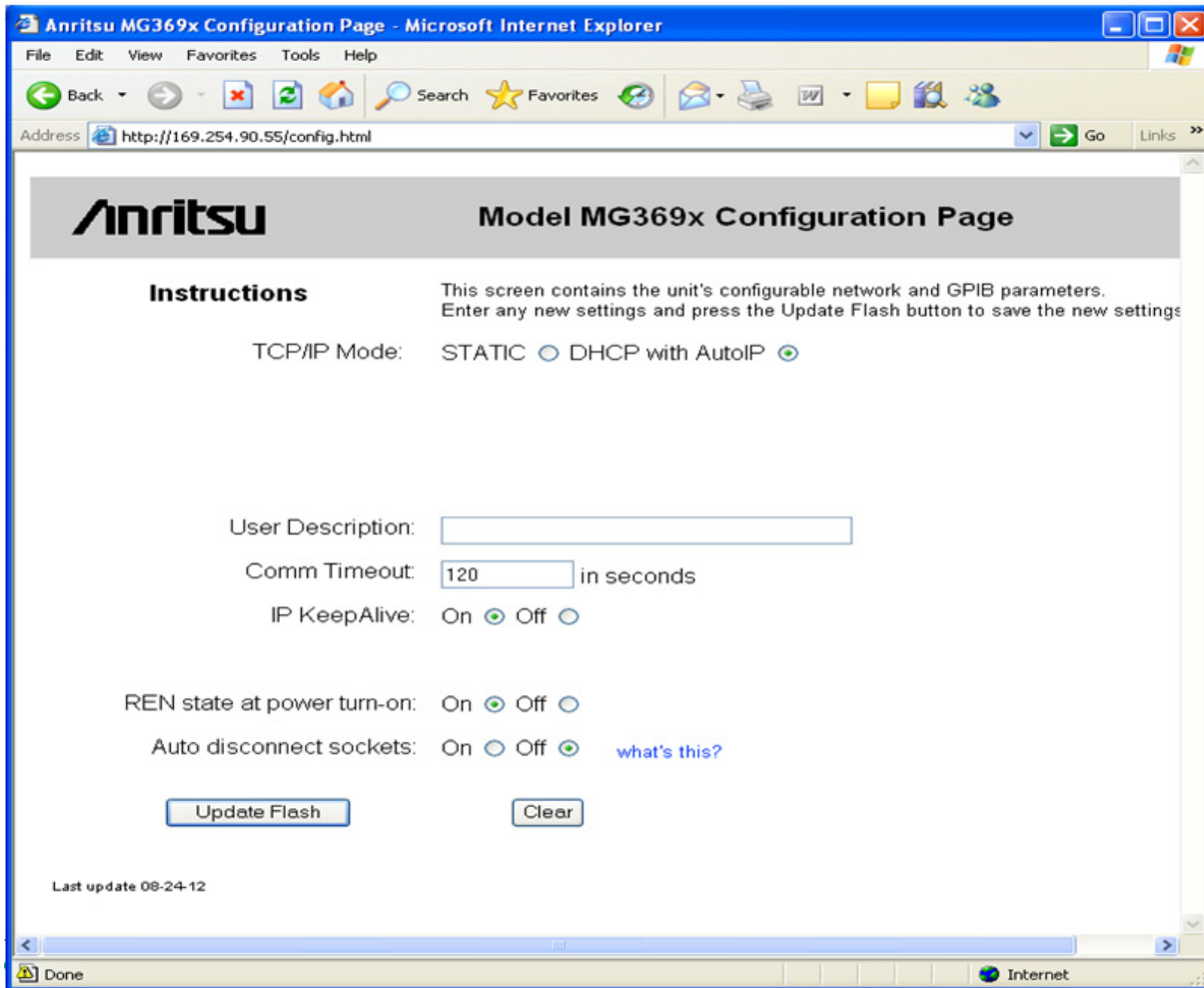


Figure 1-9. MG369xC Ethernet Device Configuration Page - DHCP with AutoIP TCP/IP Mode

- Click Reboot button for the changes to take effect or power cycle the unit later to activate the changes.

Note When rebooting, it can take three to five minutes for the changes to take effect.

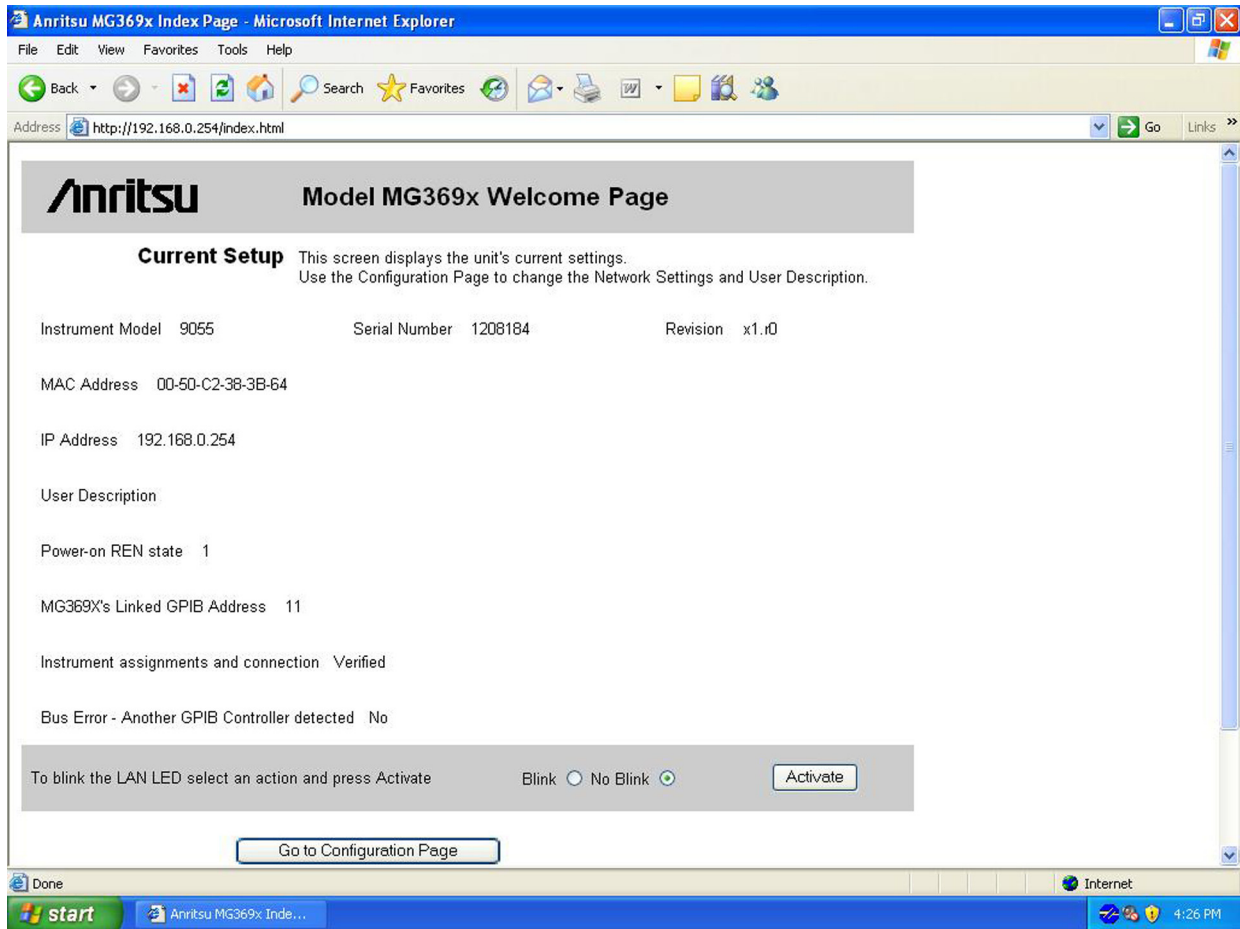


Figure 1-10. Ethernet Configuration Confirmation Page

- After reboot is complete, find the IP address that was assigned by the network to the MG369xC Ethernet/GPIB Controller.

Note The MG369xC MAC address can be used to find its IP address on the Network by sending the command “arp -a” via a DOS command prompt. The MAC address can be found on the back panel of the instrument and is also shown on the Welcome web page for the instrument.

Alternatively, the user can query “*IDN?” on the available network instruments to locate the IP address of the MG369xC.

- To validate your TCP/IP connection, at the PC browser, enter the IP address of the MG369xC.

The Model MG369xC Welcome Page should again appear, this time with the new IP address in the Current Setup summary.

Testing the TCP/IP Connection Over LAN

- Connect the MG369xC Ethernet port to the LAN.
- Using a PC that is also connected to the LAN, enter the LAN IP address into an open web browser. The Model MG369xC Welcome Page should appear, with the new IP address in the Current Setup summary..

Example PC Control of MG369xC via LAN

Following is an example of controlling the MG369xC using the NI-MAX utility.

1. On the host machine, run NI MAX and expand Devices and Interfaces.
2. Expand VISA TCP/IP Resources and select TCPIP0::192.168.0.254::inst0::INSTR.

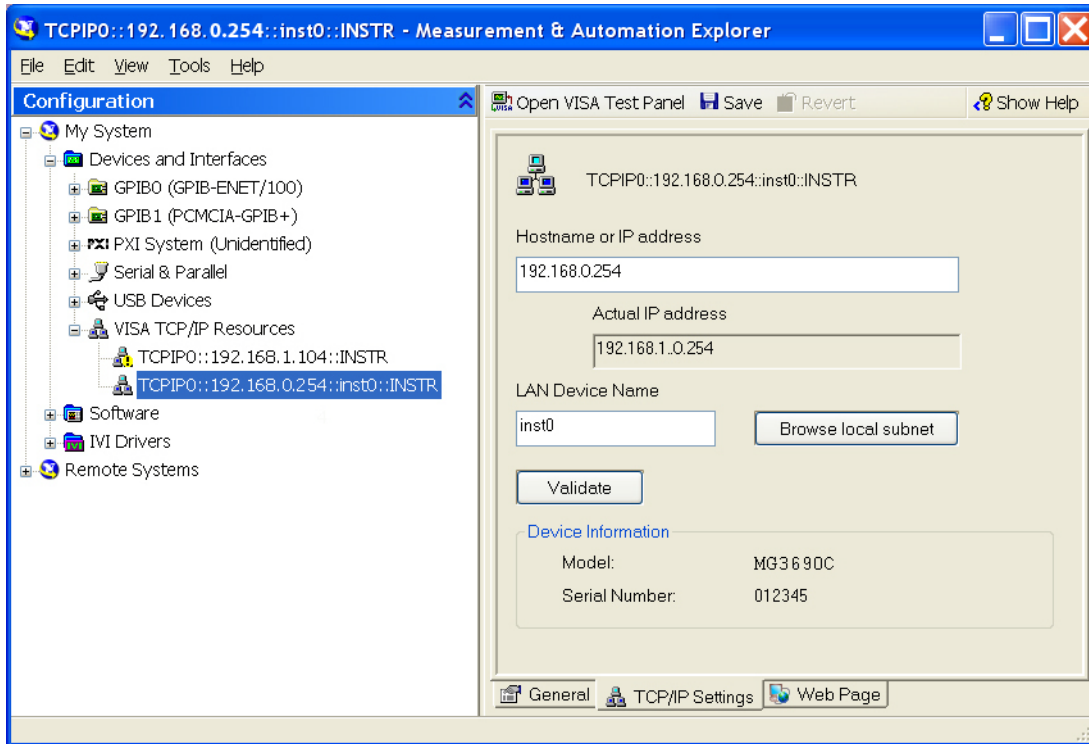


Figure 1-11. NI MAX TCP-IP Configuration

3. Right click TCPIP0::192.168.0.254::inst0::INSTR and open a VISA test panel.
4. Select the Basic I/O tab and execute the default *IDN? write within the viWrite tab buffer area.

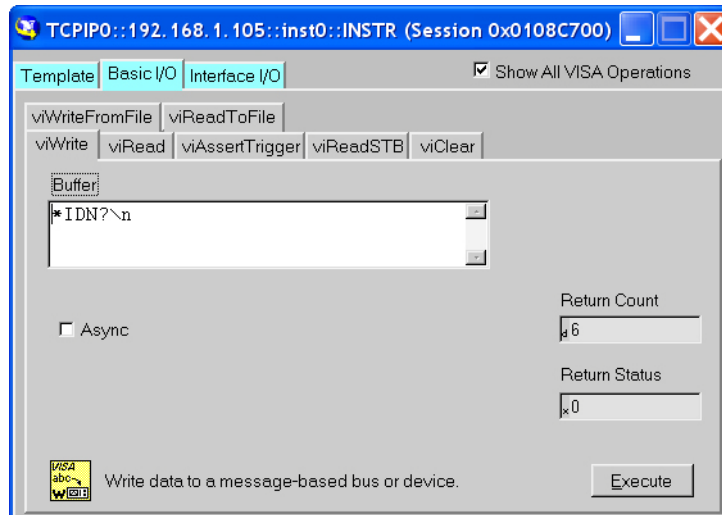


Figure 1-12. NI MAX TCP-IP Basic I/O Write

5. In the same Basic I/O tab and execute a read by clicking in the viRead tab. The Buffer area should display the instrument ID.

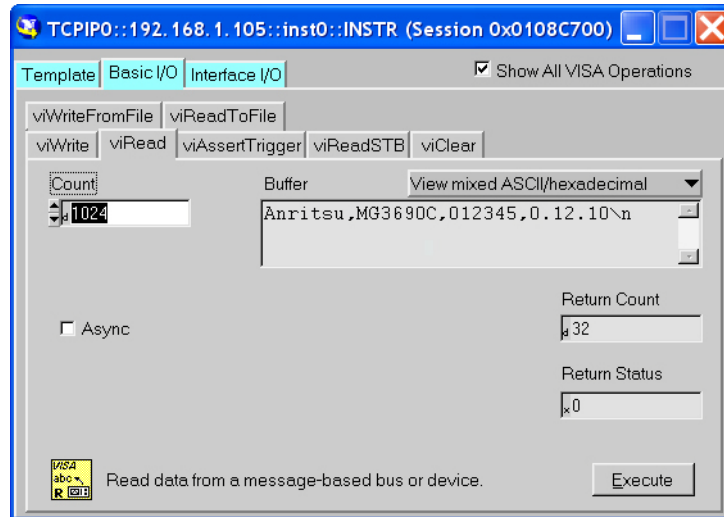


Figure 1-13. NI MAX TCP-IP Basic I/O Read

Ethernet Control Example in C# Programming using NI VISA

Note NI is a trademark of the National Instruments Corporation.

```
//Ethernet control example in C# programming using National Instruments Visa
using System;
using System.Collections.Generic;
using System.Text;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;

namespace AutoTriggerSweepButton
{
    class Program
    {
        static void Main(string[] args)
        {
            String TcpIpResourceName = "TCPIP::172.26.203.104::inst0::INSTR";
            //
            string responseString = null;
            NationalInstruments.VisaNS.MessageBasedSession mbSession = null;
            try
            {
                mbSession =
(MessageBasedSession)ResourceManager.GetLocalManager().Open(TcpIpResourceName);
                mbSession.Write("*IDN?");
                Console.WriteLine("The Instrument ID is:");
                responseString = mbSession.ReadString();
                Console.WriteLine(responseString);
                mbSession.Write("INITiate:CONTinuous OFF");
                mbSession.Write("TRIGger:SOURce IMMEDIATE");
                mbSession.Write("FREQuency:MODE SWEep1");
                mbSession.Write("SWEep1:GENeration STEPped");
                mbSession.Write("FREQuency:STARt 3 GHz");
                mbSession.Write("FREQuency:STOP 4 GHz");
                mbSession.Write("SWEep1:POINts 101");
                mbSession.Write("SWEep1:DWELL 1 ms");
                mbSession.Write("INITiate:CONTinuous ON");
                // Delay to allow some sweeps to occur.
            }
            catch { }
        }
    }
}
```

```
        Thread.Sleep(3000);
        // Stop sweeping
        mbSession.Write("INITiate:CONTinuous OFF");
        mbSession.Write("ABORt");
        Console.WriteLine("Auto Trigger Sweep Setup Complete");
    }
    catch (Exception ex)
    {
        Console.WriteLine(ex.Message);
    }
    finally
    {
        if (mbSession != null)
        {
            mbSession.Dispose();
        }
    }
}
```

Resetting Ethernet IP Address to Factory Default

Note The MG369xC MAC address can be used to find its IP address on the Network by sending the command “arp -a” via a DOS command prompt. The MAC address can be found on the back panel of the instrument and is also shown on the Welcome web page for the instrument.

Alternatively, the user can query “*IDN?” on the available network instruments to locate the IP address of the MG369xC.

In the event that Ethernet communication is lost between the LAN client and the Ethernet/GPIB Controller card on the MG369xC, the following IP address reset procedure can be performed.

This procedure resets the IP address to the factory default STATIC IP address of 192.168.0.254.

1. Power off the MG3690C and disconnect the Ethernet cable from the rear panel.
2. Power cycle the unit while shorting Pin 19 of the AUX I/O connector to an available ground (pin 2, 8, or 24 of the AUX I/O connector, or to the metal chassis) for five seconds to restore the factory default settings.
3. Power down the MG3690C and reconnect the Ethernet cable to the rear panel.
4. To verify that the IP address has been reset to 192.168.0.254, connect to a PC with its IP address set to 192.168.0.253.
5. Open Internet Explorer and type 192.168.0.254 in the browser address line.
6. The Welcome Page should appear as shown in [Figure 1-15 on page 1-25](#).

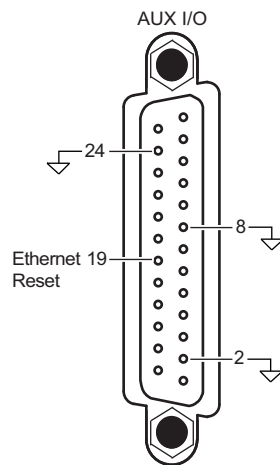


Figure 1-14. Resetting Ethernet IP Address to Factory Default

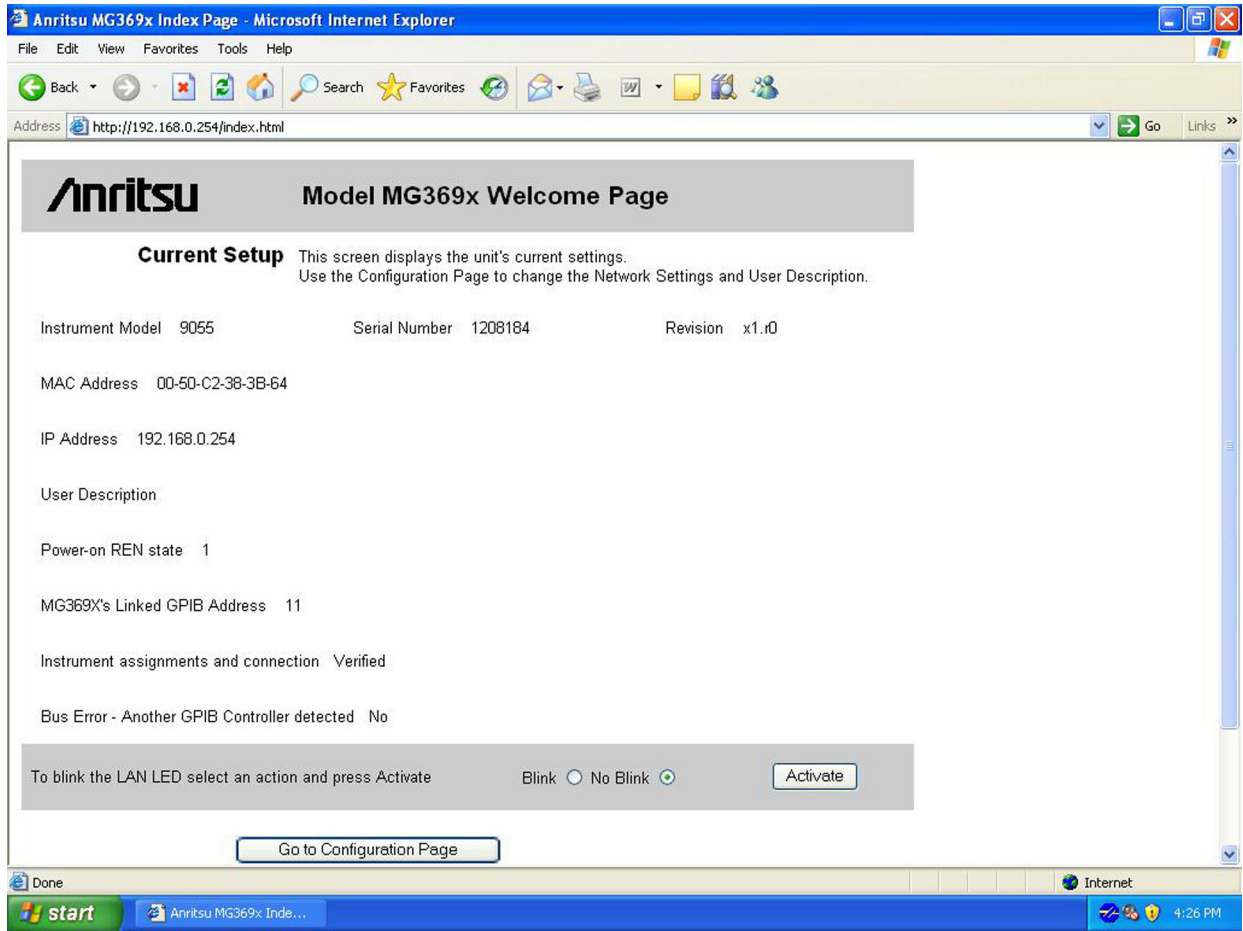


Figure 1-15. Factory IP Address Reset Verification

Chapter 2 — Programming with SCPI Commands

2-1 Introduction

This chapter provides an introduction to SCPI programming that includes descriptions of the command types, hierarchical command structure, data parameters, and notational conventions. Information on MG369xC status system and trigger system programming is also provided.

2-2 Introduction to SCPI Programming

The *Standard Commands for Programmable Instruments (SCPI)* defines a set of standard programming commands for use by all SCPI compatible instruments. SCPI is intended to give the ATE user a consistent environment for program development. It does so by defining controller messages, instrument responses, and message formats for all SCPI compatible instruments. The IEEE-488 (GPIB) interface for the MG369xC was designed to conform to the requirements of SCPI 1993.0. The set of SCPI commands implemented by the MG369xC GPIB interface provides a comprehensive set of programming functions covering all the major functions of the MG369xC signal generator.

SCPI Command Types

SCPI commands, which are also referred to as SCPI instructions, are messages to the instrument to perform specific tasks. The MG369xC command set includes:

- “Common” commands (IEEE488.2 mandated commands)
- SCPI required commands
- SCPI optional commands (per SCPI 1993.0)
- SCPI compliant commands that are unique to the MG369xC

The SCPI conformance information for the MG369xC command set is contained in Appendix B—SCPI Conformance Information.

Common Commands

The required common commands are IEEE-488.2 mandated commands that are defined in IEEE-488.2 and must be implemented by all SCPI compatible instruments. These commands ([Table 2-1](#)) are identified by the asterisk (*) at the beginning of the command keyword. These commands are used to control instrument status registers, status reporting, synchronization, and other common functions. The common commands and their syntax are described in detail in [“Common Commands” on page 3-1](#).

Table 2-1. Common Commands

*CLS	*IDN?	*RST	*STB?
*ESE	*OPC	*SRE	*TST?
*ESE?	*OPC?	*SRE?	*WAI
*ESR?			

Required and Optional SCPI Commands

The required SCPI commands are listed in [Table 2-2](#) and are described in detail in [Section 3-11](#) and [Section 3-12](#). The optional SCPI commands and MG369xC unique commands comprise the remainder (major portion) of the MG369xC command set. They control the majority of the programmable functions of the MG369xC. They are described in detail in [Chapter 3](#) starting in [Section 3-3](#).

Table 2-2. SCPI Required Commands

```

:STATus
  :OPERation
    [:EVENT]?
  :CONDition?
  :ENABle
:PRESet
:QUESTionable
  [:EVENT]?
  :CONDition?
  :ENABle
:SYSTem
  :ERRor?
  :VERSion?

```

Query Commands

All commands, unless specifically noted in the syntax descriptions in [Chapter 3](#), have a query form. As defined in IEEE-488.2, a query is a command with a question mark symbol appended (examples: *ESR? and :FREQuency:CENTer?). When a query form of a command is received, the current setting associated with the command is placed in the output buffer.

Command Names

Typical SCPI commands consist of one or more keywords, parameters, and punctuation. SCPI command keywords can be a mixture of upper and lower case characters. Except for common commands, each keyword has a long and a short form. In this manual, the long form is presented with the short form in upper case and the remainder in lower case. For example, the long form of the command keyword to control the instrument display is: DISPlay.

The short form keyword is usually the first four characters of the long form (example: DISP for DISPlay). The exception to this is when the long form is longer than four characters and the fourth character is a vowel. In such cases, the vowel is dropped and the short form becomes the first three characters of the long form. Example: the short form of the keyword POWer is POW.

Some command keywords may have a numeric suffix to differentiate between multiple instrument features such as dual channel inputs. For example: keywords EXTernal1 and EXTernal2 (or EXT1 and EXT2).

As with any programming language, the exact command keywords and command syntax must be used. The syntax of the individual commands is described in detail in [Chapter 3](#). Unrecognized versions of long form or short form commands, or improper syntax, will generate an error. Error reporting is described in [Chapter 4](#).

Hierarchical Command Structure

All SCPI commands, except the common commands, are organized in a hierarchical structure similar to the inverted tree file structure used in most computers. The SCPI standard refers to this structure as “the Command Tree.” The command keywords that correspond to the major instrument control functions are located at the top of the command tree. The command keywords for the MG369xC SCPI command set are shown in [Figure 2-1](#).

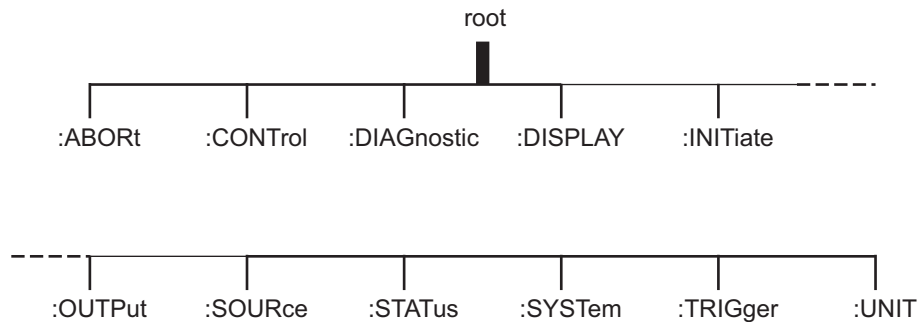


Figure 2-1. SCPI Command Tree

All MG369xC SCPI commands, except the `ABORt` command, have one or more subcommands (keywords) associated with them to further define the instrument function to be controlled. The subcommand keywords may in turn also have one or more associated subcommands (keywords). Each subcommand level adds another layer to the command tree. The command keywords and its associated subcommand keywords form a portion of the command tree called a command *subsystem*. The `:CONTRol` command subsystem is shown in [Figure 2-2](#).

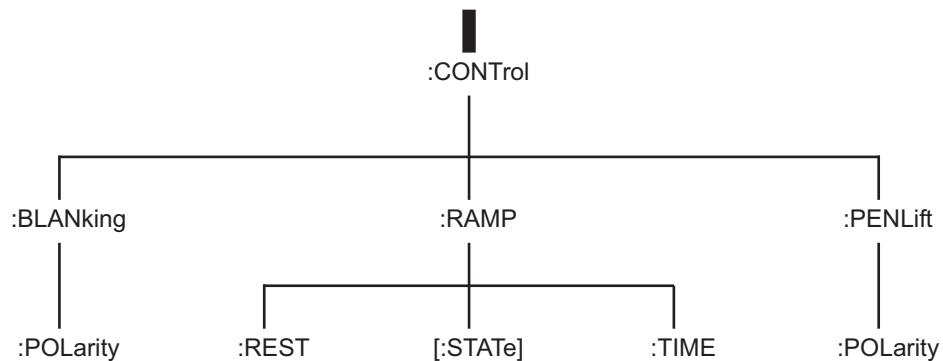


Figure 2-2. Command Subcommand Example

An overall command tree for the MG369xC SCPI command set is shown in [Figure A-1 on page A-2](#).

Data Parameters

Data parameters, referred to simply as “parameters,” are the quantitative values used as arguments for the command keywords. The parameter type associated with a particular SCPI command is determined by the type of information required to control the particular instrument function. For example, Boolean (ON|OFF) type parameters are used with commands that control switch functions.

The command descriptions in [Chapter 3](#) specify the type of data parameter to be used with each command. The most commonly used parameter types are numeric, extended numeric, discrete, and Boolean.

Numeric

Numeric parameters comprise integer numbers, or any number in decimal or scientific notation and may include polarity signs. This includes <NR1>, <NR2>, and <NR3> numeric data as defined in “[Parameter Notations](#)” on [page 2-5](#). This type of numeric element is abbreviated as <NRF> throughout this document.

Extended Numeric

Extended numeric parameters include values such as MAXimum and MINimum.

Discrete

Discrete parameters, such as INTernal and EXTernal, are used to control program settings to a predetermined finite value or condition.

Boolean

Boolean parameters represent binary conditions and may be expressed as ON, OFF or 1, 0.

Unit Suffixes

Unit suffixes are not required for data parameters, provided the values are scaled for the global default units. The MG369xC SCPI default units are: Hz (Hertz) for frequency related parameters and S (seconds) for time related parameters. For example, the command below sets the MG369xC output frequency to 3 GHz.

```
:SOURce:FREQuency:CW 3000000000
```

The global default units may be changed via use of the :UNIT Subsystem commands described in [Section 3-15](#).

2-3 Notational Conventions

The SCPI interface standardizes command syntax and style that simplifies the task of programming across a wide range of instrumentation. As with any programming language, the exact command keywords and command syntax must be used. Unrecognized commands, or improper syntax, will generate an error (refer to [Chapter 4](#) for error reporting).

General Notations

The syntax conventions that are used for all SCPI command keywords and data parameter descriptions in this manual are described below:

- : A colon links command keywords together to form commands. The colon is not an actual part of the keyword but is a signal to the SCPI interface parser. A colon must precede a root keyword immediately following a semicolon (See “[Notational Examples](#)” on page 2-6).
- ; A semicolon separates commands if multiple commands are placed on a single program line (See “[Notational Examples](#)” on page 2-6).
- [] Square brackets enclose one or more *optional* parameters.
- { } Braces enclose one or more parameters *that may be included one or more times*.
- | A vertical bar indicates “or” and is used to separate alternative parameter options.
For example: ON | OFF is the same as ON or OFF.
- < > Angle brackets enclose parameter descriptions.
- ::= Means “is defined as” For example: <a> ::= <c> indicates that <c> can replace <a>. Space(s), referred to as whitespace, must be used to separate keywords from their associated data parameters. It must not be used between keywords, or inside keywords.

For further information about SCPI command syntax and style, refer to the *Standard Commands for Programmable Instruments (SCPI) 1993.0* document.

Parameter Notations

The following syntax conventions are used for all data parameter descriptions in this manual

- <arg> ::= a generic command argument consisting of one or more of the other data types.
- <bNR1> ::= boolean values in <NR1> format; numeric 1 or 0 .
- <boolean> ::= ON | OFF. Can also be represented as 1 or 0, where 1 means ON and 0 means OFF. Boolean parameters are always returned as 1 or 0 in <NR1> format by query commands.
- <integer> ::= an unsigned integer without a decimal point (implied radix point).
- <NR1> ::= a signed integer without a decimal point (implied radix point).
- <NR2> ::= a signed number with an explicit radix point.
- <NR3> ::= a scaled explicit decimal point numeric value with and exponent (e.g., floating point number).
- <NRf> ::= <NR1> | <NR2> | <NR3>.
- <nv> ::= SCPI numeric value: <NRf> | MIN | MAX | UP | DOWN | DEF | NAN | INF | NINF or other types.
- <char> ::= <CHARACTER PROGRAM DATA> Examples: CW, FIXed, UP, and DOWN.
- <string> ::= <STRING PROGRAM DATA> ASCII characters surrounded by double quotes, example: "OFF".
- <block> ::= IEEE-488.2 block data format.
- <NA> ::= Not Applicable.

Notational Examples

The following is an example showing command syntax (It is not an actual command):

```
[SOURce]:POWer[:LEVel] [:IMMediate] [:AMPLitude]:STEP[:INCRement] sp dBm|DOWN|UP
```

Command statements read from left to right and from top to bottom. In the command statement above, the :STEP keyword immediately follows the :AMPLitude keyword *with no separating space*. A space (*sp*) is used between the command string and its argument (a <nv> type data parameter).

Note that the first keyword in the command string does not require a leading colon; however, it is good practice to always use a leading colon for *all* keywords. Note also that the :SOURce keyword is optional. This is a SCPI convention for all voltage or signal source type instruments that allows shorter command statements to be used.

The following is an example of a multiple command statement that uses two separate commands in a single statement. Note the semicolon used to join the commands. (Also note the leading colon used immediately after the semicolon.)

```
:FREQuency:STARt 10E6;:FREQuency:STOP 20E9
```

2-4 SCPI Interface Language Selection

The Series MG369xC Synthesized Signal Generator can be remotely operated using one of two external interface languages—Native or SCPI. (The Native interface language uses a set of MG369xC GPIB product specific commands to control the instrument.) Before programming with SCPI commands it is necessary to select SCPI as the external interface language.

Front Panel Selection

SCPI can be selected as the MG369xC interface language from the front panel Configure GPIB menu.

To access the Configure GPIB Menu, first press the **SYSTEM** main menu key on the front panel to access the System menu. At the menu display, press **Config** to access the System Configuration menu. Then, press **GPIB**. The Configure GPIB menu is displayed.

The Configure GPIB menu has an additional menu display. Language selection is made from this additional menu. To access the additional menu, press **More**. At the menu, press **SCPI / Native** to select SCPI. The language selection will appear on the display.

Remote Selection

SCPI can be selected as the MG369xC interface language during remote operations.

To change the interface language from Native to SCPI use the command

```
SYST:LANG "SCPI"
```

Do not use the long form of the command and do not use a leading colon (:) with the command. The command :SYSTem:LANGuage "SCPI" results in a syntax error.

Note

When the MG369xC signal generator is remotely operated using the SCPI interface language, cycling the power returns the instrument to a reset condition.

2-5 Status System Programming

The MG369xC status system (shown in [Figure 2-3 on page 2-8](#)) consists of the following SCPI-defined status-reporting structures:

- The Instrument Summary Status Byte Group
- The Standard Event Status Group
- The Operational Status Group
- The Questionable Status Group

The following paragraphs describe the registers that make up a status group and explain the status information that each status group provides.

Status Group Registers

In general, a status group consists of a condition register, a transition filter, an event register, and an enable register. Each component is briefly described in the following paragraphs.

Condition Register

The condition register is continuously updated to reflect the current status of the MG369xC. There is no latching or buffering for this register, it is updated in real time. Reading the contents of a condition register does not change its contents.

Transition Filter

The transition filter is a special register that specifies which types of bit state changes in the condition register will set corresponding bits in the event register. Negative transition filters (NTR) are used to detect condition changes from True (1) to False (0); positive transition filters (PTR) are used to detect condition changes from False (0) to True (1). Setting both positive and negative filters True allows an event to be reported anytime the condition changes. Transition filters are read-write. Transition filters are unaffected by queries or *CLS (clear status) and *RST commands.

The command :STATus:PRESet sets all negative transition filters to all 0's and sets all positive transition filters to all 1's.

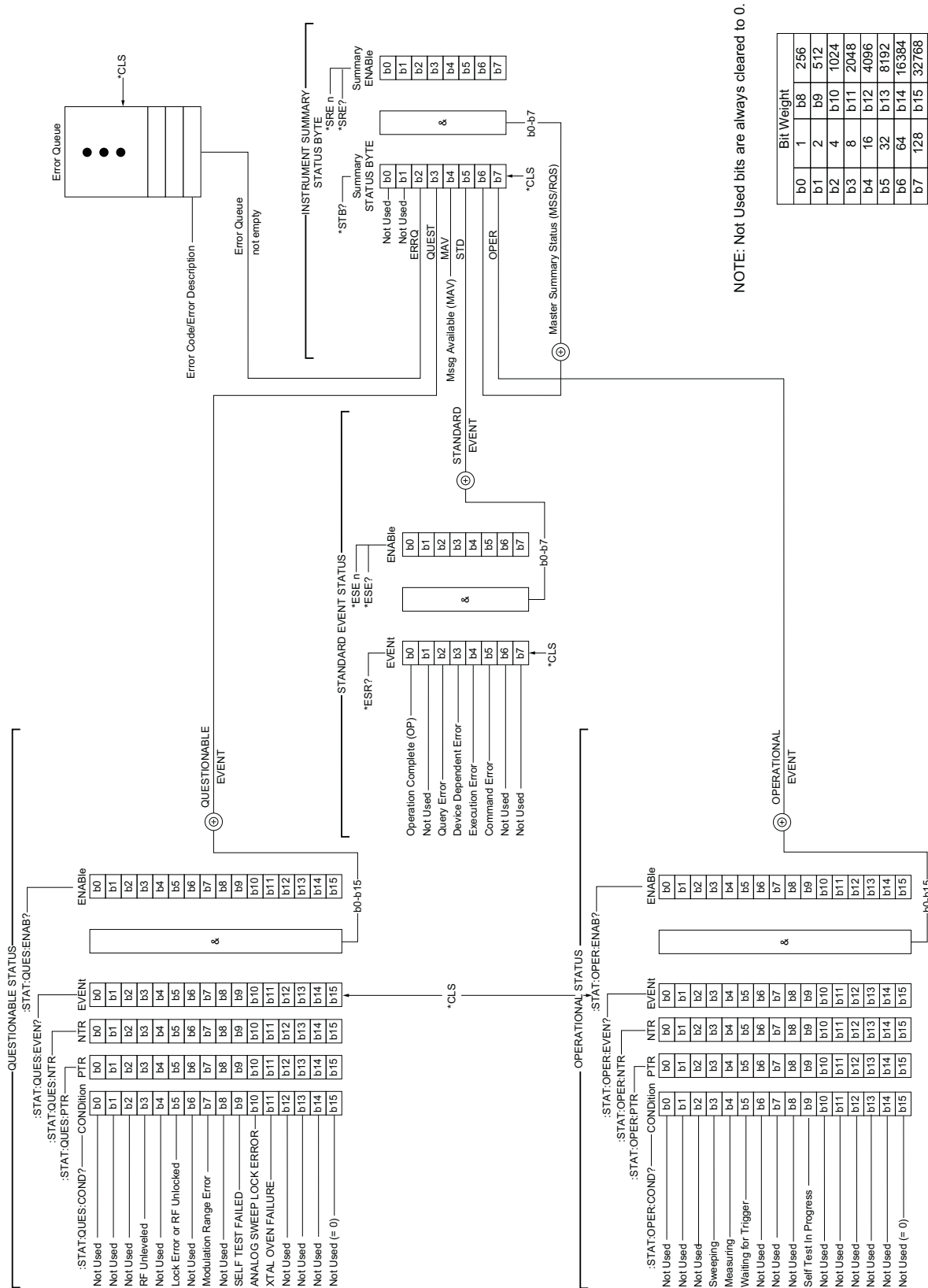
Event Register

The event register latches transition events from the condition register as specified by the transition filter. Bits in the event register are latched, and once set they remain set until cleared by a query or a *CLS command. Event registers are read only.

Enable Register

The enable register specifies the bits in the event register that can produce a summary bit. The MG369xC logically ANDs corresponding bits in the event and enable registers, and ORs all the resulting bits to obtain a summary bit. Summary bits are recorded in the Summary Status Byte. Enable registers are read-write. Querying an enable register does not affect it.

The command :STATus:PRESet sets the Operational Status Enable register and the Questionable Status Enable register to all 0's.



NOTE: Not Used bits are always cleared to 0.

Figure 2-3. MG369xC Status-Reporting Structure

Status Group Reporting

The state of certain MG369xC hardware and operational events and conditions can be determined by programming the status system. As shown in [Figure 2-3](#), the three lower status groups provide status information to the Summary Status Byte group. The Summary Status Byte group is used to determine the general nature of an event or condition and the other status groups are used to determine the specific nature of the event or condition.

Note

Programming commands for the status system, including examples of command usage, can be found in [Chapter 3](#).

The following paragraphs explain the information that is provided by each status group.

Summary Status Byte Group

The Summary Status Byte group, consisting of the Summary Status Byte Enable register and the Summary Status Byte, is used to determine the general nature of a MG369xC event or condition. The bits in the Summary Status Byte provide the following information:

Table 2-3. Summary Status Byte

Bit	Description
0, 1	Not Used. These bits are always set to 0.
2	Set to indicate the Error Queue contains data. The Error Query command can then be used to read the error message(s) from the queue.
3	Set to indicate the Questionable Status summary bit has been set. The Questionable Status Event register can then be read to determine the specific condition that caused the bit to be set.
4	Set to indicate that the MG369xC has data ready in its output queue.
5	Set to indicate that the Standard Event Status summary bit has been set. The Standard Event Status register can then be read to determine the specific event that caused the bit to be set.
6	Set to indicate that the MG369xC has at least one reason to require service. This bit is also called the Master Summary Status Bit (MSS). The individual bits in the Status Byte are ANDed with their corresponding Service Request Enable Register bits, then each bit value is ORed and input to this bit.
7	Set to indicate that the Operational Status summary bit has been set. The Operational Status Event register can then be read to determine the specific condition that caused the bit to be set.

Standard Event Status Group

The Standard Event Status group, consisting of the Standard Event Status register (an Event register) and the Standard Event Status Enable register, is used to determine the specific event that set bit 5 of the Summary Status Byte. The bits in the Standard Event Status register provide the following information:

Table 2-4. Standard Event Status

Bit	Description
0	Set to indicate that all pending MG369xC operations were completed following execution of the “*OPC” command.
1	Not Used. The bit is always set to 0.
2	Set to indicate that a query error has occurred. Query errors have SCPI error codes from –499 to –400.
3	Set to indicate that a device-dependent error has occurred. Device-dependent errors have SCPI error codes from –399 to –300 and 1 to 32767.
4	Set to indicate that an execution error has occurred. Execution errors have SCPI error codes from –299 to –200.
5	Set to indicate that a command error has occurred. Command errors have SCPI error codes from –199 to –100.
6,7	Not Used. The bits are always set to 0.

Operational Status Group

The Operational Status group, consisting of the Operational Condition register, the Operational Positive Transition register, the Operational Negative Transition register, the Operational Event register, and the Operational Event Enable register, is used to determine the specific condition that set bit 7 in the Summary Status Byte. The bits in the Operational Event register provide the following information:

Table 2-5. Operational Status Event

Bit	Description
0–2	Not Used. The bits are always set to 0.
3	Set to indicate that a sweep is in progress.
4	Set to indicate that the MG369xC is measuring.
5	Set to indicate that the MG369xC is in an armed “wait for trigger” state.
6	Not Used. The bit is always set to 0.
7	Not Used. The bit is always set to 0.
8	Not Used. The bit is always set to 0.
9	Set to indicate that MG369xC self-test is in progress.
10–14	Not Used. The bits are always set to 0.
*15	Always 0. The use of Bit 15 is not allowed by SCPI.

Questionable Status Group

The Questionable Status group, consisting of the Questionable Condition register, the Questionable Positive Transition register, the Questionable Negative Transition register, the Questionable Event register, and the Questionable Event Enable register, is used to determine the specific condition that set bit 3 in the Summary Status Byte. The bits in the Questionable Event register provide the following information:

Table 2-6. Questionable Event Status

Bit	Description
0–2	Not Used. The bits are always set to 0.
3	Set to indicate an RF unlevelled condition.
4	Not Used. The bit is always set to 0.
5	Set to indicate a phase-lock error or RF unlocked condition.
6	Not Used. The bit is always set to 0.
7	Set to indicate a modulation range error.
8	Not Used. The bit is always set to 0.
9	Set to indicate that self-test failed.
10	Set to indicate an analog sweep phase-lock error.
11	Set to indicate a failure of the crystal oven.
12–14	Not Used. The bits are always set to 0.
*15	Always 0. The use of Bit 15 is not allowed by SCPI.

2-6 Trigger System Programming

The MG369xC trigger system is used to synchronize signal generator actions with software trigger commands. The MG369xC follows the layered trigger model used in SCPI instruments. The following paragraphs describe operation and programming of the signal generator trigger system. The structure and components of the MG369xC trigger model are shown in [Figure 2-4](#).

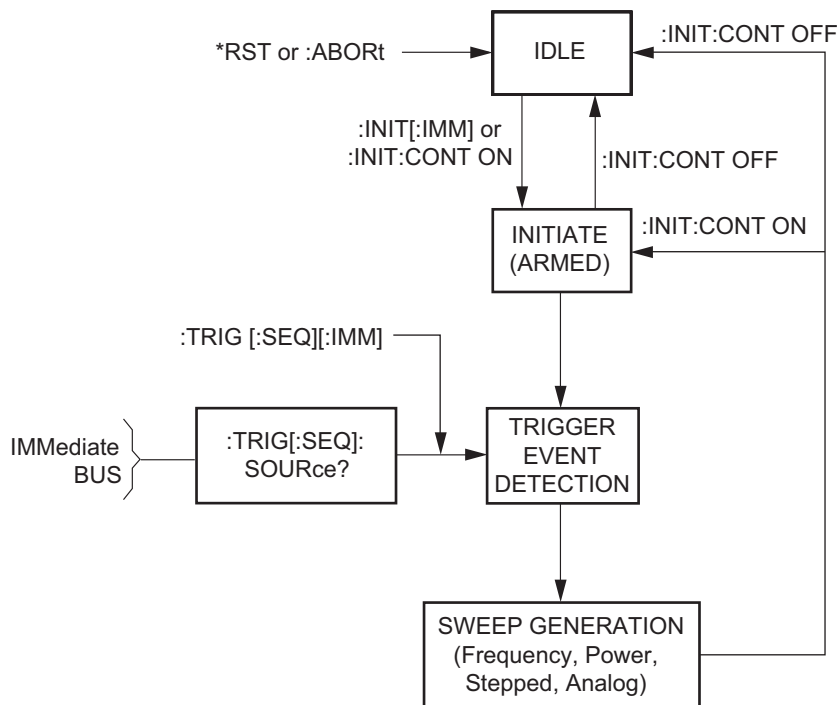


Figure 2-4. MG369xC Trigger Model

Trigger System Operation

Turning power on, or sending `*RST` or `:ABORT` forces the trigger system into the *idle* state. The trigger system remains in the *idle* state until it is initiated. Trigger system initiation can happen on a continuous basis (`:INITiate:CONTinuous ON`) or on a demand basis (`:INITiate:CONTinuous OFF`). When the command `:INITiate:CONTinuous` is set to `OFF`, the trigger system is initiated by the `:INITiate[:IMMEDIATE]` command. Note that `*RST` sets `:INITiate:CONTinuous` to `OFF`.

Once initiated, the trigger system enters an armed (wait for trigger) state. The trigger signal selected by the command `:TRIGger[:SEQUENCE]:SOURce` is examined until a `TRUE` condition is detected. The trigger signal selections are:

IMMEDIATE	The trigger signal is always <code>TRUE</code> .
BUS	The trigger signal is either the GPIB <code><GET></code> (Group Execute Trigger) message or the <code>*TRG</code> command.
HOLD	The trigger signal is never <code>TRUE</code> .

When a `TRUE` condition is detected, sweep generation of the selected sweep starts.

The command `:TRIGger[:SEQUENCE][:IMMEDIATE]` provides a one-time override of the normal downward path in the trigger-event-detection state by forcing a `TRUE` trigger signal regardless of the setting for `:TRIGger[:SEQUENCE]:SOURce`.

Upon sweep completion, if `:INITiate:CONTinuous` is set OFF, the trigger system returns to the *idle* state. If `:INITiate:CONTinuous` is set to ON, the trigger system returns to the *armed* (wait for trigger) state.

Auto Trigger Mode

Setting the command `:INITiate:CONTinuous` to ON and the command `:TRIGger[:SEQuence]:SOURce` to IMMEDIATE, places the trigger system in an auto trigger mode. This causes continuous generation of the selected sweep.

ABORt

The `:ABORt` command resets any sweep in progress and immediately returns the trigger system to the idle state. Unlike `*RST`, `:ABORt` does not change the settings programmed by other commands.

Chapter 3 — Programming Commands

3-1 Introduction

This chapter contains information on all SCPI programming commands accepted and implemented by the Series MG369xC Synthesized Signal Generator.

3-2 Common Commands

Common commands are used to control instrument status registers, status reporting, synchronization, data storage, and other common functions. All common commands are identified by the leading asterisk in the command word. The common commands are fully defined in IEEE 488.2.

IEEE 488.2 Mandated Commands

The MG369xC implements the following IEEE-488.2 mandated common commands.

*CLS (Clear Status Command)

Clear the Status Byte, the Data Questionable Event Register, the Standard Event Status Register, the Standard Operation Status Register, the error queue, the OPC pending flag, and any other registers that are summarized in the Status Byte.

*ESE <nv> (Standard Event Status Enable Command)

Sets the Standard Event Status Enable Register bits. The binary weighted <NR1> data parameter used with this command must have a value between 0 to 255. Refer to [“Status System Programming” on page 2-7](#).

*ESE? (Standard Event Status Enable Query)

Returns the value of the Standard Event Status Enable Register in <NR1> format. Refer to [“Status System Programming” on page 2-7](#).

*ESR? (Standard Event Status Register Query)

Returns the value of the Standard Event Status Register in <NR1> format. *This command clears the Standard Event Status Register.* Refer to [“Status System Programming” on page 2-7](#).

*IDN? (Identification Query)

This query returns an instrument identification string in IEEE-488.2 specified <NR1> format (four fields separated by commas). The fields are: <Manufacturer>, <Model>, <Serial #>, <Firmware revision level>; where the actual model number, serial number, and firmware version of the MG369xC queried will be passed.

*OPC (Operation Complete Command)

Enables the Operation Complete bit in the Standard Event Status Register after all pending operations are complete.

*OPC? (Operation Complete Query)

Places an ASCII “1” in the Output Queue and sets the MAV bit true in the Status Byte when all pending operations are completed (per IEEE-488.2 section 12.5.3). Message is returned in <NR1> format.

***RST** (Reset Command)

Resets the MG369xC to a pre-defined condition with all user programmable parameters set to their default values. These default parameter values are listed under each SCPI command in this manual. This command does not affect the Output Queue, Status Byte Register, Standard Event Register, or calibration data.

Note

This command clears the current front panel setup. If this setup is needed for future testing, save it as a stored setup using the *SAV command before issuing the *RST command.

***SRE <nv>** (Service Request Enable Command)

Sets the Service Request Enable Register bits. The integer data parameter used with this command must have a value between 0 to 255. A zero value resets the register. Refer to “[Status System Programming](#)” on page 2-7.

***SRE?** (Service Request Enable Query)

Returns the value of the Service Request Enable Register in <NR1> format. Bit 6 is always zero.

***STB?** (Read Status Byte Query)

Returns the content of the Status Byte Register (bits 0–5 and 7). Bit 6 is the Master Summary Status bit value. This command does not reset the status byte values.

***TST?** (Self-Test Query)

Causes the MG369xC to perform a full internal self-test. Status messages that indicate self-test results are placed in the error queue in the order they occur. Bits in the status register are also affected.

Returns the number of errors placed in the error queue. 0 means the unit passed self-test.

Caution

MG369xC self-test requires RF output power to be on. **Always** disconnect sensitive equipment from the unit before performing a self-test.

***WAI** (Wait-to-Continue Command)

This command suspends the execution of any further commands or queries until all operations for pending commands are completed. For example, the command *TRG; *WAI permits synchronous sweep operation. It causes the MG369xC to start a sweep and wait until the sweep is complete before executing the next command.

Optional Common Commands

The MG369xC implements the following IEEE 488.2 optional common commands:

***OPT?** (Option Identification Query)

This command returns a string identifying any device options.

***RCL <n>** (Recall Stored State)

This command restores the MG369xC to a front panel setup state that was previously saved to local (instrument) memory using the *SAV command (below). The *RCL <n> command restores setup <n>, where n shall be in the range of 0 to 9.

***SAV <n>** (Save Current State)

Saves the current front panel setup parameters in local (instrument) memory. The new stored setup state will be assigned the Setup Number specified by <n>, where n shall be in the range of 0 to 9.

***TRG** (Trigger Command)

Triggers instrument if :TRIGger:SOURce command data parameter is BUS. Refer to INITiate and TRIGger subsystem commands.)

Performs the same function as the Group Execute Trigger <GET> command defined in IEEE 488.1.

3-3 Subsystem Commands

Subsystem commands control all signal generator functions and some general purpose functions. All subsystem commands are identified by the colon used between keywords, as in :INITiate:CONTinuous.

The following information is provided for each subsystem command.

- The path from the subsystem root command
- The data parameters used as arguments for the command. This includes the parameter type, the available parameter choices, the range for numeric parameters, and the default parameter that is set by the *RST command
- A description of the purpose of the command
- The query form of the command (if applicable)
- An example of the use of the command
- Where necessary, notes are included to provide additional information about the command and its usage

An overall command tree for the MG369xC SCPI command set is shown in [Figure A-1 on page A-2](#).

3-4 :ABORt Subsystem

The :ABORt command is a single command subsystem. There are no subcommands or associated data parameters, as shown below. The :ABORt command, along with the :TRIGger and :INITiate commands, comprise the “Trigger Group” of commands.

:ABORt

Description: Forces the trigger system to the idle state. Any sweep in progress is aborted as soon as possible.

Parameters: None

Example: :ABORt

Sets MG369xC trigger system to idle state.

Associated

commands: :TRIGger and :INITiate

3-5 :CONTrol Subsystem

The :CONTrol subsystem sets the state of the following rear panel control outputs; RETRACE BLANK OUT, PENLIFT OUT, and HORIZ OUT. The subsystem commands and parameters are described below:

Table 3-1. :CONTrol Subsystem

Keyword	Parameter Form	Parameter Data or Units	Notes
:CONTrol			
:BLANKing			
:POLarity	<char>	NORMal INVerted	
:PENLift			
:POLarity	<char>	NORMal INVerted	
:RAMP			
:REST	<char>	START STOP	
[:STATe]	<boolean>	1 0 ON OFF	
:TIME	<nv>	numeric value	

:CONTrol:BLANKing:POLarity <char>

:CONTrol:BLANKing:POLarity?

Description: Sets the level of the rear panel RETRACE BLANK OUT blanking signal output during sweep retrace as follows:

NORMal cause the blanking signal to be a +5V level.

INVerted causes the blanking signal to be a –5V level.

Parameters: NORMal|INVerted

Default: NORMal

Examples: :CONTrol:BLANKing:POLarity INVerted

Set a –5V level for the rear panel blanking signal output during sweep retrace.

:CONTrol:BLANKing:POLarity?

Requests the currently programmed level for the rear panel blanking signal output during sweep retrace.

:CONTrol:PENLift:POLarity <char>

:CONTrol:PENLift:POLarity?

Description: Sets the internal penlift relay contacts to control the state of the rear panel PENLIFT OUT signal as follows:

NORMal sets the relay contacts to be normally open.

INVerted sets the relay contacts to be normally closed.

Parameters: NORMal | INVerted

Default: NORMal

Examples: :CONTrol:PENLift:POLarity INVerted

Set the penlift relay contacts to be normally closed.

:CONTrol:PENLift:POLarity?

Requests the currently programmed state of the penlift relay contacts.

:CONTRol:RAMP:REST <char>

:CONTRol:RAMP:REST?

Description: Sets the sweep rest point for the rear panel HORIZ OUT sweep ramp as follows:

START sets the sweep to rest at the bottom of the sweep ramp.

STOP sets the sweep to rest at the top of the sweep ramp.

Parameters: START | STOP

Default: STOP

Examples: :CONTRol:RAMP:REST STOP

Set the sweep to rest at the top of the sweep ramp.

:CONTRol:RAMP:REST?

Requests the currently programmed rest point for the sweep ramp.

:CONTRol:RAMP[:STATe] <boolean>

:CONTRol:RAMP[:STATe]?

Description: Turns the rear panel HORIZ OUT sweep ramp signal on/off.

Parameters: ON | OFF | 1 | 0

Default: OFF

Examples: :CONTRol:RAMP:STATe ON

Turns the rear panel HORIZ OUT sweep ramp signal on.

:CONTRol:RAMP:STATe?

Requests the currently programmed state of the HORIZ OUT sweep ramp signal.

:CONTRol:RAMP:TIME <nv>

:CONTRol:RAMP:TIME?

Description: Sets the rear panel HORIZ OUT sweep ramp signal time by changing the analog sweep time. [:SOURce]:SWEep:TIME will also be changed.

May not be changed while the unit is sweeping.

Parameters: Sweep time (in seconds) | MIN | MAX

Range: 30 ms to 99 sec

Default: 30 ms

Examples: :CONTRol:RAMP:TIME 100 ms

Sets the rear panel HORIZ OUT sweep ramp signal time to 100 ms.

:CONTRol:RAMP:TIME?

Requests the currently programmed time for the HORIZ OUT sweep ramp signal.

3-6 :DIAGnostic Subsystem

The :DIAGnostic subsystem consists of the query command described below:

:DIAGnostic:SNUM?

Description: This query returns the serial number of the instrument.

3-7 :DISPlay Subsystem

The :DISPlay subsystem controls the display of all frequency, power level, and modulation parameters on the front panel data display.

Table 3-2. :DISPlay Subsystem

Keyword	Parameter Form	Parameter Data or Units	Notes
:DISPlay			
[:WINDow]			
:TEXT			
:STATE	<boolean>	1 0 ON OFF	

:DISPlay[:WINDow]:TEXT:STATE <boolean>

:DISPlay:TEXT:STATE?

Description: Turns the display of the frequency, power level, and modulation parameters on the front panel data display on/off.

Parameters: ON | OFF | 1 | 0

Default: ON

Example: :DISPlay:TEXT:STATE OFF

Turns off the display of the frequency, power level, and modulation parameters on the MG369xC front panel data display (Secure mode of operation).

3-8 :INITiate Subsystem

The :INITiate subsystem controls the state of the MG369xC trigger system. The subsystem commands and parameters are described below. The :INITiate commands, along with the :ABORt and :TRIGger commands, comprise the Trigger Group of commands.

Table 3-3. :INITiate Subsystem

Keyword	Parameter Form	Parameter Data or Units	Notes
:INITiate			
[:IMMediate]	none		No Query
CONTinuous	<boolean>	1 0 ON OFF	

:INITiate [:IMMediate]

Description: Places the MG369xC trigger system into the *armed* state from the *idle* state. If trigger system is not in *idle* state, or if :INITiate:CONTinuous is ON, will produce error -213.

Parameters: none

Example: :INITiate:IMMediate

Sets MG369xC trigger to the armed state.

Associated

commands: :ABORt and :TRIGger

Note

When :INITiate or :TSweep is received by the MG369xC, all sweep-related parameters are checked for compatibility and bounds. The system will not arm if any errors exist. These errors are reported in the error queue.

:INITiate:CONTinuous <boolean>

:INITiate:CONTinuous?

Description: Continuously rearms the MG369xC trigger system after completion of a triggered sweep.

Parameters: ON | OFF | 1 | 0

Default: OFF

Example: :INITiate:CONTinuous ON

Sets MG369xC trigger to continuously armed state.

Associated

commands: :ABORt and TRIGger

Note

:INITiate:CONTinuous ON has the same action as :INITiate:IMMediate plus it sets an internal flag that causes the trigger system to rearm after completing a triggered action.
 If :TRIGger:SOURce IMMediate, :INITiate will start a sweep if one is not already in progress. In this case, to abort and restart a sweep either send :ABORt; :INITiate or :TSweep.
 If the trigger system is not idle, :INITiate will cause the error: -213, "Init ignored, trigger not idle"

3-9 :OUTPut Subsystem

The :OUTPut subsystem controls the MG369xC RF output power. The commands are used to turn the RF output power on/off and to set the state of the RF output power during frequency changes in CW and step sweep modes and during sweep retrace. The subsystem commands and parameters are described below:

Table 3-4. :OUTPut Subsystem

Keyword	Parameter Form	Parameter Data or Units	Notes
:OUTPut			
[:STATe]	<boolean>	1 0 ON OFF	
:PROTection	<boolean>	1 0 ON OFF	
:RETRace	<boolean>	1 0 ON OFF	
:IMPedance?			Query Only

:OUTPut[:STATe] <boolean>

:OUTPut[:STATe] ?

Description: Turns MG369xC RF output power on/off.

Parameters: ON | OFF | 1 | 0

Default: OFF (see note below)

Note

The SCPI programming mode reset default for RF output power state is OFF.

The MG369xC Native GPIB programming mode reset default for the RF output power state is ON.

Example: :OUTPut:STATe ON

Turns MG369xC RF output power on.

:OUTPut:PROTection <boolean>

:OUTPut:PROTection?

Description: ON causes the MG369xC RF output to be turned off (blanked) during frequency changes in CW or step sweep mode.

OFF leaves RF output turned on (unblanked).

Parameters: ON | OFF | 1 | 0

Default: ON

Examples: :OUTPut:PROTection OFF

Causes the MG369xC RF output signal to be left on during frequency changes in CW or step sweep mode.

:OUTPut:PROTection?

Requests the currently programmed state of the MG369xC RF output during frequency changes in CW or step sweep mode.

:OUTPut:PROTection:RETRace <boolean>

:OUTPut:PROTection:RETRace?

Description: ON causes the MG369xC RF output to be turned off during sweep retrace.

OFF leaves RF output turned on.

Parameters: ON | OFF | 1 | 0

Default: OFF

Examples: :OUTPut:PROTection:RETRace ON

Turns the MG369xC RF output off during sweep retrace.

:OUTPut:PROTection:RETRace?

Requests the currently programmed state of the MG369xC RF output during sweep retrace.

:OUTPut:IMPedance?

Description: Queries the MG369xC RF output impedance. The impedance is nominally 50 ohms and is not settable.

3-10 [:SOURce] Subsystems

The [:SOURce] subsystem provides control of a majority of the MG369xC functions. The subsystem commands are used to control the frequency, power level, and modulation of the RF output signal. The [:SOURce] subsystem commands and parameters are listed in [Table 3-5](#). The subsystem commands are described in detail on following pages.

Note	Note that the [:SOURce] keyword is optional for all command statements in the :SOURce subsystem.
-------------	--

Table 3-5. [:SOURce] Subsystems (1 of 4)

Keyword	Parameter Form	Parameter Data or Units	Notes
[:SOURce]			
:AM			
:LOGSens	<NRf>	numeric value	
:SENSitivity	<NRf>	numeric value	
:LOGDepth	<NRf>	numeric value	
:INTernal			
:WAVE	<char>	SINE GAUSSian RDOWn RUP SQUARE TRIangle UNIFORM	
:FREQuency	<NRf>	numeric value	
:DEPTh	<NRf>	numeric value	
:SOURce	<char>	INTernal EXTernal1 EXTernal2	
:STATe	<boolean>	1 0 ON OFF	
:TYPE	<char>	LINear LOGarithmic	
:CORRection			
[:STATe]	<boolean>	1 0 ON OFF	
:CSET			
:SElect	<char>	NONE USER1 USER2 USER3 USER4 USER5	
:FM			
:INTernal			
:WAVE	<char>	SINE GAUSSian RDOWn RUP SQUARE TRIangle UNIFORM	
:FREQuency	<NRf>	numeric value	
:DEViation	<NRf>	numeric value	
:MODE	<char>	LOCKed1 LOCKed2 UNLOCKed	
:BWIDth	<nv>	MIN MAX	
:SENSitivity	<NRf>	numeric value	
:SOURce	<char>	INTernal EXTernal1 EXTernal2	
:STATe	<boolean>	1 0 ON OFF	
[:SOURce]			
:FREQuency			

Table 3-5. [:SOURce] Subsystems (2 of 4)

Keyword	Parameter Form	Parameter Data or Units	Notes
[:CW :FIXed]	<nv>	numeric value	
STEP			
[:INCRement]	<NRf>	numeric value	
CENTer	<NRf>	numeric value	
:MODE	<char>	CW FIXed SWEep[1] SWCW ALSW LIST[1] LIST2 LIST3 LIST4	
:SPAN	<NRf>	numeric value	
:FULL			No Query
:SPAN2	<NRf>	numeric value	
:FULL	none		No Query
:START	<nv>	numeric value	
:START2	<nv>	numeric value	
:STOP	<nv>	numeric value	
:STOP2	<nv>	numeric value	
:MULTIplier	<nv>	numeric value	
:LIST<n>			<n> = 1 to 4
:INDEX	<nv>	numeric value	
:FREQuency	<nv>{,<nv>}	numeric value{,numeric value}	
:POINTs?			Query Only
:MODE	<char>	AUTO MANual	
:POWer	<nv>{,<nv>}	numeric value{,numeric value}	
:POINTs?			Query Only
:DWELL	<nv>	numeric value	
:START	<nv>	numeric value	
:STOP	<nv>	numeric value	
:CALCulate			No Query
:MARKer<n>			<n> = 1 to 10
:AOFF			No Query
:FREQuency	<nv>	numeric value	
:STATe	<boolean>	1 0 ON OFF	
:INTensity	<boolean>	1 0 ON OFF	
:VIDeo	<boolean>	1 0 ON OFF	
:POLarity	<char>	POSitive NEGative	
[:SOURce]			
:PM			
:BWIDth	<nv>	MIN MAX	
:DEViation	<NRf>	numeric value	
:INTernal			

Table 3-5. [:SOURce] Subsystems (3 of 4)

Keyword	Parameter Form	Parameter Data or Units	Notes
:WAVE	<char>	SINE GAUSSian RDOWn RUP SQUARE TRIangle UNIFORM	
:FREQuency	<NRf>	numeric value	
:SENSitivity	<NRf>	numeric value	
:SOURce	<char>	INTernal EXTernal1 EXTernal2	
:STATe	<boolean>	1 0 ON OFF	
:POWER			
[:LEVel]			
[:IMMediate]			
[:AMPLitude]	<nv>	numeric value	
:STEP			
[:INCRement]	<NRf>	numeric value	
ALTErnate	<nv>	numeric value	
:ALC			
:SOURce	<char>	INTernal DIODE[1] DIODE2 FIXEd PMETER[1] PMETER2	
:ATTenuation	<nv>	numeric value	
:STEP			
[:INCRement]	<NR1>	numeric value	
:AUTO	<boolean>	1 0 ON OFF	
:DISPlay			
:OFFSet	<NRf>	numeric value	
:STATe	<boolean>	1 0 ON OFF	
:SLOPe	<nv>	numeric value	
:STEP			
[:INCRement]	<NR1>	numeric value	
:STATe	<boolean>	1 0 ON OFF	
:PIVot	<NR1>	numeric value	
:MODE	<char>	CW FIXEd SWEep[1] SWEep2 ALSW LIST[1] LIST2 LIST3 LIST4	
:CENTer	<NRf>	numeric value	
:SPAN	<NRf>	numeric value	
:FULL			No Query
:START	<nv>	numeric value	
:STOP	<nv>	numeric value	
[:SOURce]			
:PULM			
:INTernal			
:FREQuency	<NRf>	numeric value	
:POLarity	<char>	NORMAL INVERTed	

Table 3-5. [:SOURce] Subsystems (4 of 4)

Keyword	Parameter Form	Parameter Data or Units	Notes
:SOURce	<char>	INTernal1 INTernal2 EXTernal1 EXTernal2	
:STATe	<boolean>	1 0 ON OFF	
:PULSe			
:COUNT	<NR1>	numeric value	
:DELay<n>	<NRf>	numeric value	
:PERiod	<NRf>	numeric value	
:WIDTh<n>	<NRf>	numeric value	
:STEP	<boolean>	1 0 ON OFF	
:START	<NRf>	numeric value	
:STOP	<NRf>	numeric value	
:INCRement	<NRf>	numeric value	
:TIME	<nv>	numeric value	
:ROSCillator			
:BANDwidth			
:EXTernal	<integer>	numeric value	
:SOURce?	<char>	INT EXT	Query Only
:SCAN			
:STATe	<boolean>	1 0 ON OFF	
:SWEep<n>		SWEep1 = freq sweep SWEep2 = power sweep	
:DIRection	<char>	UP DOWN	
:DWELl	<nv>	numeric value	
:AUTO	<boolean>	1 0 ON OFF	
:GENeration	<char>	ANALog SWEPped	
:POINTs	<nv>	numeric value	
[:FREQuency]			
:STEP	<nv>	numeric value	
:POWer			
:STEP	<nv>	numeric value	
:SPACing	<char>	LINear LOGarithmic	
:TIME	<nv>	numeric value	
:LLIMint	<nv>	numeric value	
:AUTO	<boolean>	1 0 ON OFF	

[:SOURce]:AM Commands

The [:SOURce]:AM command and its subcommands comprise the AM Subsystem within the :SOURce subsystem. These commands control the Amplitude Modulation function of the MG369xC.

[:SOURce]:AM:LOGSens <NRf>

[:SOURce]:AM:LOGSens?

Description: Sets the AM sensitivity for the external AM Log mode.

Parameters: Sensitivity (in dB/V)

Range: 0 to 25 dB/V

Default: 3 dB/V

Examples: [:SOURce]:AM:LOGSens 20 dB/V

Set the AM sensitivity for the external AM Log mode to 20 dB/V.

[:SOURce]:AM:LOGSens?

Requests the currently programmed AM sensitivity value for the external AM Log mode.

[:SOURce]:AM:SENSitivity <NRf>

[:SOURce]:AM:SENSitivity?

Description: Sets the AM sensitivity for the external AM Linear mode.

Parameters: Sensitivity (in Pct/V)

Range: 0 to 100 %/V

Default: 50 %/V

Examples: [:SOURce]:AM:SENSitivity 80 Pct/V

Set the AM sensitivity for the external AM Linear mode to 80 %/V.

[:SOURce]:AM:SENSitivity?

Requests the currently programmed AM sensitivity value for the external AM Linear mode.

[:SOURce]:AM:LOGDepth <NRf>

[:SOURce]:AM:LOGDepth?

Description: Sets the modulation depth of the AM signal in the internal AM Log mode.

Parameters: Modulation depth (in dB)

Range: 0 to 25 dB

Default: 3 dB

Examples: [:SOURce]:AM:LOGDepth 20 dB

Set the modulation depth in the internal AM Log mode to 20 dB.

[:SOURce]:AM:LOGDepth?

Requests the currently programmed modulation depth value for the internal AM Log mode.

[:SOURce]:AM:INTernal:WAVE <char>

[:SOURce]:AM:INTernal:WAVE?

Description: Selects the modulating waveform (from the internal AM generator) for the internal AM function, as follows:

SINE	=	Sine wave
GAUSSian	=	Guassian noise
RDOWN	=	Negative ramp
RUP	=	Positive ramp
SQUare	=	Square wave
TRIangle	=	Triangle wave
UNIFORM	=	Uniform noise

Parameters: SINE | GAUSSian | RDOWn | RUP | SQUare | TRIangle | UNIFORM

Default: SINE

Examples: [:SOURce]:AM:INTernal:WAVE TRIangle

Selects a triangle wave as the modulating waveform for the internal AM function.

[:SOURce]:AM:INTernal:WAVE?

Requests the currently selected modulating waveform for the internal AM function.

[:SOURce]:AM:INTernal:FREQuency <NRf>

[:SOURce]:AM:INTernal:FREQuency?

Description: Sets the frequency of the modulating waveform for the internal AM function (see :AM:INTernal:WAVE).

Parameters: Frequency

Range: 0.1 Hz to 1 MHz for sine wave

0.1 Hz to 100 kHz for square, triangle, and ramp waveforms

Default: 1 kHz

Examples: [:SOURce]:AM:INTernal:FREQuency 50 kHz

Sets the frequency of the modulating waveform for the internal AM function to 50 kHz.

[:SOURce]:AM:INTernal:FREQuency?

Requests the currently programmed modulating waveform frequency value for the internal AM function.

[:SOURce]:AM:DEPTh <NRf>

[:SOURce]:AM:DEPTh?

Description: Sets the modulation depth of the AM signal in the internal AM Linear mode.

Parameters: Modulation depth (in Pct)

Range: 0 to 100%

Default: 50%

Examples: [:SOURce]:AM:DEPTh 80 Pct

Set the modulation depth in the internal AM Linear mode to 80%.

[:SOURce]:AM:DEPTh?

Requests the currently programmed modulation depth value for the internal AM Linear mode.

[:SOURCE]:AM:SOURCE <char>

[:SOURCE]:AM:SOURCE?

Description: Selects the source of the AM modulating signal, as follows:

INTernal = Internal AM generator
EXTernal1 = Not used
EXTernal2 = Rear panel AM IN connector

Parameters: INTernal | EXTernal1 | EXTernal2

Default: EXTernal1

Examples: [:SOURCE]:AM:SOURCE EXTernal2

Selects the rear panel AM IN connector as the active AM modulating signal source.

[:SOURCE]:AM:SOURCE?

Requests the currently programmed AM modulating signal source.

[:SOURCE]:AM:STATE <boolean>

[:SOURCE]:AM:STATE?

Description: Enable/disable amplitude modulation of MG369xC RF output signal.

Parameters: ON | OFF | 1 | 0

Default: OFF

Examples: [:SOURCE]:AM:STATE ON

Turns amplitude modulation on.

[:SOURCE]:AM:STATE?

Requests currently programmed amplitude modulation state (on/off).

[:SOURCE]:AM:TYPE <char>

[:SOURCE]:AM:TYPE?

Description: Selects the AM operating mode.

Parameters: LINear | LOGarithmic

Default: LINear

Examples: [:SOURCE]:AM:TYPE LOGarithmic

Selects the AM Log mode.

[:SOURCE]:AM:TYPE?

Requests the currently programmed AM operating mode.

[:SOURCE]:CORREction Commands

The [:SOURCE]:CORREction command and its subcommands comprise the Correction Subsystem within the :SOURCE subsystem. These commands are used to select and apply level flatness correction to the MG369xC RF output. (Refer to “Leveling Operations” in Chapter 3 of the MG369xC Operation Manual.)

[:SOURCE]:CORREction[:STATE] <boolean>
[:SOURCE]:CORREction[:STATE]?

Description: Turns the selected user level flatness correction power-offset table on/off.

Parameters: ON | OFF | 1 | 0

Default: OFF

Example: [:SOURCE]:CORREction:STATE ON

Turns on the selected user level correction power-offset table.

Note

If :CORREction:CSET:SElect is NONE, sending the command :CORREction:STATE ON returns an error.

[:SOURCE]:CORREction:CSET:SElect <char>
[:SOURCE]:CORREction:CSET:SElect?

Description: Selects the user level flatness correction power-offset table to be applied to the MG369xC output by the command [:SOURCE]:CORREction:STATE ON.

Parameters: NONE | USER1 | USER2 | USER3 | USER4 | USER5

Default: NONE

Example: [:SOURCE]:CORREction:CSET:SElect USER3

Selects user level flatness correction power-offset table #3.

[:SOURCE]:FM Commands

The [:SOURCE]:FM command and its subcommands comprise the FM Subsystem within the :SOURCE subsystem. These commands control the Frequency Modulation function of the MG369xC.

[:SOURCE]:FM:INTernal:WAVE <char>
[:SOURCE]:FM:INTernal:WAVE?

Description: Selects the modulating waveform (from the internal FM generator) for the internal FM function, as follows:

SINE	=	Sine wave
GAUSSian	=	Guassian noise
RDOWN	=	Negative ramp
RUP	=	Positive ramp
SQUare	=	Square wave
TRIangle	=	Triangle wave
UNIFORM	=	Uniform noise

Parameters: SINE | GAUSSian | RDOWN | RUP | SQUare | TRIangle | UNIFORM

Default: SINE

Examples: [:SOURCE]:FM:INTernal:WAVE SQUare

Selects a square wave as the modulating waveform for the internal FM function.

[:SOURCE]:FM:INTernal:WAVE?

Requests the currently selected modulating waveform for the internal FM function.

[:SOURCE]:FM:INTERNAL:FREQUENCY <NRf>

[:SOURCE]:FM:INTERNAL:FREQUENCY?

Description: Sets the frequency of the modulating waveform for the internal FM function (see :FM:INTERNAL:WAVE).

Parameters: Frequency

Range: 0.1 Hz to 1 MHz for sine wave
0.1 Hz to 100 kHz for square, triangle, and ramp waveforms

Default: 1 kHz

Examples: [:SOURCE]:FM:INTERNAL:FREQUENCY 50 kHz

Sets the frequency of the modulating waveform for the internal FM function to 50 kHz.

[:SOURCE]:FM:INTERNAL:FREQUENCY?

Requests the currently programmed modulating waveform frequency value for the internal FM function.

[:SOURCE]:FM:DEVIATION <NRf>

[:SOURCE]:FM:DEVIATION?

Description: Set the modulation deviation of the FM signal for the internal FM function.

Parameters: Modulation deviation (in Hz)

Range: 10 kHz to 20 MHz in Locked, Locked Low-Noise, and Unlocked Narrow modes
100 kHz to 100 MHz in Unlocked Wide mode

Default: 1 MHz

Examples: [:SOURCE]:FM:DEVIATION 10 MHz

Set the modulation deviation of the FM signal for the internal FM function to 10 MHz.

[:SOURCE]:FM:DEVIATION?

Requests the currently programmed modulation deviation of the FM signal for the internal FM function.

[:SOURce]:FM:MODE <char>

[:SOURce]:FM:MODE?

Description: Sets the synthesis mode employed in generating the FM signal, as follows:

LOCKed[1] = Locked Narrow FM
 LOCKed2 = Locked Narrow Low-Noise FM
 UNLOCKed = Unlocked FM

If LOCKed[1] or LOCKed2 is set, the YIG phase-locked loop is used in synthesizing the FM signal. If UNLOCKed is set, the YIG phase-lock loop is disabled and the FM signal is obtained by applying the modulating signal to the tuning coils of the YIG-tuned oscillator.

Parameters: LOCKed[1] | LOCKed2 | UNLOCKed

Default: UNLOCKed

Examples: [:SOURce]:FM:MODE LOCKed[1]

Set the synthesis mode used to generate the FM signal to Locked Narrow FM.

[:SOURce]:FM:MODE?

Requests the currently programmed synthesis mode used to generate the FM signal.

Note

UNLOCKed FM synthesis mode can be set for wide or narrow mode of operation.
 (See [:SOURce]:FM:BWIDth)

[:SOURce]:FM:BWIDth <nv>

[:SOURce]:FM:BWIDth?

Description: Sets the Unlocked FM synthesis mode to wide or narrow mode of operation.

The Unlocked Wide FM synthesis mode allows maximum deviations of ± 100 MHz for DC to 100 Hz rates.

The Unlocked Narrow FM synthesis mode allows maximum deviations of ± 10 MHz for DC to 8 MHz rates.

Parameters: MIN | MAX

Range: MIN = narrow mode; MAX = wide mode

Default: MIN

Examples: [:SOURce]:FM:BWIDth MAX

Set the Unlocked FM synthesis mode to Wide mode of operation.

[:SOURce]:FM:BWIDth?

Requests the currently programmed Unlocked FM synthesis mode of operation (narrow or wide).

[:SOURce]:FM:SENSitivity <NRf>

[:SOURce]:FM:SENSitivity?

Description: Sets the FM sensitivity for the external FM function.

Parameters: Sensitivity (in Hz/V)

Range: ± 10 kHz/V to ± 20 MHz/V in Locked, Locked Low-Noise, and Unlocked Narrow modes
 ± 100 kHz/V to ± 100 MHz/V in Unlocked Wide mode

Default: 1 MHz/Volt

Examples: [:SOURce]:FM:SENSitivity 20 MHz/V

Set the FM sensitivity for the external FM function to 20 MHz/Volt.

[:SOURce]:FM:SENSitivity?

Requests the currently programmed FM sensitivity for the external FM function.

[:SOURce]:FM:SOURce <char>

[:SOURce]:FM:SOURce?

Description: Selects the source of the FM modulating signal, as follows:

INTernal = Internal FM generator
EXTernal1 = Not used
EXTernal2 = Rear panel FM IN connector

Parameters: INTernal | EXTernal1 | EXTernal2

Default: EXTernal1

Examples: [:SOURce]:FM:SOURce EXTernal2

Selects the rear panel FM IN connector as the active external FM modulating signal source.

[:SOURce]:FM:SOURce?

Requests the currently programmed FM modulating signal source.

[:SOURce]:FM:STATe <boolean>

[:SOURce]:FM:STATe?

Description: Enable/disable frequency modulation of MG369xC RF output signal.

Parameters: ON | OFF | 1 | 0

Default: OFF

Examples: [:SOURce]:FM:STATe ON

Turns frequency modulation on.

[:SOURce]:FM:STATe?

Requests the currently programmed frequency modulation state (on/off).

[:SOURce]:FREQuency Commands

The [:SOURce]:FREQuency command and its subcommands make up the Frequency Subsystem within the :SOURce subsystem. These commands control the frequency characteristics of the MG369xC.

[:SOURce]:FREQuency[:CW|:FIXed] <nv>
[:SOURce]:FREQuency[:CW]?

Description: Sets the RF output frequency of the MG369xC to the value entered.
 Parameters UP | DOWN increment/decrement the frequency by the value set by [:SOURce]:FREQuency:STEP:INCRement command.

Parameters: Frequency (in Hz) | UP | DOWN | MIN | MAX

Range: MIN to MAX (see note below)

Note Keywords :CW and :FIXed are equivalent and may be used interchangeably; they also are optional and may be omitted.
 MIN ≤ frequency ≤ MAX; values for the MINimum and MAXimum frequencies for each MG369xC model are listed in [Table 3-6](#).
 The query [:SOURce]:FREQuency:CW? MAX will return the upper frequency to which the particular model MG369xC may be programmed. Similarly, the query [:SOURce]:FREQuency:CW? MIN will return the lower frequency limit.

Default: (MIN + MAX) / 2

Examples: [:SOURce]:FREQuency:CW 3 GHz *or* :FREQ 3 GHz

Sets the RF output frequency to 3 GHz.

[:SOURce]:FREQuency:CW?

Requests the current value of the frequency parameter.

Table 3-6. Model MG369xC Minimum and Maximum Frequencies

Model	Minimum Frequency*	Maximum Frequency
MG3691C	10 MHz	10 GHz
MG3692C	10 MHz	20 GHz
MG3693C	10 MHz	31.8 GHz
MG3694C	10 MHz	40 GHz
MG3695C	10 MHz	50 GHz
MG3697C	10 MHz	70 GHz

* Minimum frequencies listed with Option 4 or Option 5 installed. Instruments with Option 22 have a minimum frequency of 0.01 Hz.

[:SOURce]:FREQUency[:CW | :FIXed]:STEP[:INCRement] <NRf>

[:SOURce]:FREQUency[:CW]:STEP[:INCRement]?

Description: Sets the step increment size used with the :FREQUency: CW command.

Parameters: Frequency (in Hz)

Range: 0.01 Hz to (MAX – MIN) (see note below)

Default: 0.1 GHz

Examples: [:SOURce]:FREQUency: CW:STEP: INCRement 1 MHz *or* :FREQ:STEP 1 MHz

Set the step increment value for the frequency parameter to 1 MHz.

[:SOURce]:FREQUency: CW:STEP: INCRement? *or* :FREQ:STEP?

Requests the current step increment value of the frequency parameter.

[:SOURce]:FREQUency:CENTer <NRf>

[:SOURce]:FREQUency:CENTer?

Description: Sets the MG369xC RF output center frequency to the value entered. :CENTER and :SPAN frequencies are coupled values. Entering the value for one will cause the other to be recalculated. (See notes under :FREQUency: SPAN)

Parameters: Frequency (in Hz)

Range: MIN to MAX (see notes below)

Stepped Sweep Center Range = MIN to MAX, where:

MIN = MIN + minimum step size

MAX = MAX – minimum step size

Note

Analog Sweep Center Range = MIN to MAX, where;

MIN = MIN + (minimum analog span) / 2

MAX = MAX – (minimum analog span) / 2

Default: (MIN + MAX) / 2

Examples: [:SOURce]:FREQUency:CENTer 4GHz

Set the MG369xC RF output center frequency to 4 GHz.

[:SOURce]:FREQUency:CENTer?

Requests the current value of the RF output center frequency.

[:SOURce]:FREQuency:MODE <char>

[:SOURce]:FREQuency:MODE?

Description: Specifies which command subsystem controls the MG369xC frequency, as follows:

CW|FIXed = [:SOURce]:FREQuency:CW|FIXed
 SWEep[1] = [:SOURce]:SWEep[1] (see notes)
 SWCW = (see notes)
 ALSW = (see notes)
 LIST<n> = [:SOURce]:LIST<n> (see notes)

:SWEep and :SWEep1 may be used interchangeably.

Note

In SWEep[1] mode, frequency will be determined by programmed values for the following :FREQuency subsystem commands: :CENTer and :SPAN, or, :START and :STOP.
 In LIST mode, frequency is determined by programmed values for :LIST<n>:FREQuency, where LIST<n> = LIST[1], LIST2, LIST3, or LIST4. :LIST and :LIST1 may be used interchangeably.
 Setting ALSW will cause the MG369xC to do alternate sweeping when properly triggered.
 Setting FIXed will return CW upon query.
 Setting SWCW will set CW and turn on CW ramp, the same as the command statement :FREQuency:MODE CW;:CONTrol:RAMP ON
 A query returns CW.

Parameters: CW | FIXed | SWEep[1] | SWCW | ALSW | LIST[1] | LIST2 | LIST3 | LIST4

Default: CW

Examples: [:SOURce]:FREQuency:MODE CW

Specifies that the MG369xC RF frequency output is to be controlled by [:SOURce]:FREQuency:CW|FIXed commands.

[:SOURce]:FREQuency:MODE?

Requests the currently selected programming mode for frequency control.

[:SOURce]:FREQuency:SPAN <NRf>

[:SOURce]:FREQuency:SPAN?

Description: Sets sweep span for SWEep[1] to value entered. :SPAN and :CENTer are coupled values (see note below).

Note

:SPAN, :CENTer, :START, and :STOP are coupled values. Entering the value for :SPAN causes the values for :START and :STOP to be recalculated.

At *RST, :SPAN = Fmax – Fmin

Parameters: Frequency (in Hz)

Range: 1 kHz to (MAX – MIN)

Default: MAX – MIN

Examples: [:SOURce]:FREQuency:SPAN 2 GHz *or* :FREQ:SPAN 2 GHz

Set the SWEep[1] sweep span to 2 GHz.

[:SOURce]:FREQuencySPAN:?

Requests the current value for SWEep[1] sweep span.

[:SOURce] :FREQuency :SPAN :FULL

Description: Sets frequency span for SWEep [1] to (MAX – MIN)
(see notes under [:SOURce] :FREQuency :CW | FIXed).

Parameters: None

Example: [:SOURce] :FREQuency :SPAN :FULL
Set the SWEep[1] frequency span to its maximum value.

[:SOURce] :FREQuency :SPAN2 <NRf>**[:SOURce] :FREQuency :SPAN2 ?**

Description: Sets sweep span for the alternate sweep to value entered. :SPAN and :CENTer are coupled values (see note below).

Note

:SPAN, :CENTer, :START, and :STOP are coupled values. Entering the value for :SPAN causes the values for :START and :STOP to be recalculated.

At *RST, :SPAN = Fmax – Fmin

Parameters: Frequency (in Hz)

Range: 1 kHz to (MAX – MIN)

Default: MAX – MIN

Examples: [:SOURce] :FREQuency :SPAN2 2 GHz *or* :FREQ :SPAN2 2 GHz

Set the sweep span for the alternate sweep to 2 GHz.

[:SOURce] :FREQuency SPAN2 : ?

Requests the current value of the sweep span for the alternate sweep.

[:SOURce] :FREQuency :SPAN2 :FULL

Description: Sets frequency span for the alternate sweep to (MAX – MIN)
(see notes under [:SOURce] :FREQuency :CW | FIXed).

Parameters: None

Example: [:SOURce] :FREQuency :SPAN :FULL
Set the frequency span for the alternate sweep to its maximum value.

[:SOURce]:FREQuency:STARt <nv>

[:SOURce]:FREQuency:STARt?

Description: Sets start frequency for SWEep [1] to the value entered. (MIN is defined in the notes under [:SOURce]:FREQuency:CW|FIXed).

Parameters: Frequency (in Hz) | MIN

Range: MIN to MAX (see note)

Note

Stepped Sweep Start Range = MIN to MAX, where: MAX = MAX – 2 × minimum frequency step size
Analog Sweep Start Range = MIN to MAX, where; MAX = MAX – minimum analog span

Default: MIN

Examples: [:SOURce]:FREQuency:STARt 2.5 GHz

Set the start frequency for SWEep[1] to 2.5 GHz .

[:SOURce]:FREQuency:STARt?

Requests the current value for SWEep[1] start frequency.

[:SOURce]:FREQuency:STARt2 <nv>

[:SOURce]:FREQuency:STARt2?

Description: Sets start frequency for the alternate sweep to the value entered. (MIN is defined in the notes under [:SOURce]:FREQuency:CW|FIXed).

Parameters: Frequency (in Hz) | MIN

Range: MIN to MAX (see notes)

Note

Stepped Sweep Start Range = MIN to MAX, where: MAX = MAX – 2 × minimum frequency step size
Analog Sweep Start Range = MIN to MAX, where; MAX = MAX – minimum analog span

Default: MIN

Examples: [:SOURce]:FREQuency:STARt2 3.5 GHz

Set the start frequency for the alternate sweep to 3.5 GHz .

[:SOURce]:FREQuency:STARt2?

Requests the current value for the alternate sweep start frequency.

[:SOURCE]:FREQUENCY:STOP <nv>

[:SOURCE]:FREQUENCY:STOP?

Description: Sets stop frequency for SWEep[1] to the value entered. (MAX is defined in the notes under [:SOURCE]:FREQUENCY:CW|FIXed).

Parameters: Frequency (in Hz) | MAX

Range: MIN to MAX (see notes)

Note

Stepped Sweep Stop Range = MIN to MAX, where: MIN = MIN + 2 × minimum frequency step size
Analog Sweep Stop Range = MIN to MAX, where; MIN = MIN + minimum analog span

Default: MAX

Examples: [:SOURCE]:FREQUENCY:STOP 15 GHz

Set the stop frequency (for SWEep[1] to 15 GHz.

[:SOURCE]:FREQUENCY:STOP?

Requests the current value for SWEep[1] stop frequency.

[:SOURCE]:FREQUENCY:STOP2 <nv>

[:SOURCE]:FREQUENCY:STOP2?

Description: Sets stop frequency for the alternate sweep to the value entered. (MAX is defined in the notes under [:SOURCE]:FREQUENCY:CW|FIXed).

Parameters: Frequency (in Hz) | MAX

Range: MIN to MAX (see notes)

Note

Stepped Sweep Stop Range = MIN to MAX, where: MIN = MIN + 2 × minimum frequency step size
Analog Sweep Stop Range = MIN to MAX, where; MIN = MIN + minimum analog span

Default: MAX

Examples: [:SOURCE]:FREQUENCY:STOP2 13 GHz

Set the stop frequency for the alternate sweep to 13 GHz.

[:SOURCE]:FREQUENCY:STOP2?

Requests the current value for the alternate sweep stop frequency.

[:SOURce]:FREQuency:MULTIplier <nv>

[:SOURce]:FREQuency:MULTIplier?

Description: Sets the value of the reference multiplier for the frequency scaling function. This command affects all entered and displayed frequencies, but it does not affect the output of the instrument.

Parameters: Reference multiplier value | MIN | MAX

Range: 0.1 to 14; MIN = 0.1; MAX = 14

Default: 1

Examples: [:SOURce]:FREQuency:MULTIplier 4

Set the frequency scaling reference multiplier value to 4.

[:SOURce]:FREQuency:MULTIplier?

Requests the current value for the frequency scaling reference multiplier.

Note

The coupling equation is: Entered/Displayed Frequency = (Hardware Frequency x Multiplier)

At *RST, the value is set to 1

[:SOURce]:LIST Commands

The [:SOURce]:LIST command and its subcommands comprise the List Subsystem within the :SOURce subsystem. These commands control the List Sweep function of the MG369xC (see notes below).

[:SOURce]:LIST<n>:INDex <nv>

[:SOURce]:LIST<n>:INDex?

(1 ≤ n ≤ 4 = selected list, see notes)

Description: Sets index of selected list to value entered.

Parameters: List index number | MIN | MAX

Range: 0 to 1999; MIN = 0; MAX = 1999

Default: 0

Examples: [:SOURce]:LIST2:INDex 122

Sets list 2 index number to 122.

[:SOURce]:LIST3:INDex?

Requests the currently programmed index value for list 3.

Note

In list sweep mode, up to four lists (LIST[1], LIST2, LIST3, and LIST4) of 2000 non-sequential frequency/power level sets can be stored and accessed. A list index (0 thru 1999) identifies each frequency/power level set in a list. When commanded, the MG369xC generates a phase-locked stepped sweep of the active list between the specified list start index and list stop index, dwelling at each list index for the specified dwell time.

LIST[1] is the same list that is available via local (front panel) control. This list is stored in non-volatile RAM to preserve any settings after the signal generator is powered off.

LIST2, LIST3, and LIST4 are stored in volatile RAM and all settings are lost when power to the MG369xC is turned off. At power up, these lists are set to their default state of 2000 index entries of 5 GHz at 0 dBm.

```
[:SOURCE]:LIST<n>:FREQUENCY <nv>{, <nv>}
```

```
[:SOURCE]:LIST<n>:FREQUENCY?
```

($1 \leq n \leq 4$ = selected list)

Description: Sets the list frequencies of the selected list starting at the list index specified by :LIST<n>:INDEX or at list index 0 if no list index is specified.

Any number of frequencies can follow this command.

Parameter: Frequency (in Hz) | MIN | MAX

Range: MIN to MAX (see notes below)

Note	MIN ≤ frequency ≤ MAX; values for the MINimum and MAXimum frequencies for each MG369xC model are listed in Table 3-6 on page 3-23 .
-------------	---

Default: 5 GHz

Examples: [:SOURCE]:LIST2:INDEX 122; [:SOURCE]:LIST2:FREQUENCY 8 GHz, 3 GHz, 11 GHz

Sets list 2 index to 122, then sets the frequency of list index 122 to 8 GHz, list index 123 to 3 GHz, and list index 124 to 11 GHz.

```
[:SOURCE]:LIST3:INDEX 1235
```

```
[:SOURCE]:LIST3:FREQUENCY?
```

Requests the frequency set at list 3 index 1235.

[:SOURce]:LIST<n>:FREQuency:POINts?

(1 ≤ n ≤ 4 = selected list)

Description: Queries the number of frequency points in the selected list. The number of points is 2000 and is not settable.

Parameters

Returned: number of frequency points in <NR1> format

[:SOURce]:LIST<n>:MODE <char>**[:SOURce]:LIST<n>:MODE?**

(1 ≤ n ≤ 4 = selected list)

Description: Sets the list trigger mode. When automatic trigger mode is selected, the output sweeps between the specified list start and stop indexes, dwelling at each list index for the specified dwell time. In manual trigger mode, the list index can be incremented using an external TTL trigger input, a Group Execute Trigger (GET), or a *TRG command. Each trigger increments the list index by one between the specified list start and stop indexes. When manual trigger mode is selected, the list index can also be incremented or decremented using the front panel cursor control keys.

Parameters: AUTO | MANual

AUTO selects automatic trigger mode.

MANual selects manual trigger mode.

Default: AUTO

Examples: [:SOURce]:LIST2:MODE AUTO

Sets the list 2 list trigger mode to automatic.

[:SOURce]:LIST2:MODE?

Queries the current list 2 list trigger mode.

[:SOURce]:LIST<n>:POWer <nv>{,<nv>}**[:SOURce]:LIST<n>:POWer?**

(1 ≤ n ≤ 4 = selected list)

Description: Sets the list power levels of the selected list starting at the list index specified by :LIST<n>:INDEX or at list index 0 if no list index is specified.

Parameters: Power level (in dBm) | MIN | MAX

Range: MIN to MAX (see note below)

Note

MIN ≤ power level ≤ MAX; values for MINimum and MAXimum power levels for each MG369xC model are listed in [Table 3-8 on page 3-40](#).

Default: 0 dBm

Examples: [:SOURce]:LIST2:INDEX 122;[:SOURce]:LIST2:POWer 2 dBm, 9 dBm, -3 dBm

Sets list 2 index to 122, then sets the power level of list index 122 to 2 dBm, list index 123 to 9 dBm, and list index 124 to -3 dBm.

[:SOURce]:LIST3:INDEX 1235

[:SOURce]:LIST3:POWer?

Requests the power level set at list 3 index 1235.

[:SOURce] :LIST<n> :POWER :POINTS?

(1 ≤ n ≤ 4 = selected list)

Description: Queries the number of power level points in the selected list. The number of points is 2000 and is not settable.

Parameters

Returned: Number of power level points in <NR1> format.

[:SOURce] :LIST<n> :DWELL <nv>**[:SOURce] :LIST<n> :DWELL?**

(1 ≤ n ≤ 4 = selected list)

Description: Sets the dwell time for each step in a sweep of the selected list to the value entered.

Parameters: Dwell time (in seconds) | MIN | MAX

Range: 0 ms to 99 sec

Default: 50 ms

Examples: [:SOURce] :LIST2 :DWELL 30 ms

Set dwell time for each step in a sweep of list 2 to 30 ms.

[:SOURce] :LIST3 :DWELL?

Requests the currently programmed value for list 3 sweep step dwell time.

[:SOURce] :LIST<n> :START <nv>**[:SOURce] :LIST<n> :START?**

(1 ≤ n ≤ 4 = selected list)

Description: Sets list start index for a sweep of the selected list to the value entered.

Parameters: List index number | MIN | MAX

Range: 0 to 1999; MIN = 0; MAX = 1999

Default: 0

Examples: [:SOURce] :LIST2 :START 85

Set the list start index for a sweep of list 2 to 85.

[:SOURce] :LIST3 :START?

Requests the currently programmed value for the list start index for a sweep of list 3.

[:SOURce] :LIST<n> :STOP <nv>**[:SOURce] :LIST<n> :STOP?**

(1 ≤ n ≤ 4 = selected list)

Description: Sets list stop index for a sweep of the selected list to the value entered.

Parameters: List index number | MIN | MAX

Range: 0 to 1999; MIN = 0; MAX = 1999

Default: 1999

Examples: [:SOURce] :LIST2 :STOP 1035

Set the list stop index for a sweep of list 2 to 1035.

[:SOURce] :LIST3 :STOP?

Requests the currently programmed value for the list stop index for a sweep of list 3.

[:SOURce]:LIST<n>:CALCulate

($1 \leq n \leq 4$ = selected list)

Description: Performs all the calculations necessary to set the frequencies and power levels for the active list (see notes below).

Note

During the initial sweep of the active list, the MG369xC performs calculations to set the frequencies and power levels. This causes the initial list sweep to take longer than each subsequent sweep. The command, [:SOURce]:LIST<n>:CALCulate (where n = the active list), initiates a process that examines every index in the active list and performs all calculations necessary to set the frequencies and power levels. This lets the initial list sweep be as fast as each subsequent sweep.

The list calculations are for the current index entries of the active list only. Any changes to the index entries of the active list or the selection of a different active list requires the calculations to be performed again.

The calculations are stored only in volatile RAM and are lost when power to the MG369xC is turned off.

Parameters: none

Example: [:SOURce]:FREQuency:MODE LIST2
[:SOURce]:LIST2:CALCulate

Performs the calculations to set the frequencies and power levels for list 2.

[:SOURCE]:MARKer Commands

The [:SOURCE]:MARKer command and its subcommands comprise the Marker Subsystem within the :SOURCE subsystem. These commands control the Frequency Marker function of the MG369xC.

[:SOURCE]:MARKer<n>:AOFF

Note

A numeric suffix <n> may be appended to any of the MARKer command headers. This specifies which of the 10 markers (1 to 10) is being altered by the command. In some cases, the command function is global to all 10 markers, such as the command :MARKers<n>:AOFF (all markers off). In these cases, the <n> in the command header is ignored.

Description: Turns all markers off. This command is an event, therefore there is no data parameter and no query form.

Parameters: none

Examples: [:SOURCE]:MARKer:AOFF

Turns all markers off.

[:SOURCE]:MARKer<n>:FREQuency <nv>

[:SOURCE]:MARKer<n>:FREQuency?

(1 ≤ n ≤ 10 = selected marker, see note)

Description: Sets frequency of selected marker to value entered. (MIN and MAX are defined in the notes under [:SOURCE]:FREQuency:CW|FIXed).

Parameters: Frequency (in Hz) | MIN | MAX

Range: MIN to MAX

Default: See default values in notes below

Examples: [:SOURCE]:MARKer3:FREQuency 2 GHz

Set the frequency of marker #3 to 2 GHz.

[:SOURCE]:MARKer5:FREQuency?

Requests the current frequency value of marker #5.

Note

:MARKer (or :MARK) is equivalent to :MARKer1 (or :MARK1).

When the MG369xC is powered up, or when *RST command is issued, markers 1 – 10 are set to their default frequency values as shown in [Table 3-7](#).

Marker 10 accesses what is displayed as Marker 0 on the MG369xC front panel data display.

Table 3-7. Marker Default Frequency Values

Marker:	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10
Freq (GHz):	2.0	8.4	2.0	5.0	8.4	8.4	8.4	8.4	8.4	3.5

[:SOURce]:MARKer<n>:STATe <boolean>

[:SOURce]:MARKer<n>:STATe?

($1 \leq n \leq 10$ = selected marker)

Description: Turns selected marker on/off (tags/untags the selected marker).

Parameters: ON | OFF | 1 | 0

Default: OFF

Examples: [:SOURce]:MARKer4:STATe ON

Turn marker #4 on.

[:SOURce]:MARKer7:STATe?

Requests current state of marker #7.

[:SOURce]:MARKer<n>:INTensity <boolean>

[:SOURce]:MARKer:INTensity?

Description: Turns intensity markers on/off. Intensity markers are available only in analog sweep frequency mode; i.e. only with :SWEep [1] mode of operation (refer to :FREQuency:MODE command). For further information about frequency markers, refer to Frequency Markers in Chapter 3 of the Series MG369xC Operation Manual.

Parameters: ON | OFF | 1 | 0

Default: OFF

Example: [:SOURce]:MARKer:INTensity ON

Turns on intensity markers. Turns off video markers if previously active.

Note

[:SOURce]:MARKer:INTensity and [:SOURce]:MARKer:VIDeo are mutually exclusive commands. That is, turning on one turns the other off. Turning both off turns off all markers.

[:SOURce]:MARKer<n>:VIDeo <boolean>

[:SOURce]:MARKer:VIDeo?

Description: Turns video markers on/off. Video markers are available in both analog and step sweep frequency mode. For further information about frequency markers, refer to Frequency Markers in Chapter 3 of the Series MG369xC Operation Manual.

Parameters: ON | OFF | 1 | 0

Default: OFF

Example: [:SOURce]:MARKer:VIDeo ON

Turns on video markers. Turns off intensity markers if previously active.

Note

[:SOURce]:MARKer:INTensity and [:SOURce]:MARKer:VIDeo are mutually exclusive commands. That is, turning on one turns the other off. Turning both off turns off all markers.

[:SOURCE]:MARKer<n>:POLarity <char>

[:SOURCE]:MARKer:POLarity?

Description: Selects +5V or –5V pulse output for each video marker as follows:

POSitive selects a +5V pulse output for each marker.

NEGative selects a –5V pulse output for each marker. This command is active only in the video marker mode (see :MARKer:VIDeo command).

Parameters: POSitive | NEGative

Default: POSitive

Examples: [:SOURCE]:MARKer:POLarity NEGative

Specifies –5V pulse output for each video marker.

[:SOURCE]:MARKer:POLarity?

Requests current polarity of the pulse output for each video marker.

[:SOURce]:PM Commands

The [:SOURce]:PM command and its subcommands comprise the Phase Modulation Subsystem with the :SOURce subsystem. These commands control the phase modulation subsystem function of the MG369xC.

Note

If the phase modulation function is not installed in the instrument, the PM command and its subcommands produce syntax errors.

[:SOURce]:PM:BWIDth <nv>

[:SOURce]:PM:BWIDth?

Description: Selects the phase modulation (Φ M) operating mode.

The Narrow Φ M mode allows maximum deviations of ± 3 radians for DC to 8 MHz rates.

The Wide Φ M mode allows maximum deviations of ± 400 radians for DC to 1 MHz rates.

Parameters: MIN | MAX

Range: MIN = narrow mode

MAX = wide mode

Default: MIN

Examples: [:SOURce]:PM:BWIDth MAX

Selects the phase modulation wide operating mode.

[:SOURce]:PM:BWIDth?

Requests the currently programmed phase modulation operating mode.

[:SOURce]:PM:DEVIation <NRf>

[:SOURce]:PM:DEVIation?

Description: Set the modulation deviation of the Φ M signal for the internal phase modulation function.

Parameters: Modulation deviation (in radians)

Range: 0.0025 to 5.0 radians in narrow mode

0.25 to 500.0 radians in wide mode

Default: 1.0000 radians

Examples: [:SOURce]:PM:DEVIation 1.25 RAD

Set the modulation deviation for the internal phase modulation function to 1.25 radians.

[:SOURce]:PM:DEVIation?

Requests the currently programmed modulation deviation value for the internal phase modulation function.

[:SOURCE]:PM:INTernal:WAVE <char>

[:SOURCE]:PM:INTernal:WAVE?

Description: Selects the modulating waveform (from the internal Φ M generator) for the internal phase modulation function, as follows:

SINE = Sine wave
 GAUSSian = Gaussian noise
 RDOWn = Negative ramp
 RUP = Positive ramp
 SQUare = Square wave
 TRIangle = Triangle wave
 UNIFORM = Uniform noise

Parameters: SINE | GAUSSian | RDOWn | RUP | SQUare | TRIangle | UNIFORM

Default: SINE

Examples: [:SOURCE]:PM:INTernal:WAVE TRIangle

Selects a triangle wave as the modulating waveform for the internal phase modulation function.

[:SOURCE]:PM:INTernal:WAVE?

Requests the currently selected modulating waveform for the internal phase modulation function.

[:SOURCE]:PM:INTernal:FREQuency <NRf>

[:SOURCE]:PM:INTernal:FREQuency?

Description: Sets the frequency of the modulating waveform for the internal phase modulation (see :PM:INTernal:WAVE)

Parameter: Frequency (in Hz)

Range: 0.1 Hz to 1 MHz for sine wave;
 0.1 Hz to 100 kHz for square, triangle, and ramp waveforms.

Default: 1 kHz

Examples: [:SOURCE]:PM:INTernal:FREQuency 55 kHz

Sets the frequency of the modulating waveform for the internal phase modulation function to 55 kHz.

[:SOURCE]:PM:INTernal:FREQuency?

Requests the currently programmed modulating waveform frequency value for the internal phase modulation function.

[:SOURce]:PM:SENSitivity <NRf>

[:SOURce]:PM:SENSitivity?

Description: Sets the Φ M sensitivity for the external phase modulation function.

Parameters: Sensitivity (in radians/V)

Range: ± 0.0025 to ± 5.0 radians/V in narrow mode
 ± 0.25 to ± 500.0 radians/V in wide mode

Default: 1.0000 radians/V

Examples: [:SOURce]:PM:SENSitivity 1.25 RAD/V

Sets the Φ M sensitivity for the external phase modulation function to 1.25 radians/volt.

[:SOURce]:PM:SENSitivity?

Requests the currently programmed Φ M sensitivity for the external phase modulation function.

[:SOURce]:PM:SOURce <char>

[:SOURce]:PM:SOURce?

Description: Selects the source of the Φ M modulating signal, as follows:

INTernal = Internal Φ M generator
 EXTernal1 = Not used
 EXTernal2 = Rear Panel FM/ Φ M IN connector

Parameters: INTernal | EXTernal1 | EXTernal2

Default: EXTernal1

Examples: [:SOURce]:PM:SOURce EXTernal2

Selects the rear panel FFM/ Φ M IN connector as the active external Φ M modulating signal source.

[:SOURce]:PM:SOURce?

Requests the currently programmed Φ M modulating signal source.

[:SOURce]:PM:STATe <boolean>

[:SOURce]:PM:STATe?

Description: Enable/disable phase modulation of the MG369xC RF output signal.

Parameters: ON | OFF | 1 | 0

Default: OFF

Examples: [:SOURce]:PM:STATe ON

Turns phase modulation on.

[:SOURce]:PM:STATe?

Requests the currently programmed phase modulation state (on/off)

[:SOURCE]:POWER Commands

The [:SOURCE]:POWER command and its subcommands comprise the Power Subsystem within the :SOURCE subsystem. These commands control the RF power output level of the MG369xC.

```
[:SOURCE]:POWER[:LEVEL][:IMMEDIATE][:AMPLITUDE] <nv>
[:SOURCE]:POWER[:LEVEL][:IMMEDIATE][:AMPLITUDE]?
```

Description: Sets the power level of the unswept RF output signal (see notes below).

Parameters: Power level (in dBm) | UP | DOWN | MIN | MAX

Range: MIN to MAX (see notes below)

Default: 0 dBm

Examples: [:SOURCE]:POWER:LEVEL:IMMEDIATE:AMPLITUDE 10 dBm

Sets the RF output power level to +10 dBm.

```
[:SOURCE]:POWER:LEVEL:IMMEDIATE:AMPLITUDE?
```

Requests the value currently programmed for the RF output power level.

```
[:SOURCE]:POWER? MAX
```

Requests the maximum RF output power level value that can be programmed for the particular MG369xC model.

Note

The MINimum and MAXimum RF output power levels that can be entered for each MG369xC model are listed in [Table 3-8](#). Note that these power levels are the limits of what can be entered, *but do not guarantee leveled operation*.

For units equipped with an attenuator (Option 2), changes in RF power output level >10 dB may change attenuator setting (see command [:SOURCE]:POWER:ATTenuation:AUTO).

Use related command :OUTPUT[:STATE] ON|OFF to turn the MG369xC RF power output on/off.

Table 3-8. Model MG369xC Minimum and Maximum Settable Power Levels

Model	W/O Step Attenuator ⁽¹⁾		W/Step Attenuator (Option 2) ⁽²⁾	
	Minimum	Maximum	Minimum	Maximum
MG3691C	-20 dBm	+30 dBm	-130 dBm	+30 dBm
MG3692C	-20 dBm	+30 dBm	-130 dBm	+30 dBm
MG3693C	-20 dBm	+30 dBm	-130 dBm	+30 dBm
MG3694C	-20 dBm	+30 dBm	-130 dBm	+30 dBm
MG3695C	-20 dBm	+30 dBm	-110 dBm	+30 dBm
MG3697C	-20 dBm	+30 dBm	-110 dBm	+30 dBm
With Option 15 (High Power) Installed				
MG3691C	-20 dBm	+30 dBm	-130 dBm	+30 dBm
MG3692C	-20 dBm	+30 dBm	-130 dBm	+30 dBm
MG3693C	-20 dBm	+30 dBm	-130 dBm	+30 dBm
MG3694C	-20 dBm	+30 dBm	-130 dBm	+30 dBm
MG3695C	-20 dBm	+30 dBm	-110 dBm	+30 dBm
MG3697C	-20 dBm	+30 dBm	-110 dBm	+30 dBm

Notes:

The power levels listed above represent the maximum and minimum front panel settings. Refer to the Series MG369xC Synthesized Signal Generators Operation Manual for instrument performance specifications.

1. Or units with a step attenuator (Option 2) and :POWER:ATTenuation:AUTO OFF

2. :POWER:ATTenuation:AUTO ON

```
[:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude]:STEP[:INCRement] <NRf>
[:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude]:STEP[:INCRement]?
```

Description: Sets the step increment size used with the :POWer:LEVel:IMMediate:AMPLitude command.

Parameters: Power level (in dB)

Range: Model dependent (see notes below)

Default: 0.1 dB

Examples: [:SOURce]:POWer:LEVel:IMMediate:AMPLitude:STEP:INCRement 5 dBm

Set the step increment value for RF output power level parameter to 5 dBm.

[:SOURce]:POWer:LEVel:IMMediate:AMPLitude:STEP:INCRement?

Requests the current step increment value for the RF output power level parameter.

Note

To determine the maximum step size for your instrument, refer to Appendix B—Performance Specification, in the Series MG369xC Operation Manual.

```
[:SOURce]:POWer[:LEVel]:ALTErnate <nv>
[:SOURce]:POWer[:LEVel]:ALTErnate?
```

Description: Sets the RF output power level for the alternate sweep if the command :POWer:MODE ALSW is set.

Parameters: Power level (in dBm) | MIN | MAX

Range: MIN to MAX (see notes below)

Default: 0 dBm

Examples: [:SOURce]:POWer:LEVel:ALTErnate 5 dBm

Sets the RF output power level for the alternate sweep to +5 dBm.

[:SOURce]:POWer:LEVel:ALTErnate?

Requests the currently programmed value for the alternate sweep RF output power level.

Note

The MINimum and MAXimum RF output power levels that can be entered for each MG369xC model are listed in [Table 3-8 on page 3-40](#). Note that these power levels are the limits of what can be entered, but do not guarantee leveled operation.

For units equipped with an attenuator (Option 2), changes in RF power output level > 10 dB may change attenuator setting (see command [:SOURce]:POWer:ATTenuation:AUTO).

Use related command :OUTPut[:STATe] ON|OFF to turn the MG369xC RF power output on/off.

[:SOURce]:POWER:ALC:SOURce <char>

[:SOURce]:POWER:ALC:SOURce?

Description: Selects (1) whether the ALC loop controls the output power level and (2) the source of the feedback signal for the ALC.

FIXed places the instrument in a fixed gain power level mode (ALC off). RF output power is unlevelled; use the [:SOURce]:POWER[:LEVEl][:IMMediate][:AMPLitude] command to set the output power to the desired level.

The other parameters turn on the ALC function and select the source of the ALC feedback signal. INTernal specifies the ALC feedback signal from the instrument's internal level detector. The remaining parameter choices select an external detector or power meter signal as the feedback signal source for the ALC, as follows:

DIODe[1] = Detector output connected to the front panel EXTERNAL ALC input

PMETer[1] = Power Meter output connected to the front panel EXTERNAL ALC input

DIODe2 = Detector output connected to the rear panel EXT ALC IN input

PMETer2 = Power Meter output connected to the rear panel EXT ALC IN input

Parameters: INTernal | DIODe[1] | DIODe2 | PMETer[1] | PMETer2 | FIXed

Default: INTernal

Examples: [:SOURce]:POWER:ALC:SOURce PMETer2

Select external power meter output connected to the rear panel EXT ALC IN input as the feedback signal for the ALC.

[:SOURce]?:POWER:ALC:SOURce?

Requests the currently programmed source of the feedback signal for the ALC.

[:SOURce]:POWER:ATTenuation <nv>

[:SOURce]:POWER:ATTenuation?

Description: This command applies only to MG369xCs equipped with an internal step attenuator (Option 2). This command sets the step attenuator (in 10 dB increments) throughout its 110 dB range (90 dB range for MG3695C and MG3697C models).

This command decouples the step attenuator from the automatic leveling control (ALC) system (see the command :POWER:ATTenuation:AUTO OFF).

Parameters: Attenuation (in dB) | UP | DOWN | MIN | MAX

Range: 0 – 110 dB
(0 – 90 dB for MG3695C and MG3697C models)

Default: 0 dB

Examples: [:SOURce]:POWER:ATTenuation 100

Sets the step attenuator setting to 100 dB.

[:SOURce]:POWER:ATTenuation?

Requests the currently programmed value for the step attenuator setting.

Warning

If POWER:ATTenuation:AUTO was set ON, the command POWER:ATTenuation <arg> will momentarily set output power and attenuation to 0 before going to the programmed attenuation. If this could possibly damage equipment, set :OUTput[:STATE] OFF before issuing this command.

Note

Values entered for parameter <dB> must be exact multiples of 10 db. Parameters UP and DOWN can be used to increase/decrease the step attenuator setting in 10 dB steps.

MIN = 0 dB attenuation, MAX = 110 dB (or 90 dB) attenuation.

If the MG369xC does not have a step attenuator installed, sending this command will result in the error message:

–241,"Hardware missing;Attenuator" .

[:SOURce]:POWER:ATTenuation:STEP[:INCRement] <NR1>

[:SOURce]:POWER:ATTenuation:STEP[:INCRement]?

Description: Sets the attenuator step increment size used with the :POWER:ATTenuation command. The step size can only be set to 0 dB or 10 dB.

Parameters: Attenuator step increment size (in dB)

Range: 0 dB | 10 dB

Default: 10 dB

Examples: [:SOURce]:POWER:ATTenuation:STEP:INCRement 10

Sets the step increment value to 10 dB.

[:SOURce]:POWER:ATTenuation:STEP:INCRement?

Requests the current step increment size for the step attenuator.

[:SOURce]:POWER:ATTenuation:AUTO <boolean>

[:SOURce]:POWER:ATTenuation:AUTO?

Description: *This command applies only to MG369xCs equipped with an internal step attenuator (Option 2). Setting to ON couples the step attenuator to the ALC system; setting to OFF decouples the step attenuator from the ALC system.*

Parameters: ON | OFF | 1 | 0

Default: ON

Examples: [:SOURce]:POWER:ATTenuation:AUTO ON
Couple the step attenuator to the ALC system.

[:SOURce]:POWER:ATTenuation:AUTO?

Requests the currently programmed coupling state for the step attenuator.

Warning	POWER:ATTenuation:AUTO OFF ON sets the output power and attenuation to 0. If this could possibly damage equipment, set :OUTput[:STATe] OFF before issuing this command.
----------------	---

Note	In MG369xCs without a step attenuator, this command's value is always OFF. Attempting to set it ON results in the error message: -241,"Hardware missing;Attenuator".
-------------	---

[:SOURce]:POWER:DISPlay:OFFSet <NRf>

[:SOURce]:POWER:DISPlay:OFFSet?

Description: Sets the offset value for the power level display offset function (see :POWER:DISPlay:OFFSet:STATe).

Parameters: Power level display offset (in dB)

Range: -100.00 to +100.00 dB

Default: 0 dB

Examples: [:SOURce]:POWER:DISPlay:OFFSet 3 dB

Sets the power level display offset value to +3 dB.

[:SOURce]:POWER:DISPlay:OFFSet?

Requests the current power level display offset value.

[:SOURce]:POWER:DISPlay:OFFSet:STATe <boolean>

[:SOURce]:POWER:DISPlay:OFFSet:STATe?

Description: Turns the power level display offset function on/off. When the function is turned on, the offset value, set by the command :POWER:DISPlay:OFFSet <arg>, is added to the displayed RF output power level. A negative offset value decreases the displayed power level.

Parameters: ON | OFF | 1 | 0

Default: OFF

Examples: [:SOURce]:POWER:DISPlay:OFFSet:STATe ON

Turns on the power level display offset function.

[:SOURce]:POWER:DISPlay:OFFSet:STATe?

Requests the current state of the power level display offset function.

[:SOURce]:POWER:SLOPe <nv>

[:SOURce]:POWER:SLOPe?

Description: Sets the value of the slope characteristic parameter for the ALC power slope function (refer to “Leveling Operations” in Chapter 3 of the MG369xC Operation Manual). This parameter is a relative number that defines the slope characteristic; a value of 128 produces a 0 dB/V slope. Parameters UP | DOWN increment/decrement the characteristic parameter by the value set by the [:SOURce]:POWER:SLOPe:STEP[:INCRement] command.

Parameters: Slope characteristic value | UP | DOWN | MIN | MAX |

DEF

Range: 0 to 255; MIN = 0; MAX = 255

Default: 128

Examples: [:SOURce]:POWER:SLOPe 140

Sets the value of the slope characteristic for the power slope function to 140.

[:SOURce]:POWER:SLOPe?

Requests the current value of the slope characteristic for the power slope function.

[:SOURce]:POWER:SLOPe:STEP[:INCRement] <NR1>

[:SOURce]:POWER:SLOPe:STEP[:INCRement]?

Description: Sets the step increment size used with the :POWER:SLOPe command (ALC power slope function).

Parameters: Slope step increment size

Range: 0 to 255

Default: 1

Examples: [:SOURce]:POWER:SLOPe:STEP:INCRement 5

Sets the step increment size to 5 for the ALC power slope function parameter.

[:SOURce]:POWER:SLOPe:STEP:INCRement?

Requests the current step increment size for the ALC power slope parameter.

[:SOURCE]:POWER:SLOPE:STATE <boolean>

[:SOURCE]:POWER:SLOPE:STATE?

Description: Turns ALC power slope function on/off (refer to “Leveling Operations” in Chapter 3 of the MG369xC Operation Manual).

Parameters: ON | OFF | 1 | 0

Default: OFF

Example: [:SOURCE]:POWER:SLOPE:STATE ON

Turns the ALC power slope function on.

[:SOURCE]:POWER:SLOPE:PIVot <NR1>

[:SOURCE]:POWER:SLOPE:PIVot?

Description: Sets the frequency where the ALC power slope function correction is zero (pivot point). The frequency range for this function is model dependent (see notes under [:SOURCE]:FREQUENCY[:CW|:FIXed]).

Parameters: Frequency (in Hz)

Range: Frequency range of the MG369xC model

Default: 2 GHz

Examples: [:SOURCE]:POWER:SLOPE:PIVot 5 GHz

Sets the ALC power slope function pivot point to 5 GHz.

[:SOURCE]:POWER:MODE <char>

[:SOURCE]:POWER:MODE?

Description: Specifies which set of commands controls the MG369xC RF output power level determining function, as follows:

CW FIXed	=	[:SOURCE]:POWER[:LEVEL] [:IMMEDIATE][:AMPLITUDE]
SWEep[1]	=	[:SOURCE]:SWEep[1] (see notes)
SWEep2	=	[:SOURCE]:SWEep2 (see notes)
ALSW	=	[:SOURCE]:POWER[:LEVEL]:ALTERNATE
LIST<n>	=	[:SOURCE]:LIST<n> (see notes)

Parameters: CW | FIXed | SWEep[1] | SWEep2 | ALSW | LIST[1] | LIST2 | LIST3 | LIST4

Default: FIXed

Examples: [:SOURCE]:POWER:MODE CW

Specifies that the MG369xC RF output power level is to be controlled by [:SOURCE]:POWER:CW|FIXed commands.

[:SOURCE]:POWER:MODE?

Requests the currently selected programming mode for RF output power level control.

Note

In SWEep[1] and SWEep2 modes, RF output power level is determined by programmed values for the following :POWER subsystem commands: :CENTER and :SPAN, or :START and :STOP.

In LIST mode, RF output power level is determined by programmed values for :LIST<n>:POWER where LIST<n> is LIST[1], LIST2, LIST3, or LIST4. :LIST and :LIST1 may be used interchangeably.

When [:SOURCE]:POWER:MODE is set to CW, the query [:SOURCE]:POWER:MODE? will return FIXed.

[:SOURce]:POWER:CENTer <NRf>

[:SOURce]:POWER:CENTer?

Description: Sets the RF output power level at the center of the power sweep to the value entered. See notes below.

Parameters: Power level (in dbm)

Range: MIN to MAX (see note below)

Default: $(\text{MIN} + \text{MAX}) / 2$

Examples: [:SOURce]:POWER:CENTer -20 dBm

Set the RF output power level at the center of the power sweep span to -20 dBm.

[:SOURce]:POWER:CENTer?

Requests the currently programmed value for the RF output power level at the center of the power sweep span.

Note

The MINimum and MAXimum RF output power levels that can be entered for each MG369xC model are listed in [Table 3-8 on page 3-40](#). Note that these power levels are the limits of what can be entered, but *do not guarantee leveled operation*.

:CENTer and :SPAN are coupled values. Setting the value for one will cause the other to be recalculated. See notes for the command :POWER:SPAN.

[:SOURce]:POWER:SPAN <NRf>

[:SOURce]:POWER:SPAN?

Description: Sets sweep span for power sweep to value entered. See notes below.

Parameters: Power level (in dBm)

Range: MIN to MAX (see note below).

Default: Leveled power span of the instrument

Examples: [:SOURce]:POWER:SPAN 10 dBm

Set the power sweep span to 10 dBm.

[:SOURce]:POWER:SPAN:?

Requests the current value for power sweep span.

Note

:SPAN, :CENTer, :START, and :STOP are coupled values. Entering the value for :SPAN cause the values for :START and :STOP to be recalculated.

When the MG369xC is powered up, or when *RST command is issued, :SPAN is set to the leveled power span of the instrument, and :CENTer is set to $(\text{MIN} + \text{MAX}) / 2$.

The MINimum and MAXimum RF output power levels that can be entered for each MG369xC model are listed in [Table 3-8 on page 3-40](#). Note that these power levels are the limits of what can be entered, but *do not guarantee leveled operation*.

[:SOURce]:POWER:SPAN:FULL

Description: Sets the power sweep span to $(\text{MAX} - \text{MIN})$. See notes under :POWER:SPAN command.

Parameters: none

Example: [:SOURce]:POWER:SPAN:FULL

Set the power sweep span to its maximum value.

[:SOURce]:POWER:START <nv>

[:SOURce]:POWER:START?

Description: Sets start RF output power level for the power sweep to the value entered. See notes under :POWER:SPAN command.

Parameters: Power level (in dBm) | MIN

Range: MIN to MAX

Default: MIN

Examples: [:SOURce]:POWER:START -10 dBm

Set the start RF output power level for the power sweep to -10 dBm.

[:SOURce]:POWER:START?

Requests the currently programmed start RF output power level for the power sweep.

[:SOURce]:POWER:STOP <nv>

[:SOURce]:POWER:STOP?

Description: Sets stop RF output power level for the power sweep to the value entered. See notes under :POWER:SPAN command.

Parameters: Power level (in dBm) | MAX

Range: MIN to MAX

Default: MAX

Examples: [:SOURce]:POWER:STOP 10 dBm

Set the stop RF output power level for the power sweep to +10 dBm.

[:SOURce]:POWER:STOP?

Requests the currently programmed stop RF output power level for the power sweep.

[:SOURce]:PULM Commands

The [:SOURce]:PULM command and its subcommands make up the Pulse Modulation Subsystem within the :SOURce subsystem. These commands control the pulse modulation function of the MG369xC.

[:SOURce]:PULM:INTernal:FREQuency <NRf>

[:SOURce]:PULM:INTernal:FREQuency?

Description: Sets the pulse repetition frequency (PRF) of the internal pulse generator to the value entered. The PRF range is determined by the pulse generator clock rate (see note below)

Parameters: Frequency (in Hz)

Range: 5.96 Hz to 25 MHz (at 100 MHz pulse generator clock rate);
0.597 Hz to 1.66 MHz (at 10 MHz pulse generator clock rate)

Default: 1 kHz

Examples: [:SOURce]:PULM:INTernal:FREQuency 50 kHz

Sets the internal PRF to 50 kHz.

[:SOURce]:PULM:INTernal:FREQuency?

Requests the value currently programmed for the internal PRF.

Note

[:SOURce]:PULM:SOURce INTernal1 | INTernal2 is used to select the internal pulse generator and its clock rate. Setting INTernal1 selects the internal pulse generator operating at a 100 MHz clock rate; setting INTernal2 selects the internal pulse generator operating at a 10 MHz clock rate.

[:SOURce]:PULM:POLarity <char>

[:SOURce]:PULM:POLarity?

Parameters: NORMal | INVerted

Default: NORMal

Description: Selects the polarity of the signal that turns the RF on during pulse modulation, as follows:

NORMal specifies positive-true operation; a TTL-high level will turn on the RF output signal

INVerted specifies negative-true operation; a TTL-low level will turn on the RF output signal

Examples: [:SOURce]:PULM:POLarity INVerted

Selects a negative true (TTL-low level) signal to turn the RF on during pulse modulation.

[:SOURce]:PULM:POLarity?

Requests the currently programmed signal polarity that will turn the RF on during pulse modulation.

[:SOURCE]:PULM:SOURce <char>

[:SOURCE]:PULM:SOURce?

Description: Selects the pulse modulation signal source, as follows:

INTernal1 selects the signal from the internal pulse generator operating at a 100 MHz clock rate

INTernal2 selects the signal from the internal pulse generator operating at a 10 MHz clock rate

EXTernal1 selects the front panel external pulse input

EXTernal2 selects the rear panel external pulse input

Parameters: INTernal1 | INTernal2 | EXTernal1 | EXTernal2

Default: INTernal1

Examples: [:SOURCE]:PULM:SOURce EXTernal2

Selects the rear panel pulse input as the pulse modulation signal source.

[:SOURCE]:PULM:SOURce?

Requests the currently programmed pulse modulation signal source.

[:SOURCE]:PULM:STATE <boolean>

[:SOURCE]:PULM:STATE?

Description: Turns the MG369xC pulse modulation function on/off.

Parameters: ON | OFF | 1 | 0

Default: ON

Examples: [:SOURCE]:PULM:STATE ON

Turns on the MG369xC pulse modulation function.

[:SOURCE]:PULM:STATE?

Requests the currently programmed state of the pulse modulation function.

[:SOURce]:PULSe Commands

The [:SOURce]:PULSe command and its subcommands make up the Pulse Subsystem within the :SOURce subsystem. These commands control the internal pulse generation function of the MG369xC.

[:SOURce]:PULSe:COUNT <NR1>

[:SOURce]:PULSe:COUNT?

Description: Sets the number of pulses generated by the internal pulse generator for each period of the pulsed waveform.

Parameters: Number of pulses

Range: 1 to 4

Default: 1

Examples: [:SOURce]:PULSe:COUNT 2

Set the number of pulses per pulse period to 2.

[:SOURce]:PULSe:COUNT?

Requests the currently programmed number of pulses per pulse period.

[:SOURce]:PULSe:DELay<n> <NRf>

[:SOURce]:PULSe:DELay<n>?

(1 ≤ n ≤ 4)

Description: Sets the pulse delay for the selected pulse to the value entered. The pulse delay range for each pulse is determined by the pulse generator clock rate (see notes below).

Parameters: Pulse delay (in seconds)

Range: See notes below

Default: 100 μs

Examples: [:SOURce]:PULSe:DELay2 200 US

Set the pulse 2 delay to 200 μs.

[:SOURce]:PULSe:DELay3?

Requests the currently programmed pulse 3 delay.

Note

[:SOURce]:PULM:SOURce INTERNAL1|INTERNAL2 is used to select the internal pulse generator and its clock rate. Setting INTERNAL1 selects the internal pulse generator operating at a 100 MHz clock rate; setting INTERNAL2 selects the internal pulse generator operating at a 10 MHz clock rate.

The pulse delay range for each pulse is shown in [Table 3-9](#).

Table 3-9. Pulse Delay Range

Pulse Delay	Clock Rate	
	100 MHz	10 MHz
Pulse 1	0 to 160 ms	0 to 1.6 s
Pulse 2	100 ns to 160 ms	300 ns to 1.6 s
Pulse 3	100 ns to 160 ms	300 ns to 1.6 s
Pulse 4	100 ns to 160 ms	300 ns to 1.6 s

[:SOURce]:PULSe:PERiod <NRf>

[:SOURce]:PULSe:PERiod?

Description: Sets the pulse period of the internal pulse generator to the value entered. The pulse period range is determined by the pulse generator clock rate (see notes below).

Parameters: Pulse period (in seconds)

Range: 100 ns to 160 ms (at 100 MHz pulse generator clock rate);
600 ns to 1.6 s (at 10 MHz pulse generator clock rate)

Default: 1 ms

Examples: [:SOURce]:PULSe:PERiod 50 MS

Set the pulse period to 50 ms.

[:SOURce]:PULSe:PERiod?

Requests the currently programmed pulse period.

Note

[:SOURce]:PULM:SOURce INTernal1|INTernal2 is used to select the internal pulse generator and its clock rate. Setting INTernal1 selects the internal pulse generator operating at a 100 MHz clock rate; setting INTernal2 selects the internal pulse generator operating at a 10 MHz clock rate.

The pulse period must be longer than the sum of the pulse delay and pulse width by a minimum of five clock cycles.

[:SOURce]:PULSe:WIDTh<n> <NRf>

[:SOURce]:PULSe:WIDTh<n>?

($1 \leq n \leq 4$)

Description: Sets the pulse width for the selected pulse to the value entered. The pulse width range is determined by the pulse generator clock rate (see note below).

Parameters: Pulse width (in seconds)

Range: 10 ns to 160 ms (at 100 MHz pulse generator clock rate);
100 ns to 1.6 s (at 10 MHz pulse generator clock rate.)

Default: 500 μ s

Examples: [:SOURce]:PULSe:WIDTh1 250 US

Set the width of pulse 1 to 250 μ s.

[:SOURce]:PULSe:WIDTh2?

Requests the currently programmed width of pulse 2.

Note

[:SOURce]:PULM:SOURce INTernal1|INTernal2 is used to select the internal pulse generator and its clock rate. Setting INTernal1 selects the internal pulse generator operating at a 100 MHz clock rate; setting INTernal2 selects the internal pulse generator operating at a 10 MHz clock rate.

[:SOURce]:PULSe:STEP <boolean>

[:SOURce]:PULSe:STEP?

Description: Turns the internal pulse stepped delay mode on/off. When on, stepped delay mode automatically increments or decrements the pulse delay 1 value according to step delay parameters (see notes below).

Parameters: ON | OFF | 1 | 0

Default: OFF

Examples: [:SOURce]:PULSe:STEP ON

Turn the internal pulse stepped delay mode on.

[:SOURce]:PULSe:STEP?

Requests currently programmed internal pulse stepped delay mode state (on/off).

Note

This mode is only available when one of the following triggering modes is specified:

Delayed (:TRIGger:SEQuence3:TYPE DELayed)

or

Triggered w/delay (:TRIGger:SEQuence3:TYPE TRGDelay).

If the step delay parameters that are set result in a fractional number of increments, then the last (fractional) one is not taken.

[:SOURce]:PULSe:STEP:START <NRf>

[:SOURce]:PULSe:STEP:START?

Description: Sets the pulse delay 1 starting time used with the :PULSe:STEP command to the value entered. The pulse delay range is determined by the pulse generator clock rate (see note).

Parameters: Pulse delay 1 starting time (in seconds)

Range: 100 ns to 160 ms (at 100 MHz pulse generator clock rate)
300 ns to 1.6 s (at 10 MHz pulse generator clock rate)

Default: 100 μ s

Examples: [:SOURce]:PULSe:STEP:START 80 US

Set the stepped delay mode pulse delay 1 starting time to 80 μ s.

[:SOURce]:PULSe:STEP:START?

Requests the currently programmed stepped delay mode pulse delay 1 starting time.

Note

[:SOURce]:PULM:SOURce INTernal1|INTernal2 is used to select the internal pulse generator and its clock rate. Setting INTernal1 selects the internal pulse generator operating at a 100 MHz clock rate; setting INTernal2 selects the internal pulse generator operating at a 10 MHz clock rate.

[:SOURce]:PULSe:STEP:STOP <NRf>

[:SOURce]:PULSe:STEP:STOP?

Description: Sets the pulse delay 1 ending time used with the :PULSe:STEP command to the value entered. The pulse delay range is determined by the pulse generator clock rate (see note).

Parameters: Pulse delay 1 ending time (in seconds)

Range: 100 ns to 160 ms (at 100 MHz pulse generator clock rate)
300 ns to 1.6 s (at 10 MHz pulse generator clock rate)

Default: 100 μ s

Examples: [:SOURce]:PULSe:STEP:STOP 120 US

Set the stepped delay mode pulse delay 1 ending time to 120 μ s.

[:SOURce]:PULSe:STEP:STOP?

Requests the currently programmed stepped delay mode pulse delay 1 ending time.

Note

[:SOURce]:PULM:SOURce INTernal1|INTernal2 is used to select the internal pulse generator and its clock rate. Setting INTernal1 selects the internal pulse generator operating at a 100 MHz clock rate; setting INTernal2 selects the internal pulse generator operating at a 10 MHz clock rate.

[:SOURce]:PULSe:STEP:INCRement <NRf>

[:SOURce]:PULSe:STEP:INCRement?

Description: Sets the step increment size used with the :PULSe:STEP command to the value entered. The pulse delay range is determined by the pulse generator clock rate (see note).

Parameters: Step size time (in seconds)

Range: 10 ns to 10 ms (at 100 MHz pulse generator clock rate)
100 ns to 1s (at 10 MHz pulse generator clock rate)

Default: 100 μ s

Examples: [:SOURce]:PULSe:STEP:INCRement 10 US

Set the step increment size for the stepped delay mode to 10 μ s.

[:SOURce]:PULSe:STEP:INCRement?

Requests the currently programmed stepped delay mode step increment size.

Note

[:SOURce]:PULM:SOURce INTernal1|INTernal2 is used to select the internal pulse generator and its clock rate. Setting INTernal1 selects the internal pulse generator operating at a 100 MHz clock rate; setting INTernal2 selects the internal pulse generator operating at a 10 MHz clock rate.

[:SOURce]:PULSe:STEP:TIME <nv>

[:SOURce]:PULSe:STEP:TIME?

Description: Sets the dwell time for each step used with the :PULSe:STEP command to the value entered.

Parameters: Dwell time (in seconds) | MIN | MAX

Range: 100 μ s to 10 sec

Default: 1 ms

Examples: [:SOURce]:PULSe:STEP:TIME 100 MS

Set the dwell-time-per-step for the stepped delay mode to 100 ms.

[:SOURce]:PULSe:STEP:TIME?

Requests the currently programmed stepped delay mode dwell-time-per-step.

[:SOURce]:ROSCillator Commands

The [:SOURce]:ROSCillator command and its subcommands comprise the Reference Oscillator subsystem within the :SOURce subsystem. These commands control the reference oscillator function of the MG369xC.

[:SOURce]:ROSCillator:BANDwidth:EXTernal <integer>

[:SOURce]:ROSCillator:BANDwidth:EXTernal?

Description: Sets the loop bandwidth for an external reference oscillator.

Parameters: Bandwidth (in Hz) 10 | 30 | 100 | 300

Range: 10 Hz to 300 Hz

Default: 100 Hz

Examples: [:SOURce]:ROSCillator:BANDwidth:EXTernal 10

Set the external reference oscillator loop bandwidth to 10 Hz.

[:SOURce]:ROSCillator:BANDwidth:EXTernal?

Request the currently programmed external reference loop bandwidth.

[:SOURce]:ROSCillator:SOURce? <char>

Description: Returns the current setting for the reference oscillator source of the instrument.

Parameters

Returned: INT | EXT

[:SOURce]:SCAN Commands

The [:SOURce]:SCAN command and its subcommands comprise the Scan Modulation Subsystem within the :SOURce subsystem. These commands control the scan modulation function of the MG369xC.

[:SOURce]:SCAN:STATE <boolean>

[:SOURce]:SCAN:STATE?

Description: Enables/disables scan modulation of the MG369xC RF output signal (see notes below).

Parameters: ON | OFF | 1 | 0

Default: OFF

Examples: [:SOURce]:SCAN:STATE ON

Sets scan modulation on.

[:SOURce]:SCAN:STATE?

Requests currently programmed scan modulation state (on/off).

Note

Scan modulation uses an internal SCAN modulator to amplitude modulate RF output signals from 1 to 20 GHz at modulation depths up to 60 dB. It requires a modulating signal from an external source.

If the SCAN modulator is not installed in the instrument, this command produces a syntax error.

[:SOURce]:SWEep Commands

The [:SOURce]:SWEep command and its subcommands comprise the Sweep Subsystem within the :SOURce subsystem. These commands control the standard stepped and analog frequency sweep functions and the step power level sweep function of the MG369xC.

[:SOURce]:SWEep<n>:DIRection <char>

[:SOURce]:SWEep:DIRection?

($1 \leq n \leq 2$; see note)

Description: Selects the direction of sweep, in the power level sweep mode only. (Sweep direction is always in the UP direction for frequency sweeps.)

Parameters: UP | DOWN

Default: UP

Examples: [:SOURce]:SWEep:DIRection DOWN

Set the power level sweep from a high level to a low level.

[:SOURce]:SWEep:DIRection?

Requests the currently programmed sweep direction.

Note

:SWEep1 (or :SWEep) signifies frequency sweep; :SWEep2 signifies power sweep.

[:SOURce] SWEep<n>:DWELl <nv>

[:SOURce] :SWEep:DWELl?

(1 ≤ n ≤ 2)

Description: Sets the dwell time for each step in a stepped frequency sweep or power level sweep to the value entered (see notes below).

Parameters: Dwell time (in seconds) | MIN | MAX

Range: 1 ms to 99 sec

Default: 1 ms

Examples: [:SOURce] :SWEep:DWELl 100 ms

Set dwell time for each step in the sweep to 100 milliseconds.

[:SOURce] :SWEep:DWELl?

Requests the currently programmed value for sweep step dwell time.

Note The value entered for dwell time cannot be less than :TIME/:POINTs. Refer to [Table 3-10](#).
When encountered, the command :SWEep:DWELl <param> command sets :SWEep:DWELl:AUTO to off.

Table 3-10. Interaction between Dwell, Sweep Time, and Points

:SWEep:XXXX:AUTO switches		Interaction
:DWELl	:TIME	
OFF	OFF	No coupling between :SWEep:DWELl, :SWEep:TIME, and :SWEep:POINTs.
OFF	ON	:SWEep:TIME is always set to the minimum value that is compatible with other settings, as follows: If :SWEep:GENERation ANALog, then :SWEep:TIME = the larger of :SWEep:TIME:LLIMit or 30 ms If :SWEep:GENERation STEPped, then :SWEep:TIME = (:DWELl × :POINTs)
ON	OFF	When :SWEep:TIME or :SWEep:POINTs are changed, :SWEep:DWELl is set to the larger of (:DWELl = :TIME/:POINTs) or :DWELl Min.
ON	ON	:SWEep:TIME is always set to minimum. In stepped sweep mode, :SWEep:DWELl is adjusted to its minimum value, then :SWEep:TIME is computed. If :SWEep:GENERator ANALog, then :SWEep:TIME = the larger of :SWEep:TIME:LLIMit or 30 ms SWEep:DWELl is unaffected by changes in :TIME or:POINTs If :SWEep:GENERator STEPped, then :SWEep:DWELl = :DWELl Min (the larger of .001 or :TIME:LLIMit/:POINTs) :SWEep:TIME = :DWELl Min × :POINTs

[:SOURCE] :SWEep<n>:DWELl:AUTO <boolean>

[:SOURCE] :SWEep:DWELl:AUTO?

(1 ≤ n ≤ 2)

Description: ON signifies that :DWELl ≈ :TIME/:POINTs. See note below.

Parameters: ON | OFF | 1 | 0

Default: ON

Examples: [:SOURCE] :SWEep:DWELl:AUTO ON

Set :SWEep:DWELl to its default value.

[:SOURCE] :SWEep:DWELl:AUTO?

Requests the currently programmed :SWEep:DWELl setting (on/off).

:DWELl = (:TIME / :POINTs – internal settling time)

Note

:DWELl, :SPAN, :TIME, :STEP, and :POINTs are all interrelated. Entering too large a value for :DWELl may invalidate the setting for :TIME.

[:SOURce]:SWEep<n>:GENeration <char>

[:SOURce]:SWEep:GENeration?

(1 ≤ n ≤ 2)

Description: Selects between analog and stepped frequency sweeps, in the SWEep [1] (frequency sweep) mode only. Refer to [Table 3-11](#).

Parameters: ANALog | STEPped (see note)

Default: SWEep1 is ANALog; SWEep2 is STEPped

Examples: [:SOURce]:SWEep:GENeration ANALog

Set SWEep[1] for analog frequency sweep.

[:SOURce]:SWEep:GENeration?

Requests the currently programmed frequency sweep mode.

Note SWEep2 (power level sweeps) are *always* stepped sweeps.

Table 3-11. Sweep Mode Compatibility and Action Chart

Mode		Sweep		Frequency Action	Power Action
:FREQ	:POW	:SWE:GEN	:SWE2:GEN		
CW	FIX	x	x	CW, F1	Fixed, L0
SWCW	FIX	x	x	CW, F1, Ramp on	Fixed, L0
SWE	FIX	ANAL	x	Analog Sweep, F1-F2	Fixed, L0
SWE	FIX	STEP	x	Stepped Sweep, F1-F2	Fixed, L0
SWE	SWE	STEP	x	Stepped Sweep F1-F2 :SWE:POINTs :SWE:STEP	Step Power after each Freq Sweep L1-L2 :SWE:POINTs :SWE:POW:STEP
SWE	SWE2	ANAL	STEP	Analog Sweep F1-F2	Step Power after each Freq Sweep L1-L2 :SWE2:POINTs :SWE2:POW:STEP
SWE	SWE2	STEP	STEP	Stepped Sweep F1-F2 :SWE:POINTs :SWE:STEP	Step Power after each Freq Sweep L1-L2 :SWE2:POINTs :SWE2:POW:STEP
CW	SWE	STEP	x	CW, F0	Step Power Sweep L1-L2
SWCW	SWE	STEP	x	CW, F0, Ramp on	Step Power Sweep L1-L2
CW	SWE2	x	STEP	CW, F0	Step Power Sweep L1-L2
SWCW	SWE2	x	STEP	CW, F0, Ramp on	Step Power Sweep L1-L2
ALSW	FIX	x	x	Alternate X Sweep F1-F2 F3-F4	Fixed L0 Trigger n L0 Trigger n+1
ALSW	ALSW	ANAL	x	Alternate Analog Sweep F1-F2 F3-F4	Alternate values L0 Trigger n L1 Trigger n+1
ALSW	ALSW	STEP	x	Alternate Stepped Sweep F1-F2 F3-F4	Alternate values L0 Trigger n L1 Trigger n+1

[:SOURce] :SWEep<n>:POINTs <nv>

[:SOURce] :SWEep:POINTs?

(1 ≤ n ≤ 2)

Description: Sets the number of points in each sweep in the stepped frequency sweep or power level sweep to the valued entered. See notes below.

Parameters: Number of points | MIN | MAX

Range: 2 to 10,001 (MAXimum) for Stepped Frequency sweeps

Default: 10,001 for SWEep1

The default value for SWEep2 depends on the power range of the particular MG369xC model (see notes below).

Examples: [:SOURce] :SWEep:POINTs 500

Set the number of points for each sweep to 500.

[:SOURce] :SWEep:POINTs?

Requests the currently programmed value for points/sweep.

The default values for SWEep2 points and time are set up to give the maximum number of points at the minimum step size for the particular MG369xC model. The default values are dependent on the instrument's available power range.

Note

:POINTs and :STEP are coupled values. Entering the value for one will cause the other to be recalculated, per formula below. Entering a new value for either parameter will not change :SPAN.

$:\text{POINTs} = (:\text{SPAN} / :\text{STEP}) + 1$

An error will be generated if the :POINTs value entered results in a step size (1) less than 1 kHz (or 0.1 Hz for models with Option 11) for a stepped frequency sweep or (2) less than 0.01 dB for a power level sweep.

[:SOURce] :SWEep<n> [:FREQuency] :STEP <nv>

[:SOURce] :SWEep:FREQuency:STEP?

(n = 1; see note below)

Description: Sets the step size for each step in SWEep [1] (frequency) linear stepped sweep to the value entered. See notes below.

Parameters: Frequency (in Hz) | MIN | MAX

Range: (see notes below)

Default: (MAX – MIN) / 10,000

Examples: [:SOURce] :SWEep:FREQuency:STEP 5 GHz

Set the step size for each step in the linear stepped frequency sweep to 5 GHz.

[:SOURce] :SWEep:FREQuency:STEP?

Requests the currently programmed step size for each step in the linear stepped frequency sweep.

:SWEep2 is not valid for this command; only :SWEep [1] will be recognized.

This parameter is not used if :SWEep:SPACing is LOGarithmic.

:POINTs and :STEP are coupled values. Entering the value for one will cause the other to be recalculated; see notes under :SWEep:POINTs command.

Note

The maximum frequency sweep step size is equal to the maximum frequency span for the particular MG369xC model / (minimum points – 1). For the models with Option 4 or Option 5, the minimum step size is 0.1 Hz and the maximum step size is 19,990,000,000 Hz.

Refer to [Table 3-6](#) for the frequency span values for each MG369xC model.

An error will be generated if the :STEP size value entered exceeds the value for :SPAN.

[:SOURce]:SWEep<n>:POWer:STEP <nv>

[:SOURce]:SWEep<n>:POWer:STEP?

(1 ≤ n ≤ 2)

Description: Sets the step size for each step in a power level sweep to the value entered. See notes below.

Parameters: Power level (in dB) | MIN | MAX

Range: MINimum = 0.01 dB
MAXimum is model dependent (see notes below)

Default: 0.01 dB for SWEep1
0.02 dB for SWEep2

Examples: [:SOURce]:SWEep2:POWer:STEP 10 dB

Set the step size for each step in the SWEep2 power level sweep to 10 dB.

[:SOURce]:SWEep2:POWer:STEP?

Requests the currently programmed step size for each step in the SWEep2 power level sweep.

:STEP and :POINTs are coupled values. Entering the value for one will cause the other to be recalculated; see notes under :SWEep:POINTs command.

Note

For standard MG369xC models, a maximum step size up to 33 dB may be used (up to 149 dB for models with Option 2 step attenuator). For MG369xC models with Option 15, a maximum step size up to 27 dB can be used (up to 141 dB with Option 2 step attenuator). However, the step size, in conjunction with the initial RF power output setting, must not produce a programmed power level below the minimum leveled output power for the particular MG369xC model. Refer to Appendix B—Performance Specifications, in the MG369xC Synthesized Signal Generator Operation Manual.

[:SOURce]:SWEep<n>:SPACing <char>

[:SOURce]:SWEep:SPACing?

(n = 1; see note below)

Description: Selects the type of SWEep [1] stepped frequency sweep (see notes).

Parameters: LINear | LOGarithmic

Default: LINear

Examples: [:SOURce]:SWEep:SPACing LOGarithmic

Selects logarithmic type stepped frequency sweep.

[:SOURce]:SWEep:SPACing?

Requests the currently programmed type of stepped frequency sweep.

Note

In LINear stepped sweep mode, the frequency sweep is linearly incremented by the step size from the start frequency to the stop frequency. The size of each step of the linear stepped sweep can be set using either the :SWEep[:FREQuency]:STEP command or the :SWEep:POINTs command.

In LOGarithmic stepped sweep mode, step size increases logarithmically with frequency and is determined by a logarithmic curve fitted between the sweep start and stop frequencies. The number of steps is set using the command :SWEep:POINTs.

[:SOURce]:SWEep<n>:TIME <nv>

[:SOURce]:SWEep<n>:TIME?

(1 ≤ n ≤ 2)

Description: Sets the sweep time for the associated SWEep [1] or SWEep2 sweep to the value entered. See notes below.

Parameters: Sweep time (in seconds) | MIN | MAX

Range: 30 ms to 99 sec

Default: 30 ms for SWEep1

The default value for SWEep2 depends on the power range of the particular MG369xC model (see notes).

Examples: [:SOURce]:SWEep1:TIME 100 ms

Set the sweep time for SWEep1 sweep to 100 ms.

[:SOURce]:SWEep2:TIME?

Requests the currently programmed time for SWEep2 sweep.

Note

The default values for SWEep2 points and time are set up to give the maximum number of points at the minimum step size for the particular MG369xC model. The default values are dependent on the instrument's available power range.

When :SWEep<n>:TIME <param> is implemented, :SWEep<n>:TIME:AUTO is set to OFF. :TIME, :DWELL, :SPAN, :STEP, and :POINTS are all interrelated. Entering too large a value for any of the four other parameters may invalidate the entered value for :TIME.

[:SOURce]:SWEep<n>:TIME:LLIMit <nv>

[:SOURce]:SWEep<n>:TIME:LLIMit?

(1 ≤ n ≤ 2)

Description: Sets the lower limit for :SWEep<n>:TIME to the value entered. No sweep time will be shorter than this value. See notes below.

Parameters: Sweep time (in seconds) | MIN | MAX

Range: 2 ms to 98.998 sec

Default: 2 ms

Examples: [:SOURce]:SWEep1:TIME:LLIMit 80 ms

Set the lower limit for SWEep1 sweep time to 80 ms.

[:SOURce]:SWEep2:TIME:LLIMit?

Requests the currently programmed lower limit for SWEep2 sweep time.

Note

When :SWEep<n>:TIME:AUTO is set ON, the internally computed sweep time will not be smaller than the sweep time set with the :SWEep<n>:Time:LLIMit command.

:SWEep2:TIME:LLIMit will be set to the same value as :SWEep:TIME:LLIMit and vice versa.

`[:SOURce]:SWEep<n>:TIME:AUTO <boolean>`

`[:SOURce]:SWEep:TIME:AUTO?`

($1 \leq n \leq 2$)

Description: ON specifies that the sweep time for the associated sweep (SWEep[1] or SWEep2) is to be calculated internally and is dependent on the sweep SPAN value. See note below.

Parameters: ON | OFF | 1 | 0

Default: ON

Examples: `[:SOURce]:SWEep:TIME:AUTO ON`

Specifies that the SWEep[1] sweep time is to be calculated internally.

`[:SOURce]:SWEep:TIME:AUTO?`

Requests the currently programmed mode for sweep time determination.

Note

When the MG369xC is powered up, or when *RST command is issued, :SWEep:TIME:AUTO is set to ON.

3-11 :STATus Subsystem

The :STATus subsystem controls the SCPI-defined status-reporting structures of the MG369xC. The subsystem commands and parameters are described below.

Table 3-12. :STATus Subsystems

Keyword	Parameter Form	Parameter Data or Units	Notes
:STATus			
:OPERation			
[:EVENT] ?			Query Only
:CONDition?			Query Only
:ENABle	<NRf>	numeric value	
:PTRansition	<NRf>	numeric value	
:NTRansition	<NRf>	numeric value	
:PRESet			No Query
:QUEStionable			
[:EVENT] ?			Query Only
:CONDition?			Query Only
:ENABle	<NRf>	numeric value	
:PTRansition	<NRf>	numeric value	
:NTRransition	<NRf>	numeric value	
:QUEue			
[:NEXT] ?			Query Only

:STATus:OPERation [:EVENT] ?

Description: Returns the contents of the MG369xC Operational Event register in <NR1> format. *When executed, this command clears the Operational Event register.*

Parameters

Returned: Event register contents

Example: :STATus:OPERation:EVENT?

Requests that the contents of the Operational Event register be returned.

:STATus:OPERation:CONDition?

Parameters

Returned: Condition register contents

Description: Returns the contents of the MG369xC Operational Condition register in <NR1> format. When executed, this command does *not* clear the Operational Condition register.

Example: :STATus:OPERation:CONDition?

Requests that the contents of the Operational Condition register be returned.

:STATus:OPERation:ENABle <NRf>

:STATus:OPERation:ENABle?

Description: Sets the bits of the Operational Enable register associated with the Operational Event register to the binary weighted integer value specified by the mask parameter.

Parameters: Mask

Range: 0 to 32767

Default: 0 (All 0's)

Examples: :STATus:OPERation:ENABle 8

Set the MG369xC Operational Enable register to a value of 8. (This will unmask the Sweeping status bit).

:STATus:OPERation:ENABle?

Requests the current value of the MG369xC Operational Enable register.

:STATus:OPERation:PTRansition <NRf>

:STATus:OPERation:PTRansition?

Description: Sets the bits of the positive transition filter for the Operational Condition register to the binary weighted integer value specified by the mask parameter.

Parameters: Mask

Range: 0 to 32767

Default: 32767 (All 1's)

Examples: :STATus:OPERation:PTRansition 512

Sets the MG369xC Positive Transition Filter for the Operational Condition register to a value of 512. When the filter detects a False to True transition in bit 9 of the Operational Condition register, indicating that MG369xC self-test is in progress, bit 9 of the Operational Event register will be set to "1".

:STATus:OPERation:PTRansition?

Requests the current value of the MG369xC Positive Transition Filter for the Operational Condition register.

:STATus:OPERation:NTRansition <NRf>

:STATus:OPERation:NTRansition?

Description: Sets the bits of the negative transition filter for the Operational Condition register to the binary weighted integer value specified by the mask parameter.

Parameters: Mask

Range: 0 to 32767

Default: 0 (All 0's)

Examples: :STATus:OPERation:NTRansition 8

Sets the MG369xC Negative Transition Filter for the Operational Condition register to a value of 8. When the filter detects a True to False transition in bit 3 of the Operational Condition register, indicating that the MG369xC is finished sweeping, bit 3 of the Operational Event register will be set to "1".

:STATus:OPERation:NTRansition?

Requests the current value of the MG369xC Negative Transition Filter for the Operational Condition register.

:STATus:PRESet

Description: This command is an event that configures the SCPI and device-dependent status reporting structures so that device-dependent events are summarized and reported. This command performs the following functions:

Sets the Operational Enable register to all 0's.
 Sets the Operational Positive Transition Filter to all 1's.
 Sets the Operational Negative Transition Filter to all 0's.
 Sets the Questionable Enable register to all 0's.
 Sets the Questionable Positive Transition Filter to all 1's.
 Sets the Questionable Negative Transition Filter to all 0's.

Parameters: None

Example: :STATus:PRESet

Configure the status reporting structures for device-dependent event reporting.

:STATus:QUESTionable[:EVENT]?

Description: Returns the contents of the MG369xC Questionable Event register in <NR1> format. *When executed, this command clears the Questionable Event register.*

Parameters

Returned: Event register contents

Example: :STATus:QUESTionable:EVENT?

Requests that the contents of the Questionable Event register be returned.

:STATus:QUESTionable:CONDition?

Description: Returns the contents of the MG369xC Questionable Condition register in <NR1> format. *When executed, this command does not clear the Questionable Condition register.*

Parameters

Returned: Condition register contents

Example: :STATus:QUESTionable:CONDition?

Requests that the contents of the Questionable Condition register be returned.

:STATus:QUESTionable:ENABle <NRf>**:STATus:QUESTionable:ENABle?**

Description: Sets the bits of the Questionable Enable register associated with the Questionable Event register to the binary weighted integer value specified by the mask parameter.

Parameters: Mask

Range: 0 to 32767

Default: 0 (All 0's)

Examples: :STATus:QUESTionable:ENABle 32

Set the MG369xC Questionable Enable register to a value of 32. (This will unmask the Lock Error or RF Unlocked status bit).

:STATus:QUESTionable:ENABle?

Requests the current value of the MG369xC Questionable Enable register.

:STATus:QUEStionable:PTRansition <NRf>**:STATus:QUEStionable:PTRansition?**

Description: Sets the bits of the positive transition filter for the Questionable Condition register to the binary weighted integer value specified by the mask parameter.

Parameters: Mask

Range: 0 to 32767

Default: 32767 (All 1's)

Examples: :STATus:QUEStionable:PTRansition 512

Sets the MG369xC Positive Transition Filter for the Questionable Condition register to a value of 512. When the filter detects a False to True transition in bit 9 of the Questionable Condition register, indicating that MG369xC self-test failed, bit 9 of the Questionable Event register will be set to "1".

:STATus:QUEStionable:PTRansition?

Requests the current value of the MG369xC Positive Transition Filter for the Questionable Condition register.

:STATus:QUEStionable:NTRansition <NRf>**:STATus:QUEStionable:NTRansition?**

Description: Sets the bits of the negative transition filter for the Questionable Condition register to the binary weighted integer value specified by the mask parameter.

Parameters: Mask

Range: 0 to 32767

Default: 0 (All 0's)

Examples: :STATus:QUEStionable:NTRansition 8

Sets the MG369xC Negative Transition Filter for the Questionable Condition register to a value of 8. When the filter detects a True to False transition in bit 3 of the Questionable Condition register, indicating that the RF is no longer unlevelled, bit 3 of the Questionable Event register will be set to "1".

:STATus:QUEStionable:NTRansition?

Requests the current value of the MG369xC Negative Transition Filter for the Questionable Condition register.

:STATus:QUEue[:NEXT]?

Parameters

Returned: Error code, error message string

Description: Returns in <NR1><string> format and deletes the oldest uncleared error code and error description from the error queue. Optional device dependent information about the error event may also be included. See notes below. The error codes and error description information for the MG369xC are listed in [Chapter 4](#).

Example: :STATus:QUEue:NEXT?

Requests that the oldest error code/error description be returned from the error queue.

Note

As errors are detected, they are placed in the error queue. The queue is first in, first out and can hold a maximum of 10 messages.

If the error queue is not empty, bit 2 of the Summary Status Byte is set.

A query returns only the oldest error code and associated error description information from the error queue. To return all error codes and associated description information, use repetitive queries until an error value of zero is returned, or until bit 2 of the status byte is 0.

If the error queue overflows, the last error message in the queue is replaced by the error:

–350, “Queue overflow”

3-12 :SYSTEM Subsystem

The :SYSTEM subsystem commands are used to implement functions that are not related to MG369xC performance. These include error query, interface language selection, system preset, and version query. The subsystem commands and parameters are described below.

Table 3-13. :STATus Subsystems

Keyword	Parameter Form	Parameter Data or Units	Notes
:SYSTEM			
:ERROR?			Query Only
:LANGUage	<string>	“SCPI” “NATIVE” “TMSL”	
:PRESet			No Query
:VERSion?			Query Only

:SYSTEM:ERROR?

Parameters

Returned: Error code, error message string

Description: Returns in <NR1><string> format and deletes the oldest uncleared error code and error description from the error queue. Optional device dependent information about the error event may also be included. See notes below. The error codes and error description information for the MG369xC are listed in [Chapter 4](#).

Example: :SYSTEM:ERROR?

Requests that the oldest error code/error description be returned from the error queue.

Note

As errors are detected, they are placed in the error queue. The queue is first in, first out and can hold a maximum of 10 messages.

If the error queue is not empty, bit 2 of the Summary Status Byte is set.

A query returns only the oldest error code and associated error description information from the error queue. To return all error codes and associated description information, use repetitive queries until an error value of zero is returned, or until bit 2 of the status byte is 0.

If the error queue overflows, the last error message in the queue is replaced by the error:

–350, “Queue overflow”

:SYSTem:LANGUage <string>**:SYSTem:LANGUage?**

Description: Selects the instrument's external interface language. "TMSL" is an alias for "SCPI". Entering either will return "SCPI" when queried. The double quotes are required and will be returned with the query reply.

"NATIVE" changes the instrument's external interface language to the MG369xC GPIB mode. Any commands issued within 1 second of the change may be garbled or lost.

Parameters: "SCPI" | "NATIVE" | "TMSL"

Default: Dependent upon the selection made at the MG369xC front panel Configure GPIB menu.

Examples: :SYSTem:LANGUage "NATIVE"

Selects "NATIVE" (MG369xC GPIB mode) as the external interface language.

:SYSTem:LANGUage?

Requests the current selection for MG369xC external interface language.

Note

When changing from NATIVE to SCPI interface language, use the command `SYST:LANG "SCPI"`. Do **not** use the long form of the command and do **not** use a leading colon (:) with the command. The command `:SYSTem:LANGUage "SCPI"` results in a syntax error.

:SYSTem:PRESet

Description: This command is synonymous with *RST and is included for programming compatibility with other instruments.

Parameters: none

Example: :SYSTem:PRESet

Sets all user programmable MG369xC parameters to their default values.

:SYSTem:VERSion?

Parameters

Returned: Version number (see note)

Description: Returns the SCPI version number in <NR2> format that the instrument software complies with.

Example: :SYSTem:VERSion?

Requests the SCPI version number that the instrument software complies with.

Note

The query response shall have the form YYYY.V where the Ys represent the year-version (i.e., 1993) and the V represents the approved revision number for that year.

3-13 :TRIGger Subsystem

The :TRIGger subsystem commands are used to control the sweep triggering functions of the MG369xC. The subsystem commands and parameters are described below. The :TRIGger command, along with the :ABORT and :INITiate commands, comprise the Trigger Group of commands.

Table 3-14. :Trigger Subsystem

Keyword	Parameter Form	Parameter Data or Units	Notes
:TRIGger			
[:IMMediate]			No Query
:SOURce	<char>	BUS IMMediate HOLD	
:SEQuence3			
:SLOPE	<char>	POSitive NEGative	
:TYPE	<char>	FREerun TRIGgered TRGDelay GATed DELayed COMPOSITE	
:SOURce	<char>	EXTernal1 EXTernal2	

:TRIGger [:IMMediate]

Description: Causes the previously selected sweep to be triggered immediately if trigger system is in armed state due to previous :INITiate:IMMediate command. See note below.

Parameters: none

Example: :TRIGger:IMMediate

Trigger selected sweep immediately if MG369xC is in trigger armed state.

Associated

commands: :ABORT and :INITiate

:TRIGger:SOURce <char>

:TRIGger:SOURce?

Description: Selects the trigger source for the previously selected sweep. The source selections are:

BUS – The source is the group execute trigger command from the GPIB. The trigger will occur when either a <GET> or *TRG command is received.

IMMediate – The trigger signal is always true.

HOLD – Do not trigger on any sweep.

Parameters: BUS | IMMediate | HOLD

Default: BUS

Examples: :TRIGger:SOURce IMMediate

Select the trigger signal to be true.

:TRIGger:SOURce?

Requests the currently programmed sweep trigger source.

Note

Only one trigger source can be specified at a time, and all others will be ignored.

Sending :TRIGger:SOURce IMM; INITiate:CONTInuous ON places the selected sweep in auto trigger mode.

:TRIGger:SEQuence3:SLOPe <char>

:TRIGger:SEQuence3:SLOPe?

(see note below)

Description: Selects whether the internal pulse generator is triggered on the rising edge (POSitive) or falling edge (NEGative) of the external trigger signal.

Parameters: POSitive | NEGative

Default: POSitive

Example: :TRIGger:SEQuence3:SLOPe NEGative

Select triggering of the internal pulse generator on the falling edge of the external trigger signal.

Note

:SEQuence3 is the trigger sequence for the internal pulse generation function of the MG369xC (see the [:SOURce]:PULSe subsystem commands).

The command :TRIGger:SEQuence3:SLOPe POSitive | NEGative only affects the triggering of the internal pulse generator when :TRIGger:SEQuence3:TYPE is set to GATed, TRIGgered, or TRGDelay.

:TRIGger:SEQuence3:TYPE <char>

:TRIGger:SEQuence3:TYPE?

Description: Selects the mode of triggering the internal pulse generator. The selections are:

FREerun – the pulse generator produces pulses at the internal PRF.

GATed – an external pulse gates the internal pulse generator on and off. When gated on, the pulse generator produces a single pulse at the internal PRF.

DELAyed – the pulse generator produces pulses delayed by pulse delay1 ([:SOURce]:PULSe:DELAy1) at the internal PRF.

TRIGgered – the pulse generator is triggered by an external trigger signal to produce pulses.

TRGDelay – the pulse generator is triggered by an external trigger signal to produce pulses delayed by pulse delay1 ([:SOURce]:PULSe:DELAy1).

COMPosite – an external pulse triggers the internal pulse generator and also pulse modulates the output signal. The internal pulse generator produces a single pulse delayed by pulse delay1 ([:SOURce]:PULSe:DELAy1) which also pulse modulates the output signal.

Parameters: FREerun | GATed | DELAyed | TRIGgered | TRGDelay | COMPosite

Default: FREerun

Example: :TRIGger:SEQuence3:TYPE TRIGgered

Selects triggering of the pulse generator by an external trigger signal.

Note

Only one mode of triggering can be specified at a time, and all others will be ignored.

GATed and COMPosite triggering modes are available in single pulse ([:SOURce]:PULSe:COUNT 1) mode only.

Pulse delay1 ([:SOURce]:PULSe:DELAy1) is not used with FREerun, GATed, and TRIGgered modes of triggering.

:TRIGger:SEQuence3:SOURce <char>

:TRIGger:SEQuence3:SOURce?

Description: Selects the source of the external trigger signal for the internal pulse generator as follows:

EXTernal1 – the front panel PULSE TRIGGER IN connector.

EXTernal2 – the rear panel PULSE TRIGGER IN connector.

Parameters: EXTernal1 | EXTernal2

Default: EXTernal1

Example: :TRIGger:SEQuence3:SOURce EXTernal2

Selects the external trigger signal from the rear panel PULSE TRIGGER IN connector.

Note

The command :TRIGger:SEQuence3:SOURce EXTernal1 | EXTernal2 is only active when :TRIGger:SEQuence3:TYPE is set to GATed, TRIGgered, TRGDelay, or COMPOSITE.

3-14 :TSweep Command

The :TSweep command is a convenience command. It is equivalent to sending :ABORT;:INITiate[:IMMEDIATE].

3-15 :UNIT Subsystem

The :UNIT subsystem commands set the default units for the frequency and time parameters that are used with all MG369xC SCPI commands described in this manual. The units selected apply to the designated command parameters for both command and response. The subsystem commands and parameters are described below.

Table 3-15. :UNIT Subsystem

Keyword	Parameter Form	Parameter Data or Units	Notes
:UNIT			
:FREQuency	<char>	HZ KHZ MHZ GHz	
:TIME	<char>	S MS US NS	

:UNIT:FREQuency <char>

:UNIT:FREQuency?

Description: Selects the global default frequency unit for all frequency related parameters used with all MG369xC SCPI commands.

Parameters: HZ | KHZ | MHZ | GHz

Default: HZ

Examples: :UNIT:FREQuency GHz

Select GHz as default frequency unit for all MG369xC frequency related command parameters.

:UNIT:FREQuency?

Requests the currently selected frequency unit.

:UNIT:TIME <char>

:UNIT:TIME?

Description: Selects the global default for all time related parameters used with all MG369xC SCPI commands.

- S = second
- MS = millisecond
- US = microsecond
- NS = nanosecond

Parameters: S | MS | US | NS

Default: S

Examples: :UNIT:TIME MS

Select millisecond as default time unit for all MG369xC time related command parameters.

:UNIT:TIME?

Requests the currently selected time unit.

Chapter 4 — Error Messages

4-1 Introduction

This chapter lists and describes each of the error messages related to MG369xC signal generator operation. In addition, it provides information about the error message elements, the error query command, the error queue, and the classes of error messages.

4-2 Error Query

The `:SYSTem:ERRor?` query command is a request for the next entry in the instrument's error queue. Error messages in the queue contain an integer in the range $[-32768, 32768]$ denoting an error code and associated descriptive text. Negative codes are reserved by the SCPI standard and defined first in this chapter. Positive error codes are instrument-dependent. An error code value of zero indicates that no error has occurred (see [Section 4-3](#)).

The `:SYSTem:ERRor?` query command is required of all SCPI implementations. The `:STATus:QUEue[:NEXT]?` query command when implemented is an alias to `:SYSTem:ERRor?`.

The instrument responds to the `:SYSTem:ERRor?` query command with an error message in the following format:

```
<error code>,"<error description>;<device-dependent info>"
```

The `<error code>` is a unique error descriptor. Certain standard error codes are described in this chapter. The `<error description>` is a short description of the error, (optionally) followed by further information about the error. Short descriptions of the standard error codes are given in this chapter. The `<device-dependent information>` part of the response may contain information which will allow the user to determine the exact error and context. For example:

```
-241,"Hardware missing;Attenuator"
```

The maximum string length of `<error description>` plus `<device-dependent information>` is 255 characters. The `<error description>` shall be sent exactly as indicated in this chapter including case.

4-3 Error Queue

As errors are detected, error messages are placed in a queue. This queue is first in, first out and can hold a maximum of 10 messages. If the queue overflows, the last error message in the queue is replaced with the error message:

```
-350,"Queue overflow"
```

Any time the queue overflows, the least recent error messages remain in the queue, and the most recent error message is discarded. Reading an error message from the head of the queue removes that error message from the queue, and opens a position at the tail of the queue for a new error message, if one is subsequently detected.

When all error messages have been read from the queue, further error queries shall return

```
0, "No error"
```

The error queue shall be cleared when any of the following occur (IEEE 488.2, section 11.4.3.4):

- Upon power up
- Upon receipt of a `*CLS` command
- Upon reading the last error message from the queue

4-4 Error Codes

The system-defined error codes are chosen on an enumerated (“1 of N”) basis. The SCPI-defined error codes and the <error description> portions of the query response are listed here. The first error described in each class (for example –100, –200, –300, –400) is a “generic” error. In selecting the proper error code to report, more specific error codes are preferred, and the generic error is used if the others are inappropriate.

4-5 No Error

This message indicates that the device has no errors.

Table 4-1. No Errors

Error Code	Error Description [description/explanation/examples]
0	“No error” The queue is completely empty. Every error in the queue has been read or the queue was purposely cleared by power-on, *CLS, etc.

4-6 Command Errors

An <error code> in the range [–199, –100] indicates that an IEEE 488.2 syntax error has been detected by the instrument’s parser. The occurrence of any error in this class should cause the command error bit (bit 5) in the standard event status register to be set. One of the following events has occurred:

- An IEEE 488.2 syntax error has been detected by the parser. That is, a controller-to-device message is received which is in violation of the IEEE 488.2 standard. Possible violations include a data element which violates the device listening formats or whose type is unacceptable to the device
- An unrecognized header was received. Unrecognized headers include incorrect device-specific headers and incorrect or unimplemented IEEE 488.2 common commands
- A Group Execute Trigger (GET) was entered into the input buffer inside of an IEEE 488.2<PROGRAM MESSAGE>

Events that generate command errors shall not generate execution errors, device-specific errors, or query errors; see the other error definitions in this chapter.

Table 4-2. Command Errors (1 of 3)

Error Code	Error Description [description/explanation/examples]
–102	“Syntax error” An unrecognized command or data type was encountered; for example, a string was received when the device does not accept strings.
–104	“Data type error” The parser recognized a data element different than one allowed; for example, numeric or string data was expected but block data was encountered.
–105	“GET not allowed” A Group Execute Trigger was received within a program message (see IEEE 488.2, 7.7).
–108	“Parameter not allowed” More parameters were received than expected for the header; for example, the *SAV command only accepts one parameter, so receiving *SAV 0, 1 is not allowed.
–109	“Missing parameter” Fewer parameters were received than required for the header; for example, the *SAV common command requires one parameter, so receiving *SAV is not allowed.
–110	“Command Header Error” An error was detected in the header. This error message should be used when the device cannot detect the more specific errors described for errors –111 through –119.
–111	“Header Separator Error” A character which is not a legal header separator was encountered while parsing the header; for example, no white space followed the header, thus *GMC“MACRO” is an error.
–112	“Program mnemonic too long” The header contains more than 12 characters (see IEEE 488.2, 7.6.1.4.1).
–113	“Undefined header” The header is syntactically correct, but it is undefined for this specific device; for example, *XYZ is not defined for any device.
–114	“Header suffix out of range” The value of the numeric suffix attached to a program mnemonic makes the header invalid.
–120	“Numeric data error” This error, as well as errors –121 through –129, are generated when parsing a data element which appears to be numeric, including the nondecimal types. This particular error message should be used if the device cannot detect a more specific error.

Table 4-2. Command Errors (2 of 3)

Error Code	Error Description [description/explanation/examples]
-121	<p>“Invalid character in number”</p> <p>An invalid character for the data type being parsed was encountered; for example, an alpha in a decimal numeric or a “9” in octal data.</p>
-123	<p>“Exponent too large”</p> <p>The magnitude of the exponent was larger than 32000 (see IEEE 488.2, 7.7.2.4.1).</p>
-124	<p>“Too many digits”</p> <p>The mantissa of a decimal numeric data element contained more than 255 digits excluding leading zeros (see IEEE 488.2, 7.7.2.4.1).</p>
-128	<p>“Numeric data not allowed”</p> <p>A legal numeric data element was received, but the device does not accept one in this position for the header.</p>
-130	<p>“Suffix error”</p> <p>This error, as well as errors –131 through –139, are generated when parsing a suffix. This particular error message should be used if the device cannot detect a more specific error.</p>
-131	<p>“Invalid suffix”</p> <p>The suffix does not follow the syntax described in IEEE 488.2, 7.7.3.2, or the suffix is inappropriate for this device.</p>
-134	<p>“Suffix too long”</p> <p>The suffix contained more than 12 characters (see IEEE 488.2, 7.7.3.4).</p>
-138	<p>“Suffix not allowed”</p> <p>A suffix was encountered after a numeric element which does not allow suffixes.</p>
-140	<p>“Character data error”</p> <p>This error, as well as errors 141 through 149, are generated when parsing a character data element. This particular error message should be used if the device cannot detect a more specific error.</p>
-141	<p>“Invalid character data”</p> <p>Either the character data element contains an invalid character or the particular element received is not valid for the header.</p>
-144	<p>“Character data too long”</p> <p>The character data element contains more than 12 characters (see IEEE 488.2, 7.7.1.4).</p>
-148	<p>“Character data not allowed”</p> <p>A legal character data element was encountered where prohibited by the device.</p>
-150	<p>“String data error”</p> <p>This error, as well as errors –151 through –159, are generated when parsing a string data element. This particular error message should be used if the device cannot detect a more specific error.</p>
-151	<p>“Invalid string data”</p> <p>A string data element was expected, but was invalid for some reasons (see IEEE 488.2, 7.7.5.2); for example, an END message was received before the terminal quote character.</p>
-158	<p>“String data not allowed”</p> <p>A legal string data element was encountered but was not allowed by the device at this point in parsing.</p>

Table 4-2. Command Errors (3 of 3)

Error Code	Error Description [description/explanation/examples]
-160	<p>“Block data error”</p> <p>This error, as well as errors –161 through –169, are generated when parsing a block data element. This particular error message should be used if the device cannot detect a more specific error.</p>
-161	<p>“Invalid block data”</p> <p>A block data element was expected, but invalid for some reason (see IEEE 488.2, 7.7.6.2); for example, an END message was received before the length was satisfied.</p>
-168	<p>“Block data not allowed”</p> <p>A legal block data element was encountered but was not allowed by the device at this point in parsing.</p>
-170	<p>“Expression error”</p> <p>This error, as well as errors –171 through –179, are generated when parsing an expression data element. This particular error message should be used if the device cannot detect a more specific error.</p>
-171	<p>“Invalid expression”</p> <p>The expression data element was invalid (see IEEE 488.2, 7.7.7.2); for example, unmatched parentheses or an illegal character.</p>
-178	<p>“Expression data not allowed”</p> <p>A legal expression data element was encountered but was not allowed by the device at this point in parsing.</p>
-180	<p>“Macro error”</p> <p>This error, as well as errors –181 through –189, are generated when defining a macro or executing a macro. This particular error message should be used if the device cannot detect a more specific error.</p>
-181	<p>“Invalid outside macro definition”</p> <p>Indicates that a macro parameter placeholder (\$<number>) was encountered outside of a macro definition.</p>
-183	<p>“Invalid inside macro definition”</p> <p>Indicates that the program message unit sequence, sent with a *DDT or *DMC command, is syntactically invalid (see IEEE 488.2, 10.7.6.3).</p>
-184	<p>“Macro parameter error”</p> <p>Indicates that a command inside the macro definition had the wrong number or type of parameters.</p>

4-7 Execution Errors

An <error code> in the range [-299,-200] indicates that an error has been detected by the instrument's execution control block. The occurrence of any error in this class should cause the execution error bit (bit 4) of the standard event status register to be set. One of the following events has occurred:

- A <PROGRAM DATA> element following a header was evaluated by the device as outside its legal input range or is otherwise inconsistent with the device's capability
- A valid program message could not be properly executed due to some device condition

Execution errors shall be reported by the device after rounding and expression evaluation operations have taken place. Rounding a numeric data element, for example, shall not be reported as an execution error. Events that generate execution errors shall not generate command errors, device-specific errors, or query errors; see the other error definitions in this chapter.

Table 4-3. Execution Errors (1 of 4)

Error Code	Error Description [description/explanation/examples]
-200	"Execution error" This is a generic syntax error for devices that cannot detect more specific errors. This code indicates only that an execution error as defined in IEEE 488.2, 11.5.1.1.5 has occurred.
-201	"Invalid while in local" Indicates that a command is not executable while the device is in local due to a hard local control (see IEEE 488.2, 5.6.1.5); for example, a device with a rotary switch receives a message which would change the switches state, but the device is in local so the message cannot be executed.
-202	"Settings lost due to rtl" Indicates that a setting associated with a hard local control (see IEEE 488.2, 5.6.1.5) was lost when the device was changed to LOCS from REMS or to LWLS from RWLS.
-210	"Trigger error" A trigger error occurred in the signal generator.
-211	"Trigger Ignored" Indicates that a GET, *TRG, or triggering signal was received and recognized by the device but was ignored because of device timing considerations; for example, the device was not ready to respond. <i>Note:</i> a DTO device always ignores GET and treats *TRG as a command error.
-212	"Arm ignored" Indicates that an arming signal was received and recognized by the device but was ignored.
-213	"Init ignored" Indicates that a request for measurement initiation was ignored as another measurement was already in progress.
-214	"Trigger deadlock" Indicates that the trigger source for the initiation of a measurement is set to GET and subsequent measurement query is received. The measurement cannot be started until a GET is received, but the GET would cause an INTERRUPTED error.
-215	"Arm deadlock" Indicates that the arm source for the initiation of a measurement is set to GET and subsequent measurement query is received. The measurement cannot be started until a GET is received, but the GET would cause an INTERRUPTED error.
-220	"Parameter error" Indicates that a program data element related error occurred. This error message should be used when the device cannot detect the more specific errors described for errors -221 to -229.

Table 4-3. Execution Errors (2 of 4)

Error Code	Error Description [description/explanation/examples]
-221	<p>“Settings conflict”</p> <p>Indicates that a legal program data element was parsed but could not be executed due to the current device state (see IEEE 488.2, 6.4.5.3 and 11.5.1.1.5).</p>
-222	<p>“Data out of range”</p> <p>Indicates that a legal program data element was parsed but could not be executed because the interpreted value was outside the legal range as defined by the device (see IEEE 488.2, 11.5.1.1.5).</p>
-223	<p>“Too much data”</p> <p>Indicates that a legal program data element of block, expression, or string type was received that contained more data than the device could handle due to memory or related device-specific requirements.</p>
-224	<p>“Illegal parameter value”</p> <p>Used where exact value, from a list of possibles, was expected.</p>
-230	<p>“Data corrupt or stale”</p> <p>Possibly invalid data; new reading started but not completed since last access.</p>
-231	<p>“Data questionable”</p> <p>Indicates that measurement accuracy is suspect.</p>
-240	<p>“Hardware error”</p> <p>Indicates that a legal program command or query could not be executed because of a hardware problem in the device. Definition of what constitutes a hardware problem is completely device-specific. This error message should be used when the device cannot detect the more specific errors described for errors –241 through –249.</p>
-241	<p>“Hardware missing”</p> <p>Indicates that a legal program command or query could not be executed because of missing device hardware; for example, an option was not installed. Definition of what constitutes missing hardware is completely device specific.</p>
-250	<p>“Mass storage error”</p> <p>Indicates that a mass storage error occurred. This error message should be used when the device cannot detect the more specific errors described for errors –251 through –259.</p>
-251	<p>“Missing mass storage”</p> <p>Indicates that a legal program command or query could not be executed because of missing mass storage; for example, an option that was not installed. Definition of what constitutes missing mass storage is device-specific.</p>
-252	<p>“Missing media”</p> <p>Indicates that a legal program command or query could not be executed because of missing media; for example, no disk. Definition of what constitutes missing media is device-specific.</p>
-253	<p>“Corrupt media”</p> <p>Indicates that a legal program command or query could not be executed because of corrupt media; for example, bad disk or wrong format. Definition of what constitutes corrupt media is device-specific.</p>
-254	<p>“Media full”</p> <p>Indicates that a legal program command or query could not be executed because the media was full; for example, there is no room on the disk. The definition of what constitutes a full media is device-specific.</p>

Table 4-3. Execution Errors (3 of 4)

Error Code	Error Description [description/explanation/examples]
-255	<p>“Directory full”</p> <p>Indicates that a legal program command or query could not be executed because the media directory was full. The definition of what constitutes a full media directory is device-specific.</p>
-256	<p>“File name not found”</p> <p>Indicates that a legal program command or query could not be executed because the file name on the device media was not found; for example, an attempt was made to read or copy a nonexistent file. The definition of what constitutes a file not being found is device-specific.</p>
-257	<p>“File name error”</p> <p>Indicates that a legal program command or query could not be executed because the file name on the device media was in error; for example, an attempt was made to copy to a duplicate file name. The definition of what constitutes a file name error is device-specific.</p>
-258	<p>“Media protected”</p> <p>Indicates that a legal program command or query could not be executed because the media was protected; for example, the write-protect tab on a disk was present. The definition of what constitutes protected media is device-specific.</p>
-260	<p>“Expression error”</p> <p>Indicates that an expression program data element related error occurred. This error message should be used when the device cannot detect the more specific errors described for errors –261 through –269.</p>
-261	<p>“Math error in expression”</p> <p>Indicates that a syntactically legal expression program data element could not be executed due to a math error; for example, a divide-by-zero was attempted. The definition of math error is device-specific.</p>
-270	<p>“Macro error”</p> <p>Indicates that a macro-related execution error occurred. This error message should be used when the device cannot detect the more specific errors described for errors –271 through –279.</p>
-271	<p>“Macro syntax error”</p> <p>Indicates that a syntactically legal macro program data sequence, according to IEEE 488.2, 10.7.2, could not be executed due to a syntax error within the macro definition (see IEEE 488.2, 10.7.6.3).</p>
-272	<p>“Macro execution error”</p> <p>Indicates that a syntactically legal macro program data sequence could not be executed due to a some error in the macro definition (see IEEE 488.2, 10.7.6.3).</p>
-273	<p>“Illegal macro label”</p> <p>Indicates that the macro label defined in the *DMC command was a legal string syntax, but could not be accepted by the device (see IEEE 488.2, 10.7.3 and 10.7.6.2); for example, the label was too long, the same as a common command header, or contained invalid header syntax.</p>
-274	<p>“Macro parameter error”</p> <p>Indicates that the macro definition improperly used a macro parameter placeholder (see IEEE 488.2, 10.7.3).</p>
-275	<p>“Macro definition too long”</p> <p>Indicates that a syntactically legal macro program data sequence could not be executed because the string or block contents were too long for the device to handle (see IEEE 488.2, 10.7.6.1).</p>
-276	<p>“Macro recursion error”</p> <p>Indicates that a syntactically legal macro program data sequence could not be executed because the device found it to be recursive (see IEEE 488.2, 10.7.6.6).</p>

Table 4-3. Execution Errors (4 of 4)

Error Code	Error Description [description/explanation/examples]
-277	<p>“Macro redefinition not allowed”</p> <p>Indicates that a syntactically legal macro label in the *DMC command could not be executed because the macro label was already defined (see IEEE 488.2, 10.7.6.4).</p>
-278	<p>“Macro header not found”</p> <p>Indicates that a syntactically legal macro label in the *GMC? query could not be executed because the header was not previously defined.</p>
-280	<p>“Program error”</p> <p>Indicates that a downloaded program-related execution error occurred. This error message should be used when the device cannot detect the more specific errors described for errors –281 through –289.</p>
-281	<p>“Cannot create program”</p> <p>Indicates that an attempt to create a program was unsuccessful. A reason for the failure might include not enough memory.</p>
-282	<p>“Illegal program name”</p> <p>The name used to reference a program was invalid; for example, redefining an existing program, deleting a nonexistent program, or in general, referencing a nonexistent program.</p>
-283	<p>“Illegal variable name”</p> <p>An attempt was made to reference a nonexistent variable in a program.</p>
-284	<p>“Program currently running”</p> <p>Certain operations dealing with programs may be illegal while the program is running; for example, deleting a running program might not be possible.</p>
-285	<p>“Program syntax error”</p> <p>Indicates that a syntax error appears in a downloaded program. The syntax used when parsing the downloaded program is device-specific.</p>
-286	<p>“Program runtime error”</p>

4-8 Device-Specific Errors

An <error code> in the range [–399,–300] or [1, 32767] indicates that the instrument has detected an error which is not a command error, a query error, or an execution error; some device operations did not properly complete, possibly due to an abnormal hardware or firmware condition. These codes are also used for self-test response errors. The occurrence of any error in this class should cause the device-specific error bit (bit 3) in the standard event status register to be set. The meaning of positive error codes is device-dependent and may be enumerated or bit mapped; the <error message> string for positive error codes is not defined by SCPI and available to the device designer. Note that the string is not optional; if the designer does not wish to implement a string for a particular error, the null string should be sent (for example, 42,""). The occurrence of any error in this class should cause the device-specific error bit (bit 3) in the standard event status register to be set. Events that generate device-specific errors shall not generate command errors, execution errors, or query errors; see the other error definitions in this chapter.

Table 4-4. Device-Specific Errors

Error Code	Error Description [description/explanation/examples]
–300	<p>“Device-specific error”</p> <p>This is the generic device-dependent error for devices that cannot detect more specific errors. This code indicates only that a Device-Dependent Error as defined in IEEE 488.2, 11.5.1.1.6 has occurred.</p>
–310	<p>“System error”</p> <p>Indicates that some error, termed “system error” by the device, has occurred. This code is device-dependent.</p>
–311	<p>“Memory error”</p> <p>Indicates that an error was detected in the device’s memory. The scope of this error is device-dependent.</p>
–312	<p>“PUD memory lost”</p> <p>Indicates that the protected user data saved by the *PUD command has been lost.</p>
–313	<p>“Calibration memory lost”</p> <p>Indicates that nonvolatile calibration data used by the *CAL? command has been lost.</p>
–314	<p>“Save/recall memory lost”</p> <p>Indicates that the nonvolatile data saved by the *SAV? command has been lost.</p>
–315	<p>“Configuration memory lost”</p> <p>Indicates that nonvolatile configuration data saved by the device has been lost. The meaning of this error is device-specific.</p>
–330	<p>“Self-test failed”</p>
–350	<p>“Queue overflow”</p> <p>A specific code entered into the queue in lieu of the code that caused the error. This code indicates that there is no room in the queue and an error occurred but was not recorded.</p>

4-9 Query Errors

An <error code> in the range [-499,-400] indicates that the output queue control of the instrument has detected a problem with the message exchange protocol described in IEEE 488.2, Chapter 6. The occurrence of any error in this class should cause the query error bit (bit 2) in the standard event status register to be set. These errors correspond to message exchange protocol errors described in IEEE 488.2, Section 6.5. One of the following is true:

- An attempt is being made to read data from the output queue when no output is either present or pending
- Data in the output queue has been lost

Events that generate query errors shall not generate command errors, execution errors, or device-specific errors; see the other error definitions in this chapter.

Table 4-5.

Error Code	Error Description [description/explanation/examples]
-400	<p>“Query error”</p> <p>This is the generic query error for devices that cannot detect more specific errors. This code indicates only that a Query Error as defined in IEEE 488.2, 11.5.1.1.7 and 6.3 has occurred.</p>
-410	<p>“Query INTERRUPTED”</p> <p>Indicates that a condition causing an INTERRUPTED Query error occurred (see IEEE 488.2, 6.3.2.3); for example, a query followed by a DAB or GET before a response was completely sent.</p>
-420	<p>“Query UNTERMINATED”</p> <p>Indicates that a condition causing as UNTERMINATED Query error occurred (see IEEE 488.2, 6.3.2.2); for example, the device was addressed to talk and an incomplete program message was received.</p>
-430	<p>“Query DEADLOCKED”</p> <p>Indicates that a condition causing a DEADLOCKED Query error occurred (see IEEE 488.2, 6.3.1.7); for example, both input and output buffer are full and the device cannot continue.</p>
-440	<p>“Query UNTERMINATED after indefinite response”</p> <p>Indicates that a query was received in the same program message after a query requesting an indefinite response was executed (see IEEE 488.2, 6.5.7.5.7).</p>

4-10 Parser Errors

An <error code> in the range [201, 212] is generated by the instrument's parser in response to the error condition described.

Table 4-6.

Error Code	Error Description [description/explanation/examples]
201	"Query only" Indicates the command is a query command only.
202	"No query allowed" Indicates that a query form of the command is not available.
203	"Parameter(s) not expected" Indicates that the command does not take any parameter(s).
204	"Constant not allowed in STATUS subsystem"
207	"Enumeric value not in union" Indicates an illegal parameter was sent with the command.
208	"Illegal number of parameters" Indicates an illegal number of parameters was sent with the command.
210	"Run out of memory handle" Indicates an internal parser problem.
211	"Unit not matched" Indicates the parameter unit does not match any defined unit for this parameter.
212	"Unit not required" Indicates no unit is required with this parameter.

4-11 Self-Test Errors

An <error code> in the range [100, 199] indicates that a failure has occurred during instrument self-test. The error messages are placed in the error queue in the order they occur.

Table 4-7. Self-Test Errors (1 of 3)

Error Code	Error Description [description/explanation/examples]
100	"Failed: DVM ground offset" Indicates self-test failed because of a calibration-related problem.
101	"Failed: DVM +10V ref." Indicates self-test failed because of either a calibration-related problem or a defective +10 Volt reference.
102	"Failed: DVM -10V ref." Indicates self-test failed because of either a calibration-related problem or a defective -10 Volt reference.
105	"Failed: Power Supply Voltages(s) out of regulation" Indicates one or more of the voltages from the power supply is out of regulation.
106	"Not locked: Power supply" Indicates the power supply is not phase-locked to the 400 kHz reference frequency.
107	"Failed; Sweep time circuitry" Indicates the sweep timing is out of tolerance or has failed.
108	"Not ready: Crystal oven cold" Indicates the 100 MHz crystal oven or the Option 16 high-stability 10 MHz crystal oscillator has not reached operating temperature.
109	"Not locked: Ext 10MHZ" Indicates the reference loop is not phase-locked to the external 10 MHz reference.
110	"Not locked: High Stability crystal" Indicates the reference loop is not phase-locked to the high stability 10 MHz crystal oscillator.
111	"Not locked: Fine Loop" Indicates one or more of the oscillators within the fine loop is not phase-locked.
112	"Not locked: Coarse Loop" Indicates the coarse loop oscillator is not phase-locked.
113	"Not locked: YIG Loop" Indicates the YIG loop is not phase-locked.
114	"Not locked: Down Converter" Indicates the local oscillator in the down converter assembly is not phase-locked.
115	"Failed: Not Locked indicator" Indicates failure of the not phase-locked indicator circuit.
116	"Failed: FM loop gain circuit" Indicates the FM loop has failed or the loop gain is out of tolerance.
117	"Failed: Linearizer circuit" Indicates failure of the Linearizer circuit on the A12 PCB.
118	"Failed: Marker Switch Point circuit" Indicates failure of the Marker Switch Point circuit on the A12 PCB.

Table 4-7. Self-Test Errors (2 of 3)

Error Code	Error Description [description/explanation/examples]
119	"Failed: Center Frequency circuit" Indicates failure of the Center Frequency circuit on the A12 PCB.
120	"Failed: Delta-F Ramp circuit" Indicates failure of the DF Ramp circuit on the A12 PCB.
121	"Failed: Unleveled indicator" Indicates failure of the not leveled detector circuit on the A10 PCB.
122	"Failed: Level reference" Indicates failure of the level reference circuit on the A10 PCB.
123	"Failed: Detector log amp" Indicates failure of the level detector log amplifier circuit on the A10 PCB.
124	"Unleveled and not locked: Full band" Indicates failure of both YIG-tuned oscillators.
125	"Unleveled and not locked: 8.4-20 GHz range" Indicates failure of the 8.4 to 20 GHz YIG-tuned oscillator.
126	"Unleveled and not locked: 2-8.4 GHz range" Indicates failure of the 2 to 8.4 GHz YIG-tuned oscillator.
127	"Failed: A10 Detector input circuit" Indicates failure of the level detector input circuit on the A10 PCB.
128	"Failed: 0.01-2 GHz range unleveled" Indicates failure of the Down Converter leveling circuitry.
129	"Failed: Switched filter or level detector" Indicates failure of either the switched filter or level detector circuitry.
130	"Failed: 2-3.3 GHz Switched filter section or level detector" Indicates failure of either the 2 to 3.3 GHz switched filter path or the level detector circuitry.
131	"Failed: 3.3-5.5 GHz Switched filter section or level detector" Indicates failure of either the 3.3 to 5.5 GHz switched filter path or the level detector circuitry.
132	"Failed: 5.5-8.4 GHz Switched filter section or level detector" Indicates failure of either the 5.5 to 8.4 GHz switched filter path or the level detector circuitry.
133	"Failed: 8.4-13.25 GHz Switched filter section or level detector" Indicates failure of either the 8.4 to 13.25 GHz switched filter path or the level detector circuitry.
134	"Failed: 13.25-20 GHz Switched filter section or level detector" Indicates failure of either the 13.25 to 20 GHz switched filter path or the level detector circuitry.
135	"Failed: A9 modulator or driver" Indicates failure of either the modulator in the switched filter assembly or the modulator driver circuitry on the A9 PCB.
138	"Failed: Freq extension unit or driver" Indicates failure of the frequency extension unit (FEU) or FEU driver circuitry on the A9 PCB.
139	"Failed: 33-40 GHz section of freq extension unit" Indicates failure of the 33 to 40 GHz section of the FEU.
140	"Failed: 26.5-33 GHz section of freq extension unit" Indicates failure of the 26.5 to 33 GHz section of the FEU.

Table 4-7. Self-Test Errors (3 of 3)

Error Code	Error Description [description/explanation/examples]
141	"Failed: 20-26.5 GHz section of freq extension unit" Indicates failure of the 20 to 26.5 GHz section of the FEU.
142	"Failed: Sample and hold circuit" Indicates failure of the sample and hold circuitry on the A10 PCB.
143	"Failed: Slope DAC or associated circuit" Indicates failure of the level slope DAC circuitry on the A10 PCB.
144	"Failed: RF was off when self test started" Indicates that some tests were not performed because the RF Output was selected OFF when self-test was started.
145	"Failed: A12 +10V ref." Indicates failure of the +10 Volt reference circuit on the A12 PCB.
146	"Failed: A12 -10V ref." Indicates failure of the -10 Volt reference circuit on the A12 PCB.
199	"Self Test Complete"

Appendix A — Overall Command Tree

A-1 Introduction

This appendix provides an overall command tree for the Series MG369xC Synthesized High Performance Signal Generator SCPI command set. The command tree is shown in [Figure A-1](#). Refer to [Chapter 3](#) for information on the individual SCPI commands.

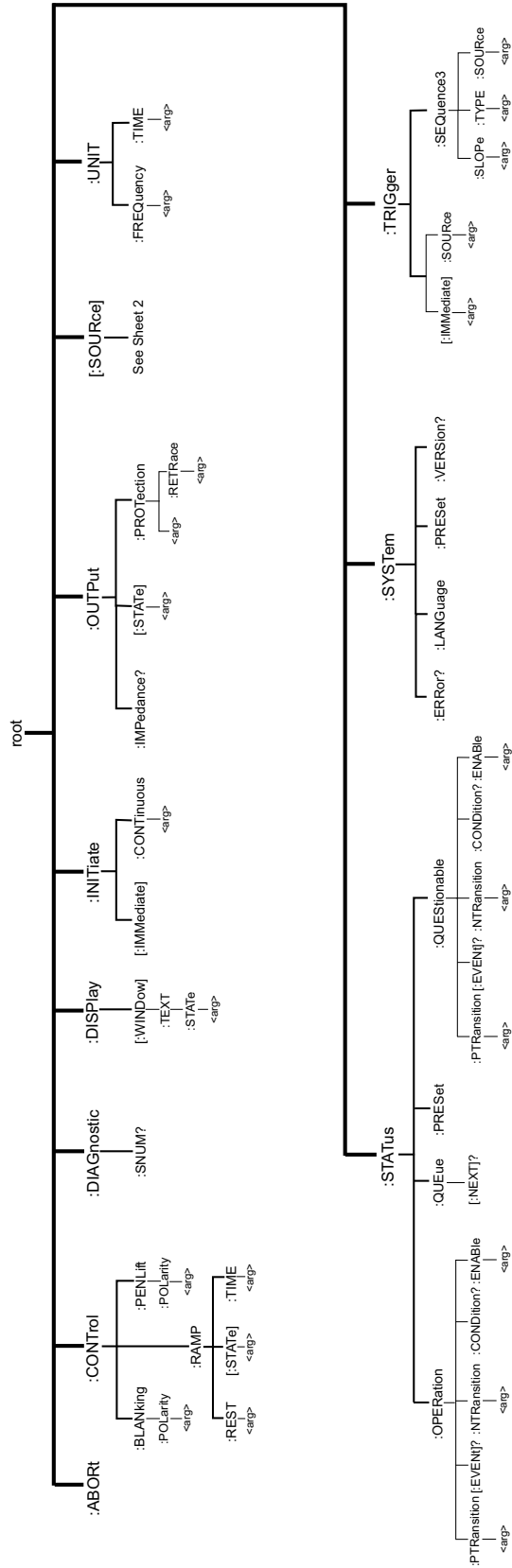


Figure A-1. Overall MG369xC SCPI Command Tree (Sheet 1 of 2)

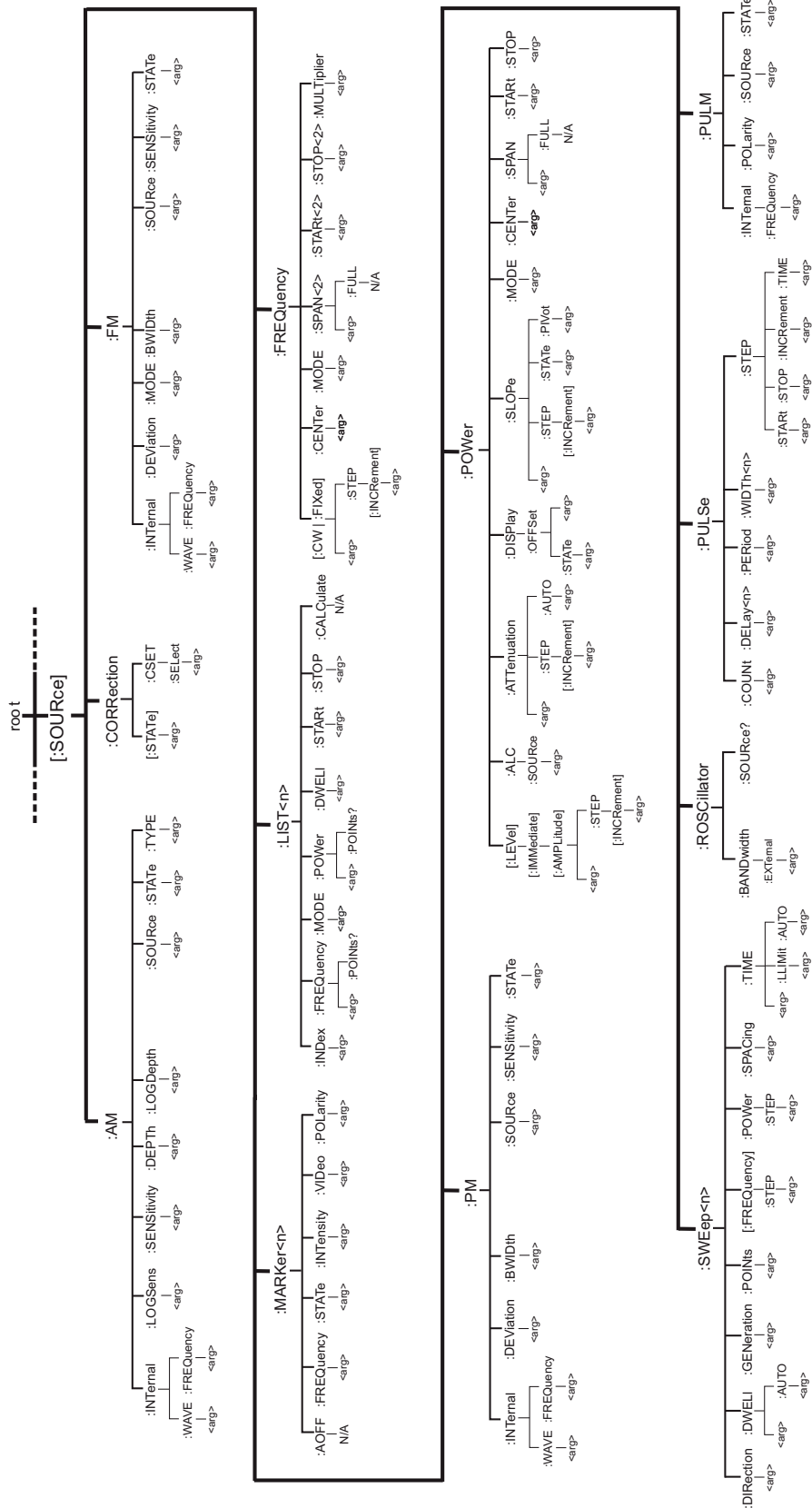


Figure A-2. Overall MG369xC SCPI Command Tree (Sheet 2 of 2)

Appendix B — SCPI Conformance Information

B-1 Introduction

This appendix provides SCPI conformance information for the MG369xC SCPI command set in the form of a command summary. The MG369xC SCPI command set commands and queries are described individually in [Chapter 3, “Programming Commands”](#).

The SCPI version that the MG369xC software supports is *Standard Commands for Programmable Instruments (SCPI) 1993.0*. [Table B-1](#) lists the SCPI Common Commands along with the conformance information for each command. The categories used for these commands are: IEEE 488.2 Required, and IEEE 488.2 Optional.

[Table B-2](#) lists all the other SCPI commands and queries in the command set. The conformance categories used are: SCPI Confirmed and Non-SCPI.

Note

In the following tables, a question mark enclosed in parentheses [(?)] at the end of an entry indicates that the particular command exists in both command and query forms.

Table B-1. SCPI Common Commands

SCPI Command	Status	Page
*CLS	IEEE 488.2 Required	3-1
*ESE <nv>	IEEE 488.2 Required	3-1
*ESE?	IEEE 488.2 Required	3-1
*ESR?	IEEE 488.2 Required	3-1
*IDN?	IEEE 488.2 Required	3-1
*OPC	IEEE 488.2 Required	3-1
*OPC?	IEEE 488.2 Required	3-1
*RST	IEEE 488.2 Required	3-2
*SRE <nv>	IEEE 488.2 Required	3-2
*SRE?	IEEE 488.2 Required	3-2
*STB?	IEEE 488.2 Required	3-2
*TST?	IEEE 488.2 Required	3-2
*WAI	IEEE 488.2 Required	3-2
*OPT?	IEEE 488.2 Optional	3-2
*RCL <n>	IEEE 488.2 Optional	3-2
*SAV <n>	IEEE 488.2 Optional	3-2
*TRG	IEEE 488.2 Optional	3-3

Table B-2. Other SCPI Commands (1 of 4)

SCPI Command	Status	Page
:ABORT	SCPI Confirmed	3-4
:CONTRol:BLANKing:POLarity(?)	Non-SCPI	3-5
:CONTRol:PENLift:POLarity(?)	Non-SCPI	3-5
:CONTRol:RAMP:REST(?)	Non-SCPI	3-6
:CONTRol:RAMP[:STATe](?)	Non-SCPI	3-6
:CONTRol:RAMP:TIME(?)	Non-SCPI	3-6
:DIAGnostic:SNUM?	Non-SCPI	3-7
:DISPlay[:WINDow]:TEXT:STATe(?)	SCPI Confirmed	3-8
:INITiate[:IMMediate]	SCPI Confirmed	3-9
:INITiate:CONTInuous(?)	SCPI Confirmed	3-9
:OUTPut[:STATe](?)	SCPI Confirmed	3-10
:OUTPut:PROTection(?)	Non-SCPI	3-10
:OUTPut:PROTection:RETRace(?)	Non-SCPI	3-11
:OUTPut:IMPedance?	Non-SCPI	3-11
[:SOURce]:AM:LOGSens(?)	Non-SCPI	3-16
[:SOURce]:AM:SENSitivity(?)	SCPI Confirmed	3-16
[:SOURce]:AM:LOGDepth(?)	Non-SCPI	3-16
[:SOURce]:AM:INTernal:WAVE(?)	Non-SCPI	3-17
[:SOURce]:AM:INTernal:FREQuency(?)	SCPI Confirmed	3-17
[:SOURce]:AM:DEPTH(?)	SCPI Confirmed	3-17
[:SOURce]:AM:SOURce(?)	SCPI Confirmed	3-18
[:SOURce]:AM:STATe(?)	SCPI Confirmed	3-18
[:SOURce]:AM:TYPE(?)	SCPI Confirmed	3-18
[:SOURce]:CORRection[:STATe](?)	SCPI Confirmed	3-19
[:SOURce]:CORRection:CSET:SElect(?)	SCPI Confirmed	3-19
[:SOURce]:FM:INTernal:WAVE(?)	Non-SCPI	3-19
[:SOURce]:FM:INTernal:FREQuency(?)	SCPI Confirmed	3-20
[:SOURce]:FM:DEVIation(?)	SCPI Confirmed	3-20
[:SOURce]:FM:MODE(?)	SCPI Confirmed	3-21
[:SOURce]:FM:BWIDth(?)	Non-SCPI	3-21
[:SOURce]:FM:SOURce(?)	SCPI Confirmed	3-22
[:SOURce]:FM:SENSitivity(?)	SCPI Confirmed	3-22
[:SOURce]:FM:STATe(?)	SCPI Confirmed	3-22
[:SOURce]:FREQuency[:CW FIXed](?)	SCPI Confirmed	3-23
[:SOURce]:FREQuency[:CW FIXed] :STEP[:INCRement](?)	Non-SCPI	3-24
[:SOURce]:FREQuency:CENTer(?)	SCPI Confirmed	3-24
[:SOURce]:FREQuency:MODE(?)	SCPI Confirmed	3-25
[:SOURce]:FREQuency:SPAN(?)	SCPI Confirmed	3-25

Table B-2. Other SCPI Commands (2 of 4)

SCPI Command	Status	Page
[[:SOURce]:FREQUency:SPAN:FULL	SCPI Confirmed	3-26
[[:SOURce]:FREQUency:START(?]	SCPI Confirmed	3-27
[[:SOURce]:FREQUency:START2(?]	SCPI Confirmed	3-27
[[:SOURce]:FREQUency:STOP(?]	SCPI Confirmed	3-28
[[:SOURce]:FREQUency:STOP2(?]	SCPI Confirmed	3-28
[[:SOURce]:FREQUency:MULTIplier(?]	SCPI Confirmed	3-29
[[:SOURce]:LIST<n>:INDEX(?]	Non-SCPI	3-29
[[:SOURce]:LIST<n>:FREQUency(?]	SCPI Confirmed	3-30
[[:SOURce]:LIST<n>:FREQUency:POINTs?	SCPI Confirmed	3-31
[[:SOURce]:LIST<n>:MODE(?]	Non-SCPI	3-31
[[:SOURce]:LIST<n>:POWER(?]	SCPI Confirmed	3-31
[[:SOURce]:LIST<n>:POWER:POINTs?	SCPI Confirmed	3-32
[[:SOURce]:LIST<n>:DWELI(?]	Non-SCPI	3-32
[[:SOURce]:LIST<n>:START(?]	Non-SCPI	3-32
[[:SOURce]:LIST<n>:STOP(?]	Non-SCPI	3-32
[[:SOURce]:LIST<n>:CALCulate	Non-SCPI	3-33
[[:SOURce]MARKer<n>:AOFF	SCPI Confirmed	3-34
[[:SOURce]MARKer<n>:FREQUency(?]	SCPI Confirmed	3-34
[[:SOURce]MARKer<n>:STATe(?]	SCPI Confirmed	3-35
[[:SOURce]MARKer<n>:INTensity(?]	Non-SCPI	3-35
[[:SOURce]MARKer<n>:VIDeo(?]	Non-SCPI	3-35
[[:SOURce]MARKer<n>:POLarity(?]	Non-SCPI	3-36
[[:SOURce]PM:BWIDth(?]	Non-SCPI	3-37
[[:SOURce]:PM:DEVIation(?]	SCPI Confirmed	3-37
[[:SOURce]:PM:INTernal:WAVE(?]	Non-SCPI	3-38
[[:SOURce]:PM:INTernal:FREQUency(?]	SCPI Confirmed	3-38
[[:SOURce]:PM:SENSitivity(?]	SCPI Confirmed	3-39
[[:SOURce]:PM:SOURce(?]	SCPI Confirmed	3-39
[[:SOURce]:PM:STATe(?]	SCPI Confirmed	3-39
[[:SOURce]:POWER[:LEVel][:IMMediate][:AMPLitude](?)	SCPI Confirmed	3-40
[[:SOURce]:POWER[:LEVel][:IMMediate][:AMPLitude]:STEP[:INCRement](?)	Non-SCPI	3-41
[[:SOURce]:POWER[:LEVel]:ALTErnate(?]	Non-SCPI	3-41
[[:SOURce]:POWER:ALC:SOURce(?]	SCPI Confirmed	3-42
[[:SOURce]:POWER:ATTenuation(?]	SCPI Confirmed	3-43
[[:SOURce]:POWER:ATTenuation:STEP[:INCRement](?)	Non-SCPI	3-43
[[:SOURce]:POWER:ATTenuation:AUTO (?]	SCPI Confirmed	3-44
[[:SOURce]:POWER:DISPlay:OFFSet (?]	Non-SCPI	3-44
[[:SOURce]:POWER:DISPlay:OFFSet:STATe(?]	Non-SCPI	3-45
[[:SOURce]:POWER:SLOPe(?]	Non-SCPI	3-45
[[:SOURce]:POWER:SLOPe:STEP[:INCRement](?)	Non-SCPI	3-45

B-1 Introduction

Table B-2. Other SCPI Commands (3 of 4)

SCPI Command	Status	Page
[:SOURce]:POWer:SLOPe:STATe(?)	Non-SCPI	3-46
[:SOURce]:POWer:SLOPe:PIVot(?)	Non-SCPI	3-46
[:SOURce]:POWer:MODE(?)	SCPI Confirmed	3-46
[:SOURce]:POWer:CENTer(?)	SCPI Confirmed	3-47
[:SOURce]:POWer:SPAN(?)	SCPI Confirmed	3-47
[:SOURce]:POWer:SPAN:FULL	SCPI Confirmed	3-47
[:SOURce]:POWer:START(?)	SCPI Confirmed	3-48
[:SOURce]:POWer:STOP(?)	SCPI Confirmed	3-48
[:SOURce]:PULM:INTernal:FREQuency(?)	SCPI Confirmed	3-49
[:SOURce]:PULM:POLarity(?)	SCPI Confirmed	3-49
[:SOURce]:PULM:SOURce(?)	SCPI Confirmed	3-50
[:SOURce]:PULM:STATe(?)	SCPI Confirmed	3-50
[:SOURce]:PULSe:COUNT(?)	Non-SCPI	3-51
[:SOURce]:PULSe:DELay<n>(?)	SCPI Confirmed	3-51
[:SOURce]:PULSe:PERiod(?)	SCPI Confirmed	3-52
[:SOURce]:PULSe:WIDTh<n>(?)	SCPI Confirmed	3-52
[:SOURce]:PULSe:STEP(?)	Non-SCPI	3-53
[:SOURce]:PULSe:STEP:START(?)	Non-SCPI	3-53
[:SOURce]:PULSe:STEP:STOP(?)	Non-SCPI	3-54
[:SOURce]:PULSe:STEP:INCRement(?)	Non-SCPI	3-54
[:SOURce]:PULSe:STEP:TIME(?)	Non-SCPI	3-55
[:SOURce]:ROSCillator:BANDwidth:EXTernal(?)	Non-SCPI	3-55
[:SOURce]:ROSCillator:SOURce?	Non-SCPI	3-55
[:SOURce]:SCAN:STATe(?)	Non-SCPI	3-56
[:SOURce]:SWEep<n>:DIRection(?)	SCPI Confirmed	3-56
[:SOURce]:SWEep<n>:DWELI(?)	SCPI Confirmed	3-57
[:SOURce]:SWEep<n>:DWELI:AUTO(?)	SCPI Confirmed	3-58
[:SOURce]:SWEep<n>:GENeration(?)	SCPI Confirmed	3-59
[:SOURce]:SWEep<n>:POINts(?)	SCPI Confirmed	3-60
[:SOURce]:SWEep<n>[:FREQuency]:STEP(?)	Non-SCPI	3-61
[:SOURce]:SWEep<n>:POWer:STEP(?)	Non-SCPI	3-62
[:SOURce]:SWEep<n>:SPACing	SCPI Confirmed	3-62
[:SOURce]:SWEep<n>:TIME(?)	SCPI Confirmed	3-63
[:SOURce]:SWEep<n>:TIME:LLIMit(?)	SCPI Confirmed	3-63
[:SOURce]:SWEep<n>:TIME:AUTO(?)	SCPI Confirmed	3-64
:STATus:OPERation[:EVENT]?	SCPI Confirmed	3-65
:STATus:OPERation:CONDition?	SCPI Confirmed	3-65
:STATus:OPERation:ENABle(?)	SCPI Confirmed	3-66
:STATus:OPERation:PTRansition(?)	SCPI Confirmed	3-66
:STATus:OPERation:NTRansition(?)	SCPI Confirmed	3-66

Table B-2. Other SCPI Commands (4 of 4)

SCPI Command	Status	Page
:STATus:PRESet	SCPI Confirmed	3-67
:STATus:QUEStionable[:EVENT]?	SCPI Confirmed	3-67
:STATus:QUEStionable:CONDition?	SCPI Confirmed	3-67
:STATus:QUEStionable:ENABle(?)	SCPI Confirmed	3-67
:STATus:QUEStionable:PTRansition(?)	SCPI Confirmed	3-68
:STATus:QUEStionable:NTRansition(?)	SCPI Confirmed	3-68
:STATus:QUEue[:NEXT]?	SCPI Confirmed	3-69
:SYSTem:ERRor?	SCPI Confirmed	3-70
:SYSTem:LANGuage(?)	SCPI Confirmed	3-71
:SYSTem:PRESet	SCPI Confirmed	3-71
:SYSTem:VERSion?	SCPI Confirmed	3-71
:TRIGger[:IMMediate]	SCPI Confirmed	3-72
:TRIGger:SOURce(?)	SCPI Confirmed	3-72
:TRIGger:SEQuence3:SLOPe(?)	SCPI Confirmed	3-73
:TRIGger:SEQuence3:TYPE(?)	Non-SCPI	3-73
:TRIGger:SEQuence3:SOURce(?)	SCPI Confirmed	3-74
:TSWeep	Non-SCPI	3-74
:UNIT:FREQuency(?)	Non-SCPI	3-75
:UNIT:TIME(?)	SCPI Confirmed	3-75

Index

A		Interface Function Messages	1-7
ABORt Subsystem	3-4	L	
B		Listener	1-4
Bus Structure	1-4	M	
C		Maintenance	1-1
CAT5 Cable	1-10	Manual Information	1-1
Command Errors	4-3	Related	1-1
Command Names	2-2	Message Types	1-7
Command Tree	A-1	N	
Common Commands	2-1, 3-1, B-1	Notational Conventions	2-5
Condition Register	2-7	O	
CONTrol Subsystem	3-5	Operation	1-9
Controller	1-4	Optional Common Commands	3-2
D		Optional SCPI Commands	2-2
Data and Instrument Status Messages	1-8	OUTPut Subsystem	3-10
Data Parameters	2-4	P	
Device-Specific Commands	1-7	Parser Errors	4-12
Device-Specific Error	4-10	Q	
DIAGnostic Subsystem	3-7	Query Command	2-2
DISPlay Subsystem	3-8	Query Errors	4-11
E		R	
Enable Register	2-7	Remote	2-6
Error Messages	4-1	Required SCPI Commands	2-2
Ethernet	1-10	RJ45 Connector	1-10
Control Example in C# using NI-VISA	1-22	S	
General Requirements	1-13	SCPI Command Types	2-1
Reset Ethernet IP Address	1-24	SCPI Commands	B-2
Setup - DHCP with AutoIP	1-17	SCPI Conformance	B-1
Setup - Static Configuration	1-14	Self-Test Errors	4-13
TCP/IP	1-10	SOURce Subsystems	3-12
VXI-11	1-10	Status Group Registers	2-7
Event Register	2-7	STATus Subsystem	3-65
Execution Errors	4-6	Subsystem Commands	3-3
F		System Controller	1-4
Front Panel Control	2-6	SYSTem Subsystem	3-70
G		T	
General Purpose Interface Bus (GPIB)	1-2	Talker	1-4
Cable Length Restrictions	1-11	TCP/IP	1-10
Setting Address	1-11	Default Gateway	1-13
Setup and Interconnection	1-10	General Requirements	1-13
GPIB Programming	1-1	IP Address	1-13
Group Reporting	2-9	MAC Address	1-13
H		Network Interface Setup	1-13
Hierarchical Command Structure	2-3	Settings	1-13
I		Subnet Mask	1-13
IEEE-488 Interface Bus	1-2	Termination, GPIB	1-12
INITiate Subsystem	3-9	Transition Filter	2-7
Interface	1-6		

U to V

TRIGger Subsystem 3-72
Trigger System Programming 2-12
TSWEEP Command 3-74

U

Unit Suffixes 2-4

V

VISA 1-10
VXI-11 1-10

Anritsu



Anritsu utilizes recycled paper and environmentally conscious inks and toner.

Anritsu Company
490 Jarvis Drive
Morgan Hill, CA 95037-2809
USA