SCPI Command Reference

Agilent Technologies
N5181A/82A MXG Signal Generators

This guide applies to the following signal generator models:

N5181A MXG Analog Signal Generator

N5182A MXG Vector Signal Generator

Due to our continuing efforts to improve our products through firmware and hardware revisions, signal generator design and operation may vary from descriptions in this guide. We recommend that you use the latest revision of this guide to ensure you have up-to-date product information. Compare the print date of this guide (see bottom of page) with the latest revision, which can be downloaded from the following website:

http://www.agilent.com/find/mxg

Agilent Technologies

Manufacturing Part Number: N5180-90004

Printed in USA

September 2006

Notice

The material contained in this document is provided “as is”, and is subject to being changed, without notice, in future editions.

Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied with regard to this manual and to any of the Agilent products to which it pertains, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or any of the Agilent products to which it pertains. Should Agilent have a written contract with the User and should any of the contract terms conflict with these terms, the contract terms shall control.

Questions or Comments about our Documentation?

We welcome any questions or comments you may have about our documentation. Please send us an E-mail at **sources_manuals@am.exch.agilent.com**.

Contents

Contents

Contents

Contents

Contents

Documentation Overview

Installation Guide	• Safety Information • Receiving the Instrument • Environmental & Electrical Requirements • Basic Setup • Accessories • Operation Verification • Regulatory Information
User's Guide	• Instrument Overview • Front Panel Operation • Security • Basic Troubleshooting
Programming Guide	• Remote Operation • Status Registers • Creating & Downloading Files
SCPI Reference	• SCPI Basics • Command Descriptions • Programming Command Compatibility
Service Guide	• Troubleshooting • Assembly Replacement • Replaceable Parts • Post-Repair Procedures • Safety and Regulatory Information
Key Help[a]	• Key function description • Related SCPI commands

a. Press the **Help** hardkey, and then the key for which you wish help.

1 SCPI Basics

This chapter describes how SCPI information is organized and presented in this guide. An overview of the SCPI language is also provided. This chapter contains the following major sections:

Command Reference Information

SCPI Command Listings

The Table of Contents lists the Standard Commands for Programmable Instruments (SCPI) without the parameters. The SCPI subsystem name will generally have the first part of the command in parenthesis that is repeated in all commands within the subsystem. The title(s) beneath the subsystem name is the remaining command syntax. The following example demonstrates this listing:

Communication Subsystem (:SYSTem:COMMunicate)
:LAN:IP
:LAN:SUBNet

The following examples show the complete commands from the above Table of Contents listing:

:SYSTem:COMMunicate:LAN:IP
:SYSTem:COMMunicate:LAN:SUBNet

Key and Data Field Cross Reference

The index is set up so applicable key and data field names can be cross-referenced to the appropriate SCPI command. There are two headings in the index where the key and data field names can be found:

- individual softkey, hardkey, or data field name (i.e. To look up the communication subsystem topic on Default Gateway softkey refer to Default Gateway softkey.)
- subsystem name (i.e. To look for the Default Gateway softkey (in the Communication Subsystem), refer to the heading labeled: “communication subsystem keys”.)

Supported Field

Within each command section, the “Supported” heading describes which signal generator configurations are supported by the SCPI command. When “All Models” is shown next to this heading, all signal generator configurations are supported by the SCPI command. When “All with Option xxx” is shown next to this heading, only the stated option(s) is supported.

SCPI Basics

This section describes the general use of the SCPI language for the Agilent MXG. It is not intended to teach you everything about the SCPI language; the SCPI Consortium or IEEE can provide that level of detailed information. For a list of the specific commands available for the signal generator, refer to the table of contents.

For additional information, refer to the following publications:

- IEEE Standard 488.1-2003, IEEE Standard For Higher Performance Protocol for the Standard Digital Interface for Programmable Instrumentation. New York, NY, 2003.
- IEEE Standard 488.2-1992, IEEE Standard Codes, Formats, Protocols and Command Commands for Use with ANSI/IEEE Standard 488.1-1987. New York, NY, 1998.

Common Terms

The following terms are used throughout the remainder of this section:

Command	A command is an instruction in SCPI consisting of mnemonics (keywords), parameters (arguments), and punctuation. You combine commands to form messages that control instruments.
Controller	A controller is any device used to control the signal generator, for example a computer or another instrument.
Event Command	Some commands are events and cannot be queried. An event has no corresponding setting; it initiates an action at a particular time.
Program Message	A program message is a combination of one or more properly formatted commands. Program messages are sent by the controller to the signal generator.
Query	A query is a special type of command used to instruct the signal generator to make response data available to the controller. A query ends with a question mark. Generally you can query any command value that you set.
Response Message	A response message is a collection of data in specific SCPI formats sent from the signal generator to the controller. Response messages tell the controller about the internal state of the signal generator.

Command Syntax

A typical command is made up of keywords prefixed with colons (:). The keywords are followed by parameters. The following is an example syntax statement:

```
[:SOURce]:PULM:INTernal[1]:FREQuency <frequency>|MAXimum|MINimum|UP|DOWN
```

In the example above, the `:INTernal[1]` portion of the command immediately follows the `:PULM` portion with no separating space. The portion following the `:INTernal[1]`, `MAXimum|MINimum|UP|DOWN`, are the parameters (argument for the command statement). There is a separating space (white space) between the command and its parameter.

SCPI Basics

Additional conventions in syntax statements are shown in Table 1-1 and Table 1-2.

Table 1-1 Special Characters in Command Syntax

Characters	Meaning	Example
|	A vertical stroke between keywords or parameters indicates alterative choices. For parameters, the effect of the command varies depending on the choice.	`[:SOURce]:AM:MOD DEEP|NORMal` `DEEP` or `NORMal` are the choices.
[ ]	Square brackets indicate that the enclosed keywords or parameters are optional when composing the command. These implied keywords or parameters will be executed even if they are omitted.	`[:SOURce]:FREQuency[:CW]?` `SOURce` and `CW` are optional items.

< >	Angle brackets around a word (or words) indicate they are not to be used literally in the command. They represent the needed item.	`[:SOURce]:FREQuency:STARt <value><unit>`
		In this command, the words `<value>` and `<unit>` should be replaced by the actual frequency and unit.
		`:FREQuency:STARt 2.5GHz`

{ }	Braces indicate that parameters can optionally be used in the command once, several times, or not at all.	`[:SOURce]:LIST:POWer <value>{,<value>}`
		a single power listing:
		`LIST:POWer 5`
		a series of power listings:
		`LIST:POWer 5,10,15,20`

Table 1-2 Command Syntax

Characters, Keywords, and Syntax	Example

Upper-case lettering indicates the minimum set of characters required to execute the command. But, each mode of the command must be in either short form or the complete long form (no in between). Example:
Correct:
`:FREQ`
`:FREQuency`
Incorrect:
:FREQuenc
	`[:SOURce]:FREQuency[:CW]?`,
	`FREQ` is the minimum requirement.

Table 1-2 Command Syntax

Characters, Keywords, and Syntax	Example
Lower-case lettering indicates the portion of the command that is optional; it can either be included with the upper-case portion of the command or omitted. This is the flexible format principle called forgiving listening. Refer to "Command Parameters and Responses" on page 7 for more information.	`:FREQuency` Either :FREQ, `:FREQuency`, or :FREQUENCY is correct.
When a colon is placed between two command mnemonics, it moves the current path down one level in the command tree. Refer to "Command Tree" on page 6 more information on command paths.	`:TRIGger:OUTPut:POLarity?` `TRIGger` is the root level keyword for this command.
If a command requires more than one parameter, you must separate adjacent parameters using a comma. Parameters are not part of the command path, so commas do not affect the path level.	`[:SOURce]:LIST:` `DWELl <value>{,<value>}`
A semicolon separates two commands in the same program message without changing the current path.	`:FREQ 2.5GHz;:POW 10dBm`
White space characters, such as `<tab>` and `<space>`, are generally ignored as long as they do not occur within or between keywords. However, you must use white space to separate the command from the parameter, but this does not affect the current path.	`:FREQ uency` or `:POWer :LEVel` are not allowed. A `<space>` between `:LEVel` and `6.2` is mandatory. `:POWer:LEVel 6.2`

Command Types

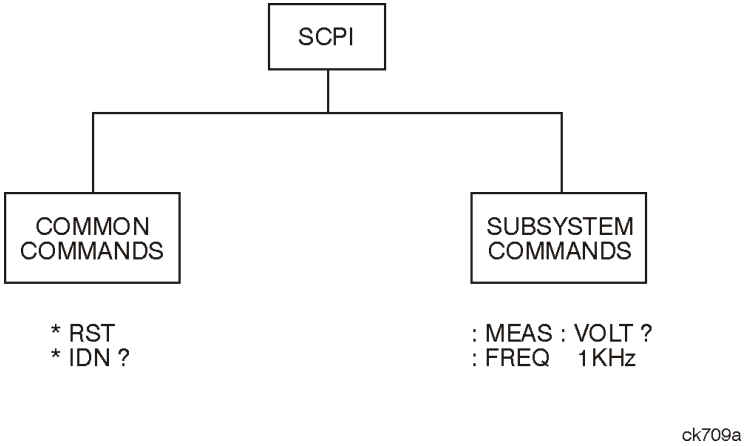
Commands can be separated into two groups: common commands and subsystem commands. Figure 1-1, shows the separation of the two command groups.

Common commands are used to manage status registers, synchronization, and data storage and are defined by IEEE 488.2. They are easy to recognize because they all begin with an asterisk. For example `*IDN?`, `*OPC`, and `*RST` are common commands. Common commands are not part of any subsystem and the signal generator interprets them in the same way, regardless of the current path setting.

Subsystem commands are distinguished by the colon (`:`). The colon is used at the beginning of a command statement and between keywords, as in `:FREQuency[:CW?]`. Each command subsystem is a set of commands that roughly correspond to a functional block inside the signal generator. For example, the power subsystem (`:POWer`) contains commands for power generation, while the status subsystem (:STATus) contains commands for controlling status registers.

SCPI Basics

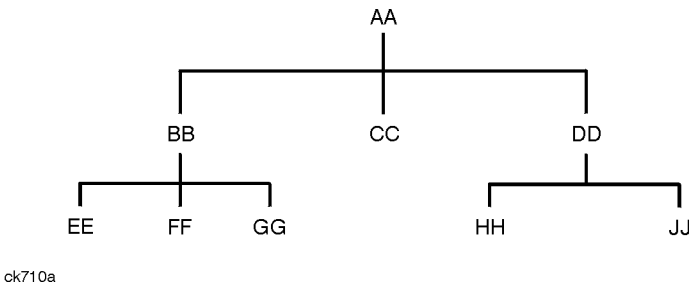
Figure 1-1 Command Types



Command Tree

Most programming tasks involve subsystem commands. SCPI uses a structure for subsystem commands similar to the file systems on most computers. In SCPI, this command structure is called a command tree and is shown in Figure 1-2.

Figure 1-2 Simplified Command Tree



The command closest to the top is the root command, or simply “the root.” Notice that you must follow a particular path to reach lower level commands. In the following example, `:POWer` represents AA, `:ALC` represents BB, `:SOURce` represents GG. The complete command path is `:POWer:ALC:SOURce?` (:AA:BB:GG).

Paths Through the Command Tree

To access commands from different paths in the command tree, you must understand how the signal generator interprets commands. The parser, a part of the signal generator firmware, decodes each message sent to the signal generator. The parser breaks up the message into component commands using a set of rules to determine the command tree path used. The parser keeps track of the current

path (the level in the command tree) and where it expects to find the next command statement. This is important because the same keyword may appear in different paths. The particular path is determined by the keyword(s) in the command statement.

A message terminator, such as a <new line> character, sets the current path to the root. Many programming languages have output statements that automatically send message terminators.

NOTE The current path is set to the root after the line-power is cycled or when *RST is sent.

Command Parameters and Responses

SCPI defines different data formats for use in program and response messages. It does this to accommodate the principle of forgiving listening and precise talking. For more information on program data types refer to IEEE 488.2.

Forgiving listening means the command and parameter formats are flexible.

For example, with the :FREQuency:REFerence:STATe ON|OFF|1|0 command, the signal generator accepts :FREQuency:REFerence:STATe ON, :FREQuency:REFerence:STATe 1, :FREQ:REF:STAT ON, :FREQ:REF:STAT 1 to turn on the frequency reference mode.

Each parameter type has one or more corresponding response data types. A setting that you program using a numeric parameter returns either real or integer response data when queried. Response data (data returned to the controller) is more concise and restricted, and is called precise talking.

Precise talking means that the response format for a particular query is always the same.

For example, if you query the power state (:POWer:ALC:STATe?) when it is on, the response is always 1, regardless of whether you previously sent :POWer:ALC:STATe 1 or :POWer:ALC:STATe ON. Table 1-3 shows the response for a given parameter type.

Table 1-3 Parameter and Response Types

Parameter Types	Response Data Types
Numeric	Real, Integer
Extended Numeric	Real, Integer
Discrete	Discrete
Boolean	Numeric Boolean
String	String
Definite Block	Arbitrary byte data[a]

a. (i.e. text, binary, discrete, real, integer, etc.-).

Numeric Parameters

Numeric parameters are used in both common and subsystem commands. They accept all commonly used decimal representations of numbers including optional signs, decimal points, and scientific notation.

If a signal generator setting is programmed with a numeric parameter which can only assume a finite value, it automatically rounds any entered parameter which is greater or less than the finite value. For example, if a signal generator has a programmable output impedance of 50 or 75 ohms, and you specified 76.1 for the output impedance, the value is rounded to 75. The following are examples of numeric parameters:

100	no decimal point required
100.	fractional digits optional
–1.23	leading signs allowed
4.56E<space>3	space allowed after the E in exponential
–7.89E–001	use either E or e in exponential
+256	leading + allowed
.5	digits left of decimal point optional

Extended Numeric Parameters

Most subsystems use extended numeric parameters to specify physical quantities. Extended numeric parameters accept all numeric parameter values and other special values as well.

The following are examples of extended numeric parameters:

100	any simple numeric value
1.2GHz	GHz can be used for exponential (E009)
200MHz	MHz can be used for exponential (E006)
–100mV	negative 100 millivolts
10DEG	10 degrees

Extended numeric parameters also include the following special parameters:

DEFault	resets the parameter to its default value
UP	increments the parameter
DOWN	decrements the parameter
MINimum	sets the parameter to the smallest possible value
MAXimum	sets the parameter to the largest possible value

Discrete Parameters

Discrete parameters use mnemonics to represent each valid setting. They have a long and a short form, just like command mnemonics. You can mix upper and lower case letters for discrete parameters.

The following examples of discrete parameters are used with the command `:TRIGger[:SEQuence]:SOURce BUS|IMMediate|EXTernal`.

BUS	GPIB, LAN, or USB triggering
IMMediate	immediate trigger (free run)
EXTernal	external triggering

Although discrete parameters look like command keywords, do not confuse the two. In particular, be sure to use colons and spaces properly. Use a colon to separate command mnemonics from each other and a space to separate parameters from command mnemonics.

The following are examples of discrete parameters in commands:

```
TRIGger:SOURce BUS

TRIGger:SOURce IMMediate

TRIGger:SOURce EXTernal
```

Boolean Parameters

Boolean parameters represent a single binary condition that is either true or false. The two-state boolean parameter has four arguments. The following list shows the arguments for the two-state boolean parameter:

ON	boolean true, upper/lower case allowed
OFF	boolean false, upper/lower case allowed
1	boolean true
0	boolean false

String Parameters

String parameters allow ASCII strings to be sent as parameters. Single or double quotes are used as delimiters.

The following are examples of string parameters:

```
'This is valid'

"This is also valid"

'SO IS THIS'
```

Real Response Data

Real response data represent decimal numbers in either fixed decimal or scientific notation. Most high-level programming languages that support signal generator input/output (I/O) handle either decimal or scientific notation transparently.

The following are examples of real response data:

+4.000000E+010, –9.990000E+002

–9.990000E+002

+4.0000000000000E+010

+1

0

Integer Response Data

Integer response data are decimal representations of integer values including optional signs. Most status register related queries return integer response data.

The following are examples of integer response data:

0	signs are optional
+100	leading + allowed
–100	leading – allowed
256	never any decimal point

Discrete Response Data

Discrete response data are similar to discrete parameters. The main difference is that discrete response data only returns the short form of a particular mnemonic, in all upper case letters.

The following are examples of discrete response data:

IMM

EXT

INT

NEG

Numeric Boolean Response Data

Boolean response data returns a binary numeric value of one or zero.

String Response Data

String response data are similar to string parameters. The main difference is that string response data returns double quotes, rather than single quotes. Embedded double quotes may be present in

string response data. Embedded quotes appear as two adjacent double quotes with no characters between them.

The following are examples of string response data:

```
"This is a string"

"one double quote inside brackets: [""]"

"Hello!"
```

Program Messages

The following commands will be used to demonstrate the creation of program messages:

`[:SOURce]:FREQuency:STARt`	`[:SOURce]:FREQuency:STOP`
`[:SOURce]:FREQuency[:CW]`	`[:SOURce]:POWer[:LEVel]:OFFSet`

Example 1

```
:FREQuency:STARt 500MHz;STOP 1000MHz
```

This program message is correct and will not cause errors; `STARt` and `STOP` are at the same path level. It is equivalent to sending the following message:

```
FREQuency:STARt 500MHz;FREQuency:STOP 1000MHz
```

Example 2

```
:POWer 10DBM;:OFFSet 5DB
```

This program message will result in an error. The message makes use of the default `POWer[:LEVel]` node (root command). When using a default node, there is no change to the current path position. Since there is no command `OFFSet` at the root level, an error results.

The following example shows the correct syntax for this program message:

```
:POWer 10DBM;:POWer:OFFSet 5DB
```

Example 3

```
:POWer:OFFSet 5DB;POWer 10DBM
```

This program message results in a command error. The path is dropped one level at each colon. The first half of the message drops the command path to the lower level command `OFFSet`; `POWer` does not exist at this level.

The `POWer 10DBM` command is missing the leading colon and when sent, it causes confusion because the signal generator cannot find `POWer` at the `POWer:OFFSet` level. By adding the leading colon, the current path is reset to the root. The following shows the correct program message:

```
:POWer:OFFSet 5DB;:POWer 10DBM
```

Example 4

```
FREQ 500MHz;POW 4DBM
```

In this example, the keyword short form is used. The program message is correct because it utilizes the default nodes of :FREQ[:CW] and :POW[:LEVel]. Since default nodes do not affect the current path, it is not necessary to use a leading colon before FREQ or POW.

File Name Variables

File name variables designate a data file and file path. File name variables are used in the SCPI command syntax whenever files are accessed. The name of the file is always required, but the file path can sometimes be optional or be designated using different formats. The following table shows these different file path formats:

Format	File Name Variable	Example
Format 1	"<file name>"	"Test_Data"
Format 2	"<file name@msus>"	"Test_Data@SEQ"[a]
Format 3	"<msus:file name>"	"SEQ:Test_Data"
Format 4	"</user/directory/file name>"	"/USER/SEQ/Test_Data"

a. Included for backwards compatibility. Not a recommended syntax.

Formats 2–4 offer programming flexibility and are equivalent. Format 1 can only be used with SCPI commands that imply the path name as part of the command syntax. Typically, SCPI load commands that access user-data files do not need to have a file path designated.

See Table 1-4 on page 14 for information on file types and directories.

NOTE The maximum length for a file name is 23 characters, excluding the file path.

Example Using Format 1

```
:CORR:FLAT:LOAD "FLAT_DATA"
```

The preceding example loads user-flatness data from a file called FLAT_DATA located in the USERFLAT directory. No file path is needed as the command syntax implies the directory where the file is located.

Example Using Format 2

```
:MEM:COPY "IQ_DATA@NVWFM","Test_DATA@WFM1"
```

The preceding example copies a file named IQ_DATA located in the WAVEFORM directory to a file named Test_DATA in volatile waveform memory (WFM1).

Example Using Format 3

```
:MEM:COPY "NVWFM:IQ_DATA","WFM1:Test_DATA"
```

The preceding example copies a file named IQ_DATA located in the WAVEFORM directory to a file named Test_DATA in volatile waveform memory (WFM1).

Example Using Format 4

```
:MEM:COPY "/USER/WAVEFORM/IQ_DATA","/USER/BBG1/WAVEFORM/IQ_DATA"
```

The preceding example copies a file named IQ_DATA located in the WAVEFORM directory to a file named IQ_DATA in volatile waveform memory (WFM1).

The following examples show commands, with different formats, that can be used to download a waveform file named Test_Data into the signal generator's volatile waveform memory (WFM1):

Command Syntax Format 3

```
:MEMory:DATA "WFM1:Test_Data",#ABC
```

Command Syntax Format 4

```
:MEMory:DATA "/USER/BBG1/WAVEFORM/Test_Data",#ABC
```

These commands are equivalent. The data block, #ABC, is described as follows:

#	This character indicates the beginning of the data block
A	Number of digits in the byte count B
B	Byte count in C
C	Waveform data

Refer to ":DATA" on page 81 and the Programming Guide for more information on data blocks and downloading waveform data.

File Types and Directory Structure

The signal generator uses a computer directory model structure for file storage. The top level directory is called the USER directory. All other directories are subdirectories located under the USER directory. Each subdirectory is dedicated to the type of data stored. For example, the BIN directory is used to store binary data whereas the MARKERS directory is used to store marker data.

NOTE When external memory is used, the files on the external memory are stored in a single directory (i.e. USER/). Each file has an extension (i.e. .waveform, .list, .markers, .state, etc.-). The SCPI commands use the paths shown in Table 1-4 on page 14 and the associated examples. But when viewed, the external memory, will not display these directories. Instead the file extensions will be displayed. For more information on the external memory capability refer to the *Programming Guide* and to the *Users Guide*.

The following table lists signal generator the subdirectories and file paths where file types are stored.

Table 1-4 File Types and Directory Structures

File System	File Type	File Path	MSUS Path
BINARY[a]	BIN	/USER/BIN	BINARY:
HDR1 - volatile arbitrary waveform header file[a]	HDR1	/USER/BBG1/HEADER	HDR1:
LIST - sweep list file	LIST	/USER/LIST	LIST:
MKR1 - volatile arbitrary waveform marker file[a]	MKR1	/USER/BBG1/MARKERS	MKR1:
NVHDR - non-volatile arbitrary waveform header file[a]	NVHDR	/USER/HEADER	NVHDR:
NVMKR - non-volatile arbitrary waveform marker file[a]	NVMKR	/USER/MARKERS	NVMKR:
NVWFM - non-volatile arbitrary waveform file[a]	NVWFM	/USER/WAVEFORM	NVWFM:
SEQ - ARB sequence file[a]	SEQ	/USER/SEQ	SEQ:
STATE	STAT	/USER/STATE	STATE:
USERFLAT - user-flatness file	UFLT	/USER/USERFLAT	USERFLAT:
WFM1 - volatile waveform file[a]	WFM1	/USER/BBG1/WAVEFORM	WFM1:

a. This feature does not apply to the N5181A.

MSUS (Mass Storage Unit Specifier) Variable

The variable "<msus>" enables a command to be file type specific when working with user files. Some commands use it as the only command parameter, while others can use it in conjunction with a file name when a command is not file type specific. When used with a file name, it is similar to Format 2 in the File Name Variables section on page 12. The difference is the file type specifier (msus) occupies its own variable and is not part of the file name syntax.

The following examples illustrate the usage of the variable "<msus>" when it is the only command parameter:

Command Syntax with the msus variable

```
:MMEMory:CATalog? "<msus>"
```

Command Syntax with the file system

```
:MMEMory:CATalog? "LIST:"
```

The variable "<msus>" is replaced with "LIST:". When the command is executed, the output displays only the files from the List file system.The following examples illustrate the usage of the variable "<file name>" with the variable "<msus>":

*Command Syntax with the file name and msus variable*s

```
:MMEMory:DELete[:NAME] "<file name>",["<msus>"]
```

Command Syntax with the file name and file system

```
:MMEMory:DELete:NAME "LIST_1","LIST:"
```

The command from the above example cannot discern which file system LIST_1 belongs to without a file system specifier and will not work without it. When the command is properly executed, LIST_1 is deleted from the List file system.

The following example shows the same command, but using Format 2 from the File Name Variables section on page 12:

```
:MMEMory:DELete:NAME "LIST_1@LIST"
```

When a file name is a parameter for a command that is not file system specific, either format ("<file name>","<msus>" or "<file name@msus>") will work.

Refer to Table 1-4 on page 14 for a listing of the file systems and types.

Quote Usage with SCPI Commands

As a general rule, programming languages require that SCPI commands be enclosed in double quotes as shown in the following example:

```
":FM:EXTernal:IMPedance 600"
```

However when a string is the parameter for a SCPI command, additional quotes or other delimiters may be required to identify the string. Your programming language may use two sets of double

quotes, one set of single quotes, or back slashes with quotes to signify the string parameter. The following examples illustrate these different formats:

`"MEMory:LOAD:LIST ""myfile"""` used in BASIC programming languages

`"MEMory:LOAD:LIST \"myfile\""` used in C, C++, Java, and PERL

`"MEMory:LOAD:LIST 'myfile'"` accepted by most programming languages

Consult your programming language reference manual to determine the correct format.

Binary, Decimal, Hexadecimal, and Octal Formats

Command values may be entered using a binary, decimal, hexadecimal, or octal format. When the binary, hexadecimal, or octal format is used, their values must be preceded with the proper identifier. The decimal format (default format) requires no identifier and the signal generator assumes this format when a numeric value is entered without one. The following list shows the identifiers for the formats that require them:

- #B identifies the number as a binary numeric value (base-2).
- #H identifies the number as a hexadecimal alphanumeric value (base-16).
- #Q identifies the number as a octal alphanumeric value (base-8).

The following are examples of SCPI command values and identifiers for the decimal value 45:

#B101101	binary equivalent
#H2D	hexadecimal equivalent
#Q55	octal equivalent

The following example sets the RF output power to 10 dBm (or the equivalent value for the currently selected power unit, such as DBUV or DBUVEMF) using the hexadecimal value 000A:

```
:POW #H000A
```

A unit of measure, such as dBm or mV, will not work with the values when using a format other than decimal.

SCPI Basics

2 Basic Function Commands

This chapter provides SCPI descriptions for subsystems dedicated to signal generator operations common to most Agilent MXG Signal Generators. This chapter contains the following major sections:

Correction Subsystem ([:SOURce]:CORRection)

:FLATness:INITialize:FSTep

Supported All Models

CAUTION The current flatness data will be overwritten once this command is executed. If needed, save the current data. Refer to ":FLATness:STORe" on page 23 for storing user flatness files.

```
[:SOURce]:CORRection:FLATness:INITialize:FSTep
```

This command replaces the loaded user flatness data with the settings from the current step array data points.

You can load only one user flatness file at a time.

The maximum number of user flatness points is 1,601. When copying the step array settings over to a user flatness file, ensure that the number of points in the step array do not exceed the maximum user flatness points.

Key Entry **Load Cal Array From Step Array**

:FLATness:LOAD

Supported All Models

```
[:SOURce]:CORRection:FLATness:LOAD "<file name>"
```

This command loads a user-flatness correction file. The "<file name>" variable is the name of the file located in the Catalog of USERFLAT Files. The directory path is implied in the command and need not be specified in the variable name. For more information on file name syntax, refer to "File Name Variables" on page 12.

Key Entry **Load From Selected File**

:FLATness:PAIR

Supported All Models

```
[:SOURce]:CORRection:FLATness:PAIR <freq.>[<freq suffix>],
<corr.>[<corr suffix>]
```

This command sets a frequency and amplitude correction pair.

The maximum number of points that can be entered is 1601. Option 501 is specific to the N5181A.

<corr.> This variable is the power correction.

Range	Frequency	Standard	Option 1EQ[a]
	Option 501[b]: 100kHz–1GHz	–110 to 18 dB	<–144 to 18 dB
	Option 503: 100kHz–3GHz	–110 to 18 dB	<–144 to 16 dB
	Option 506: 100kHz–6GHz	–110 to 18 dB	<–144 to 4 dB

a. Settable, but specified to –127 dBm with Option 1EQ
b. Option 501 is only available on the N5181A.

Key Entry **Configure Cal Array**

:FLATness:POINts

Supported All Models

```
[:SOURce]:CORRection:FLATness:POINts?
```

This query returns the number of points in the user-flatness correction file.

:FLATness:PRESet

Supported All Models

CAUTION The current correction data will be overwritten once this command is executed. Save the current data if needed. Refer to ":FLATness:STORe" on page 23 for storing user-flatness files.

```
[:SOURce]:CORRection:FLATness:PRESet
```

This command presets the user-flatness correction to a factory-defined setting that consists of one point.

Key Entry **Preset List**

:FLATness:STEP:POINts

Supported All Models

```
[:SOURce]:CORRection:FLATness:STEP:POINts <points> |MAXimum|MINimum|DEFault|
[:SOURce]:CORRection:FLATness:STEP:POINts?[MAXimum|MINimum]
```

This command is used to define the number of points in the user flatness calibration step array. See also, ":FLATness:STEP:STARt" on page 22 and ":FLATness:STEP:STOP" on page 22.

Key Entry **# Points 2**

:FLATness:STEP:STARt

Supported All Models

```
[:SOURce]:CORRection:FLATness:STEP:STARt <freq><unit> |MAXimum|MINimum|DEFault|
[:SOURce]:CORRection:FLATness:STEP:STARt? [MAXimum|MINimum]
```

This command sets the start frequency for the user flatness calibration step array. See also, ":FLATness:STEP:POINts" on page 22 and ":FLATness:STEP:STOP" on page 22.

***RST** Option 501: +1.0000000000000E+09

Option 503: +3.0000000000000E+09

Option 506: +6.0000000000000E+09

Range Option 501: 100kHz–1GHz

Option 503: 100kHz–3GHz

Option 506: 100kHz–6GHz

Key Entry **Freq Start**

Remarks Options 501 is specific to the N5181A.

:FLATness:STEP:STOP

Supported All Models

```
[:SOURce]:CORRection:FLATness:STEP:STOP <freq><unit> |MAXimum|MINimum|DEFault|
[:SOURce]:CORRection:FLATness:STEP:STOP? [MAXimum|MINimum]
```

This command sets the stop frequency for the user flatness calibration step array. See also, ":FLATness:STEP:POINts" on page 22 and ":FLATness:STEP:STARt" on page 22.

***RST** Option 501: +1.0000000000000E+09

Option 503: +3.0000000000000E+09

Option 506: +6.0000000000000E+09

Range Option 501: 100kHz–1GHz

Option 503: 100kHz–3GHz

Option 506: 100kHz–6GHz

Key Entry **Freq Stop**

Remarks Options 501 is specific to the N5181A.

:FLATness:STORe

Supported All Models

```
[:SOURce]:CORRection:FLATness:STORe "<file name>"
```

This command stores the current user-flatness correction data to a file named by the :CORRection:FLATness:STORe command. The directory path is implied in the command and need not be specified in the "<file name>" variable.

Key Entry **Store To File**

Remarks For information on file name syntax, refer to "File Name Variables" on page 12.

[:STATe]

Supported All Models

```
[:SOURce]:CORRection[:STATe] ON|OFF|1|0
[:SOURce]:CORRection[:STATe]?
```

This command enables or disables the user-flatness corrections.

***RST** 0

Key Entry **Flatness Off On**

Digital Modulation Subsystem—N5182A ([:SOURce])

:BURSt:STATe

Supported N5182A

```
[:SOURce]:BURSt:STATe ON|OFF|1|0
[:SOURce]:BURSt:STATe?
```

This command enables or disables the burst envelope function.

***RST** 0

Key Entry **Burst Envelope Off On**

:DM:CORRection:OPTimization

Supported N5182A

```
[:SOURce]:DM:CORRection:OPTimization RFOut|EXTernal
[:SOURce]:DM:CORRection:OPTimization?
```

This command enables the internal optimized path to accommodate I/Q signals.

EXT — This choice enables the Baseband Generator to external I/Q output path, applies correction terms, and provides a calibrated signal at the IQ output. When the I/Q Output is selected, the RF signals at the RF Output are uncalibrated.

RFO — This choice enables the Baseband Generator to RF output path, applies correction terms, and provides a calibrated signal at the RF output. When the RF Output is selected, the I/Q signals at the I/Q Output are uncalibrated.

***RST** RFO

Key Entry **I/Q Correction Optimized Path**

:DM:EXTernal:ALC:BANDwidth|BWIDth

Supported N5182A

NOTE Refer to the *Programming Compatibility Guide* for information on this command. This command was replaced by the ":ALC:BANDwidth|BWIDth" command shown in Chapter 6.

:DM:EXTernal:POLarity

Supported N5182A

```
[:SOURce]:DM:EXTernal:POLarity NORMal|INVert
[:SOURce]:DM:EXTernal:POLarity?
```

This command sets the phase polarity for the I/Q signal.

NORMal This choice selects normal phase polarity for the I and Q signals.

INVert This choice flips the I and Q signals by routing the I signal to the Q input of the I/Q modulator and the Q signal to the I input.

***RST** NORM

Key Entry **Int Phase Polarity Normal Invert**

:DM:IQADjustment:DELay

Supported N5182A

```
[:SOURce]:DM:IQADjustment:DELay <value><unit>
[:SOURce]:DM:IQADjustment:DELay?
```

This command enables you to change the absolute phase of both I and Q with respect to triggers and markers. A positive value delays I and Q. This value affects both the external I/Q out signals and the baseband signal modulated on the RF output. This adjustment cannot be used with constant envelope modulation and does not affect external I/Q inputs.

The variable <value> is expressed in seconds.

***RST** +0.00000000E+000

Range –400 νσ το 400 νσ

Key Entry **I/Q Delay**

Remarks This command is effective only if the state of the I/Q adjustment function is set to ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

:DM:IQADjustment:EXTernal:COFFset

Supported N5182A

```
[:SOURce]:DM:IQADjustment:EXTernal:COFFset <value>
[:SOURce]:DM:IQADjustment:EXTernal:COFFset?
```

This command sets the common mode offset voltage for both the in-phase (I) and quadrature-phase (Q) signals going to the rear panel I and Q output connectors.

The variable <value> is expressed in units of volts (mV–V).

***RST** +0.00000000E+000

Range ∠2.5 to 2.5V

Key Entry **Common Mode I/Q Offset**

Remarks This command is effective only if the state of the I/Q adjustment function is set to

ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

:DM:IQADjustment:EXTernal:DIOFfset

Supported N5182A

```
[:SOURce]:DM:IQADjustment:EXTernal:DIOFfset <value>
[:SOURce]:DM:IQADjustment:EXTernal:DIOFfset?
```

This command sets the differential offset voltage for an in-phase (I) signal routed to the I output connectors.

The variable <value> is expressed in units of volts (mV–V).

***RST** +0.00000000E+000

Range –25mV to 25mV

Key Entry **Diff. Mode I Offset**

Remarks This command is effective only if the state of the I/Q adjustment function is set to ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

:DM:IQADjustment:EXTernal:DQOFfset

Supported N5182A

```
[:SOURce]:DM:IQADjustment:EXTernal:DQOFfset <value>
[:SOURce]:DM:IQADjustment:EXTernal:DQOFfset?
```

This command sets the differential offset voltage for a quadrature-phase (Q) signal routed to the Q output connectors.

***RST** +0.00000000E+000

Range –25mV to 25mV

Key Entry **Diff. Mode Q Offset**

Remarks This command is effective only if the state of the I/Q adjustment function is set to ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

:DM:IQADjustment:EXTernal:IOFFset

Supported E4438C

```
[:SOURce]:DM:IQADjustment:EXTernal:IOFFset <value>
[:SOURce]:DM:IQADjustment:EXTernal:IOFFset?
```

This command sets the offset voltage for a signal applied to the 600 ohm I input connector.

The variable <value> is expressed in units of volts (mV–V).

***RST** +0.00000000E+000

Key Entry **External Input I Offset**

Range –100 mV to 100 mV

Remarks This command is effective only if the state of the I/Q adjustment function is set to ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

:DM:IQADjustment:IOFFset

Supported N5182A

```
[:SOURce]:DM:IQADjustment:IOFFset <value><unit>
```

```
[:SOURce]:DM:IQADjustment:IOFFset?
```

This command adjusts the I channel offset value.

When using this command to minimize the LO feedthrough signal, optimum performance is achieved when the command is sent after all other I/Q path commands are executed, such as those that change the internal phase polarity or adjust the modulator attenuator. If other adjustments are made after minimizing is performed, the LO feedthrough signal may increase.

The variable <value> is expressed in units of percent with a minimum resolution of 0.025.

***RST** +0.00000000E+000

Range –20.000 to 20.000

Key Entry **I Offset**

Remarks This command is effective only if the state of the I/Q adjustment function is set to ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

:DM:IQADjustment:QOFFset

Supported N5182A

```
[:SOURce]:DM:IQADjustment:QOFFset
[:SOURce]:DM:IQADjustment:QOFFset?
```

This command adjusts the Q channel offset value.

When using this command to minimize the LO feedthrough signal, optimum performance is achieved when the command is sent after all other I/Q path commands are executed, such as those that change the internal phase polarity or adjust the modulator attenuator. If other adjustments are made after minimizing is performed, the LO feedthrough signal may increase.

The variable <value> is expressed in units of percent with a minimum resolution of 0.025.

***RST** +0.00000000E+000

Range –20.000 to 20.000

Key Entry **Q Offset**

Remarks This command is effective only if the state of the I/Q adjustment function is set to ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

:DM:IQADjustment:EXTernal:QOFFset

Supported E4438C

```
[:SOURce]:DM:IQADjustment:EXTernal:QOFFset <value>
[:SOURce]:DM:IQADjustment:EXTernal:QOFFset?
```

This command sets the offset voltage for a signal applied to the 600 ohm Q input connector.

The variable <value> is expressed in units of volts (mV–V).

***RST** +0.00000000E+000

Range –100 mV to 100 mV

Key Entry **External Input Q Offset**

Remarks This command is effective only if the state of the I/Q adjustment function is set to ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

:DM:IQADjustment:EXTernal:QSKew

Supported N5182A

```
[:SOURce]:DM:IQADjustment:EXTernal:QSKew <value>
[:SOURce]:DM:IQADjustment:EXTernal:QSKew?
```

CAUTION This Q phase angle adjustment is uncalibrated.

This command adjusts the phase angle (quadrature skew) between the I and Q vectors by increasing or decreasing the Q phase angle. This command adjusts the signals externally input to the signal generator's front panel Q input connector. For more information on this connector, refer to the User's Guide.

The <value> variable is expressed in degrees with a minimum resolution of 0.1.

If the signal generator is operating at frequencies greater than 3.3 GHz, quadrature skew settings greater than ±5 degrees will not be within specifications.

Positive skew increases the angle from 90 degrees while negative skew decreases the angle from 90 degrees. When the quadrature skew is zero, the phase angle between the I and Q vectors is 90 degrees.

This command is effective only if the state of the I/Q adjustment function is set to ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

Example

```
:DM:IQAD:EXT:QSK 4.5
```

The preceding example increases the phase angle by 4.5 degrees.

***RST** +0.00000000E+000

Range –200 to +200

Key Entry **Quadrature Angle Adjustment**

:DM:IQADjustment:GAIN[1]

Supported N5182A

```
[:SOURce]:DM:IQADjustment:GAIN[1] <value><unit>
[:SOURce]:DM:IQADjustment:GAIN[1]?
```

This command sets the gain for the I signal relative to the Q signal.

The variable <value> is expressed in units of decibels (dB).

***RST** +0.00000000E+000

Range –1 to 1

Key Entry **I/Q Gain Balance**

Remarks This command is effective only if the state of the I/Q adjustment function is set to ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

:DM:IQADjustment:QSKew

Supported N5182A

```
[:SOURce]:DM:IQADjustment:QSKew <value>
[:SOURce]:DM:IQADjustment:QSKew?
```

This command adjusts the phase angle (quadrature skew) between the I and Q vectors by increasing or decreasing the Q phase angle.

The <value> variable is expressed in degrees with a minimum resolution of 0.1.

If the signal generator is operating at frequencies greater than 3.3 GHz, quadrature skew settings greater than ±5 degrees will not be within specifications.

Positive skew increases the angle from 90 degrees while negative skew decreases the angle from 90 degrees. When the quadrature skew is zero, the phase angle between the I and Q vectors is 90 degrees.

This command is effective only if the state of the I/Q adjustment function is set to ON. Refer to ":DM:IQADjustment[:STATe]" on page 30.

Example

```
:DM:IQAD:QSK 4.5
```

The preceding example increases the phase angle by 4.5 degrees.

***RST** +0.00000000E+000

Range –1E1 to +1E1

Key Entry **Quadrature Angle Adjustment**

:DM:IQADjustment:SKEW

Supported N5182A

```
[:SOURce]:DM:IQADjustment:SKEW <value>
[:SOURce]:DM:IQADjustment:SKEW?
```

This command changes the I/Q skew which is a time delay difference between the I and Q signals. Equal and opposite skew is applied to both I and Q and affects the RF Output and I/Q output paths simultaneously. A positive value delays the I signal relative to the Q signal, and a negative value delays the Q signal relative to the I signal.

Example

```
:DM:IQAD:SKEW 5E-9
```

The preceding example sets the time delay difference between the I and Q signals to 5 nanoseconds.

***RST** +0.00000000E+000

Range –800ns to +800ns

Key Entry **I/Q Skew**

:DM:IQADjustment[:STATe]

Supported N5182A

```
[:SOURce]:DM:IQADjustment[:STATe] ON|OFF|1|0
[:SOURce]:DM:IQADjustment[:STATe]?
```

This command enables or disables the I/Q adjustments.

Example

```
:DM:IQAD 1
```

The preceding example enables I/Q adjustments.

***RST** 0

Key Entry **I/Q Adjustments Off On**

:DM:POLarity[:ALL]

Supported N5182A

```
[:SOURce]:DM:POLarity[:ALL] NORMal|INVert
[:SOURce]:DM:POLarity?
```

This command sets the digital modulation phase polarity.

This softkey is found under the I/Q menu.

NORMal This choice selects normal phase polarity for the I and Q signals.

INVert This choice inverts the Q channel signal.

***RST** NORM

Key Entry **Int Phase Polarity Normal Invert**

:DM:SOURce

Supported N5182A

```
[:SOURce]:DM:SOURce EXTernal|INTernal|SUM
[:SOURce]:DM:SOURce?
```

This command selects the I/Q modulator source.

This softkey is found under the I/Q menu.

EXTernal — This choice selects a 50 ohm impedance for the I and Q input connectors and routes the applied signals to the I/Q modulator.

INTernal — This choice selects the internal baseband generator as the source for the I/Q modulator and requires Option 651/652/654.

Sum — This choice selects the internal baseband generator and combines that signal with an external source and routes the applied signals to the I/Q modulator and requires Option 651/652/654.

***RST** INT

Key Entry **External** **Internal** **Sum**

:DM:STATe

Supported N5182A

```
[:SOURce]:DM:STATe ON|OFF|1|0
[:SOURce]:DM:STATe?
```

This command enables or disables the I/Q modulator.

The I/Q modulator is enabled whenever a digital format is turned on.

The I/Q annunciator will be shown on the signal generator display whenever the I/Q modulator is on.

ON (1) — This choice enables the internal I/Q modulator.

OFF (0) — This choice disables the internal I/Q modulator. You can turn off the I/Q modulation with this choice even though a digital modulation format is enabled. With this configuration, the RF output signal will not be modulated, but the I/Q signals may be present at the rear panel I and Q outputs depending on the rear panel output selection.

***RST** 0

Key Entry **I/Q Off On**

Frequency Subsystem ([:SOURce])

:FREQuency:CENTer

Supported All Models

```
[:SOURce]:FREQuency:CENTer <num>[<freq_suffix>]|UP|DOWN
[:SOURce]:FREQuency:CENTer? [MAXimum|MINimum]
```

This command sets the center frequency for a step sweep. The center frequency symmetrically divides the selected frequency span and is coupled to the start and stop frequency settings. The frequency range and reset values are dependent on the signal generator model and option number.

The query returns the start and stop frequencies if the optional MAXimum or MINimum are used.

*RST Option 501[a]:+1.0000000000000E+9
Option 503: +3.0000000000000E+9
Option 506: +6.0000000000000E+9

a. Option 501 is available only on the N5181A.

Range[a] Option 501: 250 kHz–1 GHz
Option 503: 250 kHz–3 GHz
Option 506: 250 kHz–6 GHz

a. Settable, but not specified to 100 Hz.

Example

```
:FREQ:CENT .5 GHz
```

The preceding example sets the center frequency for a sweep to .5 GHz.

Key Entry **Freq Center**

:FREQuency:CHANnels:BAND

Supported All Models

```
[:SOURce]:FREQuency:CHANnels:BAND NBASe|NMOBile|BPGSm|MPGSm|BEGSm|MEGSm|
BRGSm|MRGSm|BDCS|MDCS|BPCS|MPCS|B450|GM450|B480|M480|B850|M850|B8|M8|B15|M15|B390|B42
0|B460|B915|M380|M410|M450|M870|PHS|DECT
[:SOURce]:FREQuency:CHANnels:BAND?
```

This command sets the frequency of the signal generator by specifying a frequency channel band. The frequency channel state must be enabled for this command to work. See ":FREQuency:CHANnels[:STATe]" on page 35.

Table 2-1 Frequency Channel Bands

SCPI Parameter	Frequency Channel Band Selected	Standard
NBASe	Standard Base	NADC
NMOBile	Standard Mobile	NADC
BPGSm	P-Gsm 900 Base	GSM
MPGSm	P-Gsm 900 Mobile	GSM
BEGSm	E-Gsm 900 Base	GSM

Table 2-1 Frequency Channel Bands

SCPI Parameter	Frequency Channel Band Selected	Standard
MEGSm	E-Gsm 900 Mobile	GSM
BRGSm	R-Gsm 900 Base	GSM
MRGSm	R-Gsm 900 Mobile	GSM
BDCS	DCS 1800 Base	GSM
MDCS	DCS 1800 Mobile	GSM
BPCS	PCS 1900 Base	GSM
MPCS	PCS 1900 Mobile	GSM
B450	Gsm 450 Base	GSM
GM450	Gsm 450 Mobile	GSM
B480	Gsm 480 Base	GSM
M480	Gsm 480 Mobile	GSM
B850	Gsm 850 Base	GSM
M850	Gsm 850 Mobile	GSM
B8	800MHz Base	PDC
M8	800MHz Mobile	PDC
B15	1500MHz Base	PDC
M15	1500MHz Mobile	PDC
B390	Base 390-400	TETRA
B420	Base 420-430	TETRA
B460	Base 460-470	TETRA
B915	Base 915-921	TETRA
M380	Mobile 380-390	TETRA
M410	Mobile 410-420	TETRA
M450	Mobile 450-460	TETRA
M870	Mobile 870-876	TETRA
PHS	Standard PHS	PHS
DECT	Standard DECT	DECT

Example

```
:FREQ:CHAN:BAND DECT
```

The preceding example sets the frequency band to standard DECT.

***RST** BPGS

Key Entry

P-GSM Base PCS Base	E-GSM Base GSM 450 Base	R-GSM Base GSM 480 Base	DCS Base GSM 850 Base
NADC Base	800MHz Base	1500MHz Base	
Tetra Base 390 / 400	Tetra Base 420 / 430	Tetra Base 460 / 470	
Tetra Base 915 / 921	PHS Standard	DECT Standard	
P-GSM Mobile	E-GSM Mobile	R-GSM Mobile	DCS Mobile

PCS Mobile	**GSM 450 Mobile**	**GSM 480 Mobile**	**GSM 850 Mobile**
NADC Mobile	**800MHz Mobile**	**1500MHz Mobile**	
Tetra Mobile 380 / 390	**Tetra Mobile 410 / 420**	**Tetra Mobile 450 / 460**	
Tetra Mobile 870 / 876			

:FREQuency:CHANnels:NUMBer

Supported All Models

```
[:SOURce]:FREQuency:CHANnels:NUMBer <number>
[:SOURce]:FREQuency:CHANnels:NUMBer?
```

This command sets the frequency of the signal generator by specifying a channel number of a given frequency band.

The channel band and channel state must be enabled for this command to work. Refer to ":FREQuency:CHANnels[:STATe]" on page 35.

Example

```
:FREQ:CHAN:NUMB 24
```

The preceding example sets the channel number to 24 for the current band.

***RST** +1

Range	P-GSM Base/Mobile:	1–24
	E-GSM and R-GSM Base/Mobile:	1–1023
	DCS Base/Mobile:	512–885
	PCS Base/Mobile:	512–900
	GSM-450 Base/Mobile:	259–293
	GSM-480 Base/Mobile:	306–340
	GSM-850 Base/Mobile:	128–251
	NADC Base/Mobile:	1–1023
	800MHz Base/Mobile:	0–640
	1500MHz Base/Mobile:	0–960
	TETRA 380/390 Mobile:	3600–4000
	TETRA 390/4000 Base:	3600–4000
	TETRA 410/420 Mobile:	800–1200
	TETRA 420/430 Base:	800–1200
	TETRA 460/470: 2400 through 2800	2400–2800
	TETRA 870/876 Mobile:	600–640
	TETRA 915/921 Base:	600–940

PHS Standard:	1–255
DECT Standard:	0–9

Key Entry **Channel Number**

:FREQuency:CHANnels[:STATe]

Supported All Models

```
[:SOURce]:FREQuency:CHANnels[:STATe] ON|OFF|1|0
[:SOURce]:FREQuency:CHANnels[:STATe]?
```

This command enables or disables the frequency channel and band selection. The signal generator frequency will be set to the channel frequency when the state is on. To set frequency channel bands refer to ":FREQuency:CHANnels:BAND" on page 32.

Example

```
:FREQ:CHAN ON
```

The preceding example turns on the frequency channel.

***RST** 0

Key Entry **Freq Channels Off On**

:FREQuency:FIXed

Supported All Models

```
[:SOURce]:FREQuency:FIXed <value><unit>
[:SOURce]:FREQuency:FIXed?
```

This command sets the signal generator output frequency.

***RST**	Option 501:+1.0000000000000E+09
	Option 503: +3.0000000000000E+09
	Option 506: +6.0000000000000E+09
Range	Option 501: 100kHz–1GHz
	Option 503: 100kHz–3GHz
	Option 506: 100kHz–6GHz

Remarks A frequency change may affect the current output power. Refer to "[:LEVel][:IMMediate][:AMPLitude]" on page 57 for the correct specified frequency and amplitude settings. To set the frequency mode refer to ":FREQuency:MODE" on page 36. Option 501 is specific to the N5181A.

:FREQuency:MODE

Supported All Models

```
[:SOURce]:FREQuency:MODE CW|FIXed|LIST
[:SOURce]:FREQuency:MODE?
```

This command sets the frequency mode of the signal generator to CW or swept.

CW and FIXed — These choices are synonymous with one another and stops a frequency sweep, allowing the Agilent MXG to operate at a set frequency. Refer to ":FREQuency[:CW]" on page 39 for setting the frequency in the CW mode and to ":FREQuency:FIXed" on page 35 for setting the frequency in the FIXed mode.

LIST — This choice selects the swept frequency mode. If sweep triggering is set to immediate along with continuous sweep mode, executing the command starts the LIST or STEP frequency sweep.

NOTE To perform a frequency and amplitude sweep, you must also select LIST as the power mode. See ":MODE" on page 54 for selecting the list mode for an amplitude sweep.

***RST** CW

Key Entry **Freq** **Freq Off**

:FREQuency:MULTiplier

Supported All Models

```
[:SOURce]:FREQuency:MULTiplier <value>
[:SOURce]:FREQuency:MULTiplier?
```

This command sets the multiplier for the signal generator carrier frequency. This displayed frequency equals the actual frequency times the multiplier.

***RST** +1.00000000E+000

Range Negative Values: –1000 to –.001 Positive Values: .001–1000

Key Entry **Freq Multiplier**

Remarks For any multiplier other than one, the MULT indicator is shown in the frequency area of the display.

:FREQuency:OFFSet

Supported All Models

```
[:SOURce]:FREQuency:OFFSet <value><unit>
[:SOURce]:FREQuency:OFFSet?
```

This command sets the frequency offset.

The query of this command returns a value equal to the original output frequency times the multiplier value, plus the frequency offset value. This displayed frequency equals the actual frequency times the multiplier.

When an offset has been entered, the OFFS indicator is turned on in the frequency area of the display.

The frequency offset state is turned on when any non-zero value is entered; entering zero will turn it off. Refer to :FREQuency:OFFSet:STATe for setting the offset state independent of entering offset values.

***RST** +0.0000000000000E+00

Range –200GHz to 200GHz

Key Entry **Freq Offset**

:FREQuency:OFFSet:STATe

Supported All Models

```
[:SOURce]:FREQuency:OFFSet:STATe ON|OFF|1|0
[:SOURce]:FREQuency:OFFSet:STATe?
```

This command enables or disables the offset frequency.

***RST** 0

Key Entry **Freq Offset**

Remarks Entering OFF (0) will set the frequency offset to 0 Hz.

:FREQuency:REFerence

Supported All Models

```
[:SOURce]:FREQuency:REFerence <value><unit>
[:SOURce]:FREQuency:REFerence?
```

This command sets the output reference frequency.

***RST** +0.0000000000000E+00

Range Option 501: 0Hz–1GHz

Option 503: 0Hz–3GHz

Option 506: 0Hz–6GHz

Key Entry **Freq Ref Set**

Remarks Option 501 is specific to the N5181A.

:FREQuency:REFerence:SET

Supported All Models

```
[:SOURce]:FREQuency:REFerence:Set
```

This command sets the current CW output frequency, along with any offset, as a 0 hertz reference value.

***RST** +0.0000000000000E+00

Key Entry **Freq Ref Set**

:FREQuency:REFerence:STATe

Supported All Models

```
[:SOURce]:FREQuency:REFerence:STATe ON|OFF|1|0
[:SOURce]:FREQuency:REFerence:STATe?
```

This command enables or disables the frequency reference mode.

When the frequency reference mode is on, subsequent frequency parameters are set relative to the reference value.

***RST** 0

Key Entry **Freq Ref Off On**

:FREQuency:SPAN

Supported All Models

```
[:SOURce]:FREQuency:SPAN <num>[<freq_suffix>]|UP|DOWN
[:SOURce]:FREQuency:SPAN? [MAXimum|MINimum]
```

This command sets the length of the frequency range for a step sweep. Span setting is symmetrically divided by the selected center frequency and is coupled to the start and stop frequency settings. The span range is dependent on the signal generator model and option number.

Example

```
:FREQ:SPAN 100MHz
```

The preceding example sets the frequency span to 100 megahertz.

***RST** +0.0000000000000E+00

Key Entry **Freq Span**

:FREQuency:STARt

Supported All Models

```
[:SOURce]:FREQuency:STARt <value><unit>
[:SOURce]:FREQuency:STARt?
```

This command sets the first frequency point in a step sweep.

***RST** Option 501: +1.0000000000000E+09

Option 503: +3.0000000000000E+09

Option 506: +6.0000000000000E+09

Range Option 501: 100kHz–1GHz

Option 503: 100kHz–3GHz

Option 506: 100kHz–6GHz

Key Entry **Freq Start**

Remarks Option 501 is specific to the N5181A.

:FREQuency:STOP

Supported All Models

```
[:SOURce]:FREQuency:STOP <value><unit>
[:SOURce]:FREQuency:STOP?
```

This command sets the last frequency point in a step sweep.

***RST** Option 501: +1.0000000000000E+09

Option 503: +3.0000000000000E+09

Option 506: +6.0000000000000E+09

Range Option 501: 100kHz–1GHz

Option 503: 100kHz–3GHz

Option 506: 100kHz–6GHz

Key Entry **Freq Stop**

Remarks Option 501 is specific to the N5181A.

:FREQuency[:CW]

Supported All Models

```
[:SOURce]:FREQuency[:CW] <value><unit>
[:SOURce]:FREQuency[:CW]?
```

This command sets the signal generator output frequency for the CW frequency mode.

***RST** Option 501: +1.0000000000000E+09

Option 503: +3.0000000000000E+09

Option 506: +6.0000000000000E+09

Range Option 501: 100kHz–1GHz

Option 503: 100kHz–3GHz

Option 506: 100kHz–6GHz

Key Entry **Freq**

Remarks To set the frequency mode to CW, refer to ":FREQuency:MODE" on page 36.

Option 501 is specific to the N5181A.

:PHASe:REFerence

Supported All Models

```
[:SOURce]:PHASe:REFerence
```

This command sets the current output phase as a zero reference.

Subsequent phase adjustments are set relative to the new reference.

Key Entry **Phase Ref Set**

:PHASe[:ADJust]

Supported All Models

```
[:SOURce]:PHASe[:ADJust] <value><unit>
[:SOURce]:PHASe[:ADJust]?
```

This command adjusts the phase of the modulating signal.

The query will only return values in radians.

***RST** +0.00000000E+000

Range Radians: –3.14 to 3.14RAD Degrees: –180 to 179DEG

Key Entry **Adjust Phase**

:ROSCillator:BANDwidth:EXTernal

Supported All Models

```
[:SOURce]:ROSCillator:BANDwidth:EXTernal
<value>[<units>]|NARRow|WIDE|MINimum|MAXimum|
[:SOURce]:ROSCillator:BANDwidth:EXTernal? |MINimum|MAXimum|
```

This command selects the external frequency bandwidth as the source for the measurement.

For values greater than 9.5 Hz, 73 Hz is used.

***RST** +9.50000000E+000

Range .5 or 73 Hz

Key Entry Ref Oscillator Ext Bandwidth

:ROSCillator:FREQuency:EXTernal

Supported All Models

```
[:SOURce]:ROSCillator:FREQuency:EXTernal <value>
[:SOURce]:ROSCillator:FREQuency:EXTernal?
```

This command makes External Ref Frequency the active function. The value that you enter sets the frequency of the external reference oscillator.

***RST** +1.0000000000000E+07 Hz

Range +1.0000000000000E+06 to +5.0000000000000E+07 Hz

Key Entry Ref Oscillator Ext Freq

Remarks If the entered frequency does not match the frequency of the entered reference, an unlocked condition will occur and an error message will appear.

:ROSCillator:SOURce

Supported All Models

```
[:SOURce]:ROSCillator:SOURce?
```

This command queries the current reference oscillator source: INT (internal) or EXT (external).

:ROSCillator:SOURce:AUTO

Supported All Models

```
[:SOURce]:ROSCillator:SOURce:AUTO ON|OFF|1|0
[:SOURce]:ROSCillator:SOURce:AUTO?
```

This command enables or disables the ability of the signal generator to automatically select between the internal and an external reference oscillator.

ON (1)	This choice enables the signal generator to detect when a valid reference signal is present at the 10 MHz IN connector and automatically switches from internal to external frequency reference.
OFF (0)	This choice selects the internal reference oscillator and disables the switching capability between the internal and an external frequency reference.

***RST** 1

Key Entry **Ref Oscillator Source Auto Off On**

List/Sweep Subsystem ([:SOURce])

A complete sweep setup requires commands from other subsystems. Table 2-2 shows the function and location of these other commands.

Table 2-2 Location of Commands from the other Subsystems

Sweep Type	Function	Command Location	Key Entry under Sweep/List key
List and Step	Start/stop frequency sweep	":FREQuency:MODE" on page 36	**Freq Off On**
	Start/stop amplitude sweep	":MODE" on page 54	**Amptd Off On**
	Start/stop frequency and amplitude sweep[a]	":MODE" on page 54 ":FREQuency:MODE" on page 36	**Freq & Amptd Off On**
	Enables or Disables the waveform sweep	"[:STATe]" on page 164	**Waveform Off On**
	Set up and control sweep triggering[b]	"Trigger Subsystem" on page 115	See the "Trigger Subsystem"
List	Load a list sweep file	":LOAD:LIST" on page 88 and page 88	**Load From Selected File**
	Store list sweep data to a file	":STORe:LIST" on page 85 and page 85	**Store To File**
	Selects the waveform for the current waveform sequence	":LIST:WAVeform" on page 48	no softkey
Step	Start frequency sweep	":FREQuency:STARt" on page 38	**Freq Start**
	Store list sweep data to a file	":STORe:LIST" on page 85 and page 85	**Store To File**
	Start amplitude sweep	":STARt" on page 55	**Amptd Start**
	Stop amplitude sweep	":STOP" on page 56	**Amptd Stop**

a. Execute both commands to start or stop a frequency and amplitude sweep.
b. For point to point triggering, see ":LIST:TRIGger:SOURce" on page 46.

:LIST:DIRection

Supported All Models

```
[:SOURce]:LIST:DIRection UP|DOWN
[:SOURce]:LIST:DIRection?
```

This command sets the direction of a list or step sweep.

UP — This choice enables a sweep in an ascending order:

- first to last point for a list sweep
- start to stop for a step sweep

DOWN — This choice reverses the direction of the sweep.

***RST** UP

Key Entry Sweep Direction Down Up

:LIST:DWELl

Supported All Models

```
[:SOURce]:LIST:DWELl <value>{,<value>}
[:SOURce]:LIST:DWELl?
```

This command sets the dwell time for the current list sweep points.

Dwell time is used when IMMediate is the trigger source. Refer to ":LIST:TRIGger:SOURce" on page 46 for the trigger setting.

The dwell time is the amount of time the sweep is guaranteed to pause after setting the frequency and/or power for the current point.

The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

The variable <value> is expressed in units of seconds with a 0.001 resolution.

NOTE The dwell time (<value>) does not begin until the signal generator has settled for the current frequency and/or amplitude change.

Range 0.0001–100

:LIST:DWELl:POINts

Supported All Models

```
[:SOURce]:LIST:DWELl:POINts?
```

This command queries the signal generator for the number of dwell points in the current list sweep file.

:LIST:DWELl:TYPE

Supported All Models

```
[:SOURce]:LIST:DWELl:TYPE LIST|STEP
[:SOURce]:LIST:DWELl:TYPE?
```

This command toggles the dwell time for the list sweep points between the values defined in the list sweep and the value for the step sweep.

LIST This choice selects the dwell times from the list sweep. Refer to ":LIST:DWELl" on page 43 for setting the list dwell points.

STEP This choice selects the dwell time from the step sweep. Refer to ":SWEep:DWELl" on page 48 for setting the step dwell.

***RST** LIST

Key Entry **Dwell Type List Step**

:LIST:FREQuency

Supported All Models

```
[:SOURce]:LIST:FREQuency <value>{,<value>}
[:SOURce]:LIST:FREQuency?
```

This command sets the frequency values for the current list sweep points.

The maximum number of list sweep points is 1,601.

The variable <value> is expressed in units of Hertz.

The setting enabled by this command is not affected by signal generator power-on, preset, or *RST. Option 501 is specific to the N5181A.

Range Option 501: 1.0E3–1E9

Option 503: 1.0E3–3E9

Option 506: 1.0E3–6E9

:LIST:FREQuency:POINts

Supported All Models

```
[:SOURce]:LIST:FREQuency:POINts?
```

This command queries the current list sweep file for the number of frequency points.

:LIST:MANual

Supported All Models

`[:SOURce]:LIST:MANual <value>|UP|DOWN`

`[:SOURce]:LIST:MANual?`

This command sets a list or step sweep point as the current sweep point controlling the frequency and power output.

If list or step mode is controlling frequency or power, or both, then the indexed point in the respective list(s) will be used.

Entering a value with this command will have no effect, unless MANual is the selected mode. Refer to ":LIST:MODE" on page 45 for setting the proper mode.

If the point selected is beyond the length of the longest enabled list, then the point will be set to the maximum possible point, and an error will be generated.

Range List Sweep: 1–1601 Step Sweep: 2-65535

Key Entry **Manual Point**

:LIST:MODE

Supported All Models

`[:SOURce]:LIST:MODE AUTO|MANual`

`[:SOURce]:LIST:MODE?`

This command sets the operating mode for the current list or step sweep.

AUTO This choice enables the selected sweep type to perform a sweep of all points.

MANual This choice enables you to select a single sweep point. The selected point controls the frequency and/or amplitude according to the sweep type. Refer to ":LIST:MANual" on page 45 for selecting a sweep point.

***RST** AUTO

Key Entry **Manual Mode Off On**

:LIST:POWer

Supported All Models

`[:SOURce]:LIST:POWer <value>{,<value>}`

`[:SOURce]:LIST:POWer?`

This command sets the amplitude for the current list sweep points.

The maximum number of list sweep points is 1,601.

Range Refer to "[:LEVel][:IMMediate][:AMPLitude]" on page 57 for output power ranges.

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:LIST:POWer:POINts

Supported All Models

```
[:SOURce]:LIST:POWer:POINts?
```

This command queries the number of power points in the current list sweep file.

:LIST:RETRace

Supported All Models

```
[:SOURce]:LIST:RETRace ON|OFF|1|0
[:SOURce]:LIST:RETRace?
```

This command configures the sweep to retrace to the first sweep point, or stop at the last sweep point upon completion of each sweep.

On (1) The sweep retraces to the first sweep point.

Off (0) The sweep stays at the last sweep point of the completed sweep and stays there until sweep is initiated and triggered again. When sweep is initiated and triggered again the sweep point moves to the first point of the sweep.

***RST** 1

Key Entry **Sweep Retrace Off On**

:LIST:TRIGger:SOURce

Supported All Models

```
[:SOURce]:LIST:TRIGger:SOURce BUS|IMMediate|EXTernal|KEY|TIMer|MANual
[:SOURce]:LIST:TRIGger:SOURce?
```

This command sets the point trigger source for a list or step sweep event.

To set the sweep trigger, see ":TRIGger[:SEQuence]:SOURce" on page 116.

BUS This choice enables GPIB triggering using the `*TRG` or `GET` command, or LAN and USB triggering using the `*TRG` command.

IMMediate This choice enables immediate triggering of the sweep event.

EXTernal This choice enables the triggering of a sweep event by an externally applied signal at the TRIGGER IN connector.

Trigger KEY This choice enables triggering by pressing the front-panel **Trigger** hardkey.

TIMer This choice enables the trigger timer.

Example

```
:LIST:TRIG:SOUR BUS
```

The preceding example sets the trigger source to the instrument BUS.

***RST** IMM

Key Entry **Bus** **Free Run** **Ext** **Trigger Key** **Timer Trigger**

:LIST:TYPE

Supported All Models

```
[:SOURce]:LIST:TYPE LIST|STEP
[:SOURce]:LIST:TYPE?
```

This command toggles between the two types of sweep.

LIST	This type of sweep has arbitrary frequencies and amplitudes.
STEP	This type of sweep has equally spaced frequencies and amplitudes.

***RST** STEP

Key Entry **Sweep Type List Step**

:LIST:TYPE:LIST:INITialize:FSTep

Supported All Models

CAUTION The current list sweep data will be overwritten once this command is executed. If needed, save the current data. Refer to ":STORe:LIST" on page 85 for storing list sweep files.

```
[:SOURce]:LIST:TYPE:LIST:INITialize:FSTep
```

This command replaces the loaded list sweep data with the settings from the current step sweep data points.

You can load only one sweep list at a time.

The maximum number of list sweep points is 1,601. When copying the step sweep settings over to a list sweep, ensure that the number of points in the step sweep do not exceed the maximum list sweep points.

Key Entry **Load List From Step Sweep**

:LIST:TYPE:LIST:INITialize:PRESet

Supported All Models

CAUTION The current list sweep data will be overwritten once this command is executed. If needed, save the current data. Refer to ":STORe:LIST" on page 85 for storing list sweep files.

```
[:SOURce]:LIST:TYPE:LIST:INITialize:PRESet
```

This command replaces the current list sweep data with a factory-defined file consisting of one point at a frequency, amplitude, and dwell time.

Key Entry **Preset List**

:LIST:WAVeform

Supported N5182A

CAUTION The current list sweep data will be overwritten once this command is executed. If needed, save the current data. Refer to ":STORe:LIST" on page 85 for storing list sweep files.

`[:SOURce]:LIST:WAVeform <name>{,<name>}`

`[:SOURce]:LISt:WAVeform?`

This command sets the waveform values for the current list waveform sequence.

Example

```
:LIST:WAV "WFM1:RAMP_TEST_WFM","WFM1:SINE_TEST_WFM"
```

The preceding example loads the waveforms RAMP_TEST_WFM and SINE_TEST_WFM into the waveform section of the List Table.

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:LIST:WAVeform:POINts

Supported N5182A

`[:SOURce]:LIST:WAVeform:POINts?`

This query returns the number of waveform points in the current list sweep file.

:SWEep:DWELl

Supported All Models

`[:SOURce]:SWEep:DWELl <value>`

`[:SOURce]:SWEep:DWELl?`

This command enables you to set the dwell time for a step sweep.

The variable <value> is expressed in units of seconds with a 0.001 resolution.

The dwell time is the amount of time the sweep is guaranteed to pause after setting the frequency and/or power for the current point.

NOTE The dwell time (<value>) does not begin until the signal generator has settled for the current frequency and/or amplitude change.

***RST** +2.00000000E–003

Range 0.0001–100

Key Entry **Step Dwell**

Remarks Dwell time is used when the trigger source is set to `IMMediate`. Refer to ":LIST:TRIGger:SOURce" on page 46 for the trigger setting.

:SWEep:POINts

Supported All Models

```
[:SOURce]:SWEep:POINts <value>
[:SOURce]:SWEep:POINts?
```

This command defines the number of step sweep points.

***RST** 101

Range 2–65535

Key Entry **# Points**

:SWEep:SPACing

Supported All Models

```
[:SOURce]:SWEep:SPACing LINear|LOGarithmic
[:SOURce]:SWEep:SPACing?
```

This command enables the signal generator linear or logrithmic sweep modes. These commands require the signal generator to be in step mode.

These commands require the signal generator to be in step mode.

The instrument uses the specified start frequency, stop frequency, and number of points for both linear and log measurements.

***RST** LIN

Key Entry **Step Spacing LIN LOG**

Power Subsystem ([:SOURce]:POWer)

:ALC:LEVel

Supported All Models

```
[:SOURce]:POWer:ALC:LEVel <value><unit>
[:SOURce]:POWer:ALC:LEVel?
```

This command sets the automatic leveling control (ALC) level. Use this command after setting the attenuation auto mode to On. Refer to ":ATTenuation:AUTO" on page 53 for setting the attenuation auto mode.

The ALC is used to maintain the signal generator's output power level by compensating for power fluctuations due to drift, band changes, or load variations. After you set the ALC level, the signal generator's output power is monitored and corrected so that the power level setting is maintained.

Example

```
:POW:ALC:LEV 10DB
```

The preceding example sets the ALC to 10 dB.

***RST** +1.00000000E+000

Range -20 to 20

Key Entry **Set ALC Level**

:ALC:SEARch

Supported All Models

```
[:SOURce]:POWer:ALC:SEARch ON|1|ONCE
[:SOURce]:POWer:ALC:SEARch?
```

This command enables or disables the internal power search calibration. A power search is recommended for pulse-modulated signals with pulse widths less than one microsecond.

ON (1)	This choice executes the power search automatically with each change in RF frequency or power.
ONCE	This choice executes a single power search of the current RF output signal.

***RST** 1

Key Entry **Do Power Search**

Remarks Use this command when the ALC state is set to OFF (0). Refer to ":ALC[:STATe]" on page 52 for setting the ALC state.

:ALC:SEARch:REFerence

Supported All Models

```
[:SOURce]:POWer:ALC:SEARch:REFerence FIXed|MODulated
[:SOURce]:POWer:ALC:SEARch:REFerence?
```

This command sets either fixed or modulated modes of power search.

FIXed	This choice uses a 0.5 volt reference.
MODulated	This choice uses the RMS value of the current I/Q modulation.
***RST**	FIX
Key Entry	**Power Search Reference Fixed Mod**

:ALC:SEARch:SPAN:START

Supported All Models

```
[:SOURce]:POWer:ALC:SEARch:SPAN:START <value><units>
[:SOURce]:POWer:ALC:SEARch:SPAN:START?
```

This command sets the start frequency for a span power search over a user specified range.

The start frequency has no default value. The start frequency value will be the last value set before powering off the instrument.

Key Entry **Start Frequency**

:ALC:SEARch:SPAN:STOP

Supported All Models

```
[:SOURce]:POWer:ALC:SEARch:SPAN:STOP <value><units>
[:SOURce]:POWer:ALC:SEARch:SPAN:STOP?
```

This command sets the stop frequency for a span power search over a user specified range.

The stop frequency has no default value. The stop frequency value will be the last value set before powering off the instrument.

Key Entry **Stop Frequency**

:ALC:SEARch:SPAN:TYPE

Supported All Models

```
[:SOURce]:POWer:ALC:SEARch:SPAN:TYPE FULL|USER
[:SOURce]:POWer:ALC:SEARch:SPAN:TYPE?
```

This command enables you to select the frequency range for a span power search. You can specify the range (USER) or you can select the full range (FULL) of the signal generator.

Key Entry **Span Type User Full**

:ALC:SEARch:SPAN[:STATe]

Supported All Models

```
[:SOURce]:POWer:ALC:SEARch:SPAN[:STATe] ON|OFF|1|0
[:SOURce]:POWer:ALC:SEARch:SPAN[:STATe]?
```

This command enables (1) or disables (0) the span mode, allowing you to perform power searches over a selected range of frequencies. The power search corrections are then stored and used whenever the signal generator is tuned within the selected range.

:ALC[:STATe]

Supported All Models

```
[:SOURce]:POWer:ALC[:STATe] ON|OFF|1|0
[:SOURce]:POWer:ALC[:STATe]?
```

This command enables or disables the automatic leveling control (ALC) circuit.

***RST** 1

Key Entry **ALC Off On**

Remarks The purpose of the ALC circuit is to hold output power at a desired level by adjusting the signal generator's power circuits to compensate for power drift. Power drift occurs over time and changes in temperature. Refer to the N5181A/82A Agilent MXG Signal Generators User's Guide for more information on the ALC.

:ATTenuation

Supported All Models

```
[:SOURce]:POWer:ATTenuation <value><unit>
[:SOURce]:POWer:ATTenuation?
```

This command sets the signal generator's attenuator level. Before setting the attenuator level, set the ":ATTenuation:AUTO" function to Off which will disable ALC control.

In normal operation the attenuator level is selected by the signal generator's automatic loop control (ALC) which maintains the output power by adjusting internal circuits to compensate for any power fluctuations due to drift, band changes, or load variations. In some applications, such as fast pulse modulation, the ALC may not respond quickly enough to compensate for the pulse rise times. In this case you can set the attenuator and override any ALC adjustments.

The output power is the ALC level minus the attenuator setting. The attenuator is set in increments of 5 dB.

Example

```
:POW:ATT 10DB
```

The preceding example sets the attenuator to 10 dB.

***RST** +115

Range 0 to 115 dB

Key Entry **Set Atten**

:ATTenuation:AUTO

Supported All Models

```
[:SOURce]:POWer:ATTenuation:AUTO ON|OFF|1|0
[:SOURce]:POWer:ATTenuation:AUTO?
```

This command sets the state of the attenuator auto mode function.

ON (1) This selection allows the signal generator's automatic level control (ALC) to adjust the attenuator so that a specified RF power level, at the Agilent MXG's RF output connector, is maintained.

OFF (0) This choice allows for a user-selected attenuator setting that is not affected by the signal generator's ALC circuitry.

The OFF (0) selection can be used to eliminate power discontinuity normally associated with attenuator switching during power adjustments.

***RST** 1

Key Entry **Atten Hold Off On**

Remarks Refer to the ":ALC:LEVel" on page 50.

:ATTenuation:BYPass

Supported All Models

```
[:SOURce]:POWer:ATTenuation:BYPass ON|OFF|1|0
[:SOURce]:POWer:ATTenuation:BYPass?
```

This command enables or disables the attenuator bypass setting. The attenuator hold mode must be enabled to use this command.

ON (1) This selection allows the signal generator's automatic level control (ALC) to adjust the attenuator hold mode. Output power is controlled solely by the ALC setting.

OFF (0) This choice allows for a user-selected attenuator setting combined with the ALC setting.

***RST** 0

Key Entry **Atten Bypass Off On**

:MODE

Supported All Models

```
[:SOURce]:POWer:MODE FIXed|LIST
[:SOURce]:POWer:MODE?
```

This command sets the signal generator power mode to fixed or swept.

FIXed — This choice stops a power sweep, allowing the signal generator to operate at a fixed power level. Refer to "[:LEVel][:IMMediate][:AMPLitude]" on page 57 for setting the output power level.

LIST — This choice selects the swept power mode. If sweep triggering is set to immediate along with continuous sweep mode, executing the command starts the LIST or STEP power sweep.

NOTE To perform a frequency and amplitude sweep, you must also select LIST as the frequency mode. See ":FREQuency:MODE" on page 36 for selecting the list mode for a frequency sweep.

***RST** FIX

Key Entry **SWEEP Amptd Off On**

:PROTection:STATe

Supported All models

```
[:SOURce]:POWer:PROTection[:STATe] ON|OFF|1|0
[:SOURce]:POWer:PROTection[:STATe]?
```

This command enables or disables the power search protection function. The power search protection function sets the attenuator to its maximum level whenever a power search is initiated. This can be used to protect devices that are sensitive to high average power or high power changes. The trade off on using the power protection function is decreased attenuator life, as the attenuator will switch to its maximum setting during a power search.

NOTE Continual or excessive use of the power search protection function can decrease attenuator life.

ON (1) — Causes the attenuator to switch to and hold its maximum level setting during a power search.

OFF (0) — Sets the attenuator normal mode. The attenuator is not used during power search.

Example

```
:POW:PROT ON
```

The preceding example enables the power inhibit function.

***RST** 0

Key Entry **RF During Power Search Normal Minimum**

:REFerence

Supported All Models

```
[:SOURce]:POWer:REFerence <value><unit>
[:SOURce]:POWer:REFerence?
```

This command sets the power level for the signal generator RF output reference.

The RF output power is referenced to the value entered in this command.

***RST** +0.00000000E+000

Range -400 to 300dBm

Key Entry **Amptd Ref Set**

:REFerence:STATe

Supported All Models

```
[:SOURce]:POWer:REFerence:STATe ON|OFF|1|0
[:SOURce]:POWer:REFerence:STATe?
```

This command enables or disables the RF output reference.

Once the reference state is ON, all subsequent output power settings are set relative to the reference value.

ON(1) This choice will set the power reference state to ON. The unit displayed for commands, ":ANNotation:AMPLitude:UNIT" on page 71 and ":POWer" on page 114 will be expressed in dB.

OFF(0) This choice will set the power reference state to OFF.

***RST** 0

Key Entry **Amptd Ref Off On**

Remarks Amplitude offsets can be used with the amplitude reference mode.

:STARt

Supported All Models

```
[:SOURce]:POWer:STARt <value><unit>
[:SOURce]:POWer:STARt?
```

This command sets the first amplitude point in a step sweep.

***RST** –1.35000000E+002

Range Refer to "[:LEVel][:IMMediate][:AMPLitude]" on page 57 for the output power ranges.

Key Entry **Amptd Start**

:STOP

Supported All Models

```
[:SOURce]:POWer:STOP <value><unit>
[:SOURce]:POWer:STOP?
```

This command sets the last amplitude point in a step sweep.

***RST** –1.35000000E+002

Range Refer to "[:LEVel][:IMMediate][:AMPLitude]" on page 57 for the output power ranges.

Key Entry **Amptd Stop**

[:LEVel][:IMMediate]:OFFSet

Supported All Models

```
[:SOURce]:POWer[:LEVel][:IMMediate]:OFFSet <value><unit>
[:SOURce]:POWer[:LEVel][:IMMediate]:OFFSet?
```

This command sets the power offset value.

***RST** +0.00000000E+000

Range –200 dB to 200 dB

Key Entry **Amptd Offset**

Remarks This simulates a power level at a test point beyond the RF OUTPUT connector without changing the actual RF output power. The offset value only affects the displayed amplitude setting.

You can enter an amplitude offset any time in either normal operation or amplitude reference mode.

[:LEVel][:IMMediate][:AMPLitude]

Supported All Models

```
[:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude] <value><unit>
[:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude]?
```

This command sets the RF output power.

***RST** –1.10000000E+002 (Standard) or –1.44000000E+002 (with Option 1EQ)

Range

Frequency	*N5181A with Option 501, 503 and 506*	*N5182A with Option 503 and Option 506*	*N5181A/82A with Option 1EQ*[a]
<250 kHz–250 GHz	–110 to 10 dBm	–110 to 10 dBm	<–144 to 10 dBm
>250 kHz–1 GHz	–110 to 18 dBm	–110 to 18 dBm	<–144 to 18 dBm
>1–3 GHz	–110 to 16 dBm	–110 to 16 dBm	<–144 to 16 dBm
>3–4 GHz	–110 to 16 dBm	–110 to 16 dBm	<–144 to 16 dBm
>4–6 GHz	–110 to 4 dBm	–110 to 4 dBm	<–144 to 4 dBm

a. Settable, but specified to –127 dBm with Option 1EQ.

Key Entry **Amplitude**

Remarks The ranges for this command are specified values from the data sheet.

Tsweep Subsystem ([:SOURce])

:TSWeep

Supported All Models

`[:SOURce]:TSWeep`

This command aborts the current sweep, then either arms or arms and starts a single list, depending on the trigger type.

The command performs the following:

- arms a single sweep when BUS, EXTernal, or Trigger KEY is the trigger source selection
- arms and starts a single sweep when IMMediate is the trigger source selection

Key Entry **Single Sweep**

3 System Commands

This chapter provides SCPI descriptions for subsystems dedicated to peripheral signal generator operations common to all Agilent MXG models. This chapter contains the following major sections:

Calibration Subsystem (:CALibration)

:DCFM

Supported All

`:CALibration:DCFM`

This command initiates a DCFM or DCΦM calibration depending on the currently active modulation. This calibration eliminates any dc or modulation offset of the carrier signal.

NOTE If the calibration is performed with a dc signal applied, any deviation provided by the dc signal will be removed and the new zero reference point will be at the applied dc level. The calibration will have to be performed again when the dc signal is disconnected to reset the carrier signal to the correct zero reference.

Key Entry **DCFM / DCFM Cal**

Remarks Use this calibration for externally applied signals. While the calibration can also be performed for internally generated signals, dc offset is not a normal characteristic for them.

:IQ:DC

Supported N5182A

`:CALibration:IQ:DC`

This command performs a one to two second adjustment that is not traceable to a standard. However, it will minimize errors associated with IQ gain, quadrature, and offset voltages. This adjustment minimizes errors for the current signal generator setting and at a single frequency. The DC adjustment is volatile and must be repeated with each signal generator setting change. This command can be sent while the RF On/Off is set to Off and the adjustment will still be valid when the RF is enabled. IQ must be on to perform this adjustment.

The I/Q DC adjustment is dependent upon a number of instrument settings. If any of the instrument settings change, the adjustment will become invalid. The dependent instrument settings are:

- RF frequency
- I/Q attenuation level
- Baseband generator settings
- I/Q polarity settings
- Baseband filter settings
- Path settings (Internal I/Q Mux Path 1 or Path 2)
- I/Q calibration (the I/Q DC calibration will be invalidated if any other I/Q calibration is executed or if the **Revert to Factory Default** key is pressed)
- Temperature (±5 degrees)

The following instrument states will not invalidate the I/Q DC calibration:

- Power level changes
- I/Q Impairments

Key Entry **Execute Cal** (with **Calibration Type User Full** set to DC)

:IQ:DEFault

Supported N5182A

```
:CALibration:IQ:DEFault
```

This command will restore the original factory calibration data for the internal I/Q modulator.

Key Entry **Revert to Default Cal Settings**

:IQ:FULL

Supported N5182A

```
:CALibration:IQ:FULL
```

This command performs an adjustment to the I/Q offset, gain and quadrature for the full-frequency range (regardless of the start and stop frequency settings) and stores the results in the signal generator's firmware.

Key Entry **Execute Cal** (with **Calibration Type User Full** set to Full)

Remarks Start and stop frequencies will default to the full frequency range of the signal generator.

:IQ:STARt

Supported N5182A

```
:CALibration:IQ:STARt <value><unit>
:CALibration:IQ:STARt?
```

This command sets the start frequency and automatically sets the calibration type to User for an I/Q calibration.

The start frequency must be less than the current value of the stop frequency.

Range Option 503: 100kHz–3GHz

Option 506: 100kHz–6GHz

Key Entry **Start Frequency**

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or `*RST`.

:IQ:STOP

Supported N5182A

```
:CALibration:IQ:STOP <value><unit>
:CALibration:IQ:STOP?
```

This command sets the stop frequency and automatically sets the calibration type to User for an I/Q calibration.

The stop frequency must be greater than the current value of the start frequency.

Range Option 503: 100kHz–3GHz

Option 506: 100kHz–6GHz

Key Entry **Stop Frequency**

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:IQ:TYPE

Supported N5182A

```
:CALibration:IQ:TYPE DC|USER|FULL
:CALibration:IQ:TYPE?
```

This command sets the IQ calibration type.

Key Entry **Calibration Type DC User Full**

***RST** DC

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:IQ:[:USER]

Supported N5182A

```
:CALibration:IQ[:USER]
```

This command performs a IQ calibration according to the IQ calibration type. For information on selecting the type of IQ calibration, refer to ":IQ:TYPE" on page 62.

Key Entry **Execute Cal**

Communication Subsystem (:SYSTem:COMMunicate)

NOTE The settings enabled by the LAN commands are not affected by signal generator power-on, preset, or *RST.

:GPIB:ADDRess

Supported All

```
:SYSTem:COMMunicate:GPIB:ADDRess <number>
:SYSTem:COMMunicate:GPIB:ADDRess?
```

This command sets the signal generator's GPIB address.

Range 0–30

Key Entry **GPIB Address**

:GTLocal

Supported All

```
:SYSTem:COMMunicate:GTLocal
```

This command sets the signal generator to local mode which enables front panel operation.

Key Entry **Local**

:LAN:CONFig

Supported All Models

```
:SYSTem:COMMunicate:LAN:CONFig DHCP|MANual|AUTO
:SYSTem:COMMunicate:LAN:CONFig?
```

NOTE The SCPI query for the LAN setup returns the last power on state setting, which may or may not be the currently displayed setting.

This command sets the signal generator's internet protocol (IP) address.

`MANual`	The user assigns an IP address to the signal generator.
`DHCP`	The network assigns an IP address to the signal generator.
`AUTO`	The network assigns an IP address to the signal generator with a fallback to Auto-IP if DHCP fails. If both DHCP and Auto-IP fail, manual configuration will be used.

Example

```
:SYST:COMM:LAN:CONF DHCP
```

The preceding example sets up the signal generator LAN configuration to use a DHCP IP address.

Key Entry **LAN Config**

Remarks The SCPI query returns the current setting, not the saved setting.

:LAN:DOMain

Supported All

```
:SYSTem:COMMunicate:LAN:DOMain <string>
:SYSTem:COMMunicate:LAN:DOMain?
```

This command defines the domain name of the signal generator's DNS server.

This entry defines the DNS server for the signal generator LAN connection.

Key Entry **Domain Name**

Remarks The SCPI query returns the current setting, not the saved setting.

:LAN:DNS:DYNamic

Supported All

```
:SYSTem:COMMunicate:LAN:DNS:DYNamic ON|OFF|1|0
:SYSTem:COMMunicate:LAN:DNS:DYNamic?
```

This command turns dynamic Domain Name System (DNS) on/off.

Key Entry Dynamic DNS Naming Off On

Remarks The SCPI query returns the current setting, not the saved setting.

:LAN:DNS:OVERride

Supported All

```
:SYSTem:COMMunicate:LAN:DNS:OVERride ON|OFF|1|0
:SYSTem:COMMunicate:LAN:DNS:OVERride?
```

This command enables you to override the DNS server that is returned by the DHCP server. The LAN configuration type must be set to Auto or DHCP to use this feature.

Key Entry **DNS Server Override Off On**

Remarks The SCPI query returns the current setting, not the saved setting.

If DNS Service Override is set to On, the DNS server setting defined with the DNS Server softkey is used.

If DNS Service Override is set to Off, the setting returned by the DHCP Server is used.

:LAN:DNS[:SERVer]

Supported All

```
:SYSTem:COMMunicate:LAN:DNS[:SERVer] <ipstring>
```

This command defines the IP address of the signal generator DNS server.

This entry defines the DNS server for the signal generator LAN connection.

Key Entry DNS Server

Remarks The SCPI query returns the current setting, not the saved setting.

:LAN:GATeway

Supported All

```
:SYSTem:COMMunicate:LAN:GATeway "<ipstring>"
:SYSTem:COMMunicate:LAN:GATeway?
```

This command sets the gateway for local area network (LAN) access to the signal generator from outside the current sub-network.

Key Entry Default Gateway

Remarks Using an empty string restricts access to the signal generator to local hosts on the LAN.

The SCPI query returns the current setting, not the saved setting.

:LAN:HOSTname

Supported All

```
:SYSTem:COMMunicate:LAN:HOSTname "<string>"
:SYSTem:COMMunicate:LAN:HOSTname?
```

This command sets the signal generator's local area network (LAN) connection hostname.

Key Entry Hostname

Remarks The SCPI query returns the current setting, not the saved setting.

:LAN:IDENtify

Supported All

```
:SYSTem:COMMunicate:LAN:IDENtify ON|OFF|1|0
```

This command controls the LAN identify feature.

`ON(1)` The command enables device identification by displaying the full-screen message "Identify: <IP Address>" on the signal generator's front panel; the LAN Status indicator will also show "IDENTIFY". For more information, refer to the Programming Guide.

`OFF(0)` This command disables device identification by clearing the message on the signal generator's front panel and returning the LAN Status indicator to display the current network state. For more information, refer to the Programming Guide.

:LAN:IP

Supported All

```
:SYSTem:COMMunicate:LAN:IP "<ipstring>"
:SYSTem:COMMunicate:LAN:IP?
```

This command sets the signal generator's local area network (LAN) internet protocol (IP) address for your IP network connection.

Key Entry **IP Address**

:LAN:KEEP:TIMeout

Supported All

```
:SYSTem:COMMunicate:LAN:KEEP:TIMeout <value>
:SYSTem:COMMunicate:LAN:KEEP:TIMeout?
```

This command sets the length of time for the TCP Keep Alive setting.

Range 0 sec to 3600 sec

Key Entry **TCP Keep Alive Timeout**

:LAN:KEEP[:STATe]

Supported All

```
:SYSTem:COMMunicate:LAN:KEEP[:STATe] ON|OFF|1|0
:SYSTem:COMMunicate:LAN:KEEP[:STATe]?
```

This command enables or disables the TCP Keep Alive feature.

Key Entry **TCP Keep Alive Off On**

:LAN:MONitor

Supported All

```
:SYSTem:COMMunicate:LAN:MONitor ON|OFF|1|0
:SYSTem:COMMunicate:LAN:MONitor?
```

This command enables or disables the LAN connection monitoring.

Key Entry **Connection Monitoring Off On**

:LAN:NBIos

Supported All

```
:SYSTem:COMMunicate:LAN:NBIos ON|OFF|1|0
:SYSTem:COMMunicate:LAN:NBIos?
```

This command enables or disables the RFC NETBIOS naming feature.

Key Entry **RFC NETBIOS Naming Off On**

:LAN:SUBNet

Supported All

```
:SYSTem:COMMunicate:LAN:SUBNet "<ipstring>"
:SYSTem:COMMunicate:LAN:SUBNet?
```

This command sets the signal generator's local area network (LAN) subnet mask address for your internet protocol (IP) network connection.

NOTE An error will occur if the IP address, Gateway, and subnet mask have conflicting settings.

Key Entry **Subnet Mask**

Remarks The SCPI query returns the current setting, not the saved setting.

Diagnostic Subsystem (:DIAGnostic[:CPU]:INFOrmation)

:CCOunt:ATTenuator

Supported All

```
:DIAGnostic[:CPU]:INFormation:CCOunt:ATTenuator?
```

This query returns the cumulative number of times that the attenuator has been switched.

Key Entry **Diagnostic Info**

:CCOunt:PON

Supported All

```
:DIAGnostic[:CPU]:INFormation:CCOunt:PON?
```

This query returns the cumulative number of times the signal generator has been powered-on.

Key Entry **Diagnostic Info**

:CCOunt:PROTection

Supported All

```
:DIAGnostic[:CPU]:INFormation:CCOunt:PROTection?
```

This query returns the cumulative number of times the reverse power protection has been cycled.

Key Entry **Diagnostic Info**

:DISPlay:OTIMe

Supported All Models

```
:DIAGnostic[:CPU]:INFormation:DISPlay:OTIMe?
```

This query returns the cumulative number of hours the display has been on.

Key Entry **Diagnostic Info**

:LICense:AUXiliary

Supported All Models

```
:DIAGnostic[:CPU]:INFormation:LICense:AUXiliary?
```

This query returns a list of licenses for software applications associated with the signal generator that have the software license file installed on the PC, as opposed to a license key installed on the signal generator. However this query does not return demo licenses for Arb-based applications.

Key Entry **Auxiliary Software Options**

Remarks If you use the signal generator with a PC that has a copy of a software application for which a license shows with this query, the software automatically accesses and installs the license on the PC.

To access Arb-based demo software licenses, see :LICense:WAVeform. To view option numbers for software applications that use license keys, see ":OPTions" on page 69.

:LICense:WAVeform

Supported All Models

```
:DIAGnostic[:CPU]:INFormation:LICense:WAVeform?
```

This query returns a list of Arb-based licenses (including demo) for software applications associated with the signal generator that have the software license file installed on the PC, as opposed to a license key installed on the signal generator. These waveform licenses are created by the software application in a license file on the PC. Refer to ":WLICence[:VALue]" on page 70 for more information.

The response format is a series of comma-separated entries enclosed in quotation marks. The first field is the waveform type number and the second is a text description of the license.

Key Entry **Waveform Licenses**

Remarks If a license appears in this list, this means that you can transfer waveform files, created with the associated Arb-based software application to another signal generator if the other signal generator has the same license. For more information, refer to the command, ":LICense:AUXiliary" on page 68.

For a list of option numbers for software applications that use license keys, see ":OPTions".

:OPTions

Supported All Models

```
:DIAGnostic[:CPU]:INFormation:OPTions?
```

This query returns a comma-separated list of internally installed signal generator options.

Key Entry **Instrument Options**

:OPTions:DETail

Supported All Models

```
:DIAgnostic[:CPU]:INFormation:OPTions:DETail?
```

This query returns the options that are installed along with the option revision and DSP version if applicable.

Key Entry **Options Info**

:OTIMe

Supported All Models

```
:DIAGnostic[:CPU]:INFormation:OTIMe?
```

This query returns the cumulative number of hours that the signal generator has been on.

Key Entry **Diagnostic Info**

:REVision

Supported All Models

```
:DIAGnostic[:CPU]:INFormation:REVision?
```

This query returns the CPU bootstrap read only memory (boot ROM) revision date. In addition, the query returns the revision, creation date, and creation time of the main firmware.

Key Entry **Diagnostic Info**

:SDATe

Supported All Models

```
:DIAGnostic[:CPU]:INFormation:SDATe?
```

This query returns the date and time of the firmware revision.

Key Entry **Diagnostic Info**

:WLICence[:VALue]

Supported N5182A with Option 651/652/654

```
:DIAGnostic[:CPU]:INFormation:WLIcense[:VALue]? <type_num>
```

This query returns the number of seconds remaining on the waveform license for the type of waveform designated by the <type_num> variable number. The type variable number is obtained using the :LICense:WAVeform command shown on page 69. Zero is returned for non-existent and expired licenses. The value 2^32 -1 (4,294,967,295) is returned for licenses that do not expire.

Display Subsystem (:DISPlay)

:ANNotation:AMPLitude:UNIT

Supported All Models

```
:DISPlay:ANNotation:AMPLitude:UNIT DBM|DBUV|DBUVEMF|V|VEMF|DB
:DISPlay:ANNotation:AMPLitude:UNIT?
```

This command sets the displayed front panel amplitude units.

If the amplitude reference state is set to on, the query returns units expressed in dB. Setting any other unit will cause a setting conflict error stating that the amplitude reference state must be set to off. Refer to, ":REFerence:STATe" on page 55 for more information.

***RST** DBM

:ANNotation:CLOCk:DATE:FORMat

Supported All Models

```
:DISPlay:ANNotation:CLOCk:DATE:FORMat MDY|DMY
:DISPlay:ANNotation:CLOCk:DATE:FORMat?
```

This command enables the selection of the date format. The choices are month-day-year (MDY) or day-month-year (DMY) format.

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:ANNotation:CLOCk[:STATe]

Supported All Models

```
:DISPlay:ANNotation:CLOCk[:STATe] ON|OFF|1|0
:DISPlay:ANNotation:CLOCk[:STATe]?
```

This command enables or disables the digital clock view in the lower right side of the front panel display.

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:BRIGhtness

Supported All Models

```
:DISPlay:BRIGhtness <value>
:DISPlay:BRIGhtness?
```

This command sets the display brightness (intensity). The brightness can be set to the minimum level (0.02), maximum level (1), or in between by using fractional numeric values (0.03–0.99).

Range 0.02–1

Key Entry **Brightness**

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:CAPTure

Supported All Models

```
:DISPlay:CAPTure
```

This event command enables the user to capture the current display and store it in the signal generator's memory.

Remarks The display capture is stored as DISPLAY.BMP in the Binary file system. This file is overwritten with each subsequent display capture. The file can be down-loaded in the following manner:

1. Log on to the signal generator using ftp.
2. Change (cd) to the BIN directory.
3. Retrieve the file by using the get command or by using the :MEM:DATA query on page 81.

:CMAP:DEFaults

Supported All Models

```
:DISPlay:CMAP:DEFault [<palette:{BRIGht}|DARK|MONOchrome>]
```

This command selects the color palette for the instrument display.

Key Entry **Bright Color** Dark Color **Monochrome**

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:CONTrast

Supported All Models

```
:DISPlay:CONTrast <value>
:DISPlay:CONTrast?
```

This command sets the contrast of the LCD display. The contrast can be set to the maximum level (1), minimum level (0), or in between by using fractional numeric values (0.001–0.999).

Range 0–1

Key Entry Display contrast hardkeys are located below the display.

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:REMote

Supported All Models

```
:DISPlay:REMote ON|OFF|1|0
:DISPlay:REMote?
```

This command enables or disables the display updating when the signal generator is remotely controlled.

ON (1) This choice updates the signal generator display (Text Area) so you can see the settings as the commands are executed, however, this will degrade the signal generator speed. Frequency Area, Amplitude Area, and status LEDs continue to update. For more information on the front panel display description, refer to the *User's Guide*.

OFF (0) This choice turns off the display (Text Area) updating while further optimizing the signal generator for speed. No Text Area updates occur but the Frequency Area, Amplitude Area, and status LEDs continue to update. For more information on the front panel display description, refer to the *User's Guide*.

Key Entry Update in Remote Off On

Remarks The setting enabled by this command is not affected by signal generator preset or *RST. However, cycling the signal generator power will reset it to zero.

[:WINDow][:STATe]

Supported All Models

```
:DISPlay[:WINDow][:STATe] ON|OFF|1|0
:DISPlay[:WINDow][:STATe]?
```

This command is used to either blank out (OFF or 0) the display screen or turn it on (ON or 1).

Remarks *RST and presetting the signal generator or cycling the power will turn the display on.

IEEE 488.2 Common Commands

*CLS

Supported All Models

*CLS

The Clear Status (CLS) command clears the status byte by emptying the error queue and clearing all the event registers including the Data Questionable Event Register, the Standard Event Status Register, the Standard Operation Status Register and any other registers that are summarized in the status byte.

*ESE

Supported All Models

*ESE <data>

The Standard Event Status Enable (ESE) command sets the Standard Event Status Enable Register.

The variable <data> represents the sum of the bits that will be enabled.

Range 0–255

Remarks The setting enabled by this command is not affected by signal generator preset or *RST. However, cycling the signal generator power will reset this register to zero.

Refer to the *Programming Guide* for more information.

*ESE?

Supported All Models

*ESE?

The Standard Event Status Enable (ESE) query returns the value of the Standard Event Status Enable Register.

Remarks Refer to the *Programming Guide* for more information.

*ESR?

Supported All Models

CAUTION This is a destructive read. The data in the register is latched until it is queried. Once queried, the data is cleared.

`*ESR?`

The Standard Event Status Register (ESR) query returns the value of the Standard Event Status Register.

Remarks Refer to the *Programming Guide* for more information.

*IDN?

Supported All Models

`*IDN?`

The Identification (IDN) query outputs an identifying string. The response will show the following information:

`<company name>, <model number>, <serial number>, <firmware revision>`

Key Entry Diagnostic Info

Remarks The identification information can be modified. Refer to :SYST:IDN on page 103 for more information.

*OPC

Supported All Models

`*OPC`

The Operation Complete (OPC) command sets bit 0 in the Standard Event Status Register when all pending operations have finished.

*OPC?

Supported All Models

*OPC?

The Operation Complete (OPC) query returns the ASCII character 1 in the Standard Event Status Register when all pending operations have finished.

This query stops any new commands from being processed until the current processing is complete.

*OPT?

Supported All Models

*OPT?

The options (OPT) query returns a comma-separated list of all of the instrument options currently installed on the signal generator.

Key Entry **Instrument Options**

*PSC

Supported

*PSC ON|OFF|1|0

The Power-On Status Clear (PSC) command controls the automatic power-on clearing of the Service Request Enable Register, the Standard Event Status Enable Register, and device-specific event enable registers.

ON (1) This choice enables the power-on clearing of the listed registers.

OFF (0) This choice disables the clearing of the listed registers and they retain their status when a power-on condition occurs.

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

*PSC?

Supported All Models

*PSC?

The Power-On Status Clear (PSC) query returns the flag setting as enabled by the *PSC command.

*RCL

Supported All Models

*RCL <reg>,<seq>

The Recall (RCL) command recalls the state from the specified memory register <reg> of the specified sequence <seq>.

Range Registers: 0–99 Sequences: 0–9

Key Entry **RECALL Reg** **Select Seq:**

*RST

Supported All Models

```
*RST
```

The Reset (RST) command resets most signal generator functions to factory-defined conditions.

Remarks Each command shows the *RST value if the setting is affected.

The settings enabled by this command is not affected by a signal generator power-on, preset, or *RST.

*RST uses the factory preset state which is better for automated testing, for example sweep mode is set to single.

For a comparison of the SCPI preset commands, refer to Table 3-1, "Preset SCPI Commands Overview," on page 106.

*SAV

Supported All Models

```
*SAV <reg>,<seq>
```

The Save (SAV) command saves signal generator settings to the specified memory register <reg> of the specified sequence <seq>.

Range *Registers*: 0–99 *Sequences*: 0–9

Key Entry **Save Reg** **Save Seq[n] Reg[nn]**

Remarks The save function does not save all signal generator settings. Refer to the *N5181A/82A Agilent MXG Signal Generators User's Guide* for more information on the save function. Refer to "*RCL" on page 76 for information on recalling saved signal generator settings.

*SRE

Supported All Models

```
*SRE <data>
```

The Service Request Enable (SRE) command sets the value of the Service Request Enable Register.

The variable <data> is the decimal sum of the bits that will be enabled. Bit 6 (value 64) is ignored and cannot be set by this command.

Range 0–255

Remarks Refer to the *Programming Guide* for more information.

Entering values from 64 to 127 is equivalent to entering values from 0 to 63.

The setting enabled by this command is not affected by signal generator preset or *RST. However, cycling the signal generator power will reset it to zero.

*SRE?

Supported All Models

*SRE?

The Service Request Enable (SRE) query returns the value of the Service Request Enable Register.

Range 0–63 or 128–191

Remarks Refer to the *Programming Guide* for more information.

*STB?

Supported All Models

*STB?

The Read Status Byte (STB) query returns the value of the status byte including the master summary status (MSS) bit.

Range 0–255

Remarks Refer to the *Programming Guide* for more information.

*TRG

Supported All Models

*TRG

The Trigger (TRG) command triggers the device if BUS is the selected trigger source, otherwise, *TRG is ignored.

*TST?

Supported All Models

*TST?

The Self-Test (TST) query initiates the internal self-test and returns one of the following results:

0 This shows that all tests passed.

1 This shows that one or more tests failed.

Key Entry **Run Complete Self Test**

*WAI

Supported All Models

*WAI

The Wait-to-Continue (WAI) command causes the signal generator to wait until all pending commands are completed, before executing any other commands.

Memory Subsystem (:MEMory)

:CATalog:BINary?

Supported N5182A with Option 651/652/654

```
:MEMory:CATalog:BINary?
```

This query outputs a list of the binary files. The return data will be in the following form:

```
<mem used>,<mem free>{,"<file listing>"}
```

The signal generator will return the two memory usage parameters and as many file listings as there are files in the directory list. Each file listing parameter will be in the following form:

```
"<file name>,<file type>,<file size>"
```

Example Output

```
1818624,519962624,"GEN_FILE11,BIN,5"
```

Key Entry Binary

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

:CATalog:LIST?

Supported All Models

```
:MEMory:CATalog:LIST?
```

This query outputs a list of the list sweep files. The return data will be in the following form:

```
<mem used>,<mem free>{,"<file listing>"}
```

The signal generator will return the two memory usage parameters and as many file listings as there are files in the directory list. Each file listing parameter will be in the following form:

```
"<file name>,<file type>,<file size>"
```

Example Output

```
1818624,519962624,"LAST,LIST,122","LIST10,LIST,69"
```

Key Entry List

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

:CATalog:SEQ?

Supported N5182A with Option 651/652/654

```
:MEMory:CATalog:SEQ?
```

This query outputs a list of the arbitrary waveform sequence files. The return data will be in the following form:

```
<mem used>,<mem free>{,"<file listing>"}
```

The signal generator will return the two memory usage parameters and as many file listings as there

are files in the directory list. Each file listing parameter will be in the following form:

"<file name>,<file type>,<file size>"

Example Output

1818624,519962624,"SEQ1_TEST,SEQ,206","SEQ_TEST,SEQ,169"

Key Entry **SEQ**

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

:CATalog:STATe?

Supported All Models

:MEMory:CATalog:STATe?

This query outputs a list of the state files. The return data will be in the following form:

<mem used>,<mem free>{,"<file listing>"}

The signal generator will return the two memory usage parameters and as many file listings as there are files in the directory list. Each file listing parameter will be in the following form:

"<file name,file type,file size>"

Example Output

1818624,519962624,"0_00,STAT,641"

Key Entry **State**

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

The :MEM:CAT:STAT command requires the use of registry number and sequence number variables. The ranges are 0-99 for <reg_num> and 0-9 for <seq_num>.

:CATalog:UFLT?

Supported All Models

:MEMory:CATalog:UFLT?

This query outputs a list of the user-flatness correction files. The return data will be in the following form:

<mem used>,<mem free>{,"<file listing>"}

The signal generator will return the two memory usage parameters and as many file listings as there are files in the directory list. Each file listing parameter will be in the following form:

"<file name,file type,file size>"

Example Output

1818624,519962624,"FLAT_1,UFLT,16","LAST,UFLT,16"”

Key Entry **User Flatness**

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

:CATalog[:ALL]?

Supported All Models

```
:MEMory:CATalog[:ALL]?
```

This query outputs a list of all the files in the memory subsystem. However it does not include files stored on the Option 651/652/654 baseband generator. The return data will be in the following form:

```
<mem used>,<mem free>{,"<file listing>"}
```

The signal generator will return the two memory usage parameters and as many file listings as there are files in the memory subsystem. Each file listing parameter will be in the following form:

```
"<file name,file type,file size>"
```

Example Output

```
1818624,519962624,0_00@STATE,STAT,641","0_01@STATE,STAT,669","A@NVHDR,NVHDR,132","A@NVMKR,NVMKR,0","A@NVWFM,NVWFM,9","COPY12@STATE,STAT,669","FLAT_1@USERFLAT,UFLT,16","GEN_FILE11@BINARY,BIN,5","LAST@LIST,LIST,122","LAST@USERFLAT,UFLT,16","PERSISTENT@STATE,STAT,1056",SEQ1_TEST@SEQ,SEQ,206
```

Key Entry **All**

Remarks Refer to the Table 1-4 on page 14 for a listing of the file types and the table on page 15 for information on the "<file name>" syntax.

:COPY[:NAME]

Supported All Models

```
:MEMory:COPY[:NAME] "<file name>","<file name>"
```

This command makes a duplicate of the requested file.

Key Entry **Copy File**

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

When copying a waveform file from volatile to non-volatile memory, the marker file and file header, associated with the waveform file, will automatically be copied at the same time.

:DATA

Supported All Models

```
:MEMory:DATA "<file_name>",<data_block>
:MEMory:DATA? "<file_name>"
```

This command loads data into signal generator memory using the <data_block> parameter and saves the data to a file designated by the "<file_name>" variable. The query returns the file contents of the file as a datablock.

A waveform file must be located in volatile waveform memory (WFM1) before it can be played by the signal generator's dual ARB player.

For downloads directly into volatile waveform memory use the path "WFM1:<file_name>". For downloads to non-volatile waveform memory, use the path "NVWFM:<file_name>".

`"<file_name>"` This variable names the destination file, including the directory path.

`<data_block>` This parameter represents the data and file length parameters. The data in the file is represented by the <data_block> variable.

Refer to the *Programming Guide* for more information on programming the status registers.

Example

```
:MEM:DATA "NVWFM:IQ_Data",#210Qaz37pY9oL
```

The preceding example downloads 10 bytes of data to a file, IQ_Data, in the signal generator's non-volatile memory. The table shown below describes the command parameters.

• **"NVWFM:IQ_Data"**	**IQ_Data is the file name. The directory path is not needed. The path "/USER/WAVEFORM/" is implied.**
• #210Qaz37pY9oL	Data block
#	This character indicates the beginning of the data block
2	Number of digits in the byte count
10	Byte count
Qaz37pY9oL	10 bytes of data

NOTE The data, `Qaz37pY9oL`, in the above command are not valid and are shown for example purposes only. Typically, ascii characters representing data are unprintable.

Remarks See "File Name Variables" on page 12 for information on the file name syntax.

:DATA:APPend

Supported All Models

```
:APPend "<file_name>",<data_block>
```

This commands appends data to an existing file stored in signal generator memory.

"<file_name>" This variable names the destination file and directory path.

`<data_block>` This parameter represents the data and file length parameters. The data in the file is represented by the <data_block> variable. The file length parameters are used by the signal generator for allocating memory.

Refer to the *Programming Guide* for more information on downloading and using files.

Example

```
:MEM:DATA:APPend "NVWFM:IQ_Data",#14Y9oL
```

The preceding example downloads and appends the data, Y9oL, to an existing file named IQ_Data

stored in the signal generator's non-volatile memory (NVWFM).

• **"NVWFM:IQ_Data"**	**IQ_Data the file name. The directory path is not needed. The path "/USER/WAVEFORM/" is implied.**
• #14Y9oL	Data block
#	This character indicates the beginning of the data block
1	Number of digits in the byte count
4	Byte count
Y9oL	4 bytes of data

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

:DELete:ALL

Supported All Models

CAUTION Using this command deletes all non-volatile user files including binary, list, state, and flatness correction files, and any saved setups which use the front panel table editor. However, this does not include files stored on the Option 651/652/654 ARB generator. You cannot recover the files after executing this command.

`:MEMory:DELete:ALL`

This command clears the file system of all non-volatile user files.

Key Entry **Delete All Files**

:DELete:BINary

Supported N5182A with Option 651/652/654

`:MEMory:DELete:BINary`

This command deletes all binary files.

Key Entry **Delete All Binary Files**

:DELete:LIST

Supported All Models

`:MEMory:DELete:LIST`

This command deletes all List files.

Key Entry **Delete All List Files**

:DELete:SEQ

Supported N5182A with Option 651/652/654

```
:MEMory:DELete:SEQ
```

This command deletes all sequence files.

Key Entry **Delete All Sequence Files**

:DELete:STATe

Supported All Models

```
:MEMory:DELete:STATe
```

This command deletes all state files.

Key Entry **Delete All Models State Files**

:DELete:UFLT

Supported All Models

```
:MEMory:DELete:UFLT
```

This command deletes all user-flatness correction files.

Key Entry **Delete All UFLT Files**

:DELete[:NAME]

Supported All Models

```
:MEMory:DELete[:NAME] "<file name>"
```

This command clears the user file system of "<file name>".

Key Entry **Delete File**

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

When deleting a waveform (WFM1) file from memory, the marker file and file header, associated with the waveform file, will also be deleted.

:FREE[:ALL]

Supported All Models

```
:MEMory:FREE[:ALL]?
```

This command returns the number of bytes left in the non-volatile user file system.

Key Entry **All**

:LOAD:LIST

Supported All Models

```
:MEMory:LOAD:LIST "<file name>"
```

This command loads a list sweep file.

Key Entry **Load From Selected File**

:MOVE

Supported All Models

```
:MEMory:MOVE "<src_file>","<dest_file>"
```

This command renames the requested file in the memory catalog.

Key Entry **Rename File**

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

:SIZE

Supported All Models

```
:MEMory:SIZE? <"filename">
```

This command returns the size of the file named <"filename"> in bytes or a –1, if the file does not exist. If the MSUS or directory is invalid, an "`ERROR: –257, File name error`" will be reported.

:STATe:COMMent

Supported All Models

```
:MEMory:STATe:COMMent <reg_num>,<seq_num>,"<comment>"
:MEMory:STATe:COMMent? <reg_num>,<seq_num>
```

This command lets you to add a descriptive comment to the saved state <reg_num>,<seq_num>. Comments can be up to 55 characters long.

Key Entry **Add Comment To Seq[n] Reg[nn]**

:STORe:LIST

Supported All Models

```
:MEMory:STORe:LIST "<file name>"
```

This command stores the current list sweep data to a file.

Key Entry **Store To File**

:CATalog

Supported All Models

```
:MMEMory:CATalog? "<msus>"
```

This command outputs a list of the files from the specified file system.

The variable "<msus>" (mass storage unit specifier) represents "<file system>". The file systems and types are shown in Table 1-4 on page 14.

The return data will be in the following form:

```
<mem used>,<mem free>{,"<file listing>"}
```

The signal generator will return the two memory usage parameters and as many file listings as there are files in the specified file system. Each file listing will be in the following format:

```
"<file name,file type,file size>"
```

Key Entry **Binary** **List** **State** **User Flatness**

Seq WFM1 **NVMKR** **NVWFM**

Remarks Refer to "MSUS (Mass Storage Unit Specifier) Variable" on page 15 for information on the use of the "<msus>" variable.

:COPY

Supported All Models

```
:MMEMory:COPY "<file name>","<file name>"
```

This command makes a duplicate of the requested file.

Key Entry **Copy File**

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

When copying a waveform file from volatile to non-volatile memory, the marker file and file header, associated with the waveform file, will automatically be copied at the same time.

:DATA

Supported All Models

```
:MMEMory:DATA "<file name>",<datablock>
:MMEMory:DATA? "<file name>"
```

This command loads <datablock> into the memory location "<file name>".

The query returns the <datablock> associated with the "<file name>".

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

:DELete:NVWFm

Supported N5182A with Option 651/652/654

```
:MMEMory:DELete:NVWFm
```

This command clears the user file system of all non-volatile arbitrary waveform files.

Key Entry **Delete All NVWFM Files**

:DELete:WFM

Supported N5182A with Option 651/652/654

```
:MMEMory:DELete:WFM
```

This command clears the user file system of all volatile arbitrary waveform files stored on the BBG.

Key Entry **Delete All WFM1 Files**

:DELete[:NAME]

Supported All

On the

```
:MMEMory:DELete[:NAME] "<file name>",["<msus>"]
```

This command clears the user file system of "<file name>" with the option of specifying the file system separately. For a list of file systems refer to Table 1-4 on page 14.

The variable "<msus>" (mass storage unit specifier) represents the file system.

Key Entry **Delete File**

Remarks If the optional variable "<msus>" is omitted, the file name needs to include the file system extension. Refer to "File Name Variables" on page 12 and "MSUS (Mass Storage Unit Specifier) Variable" on page 15 for information on the use of the file variables.

When deleting a waveform file from memory, the marker file and file header, associated with the waveform file, will also be deleted.

:HEADer:CLEar

Supported N5182A with Option 651/652/654

```
:MMEMory:HEADer:CLEar "<file name>"
```

This command sets the file header field settings to unspecified for the "<file name>" variable.

Key Entry **Clear Header**

Remarks In addition to waveforms currently running in the signal generator, it is possible to change or delete file header information on files that are not currently running but are stored in either the internal or external non-volatile memory (Example: :MMEMory:HEADer:CLEar "NVWFM:file_name").

Refer to "File Name Variables" on page 12 for information on the file name syntax.

:HEADer:DESCription

Supported N5182A with Option 651/652/654

```
:MMEMory:HEADer:DESCription "<file name>","<description>"
:MMEMory:HEADer:DESCription? "<file name>"
```

This command inserts a description for the file header.

Key Entry **Edit Description**

Remarks In addition to waveforms currently running in the signal generator, it is possible

to change or delete file header information on files that are not currently running but are stored in either the internal or external non-volatile memory (Example: :MMEMory:HEADer:DESCription "NVWFM:file_name","example_file_name").

The header description is limited to 32 characters. Refer to "File Name Variables" on page 12 for information on the file name syntax.

:LOAD:LIST

Supported All

```
:MMEMory:LOAD:LIST "<file name>"
```

This command loads a List sweep file.

Key Entry **Load From Selected File**

:MOVE

Supported All

```
:MMEMory:MOVE "<src_file>","<dest_file>"
```

This command renames the requested file in the memory catalog.

Key Entry **Rename File**

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

:STORe:LIST

Supported All

```
:MMEMory:STORe:LIST "<file name>"
```

This command stores the current list sweep data to a file.

Key Entry **Store To File**

Output Subsystem (:OUTPut)

:BLANking:AUTO

Supported All

```
:OUTPut:BLANking:AUTO ON|OFF|1|0
:OUTPut:BLANking:AUTO?
```

This command turns the RF output on or off during frequency band changes. Frequency band changes can cause the signal generator's RF output to fluctuate. The output blanking function, when active, turns off the RF output until the frequency and power settles.

`ON(1)` The RF output turns off when crossing a frequency band.

`OFF(0)` The RF output stays on, if possible, when crossing a frequency band. Refer to the *Data sheet*.

***RST** 1

Key Entry **Output Blanking Off On Auto**

Remarks Refer to the signal generator's data sheet for information on frequency switching speeds, settling times, and frequency band information.

:BLANking:STATe

Supported All

```
:OUTPut:BLANking:STATe ON|OFF|1|0
:OUTPut:BLANking:STATe?
```

This command enables or disables the RF output blanking state.

`ON(1)` The RF output turns off during frequency changes.

`OFF(0)` The RF output stays on, if possible, during frequency changes. Refer to the *Data sheet*.

***RST** 0

Remarks Refer to the signal generator's data sheet for information on frequency switching speeds, settling times, and frequency band information.

:MODulation[:STATe]

Supported All

```
:OUTPut:MODulation[:STATe] ON|OFF|1|0
:OUTPut:MODulation[:STATe]?
```

This command enables or disables the modulation of the RF output with the currently active modulation type(s).

***RST** 1

Key Entry **Mod On / Off**

Remarks Some modulation types can be simultaneously enabled such as pulse and AM.

An annunciator on the signal generator is always displayed to indicate whether modulation is switched on or off.

:PROTection[:STATe]

Supported All

```
:OUTPut:PROTection[:STATe] ON|OFF|1|0
:OUTPut:PROTection[:STATe]?
```

This command enables or disables the reverse power protection (RPP) circuit on the RF output.

CAUTION The RPP is used to protect the output amplifiers in the source by tripping a relay whenever an external high level RF signal is detected on the RF output. When the relay is tripped the output signal of the source is disabled and a warning message is displayed.

***RST** 1

Key Entry **Output Off On Auto**

[:STATe]

Supported All

```
:OUTPut[:STATe] ON|OFF|1|0
:OUTPut[:STATe]?
```

This command enables or disables the RF output.

***RST** 0

Key Entry **RF On / Off**

Remarks Although you can configure and engage various modulations, no signal is available at the RF OUTPUT connector until this command is executed.

An annunciator is always displayed on the signal generator to indicate whether the RF output is switched on or off.

Route Subsystem (:ROUTe)

[:CONNector]:SOUT

Supported N5181A/82A

```
:ROUTe[:CONNectors]:SOUT SWEep|SETTled|PVIDeo
:ROUTe:CONNectors:SOUT?
```

This command selects a signal to be routed to the rear panel Sweep Out connector.

SWEep	This choice routes the sweep out signal to the Sweep Out connector.
SETTled	This choice routes the source settled signal to the Sweep Out connector.
PVIDeo	This choice routes the pulse video signal to the Sweep Out connector.

***RST** SWE

Key Entry **Route Sweep Out**

[:CONNector]:TOUT

Supported N5181A/82A

```
:ROUTe[:CONNectors]:TOUT SWEep|SETTled|PVIDeo|PSYNc
:ROUTe[:CONNectors]:TOUT?
```

This command selects a signal to be routed to the rear panel Trig Out connector.

SWEep	This choice routes the sweep trigger out signal to the Trig Out connector.
SETTled	This choice routes the source settled signal to the Trig Out connector.
PVIDeo	This choice routes the pulse video signal to the Trig Out connector.
PSYNc	This choice routes the pulse sync signal to the Trig Out connector.

***RST** SWE

Key Entry **Route Trig Out**

Status Subsystem (:STATus)

:OPERation:CONDition

Supported All

```
:STATus:OPERation:CONDition?
```

This query returns the decimal sum of the bits for the registers that are set to one and are part of the Standard Operation Status Group. For example, if a sweep is in progress (bit 3), the value 8 is returned.

Range 0–32767

Remarks The data in this register is continuously updated and reflects current conditions.

Refer to the *Programming Guide* for more information.

:OPERation:ENABle

Supported All

```
:STATus:OPERation:ENABle <value>
:STATus:OPERation:ENABle?
```

This command determines which bits in the Standard Operation Event Register will set the Standard Operation Status Summary bit (bit 7) in the Status Byte Register.

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:OPERation:NTRansition

Supported All

```
:STATus:OPERation:NTRansition <value>
:STATus:OPERation:NTRansition?
```

This command determines which bits in the Standard Operation Condition Register will set the corresponding bit in the Standard Operation Event Register when that bit has a negative transition (1 to 0).

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:OPERation:PTRansition

Supported All

```
:STATus:OPERation:PTRansition <value>
:STATus:OPERation:PTRansition?
```

This command determines which bits in the Standard Operation Condition Register will set the corresponding bit in the Standard Operation Event Register when that bit has a positive transition (0 to 1).

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:OPERation:SUPPress

Supported All

```
:STATus:OPERation:SUPPress 0|1|ON|OFF
:STATus:OPERation:SUPPress?
```

This command disables the instrument's management of the Standard Operation Condition Register and saves 50 us of switching time.

***RST** OFF

Remarks Refer to the *Programming Guide* for more information.

:OPERation[:EVENt]

Supported All

CAUTION This is a destructive read. The data in the register is latched until it is queried. Once queried, the data is cleared.

```
:STATus:OPERation[:EVENt]?
```

This query returns the decimal sum of the bits in the Standard Operation Event Register.

Range 0–32767

Remarks The equivalent PTR or NTR filters must be set before the condition register can set the corresponding bit in the event register.

Refer to the *Programming Guide* for more information.

:PRESet

Supported All

```
:STATus:PRESet
```

This command presets all transition filters, enable registers, and error/event queue enable registers.

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:CALibration:CONDition

Supported All

```
:STATus:QUEStionable:CALibration:CONDition?
```

This query returns the decimal sum of the bits in the Data Questionable Calibration Condition Register. For example, if the DCFM or DCΦM zero calibration fails (bit 0), a value of 1 is returned.

Range 0–32767

Remarks The data in this register is continuously updated and reflects the current conditions.

Refer to the *Programming Guide* for more information.

:QUEStionable:CALibration:ENABle

Supported All

```
:STATus:QUEStionable:CALibration:ENABle <value>
:STATus:QUEStionable:CALibration:ENABle?
```

This command determines which bits in the Data Questionable Calibration Event Register will set the calibration summary bit (bit 8) in the Data Questionable Condition Register.

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:CALibration:NTRansition

Supported All

```
:STATus:QUEStionable:CALibration:NTRansition <value>
:STATus:QUEStionable:CALibration:NTRansition?
```

This command determines which bits in the Data Questionable Calibration Condition Register will set the corresponding bit in the Data Questionable Calibration Event Register when that bit has a negative transition (1 to 0).

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:CALibration:PTRansition

Supported All

```
:STATus:QUEStionable:CALibration:PTRansition <value>
:STATus:QUEStionable:CALibration:PTRansition?
```

This command determines which bits in the Data Questionable Calibration Condition Register will set the corresponding bit in the Data Questionable Calibration Event Register when that bit has a positive transition (0 to 1).

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:CALibration[:EVENt]

Supported All

CAUTION This is a destructive read. The data in the register is latched until it is queried. Once queried, the data is cleared.

```
:STATus:QUEStionable:CALibration[:EVENt]?
```

This command returns the decimal sum of the bits in the Data Questionable Calibration Event Register.

Range 0–32767

Remarks The equivalent PTR or NTR filters must be set before the condition register can set the corresponding bit in the event register.

Refer to the *Programming Guide* for more information.

:QUEStionable:CONDition

Supported All

```
:STATus:QUEStionable:CONDition?
```

This query returns the decimal sum of the bits in the Data Questionable Condition Register. For example, if the ALC Heater Detector is cold (bit 4), a value of 16 is returned.

Range 0–32767

Remarks The data in this register is continuously updated and reflects current conditions.

Refer to the *Programming Guide* for more information.

:QUEStionable:ENABle

Supported All

```
:STATus:QUEStionable:ENABle <value>
:STATus:QUEStionable:ENABle?
```

This command determines which bits in the Data Questionable Event Register will set the Data Questionable Status Group Summary bit (bit 3) in the Status Byte Register.

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:FREQuency:CONDition

Supported All

```
:STATus:QUEStionable:FREQuency:CONDition?
```

This query returns the decimal sum of the bits in the Data Questionable Frequency Condition Register. For example, if the 1 GHz internal reference clock is unlocked (bit 2), a value of 4 is returned.

Range 0–32767

Remarks The data in this register is continuously updated and reflects current conditions.

Refer to the *Programming Guide* for more information.

:QUEStionable:FREQuency:ENABle

Supported All

```
:STATus:QUEStionable:FREQuency:ENABle <value>
:STATus:QUEStionable:FREQuency:ENABle?
```

This command determines which bits in the Data Questionable Frequency Event Register will set the frequency summary bit (bit 5) in the Data Questionable Condition Register.

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:FREQuency:NTRansition

Supported All

```
:STATus:QUEStionable:FREQuency:NTRansition <value>
:STATus:QUEStionable:FREQuency:NTRansition?
```

This command determines which bits in the Data Questionable Frequency Condition Register will set the corresponding bit in the Data Questionable Frequency Event Register when that bit has a negative transition (1 to 0).

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:FREQuency:PTRansition

Supported All

```
:STATus:QUEStionable:FREQuency:PTRansition <value>
:STATus:QUEStionable:FREQuency:PTRansition?
```

This command determines which bits in the Data Questionable Frequency Condition Register will set the corresponding bit in the Data Questionable Frequency Event Register when that bit has a positive transition (0 to 1).

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:FREQuency[:EVENt]

Supported All

CAUTION This is a destructive read. The data in the register is latched until it is queried. Once queried, the data is cleared.

```
:STATus:QUEStionable:FREQuency[:EVENt]?
```

This query returns the decimal sum of the bits in the Data Questionable Frequency Event Register.

Range 0–32767

Remarks The equivalent PTR or NTR filters must be set before the condition register can set the corresponding bit in the event register.

Refer to the *Programming Guide* for more information.

:QUEStionable:NTRansition

Supported All

```
:STATus:QUEStionable:NTRansition <value>
:STATus:QUEStionable:NTRansition?
```

This command determines which bits in the Data Questionable Condition Register will set the corresponding bit in the Data Questionable Event Register when that bit has a negative transition (1 to 0).

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:POWer:CONDition

Supported All

```
:STATus:QUEStionable:POWer:CONDition?
```

This query returns the decimal sum of the bits in the Data Questionable Power Condition Register. For example, if the RF output signal is unleveled (bit 1), a value of 2 is returned.

Range 0–32767

Remarks The data in this register is continuously updated and reflects current conditions.

Refer to the *Programming Guide* for more information.

:QUEStionable:POWer:ENABle

Supported All

```
:STATus:QUEStionable:POWer:ENABle <value>
:STATus:QUEStionable:POWer:ENABle?
```

This command determines which bits in the Data Questionable Power Event Register will set the power summary bit (bit 3) in the Data Questionable Condition Register.

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:POWer:NTRansition

Supported All

```
:STATus:QUEStionable:POWer:NTRansition <value>
:STATus:QUEStionable:POWer:NTRansition?
```

This command determines which bits in the Data Questionable Power Condition Register will set the corresponding bit in the Data Questionable Power Event Register when that bit has a negative transition (1 to 0).

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:POWer:PTRansition

Supported All

```
:STATus:QUEStionable:POWer:PTRansition <value>
:STATus:QUEStionable:POWer:PTRansition?
```

This command determines which bits in the Data Questionable Power Condition Register will set the corresponding bit in the Data Questionable Power Event Register when that bit has a positive transition (0 to 1).

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable:POWer[:EVENt]

Supported All

CAUTION This is a destructive read. The data in the register is latched until it is queried. Once queried, the data is cleared.

```
:STATus:QUEStionable:POWer[:EVENt]?
```

This query returns the decimal sum of the bits in the Data Questionable Power Event Register.

Range 0–32767

Remarks The equivalent PTR or NTR filters must be set before the condition register can set the corresponding bit in the event register.

Refer to the *Programming Guide* for more information.

:QUEStionable:PTRansition

Supported All

```
:STATus:QUEStionable:PTRansition <value>
:STATus:QUEStionable:PTRansition?
```

This command determines which bits in the Data Questionable Condition Register will set the corresponding bit in the Data Questionable Event Register when that bit has a positive transition (0 to 1).

The variable <value> is the sum of the decimal values of the bits that you want to enable.

Range 0–32767

Remarks Refer to the *Programming Guide* for more information.

:QUEStionable[:EVENt]

Supported All

CAUTION This is a destructive read. The data in the register is latched until it is queried. Once queried, the data is cleared.

```
:STATus:QUEStionable[:EVENt]?
```

This query returns the decimal sum of the bits in the Data Questionable Event Register.

Range 0–32767

Remarks The equivalent PTR or NTR filters must be set before the condition register can set the corresponding bit in the event register.

Refer to the *Programming Guide* for more information.

System Subsystem (:SYSTem)

:CAPability

Supported All

```
:SYSTem:CAPability?
```

This query returns the signal generator's capabilities and outputs the appropriate specifiers:

```
(RFSOURCE WITH((AM|FM|PULM|PM)&(FSSWEEP|FLIST)&(PSSWEEP|PLIST) &TRIGGER&REFERENCE))
```

This is a list of the SCPI-defined basic functionality of the signal generator and the additional capabilities it has in parallel (a&b) and singularly (a|b).

:DATE

Supported All

```
:SYSTem:DATe <year>,<month>,<day>
:SYSTem:DATe?
```

This command sets the date as shown in the lower right area of the signal generator display.

<year> This variable requires a four digit integer.

The query returns the date in the following format:

<+year>, <+month>, <+day>

Range <month>: 1–12 <day>: 1–31

Key Entry **Time / Date**

:ERRor:CODE[:NEXt]

Supported All

```
:SYSTem:ERRor:CODE[:NEXt]?
```

This query returns the next error message number from the signal generator SCPI error queue. If there are no error messages, the query returns the following output:

```
+0
```

When there is more than one error message, the query will need to be sent for each message.

The Agilent MXG deletes the error messages from the front panel error queue after viewing the last message.

Key Entry **Error Info** **View Next Error Message**

:ERRor[:NEXt]

Supported All

```
:SYSTem:ERRor[:NEXt]?
```

This query returns the next error message from the signal generator SCPI error queue. If there are no error messages, the query returns the following output:

```
+0,"No error"
```

When there is more than one error message, the query will need to be sent for each message.

The Agilent MXG deletes the error messages from the front panel error queue after viewing the last message.

Key Entry **Error Info** **View Next Error Message**

:ERRor:SCPI[:SYNTax]

Supported All

```
:SYSTem:ERRor:SCPI[:SYNTax] ON|OFF|1|0
:SYSTem:ERRor:SCPI[:SYNTax]?
```

This command enables or disables the reporting of SCPI syntax errors to the error queue.

***RST** 0

:FILesystem:SAFemode

Supported All

```
:SYSTem:FILesystem:SAFemode ON|OFF|1|0
:SYSTem:FILesystem:SAFemode?
```

This command selects the safe mode for file handling. When safe mode is set to OFF, volatile waveform files can be edited and saved while the signal generator plays the file without signal interruption. However, it is possible with complex waveforms, for corruption of memory to occur which will be reported as an error on the front-panel display and require a reboot of the signal generator to resolve.

Example

```
:SYST:FIL:SAF ON
```

The preceding example enables the safe mode setting and waveform files cannot be edited without signal disruption while the signal generator plays them.

***RST** On

:FILesystem:STORage:EXTernal

Supported All

```
:SYSTem:FILesystem:STORage:EXTernal?
```

This query checks to see if the external USB port is actively being used for data storage and retrieval on the signal generator. A returned value of 1 means the external USB media is being used for data storage and retrieval. For more information on non-volatile storage media settings, refer to ":FILesystem:STORage:EXTernal:PATH" on page 102, ":FILesystem:STORage:TYPE" on page 102 and ":FILesystem:STORage:TYPE:AUTO" on page 103

:FILesystem:STORage:EXTernal:PATH

Supported All

```
:SYSTem:FILesystem:STORage:EXTernal:PATH <"external media root path">
:SYSTem:FILesystem:STORage:EXTernal:PATH?
```

This command selects the directory storage path on the external media. For more information, refer to the signal generator's softkey Help. For more information on non-volatile storage media settings, refer to ":FILesystem:STORage:EXTernal" on page 102, ":FILesystem:STORage:TYPE" on page 102 and ":FILesystem:STORage:TYPE:AUTO" on page 103.

Remarks When reading and writing files from or to the external media, different memory subsystem file types are marked by having a particular extender on the filename. Refer to "External Memory Path Options" table on page 102.

External Memory Path Options

Extender	Memory Subsystem
.waveform	NVWFM
.markers	NVMKR
.header	NVHDR
.state	STATE
.list	LIST
.uflat	UFLAT
.seq	SEQ
All others	BIN

:FILesystem:STORage:TYPE

Supported All

```
:SYSTem:FILesystem:STORage:TYPE INTernal|EXTernal
:SYSTem:FILesystem:STORage:TYPE?
```

This command selects the non-volatile storage location on the signal generator. For more information on non-volatile storage media settings, refer to ":FILesystem:STORage:EXTernal" on page 102, ":FILesystem:STORage:EXTernal:PATH" on page 102 and ":FILesystem:STORage:TYPE:AUTO" on page 103.

Key Entry **Storage Type Int Ext Auto**

Example

```
:SYST:FIL:STOR:TYPE EXT
```

The preceding example selects the external USB port as the location for non-volatile file storage on the signal generator.

:FILesystem:STORage:TYPE:AUTO

Supported All

```
:SYSTem:FILesystem:STORage:TYPE:AUTO ON|OFF|1|0
:SYSTem:FILesystem:STORage:TYPE:AUTO?
```

This command enables the signal generator to auto-detect when the external media is connected. When AUTO (ON|1) is selected, the file system uses the external media, if available. When the external media is removed, the file system uses the internal media.

Enables you to select how the signal generator's non-volatile user file system behaves when you attach an external media device to the front panel USB port. For more information, refer to the signal generator's softkey Help. For more information on non-volatile storage media settings, refer to ":FILesystem:STORage:EXTernal" on page 102, ":FILesystem:STORage:EXTernal:PATH" on page 102 and ":FILesystem:STORage:TYPE" on page 102.

Key Entry **Storage Type Int Ext Auto**

Example

```
:SYST:FIL:STOR:TYPE:AUTO ON
```

The preceding example selects AUTomatic as the non-volatile storage setting and the signal generator will detect if the external USB port has a memory storage device connected.

***RST**

Remarks When the external media is removed, the external non-volatile user file system effectively does not exist.

:IDN

Supported All

```
:SYSTem:IDN "string"
```

This command modifies the identification string that the `*IDN?` query returns. Sending an empty string sets the query output of `*IDN?` to its factory shipped setting. The maximum string length is 72 characters.

Remarks Modification of the *IDN? query output enables the signal generator to identify itself as another signal generator when used as a replacement.

The display diagnostic information, shown by pressing the **Diagnostic Info** softkey, is not affected by this command.

:LANGuage

Supported All

```
:SYSTem:LANGuage "SCPI"|"COMP"|"8648"
:SYSTem:LANGuage?
```

This command sets the remote language for the signal generator.

SCPI This choice provides compatibility for SCPI commands.

COMP This choice provides compatibility for the 8656B, 8657A/B signal generator which is supported by using the GPIB interface.

8648 This choice provides compatibility for the 8648A/B/C/D signal generator which is supported only through a GPIB interface.

Key Entry SCPI 8656B,8657A/B 8648A/B/C/D

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

For more information on supported SCPI commands and programming codes, refer to the *Programming Compatibility Guide*.

:LICense:INSTall

Supported All

```
:SYSTem:LICense:INSTall <license_line>|<block_of_license_lines>
```

This command installs the licenses into the signal generator.

<license_line> This choice installs a license line.

<block_of_license_lines> This choice installs a block of license lines.

Example

```
:SYST:LIC:INST "FEATURE 403 aspk 0 permanent 0 389D66FB107E9B02
HOSTID=N5182A,US00000068"
```

The preceding example installs license "FEATURE 403 aspk 0 permanent 0 389D66FB107E9B02 HOSTID=N5182A,US00000068", into the signal generator.

or to install a block of license files:

```
:SYST:LIC:INST #210Qaz37pY9oL
```

The preceding SCPI command shows an example of the syntax for installing a block of licenses into the signal generator. For more on handling block data, refer to the *Programming Guide*.

NOTE The data, Qaz37pY9oL, in the above command are not valid and are shown for example purposes only. Typically, ascii characters representing data are unprintable.

For additional information on downloading and installing licenses for applications, refer to the Agilent License Manager at *http://www.agilent.com/find/LicenseManager*.

:LICense:EXTernal:LIST

Supported All

```
:SYSTem:LICense:EXTernal:LIST?
```

This query provides a listing of the current licenses for external software installed on the signal generator.

:LICense:LIST

Supported All

```
:SYSTem:LICense:LIST?
```

This query provides a listing of the current licenses installed on the signal generator.

:LICense:REMove

Supported All

```
:SYSTem:LICense:REMove <license_line>
```

This command removes a single license line.

Example

To remove a license line:

```
:SYST:LIC:REM "FEATURE 403 aspk 0 permanent 0 389D66FB107E9B02
HOSTID=N5182A,US00000068"
```

The preceding example removes a license "FEATURE 403 aspk 0 permanent 0 389D66FB107E9B02 HOSTID=N5182A,US00000068", from the signal generator.

Remarks To remove multiple license lines: Repeat the process for removing a single license for each license line to be removed.

:PDOWn

Supported All

```
:SYSTem:PDOWn
```

This command turns off the instrument.

:PON:TYPE

Supported All

```
:SYSTem:PON:TYPE PRESet|LAST|USER
:SYSTem:PON:TYPE?
```

This command sets the defined conditions for the signal generator at power on.

PRESet This choice sets the conditions to factory- or user-defined as determined by the choice for the preset type.

LAST — This choice retains the settings at the time the signal generator was last powered down.

USER — This choice sets the power on state to be the user preset value.

Key Entry — **Power On Last Preset**

Remarks — The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

For a comparison of the SCPI preset commands, refer to Table 3-1, "Preset SCPI Commands Overview," on page 106.

:PRESet

NOTE If this SCPI command is not responding as expected, use the E4428C/38C compatibility command: `:SYST:PRESet:TYPE:NORMal` to return the front panel **Preset** key to its factory default functionality.

Supported — All

`SYSTem:PRESet`

This command returns the signal generator to a set of defined conditions. It is equivalent to pressing the front panel **Preset** hardkey.

Key Entry — **Preset**

Remarks — The defined conditions are either factory- or user-defined.

For a comparison of the SCPI preset commands, refer to Table 3-1, "Preset SCPI Commands Overview," on page 106.

Table 3-1 Preset SCPI Commands Overview

Command	Description	Remarks
*RST	This IEEE 488.2 Common Command uses the factory preset settings for the instrument preset.	Optimized for automated testing
:SYSTem:PRESet:PERSistent	Only the instrument's persistent parameters are returned to factory default value.	
:SYSTem:PON:TYPE PRESet|LAST|USER :SYSTem:PON:TYPE?	Sets the power on state (PON) to be the same as the front panel green Preset hardkey, or the last state, or to the user state.	
:SYSTem:PRESet	Performs the same preset as currently set for the front panel green Preset hardkey.	
:SYSTem:PRESet[:USER]:SAVE	Saves the current instrument state as the user preset state.	

Table 3-1 Preset SCPI Commands Overview

Command	Description	Remarks
:SYSTem:PRESet:ALL	Sets the instrument to the same default conditions performed by sequentially inputting: SYSTem:PRESet + `:SYSTem:PERSistent:PRESet`	
:SYSTem:PRESet:USER	Executes a user preset.	

:PRESet:ALL

Supported All

```
:SYSTem:PRESet:ALL
```

This command sets all states of the signal generator back to their factory default settings, including states that are not normally affected by signal generator power-on, preset, or *RST.

For a comparison of the SCPI preset commands, refer to Table 3-1, “Preset SCPI Commands Overview,” on page 106.

:PRESet:LANGuage

Supported All

```
:SYSTem:PRESet:LANGuage"SCPI"|"COMP"|"8648"
:SYSTem:PRESet:LANGuage?
```

This command sets the remote language that is available when the signal generator is preset.

SCPI	This choice provides compatibility for SCPI commands.
COMP	This choice provides compatibility for the 8656B, 8657A/B signal generator which is supported by using the GPIB interface.
8648	This choice provides compatibility for the 8648A/B/C/D signal generator which is supported only through a GPIB interface.

***RST** "SCPI"

Key Entry **SCPI** **8656B,8657A/B** **8648A/B/C/D**

:PRESet:PERSistent

Supported All

`:SYSTem:PRESet:PERSistent`

This command sets the states that are not affected by signal generator power-on, preset, or *RST to their factory default settings.

Key Entry **Restore System Settings to Default Values**

Remarks For a list of the persistent instrument factory default values refer to the *Programming Guide*.

For a comparison of the SCPI preset commands, refer to Table 3-1, "Preset SCPI Commands Overview," on page 106.

:PRESet:TYPE

Supported All

`:SYSTem:PRESet:TYPE NORMal|USER`

This command defines the Preset hardkey as either factory preset or as the user preset saved in memory.

NORMal This choice uses the factory-defined defaults when **Preset** is pressed.

COMP This choice uses the user-defined preset saved in the instrument when **Preset** is pressed. Refer to ":PRESet:USER" on page 108 and ":PRESet[:USER]:SAVE" on page 108.

Key Entry **Preset**

Remarks This command will return an error, if the USER parameter is sent without a user preset saved in the instrument.

:PRESet:USER

Supported All

`:SYSTem:PRESet:USER`

This command presets the signal generator to the user's saved state.

Key Entry **Execute User Preset**

Remarks This command presets the signal generator to the saved user-defined state.

For a comparison of the SCPI preset commands, refer to Table 3-1, "Preset SCPI Commands Overview," on page 106.

:PRESet[:USER]:SAVE

Supported All

`:SYSTem:PRESet[:USER]:SAVE`

This command saves your user-defined preset conditions to a state file.

Key Entry **Save User Preset**

Remarks Only one user-defined preset file can be saved. Subsequent saved user-defined preset files will overwrite the previously saved file.

For a comparison of the SCPI preset commands, refer to Table 3-1, "Preset SCPI Commands Overview," on page 106.

:SECurity:DISPlay

Supported All Models

```
:SYSTem:SECurity:DISPlay ON|OFF|1|0
:SYSTem:SECurity:DISPlay?
```

This command enables or disables the secure display mode.

On(1) This selection turns the signal generator display back on, showing the current settings. Cycling the signal generator power also restores the display, however the current settings may change depending on the power-on configuration choice. See ":PON:TYPE" on page 105 for information on the power-on choices available.

OFF(0) This selection blanks the signal generator's display, hiding the settings and disabling the front panel keys. While in this mode, the display shows `*** SECURE DISPLAY ACTIVATED ***`.

For more information about security functions, refer to the *N5181A/82A Agilent MXG Signal Generators User's Guide*.

Example

```
:SYST:SEC:DISP OFF
```

The preceding example enables the secure display mode.

***RST** 1

Range N/A

Key Entry **Activate Security Display**

:SECurity:ERASeall

Supported All Models

```
:SYSTem:SECurity:ERASeall
```

This command removes all user files, flatness correction files, and baseband generator files. In addition, all table editor files are returned to their original factory values.

This command differs from the :DELete:ALL command, which does not reset table editors to factory values. For more information about security functions, refer to the *N5181A/82A Agilent MXG Signal Generators User's Guide*.

Key Entry **Erase All**

:SECurity:LEVel

Supported All Models

```
:SYSTem:SECurity:LEVel NONE|ERASe|OVERwrite|SANitize
:SYSTem:SECurity:LEVel?
```

This command selects the security level operation for the signal generator.

NONE — This selection causes the signal generator to reset to factory default settings.

ERASe — This selection removes all user files, table editor files, flatness correction files, and baseband generator files.

OVERwrite — This selection removes all user files, table editor files, flatness correction files, and baseband generator files. The memory is then overwritten with random data.

SRAM	All addressable locations will be overwritten with random characters.
Hard Disk	All addressable locations will be overwritten with random characters.
Flash Memory	The flash blocks will be erased.

SANitize — This selection removes all user files, table editor files, flatness correction files, and baseband generator files using the same techniques as the OVERwrite selection for SRAM and flash memory. For the hard disk, the signal generator overwrites all addressable locations with a single character, its complement, and then with a random character.

Once you select the security level, you must execute the command from :SECurity:LEVel:STATe to arm the security level. The selected level of security operation will be executed after reboot.

NOTE Once you select a security level and arm it, you cannot change the level.

For other cleaning and security operation descriptions, see ":SECurity:ERASeall" on page 109, ":SECurity:OVERwrite" on page 111, and ":SECurity:SANitize" on page 111. For more information about security functions, refer to the *N5181A/82A Agilent MXG Signal Generators User's Guide*.

Example

```
:SYST:SEC:LEV ERASE
```

The preceding example sets the secure mode so it resets the signal generator to factory settings after completing the security operation.

Key Entry **None** **Erase** **Overwrite** **Sanitize**

:SECurity:LEVel:STATe

Supported All Models

CAUTION Ensure that you select the security level prior to executing this command with the ON (1) selection. Once you enable the state, you cannot reduce the security level.

```
:SYSTem:SECurity:LEVel:STATe ON|OFF|1|0
:SYSTem:SECurity:LEVel:STATe?
```

This command arms and executes the current security level parameter.

On (1) This selection arms and prevents any changes to the current security level. Refer to ":SECurity:LEVel" on page 110 for setting the security level.

OFF (0) This selection performs the actions required for the current security level setting. Cycling the signal generator power also performs the same function.

For more information about security functions, refer to the *N5181A/82A Agilent MXG Signal Generators User's Guide*.

Example

```
:SYST:SEC:LEV:STAT ON
```

The preceding example arms the secure mode selected with the SYSTem:SECurity:LEVel command.

Key Entry **Enter Secure Mode**

:SECurity:OVERwrite

Supported All Models

```
:SYSTem:SECurity:OVERwrite
```

This command removes all user files, table editor files values, flatness correction files, and baseband generator files. The memory is then overwritten with random data as described below. For more information about security functions, refer to the *N5181A/82A Agilent MXG Signal Generators User's Guide*.

SRAM All addressable locations will be overwritten with random characters.

HARD DISK All addressable locations will be overwritten with random characters.

FLASH MEMORY The flash blocks will be erased.

Key Entry **Erase and Overwrite All**

:SECurity:SANitize

Supported All Models

```
:SYSTem:SECurity:SANitize
```

This command removes all user files, table editor files values, flatness correction files, and baseband generator files. The memory is then overwritten with a sequence of data as described below. For more information about security functions, refer the *User's Guide*.

SRAM	All addressable locations will be overwritten with random characters.
HARD DISK	All addressable locations will be overwritten with a single character and then a random character.
FLASH MEMORY	The flash blocks will be erased.

Key Entry **Erase and Sanitize All**

:SSAVer:DELay

Supported All

```
:SYSTem:SSAVer:DELay <value>
:SYSTem:SSAVer:DELay?
```

This command sets the amount of time before the display light or display light and text is switched off. This will occur if there is no input via the front panel during the delay period.

The variable `<value>` is a whole number measured in hours.

Range 1–12

Key Entry **Screen Saver Delay:**

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or `*RST`.

Refer to ":SSAVer:MODE" on page 112 for selecting the screen saver mode.

:SSAVer:MODE

Supported All

```
:SYSTem:SSAVer:MODE LIGHt|TEXT
:SYSTem:SSAVer:MODE?
```

This command toggles the screen saver mode between light only or light and text.

LIGHt	This choice enables only the light to turn off during the screen saver operation while leaving the text visible on the darkened screen.
TEXT	This choice enables both the display light and text to turn off during the screen saver operation.

Key Entry **Screen Saver Mode**

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or `*RST`.

:SSAVer:STATe

Supported All

```
:SYSTem:SSAVer:STATe ON|OFF|1|0
:SYSTem:SSAVer:STATe?
```

This command enables or disables the display screen saver.

Key Entry **Screen Saver Off On**

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:TIME

Supported All

```
:SYSTem:TIME <hour>,<minute>,<second>
:SYSTem:TIME?
```

This command sets the time displayed in the lower right area of the signal generator's display.

Range <hour>: 0–23 <minute>: 0–59 <second>: 0–59

Key Entry **Time / Date**

:VERSion

Supported All

```
:SYSTem:VERSion?
```

This command returns the SCPI version number with which the signal generator complies.

Unit Subsystem (:UNIT)

:POWer

Supported All

```
:UNIT:POWer DBM|DBUV|DBUVEMF|V|VEMF|DB
:UNIT:POWer?
```

This command terminates an amplitude value in the selected unit of measure.

If the amplitude reference state is set to on, the query returns units expressed in dB and the dB choice will be displayed. Setting any other unit will cause a setting conflict error stating that the amplitude reference state must be set to off. Refer to, ":REFerence:STATe" on page 55 for more information.

***RST** DBM

Key Entry **dBm** **dBuV** **dBuVemf** **V** mVem **dB** f

Remarks All power values in this chapter are shown with dBm as the unit of measure. If a different unit of measure is selected, replace dBm with the newly selected unit whenever it is indicated for the value.

Trigger Subsystem

:ABORt

Supported All

```
:ABORt
```

This command causes the List or Step sweep in progress to abort. If `INIT:CONT[:ALL]` is set to ON, the sweep will immediately re-initiate. The pending operation flag affecting `*OPC`, `*OPC?`, and `*WAI` will undergo a transition once the sweep has been reset.

:INITiate:CONTinuous[:ALL]

Supported All

```
:INITiate:CONTinuous[:ALL] ON|OFF|1|0
:INITiate:CONTinuous[:ALL]?
```

This command selects either a continuous or single list or step sweep. Execution of this command does not affect a sweep in progress.

ON (1) This choice selects continuous sweep where, after the completion of the previous sweep, the current sweep will restart automatically or wait until the appropriate trigger source is received.

OFF (0) This choice selects a single sweep. Refer to ":INITiate[:IMMediate][:ALL]" on page 115 for single sweep triggering information.

***RST** 0

Key Entry **Sweep Repeat Single Cont**

Remarks Execution of this command will not affect a sweep in progress.

:INITiate[:IMMediate][:ALL]

Supported All

```
:INITiate[:IMMediate][:ALL]
```

This command either sets or sets and starts a single List or Step sweep, depending on the trigger type. The command performs the following:

- arms a single sweep when BUS, EXTernal, or KEY is the trigger source selection
- arms and starts a single sweep when IMMediate is the trigger source selection

This command is ignored if a sweep is in progress. See ":INITiate:CONTinuous[:ALL]" on page 115 for setting continuous or single sweep. See ":TRIGger[:SEQuence]:SOURce" on page 116 to select the trigger source.

Key Entry **Single Sweep**

:TRIGger:OUTPut:POLarity

Supported All

```
:TRIGger:OUTPut:POLarity POSitive|NEGative
:TRIGger:OUTPut:POLarity?
```

Sets the TTL signal level present at the TRIGGER OUT connector to either high (5 vdc) or low (0 vdc). The trigger out is asserted after the frequency and/or power is set while the sweep is waiting for its step trigger.

Example

```
:TRIG:OUTP:POL NEG
```

The preceding example sets the trigger out polarity to be low when the trigger is preset.

***RST** POS

Key Entry **Trigger Out Polarity Neg Pos**

:TRIGger[:SEQuence]:SLOPe

Supported All

```
:TRIGger[:SEQuence]:SLOPe POSitive|NEGative
:TRIGger[:SEQuence]:SLOPe?
```

This command sets the polarity of an external signal at the TRIG IN connector that will trigger a list or step sweep.

***RST** POS

Key Entry **Trigger In Polarity Neg Pos**

:TRIGger[:SEQuence]:SOURce

Supported All

```
:TRIGger[:SEQuence]:SOURce BUS|IMMediate|EXTernal|KEY|TIMer
:TRIGger[:SEQuence]:SOURce?
```

This command sets the sweep trigger source for a list or step sweep.

BUS	This choice enables GPIB triggering using the `*TRG` or `GET` command. The `*TRG` SCPI command can be used with any combination of GPIB, LAN, or USB. The `GET` command requires USB, GPIB, or LAN-VXI-11.
IMMediate	This choice enables immediate triggering of the sweep event.
EXTernal	This choice enables the triggering of a sweep event by an externally applied signal at the TRIG IN connector.
Trigger KEY	This choice enables triggering through front panel interaction by pressing the **Trigger** hardkey.
TIMer Trigger	This choice enables the sweep trigger timer.

***RST** IMM

Remarks The wait for the BUS, EXTernal, or KEY trigger can be bypassed by sending the `:TRIGger[:SEQuence][:IMMediate]` command.

Example

```
:TRIG:SOUR BUS
```

The preceding example sets the sweep trigger source to BUS.

***RST** IMM

Key Entry **Bus** **Free Run** **Ext** **Trigger Key** **Timer Trigger**

:TRIGger[:SEQuence]:TIMer

Supported All Models

```
:TRIGger[:SEQuence]:TIMer <period>
:TRIGger[:SEQuence]:TIMer?
```

This command sets the period of the timer trigger.

***RST** 1 ms

Range .5ms–1000s

Key Entry Trig Timer Period

:TRIGger[:SEQuence][:IMMediate]

Supported All Models

```
:TRIGger[:SEQuence][:IMMediate]
```

This event command causes an armed List or Step sweep to immediately start without the selected trigger occurring.

4 Analog Modulation Commands

This chapter provides SCPI descriptions for subsystems dedicated to analog commands common to all Agilent MXG signal generator models. This chapter contains the following major sections:

Amplitude Modulation Subsystem–Option UNT ([:SOURce])

:AM:EXTernal:COUPling

Supported All Models with Option UNT

```
[:SOURce]:AM:EXTernal:COUPling AC|DC
[:SOURce]:AM:EXTernal:COUPling?
```

This command sets the coupling for the amplitude modulation source through the selected external input connector.

AC This choice will only pass ac signal components.

DC This choice will pass both ac and dc signal components.

***RST** DC

Key Entry **Ext Coupling DC AC**

Remarks The command does not change the currently active source or switch the current modulation on or off. The modulating signal may be the sum of several signals, either internal or external sources.

:AM:INTernal:FREQuency

Supported All Models with Option UNT

```
[:SOURce]:AM:INTernal:FREQuency <value><unit>|UP|DOWN
[:SOURce]:AM:INTernal:FREQuency?
```

This command sets the internal amplitude modulation rate for the following applications:

- the start frequency for a swept-sine waveform
- the frequency rate for all other waveforms

***RST** +4.00000000E+002

Range Swept-Sine & Sine: 0.1 Hz–20 MHz

Key Entry **AM Rate**

:AM:INTernal:FREQuency:STEP[:INCRement]

Supported All Models with Option UNT

```
[:SOURce]:AM:INTernal:FREQuency:STEP[:INCRement] <num>
[:SOURce]:AM:INTernal:FREQuency:STEP[:INCRement]?
```

This command sets the step increment for the amplitude modulation internal frequency.

The variable <num> is expressed in units of Hertz.

Range 0.5–1E6

Key Entry **Incr Set**

Remarks The value set by this command is used with the UP and DOWN choices for the AM frequency setting. Refer to ":AM:INTernal:FREQuency" on page 120 for more information.

The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:AM:INTernal:FUNCtion:SHAPe

Supported All Models with Option UNT

```
[:SOURce]:AM:INTernal:FUNCtion:SHAPe SINE
[:SOURce]:AM:INTernal:FUNCtion:SHAPe?
```

This command sets the AM waveform type.

***RST** SINE

:AM:SOURce

Supported All Models with Option UNT

```
[:SOURce]:AM:SOURce INT|EXT
[:SOURce]:AM:SOURce?
```

This command sets the source to generate the amplitude modulation.

INT This choice selects the internal source to provide an ac-coupled signal.

EXT This choice selects the EXT INPUT connector to provide an externally applied signal that can be ac- or dc-coupled.

***RST** INT

Key Entry **Internal** **Ext**

Remarks A 1.0 V_p input is required for calibrated AM depth settings.

The externally applied, ac-coupled input signal is tested for a voltage level and a display annunciator will report a high or low condition if that voltage is > ±3% of 1 V_p.

:AM:STATe

Supported All Models with Option UNT

```
[:SOURce]:AM:STATe ON|OFF|1|0
[:SOURce]:AM:STATe?
```

This command enables or disables the amplitude modulation for the selected path.

***RST** 0

Key Entry **AM Off On**

Remarks The RF carrier is modulated when you have set the signal generator's modulation state to ON, see ":MODulation[:STATe]" on page 89 for more information.

Whenever amplitude modulation is enabled, the AM annunciator is turned on in the display.

:AM:TYPE

Supported All Models with Option UNT

```
[:SOURce]:AM:TYPE LINear|EXPonential
[:SOURce]:AM:TYPE?
```

This command enables LINear or EXPonential amplitude modulation.

The units effected are the AM Depth settings.

LIN This choice selects linear (percent/volt) AM.

EXP This choice selects exponential (db/volt).

***RST** LIN

Key Entry **AM Type LIN EXP**

:AM[:DEPTh]:EXPonential

Supported All Models with Option UNT

```
[:SOURce]:AM[:DEPTh]:EXPonential <value>
[:SOURce]:AM[:DEPTh]:EXPonential?
```

This commands sets the amplitude modulation depth in dB.

***RST** +4.00000000E+001

Range 0–40 dB

Key Entry **AM Depth**

Remarks Refer to ":AM[:DEPTh]:STEP[:INCRement]" on page 122 for setting the value associated with UP and DOWN choices.

:AM[:DEPTh]:STEP[:INCRement]

Supported All Models with Option UNT

```
[:SOURce]:AM[:DEPTh]:STEP[:INCRement] <value><unit>
[:SOURce]:AM[:DEPTh]:STEP[:INCRement]?
```

This command sets the AM depth step increment.

Range 0.1–100%

Key Entry **Incr Set**

Remarks The value set by this command is used with the UP and DOWN choices for the AM depth setting. Refer to ":AM[:DEPTh][:LINear]" on page 123 for more information.

The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:AM[:DEPTh][:LINear]

Supported All Models with Option UNT

```
[:SOURce]:AM[:DEPTh][:LINear] <value><unit>|UP|DOWN
[:SOURce]:AM[:DEPTh][:LINear]?
```

This commands sets the amplitude modulation depth in percent.

***RST** +1.00000000E–001

Range 0.00–90%

Key Entry **AM Depth**

Remarks Refer to ":AM[:DEPTh]:STEP[:INCRement]" on page 122 for setting the value associated with UP and DOWN choices.

Frequency Modulation Subsystem–Option UNT ([:SOURce])

:FM:EXTernal:COUPling

Supported All Models with Option UNT

```
[:SOURce]:FM:EXTernal:COUPling AC|DC
[:SOURce]:FM:EXTernal:COUPling?
```

This command sets the coupling for the frequency modulation source through the selected external input connector.

Use this command with the ":DCFM" on page 60 to remove the effects of DC and optimize the DCFM calibration.

AC This choice only passes ac signal components.

DC This choice passes both ac and dc signal components.

***RST** DC

Key Entry **Ext Coupling DC AC**

Remarks The command does not change the currently active source or switch the current modulation on or off. The modulating signal may be the sum of several signals, either internal or external sources.

:FM:INTernal:FREQuency

Supported All Models with Option UNT

```
[:SOURce]:FM:INTernal:FREQuency <value><unit>|UP|DOWN
[:SOURce]:FM:INTernal:FREQuency?
```

This command sets the internal frequency modulation rate for the following applications:

- the start frequency for a swept-sine waveform
- the frequency rate for all other waveforms

***RST** +4.00000000E+002

Range All Waveforms: 0.1 Hz–2 MHz

Key Entry **FM Rate**

:FM:INTernal:FREQuency:STEP[:INCRement]

Supported All Models with Option UNT

```
[:SOURce]:FM:INTernal:FREQuency:STEP[:INCRement] <num>
[:SOURce]:FM:INTernal:FREQuency:STEP[:INCRement]?
```

This command sets the step increment for the internal frequency modulation.

The variable <num> sets the entered value in units of Hertz.

***RST** +5.00000000E+002

Range 0.5–1E6

Key Entry **Incr Set**

Remarks The value set by this command is used with the UP and DOWN choices for the FM frequency setting. Refer to ":FM:INTernal:FREQuency" on page 124 for more information.

The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:FM:INTernal:FUNCtion:SHAPe

Supported All Models with Option UNT

```
[:SOURce]:FM:INTernal:FUNCtion:SHAPe SINE
[:SOURce]:FM:INTernal:FUNCtion:SHAPe?
```

This command sets the FM waveform type.

***RST** SINE

Remarks The waveform selection is only valid when INT is the source selection. Refer to ":FM:SOURce" on page 125 for type source selection.

:FM:SOURce

Supported All Models with Option UNT

```
[:SOURce]:FM:SOURce INT|EXT
[:SOURce]:FM:SOURce?
```

This command sets the source to generate the frequency modulation.

INT This choice selects the internal source to provide an ac-coupled signal.

EXT This choice selects the FM rear panel connector to provide an externally applied signal that can be ac- or dc-coupled.

***RST** INT

Key Entry **Internal** **Ext**

Remarks The externally applied, ac-coupled input signal is tested for a voltage level and a display annunciator will report a high or low condition if that voltage is > ±3% of 1 V_p.

:FM:STATe

Supported All Models with Option UNT

```
[:SOURce]:FM:STATe ON|OFF|1|0
[:SOURce]:FM:STATe?
```

This command enables or disables the frequency modulation for the selected path.

***RST** 0

Key Entry **FM Off On**

Remarks The RF carrier is modulated when you set the signal generator's modulation state to ON, see ":MODulation[:STATe]" on page 89 for more information.

Whenever frequency modulation is enabled, the FM annunciator is turned on in the display.

:FM[:DEViation]

Supported All Models with Option UNT

```
[:SOURce]:FM[:DEViation] <value><unit>
[:SOURce]:FM[:DEViation]?
```

This command sets the frequency modulation deviation.

***RST** +1.00000000E+003

Range

Frequency	*Deviation*
<250 MHz[a]	0–20 MHz
250 MHz – <375 MHz	0–2.5 MHz
375 MHz – <750 MHz	0–5 MHz
750 MHz – <1.5 GHz	0–10 MHz
1.5 GHz – <3.000001 GHz	0–20 MHz
3.000001 GHz – 6 GHz	0–40 MHz

a. Settable, but not specified to 100 kHz.

Key Entry **FM DEV**

:FM[:DEViation]:STEP[:INCRement]

Supported All Models with Option UNT

```
[:SOURce]:FM[:DEViation]:STEP[:INCRement] <value><unit>|GHz|MHz|kHz|Hz
[:SOURce]:FM[:DEViation]:STEP[:INCRement]?
```

This command sets the step increment for the FM deviation of the signal generator.

***RST** +5.00000000E+003

Key Entry **Incr Set**

Remarks The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

Phase Modulation Subsystem–Option UNT ([:SOURce])

:PM:BANDwidth|BWIDth

Supported All Models with Option UNT

```
[:SOURce]:PM:BANDwidth|BWIDth NORMal|HIGH
[:SOURce]:PM:BANDwidth|BWIDth?
```

This command toggles between normal phase modulation and high bandwidth phase modulation mode.

***RST** NORM

Key Entry **FM ΦM Normal High BW**

:PM:EXTernal:COUPling

Supported All Models with Option UNT

```
[:SOURce]:PM:EXTernal:COUPling AC|DC
[:SOURce]:PM:EXTernal:COUPling?
```

This command sets the coupling for the phase modulation source through the selected external input connector.

Use this command with the ":DCFM" on page 60 to remove the effects of DC and optimize the DCFM calibration.

AC This choice will only pass ac signal components.

DC This choice will pass both ac and dc signal components.

***RST** DC

Key Entry **Ext Coupling DC AC**

Remarks This command does not change the currently active source or switch the current modulation on or off. The modulating signal may be the sum of several signals, either internal or external sources.

:PM:INTernal:FREQuency

Supported All Models with Option UNT

```
[:SOURce]:PM:INTernal:FREQuency <value><unit>|UP|DOWN
[:SOURce]:PM:INTernal:FREQuency?
```

This command sets the internal modulation frequency rate for the following applications:

- the start frequency for a swept-sine waveform
- the frequency rate for all other waveforms

***RST** +4.00000000E+002

Range All Waveforms: 0.1 Hz–2 MHz (Wideband)All Waveforms: 0.1 Hz–1 MHz (narrowband)

Key Entry **ΦM Rate**

:PM:INTernal:FREQuency:STEP[:INCRement]

Supported All Models with Option UNT

```
[:SOURce]:PM:INTernal:FREQuency:STEP[:INCRement] <num>
[:SOURce]:PM:INTernal:FREQuency:STEP[:INCRement]?
```

This command sets the step increment of the phase modulation internal frequency.

The variable <num> sets the entered value in units of Hertz.

Range 0.5–1E6

Key Entry **Incr Set**

Remarks The value set by this command is used with the UP and DOWN choices for the FM frequency command. Refer to ":PM:INTernal:FREQuency" on page 127 for more information.

The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

:PM:INTernal:FUNCtion:SHAPe

Supported All Models with Option UNT

```
[:SOURce]:PM:INTernal:FUNCtion:SHAPe SINE
[:SOURce]:PM:INTernal:FUNCtion:SHAPe?
```

This command sets the phase modulation waveform type.

***RST** SINE

:PM:SOURce

Supported All Models with Option UNT

```
[:SOURce]:PM:SOURce INT|EXT
[:SOURce]:PM:SOURce?
```

This command sets the source to generate the phase modulation.

INT This choice selects internal source 1 to provide an ac-coupled signal.

EXT This choice selects the FM rear panel connector to provide an externally applied signal that can be ac- or dc-coupled.

***RST** INT

Key Entry **Internal 1** **Ext1**

Remarks The externally applied, ac-coupled input signal is tested for a voltage level and a display annunciator will report a high or low condition if that voltage is > ±3% of 1 V_p.

:PM:STATe

Supported All Models with Option UNT

```
[:SOURce]:PM:STATe ON|OFF|1|0
[:SOURce]:PM:STATe?
```

This command enables or disables the phase modulation for the selected path.

***RST** 0

Key Entry **ΦM Off On**

Remarks The RF carrier is modulated when you set the signal generator's modulation state to ON, see ":MODulation[:STATe]" on page 89 for more information.

Whenever phase modulation is enabled, the ΦM annunciator is turned on in the display

:PM[:DEViation]

Supported All Models with Option UNT

```
[:SOURce]:PM[:DEViation] <value><unit>|UP|DOWN
[:SOURce]:PM[:DEViation]?
```

This command sets the deviation of the phase modulation.

The variable <unit> will accept RAD (radians), PIRAD (pi-radians), and DEG (degrees); however, the query will only return values in radians.

***RST** +0.00000000E+000

Range

Frequency	*Normal Bandwidth*	*High Bandwidth*
<250 MHz[a]	0–5 RAD	0–0.500 RAD
250 MHz – <375 MHz	0–1.25 RAD	0–0.125 RAD
375 MHz – <750 MHz	0–2.5 RAD	0–0.2500 RAD
750 MHz – <1.5 GHz	0–5 RAD	0–0.500 RAD
1.5 GHz – <3.000001 GHz	0–10 RAD	0–1 RAD
> 3 GHz – 6 GHz	0–20 RAD	0–2 RAD

a. Settable, but not specified to 100 kHz. For more information on specifications, refer to the *Data Sheet*.

Key Entry **ΦM Dev**

Remarks Refer to ":PM[:DEViation]:STEP[:INCRement]" on page 130 for setting the value associated with the UP and DOWN choices.

:PM[:DEViation]:STEP[:INCRement]

Supported All Models with Option UNT

```
[:SOURce]:PM[:DEViation]:STEP[:INCRement] <value><unit>
[:SOURce]:PM[:DEViation]:STEP[:INCRement]?
```

This command sets the phase modulation deviation step increment.

Range 0.001–1E3RAD

Key Entry **Incr Set**

Remarks The value set by this command is used with the UP and DOWN choices for the FM deviation command. Refer to ":PM[:DEViation]" on page 129 for more information.

The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

Pulse Modulation Subsystem–Option UNU and UNW([:SOURce])

:PULM:EXTernal:POLarity

Supported All with Option UNU and UNW

```
[:SOURce]:PULM:EXTernal:POLarity NORMal|INVerted
[:SOURce]:PULM:EXTernal:POLarity?
```

This command selects the polarity of the TTL input signal at the TRIG IN rear panel connector. The signal generator can respond to either a normal (a TTL high) or an inverted (TTL low) signal.

Example

```
:PULM:EXT:POL NORM
```

The preceding example selects normal (TTL high) polarity.

***RST** Normal

Key Entry **Ext Polarity Normal Inverted**

:PULM:INTernal[1]:DELay:STEP

Supported All with Option UNU and UNW

```
[:SOURce]:PULM:INTernal[1]:DELay:STEP <num><time_suffix>
[:SOURce]:PULM:INTernal[1]:DELay:STEP?
```

This command sets the step increment for the pulse delay.

The step value, set by this command, is used with the UP and DOWN choices in the ":PULM:INTernal[1]:DELay[1]|2" on page 131 command.

The step value set with this command is not affected by a signal generator power-on, preset, or *RST command.

Example

```
:PULM:INT:DEL:STEP 10NS
```

The preceding example sets the pulse delay step value to 10 nanoseconds.

Range 10nS to (pulse period – 20 nS)

Key Entry **Incr Set**

:PULM:INTernal[1]:DELay[1]|2

Supported All with Option UNU and UNW

```
[:SOURce]:PULM:INTernal[1]:DELay[1]|[2] <num><time_suffix>|UP|DOWN
[:SOURce]:PULM:INTernal[1]:DELay[1]|[2]
```

This command sets the pulse delay for the internally-generated pulse modulation using the variable <num>[<time_suffix>]. The command, used with the UP|DOWN parameters, will change the delay by a user-defined step value. Refer to the :PULM:INTernal[1]:DELay:STEP command on page 131 for setting the value associated with the UP and DOWN choices.

The optional variable <time_suffix> accepts nS (nanoseconds) to S (seconds).

The range value is dependent on the pulse period. Refer to ":PULM:INTernal[1]:PERiod" on page 133 for pulse period settings.

Use Delay1 with the DOUBlet parameter and Delay1 and Delay2 with the ADOublet parameter (refer to ":PULM:Source:INTernal" on page 135).

Example

```
:PULM:INT:DEL 200E-9
```

The preceding example sets the internal pulse delay to 200 nanoseconds.

***RST** +0.00000000E+000

Range *Internal Free Run*: depends on pulse period and pulse width settings

Internal Triggered, Adjustable Doublet, & Triggered Doublet: 70nS to (42 S - 10 nS - pulse width)

Key Entry **Pulse Delay**

:PULM:INTernal[1]:FREQuency

Supported All with Option UNU and UNW

```
[:SOURce]:PULM:INTernal[1]:FREQuency <frequency>|MAXimum|MINimum|UP|DOWN
[:SOURce]:PULM:INTernal[1]:FREQuency?
```

This command sets the pulse rate for the internally-generated square wave using the variable <frequency>. The command, used with the UP|DOWN parameters, will change the frequency by a user-defined step value. Refer to the :PULM:INTernal[1]:FREQuency:STEP command for setting the value associated with the UP and DOWN choices.

This command is used when SQUare is the pulse modulation type. Refer to ":PULM:SOURce" on page 135 for the pulse modulation type selection.

Example

```
:PULM:INT:FREQ 1MHz
```

The preceding example sets the square wave pulse rate to 1 megahertz.

***RST** +4.00000000E+002

Range 0.1Hz–10MHz

Key Entry **Pulse Rate**

:PULM:INTernal[1]:FREQuency:STEP

Supported All with Option UNU and UNW

```
[:SOURce]:PULM:INTernal[1]:FREQuency:STEP[:INCRement] <freq>|MAXimum|MINimum|DEFault
[:SOURce]:PULM:INTernal[1]:FREQuency:STEP[:INCRement]?
```

This command sets the step value for the internally-generated square wave pulse rate.

This command is used when SQUare is the pulse modulation type. Refer to ":PULM:SOURce" on page 135 for the pulse modulation type selection. Refer to ":PULM:SOURce" on page 135 for the pulse modulation type selection. The step value, set with this command, is used with the UP and DOWN choices in the :PULM:INTernal[1]:FREQuency command.

The step value set with this command is not affected by a power-on, preset, or *RST command.

Example

```
:PULM:INT:FREQ:STEP MIN
```

The preceding example sets the step value for the square wave pulse rate to 0.1 Hz, the minimum rate.

Range 0.1Hz–10MHz

:PULM:INTernal[1]:PERiod

Supported All with Option UNU and UNW

```
[:SOURce]:PULM:INTernal[1]:PERiod <period>|MAXimum|MINimum|UP|DOWN
[:SOURce]:PULM:INTernal[1]:PERiod?
```

This command sets the pulse period for the internally–generated pulse modulation using the variables <value><units>. The command, used with the UP|DOWN parameters, will change the pulse period by a user–defined step value. Refer to the :PULM:INTernal[1]:PERiod:STEP[:INCRement] command for setting the value associated with the UP and DOWN choices.

If the entered value for the pulse period is equal to or less than the value for the pulse width, the pulse width changes to a value that is less than the pulse period. Refer to ":PULM:INTernal[1]:PWIDth[1]|2" on page 134 for setting the pulse width.

Example

```
:PULM:INT:PER .5S
```

The preceding example sets the period of the internally–generated pulse to 500 milliseconds.

***RST** +4.00000000E–006

Range 30nS–42S

Key Entry **Pulse Period**

:PULM:INTernal[1]:PERiod:STEP[:INCRement]

Supported All with Option UNU

```
[:SOURce]:PULM:INTernal[1]:PERiod:STEP[:INCRement] <step>|UP|DOWN
[:SOURce]:PULM:INTernal[1]:PERiod:STEP[:INCRement]?
```

This command sets the step value for the internal pulse period using the variable <value><units>.

The step value, set with this command, is used with the UP and DOWN choices available in the :PULM:INTernal[1]:PERiod command.

The step value set with this command is not affected by a power-on, preset, or *RST command.

Example

```
:PULM:INT:PER:STEP .1S
```

The preceding example sets the square wave pulse rate to 100 milliseconds.

Range 30nS–42S

:PULM:INTernal[1]:PWIDth:STEP

Supported All with Option UNU

```
[:SOURce]:PULM:INTernal[1]:PWIDth:STEP <num><time_suffix>|MAXimum|MINimum|DEFault
[:SOURce]:PULM:INTernal[1]:PWIDth:STEP?
```

This command sets the step increment for the pulse width using the variable <num><time_suffix>.

The step value, set by this command, is used with the UP and DOWN choices available in the :PULM:INTernal[1]:PWIDth[1]|2command.

The step value, set with this command, is not affected by a power-on, preset, or *RST command.

Example

```
:PULM:INT:PWID:STEP 100NS
```

The preceding example sets the pulse width step to 100 nanoseconds.

Range 20nS to (pulse period - 10 nS)

:PULM:INTernal[1]:PWIDth[1]|2

Supported All with Option UNU and UNW

```
[:SOURce]:PULM:INTernal[1]:PWIDth[1]|2 <num><time_suffix>|UP|DOWN
[:SOURce]:PULM:INTernal[1]:PWIDth[1]|2?
```

This command sets the pulse width for the internally generated pulse signal.

This command sets the pulse width for the internally-generated pulse modulation using the variables <num><time_suffix>. The command, used with the UP|DOWN parameters, will change the pulse width by a user-defined step value. Refer to the :PULM:INTernal[1]:PWIDth:STEP command for setting the value associated with the UP and DOWN choices.

If the entered value for the pulse width is equal to or greater than the value for the pulse period, the pulse width changes to a value that is less than the pulse period. For more information, refer to the command ":PULM:INTernal[1]:PERiod" on page 133.

Use PWIDTH1 with the DOUBlet parameter and PWIDTH1 and PWIDTH2 with the ADOublet parameter (refer to ":PULM:Source:INTernal" on page 135).

NOTE A power search is recommended for signals with pulse widths less than one microsecond. Refer to ":ALC:SEARch" on page 50.

Example

```
:PULM:INT:PWIDth 100MS
```

The preceding example sets the pulse width to 100 milliseconds.

***RST** +2.00000000E–006

Range 20nS to (pulse period - 10 nS)

Key Entry **Pulse Width**

:PULM:SOURce

Supported All with Option UNU

```
[:SOURce]:PULM:SOURce?
```

This query returns the source of pulse modulation.

The INTernal selection accesses one of the six internally generated modulation inputs while EXTernal selects an external pulse (Ext Pulse) input. To select an internally generated modulation input, refer to ":PULM:Source:INTernal" on page 135.

:PULM:Source:INTernal

Supported All with Option UNU and UNW

```
[:SOURce]:PULM:SOURce:INTernal SQUare|FRUN|TRIGgered|ADOublet|DOUBlet|GATEd
[:SOURce]:PULM:SOURce:INTernal?
```

This command selects one of the six internally generated modulation inputs. There is one external source: Ext Pulse selected by ":PULM:SOURce" on page 135.

User Delay[1] with the DOUBlet parameter and Delay1 and Delay2 with the ADOublet parameter.

Example

```
:PULM:SOUR:INT SQU
```

The preceding example selects the internally–generated square wave pulse modulation format.

***RST** FRUN (Int Free–Run)

Key Entry **Square** **Free-Run** **Triggered** **Adjustable Doublet** **Trigger Doublet** **Gated**

:PULM:STATe

Supported All with Option UNU

```
[:SOURce]:PULM:STATe ON|OFF|1|0
[:SOURce]:PULM:STATe?
```

This command enables or disables pulse modulation for the selected path.

When pulse modulation is enabled, the PULSE annunciator appears on the signal generator's front-panel display.

Example

```
:PULM:STAT ON
```

The preceding example enables the pulse modulation.

***RST** 0

Key Entry **Pulse Off On**

5 Component Test Digital Commands

This chapter provides SCPI descriptions for commands dedicated to digital component testing using the N5182A Agilent MXG Vector Signal Generator. This chapter contains the following major sections:

All Subsystem–Option 651/652/654 ([:SOURce])

:RADio:ALL:OFF

Supported N5182A with Option 651/652/654

`[:SOURce]:RADio:ALL:OFF`

This command turns off all digital modulation formats.

Remarks This command does not affect analog modulation.

AWGN Real-Time Subsystem–Option 403 ([:SOURce]:RADio:AWGN:RT)

:BWIDth

Supported N5182A with Option 403

```
[:SOURce]:RADio[1]:AWGN:RT:BWIDth <value>
[:SOURce]:RADio[1]:AWGN:RT:BWIDth?
```

This command adjusts the flat bandwidth of the real-time AWGN waveform.

The variable <value> is expressed in units of Hertz (Hz–MHz).

***RST** +1.00000000E+006

Range 1–1.0E8

Key Entry **Bandwidth**

:IQ:MODulation:ATTen

Supported All with Option 403

```
[:SOURce]:RADio:AWGN:RT:IQ:MODulation:ATTen <value>
[:SOURce]:RADio:AWGN:RT:IQ:MODulation:ATTen?
```

This command attenuates the I/Q signals being modulated through the signal generator's RF path.

The variable <value> is expressed in units of decibels (dB).

***RST** Varies (box dependent)

Range 0–40

Key Entry **Modulator Atten Manual Auto**

:IQ:MODulation:ATTen:AUTO

Supported All with Option 403

```
[:SOURce]:RADio:AWGN:RT:IQ:MODulation:ATTen:AUTO ON|OFF|1|0
[:SOURce]:RADio:AWGN:RT:IQ:MODulation:ATTen:AUTO?
```

This command enables or disables the I/Q attenuation auto mode.

ON (1) This choice enables the attenuation auto mode which optimizes the modulator attenuation for the current conditions.

OFF (0) This choice holds the attenuator at its current setting or at a selected value. Refer to ":IQ:MODulation:ATTen" on page 139 for setting the attenuation value.

***RST** 1

Key Entry **Modulator Atten Manual Auto**

[:STATe]

Supported N5182A with Option 403

```
[:SOURce]:RADio:AWGN[1]:RT[:STATe] ON|OFF|1|0
[:SOURce]:RADio:AWGN[1]:RT[:STATe]?
```

This command enables or disables the operating state of real-time AWGN.

***RST** 0

Key Entry **Real-time AWGN Off On**

Dual ARB Subsystem–Option 651/652/654 ([:SOURce]:RADio[1]:ARB)

:BASEband:FREQuency:OFFSet

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:BASeband:FREQuency:OFFSet <value>
[:SOURce]:RADio[1]:ARB:BASeband:FREQuency:OFFSet?
```

Makes Baseband Freq Offset the active function. The value that you enter sets the frequency offset in the signal at baseband. The frequency offset shifts the signal off of the modulated RF carrier by the offset amount.

***RST** <*value*>: 0 Hz

Range <*value*>: –50 to 50 MHz

Key Entry **Baseband Frequency Offset**

Remarks This feature is useful for moving the signal such that the carrier feed through is not in the center of the signal.

:CLIPping

Supported N5182A with Option 651/652/654

NOTE Clipping cannot be undone (i.e. restoring clipping value to 100% will have no effect on a previously clipped waveform.)

```
[:SOURce]:RADio[1]:ARB:CLIPping "<file name>",IJQ|IORQ,<value>[,<value>]
```

This command sets the clipping level of the selected waveform segment to a percentage of its highest peak.

The variable <value> is expressed in units of percent.

IJQ This choice clips the composite I/Q waveform.

IORQ This choice clips I and Q separately. When this choice is enabled, percentage values for both I and Q must be specified.

***RST** IJQ <*value*>: +100

Range <*value*>: 10–100 (0.1% resolution)

Key Entry **Clipping Type |I+jQ| |I|,|Q|**

Remarks A value of 100 percent equates to no clipping.

Refer to "File Name Variables" on page 12 for information on the file name syntax.

:GENerate:SINE

Supported N5182A with Option 651, 652 or 654

```
[:SOURce]:RADio[1]:ARB:GENerate:SINE ["<file_name>"],[<osr>],[<scale>],
[I|Q|IQ]
```

This command creates a sine wave waveform file and saves it in the signal generator's volatile waveform memory (WFM1).

"<file_name>" This variable names the file used to save the generated sine wave data. If no ["<file_name>"] parameter is defined, the default file_name is: "SINE_TEST_WFM".

<osr> This variable sets the oversample ratio, which must be an even number and ≥ 4. The <osr> variable is expressed in samples. If the oversample ratio is < 60 (the minimum number of samples or I/Q points required for a waveform), multiple waveform periods are generated to create a waveform file with ≥ 60 samples. The number of periods created is 60 ÷ <osr> (quotient will round up to an integer value). A waveform with an oversample ratio ≥ 60 has one period. If no ["<osr>"] parameter is defined, the default value is: 200.

<scale> This variable sets the scale factor for the waveform. The scale factor is a real number from zero to one. If no ["<scale>"] parameter is defined, the default value is: 1.

I|Q|IQ Selects I, Q, or I and Q paths for the waveform data. Sinewave data is generated and applied to the I path if the I path is selected; Q data are set to zeros. Sine data is generated and applied to the Q path if the Q path is selected; I data are set to zeros. If the I and Q paths are selected, sinewave data are applied to the I and Q paths.

Example

```
:RAD:ARB:GEN:SINE "Sine_Wave",60,.5,IQ
```

The preceding example generates an I/Q sine wave and saves the data to a file named Sine_Wave. The oversampling ratio is 60, the scaling is set for 50%, and the data is applied to both the I and Q paths.

The signal generator's baseband option and available baseband memory determine the maximum number of samples for the waveform.

Range *Standard:* 4E0 – 8E6

OSR Option 019: 4E0 – 64E6

Scale: 0–1

:HEADer:CLEar

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:HEADer:CLEar
```

This command clears the header information from the file header used by this modulation format (i.e. all file header fields are set to unspecified).

Key Entry **Clear Header**

Remarks A waveform must be selected for this command to function.

:HEADER:NOISe:RMS[:OVERride]

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:HEADER:NOISe:RMS:OVERride "<file_name>",<value>|UNSPecified
[:SOURce]:RADio[1]:ARB:HEADER:NOISe:RMS:OVERride? "<file_name>"
```

This command sets the value of the waveform's I and Q RMS (root mean square) for noise.

The RMS is used strictly for calculating the relative power of the noise in the currently displayed header. The RMS is specified in normalized linear units with |+1| or |–1| as full scale on I or Q, therefore the largest RMS that can be specified is the square root of 2 (1.414213562). If the value is unspecified, then the waveform file header's RMS is used.

This value is useful if you wish to have the noise be relative to only a portion of the waveform, such as a pilot channel, or be relative to only a single carrier that is mixed with other carriers.

For setting the header's RMS value, see ":HEADer:RMS" on page 143.

"<file_name>" This variable names the waveform file to which the RMS value will be applied. The file name variable can designate a file in the WFM1, NVWFM, or SEQ directories. For information on the file name syntax, refer to "File Name Variables" on page 12.

<value> This variable is the user-measured RMS noise value for the specified carrier.

`UNSPecified` Sets RMS as unspecified, which causes the general RMS value to be used for calculating the relative noise power.

Example

```
:RAD:ARB:HEADER:NOISe:RMS:OVER "WFM1:Sine_Wave",.835
```

The preceding example sets the file header RMS noise override value for a file type WFM1, named Sine_Wave, to .835.

```
:RAD:ARB:HEADER:NOISe:RMS:OVER "WFM1:Sine_Wave",UNSP
```

In the second example, the signal generator calculates the RMS, using the waveform file header's RMS value. For setting the header's RMS value, see ":HEADer:RMS" on page 143.

The RMS value is expressed in volts.

Key Entry **Edit Noise RMS Overide Unspecified Enter**

:HEADer:RMS

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:HEADER:RMS "<file_name>",<value>|UNSPecified
[:SOURce]:RADio[1]:ARB:HEADER:RMS? "<file_name>"
```

This command sets the file header RMS value for the selected waveform file. The Agilent MXG uses the RMS value with the dual ARB's real-time noise function and to optimize the modulator drive level.

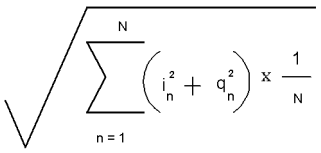
The signal generator reads the RMS value from the file header when real-time noise is enabled and the dual ARB is turned on. If the value is unspecified, then it is calculated and stored in the header automatically.

When the waveform file is saved from volatile waveform memory (WFM1) to non-volatile waveform memory (NVWFM), the RMS value, auto-calculated or user-defined, is also saved.

For setting the header noise carrier RMS override value, see ":HEADER:NOISe:RMS[:OVERride]" on page 143.

"<file_name>" This variable names the waveform file to which the RMS value will be applied. The file name variable can designate a file in the WFM1, NVWFM, or SEQ directories. For information on the file name syntax, refer to "File Name Variables" on page 12.

<value> This variable is the user-measured RMS value for the specified waveform. The following figure shows the RMS calculation.



`UNSPecified` Using this variable in the command clears the RMS value and sets it to unspecified. An unspecified RMS value causes the signal generator to calculate the value when the ARB personality is turned on. The RMS calculation includes rise/fall times and does not include consecutive zero level samples. DC offsets and noise are also included in the RMS measurement. Because the signal generator calculation uses so many parameters, you may achieve better results calculating your own RMS value.

Examples

```
[:SOURce]:RADio[1]:ARB:HEADER:RMS "WFM1:Sine_Wave",.835
```

The first example shows a user-measured RMS value for the Sine_Wave waveform file in the waveform's file header.

```
:RAD:ARB:HEADER:RMS "WFM1:Sine_Wave",UNSP
```

In the second example, the signal generator calculates the RMS value when the ARB is turned on with this file selected or a sequence which contains the file selected.

The RMS value is expressed in volts.

Range 0 – 1.414213562373095

Key Entry **Edit RMS** Enter **Unspecified** **Calculate**

:HEADer:SAVE

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:HEADer:SAVE
```

This command saves the header information to the file header used by this modulation format.

Key Entry **Save Setup To Header**

Remarks A waveform must be selected for this command to function.

:IQ:MODulation:ATTen

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:IQ:MODulation:ATTen <value>
[:SOURce]:RADio[1]:ARB:IQ:MODulation:ATTen?
```

This command sets the attenuation level of the I/Q signals being modulated through the signal generator RF path.

The variable <value> is expressed in units of decibels (dB).

***RST** Varies (box dependent)

Range 0–40

Key Entry **Modulator Atten Manual Auto**

:IQ:MODulation:ATTen:AUTO

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:IQ:MODulation:ATTen:AUTO ON|OFF|1|0
[:SOURce]:RADio[1]:ARB:IQ:MODulation:ATTen:AUTO?
```

This command enables or disables the I/Q attenuation auto mode.

ON (1) This choice enables the attenuation auto mode which optimizes the modulator attenuation for the current conditions.

OFF (0) This choice holds the attenuator at its current setting or at a selected value. Refer to ":IQ:MODulation:ATTen" on page 145 for setting the attenuation value.

***RST** 1

Key Entry **Modulator Atten Manual Auto**

:MARKer:CLEar

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:MARKer:CLEar "<file_name>",<marker>,<first_point>,
<last_point>
```

This command clears a single marker point or a range of marker points on a waveform segment for the selected marker (1–4). The dual ARB player and all of the ARB modulation formats use this command.

"<file_name>" This variable specifies the name of the waveform file in volatile waveform memory (WFM1). For information on the file name syntax, see "File Name Variables" on page 12.

<marker> This variable selects the marker number; an integer value from one to four.

<first_point> This variable defines the first point in a range of points. The number must be greater than or equal to one, and less than or equal to the total number of waveform points.

If you enter a value for either the first marker point or the last marker point that would make the first marker point occur after the last, the last marker point automatically adjusts to match the first marker point.

<last_point> This variable defines the last point in a range of points. The number must be greater than or equal to the first point, and less than or equal to the total number of waveform points.

To clear a single marker point, use the same marker point for the first and last point variables. For more information on markers and ARB files, refer to the *User's Guide*.

Example

```
:RAD:ARB:MARK:CLE "Test_Data",1,1,300
```

The preceding example clears marker 1 from the first point through the 300th point in the Test_Data file.

Range <*marker*>: 1–4

<*first_Point*>: 1–number of waveform points

<*last_point*>: <first_Point>–number of waveform points

Key Entry **Set Marker Off Range Of Points** **Marker 1 2 3 4** **First Mkr Point** **Last Mkr Point**

:MARKer:CLEar:ALL

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:MARKer:CLEar:ALL "<file_name>",<marker>
```

This command clears all marker points on a waveform segment for the selected marker (1–4). The dual ARB player and all of the ARB formats use this command. With all marker points cleared, the event output signal level is set low.

"<file_name>" This variable specifies the name of the waveform file in volatile waveform memory (WFM1). For information on the file name syntax, see "File Name Variables" on page 12.

<marker> This variable selects the marker number; an integer value from one to four.

Example

```
:RAD:ARB:MARK:CLE:ALL "Test_Data",1
```

The preceding example clears marker 1 from the all waveform points in the Test_Data file.

Range 1–4

Key Entry **Marker 1 2 3 4** **Set Marker Off All Points**

:MARKer:ROTate

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:MARKer:ROTate "<file_name>",<rotate_count>
```

This command shifts the marker points for all markers in a waveform segment earlier or later by the value of the <rotate_count> variable. The dual ARB player and all of the ARB formats use this command.

You can use a positive or negative value. When a marker point is close to the end of the waveform and the <rotate_count> value is greater than the number of remaining marker points, but less than the total number of marker points, the marker points that would move beyond the end of the waveform wrap to the beginning of the waveform. For example, if a marker point resides at sample point 195 out of 200, and the <rotate_count> value is twenty-five, the marker point wraps to the beginning of the waveform and continues out to the twentieth waveform point.

To set the marker points in a waveform, refer to ":MARKer[:SET]" on page 147.

"<file_name>" This variable specifies the name of the waveform file in volatile waveform memory (WFM1). For information on the file name syntax, see "File Name Variables" on page 12.

Example

```
:RAD:ARB:MARK:ROT "Test_Data",100
```

The preceding example shifts all markers set in the Test_Data file 100 points later. If the first set point in the file is at 50, then after sending this command, the first set point will be 150 (assuming the Test_Data file has at least 150 points and no later set points wrapped around to the beginning of the file).

Range – (n – 1) to (n – 1)

n = number of points in the waveform

:MARKer[:SET]

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:MARKer[:SET] "<file_name>",<marker>,<first_point>,
<last_point>,<skip_count>
```

This command sets a single marker point or a range of marker points on a waveform segment for the selected marker (1–4). The dual ARB player and all of the ARB formats use this command.

The Agilent MXG provides four independent markers. Two of the markers route output signals to rear-panel event connectors, Marker-1 to Event1 BNC and Marker-2 to Aux I/O. A marker consists of

marker points placed at defined sample points in a waveform segment. This means that a marker point cannot be less than one or greater than the last sample point in the waveform. Marker points are cumulative, so multiple command executions with different range values, without first clearing the existing points, places additional marker points on the waveform. Because of this cumulative behavior, it is a good practice to clear existing marker points prior to setting new points. This will eliminate unexpected marker pulses. Refer to ":MARKer:CLEar" on page 146 and ":MARKer:CLEar:ALL" on page 146 for information on clearing marker points.

For waveforms generated on the signal generator (baseband generator), the Agilent MXG automatically places a marker point at the first waveform sample for markers one and two.

NOTE You can set markers for either positive or negative polarity. The following discussions for this command assume positive marker polarity. When using negative marker polarity, the marker pulses occur during the periods of no marker points.

There are three ways to place marker points using this command:

- consecutive marker points over a range that collectively create a single marker pulse that spans the range
- equally spaced marker points over a range, so that a marker pulse occurs at each sample point that coincides with a marker point (Using this method, you can configure a clock signal by setting the <skip_count> variable to one.)
- a single marker point placed at a specific sample point in the waveform, which outputs a single pulse relative to the marker point location (To configure a single marker point, set the first and last points to the same number.)

For more information on markers, refer to the *User's Guide*.

The following list describes the command variables:

"<file_name>" This variable specifies the name of the waveform file in volatile waveform memory (WFM1). For information on the file name syntax, see "File Name Variables" on page 12.

<marker> This variable selects the marker number; an integer value from one to four.

<first_point> This variable defines the first point in the range over which the marker is placed. This number must be greater than or equal to one, and less than or equal to the total number of waveform points.

If you enter a value for either the first marker point or the last marker point that would make the first marker point occur after the last, the last marker point is automatically adjusted to match the first marker point.

<last_point> This variable defines the last point in the range over which the marker will be placed. This value must be greater than or equal to the first point, and less than or equal to the total number of waveform points.

<skip_count> This variable defines the marker point pattern across the range. A zero value means the marker points occur consecutively across the range. A value greater than zero creates a repeating marker point pattern across the range, where the gap between the marker points is equal to the <skip_count> value. The gaps begin after the first marker point. Each marker point in the pattern, which is only one point wide, produces a marker pulse.

Example

```
:RAD:ARB:MARK "Test_Data",1,40,100,2
```

The preceding example sets marker 1 on the first point, 40, the last point, 100, and every third point (skip 2) between 40 and 100 (assuming the Test_Data file has at least 100 points).

Range <*marker*>: 1–4

<*first_Point*>: 1–number of waveform points

<*last_point*>: <first_Point>–number of waveform points

<*skip_count*>: 0–number of points in the range

Key Entry **Set Marker on Range Of Points** **Marker 1 2 3 4** **First Mkr Point** **Last Mkr Point** **# Skipped Points** **Apply to Waveform**

:MDEStination:ALCHold

Supported N5182A with Option 651/652/654

CAUTION Incorrect automatic level control (ALC) sampling can create a sudden unleveled condition that may create a spike in the RF output potentially damaging a DUT or connected instrument. Ensure that you set markers to let the ALC sample over an amplitude that accounts for the high power levels within the signal.

```
[:SOURce]:RADio[1]:ARB:MDEStination:ALCHold NONE|M1|M2|M3|M4
[:SOURce]:RADio[1]:ARB:MDEStination:ALCHold?
```

This command enables or disables the marker ALC hold function for the selected marker. For setting markers, see ":MARKer[:SET]" on page 147.

Use the ALC hold function when you have a waveform signal that incorporates idle periods, or when the increased dynamic range encountered with RF blanking is not desired. The ALC leveling circuitry responds to the marker signal during the marker pulse (marker signal high), averaging the modulated signal level during this period.

The ALC hold function operates during the low periods of the marker signal. The marker polarity determines when the marker signal is high. For a positive polarity, this is during the marker points. For a negative polarity, this is when there are no marker points. For setting a marker's polarity, see ":MPOLarity:MARKer1|2|3|4" on page 151.

NOTE Do not use the ALC hold for more than 100 ms, because it can affect the waveform's output amplitude.

The marker signal has a minimum of a two-sample delay in its response relative to the waveform signal response. To compensate for the marker signal delay, offset marker points from the waveform sample point at which you want the ALC sampling to begin.

The ALC hold setting is part of the file header information, so saving the setting to the file header saves the current marker routing for the waveform file.

NOTE A waveform file that has unspecified settings in the file header uses the previous waveform's routing settings.

For more information on the marker ALC hold function, see the *User's Guide*. For setting the marker points, see ":MARKer[:SET]" on page 147.

NONE	This terminates the marker ALC hold function.
M1–M4	These are the marker choices. The ALC hold feature uses only one marker at a time.

***RST** NONE

Example

```
:RAD:ARB:MDES:ALCH M1
```

The preceding example routes marker 1 to the ALC Hold function.

Key Entry **None** **Marker 1** **Marker 2** **Marker 3** **Marker 4**

:MDEStination:PULSe

Supported N5182A with Option 651/652/654

CAUTION The pulse function incorporates ALC hold. Incorrect automatic level control (ALC) sampling can create a sudden unleveled condition that may create a spike in the RF output potentially damaging a DUT or connected instrument. Ensure that you set markers to let the ALC sample over an amplitude that accounts for the high power levels within the signal.

```
[:SOURce]:RADio[1]:ARB:MDEStination:PULSe NONE|M1|M2|M3|M4
[:SOURce]:RADio[1]:ARB:MDEStination:PULSe?
```

This command enables or disables the marker pulse/RF blanking function for the selected marker.

This function automatically uses the ALC hold function, so there is no need to select both the ALC hold and pulse/RF blanking functions for the same marker.

NOTE Do not use ALC hold for more than 100 ms, because it can affect the waveform's output amplitude.

The signal generator blanks the RF output when the marker signal goes low. The marker polarity determines when the marker signal is low. For a positive polarity, this is during the marker points. For a negative polarity, this is when there are no marker points. For setting a marker's polarity, see ":MPOLarity:MARKer1|2|3|4" on page 151.

NOTE Set marker points prior to using this function. Enabling this function without setting marker points may create a continuous low or high marker signal, depending on the marker polarity. This causes either no RF output or a continuous RF output. See ":MARKer[:SET]" on page 147 for setting the marker points.

The marker signal has a minimum of a two-sample delay in its response relative to the waveform signal response. To compensate for the marker signal delay, offset marker points from the waveform sample point at which you want the RF blanking to begin. The RF blanking setting is part of the file header information, so saving the setting to the file header saves the current marker routing for the waveform file.

NOTE A waveform file that has unspecified settings in the file header uses the previous waveform's routing settings. This could create the situation where there is no RF output signal, because the previous waveform used RF blanking.

For more information on the marker RF blanking function, see the *User's Guide*.

NONE This terminates the marker RF blanking/pulse function.

M1–M4 These are the marker choices. The RF blanking/pulse feature uses only one marker at a time.

Example

```
:RAD:ARB:MDES:PULS M2
```

The preceding example routes marker 2 to Pulse/RF Blanking.

***RST** NONE

Key Entry **None** **Marker 1** **Marker 2** **Marker 3** **Marker 4**

:MPOLarity:MARKer1|2|3|4

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:MPOLarity:MARKer1|2|3|4 NEGative|POSitive
[:SOURce]:RADio[1]:ARB:MPOLarity:MARKer1|2|3|4?
```

This command sets the polarity for the selected marker. For a positive marker polarity, the marker signal is high during the marker points. For a negative marker polarity, the marker signal is high during the period of no marker points.

Example

```
:RAD:ARB:MPOL:MARK3 NEG
```

The preceding example sets the polarity for marker 3 to negative.

***RST** POS

Key Entry **Marker 1 Polarity Neg Pos** **Marker 2 Polarity Neg Pos** **Marker 3 Polarity Neg Pos** **Marker 4 Polarity Neg Pos**

:NOISe:BANDwidth

Supported N5182A with Option 403

```
[:SOURce]:RADio[1]:ARB:NOISe:BANDwidth <value><unit>
[:SOURce]:RADio[1]:ARB:NOISe:BANDwidth?
```

This command selects the noise bandwidth value of the real-time noise for an ARB waveform.

Typically, this value is set slightly wider than the signal bandwidth.

***RST** +1.00000000E+000

Range 1Hz–100 MHz

Key Entry **Noise Bandwidth**

:NOISe:CBWidth

Supported N5182A with Option 403

```
[:SOURce]:RADio[1]:ARB:NOISe:CBWidth <value><unit>
[:SOURce]:RADio[1]:ARB:NOISe:CBWidth?
```

This command selects the carrier bandwidth over which the additive white gaussian noise (AWGN) is applied. The noise power will be integrated over the selected bandwidth for the purposes of calculating carrier to noise ratio (C/N). The carrier bandwidth is limited to the ARB sample rate, but cannot exceed 125 MHz. For more information, refer to ":NOISe[:STATe]" and ":NOISe:BANDwidth".

***RST** +1.00000000E+000

Range 1Hz–100 MHz

Key Entry Carrier Bandwidth

:NOISe:CN

Supported N5182A with Option 403

```
[:SOURce]:RADio[1]:ARB:NOISe:CN <value><unit>
[:SOURce]:RADio[1]:ARB:NOISe:CN?
```

This command sets the carrier to noise ratio (C/N) in dB. The carrier power is defined as the total modulated signal power without noise power added. The noise power is applied over the specified bandwidth of the carrier signal. For more information, refer to ":NOISe:CBWidth" on page 152.

Example

```
:RAD:ARB:NOIS:CN 50DB
```

The preceding example sets the carrier to noise ratio to 50 dB.

***RST** +0.00000000E+000

Range –100 to 100 dB

Key Entry Carrier to Noise Ratio

:NOISe[:STATe]

Supported N5182A with Option 403

```
[:SOURce]:RADio[1]:ARB:NOISe[:STATe] ON|OFF|1|0
[:SOURce]:RADio[1]:ARB:NOISe[:STATe]?
```

This command enables or disables adding real-time additive white gaussian noise (AWGN) to the carrier modulated by the waveform being played by the dual ARB waveform player.

Maximum bandwidth cannot exceed 125 MHz.

Example

```
:RAD:ARB:NOIS ON
```

The preceding example applies real-time AWGN to the carrier.

***RST** 0

Key Entry **Real-time Noise Off On**

:RETRigger

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:RETRigger ON|OFF|1|0|IMMediate
[:SOURce]:RADio[1]:ARB:RETRigger?
```

This command enables or disables the ARB retriggering mode; the retrigger mode controls how the retriggering function performs while a waveform is playing.

ON (1)	This choice (Buffered Trigger) specifies that if a trigger occurs while a waveform is playing, the waveform will retrigger at the end of the current waveform sequence and play once more.
OFF (0)	This choice (No Retrigger) specifies that if a trigger occurs while a waveform is playing, the trigger will be ignored.
IMMediate	This choice (Restart on Trigger) specifies that if a trigger occurs while a waveform is playing, the waveform will reset and replay from the start immediately upon receiving a trigger.

***RST** ON

Key Entry **No Retrigger** **Buffered Trigger** **Restart on Trigger**

Remarks This command applies to the single trigger type only.

:RSCaling

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:RSCaling <value>
[:SOURce]:RADio[1]:ARB:RSCaling?
```

This command adjusts the scaling value in percent that is applied to a waveform while it is playing. The variable <value> is expressed as a percentage. Runtime scaling does not alter the waveform data file. For more information about runtime scaling, refer to the *User's Guide*.

Example

```
:RAD:ARB:RSC 50
```

The preceding example applies a 50% scaling factor to the selected waveform.

***RST** +7.00000000E+001

Range 1–100

Key Entry **Waveform Runtime Scaling**

Remarks Runtime scaling does not alter the waveform data file.

:SCALing

Supported N5182A with Option 651/652/654

`[:SOURce]:RADio[1]:ARB:SCALing "<file_name>",<value>`

This command scales the designated `"<file_name>"` waveform file while it is being played by the dual ARB player. The variable `<value>` is expressed as a percentage, 1–100%. For information on file name syntax, see "File Name Variables" on page 12.

Unlike runtime scaling (:RSCaling), Scaling is additive and permanent. You cannot scale up. If you scale a waveform file by 60% and then scale it again to 80% you will scale down the 60% waveform file. For more information about waveform file scaling, refer to the *User's Guide*.

Example

```
:RAD:ARB:SCAL "Test_Data", 50
```

The preceding example applies a 50% scaling factor to the Test_Data waveform file.

Range 1–100

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

Key Entry **Scaling** **Scale Waveform Data**

Remarks Refer to "File Name Variables" on page 12 for information on the file name syntax.

:SCLock:RATE

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:SCLock:RATE <value>
[:SOURce]:RADio[1]:ARB:SCLock:RATE?
```

This command sets the sample clock rate for the dual ARB format.

The variable <value> is expressed in units of hertz.

***RST** +1.25000000E+008

Range 1–1.25E8

Range *OSR Option 651*: 4E0 – 125E8

OSR Option 652: 4E0 – 125E8

OSR Option 654: 4E0 – 125E8

Scale: 0–1

Key Entry **ARB Sample Clock**

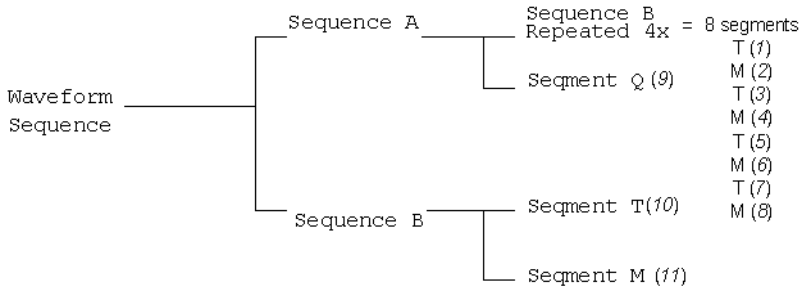
:SEQuence

Supported All with Option 651/652/654

```
[:SOURce]:RADio[1]:ARB:SEQuence[:MWAVeform] <filename>,<waveform1>,<reps>,NONE|M1|M2
|M3|M4|M1M2|M1M3|M1M4|M2M3|M2M4|M3M4|M1M2M3|M1M2M4|M1M3M4|M2M3M4|M1M2M3M4|ALL,{,<wave
form2>,<reps>,NONE|M1|M2|M3|M4|M1M2|M1M3|M1M4|M2M3|M2M4|M3M4|M1M2M3|M1M2M4|M1M3M4|M2M
3M4|M1M2M3M4|ALL,} [:SOURce]:RADio[1]:ARB:SEQuence[:MWAVeform]? <filename>
```

This command creates a waveform sequence. A waveform sequence is made up of segments and other sequences. Any number of segments, up to a segment count limit of 1024, can be used to create a sequence. The count limit is determined by the number of segments in the waveform sequence. Repeated segments are included in the count limit.

For example, using the figure below, suppose a waveform is created using two sequences: Sequence_A and Sequence_B. Sequence_A consists of Sequence_B and Segment_Q with Sequence_B repeated four times. The total segment count for this waveform sequence would be eleven.



The query returns the contents and segment settings of the waveform sequence file

The segments and sequences play in the same order as placed into the waveform sequence by the command. Once you create the file, you cannot edit the segment settings or add further waveform segments unless you use the signal generator's front panel. Using the same waveform sequence name overwrites the existing file with that name. To use a segment's marker settings, you must enable the segment's markers within the segment or within the waveform sequence. A sequence is stored in the catalog of SEQ files USER/SEQ or SEQ: directory.

When you create a waveform sequence, the Agilent MXG also creates a file header for the sequence. This file header takes priority over segment or nested sequence file headers. Refer to the *User's Guide* for more information on file headers. To save the file header, see ":HEADer:SAVE" on page 145.

"<file_name>" This variable names the waveform *sequence* file. For information on the file name syntax, see "File Name Variables" on page 12.

"<waveform1>" This variable specifies the name of an existing waveform *segment* or sequence file. A waveform segment or the waveform segments in a specified sequence must reside in volatile memory, WFM1, before it can be played by the dual ARB player.

For information on the file name syntax, see "File Name Variables" on page 12, and for more information on waveform segments, see the *User's Guide*.

"<waveform2>" This variable specifies the name of a second existing waveform *segment* or sequence file. The same conditions required for waveform1 apply for this segment or sequence. Additional segments and other sequences can be inserted into the file.

<reps> This variable sets the number of times a segment or sequence plays (repeats) before the next segment or sequence plays.

NONE This choice disables all four markers for the waveform. Disabling markers means that the waveform sequence ignores the segment's or sequence's marker settings.

M1, M2, M3, M4 These choices, either individually or a combination of them, enable the markers for the waveform segment or sequence. Markers not specified are ignored for that segment or sequence.

ALL This choice enables all four markers in the waveform segment or sequence.

Example

```
:RAD:ARB:SEQ "SEQ:Test_Data","WFM1:ramp_test_wfm",25,M1M4,
"WFM1:sine_test_wfm",100,ALL
```

NOTE A carriage return or line feed is never included in a SCPI command. The example above contains a carriage return so that the text will fit on the page.

The preceding example creates a waveform sequence file named `Test_Data`. This file consists of the factory-supplied waveform segments, `ramp_test_wfm` and `sine_test_wfm`. The waveform is stored in the signal generator's `SEQ:` directory.

- The first segment, `ramp_test_wfm`, has 25 repetitions with markers 1 and 4 enabled.
- The second segment, `sine_test_wfm`, has 100 repetitions with all four markers enabled.

Range <reps>: 1–65535

Key Entry **Build New Waveform Sequence** **Name and Store** **Insert Waveform** **Edit Repetitions** **Toggle Marker 1** **Toggle Marker 2** **Toggle Marker 3** **Toggle Marker 4**

Remarks These softkeys are located under the ARB menu.

:TRIGger:TYPE

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:TRIGger:TYPE CONTinuous|SINGle|GATE|SADVance
[:SOURce]:RADio:ARB:TRIGger:TYPE?
```

This command sets the trigger mode (type) that controls the waveform's playback.

Triggers control the playback by telling the Agilent MXG when to play the modulating signal (waveform). Depending on the trigger settings for the Agilent MXG, the waveform playback can occur once, continuously, or the Agilent MXG may start and stop playing the waveform repeatedly (GATE mode).

A trigger signal comprises both positive and negative signal transitions (states), which are also called high and low periods. You can configure the Agilent MXG to trigger on either state of the trigger signal. It is common to have multiple triggers, also referred to as trigger occurrences or events, occur when the signal generator requires only a single trigger. In this situation, the Agilent MXG recognizes the first trigger and ignores the rest.

When you select a trigger mode, you may lose the signal (carrier plus modulating) from the RF output until you trigger the waveform. This is because the Agilent MXG sets the I and Q signals to zero volts prior to the first trigger event, which suppresses the carrier. After the first trigger event, the waveform's final I and Q levels determine whether you will see the carrier signal or not (zero = no carrier, other values = carrier visible). At the end of most files, the final I and Q points are set to a value other than zero.

There are four parts to configuring the trigger:

- Choosing the trigger type, which controls the waveform's transmission.
- Setting the waveform's response to triggers:
  - CONTinuous, see ":TRIGger:TYPE:CONTinuous[:TYPE]" on page 158
  - SINGle, see ":RETRigger" on page 153
  - SADVance, see ":TRIGger:TYPE:SADVance[:TYPE]" on page 159
  - GATE, selecting the mode also sets the response
- Selecting the trigger source (see ":TRIGger[:SOURce]" on page 160), which determines how the Agilent MXG receives its trigger signal, internally or externally. The GATE choice requires an external trigger.
- Setting the trigger polarity when using an external source:
  - CONTinuous, SINGle, and SADVance see ":TRIGger[:SOURce]:EXTernal:SLOPe" on page 161
  - GATE, see ":TRIGger:TYPE:GATE:ACTive" on page 158

For more information on triggering, see the *User's Guide*.

The following list describes the trigger type command choices:

CONTinuous	Upon triggering, the waveform repeats continuously.
SINGle	Upon triggering, the waveform segment or sequence plays once.
SADVance	The trigger controls the segment advance within a waveform sequence. To use this choice, a waveform sequence must be the active waveform. Ensure that all segments in the sequence reside in volatile memory.
GATE	An external trigger signal repeatedly starts and stops the waveform's playback (transmission). The time duration for playback depends on the duty period of the trigger signal and the gate polarity selection (see ":TRIGger:TYPE:GATE:ACTive" on page 158). The waveform plays during the inactive state and stops during the active polarity selection state. The active state can be set high or low. The gate mode works only with an external trigger source.
***RST**	CONT
Key Entry	**Continuous** **Single** **Gate** **Segment Advance**

:TRIGger:TYPE:CONTinuous[:TYPE]

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:TRIGger:TYPE:CONTinuous[:TYPE] FREE|TRIGger|RESet
[:SOURce]:RADio:ARB:TRIGger:TYPE:CONTinuous[:TYPE]?
```

This command selects the waveform's response to a trigger signal while using the continuous trigger mode.

For more information on triggering and to select the continuous trigger mode, see ":TRIGger:TYPE" on page 156.

The following list describes the waveform's response to each of the command choices:

FREE — Turning the ARB format on immediately triggers the waveform. The waveform repeats until you turn the format off, select another trigger, or choose another waveform file.

TRIGger — The waveform waits for a trigger before play begins. When the waveform receives the trigger, it plays continuously until you turn the format off, select another trigger, or choose another waveform file.

RESet — The waveform waits for a trigger before play begins. When the waveform receives the trigger, it plays continuously. Subsequent triggers reset the waveform to the beginning. For a waveform sequence, this means to the beginning of the first segment in the sequence.

***RST** FREE

Key Entry **Free Run** **Trigger & Run** **Reset & Run**

:TRIGger:TYPE:GATE:ACTive

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:TRIGger:TYPE:GATE LOW|HIGH
[:SOURce]:RADio:ARB:TRIGger:TYPE:GATE?
```

This command selects the active state (gate polarity) of the gate while using the gating trigger mode.

The LOW and HIGH selections correspond to the low and high states of an external trigger signal. For example, when you select HIGH, the active state occurs during the high of the trigger signal. When the active state occurs, the Agilent MXG stops the waveform playback at the last played sample point, then restarts the playback at the next sample point when the active state occurs. For more information on triggering and to select gating as the trigger mode, see ":TRIGger:TYPE" on page 156.

The following list describes the Agilent MXG's gating behavior for the polarity selections:

LOW — The waveform playback stops when the trigger signal goes high and restarts when the trigger signal goes low.

HIGH — The waveform playback stops when the trigger signal goes low and restarts when the trigger signal goes high.

***RST** HIGH

Key Entry **Active Low** **Active High**

:TRIGger:TYPE:SADVance[:TYPE]

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:TRIGger:TYPE:SADVance[:TYPE] SINGle|CONTinuous
[:SOURce]:RADio:ARB:TRIGger:TYPE:SADVance[:TYPE]?
```

This commands selects the waveform's response to a trigger signal while using the segment advance (SADVance) trigger mode.

When the Agilent MXG receives multiple trigger occurrences when only one is required, the signal generator uses the first trigger and ignores the rest. For more information on triggering and to select segment advance as the trigger mode, see ":TRIGger:TYPE" on page 156.

The following list describes the waveform's response to each of the command choices:

SINGle — Each segment in the sequence requires a trigger to play, and a segment plays only once, ignoring a segment's repetition value (see ":SEQuence" on page 155 for repetition information). The following list describes a sequence's playback behavior with this choice:

- After receiving the first trigger, the first segment plays to completion.
- When the waveform receives a trigger after a segment completes, the sequence advances to the next segment and plays that segment to completion.
- When the waveform receives a trigger during play, the current segment plays to completion. Then the sequence advances to the next segment, and it plays to completion.
- When the waveform receives a trigger either during or after the last segment in a sequence plays, the sequence resets and the first segment plays to completion.

CONTinuous — Each segment in the sequence requires a trigger to play. After receiving a trigger, a segment plays continuously until the waveform receives another trigger. The following list describes a sequence's playback behavior with this choice:

- After receiving the first trigger, the first segment plays continuously.
- A trigger during the current segment play causes the segment to play to the end of the segment file, then the sequence advances to the next segment, which plays continuously.
- When last segment in the sequence receives a trigger, the sequence resets and the first segment plays continuously.

Example

```
:RAD:ARB:TRIG:TYPE:SADV CONT
```

The preceding example selects the continuous segment advance mode.

***RST** CONT

Key Entry **Single** **Continuous**

:TRIGger[:SOURce]

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:TRIGger[:SOURce] KEY|EXT|BUS
[:SOURce]:RADio:ARB:TRIGger[:SOURce]?
```

This command sets the trigger source.

For more information on triggering, see ":TRIGger:TYPE" on page 156. The following list describes the command choices:

KEY — This choice enables manual triggering by pressing the front-panel **Trigger** hardkey.

EXT — An externally applied signal triggers the waveform. This is the only choice that works with gating. The following conditions affect an external trigger:

- The input connector selected for the trigger signal. You have a choice between the rear-panel PATTERN TRIG IN connector or the PATT TRIG IN 2 pin on the rear-panel AUXILIARY I/O connector. To make the connector selection, see ":TRIGger[:SOURce]:EXTernal[:SOURce]" on page 162.

  For more information on the connectors and on connecting the cables, see the *User's Guide*.

- The trigger signal polarity:
  - gating mode, see ":TRIGger:TYPE:GATE:ACTive" on page 158
  - continuous and single modes, see ":TRIGger[:SOURce]:EXTernal:SLOPe" on page 161
- The time delay between when the Agilent MXG receives a trigger and when the waveform responds to the trigger. There are two parts to setting the delay:
  - setting the amount of delay, see ":TRIGger[SOURce]:EXTernal:DELay" on page 160
  - turning the delay on, see ":TRIGger[:SOURce]:EXTernal:DELay:STATe" on page 161

BUS — This choice enables triggering over the GPIB or LAN using the `*TRG` or `GET` commands or the AUXILIARY INTERFACE (USB) using the `*TRG` command.

***RST** EXT

Key Entry **Trigger Key** **Ext** **Bus**

:TRIGger[SOURce]:EXTernal:DELay

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:DELay <value>
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:DELay?
```

This command sets the amount of time to delay the Agilent MXG's response to an external trigger.

The delay is a path (time) delay between when the Agilent MXG receives the trigger and when it responds to the trigger. For example, configuring a trigger delay of two seconds, causes the Agilent MXG to wait two seconds after receipt of the trigger before the Agilent MXG plays the waveform.

The delay does not occur until you turn it on (see ":TRIGger[:SOURce]:EXTernal:DELay:STATe" on page 161). You can set the delay value either before or after turning it on.

For more information on configuring an external trigger source and to select external as the trigger source, see ":TRIGger[:SOURce]" on page 160.

The unit of measurement for the variable <value> is in seconds (nsec–sec).

***RST** +1.00000000E–003

Range 1E–8 to 3E1

Key Entry **Ext Delay Time**

:TRIGger[:SOURce]:EXTernal:DELay:STATe

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:DELay:STATe ON|OFF|1|0
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:DELay:STATe?
```

This command enables or disables the operating state of the external trigger delay function.

For setting the delay time, see ":TRIGger[SOURce]:EXTernal:DELay" on page 160, and for more information on configuring an external source, see":TRIGger[:SOURce]" on page 160.

***RST** 0

Key Entry **Ext Delay Off On**

:TRIGger[:SOURce]:EXTernal:SLOPe

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:SLOPe POSitive|NEGative
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:SLOPe?
```

This command sets the polarity for an external trigger signal while using the continuous, single triggering mode. To set the polarity for gating, see ":TRIGger:TYPE:GATE:ACTive" on page 158.

The POSitive and NEGative selections correspond to the high (positive) and low (negative) states of the external trigger signal. For example, when you select POSitive, the waveform responds (plays) during the high state of the trigger signal. When the Agilent MXG receives multiple trigger occurrences when only one is required, the signal generator uses the first trigger and ignores the rest.

For more information on configuring an external trigger source and to select external as the trigger source, see ":TRIGger[:SOURce]" on page 160.

***RST** NEG

Key Entry **Ext Polarity Neg Pos**

:TRIGger[:SOURce]:EXTernal[:SOURce]

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal[:SOURce] EPT1|EPT2|
EPTRIGGER1|EPTRIGGER2
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal[:SOURce]?
```

This command selects which PATTERN TRIG IN connection the Agilent MXG uses to accept an externally applied trigger signal when external is the trigger source selection.

For more information on configuring an external trigger source and to select external as the trigger source, see ":TRIGger[:SOURce]" on page 160. For more information on the rear-panel connectors, see the *User's Guide*.

The following list describes the command choices:

EPT1 This choice is synonymous with EPTRIGGER1 and selects the PAT TRIG rear-panel connector.

EPT2 This choice is synonymous with EPTRIGGER2 and selects the PATT TRIG IN 2 pin on the rear-panel AUX I/O connector.

EPTRIGGER1 This choice is synonymous with EPT1 and selects the PAT TRIG rear-panel connector.

EPTRIGGER2 This choice is synonymous with EPT2 and selects the PATT TRIG IN 2 pin on the rear-panel AUXILIARY I/O connector.

***RST** EPT1

Key Entry **Patt Trig In 1** **Patt Trig In 2**

:WAVeform

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:WAVeform "WFM1:file_name"|"SEQ:file_name"
[:SOURce]:RADio:ARB:WAVeform?
```

This command selects a waveform file or sequence, for the dual ARB player to play. The file must be present in volatile memory, WFM1, or in the SEQ directory. If a file is in non-volatile memory (NVWFM), use the command ":COPY[:NAME]" on page 81 to copy the file to WFM1.Any specified values in the header are applied to the ABR upon selection. Unspecified fields in the header cause no change in the ARB state.

"WFM1:file_name" This variable names a waveform file residing in volatile memory (WFM1:). For information on the file name syntax, see "File Name Variables" on page 12.

"SEQ:file_name" This variable names a sequence file residing in the catalog of sequence files. For more information on the file name syntax, see "File Name Variables" on page 12.

Example

```
:RAD:ARB:WAV "WFM1:Test_Data"
```

The preceding example selects the file Test_Data from the list of files in volatile waveform memory, WFM1, and applies its file header settings.

Key Entry **Select Waveform**

:WAVeform:NHEAders

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB:WAVeform:NHEaders "WFM1:file_name"|"SEQ:filename"
[:SOURce]:RADio:ARB:WAVeform:NHEaders?
```

This command, for the dual ARB mode, allows for a fast selection of a segment or sequence waveform file. No header information or settings are applied to the segment or sequence waveform file when this command is used. This will improve the access or loading speed of the waveform file. The file must be in volatile waveform memory (WFM1), or in the SEQ directory. If a file is in non-volatile waveform memory (NVWFM), use the command ":COPY[:NAME]" on page 81 to copy files to WFM1.

"WFM1:file_name" This variable names a waveform file residing in volatile memory:WFM1. For information on the file name syntax, see "File Name Variables" on page 12.

"SEQ:filename" This variable names a sequence file residing in the catalog of sequence files. For more information on the file name syntax, see "File Name Variables" on page 12.

Example

```
:RAD:ARB:WAV:NHE "Test_Data"
```

The preceding example selects the file Test_Data, without applying header settings.

[:STATe]

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio:ARB[:STATe] ON|OFF|1|0
[:SOURce]:RADio:ARB[:STATe]?
```

This command enables or disables the arbitrary waveform generator function.

***RST** 0

Key Entry **ARB Off On**

LARB Subsystem–Option 651/652/654 ([:SOURce]:RADio:LARB)

[:STATe]

Supported N5182A with Option 651/652/654

```
[:SOURce]:RADio[1]:LARB[:STATe] ON|OFF|1|0
[:SOURce]:RADio[1]:LARB[:STATe]?
```

This command enables or disables the waveform sweep function, when the signal generator is in list sweep mode.

***RST** 0

Key Entry **Waveform Off On**

Remarks The **Sweep Type List Step** softkey must be set to List for this command to function.

6 SCPI Compatibility

This chapter provides a comprehensive listing of SCPI commands and programming codes for signal generator models that are supported by the N5181A/82A.

Overview

This Chapter contains the following major sections:

The following list shows the supported models along with the language type for each one:

N5181A/82A	SCPI commands
E44xxB	SCPI commands
E4428C/38C	SCPI commands
8648A/B/C/D	SCPI commands
8656B	programming codes
8657A/B/D/J	programming codes

These commands and programming codes are separated into compatible and non-compatible sections. In many instances, the non-compatible section has the least number of commands/codes, thus providing a more time-efficient way of determining whether or not a command/code is supported by the N5181A/82A.

In some cases, SCPI commands are only partially supported. This usually occurs due to a variance in parameters between the N5181A/82A and other signal generator models. When this condition occurs, the SCPI command will appear in both the compatible and non-compatible sections showing the exact SCPI command syntax that is supported and unsupported.

In addition to providing the compatible command/code listing, this chapter also provides you with N5181A/82A SCPI commands that lets you change the signal generator identification output (see ":SYSTem:IDN" on page 167), select a compatible programming language (see ":SYSTem:LANGuage" on page 168), and query the signal generator for errors (see ":SYSTem:ERRor[:NEXT]" on page 168).

Changing the Signal Generator Identification String

:SYSTem:IDN

Supported All

`:SYSTem:IDN "<string>"`

This Agilent MXG signal generator command modifies the identification string that the `*IDN?` query returns. Sending an empty string returns the `*IDN?` query output to its factory shipped setting. The maximum string length is 72 characters.

Modification of the `*IDN?` query output enables the Agilent MXG signal generator to identify itself as another signal generator when it is used as a backward compatible replacement. This modification of the identification string does not affect the display diagnostic information, which is shown by pressing the **Diagnostic Info** softkey.

Functional N5181A/82A SCPI Commands While in a Compatible Language Mode

The commands in this section are used for configuring the signal generator compatible programming language and for isolating problems.

:PRESet:LANGuage

Supported All

```
:SYSTem:PRESet:LANGuage "SCPI"|"COMP"|"8648"
:SYSTem:PRESet:LANGuage?
```

This command sets the remote language that is available when the signal generator is preset.

SCPI This choice provides compatibility for SCPI commands.

COMP This choice provides compatibility for the 8656B, 8657A/B signal generator which is supported by using the GPIB interface.

8648 This choice provides compatibility for the 8648A/B/C/D signal generator which is supported only through a GPIB interface.

***RST** "SCPI"

Key Entry **SCPI** **8656B,8657A/B** **8648A/B/C/D**

:SYSTem:LANGuage

Supported All

```
:SYSTem:LANGuage "SCPI"|"COMP"|"8648"
:SYSTem:LANGuage?
```

This command sets the remote language for the signal generator.

The setting enabled by this command is not affected by signal generator power-on, preset, or *RST.

SCPI This choice provides compatibility for SCPI commands.

COMP This choice provides compatibility for the 8656B, 8657A/B signal generator which is supported only through a GPIB interface.

8648 This choice provides compatibility for the 8648A/B/C/D signal generator which is supported only through a GPIB interface.

Key Entry **SCPI** **8656B,8657A/B** **8648A/B/C/D**

:SYSTem:ERRor[:NEXT]

Supported All

```
:SYSTem:ERRor[:NEXT]?
```

This query returns the most recent error message from the signal generator error queue. If there are no error messages, the query returns the following output:

```
+0,"No error"
```

When there is more than one error message, the query will need to be sent for each message. Each error message is erased after being queried.

Key Entry View Next Error Message

E44xxB Compatible Commands

NOTE The Agilent MXG has only one AM path; and only one internal and one external source. If executed, the "2" path or "2" internal or external source commands will result in a "ERROR: -113, Undefined Header" to be generated in the signal generator.

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
IEEE Common Commands		
*CLS	✓	
*ESE <data> *ESE?	✓	
*ESR?	✓	
*IDN?	✓	
*OPC *OPC?	✓	
*RCL <reg_num>	✓	
*RST	✓	
*SAV <reg_num>	✓	
*SRE <data> *SRE?	✓	
*STB?	✓	
*TRG	✓	
*TST?	✓	
*WAI	✓	
Calibration Subsystem		
:CALibration:DCFM	✓	
:CALibration:IQ	✓	
:CALibration:IQ:DEFault	✓	
:CALibration:IQ:FULL	✓	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
:CALibration:IQ:STARt <value> :CALibration:IQ:STARt?	✓	
:CALibration:IQ:STOP <value> :CALibration:IQ:STOP?	✓	
Communication Subsystem		
:SYSTem:COMMunicate:GPIB:ADDRess <number> :SYSTem:COMMunicate:GPIB:ADDRess?	✓	
:SYSTem:COMMunicate:SERial:BAUD <number> :SYSTem:COMMunicate:SERial:BAUD?	–	
:SYSTem:COMMunicate:SERial:CONTrol:RTS ON|OFF|IBFull|RFR :SYSTem:COMMunicate:SERial:CONTrol:RTS?	–	
:SYSTem:COMMunicate:SERial:ECHO ON|OFF :SYSTem:COMMunicate:SERial:ECHO?	–	
:SYSTem:COMMunicate:SERial:RESet	–	
:SYSTem:COMMunicate:SERial:TOUT <value> :SYSTem:COMMunicate:SERial:TOUT?	–	
:SYSTem:COMMunicate:SERial:CONTrol:RTS ON|OFF|IBFull|RFR :SYSTem:COMMunicate:SERial:CONTrol:RTS?	–	
Diagnostic Subsystem		
:DIAGnostic[:CPU]:INFOrmation:BOARds?	–	
:DIAGnostic[:CPU]:INFOrmation:CCOunt:ATTenuator?	✓	
:DIAGnostic[:CPU]:INFOrmation:CCOunt:PON?	✓	
:DIAGnostic[:CPU]:INFOrmation:CCOunt:PROTection?	✓	
:DIAGnostic[:CPU]:INFOrmation:DISPlay:OTIMe?	✓	
:DIAGnostic[:CPU]:INFOrmation:LIDN?	✓	
:DIAGnostic[:CPU]:INFOrmation:OPTions?	✓	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
:DIAGnostic[:CPU]:INFOrmation:OPTions:DETail?	✓	
:DIAGnostic[:CPU]:INFOrmation:OTIMe?	✓	
:DIAGnostic[:CPU]:INFOrmation:SDATe?	✓	
Display Subsystem		
:DISPlay:BRIGhtness <value> :DISPlay:BRIGhtness?	✓	
:DISPlay:CONTrast <value> :DISPlay:CONTrast?	✓	
:DISPlay:INVerse ON|OFF|1|0	✓	Supported but the following query is not supported: :DISPlay:INVerse?
:DISPlay:REMote ON|OFF|1|0 :DISPlay:REMote?	✓	
Memory Subsystem		
:MEMory:CATalog:BINary?	✓	
:MEMory:CATalog:BIT?	–	
:MEMory:CATalog:CDMa?	–	
:MEMory:CATalog:DMOD?	–	
:MEMory:CATalog:DWCDma?	–	
:MEMory:CATalog:FCDMa?	–	
:MEMory:CATalog:FIR?	–	
:MEMory:CATalog:FSK?	–	
:MEMory:CATalog:FWCDma?	–	
:MEMory:CATalog:IQ?	–	
:MEMory:CATalog:LIST?	✓	
:MEMory:CATalog:MCDMa?	–	
:MEMory:CATalog:MDMod?	–	
:MEMory:CATalog:MDWCdma?	–	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
:MEMory:CATalog:MFCdma?	–	
:MEMory:CATalog:MFWCdma?	–	
:MEMory:CATalog:MTONe?	–	
:MEMory:CATalog:RCDMa?	–	
:MEMory:CATalog:RWCDma?	–	
:MEMory:CATalog:SEQ?	✓	
:MEMory:CATalog:SHAPe?	–	
:MEMory:CATalog:STATe?	✓	
:MEMory:CATalog:UWCDma?	–	
:MEMory:CATalog:WCDMa?	–	
:MEMory:CATalog[:ALL]?	✓	
:MEMory:COPY[:NAME] "<file name>","<file name>"	✓	
:MEMory:DATA "<file name>",<datablock>	✓	
:MEMory:DATA? "<file name>"	✓	
:MEMory:DATA:BIT "<file name>",<bit_count>,<datablock>	–	
:DATA:BIT? "<file name>"	–	
:MEMory:DATA:FIR "<file name>",osr,coefficient{,coefficient}	–	
:MEMory:DATA:FIR? "<file name>"	–	
:MEMory:DATA:FSK "<file name>",<num_states>,<f0>,<f1>,...<f(n)>[,<diff_state>,<num_diff_states>,<diff0>,<diff1>,...<diff(n)>]	–	
:MEMory:DATA:FSK? "<file name>"	–	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
:MEMory:DATA:IQ "<file name>",<offsetQ>,<num_states>,<i0>,<q0>,<i1>,<q1>,...<i(n),q(n)>[,<diff_state>,<num_diff_states>,<diff0>,<diff1>,...<diff(n)>]	–	
:MEMory:DATA:IQ? "<file name>"	–	
:MEMory:DATA:PRAM?	–	
:MEMory:DATA:PRAM:BLOCk <datablock>	–	
:MEMory:DATA:PRAM:LIST <uint8>{,<uint8>,<...>}	–	
:MEMory:DATA:SHAPe "<file name>",<num_rise_points>,<rp0>,<rp1>,...<rp(n)>,<num_fall_points>,<fp0>,<fp1>,...<fp(n)>	–	
:MEMory:DATA:SHAPe? "<file name>"	–	
:MEMory:DELete:ALL	✓	
:MEMory:DELete:BINary	✓	
:MEMory:DELete:BIT	–	
:MEMory:DELete:CDMa	–	
:MEMory:DELete:DMOD	–	
:MEMory:DELete:DWCDma	–	
:MEMory:DELete:FCDMa	–	
:MEMory:DELete:FIR	–	
:MEMory:DELete:FSK	–	
:MEMory:DELete:FWCDma	–	
:MEMory:DELete:IQ	–	
:MEMory:DELete:LIST	–	
:MEMory:DELete:MCDMa	–	
:MEMory:DELete:MDMod	–	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG – = Not supported by Agilent MXG	E44xxB	Remarks
:MEMory:DELete:MDWCdma	–	
:MEMory:DELete:MFCdma	–	
:MEMory:DELete:MFWCdma	–	
:MEMory:DELete:MTONe	–	
:MEMory:DELete:RCDMa	–	
:MEMory:DELete:RWCDma	–	
:MEMory:DELete:SEQ	✓	
:MEMory:DELete:SHAPe	–	
:MEMory:DELete:STATe	✓	
:MEMory:DELete:UWCDma	–	
:MEMory:DELete:WCDMa	–	
:MEMory:DELete[:NAME] "<file name>"	✓	
:MEMory:FREE[:ALL]?	✓	
:MEMory:LOAD:LIST "<file name>"	✓	
:MEMory:MOVE <src_file>,<dest_file>	✓	
:MEMory:STATe:COMMent <reg_num>,<seq_num>,"<comment>"	✓	
:MEMory:STATe:COMMent? <reg_num>,<seq_num>	✓	
:MMEMory:CATalog? "<msus>"	✓	
:MMEMory:COPY "<file name>","<file name>"	✓	
:MMEMory:DATA "<file name>",<datablock>	✓	
:MMEMory:DATA? "<file name>"	✓	
:MMEMory:DELete[:NAME] "<file name>",["<msus>"]	✓	
:MMEMory:LOAD:ARB:ALL	✓	
:MMEMory:LOAD:LIST "<file name>"	✓	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
:MMEMory:MOVE <src_file>,<dest_file>	✓	
:MMEMory:STORe:ARB:ALL	✓	
:MMEMory:STORe:LIST "<file name>"	✓	
:MEMory:STORe:LIST "<file name>"	3	
Output Subsystem		
:OUTPut:BLANking:AUTO ON|OFF|1|0 :OUTPut:BLANking:AUTO?	✓	
:OUTPut:BLANking[:STATe] ON|OFF|1|0 :OUTPut:BLANking[:STATe]?	✓	
:OUTPut:MODulation[:STATe] ON|OFF|1|0 :OUTPut:MODulation[:STATe]?	✓	
:OUTPut:PROTection:CLEar	✓	
:OUTPut:PROTection:MODE "NORMAL"|"8648" :OUTPut:PROTection:MODE?	–	
:OUTPut:PROTection[:STATe] ON|OFF|1|0 :OUTPut:PROTection[:STATe]?	✓	
:OUTPut:PROTection:TRIPped?	✓	
:OUTPut:SETTled:POLarity NORMal|INVerted :OUTPut:SETTled:POLarity?	–	
:OUTPut:SETTled:RFOFf NORMal|INVerted :OUTPut:SETTled:RFOFf?	–	
:OUTPut:SETTled[:STATe]?	–	
:OUTPut[:STATe] ON|OFF|1|0 :OUTPut[:STATe]?	✓	
Route Subsystem		
:ROUTe:HARDware:DGENerator:...	–	*This subsystem is not supported.*

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
Status Subsystem		
:STATus:OPERation:CONDition?	✓	
:STATus:OPERation:ENABle <value> :STATus:OPERation:ENABle?	✓	
:STATus:OPERation:NTRansition <value> :STATus:OPERation:NTRansition?	✓	
:STATus:OPERation:PTRansition <value> :STATus:OPERation:PTRansition?	✓	
:STATus:OPERation[:EVENt]?	✓	
:STATus:PRESet	✓	
:STATus:QUEStionable:BERT:CONDition?	–	
:STATus:QUEStionable:BERT:ENABle <value> :STATus:QUEStionable:BERT:ENABle?	–	
:STATus:QUEStionable:BERT:NTRansition <value> :STATus:QUEStionable:BERT:NTRansition?	–	
:STATus:QUEStionable:BERT:PTRansition <value> :STATus:QUEStionable:BERT:PTRansition?	–	
:STATus:QUEStionable:BERT[:EVENt]?	–	
:STATus:QUEStionable:CALibration:CONDition?	✓	
:STATus:QUEStionable:CALibration:ENABle <value> :STATus:QUEStionable:CALibration:ENABle?	✓	
:STATus:QUEStionable:CALibration:NTRansition <value> :STATus:QUEStionable:CALibration:NTRansition?	✓	
:STATus:QUEStionable:CALibration:PTRansition <value> :STATus:QUEStionable:CALibration:PTRansition?	✓	
:STATus:QUEStionable:CALibration[:EVENt]?	✓	
:STATus:QUEStionable:CONDition?	✓	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG – = Not supported by Agilent MXG	E44xxB	Remarks
:STATus:QUEStionable:ENABle <value> :STATus:QUEStionable:ENABle?	✓	
:STATus:QUEStionable:FREQuency:CONDition?	✓	
:STATus:QUEStionable:FREQuency:ENABle <value> :STATus:QUEStionable:FREQuency:ENABle?	✓	
:STATus:QUEStionable:FREQuency:NTRansition <value> :STATus:QUEStionable:FREQuency:NTRansition?	✓	
:STATus:QUEStionable:FREQuency:PTRansition <value> :STATus:QUEStionable:FREQuency:PTRansition?	✓	
:STATus:QUEStionable:FREQuency[:EVENt]?	✓	
:STATus:QUEStionable:MODulation:CONDition?	–	
:STATus:QUEStionable:MODulation:ENABle <value> :STATus:QUEStionable:MODulation:ENABle?	–	
:STATus:QUEStionable:MODulation:NTRansition <value> :STATus:QUEStionable:MODulation:NTRansition?	–	
:STATus:QUEStionable:MODulation:PTRansition <value> :STATus:QUEStionable:MODulation:PTRansition?	–	
:STATus:QUEStionable:MODulation[:EVENt]?	–	
:STATus:QUEStionable:NTRansition <value> :STATus:QUEStionable:NTRansition?	✓	
:STATus:QUEStionable:POWer:CONDition?	✓	
:STATus:QUEStionable:POWer:ENABle <value> :STATus:QUEStionable:POWer:ENABle?	✓	
:STATus:QUEStionable:POWer:NTRansition <value> :STATus:QUEStionable:POWer:NTRansition?	✓	
:STATus:QUEStionable:POWer:PTRansition <value> :STATus:QUEStionable:POWer:PTRansition?	✓	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
:STATus:QUEStionable:POWer[:EVENt]?	✓	
:STATus:QUEStionable:PTRansition <value> :STATus:QUEStionable:PTRansition?	✓	
:STATus:QUEStionable[:EVENt]?	✓	
System Subsystem		
:SYSTem:CAPability?	✓	
:SYSTem:ERRor[:NEXT]?	✓	
:SYSTem:HELP:MODE SINGle	✓	*Supported but the following parameter is not supported:* CONTinuous *Supported but the following query is not supported:* :SYSTem:HELP:MODE?
:SYSTem:LANGuage "SCPI"|"COMP"|"8648" :SYSTem:LANGuage?	✓	*Supported but the following parameters are not supported:* "NADC"|"PDC"|"PHS"
:SYSTem:PON:TYPE PRESet|LAST :SYSTem:PON:TYPE?	✓	
:SYSTem:PRESet	✓	
:SYSTem:PRESet:ALL	✓	
:SYSTem:PRESet:LANGuage "SCPI"|"COMP"|"8648" :SYSTem:PRESet:LANGuage?	✓	*Supported but the following parameters are not supported:* "NADC"|"PDC"|"PHS"
:SYSTem:PRESet:PERSistent	✓	
:SYSTem:PRESet:TYPE NORMal|USER :SYSTem:PRESet:TYPE?	✓	
:SYSTem:PRESet:PN9 NORMal|QUICk :SYSTem:PRESet:PN9?	✓	
:SYSTem:PRESet[:USER]:SAVE	✓	
:SYSTem:SSAVer:DELay <value> :SYSTem:SSAVer:DELay?	✓	
:SYSTem:SSAVer:MODE LIGHt|TEXT :SYSTem:SSAVer:MODE?	✓	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
:SYSTem:SSAVer:STATe ON|OFF :SYSTem:SSAVer:STATe?	✓	
:SYSTem:VERSion?	✓	
Trigger Subsystem		
:ABORt	✓	
:INITiate:CONTinuous[:ALL] ON|OFF|1|0 :INITiate:CONTinuous[:ALL]?	✓	
:INITiate[:IMMediate][:ALL]	✓	
:TRIGger:OUTPut:POLarity POSitive|NEGative :TRIGger:OUTPut:POLarity?	✓	
:TRIGger[:SEQuence]:SLOPe POSitive|NEGative :TRIGger[:SEQuence]:SLOPe?	✓	
:TRIGger[:SEQuence]:SOURce BUS|IMMediate|EXTernal|KEY :TRIGger[:SEQuence]:SOURce?	✓	
:TRIGger[:SEQuence][:IMMediate]	✓	
Unit Subsystem		
:UNIT:POWer DBM|DBUV|DBUVEMF|V|VEMF :UNIT:POWer?	✓	
Unit Subsystem		
[:SOURce]:AM:WIDeband:STATe ON|OFF|1|0 [:SOURce]:AM:WIDeband:STATe?	✓	
[:SOURce]:AM[1]|2:EXTernal[1]|2:COUPling AC|DC [:SOURce]:AM[1]|2:EXTernal[1]|2:COUPling?	✓	
[:SOURce]:AM[1]|2:INTernal[1]:FREQuency <value><unit> [:SOURce]:AM[1]|2:INTernal[1]:FREQuency?	✓	
[:SOURce]:AM[1]|2:INTernal[1]:FREQuency:ALTernate <value><unit> [:SOURce]:AM[1]|2:INTernal[1]:FREQuency:ALTernate?	✓	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:AM[1]|2:INTernal[1]:FREQuency:ALTernate:AMPLitude: PERCent <value><unit> [:SOURce]:AM[1]|2:INTernal[1]:FREQuency:ALTernate:AMPLitude:PERCent?	✓	
[:SOURce]:AM[1]|2:INTernal[1]:FUNCtion:SHAPe <enum> [:SOURce]:AM[1]|2:INTernal[1]:FUNCtion:SHAPe?	✓	
[:SOURce]:AM[1]|2:INTernal[1]:SWEep:TIME <value><unit> [:SOURce]:AM[1]|2:INTernal[1]:SWEep:TIME?	✓	
[:SOURce]:AM[1]|2:INTernal[1]:SWEep:TRIGger <enum> [:SOURce]:AM[1]|2:INTernal[1]:SWEep:TRIGger?	✓	
[:SOURce]:AM[1]|2:SOURce INT[1]|EXT1|EXT2 [:SOURce]:AM[1]|2:SOURce?	✓	
[:SOURce]:AM[1]|2:STATe ON|OFF|1|0 [:SOURce]:AM[1]|2:STATe?	✓	
[:SOURce]:AM[1]|2[:DEPTh] <value><unit> [:SOURce]:AM[1]|2[:DEPTh]?	✓	
[:SOURce]:AM[1]|2[:DEPTh]:TRACk ON|OFF|1|0 [:SOURce]:AM[1]|2[:DEPTh]:TRACk?	✓	
AWGN ARB Subsystem		
[:SOURce]:RADio:AWGN:ARB...	–	*This subsystem is not supported.*
Bluetooth Subsystem		
[:SOURce]:RADio:BLUEtooth:ARB:...	–	*This subsystem is not supported.*
Calculate Subsystem		
:CALCulate:BERT:...	–	*This subsystem is not supported.*
CDMA ARB Subsystem		
[:SOURce]:RADio:CDMA:ARB:...	–	*This subsystem is not supported.*
CDMA2000 ARB Subsystem		

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:RADio:CDMA2000:ARB:...	–	*This subsystem is not supported.*
CDMA2000 BBG Subsystem		
[:SOURce]:RADio:CDMA2000[:BBG]:...	–	*This subsystem is not supported.*
Custom Subsystem		
[:SOURce]:RADio:CUSTom:...	–	*This subsystem is not supported.*
Data Subsystem		
:DATA:...	–	*This subsystem is not supported.*
Digital Modulation Subsystem		
[:SOURce]:BURSt:SOURce INTernal[1]	✓	*Supported but the following parameter is not supported:* EXTernal[1] *Supported but the following query is not supported:* [:SOURce]:BURSt:SOURce?
[:SOURce]:DM:BBFilter <value>|THRough [:SOURce]:DM:BBFilter?	✓	*Command accepted without error but does nothing.*
[:SOURce]:DM:EXTernal:POLarity NORMal|INVerted [:SOURce]:DM:EXTernal:POLarity?	✓	
[:SOURce]:BURSt:STATe ON|OFF|1|0 [:SOURce]:BURSt:STATe?	✓	
[:SOURce]:DM:EXTernal:ALC:BANDwidth|BWIDth NORMal|NARRow [:SOURce]:DM:EXTernal:ALC:BANDwidth|BWIDth?	✓	
[:SOURce]:DM:EXTernal:HICRest[:STATe] ON|OFF|1|0	✓	*Supported but the following query is not supported:* [:SOURce]:DM:EXTernal:HICRest[:STATe]?
[:SOURce]:DM:IQADjustment:Gain <value><unit> [:SOURce]:DM:IQADjustment:Gain?	✓	
[:SOURce]:DM:IQADjustment:IOFFset <value><unit> [:SOURce]:DM:IQADjustment:IOFFset?	✓	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:DM:IQADjustment:QOFFset <value><unit> [:SOURce]:DM:IQADjustment:QOFFset?	✓	
[:SOURce]:DM:IQADjustment:QSKew <value><unit> [:SOURce]:DM:IQADjustment:QSKew?	✓	
[:SOURce]:DM:IQADjustment[:STATe] ON|OFF|1|0 [:SOURce]:DM:IQADjustment[:STATe]?	✓	
[:SOURce]:DM:SOURce EXTernal|INTernal[1] [:SOURce]:DM:SOURce?	✓	
[:SOURce]:DM:STATe ON|OFF|1|0 [:SOURce]:DM:STATe?	✓	
Dmodulation Subsystem		
[:SOURce]:RADio:DMODulation:...	–	*This subsystem is not supported.*
Dect Subsystem		
[:SOURce]:RADio:DECT:ALPHa...	–	*This subsystem is not supported.*
Dual ARB Subsystem		
[:SOURce]:RADio:ARB:CLIPping "<file name>",IJQ|IORQ,<10-100%>	✓	
[:SOURce]:RADio:ARB:CLOCk:REFerence:EXTernal:FREQuency <value> [:SOURce]:RADio:ARB:CLOCk:REFerence:EXTernal:FREQuency?	–	
[:SOURce]:RADio:ARB:CLOCk:REFerence[:SOURce] INTernal|EXTernal [:SOURce]:RADio:ARB:CLOCk:REFerence[:SOURce]?	–	
[:SOURce]:RADio:ARB:CLOCk:SRATe <value> [:SOURce]:RADio:ARB:CLOCk:SRATe?	✓	
[:SOURce]:RADio:ARB:MARKer:CLEar "<file name>",<mkr(1|2)>,<first_point>, <last_point>	✓	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:RADio:ARB:MARKer:CLEar:ALL "<file name>",<mkr(1|2)>	✓	
[:SOURce]:RADio:ARB:MARKer:POLarity NEGative|POSitive [:SOURce]:RADio:ARB:MARKer:POLarity?	✓	
[:SOURce]:RADio:ARB:MARKer:RFBLank ON|OFF|1|0 [:SOURce]:RADio:ARB:MARKer:RFBLank?	✓	
[:SOURce]:RADio:ARB:MARKer[:SET] "<file name>",<mkr(1|2)>,<first_Point>, <last_point>,<skip_count>	✓	
[:SOURce]:RADio:ARB:RETRigger 1|0	✓	This command is not recommended; the following command is the preferred syntax for the ESG E44xxB.
[:SOURce]:RADio:ARB:RETRigger ON|OFF [:SOURce]:RADio:ARB:RETRigger?	✓	This query for the Agilent MXG Vector Signal Generator (N5182A) only returns the string ON or OFF. This is different from the ESG E44xxB query which returns a 1 or 0.
[:SOURce]:RADio:ARB:RFILter <value>|THRough	✓	This command performs no function within the Agilent MXG Vector Signal Generator (N5182A), however it is accepted when executed without errors. Notice that the query form of the command is not compatible.
[:SOURce]:RADio:ARB:RFILter?	–	
[:SOURce]:RADio:ARB:SCALing "<file name>",<1%–100%>	✓	
[:SOURce]:RADio:ARB:SEQuence "<file name>","<waveform>",<reps>,<mkr1(1|0)>, <mkr2(1|0)>{,"<waveform>",<rep>,<mkr1(1|0)>,<mkr2(1|0)>} [:SOURce]:RADio:ARB:SEQuence? "<file name>"	✓	
[:SOURce]:RADio:ARB:TRIGger:TYPE CONTinuous|SINGle|GATE|SADVance [:SOURce]:RADio:ARB:TRIGger:TYPE?	✓	

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:RADio:ARB:TRIGger:TYPE:GATE:ACTive LOW|HIGH [:SOURce]:RADio:ARB:TRIGger:TYPE:GATE:ACTive?	✓	
[:SOURce]:RADio:ARB:TRIGger[:SOURce] KEY|EXT|BUS [:SOURce]:RADio:ARB:TRIGger[:SOURce]?	✓	
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:DELay <value> [:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:DELay?	✓	
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:DELay:STATe ON|OFF|1|0 [:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:DELay:STATe?	✓	
[:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:SLOPe POSitive|NEGative [:SOURce]:RADio:ARB:TRIGger[:SOURce]:EXTernal:SLOPe?	✓	
[:SOURce]:RADio:ARB:WAVeform "<file name>" [:SOURce]:RADio:ARB:WAVeform?	✓	
[:SOURce]:RADio:ARB[:STATe] ON|OFF|1|0 [:SOURce]:RADio:ARB[:STATe]?	✓	
Edge Subsystem		
[:SOURce]:RADio:EDGE:...	–	*This subsystem is not supported.*
GSM Subsystem		
[:SOURce]:RADio:GSM:...	–	*This subsystem is not supported.*
Input Subsystem		
:INPut:BERT[:BASeband]:...	–	*This subsystem is not supported.*
Measure Subsystem		
:MEASure...	–	*This subsystem is not supported.*
Multi-Tone Subsystem		

Table 6-1 E44xxB Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:RADio:MTONe:ARB:...	–	*This subsystem is not supported.*
NADC Subsystem		
[:SOURce]:RADio:NADC:...	–	*This subsystem is not supported.*
PDC Subsystem		
[:SOURce]:RADio:PDC:...	–	*This subsystem is not supported.*
PHS Subsystem		
[:SOURce]:RADio:PHS:...	–	*This subsystem is not supported.*
Sense Subsystem		
:SENSe:BERT:...	–	*This subsystem is not supported.*
Tetra Subsystem		
[:SOURce]:RADio:TETRa:...	–	*This subsystem is not supported.*
Wideband CDMA ARB Subsystem		
[:SOURce]:RADio:WCDMa:TGPP:ARB:...	–	*This subsystem is not supported.*
Wideband CDMA BBG Subsystem		
[:SOURce]:RADio:WCDMa:TGPP[:BBG]:...	–	*This subsystem is not supported.*

E4428C/38C Compatible Commands

This section contains commands that have either been replaced or deleted from the *E4428C/38C ESG Signal Generators SCPI Command Reference*.

NOTE The Agilent MXG has only one AM path; and only one internal and one external source. If executed, the "2" path or "2" internal or external source commands will result in a "ERROR: -113, Undefined Header" to be generated in the signal generator.

The Agilent MXG has only one EXT path. If the EXT2 source command is executed, it will be accepted by the signal generator, but the results are the same as when the EXT[1] command is executed, and Ext Pulse will be selected in the Agilent MXG.

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
System Function Commands		
IEEE Common Commands		
*CLS	✓	
*ESE <data> *ESE?	✓	
*ESR?	✓	
*IDN?	✓	
*OPC *OPC?	✓	
*RCL <reg_num>	✓	
*RST	✓	
*SAV <reg_num>	✓	
*SRE <data> *SRE?	✓	
*STB?	✓	
*TRG	✓	
*TST?	✓	
*WAI	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
Calibration Subsystem		
:CALibration:DCFM	✓	
:CALibration:IQ	✓	
:CALibration:IQ:DC	✓	
:CALibration:IQ:DEFault	✓	
:CALibration:IQ:FULL	✓	
:CALibration:IQ:STARt <value><units>	✓	
:CALibration:IQ:STARt?	✓	
:CALibration:IQ:STOP <value><units>	✓	
:CALibration:IQ:STOP?	✓	
:CALibration:WBIQ	–	
:CALibration:WBIQ:DC	✓	
:CALibration:WBIQ:DEFault	✓	
:CALibration:WBIQ:FULL	✓	
:CALibration:WBIQ:STARt <value><units>	✓	
:CALibration:WBIQ:STARt?	✓	
:CALibration:WBIQ:STOP <value><units>	✓	
:CALibration:WBIQ:STOP?	✓	
Communication Subsystem		
:SYSTem:COMMunicate:GPIB:ADDRess <number>	✓	
:SYSTem:COMMunicate:GPIB:ADDRess?	✓	
:SYSTem:COMMunicate:GTLocal	✓	
:SYSTem:COMMunicate:LAN:CONFig DHCP|MANual	✓	
:SYSTem:COMMunicate:LAN:CONFig?	✓	
:SYSTem:COMMunicate:LAN:GATeway <ipstring>	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:SYSTem:COMMunicate:LAN:GATeway?	✓	
:SYSTem:COMMunicate:LAN:HOSTname <string>	✓	
:SYSTem:COMMunicate:LAN:HOSTname?	✓	
:SYSTem:COMMunicate:LAN:IP <ipstring>	✓	
:SYSTem:COMMunicate:LAN:IP?	✓	
:SYSTem:COMMunicate:LAN:SUBNet <ipstring>	✓	
:SYSTem:COMMunicate:LAN:SUBNet?	✓	
:SYSTem:COMMunicate:PMETer:ADDRess <value>	✓	*Command accepted without error but does nothing.*
:SYSTem:COMMunicate:PMETer:ADDRess?	✓	*Command accepted without error but does nothing.*
:SYSTem:COMMunicate:PMETer:CHANnel A|B	✓	*Command accepted without error but does nothing.*
:SYSTem:COMMunicate:PMETer:CHANnel?	✓	*Command accepted without error but does nothing.*
:SYSTem:COMMunicate:PMETer:IDN E4418B|E4419B|E4416A|E4417A	✓	*Command accepted without error but does nothing.*
:SYSTem:COMMunicate:PMETer:IDN?	✓	*Command accepted without error but does nothing.*
:SYSTem:COMMunicate:PMETer:TIMEout <num>[<time suffix>]	✓	*Command accepted without error but does nothing.*
:SYSTem:COMMunicate:PMETer:TIMEout?	✓	*Command accepted without error but does nothing.*
:SYSTem:COMMunicate:SERial:BAUD <number>	–	
:SYSTem:COMMunicate:SERial:BAUD?	–	
:SYSTem:COMMunicate:SERial:ECHO ON|OFF	–	
:SYSTem:COMMunicate:SERial:ECHO?	–	
:SYSTem:COMMunicate:SERial:RESet	–	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:SYSTem:COMMunicate:SERial:TOUT <value>	–	
:SYSTem:COMMunicate:SERial:TOUT?	–	
Diagnostic Subsystem		
:DIAGnostic[:CPU]:INFOrmation:BOARds?	–	
:DIAGnostic[:CPU]:INFOrmation:CCOunt:ATTenuator?	✓	
:DIAGnostic[:CPU]:INFOrmation:CCOunt:PON?	✓	
:DIAGnostic[:CPU]:INFOrmation:CCOunt:PROTection?	✓	
:DIAGnostic[:CPU]:INFOrmation:DISPlay:OTIMe?	✓	
:DIAGnostic[:CPU]:INFOrmation:LICense:AUXiliary?	✓	
:DIAGnostic[:CPU]:INFOrmation:LICense:WAVeform?	✓	
:DIAGnostic[:CPU]:INFOrmation:OPTions?	✓	
:DIAGnostic[:CPU]:INFOrmation:OPTions:DETail?	✓	
:DIAGnostic[:CPU]:INFOrmation:OTIMe?	✓	
:DIAGnostic[:CPU]:INFOrmation:REVision?	✓	
:DIAGnostic[:CPU]:INFOrmation:SDATe?	✓	
:DIAGnostic[:CPU]:INFOrmation:WLICense[:VALue]? <waveformType>	✓	
Memory Subsystem		
:MEMory:CATalog:BINary?	✓	
:MEMory:CATalog:BIT?	–	
:MEMory:CATalog:CDMa?	–	
:MEMory:CATalog:DMOD?	–	
:MEMory:CATalog:DWCDma?	–	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:MEMory:CATalog:FCDMa?	–	
:MEMory:CATalog:FSK?	–	
:MEMory:CATalog:IQ?	–	
:MEMory:CATalog:LIST?	–	
:MEMory:CATalog:MCDMa?	–	
:MEMory:CATalog:MDMod?	–	
:MEMory:CATalog:MDWCdma?	–	
:MEMory:CATalog:MFCdma?	–	
:MEMory:CATalog:MTONe?	–	
:MEMory:CATalog:FIR?	–	
:MEMory:CATalog:RCDMa?	–	
:MEMory:CATalog:SEQ?	✓	
:MEMory:CATalog:SHAPe?	–	
:MEMory:CATalog:STATe?	–	
:MEMory:CATalog:UFLT?	–	
:MEMory:CATalog:UPC?	–	
:MEMory:CATalog:UWCDma?	–	
:MEMory:CATalog[:ALL]?	✓	
:MEMory:COPY[:NAME] <"filename">,<"filename">	✓	
:MEMory:DATA <"filename">,<datablock>	✓	
:MEMory:DATA? <"filename">	✓	
:MEMory:DATA:APPend <"filename">,<datablock>	✓	
:MEMory:DATA:BIT <"filename">,<bit_count>,<datablock> :MEMory:DATA:BIT? <"filename">	–	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:MEMory:DATA:FIR <"filename">,osr,coefficient{,coefficient} :MEMory:DATA:FIR? <"filename">	–	
:MEMory:DATA:FSK <"filename">,num_states,f0,f0,...[,diff_state,num_diff_states,diff0,diff1,...]	–	
:MEMory:DATA:FSK? <"filename">	–	
:MEMory:DATA:PRAM[1]|2|3|4:FILE:BLOCk <"filename">,<datablock>	–	
:MEMory:DATA:PRAM[1]|2|3|4:FILE:LIST <"filename">,<uint8>[,<uint8>,<...>]	–	
:MEMory:DATA:IQ <"filename">,offsetQ,num_states,i0,q0,i1,q1,...[,diff_state,num_diff_states,diff0,diff1,...]	–	
:MEMory:DATA:IQ? <"filename">	–	
:MEMory:DATA:SHAPe <"filename">,num_rise_points,rp0,rp1,...num_fall_points,fp0,fp1,... :MEMory:DATA:SHAPe? <"filename">	–	
:MEMory:DATA:UNPRotected <"filename">,<datablock>	✓	
:MEMory:DELete:ALL	✓	
:MEMory:DELete:BINary	✓	
:MEMory:DELete:BIT	–	
:MEMory:DELete:CDMa	–	
:MEMory:DELete:DMOD	–	
:MEMory:DELete:DWCDma	–	
:MEMory:DELete:FCDMa	–	
:MEMory:DELete:FIR	–	
:MEMory:DELete:FSK	–	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:MEMory:DELete:IQ	–	
:MEMory:DELete:LIST	–	
:MEMory:DELete:MCDMa	–	
:MEMory:DELete:MDMod	–	
:MEMory:DELete:MDWCdma	–	
:MEMory:DELete:MFCdma	–	
:MEMory:DELete:MTONe	–	
:MEMory:DELete:RCDMa	–	
:MEMory:DELete:SEQ	✓	
:MEMory:DELete:SHAPe	–	
:MEMory:DELete:STATe	–	
:MEMory:DELete:UFLT	–	
:MEMory:DELete:UPC	–	
:MEMory:DELete:UWCDma	–	
:MEMory:DELete[:NAME] <"filename">	✓	
:MEMory:FREE[:ALL]?	✓	
:MEMory:LOAD:LIST <"filename">	✓	
:MEMory:MOVE <src_file>,<dest_file>	✓	
:MEMory:STATe:COMMent <reg_num>,<seq_num>,<"comment">	✓	
:MEMory:STATe:COMMent? <reg_num>,<seq_num>	✓	
:MEMory:STORe:LIST <"filename">	✓	
:MMEMory:CATalog? <"msus">	✓	
:MMEMory:COPY <"filename">,<"filename">	✓	
:MMEMory:DATA <"filename">,<datablock>	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:MMEMory:DATA? <"filename">	✓	
:MMEMory:DELete:NVWFm	✓	
:MMEMory:DELete:WFM	✓	
:MMEMory:DELete:WFM1	✓	
:MMEMory:DELete[:NAME] <"filename">,[<"msus">]	✓	
:MMEMory:HEADer:CLEar <filename>	✓	
:MMEMory:HEADer:DESCription <"filename">, <"description">	✓	
:MMEMory:HEADer:DESCription? <"filename">	✓	
:MMEMory:LOAD:LIST <"filename">	✓	
:MMEMory:MOVE <src_file>,<dest_file>	✓	
:MMEMory:STORe:LIST <"filename">	✓	
Output Subsystem		
:OUTPut:BLANking:AUTO ON|OFF|1|0 :OUTPut:BLANking:AUTO?	✓	
:OUTPut:BLANking[:STATe] ON|OFF|1|0 :OUTPut:BLANking[:STATe]?	✓	
:OUTPut:MODulation[:STATe] ON|OFF|1|0 :OUTPut:MODulation[:STATe]?	✓	
:OUTPut:PROTection[:STATe] ON|OFF|1|0 :OUTPut:PROTection[:STATe]?	✓	
:OUTPut:SETTled:POLarity NORMal|INVerted :OUTPut:SETTled:POLarity?	–	
:OUTPut:SETTled:RFOFf NORMal|INVerted :OUTPut:SETTled:RFOFf?	–	
:OUTPut:SETTled[:STATe]?	–	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:OUTPut[:STATe] ON|OFF|1|0 :OUTPut[:STATe]?	✓	
Route Subsystem		
:ROUTe:HARDware:DGENerator:...	–	*This subsystem is not supported.*
Status Subsystem		
:STATus:OPERation:BASeband:CONDition?	–	
:STATus:OPERation:BASeband:ENABle <value> :STATus:OPERation:BASeband:ENABle?	–	
:STATus:OPERation:BASeband:NTRansition <value> :STATus:OPERation:BASeband:NTRansition?	–	
:STATus:OPERation:BASeband:PTRansition <value> :STATus:OPERation:BASeband:PTRansition?	–	
:STATus:OPERation:BASeband[:EVENt]?	–	
:STATus:OPERation:CONDition?	✓	
:STATus:OPERation:ENABle <value> :STATus:OPERation:ENABle?	✓	
:STATus:OPERation:NTRansition <value> :STATus:OPERation:NTRansition?	✓	
:STATus:OPERation:PTRansition <value> :STATus:OPERation:PTRansition?	✓	
:STATus:OPERation[:EVENt]?	✓	
:STATus:PRESet:STATus:QUEStionable:CALibration :ENABle <value> :STATus:QUEStionable:CALibration:ENABle?	✓	
:STATus:QUEStionable:BERT:CONDition? :STATus:QUEStionable:BERT:ENABle <value> :STATus:QUEStionable:BERT:ENABle?	–	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:STATus:QUEStionable:BERT:NTRansition <value> :STATus:QUEStionable:BERT:NTRansition?	–	
:STATus:QUEStionable:BERT:PTRansition <value> :STATus:QUEStionable:BERT:PTRansition?	–	
:STATus:QUEStionable:BERT[:EVENt]?	–	
:STATus:QUEStionable:CALibration:CONDition?	–	
:STATus:QUEStionable:CALibration:NTRansition <value> :STATus:QUEStionable:CALibration:NTRansition?	✓	
:STATus:QUEStionable:CALibration:PTRansition <value> :STATus:QUEStionable:CALibration:PTRansition?	✓	
:STATus:QUEStionable:CALibration[:EVENt]? :STATus:QUEStionable:CONDition?	✓	
:STATus:QUEStionable:ENABle <value> :STATus:QUEStionable:ENABle?	✓	
:STATus:QUEStionable:FREQuency:CONDition?	✓	
:STATus:QUEStionable:FREQuency:ENABle <value> :STATus:QUEStionable:FREQuency:ENABle?	✓	
:STATus:QUEStionable:FREQuency:NTRansition <value> :STATus:QUEStionable:FREQuency:NTRansition?	✓	
:STATus:QUEStionable:FREQuency:PTRansition <value> :STATus:QUEStionable:FREQuency:PTRansition?	✓	
:STATus:QUEStionable:FREQuency[:EVENt]?	✓	
:STATus:QUEStionable:MODulation:CONDition?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:STATus:QUEStionable:MODulation:ENABle <value> :STATus:QUEStionable:MODulation:ENABle?	✓	
:STATus:QUEStionable:MODulation:NTRansition <value> :STATus:QUEStionable:MODulation:NTRansition?	✓	
:STATus:QUEStionable:MODulation:PTRansition <value> :STATus:QUEStionable:MODulation:PTRansition?	✓	
:STATus:QUEStionable:MODulation[:EVENt]?	✓	
:STATus:QUEStionable:NTRansition <value> :STATus:QUEStionable:NTRansition?	✓	
:STATus:QUEStionable:POWer:CONDition?	✓	
:STATus:QUEStionable:POWer:ENABle <value> :STATus:QUEStionable:POWer:ENABle?	✓	
:STATus:QUEStionable:POWer:NTRansition <value> :STATus:QUEStionable:POWer:NTRansition?	✓	
:STATus:QUEStionable:POWer:PTRansition <value> :STATus:QUEStionable:POWer:PTRansition?	✓	
:STATus:QUEStionable:POWer[:EVENt]?	✓	
:STATus:QUEStionable:PTRansition <value> :STATus:QUEStionable:PTRansition?	✓	
:STATus:QUEStionable[:EVENt]?	✓	
System Subsystem		
:SYSTem:CAPability?	✓	
:SYSTem:ERRor[:NEXT]?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:SYSTem:HELP:MODE SINGle	✓	*Supported but the following parameter is not supported:* CONTinuous *Supported but the following query is not supported:* :SYSTem:HELP:MODE?
:SYSTem:LANGuage "SCPI"|"COMP"|"8648" :SYSTem:LANGuage?	✓	*Supported but the following parameters are not supported:* "8340"|"8360"|"83712"|"83732"|"83752"|"8757"|"8662"|"8663"|"NADC"|"PDC"|"PHS"
:SYSTem:PON:TYPE PRESet|LAST :SYSTem:PON:TYPE?	✓	
:SYSTem:PRESet	✓	Always performs the same action as the **Preset** hardkey. For related **Preset** hardkey information, refer to ":SYSTem:PRESet:TYPE NORMal|USER :SYSTem:PRESet:TYPE?" on page 198
:SYSTem:PRESet:ALL	✓	
:SYSTem:PRESet:LANGuage "SCPI"|"COMP"|"8648" :SYSTem:PRESet:LANGuage?	✓	*Supported but the following parameters are not supported:* "8340"|"8360"|"83712"|"83732"|"83752"|"8757"|"8662"|"8663"|"NADC"|"PDC"|"PHS"
:SYSTem:PRESet:PERSistent	✓	
:SYSTem:PRESet:TYPE NORMal|USER :SYSTem:PRESet:TYPE?	✓	This command toggles the **Preset** hardkey state between factory- and user-defined conditions. The setting enabled by this command is not affected by signal generator power-on, preset, or *RST. **NOTE** If the **Preset** hardkey is not responding correctly, using the SCPI command: :SYSTem:PRESet:TYPE NORMal will return the Preset hardkey to its default factory behavior.

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:SYSTem:PRESet[:USER]:SAVE	✓	
:SYSTem:PRESet:PN9 NORMal|QUICk :SYSTem:PRESet:PN9?	–	
:SYSTem:SSAVer:DELay <value> :SYSTem:SSAVer:DELay?	✓	
:SYSTem:SSAVer:MODE LIGHt|TEXT :SYSTem:SSAVer:MODE?	✓	
:SYSTem:SSAVer:STATe ON|OFF :SYSTem:SSAVer:STATe?	✓	
:SYSTem:VERSion?	✓	
Trigger Subsystem		
:ABORt	✓	
:INITiate:CONTinuous[:ALL] ON|OFF|1|0 :INITiate:CONTinuous[:ALL]?	✓	
:INITiate[:IMMediate][:ALL]	✓	
:TRIGger:OUTPut:POLarity POSitive|NEGative :TRIGger:OUTPut:POLarity?	✓	
:TRIGger[:SEQuence]:SLOPe POSitive|NEGative :TRIGger[:SEQuence]:SLOPe?	✓	
:TRIGger[:SEQuence]:SOURce BUS|IMMediate|EXTernal|KEY :TRIGger[:SEQuence]:SOURce?	✓	
:TRIGger[:SEQuence][:IMMediate]	✓	
Unit Subsystem		
:UNIT:POWer DBM|DBUV|DBUVEMF|V|VEMF|DB :UNIT:POWer?	✓	
Amplitude Modulation Subsystem		

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:AM:INTernal:FREQuency:STEP[:INCRement] <num> [:SOURce]:AM:INTernal:FREQuency:STEP[:INCRement]?	✓	
[:SOURce]:AM:MODE DEEP	✓	*Command accepted without error but does nothing.*
[:SOURce]:AM:MODE NORMal [:SOURce]:AM:MODE?	✓	
[:SOURce]:AM:WIDeband:STATe OFF|0 [:SOURce]:AM:WIDeband:STATe?	✓	
[:SOURce]:AM[1]|2:EXTernal[1]|2:COUPling AC|DC [:SOURce]:AM[1]|2:EXTernal[1]|2:COUPling?	✓	
[:SOURce]:AM[1]|2:EXTernal[1]|2:IMPedance <50|600> [:SOURce]:AM[1]|2:EXTernal[1]|2:IMPedance?	✓	*Command accepted without error but does nothing.*
[:SOURce]:AM[1]|2:INTernal[1]:FREQuency:ALTernate <value><unit> [:SOURce]:AM[1]|2:INTernal[1]:FREQuency:ALTernate?	–	
[:SOURce]:AM[1]|2:INTernal[1]:FREQuency:ALTernate:AMPLitude:PERCent <value><unit> [:SOURce]:AM[1]|2:INTernal[1]:FREQuency:ALTernate:AMPLitude:PERCent?	–	
[:SOURce]:AM[1]|2:INTernal[1]|2:FUNCtion:SHAPe SINE [:SOURce]:AM[1]|2:INTernal[1]|2:FUNCtion:SHAPe?	✓	*Supported but the following parameters are not supported:* "TRIangle"|"SQUare"|"RAMP"|"NOISe"|"DUALsine"|"SWEPtsine"
[:SOURce]:AM[1]|2:INTernal[1]|2:FUNCtion:NOISe GAUSsian|UNIForm [:SOURce]:AM[1]|2:INTernal[1]|2:FUNCtion:NOISe?	–	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:AM[1]|2:INTernal[1]|2:FUNCtion:RAMP POSitive|NEGative [:SOURce]:AM[1]|2:INTernal[1]|2:FUNCtion:RAMP?	–	
[:SOURce]:AM[1]|2:INTernal[1]:SWEep:RATE <value><unit> [:SOURce]:AM[1]|2:INTernal[1]:SWEep:RATE?	–	
[:SOURce]:AM[1]|2:INTernal[1]:SWEep:TIME <value><unit> [:SOURce]:AM[1]|2:INTernal[1]:SWEep:TIME?	–	
[:SOURce]:AM[1]|2:INTernal[1]:SWEep:TRIGger IMMediate|KEY|EXTernal|BUS [:SOURce]:AM[1]|2:INTernal[1]:SWEep:TRIGger?	–	
[:SOURce]:AM[1]|2[:DEPTh]:EXPonential <value> [:SOURce]:AM[1]|2[:DEPTh]:EXPonential?	–	
[:SOURce]:AM[1]|2[:DEPTh][:LINear]:TRACk ON|OFF|1|0 [:SOURce]:AM[1]|2[:DEPTh][:LINear]:TRACk?	–	
[:SOURce]:AM[1]|2:INTernal[1]|2:FREQuency <value><unit>|UP|DOWN [:SOURce]:AM[1]|2:INTernal[1]|2:FREQuency?	✓	
[:SOURce]:AM[1]|2:POLarity NORMal|INVerted [:SOURce]:AM[1]|2:POLarity?	✓	
[:SOURce]:AM[1]|2:SOURce INT[1]|INT2|EXT[1]|EXT2 [:SOURce]:AM[1]|2:SOURce?	✓	*The Agilent MXG accepts the* EXT2 *parameter but only has a single external output and selects* EXT *on the signal generator if* EXT2 *is used.*
[:SOURce]:AM[1]|2:STATe ON|OFF|1|0 [:SOURce]:AM[1]|2:STATe?	✓	
[:SOURce]:AM[1]|2:TYPE LINear|EXPonential [:SOURce]:AM[1]|2:TYPE?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:AM:WIDeband:SENSitivity <value> [:SOURce]:AM:WIDeband:SENSitivity?	–	
[:SOURce]:AM:WIDeband:STATe ON|1	–	
[:SOURce]:AM[1]|2[:DEPTh][:LINear] <value><unit>|UP|DOWN [:SOURce]:AM[1]|2[:DEPTh][:LINear]?	✓	
[:SOURce]:AM[:DEPTh]:STEP[:INCRement] <value><unit> [:SOURce]:AM[:DEPTh]:STEP[:INCRement]?	✓	
Frequency Subsystem		
[:SOURce]:FREQuency:CENTer <num>[<freq suffix>]|MAXimum|MINimum|UP|DOWN [:SOURce]:FREQuency:CENTer? [MAXimum|MINimum]	✓	
[:SOURce]:FREQuency:CHANnels:BAND NBASe|NMOBile|BPGSm|MPGSm|BEGSm|MEGSm|BRGSm|MRGSm|GM450|M480|M850|B450|B480|B850BDCS|MDCS|BPCS|MPCS|B8|M8|B15|M15|B390|B420|B460|B915|M380|M410|M450|M870|PHS|DECT [:SOURce]:FREQuency:CHANnels:BAND?	✓	
[:SOURce]:FREQuency:CHANnels:NUMBer <number> [:SOURce]:FREQuency:CHANnels:NUMBer?	✓	
[:SOURce]:FREQuency:CHANnels[:STATe] ON|OFF|1|0 [:SOURce]:FREQuency:CHANnels[:STATe]?	✓	
[:SOURce]:FREQuency:FIXed <value><unit>|UP|DOWN [:SOURce]:FREQuency:FIXed?	✓	
[:SOURce]:FREQuency:MANual <value><unit> [:SOURce]:FREQuency:MANual?	–	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:FREQuency:MODE FIXed|CW|SWEep|LIST [:SOURce]:FREQuency:MODE?	✓	
[:SOURce]:FREQuency:MULTiplier <value> [:SOURce]:FREQuency:MULTiplier?	✓	
[:SOURce]:FREQuency:OFFSet <value><unit> [:SOURce]:FREQuency:OFFSet?	✓	
[:SOURce]:FREQuency:OFFSet:STATe ON|OFF [:SOURce]:FREQuency:OFFSet:STATe?	✓	
[:SOURce]:FREQuency:REFerence <value><unit> [:SOURce]:FREQuency:REFerence?	✓	
[:SOURce]:FREQuency:REFerence:SET	✓	
[:SOURce]:FREQuency:REFerence:STATe ON|OFF|1|0 [:SOURce]:FREQuency:REFerence:STATe?	✓	
[:SOURce]:FREQuency:SPAN <num>[<freq suffix>]|MAXimum|MINimum|UP|DOWN [:SOURce]:FREQuency:SPAN? [MAXimum|MINimum]	✓	
[:SOURce]:FREQuency:STARt <value><unit> [:SOURce]:FREQuency:STARt?	✓	
[:SOURce]:FREQuency:STOP <value><unit> [:SOURce]:FREQuency:STOP?	✓	
[:SOURce]:FREQuency:SYNThesis <value> [:SOURce]:FREQuency:SYNThesis?	–	
[:SOURce]:FREQuency[:CW] <value><unit>|UP|DOWN [:SOURce]:FREQuency[:CW]?	✓	
[:SOURce]:FREQuency[:CW]:STEP[:INCRement] <value><unit> [:SOURce]:FREQuency[:CW]:STEP[:INCRement]?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:FREQuency[:FIXed]:STEP[:INCRement] <value><unit> [:SOURce]:FREQuency[:FIXed]:STEP[:INCRement]?	✓	
[:SOURce]:PHASe:REFerence	✓	
[:SOURce]:PHASe[:ADJust] <value><unit> [:SOURce]:PHASe[:ADJust]?	✓	
[:SOURce]:ROSCillator:BANDwidth:DEFaults	–	
[:SOURce]:ROSCillator:BANDwidth:EXTernal <value> [:SOURce]:ROSCillator:BANDwidth:EXTernal?	–	
[:SOURce]:ROSCillator:BANDwidth:INTernal <value> [:SOURce]:ROSCillator:BANDwidth:INTernal?	–	
[:SOURce]:ROSCillator:SOURce?	✓	
[:SOURce]:ROSCillator:SOURce:AUTO ON|OFF|1|0 [:SOURce]:ROSCillator:SOURce:AUTO?	✓	
Frequency Modulation Subsystem		
[:SOURce]:FM:INTernal:FREQuency:STEP[:INCRement] <num> [:SOURce]:FM:INTernal:FREQuency:STEP[:INCRement]?	✓	
[:SOURce]:FM[1]|2:EXTernal[1]|2:COUPling AC|DC [:SOURce]:FM[1]|2:EXTernal[1]|2:COUPling?	✓	
[:SOURce]:FM[1]|2:EXTernal[1]|2:IMPedance <50|600> [:SOURce]:FM[1]|2:EXTernal[1]|2:IMPedance?	✓	*Command accepted without error but does nothing.*
[:SOURce]:FM[1]|2:INTernal[1]:FUNCtion:SHAPe SINE [:SOURce]:FM[1]|2:INTernal[1]:FUNCtion:SHAPe?	✓	*Supported but the following parameters are not supported:* TRIangle|SQUare|RAMP|NOISe|DUALsine|SWEPtsine

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:FM[1]|2:INTernal[1]|2:FREQuency <value><unit> [:SOURce]:FM[1]|2:INTernal[1]|2:FREQuency?	✓	
[:SOURce]:FM[1]|2:INTernal[1]:FREQuency:ALTernate <value><unit> [:SOURce]:FM[1]|2:INTernal[1]:FREQuency:ALTernate?	–	
[:SOURce]:FM[1]|2:INTernal[1]:FREQuency:ALTernate:AMPLitude:PERCent <value><unit> [:SOURce]:FM[1]|2:INTernal[1]:FREQuency:ALTernate:AMPLitude:PERCent?	–	
[:SOURce]:FM[1]|2:INTernal[1]|2:FUNCtion:NOISe GAUSsian|UNIForm [:SOURce]:FM[1]|2:INTernal[1]|2:FUNCtion:NOISe?	–	
[:SOURce]:FM[1]|2:INTernal[1]|2:FUNCtion:RAMP POSitive|NEGative [:SOURce]:FM[1]|2:INTernal[1]|2:FUNCtion:RAMP?	–	
[:SOURce]:FM[1]|2:INTernal[1]:SWEep:RATE <value><unit> [:SOURce]:FM[1]|2:INTernal[1]:SWEep:RATE?	–	
[:SOURce]:FM[1]|2:SOURce INT[1]|INT2|EXT[1]|EXT2 [:SOURce]:FM[1]|2:SOURce?	✓	*The Agilent MXG accepts the* EXT2 *parameter but only has a single external output and selects* EXT *on the signal generator if* EXT2 *is used.*
[:SOURce]:FM[1]|2:STATe ON|OFF|1|0 [:SOURce]:FM[1]|2:STATe?	✓	
[:SOURce]:FM[1]|2[:DEViation] <value><unit> [:SOURce]:FM[1]|2[:DEViation]?	✓	
[:SOURce]:FM[1]|2:INTernal[1]:SWEep:TIME <value><unit> [:SOURce]:FM[1]|2:INTernal[1]:SWEep:TIME?	–	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:FM[1]|2:INTernal[1]:SWEep:TRIGger IMMediate|KEY|EXTernal|BUS [:SOURce]:FM[1]|2:INTernal[1]:SWEep:TRIGger?	–	
[:SOURce]:FM[1]|2[:DEViation]:TRACk ON|OFF|1|0 [:SOURce]:FM[1]|2[:DEViation]:TRACk?	–	
List/Sweep Subsystem		
[:SOURce]:LIST:DIRection UP|DOWN [:SOURce]:LIST:DIRection?	✓	
[:SOURce]:LIST:DWELl <value>{,<value>} [:SOURce]:LIST:DWELl?	✓	
[:SOURce]:LIST:DWELl:POINts?	✓	
[:SOURce]:LIST:DWELl:TYPE LIST|STEP [:SOURce]:LIST:DWELl:TYPE?	✓	
[:SOURce]:LIST:FREQuency <value>{,<value>} [:SOURce]:LIST:FREQuency?	✓	
[:SOURce]:LIST:FREQuency:POINts?	✓	
[:SOURce]:LIST:MANual <value>|UP|DOWN [:SOURce]:LIST:MANual?	✓	
[:SOURce]:LIST:MODE AUTO|MANual [:SOURce]:LIST:MODE?	✓	
[:SOURce]:LIST:POWer <value>{,<value>} [:SOURce]:LIST:POWer?	✓	
[:SOURce]:LIST:POWer:POINts?	✓	
[:SOURce]:LIST:RETRace ON|OFF|0|1 [:SOURce]:LIST:RETRace?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:LIST:TRIGger:SOURce BUS|IMMediate|EXTernal|KEY [:SOURce]:LIST:TRIGger:SOURce?	✓	
[:SOURce]:LIST:TYPE LIST|STEP [:SOURce]:LIST:TYPE?	✓	
[:SOURce]:LIST:TYPE:LIST:INITialize:FSTep	✓	
[:SOURce]:LIST:TYPE:LIST:INITialize:PRESet	✓	
[:SOURce]:SWEep:CONTrol:STATe ON|OFF|1|0 [:SOURce]:SWEep:CONTrol:STATe?	–	
[:SOURce]:SWEep:CONTrol:TYPE MASTer|SLAVe [:SOURce]:SWEep:CONTrol:TYPE?	–	
[:SOURce]:SWEep:DWELl <value> [:SOURce]:SWEep:DWELl?	✓	
[:SOURce]:SWEep:GENeration ANALog|STEPped [:SOURce]:SWEep:GENeration?	✓	*Command accepted without error but does nothing.*
[:SOURce]:SWEep:MODE AUTO|MANual [:SOURce]:SWEep:MODE?	✓	
[:SOURce]:SWEep:POINts <value> [:SOURce]:SWEep:POINts?	✓	
[:SOURce]:SWEep:TIME 10mS - 99S [:SOURce]:SWEep:TIME?	✓	
[:SOURce]:SWEep:TIME:AUTO ON|OFF|0|1 [:SOURce]:SWEep:TIME:AUTO?	✓	
Low Frequency Output Subsystem		
[:SOURce]:LFOutput:...	–	*This subsystem is not supported.*
Phase Modulation Subsystem		

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:PM[1]|2:INTernal[1]:FREQuency:ALTernate <value><unit> [:SOURce]:PM[1]|2:INTernal[1]:FREQuency:ALTernate?	–	
[:SOURce]:PM[1]|2:INTernal[1]:FREQuency:ALTernate:AMPLitude:PERCent <value><unit> [:SOURce]:PM[1]|2:INTernal[1]:FREQuency:ALTernate:AMPLitude:PERCent?	–	
[:SOURce]:PM:INTernal:FREQuency:STEP[:INCRement] [:SOURce]:PM:INTernal:FREQuency:STEP[:INCRement]?	✓	
[:SOURce]:PM[1]|2:BANDwidth|BWIDth NORMal|HIGH [:SOURce]:PM[1]|2:BANDwidth|BWIDth?	✓	
[:SOURce]:PM[1]|2:EXTernal[1]:COUPling AC|DC [:SOURce]:PM[1]|2:EXTernal[1]:COUPling?	✓	
[:SOURce]:PM[1]|2:EXTernal[1]|2:IMPedance <50|600> [:SOURce]:PM[1]|2:EXTernal[1]|2:IMPedance?	✓	
[:SOURce]:PM[1]|2:INTernal[1]|2:FUNCtion:NOISe GAUSsian|UNIForm [:SOURce]:PM[1]|2:INTernal[1]|2:FUNCtion:NOISe?	–	
[:SOURce]:PM[1]|2:INTernal[1]|2:FUNCtion:RAMP POSitive|NEGative [:SOURce]:PM[1]|2:INTernal[1]|2:FUNCtion:RAMP?	–	
[:SOURce]:PM[1]|2:INTernal[1]:FUNCtion:SHAPe SINE [:SOURce]:PM[1]|2:INTernal[1]:FUNCtion:SHAPe?	✓	*Supported but the following parameters are not supported:* TRIangle|SQUare|RAMP|NOISe|DUALsine|SWEPtsine

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:PM[1]|2:INTernal[1]|2:FREQuency <value><unit> [:SOURce]:PM[1]|2:INTernal[1]|2:FREQuency?	✓	
[:SOURce]:PM[1]|2:INTernal[1]:SWEep:RATE <value><unit> [:SOURce]:PM[1]|2:INTernal[1]:SWEep:RATE?	–	
[:SOURce]:PM[1]|2:INTernal[1]:SWEep:TIME <value><unit> [:SOURce]:PM[1]|2:INTernal[1]:SWEep:TIME?	–	
[:SOURce]:PM[1]|2:INTernal[1]:SWEep:TRIGger IMMediate|KEY|EXTernal|BUS [:SOURce]:PM[1]|2:INTernal[1]:SWEep:TRIGger?	–	
[:SOURce]:PM[1]|2[:DEViation]:TRACk ON|OFF|1|0 [:SOURce]:PM[1]|2[:DEViation]:TRACk?	–	
[:SOURce]:PM[1]|2:SOURce INT[1]|INT2|EXT[1]|EXT2 [:SOURce]:PM[1]|2:SOURce?	✓	*The Agilent MXG accepts the* EXT2 *parameter but only has a single external output and selects* EXT *on the signal generator if* EXT2 *is used.*
[:SOURce]:PM[1]|2:STATe ON|OFF|1|0 [:SOURce]:PM[1]|2:STATe?	✓	
[:SOURce]:PM[1]|2[:DEViation] <value><unit> [:SOURce]:PM[1]|2[:DEViation]?	✓	
[:SOURce]:PM[:DEViation]:STEP[:INCRement] <value><unit> [:SOURce]:PM[:DEViation]:STEP[:INCRement]?	✓	
Power Subsystem		
[:SOURce]:POWer:ALC:BANDwidth|BWIDth <num>[freq suffix] [:SOURce]:POWer:ALC:BANDwidth|BWIDth?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:POWer:ALC:BANDwidth|BWIDth:AUTO ON|OFF|1|0 [:SOURce]:POWer:ALC:BANDwidth|BWIDth:AUTO?	✓	
[:SOURce]:POWer:ALC:LEVel <value>dB [:SOURce]:POWer:ALC:LEVel?	✓	
[:SOURce]:POWer:ALC:SEARch ON|OFF|1|0|ONCE [:SOURce]:POWer:ALC:SEARch?	✓	
[:SOURce]:POWer:ALC:SEARch:REFerence FIXed|MODulated [:SOURce]:POWer:ALC:SEARch:REFerence?	✓	
[:SOURce]:POWer:ALC:SEARch:SPAN:POINts <value> [:SOURce]:POWer:ALC:SEARch:SPAN:POINts?	✓	
[:SOURce]:POWer:ALC:SEARch:SPAN:STARt <value><units> [:SOURce]:POWer:ALC:SEARch:SPAN:STARt?	✓	
[:SOURce]:POWer:ALC:SEARch:SPAN:STOP <value><units> [:SOURce]:POWer:ALC:SEARch:SPAN:STOP?	✓	
[:SOURce]:POWer:ALC:SEARch:SPAN:TYPE FULL|USER [:SOURce]:POWer:ALC:SEARch:SPAN:TYPE?	✓	
[:SOURce]:POWer:ALC:SEARch:SPAN[:STATe] ON|OFF|1|0 [:SOURce]:POWer:ALC:SEARch:SPAN[:STATe]?	✓	
[:SOURce]:POWer:ALC:SOURce INTernal|DIODe|MMHead [:SOURce]:POWer:ALC:SOURce?	✓	*Supported but the following parameters are not supported:* DIODe|MMHead
[:SOURce]:POWer:ALC:SOURce:EXTernal:COUPling (0dB-32dB) [:SOURce]:POWer:ALC:SOURce:EXTernal:COUPling?	–	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:POWer:ALC[:STATe] ON|OFF|1|0 [:SOURce]:POWer:ALC[:STATe]?	✓	
[:SOURce]:POWer:ALTernate:AMPLitude <value>dB [:SOURce]:POWer:ALTernate:AMPLitude?	–	
[:SOURce]:POWer:ALTernate:MANual MAIN|DELTa [:SOURce]:POWer:ALTernate:MANual?	–	
[:SOURce]:POWer:ALTernate:STATe ON|OFF|1|0 [:SOURce]:POWer:ALTernate:STATe?	–	
[:SOURce]:POWer:ALTernate:TRIGger[:SOURce] INTernal|EXTernal|MANual [:SOURce]:POWer:ALTernate:TRIGger[:SOURce]?	–	
[:SOURce]:POWer:ATTenuation <value><unit> [:SOURce]:POWer:ATTenuation?	✓	
[:SOURce]:POWer:ATTenuation:AUTO ON|OFF|1|0 [:SOURce]:POWer:ATTenuation:AUTO?	✓	
[:SOURce]:POWer:MODE FIXed|LIST|SWEep [:SOURce]:POWer:MODE?	✓	
[:SOURce]:POWer:PROTection[:STATe] ON|OFF|1|0 [:SOURce]:POWer:PROTection[:STATe]?	✓	
[:SOURce]:POWer:REFerence <value><unit> [:SOURce]:POWer:REFerence?	✓	
[:SOURce]:POWer:REFerence:STATe ON|OFF|1|0 [:SOURce]:POWer:REFerence:STATe?	✓	
[:SOURce]:POWer:STARt <value><unit> [:SOURce]:POWer:STARt?	✓	
[:SOURce]:POWer:STOP <value><unit> [:SOURce]:POWer:STOP?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:POWer[:LEVel][:IMMediate]:OFFSet <value><unit> [:SOURce]:POWer[:LEVel][:IMMediate]:OFFSet?	✓	
[:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude] <value><unit> |UP|DOWN [:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude]?	✓	
[:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude]:STEP[:INCRement] <value> [:SOURce]:POWer[:LEVel][:IMMediate][:AMPLitude]:STEP[:INCRement]?	✓	
Pulse Modulation Subsystem		
[:SOURce]:PULM:EXTernal:POLarity NORMal|INVerted [:SOURce]:PULM:EXTernal:POLarity?	✓	
[:SOURce]:PULM:INTernal[1]:DELay <delay>|UP|DOWN [:SOURce]:PULM:INTernal[1]:DELay? [UP|DOWN]	✓	
[:SOURce]:PULM:INTernal[1]:DELay:STEP <step> [:SOURce]:PULM:INTernal[1]:DELay:STEP?	✓	
[:SOURce]:PULM:INTernal[1]:FREQuency <frequency>|MAXimum|MINimum|UP|DOWN [:SOURce]:PULM:INTernal[1]:FREQuency?	✓	
[:SOURce]:PULM:INTernal[1]:FREQuency:STEP[:INCRement] <freq>|MAXimum|MINimum|DEFault [:SOURce]:PULM:INTernal[1]:FREQuency:STEP[:INCRement]? [MIN|MAX|DEF]	✓	
[:SOURce]:PULM:INTernal[1]:FUNCtion:SHAPe SQUare|PULSe [:SOURce]:PULM:INTernal[1]:FUNCtion:SHAPe?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:PULM:INTernal[1]:PERiod <period>|MAXimum|MINimum|UP|DOWN [:SOURce]:PULM:INTernal[1]:PERiod?	✓	
[:SOURce]:PULM:INTernal[1]:PERiod:STEP[:INCRement] <step>|UP|DOWN [:SOURce]:PULM:INTernal[1]:PERiod:STEP[:INCRement]?	✓	
[:SOURce]:PULM:INTernal[1]:PWIDth <width> [:SOURce]:PULM:INTernal[1]:PWIDth?	✓	
[:SOURce]:PULM:INTernal[1]:PWIDth:STEP <step>|DEFault|MAXimum|MINimum [:SOURce]:PULM:INTernal[1]:PWIDth:STEP?	✓	
[:SOURce]:PULM:SOURce INT|EXT[1]|EXT2 [:SOURce]:PULM:SOURce?	✓	*Supported but the following parameters are not supported:* |SCALar| *Also, the Agilent MXG accepts the* EXT2 *parameter but only has a single external output and selects* EXT *on the signal generator if* EXT2 *is used.*
[:SOURce]:PULM:SOURce:INTernal SQUare|FRUN|TRIGgered|DOUBlet|GATEd [:SOURce]:PULM:SOURce:INTernal?	✓	
[:SOURce]:PULM:STATe ON|OFF|1|0 [:SOURce]:PULM:STATe?	✓	
Digital Function Commands		
All Modulation Subsystem		
[:SOURce]:RADio[1]:ALL:OFF	✓	
AWGN ARB Subsystem		
[:SOURce]:RADio[1]:AWGN...	–	*This subsystem is not supported.*
AWGN Real Time Subsystem		

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:RADio:AWGN:RT:BWIDth <val> [:SOURce]:RADio:AWGN:RT:BWIDth?	✓	
[:SOURce]:RADio:AWGN:RT[:STATe] ON|OFF|1|0 [:SOURce]:RADio:AWGN:RT[:STATe]?	✓	
Bluetooth Subsystem		
[:SOURce]:RADio[1]:BLUEtooth:ARB:...	–	*This subsystem is not supported.*
Calculate Subsystem		
:CALCulate:BERT:BTS:LOOPback:...	–	*This subsystem is not supported.*
CDMA ARB Subsystem		
[:SOURce]:RADio[1]:CDMA:ARB:...	–	*This subsystem is not supported.*
Custom Subsystem		
[:SOURce]:RADio[1]:CUSTom:...	–	*This subsystem is not supported.*
Data Subsystem		
:DATA:BERT:...	–	*This subsystem is not supported.*
DECT Subsystem		
[:SOURce]:RADio[1]:DECT:...	–	*This subsystem is not supported.*
Dmodulation Subsystem		
[:SOURce]:RADio[1]:DMODulation:ARB:...	–	*This subsystem is not supported.*
Digital Subsystem		
:DIGital...	–	*This subsystem is not supported.*
Digital Modulation Subsystem		
[:SOURce]:BURSt:SOURce INTernal[1]	✓	*Supported but the following parameter is not supported:* EXTernal[1] *Supported but the following query is not supported:* [:SOURce]:BURSt:SOURce?

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:BURSt:STATe ON|OFF|1|0 [:SOURce]:BURSt:STATe?	✓	
[:SOURce]:BURSt:TYPE:EXT LOG|LIN [:SOURce]:BURSt:TYPE:EXT?	–	
[:SOURce]:BURSt:TYPE:INT LOG|LIN [:SOURce]:BURSt:TYPE:INT?	–	
[:SOURce]:BURSt:TYPE[:ALL] LOG|LIN	–	
[:SOURce]:DM:EXTernal:FILTer 40e6|THRough	✓	*Commands are accepted without error by the signal generator, but no action occurs. But the following query is not supported:* [:SOURce]:DM:EXTernal:FILTer? If the query is used, an
[:SOURce]:DM:EXTernal:FILTer:AUTO ON|OFF|1|0	✓	Supported but the following query is not supported: [:SOURce]:DM:EXTernal:FILTer:AUTO?
[:SOURce]:DM:EXTernal:HCRest[:STATe] ON|OFF|1|0 [:SOURce]:DM:EXTernal:HCRest[:STATe]?	✓	
[:SOURce]:DM:EXTernal:POLarity NORMal|INVert|INVerted [:SOURce]:DM:EXTernal:POLarity?	✓	
[:SOURce]:DM:EXTernal:SOURce EXTernal|INTernal|BBG1|OFF|SUM [:SOURce]:DM:EXTernal:SOURce?	✓	*Supported but the following parameters are not supported:* |BBG2|BBG3|BBG4|EXT600|
[:SOURce]:DM:IQADjustment:BBG[1]|2:DELay <value><unit> [:SOURce]:DM:IQADjustment:BBG[1]|2:DELay?	✓	
[:SOURce]:DM:IQADjustment:BBG[1]|2:DELay:EVENts ON|OFF|1|0 [:SOURce]:DM:IQADjustment:BBG[1]|2:DELay:EVENts?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:DM:IQADjustment:BBG[1]|2:SKEW:PATH {RF}|BB [:SOURce]:DM:IQADjustment:BBG[1]|2:SKEW:PATH?	✓	
[:SOURce]:DM:IQADjustment:BBG[1]|2:SKEW[:DELay] <value><unit> [:SOURce]:DM:IQADjustment:BBG[1]|2:SKEW[:DELay]?	✓	
[:SOURce]:DM:IQADjustment:EXTernal:COFFset <value><unit> [:SOURce]:DM:IQADjustment:EXTernal:COFFset?	✓	
[:SOURce]:DM:IQADjustment:EXTernal:DIOFfset <value><unit> [:SOURce]:DM:IQADjustment:EXTernal:DIOFfset?	✓	
[:SOURce]:DM:IQADjustment:EXTernal:DQOFfset <value><unit> [:SOURce]:DM:IQADjustment:EXTernal:DQOFfset?	✓	
[:SOURce]:DM:IQADjustment:EXTernal:IOFFset <value><unit> [:SOURce]:DM:IQADjustment:EXTernal:IOFFset?	✓	
[:SOURce]:DM:IQADjustment:EXTernal:IQATten <value><unit> [:SOURce]:DM:IQADjustment:EXTernal:IQATten?	–	
[:SOURce]:DM:IQADjustment:EXTernal:QOFFset <value><unit> [:SOURce]:DM:IQADjustment:EXTernal:QOFFset?	✓	
[:SOURce]:DM:IQADjustment:GAIN? [:SOURce]:DM:IQADjustment:GAIN[1|2] <value><unit>	–	
[:SOURce]:DM:IQADjustment:IOFFset <value><unit> [:SOURce]:DM:IQADjustment:IOFFset?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:DM:IQADjustment:QOFFset <value><unit> [:SOURce]:DM:IQADjustment:QOFFset?	✓	
[:SOURce]:DM:IQADjustment:QSKew <value><unit> [:SOURce]:DM:IQADjustment:QSKew?	✓	
[:SOURce]:DM:IQADjustment[:STATe] ON|OFF|1|0 [:SOURce]:DM:IQADjustment[:STATe]?	✓	
[:SOURce]:DM:MODulation:ATTen <value><unit> [:SOURce]:DM:MODulation:ATTen?	✓	
[:SOURce]:DM:MODulation:ATTen:AUTO ON|OFF|1|0 [:SOURce]:DM:MODulation:ATTen:AUTO?	✓	
[:SOURce]:DM:MODulation:ATTen:EXTernal DEFault|MANual|MEASure [:SOURce]:DM:MODulation:ATTen:EXTernal?	✓	
[:SOURce]:DM:MODulation:ATTen:EXTernal:LEVel <value> <volt_units> [:SOURce]:DM:MODulation:ATTen:EXTernal:LEVel?	✓	
[:SOURce]:DM:MODulation:ATTen:EXTernal:LEVel:MEASurement	✓	
[:SOURce]:DM:MODulation:ATTen:OPTimize:BANDwidth <value> <bw_rate_units> [:SOURce]:DM:MODulation:ATTen:OPTimize:BANDwidth?	✓	
[:SOURce]:DM:MODulation:FILTer THRough	✓	*Supported but the following query generates an* `error: –113, Undefined header`: [:SOURce]:DM:MODulation:FILTer?
[:SOURce]:DM:MODulation:FILTer:AUTO ON|OFF|1|0|2.1e6|40e6| [:SOURce]:DM:MODulation:FILTer:AUTO?	✓	*Commands are accepted by the signal generator, but no action is taken.* (An `error –113, Undefined header` *will be displayed on the signal generator.*)

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:DM:POLarity[:ALL] NORMal|INVert|INVerted [:SOURce]:DM:POLarity?	✓	
[:SOURce]:DM:SKEW:PATH RF|BB [:SOURce]:DM:SKEW:PATH?	–	
[:SOURce]:DM:SKEW[:STATe] ON|OFF|1|0| [:SOURce]:DM:SKEW[:STATe]?	✓	
[:SOURce]:DM:SOURce[1]|2 EXTernal|INTernal|BBG1|OFF [:SOURce]:DM:SOURce?	✓	*Supported but the following parameters are not supported:* |BBG2|BBG3|BBG4|EXT600|
[:SOURce]:DM:SRATio <value><unit> [:SOURce]:DM:SRATio?	✓	
[:SOURce]:DM:STATe ON|OFF|1|0 [:SOURce]:DM:STATe?	✓	
Display Subsystem		
:DISPlay:ANNotation:AMPLitude:UNIT DBM|DBUV|DBUVEMF|V|VEMF|DB :DISPlay:ANNotation:AMPLitude:UNIT?	✓	
:DISPlay:ANNotation:CLOCk:DATE:FORMat MDY|DMY :DISPlay:ANNotation:CLOCk:DATE:FORMat?	✓	
:DISPlay:ANNotation:CLOCk[:STATe] ON|OFF|1|0 :DISPlay:ANNotation:CLOCk[:STATe]?	✓	
:DISPlay:BRIGhtness <value> :DISPlay:BRIGhtness?	✓	
:DISPlay:CAPTure	✓	
:DISPlay:CONTrast <value> :DISPlay:CONTrast?	✓	
:DISPlay:INVerse ON|OFF|1|0	✓	Supported but the following query is not supported: :DISPlay:INVerse?

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
:DISPlay:REMote ON|OFF|1|0 :DISPlay:REMote?	✓	
:DISPlay[:WINDow][:STATe] ON|OFF|1|0 :DISPlay[:WINDow][:STATe]?	✓	
Dual ARB Subsystem		
[:SOURce]:RADio2:ARB:VCO:INTernal:SOURce:BBG1 ON|OFF|1|0 [:SOURce]:RADio2:ARB:VCO:INTernal:SOURce:BBG1?	–	
[:SOURce]:RADio[1]:ARB:CLIPping "filename",IJQ|IORQ,<10-100%>[,<10-100%>]	✓	
[:SOURce]:RADio[1]:ARB:DACS:ALIGn	✓	
[:SOURce]:RADio[1]:ARB:GENerate:SINE ["filename"],[<osr>],[<scale>],[I|Q|{IQ}]	✓	
[:SOURce]:RADio[1]:ARB:HEADer:CLEar	✓	
[:SOURce]:RADio[1]:ARB:HEADer:RMS <"filename">, <rms:0 - 1.414213562373095>|UNSPecified [:SOURce]:RADio[1]:ARB:HEADer:RMS? <"filename">	✓	
[:SOURce]:RADio[1]:ARB:HEADer:SAVE	✓	
[:SOURce]:RADio[1]:ARB:HCRest[:STATe] ON|OFF|1|0 [:SOURce]:RADio[1]:ARB:HCRest[:STATe]?	✓	*Command accepted without error but does nothing.*
[:SOURce]:RADio[1]:ARB:IQ:EXTernal:FILTer 40e6|THRough	✓	*Commands are accepted by the signal generator, but no action is taken. But the following query is not supported and generates an* ERROR: –113, Undefined header: [:SOURce]:RADio[1]:ARB:IQ:EXTernal:FILTer?

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:RADio[1]:ARB:IQ:EXTernal:FILTer:AUTO ON|OFF|1|0	✓	*Commands are accepted by the signal generator, but no action is taken. But the following query is not supported and generates an* ERROR: –113, Undefined header: [:SOURce]:RADio[1]:ARB:IQ:EXTernal:FILTer:AUTO?
[:SOURce]:RADio[1]:ARB:IQ:MODulation:ATTen <value><unit> [:SOURce]:RADio[1]:ARB:IQ:MODulation:ATTen?	✓	
[:SOURce]:RADio[1]:ARB:IQ:MODulation:ATTen:AUTO ON|OFF|1|0 [:SOURce]:RADio[1]:ARB:IQ:MODulation:ATTen:AUTO?	✓	
[:SOURce]:RADio[1]:ARB:IQ:MODulation:FILTer 2.1e6|40e6|THRough	✓	*Commands are accepted by the signal generator, but no action is taken. But the following query is not supported and generates an* ERROR: –113, Undefined header: [:SOURce]:RADio[1]:ARB:IQ:MODulation:FILTer?
[:SOURce]:RADio[1]:ARB:IQ:MODulation:FILTer:AUTO ON|OFF|1|0	✓	*Commands are accepted by the signal generator, but no action is taken. But the following query is not supported and generates an* ERROR: –113, Undefined header: [:SOURce]:RADio[1]:ARB:IQ:MODulation:FILTer:AUTO?
[:SOURce]:RADio[1]:ARB:MARKer:CLEar "filename",<mkr(1|2|3|4)>,<first_Point>,<last_point>	✓	
[:SOURce]:RADio[1]:ARB:MARKer:CLEar:ALL "filename",<mkr(1|2|3|4)>	✓	
[:SOURce]:RADio[1]:ARB:MARKer:ROTate "filename",<rotate_count>	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG – = Not supported by Agilent MXG	N51xxA	Remarks
[:SOURce]:RADio[1]:ARB:MARKer:[SET] "filename",<mkr(1|2|3|4)>,<first_Point>,<last_point>,<skip_count>	✓	
[:SOURce]:RADio[1]:ARB:MDEStination:AAMPlitude {NONE}|M1|M2|M3|M4 [:SOURce]:RADio[1]:ARB:MDEStination:AAMPlitude?	–	
[:SOURce]:RADio[1]:ARB:MDEStination:ALCHold {NONE}|M1|M2|M3|M4 [:SOURce]:RADio[1]:ARB:MDEStination:ALCHold?	✓	
[:SOURce]:RADio[1]:ARB:MDEStination:PULSe {NONE}|M1|M2|M3|M4 [:SOURce]:RADio[1]:ARB:MDEStination:PULSe?	✓	
[:SOURce]:RADio[1]:ARB:MPOLarity:MARKer1 NEGative|{POSitive} [:SOURce]:RADio[1]:ARB:MPOLarity:MARKer1?	✓	
[:SOURce]:RADio[1]:ARB:MPOLarity:MARKer2 NEGative|{POSitive} [:SOURce]:RADio[1]:ARB:MPOLarity:MARKer2?	✓	
[:SOURce]:RADio[1]:ARB:MPOLarity:MARKer3 NEGative|{POSitive} [:SOURce]:RADio[1]:ARB:MPOLarity:MARKer3?	✓	
[:SOURce]:RADio[1]:ARB:MPOLarity:MARKer4 NEGative|{POSitive} [:SOURce]:RADio[1]:ARB:MPOLarity:MARKer4?	✓	
[:SOURce]:RADio[1]:ARB:NOISe:BFACtor <1 - 2{1}> [:SOURce]:RADio[1]:ARB:NOISe:BFACtor?	✓	
[:SOURce]:RADio[1]:ARB:NOISe:CBWidth <1Hz-80Mhz {1Hz}> [:SOURce]:RADio[1]:ARB:NOISe:CBWidth?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:RADio[1]:ARB:NOISe:CN <-100dB - 100dB {0dB}> [:SOURce]:RADio[1]:ARB:NOISe:CN?	✓	
[:SOURce]:RADio[1]:ARB:NOISe[:STATe] ON|{OFF}|1|0 [:SOURce]:RADio[1]:ARB:NOISe[:STATe]?	✓	
[:SOURce]:RADio[1]:ARB:REFerence:EXTernal:FREQuency <value> [:SOURce]:RADio[1]:ARB:REFerence:EXTernal:FREQuency?	–	
[:SOURce]:RADio[1]:ARB:REFerence[:SOURce] INTernal|EXTernal [:SOURce]:RADio[1]:ARB:REFerence[:SOURce]?	✓	
[:SOURce]:RADio[1]:ARB:RETRigger ON|OFF|IMMediate [:SOURce]:RADio[1]:ARB:RETRigger?	✓	
[:SOURce]:RADio[1]:ARB:RSCaling <1%-100%> [:SOURce]:RADio[1]:ARB:RSCaling?	✓	
[:SOURce]:RADio[1]:ARB:SCALing "filename",<1%-100%>	✓	
[:SOURce]:RADio[1]:ARB:SCLock:RATE <1Hz - 100MHz {100MHz}> [:SOURce]:RADio[1]:ARB:SCLock:RATE?	✓	*Agilent MXG range is 1 kHz – 125 MHz with a default of 125 MHz.*
[:SOURce]:RADio[1]:ARB:SEQuence[:MWAVeform] <filename>,<waveform>,<reps>,NONE|M1|M2|M3|M4|M1M2|M1M3|M1M4|M2M3|M2M4|M3M4|M1M2M3|M1M2M4|M1M3M4|M2M3M4|M1M2M3M4|ALL,{,<waveform>,<reps>,NONE|M1|M2|M3|M4|M1M2|M1M3|M1M4|M2M3|M2M4|M3M4|M1M2M3|M1M2M4|M1M3M4|M2M3M4|M1M2M3M4|ALL,} [:SOURce]:RADio[1]:ARB:SEQuence[:MWAVeform]? <filename>	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:RADio[1]:ARB:TRIGger:TYPE CONTinuous|SINGle|GATE|SADVance [:SOURce]:RADio[1]:ARB:TRIGger:TYPE?	✓	
[:SOURce]:RADio[1]:ARB:TRIGger:TYPE:CONTinuous [:TYPE] FREE|TRIGger|RESet [:SOURce]:RADio[1]:ARB:TRIGger:TYPE:CONTinuous [:TYPE]?	✓	
[:SOURce]:RADio[1]:ARB:TRIGger:TYPE:GATE LOW|HIGH [:SOURce]:RADio[1]:ARB:TRIGger:TYPE:GATE?	✓	
[:SOURce]:RADio[1]:ARB:TRIGger:TYPE:SADVance:SORDer LINear|DYNamic [:SOURce]:RADio[1]:ARB:TRIGger:TYPE:SADVance:SORDer?	–	
[:SOURce]:RADio[1]:ARB:TRIGger:TYPE:SADVance:THOFf ON|OFF|1|0 [:SOURce]:RADio[1]:ARB:TRIGger:TYPE:SADVance:THOFf?	–	
[:SOURce]:RADio[1]:ARB:TRIGger:TYPE:SADVance[:TYPE] SINGle|CONTinuous [:SOURce]:RADio[1]:ARB:TRIGger:TYPE:SADVance[:TYPE]?	–	
[:SOURce]:RADio[1]:ARB:TRIGger[:SOURce] KEY|BUS|EXT [:SOURce]:RADio[1]:ARB:TRIGger[:SOURce]?	✓	
[:SOURce]:RADio[1]:ARB:TRIGger[:SOURce]:EXTernal:DELay <value> [:SOURce]:RADio[1]:ARB:TRIGger[:SOURce]:EXTernal:DELay?	✓	
[:SOURce]:RADio[1]:ARB:TRIGger[:SOURce]:EXTernal:DELay:STATe ON|OFF|1|0 [:SOURce]:RADio[1]:ARB:TRIGger[:SOURce]:EXTernal:DELay:STATe?	✓	

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
[:SOURce]:RADio[1]:ARB:TRIGger[:SOURce]:EXTernal:SLOPe POSitive|NEGative [:SOURce]:RADio[1]:ARB:TRIGger[:SOURce]:EXTernal:SLOPe?	✓	
[:SOURce]:RADio[1]:ARB:TRIGger[:SOURce]:EXTernal[:SOURce] EPT1|EPT2|EPTRIGGER1|EPTRIGGER2 [:SOURce]:RADio[1]:ARB:TRIGger[:SOURce]:EXTernal[:SOURce]?	✓	
[:SOURce]:RADio[1]:ARB:VCO:CLOCk:RATE?	–	
[:SOURce]:RADio[1]:ARB:VCO:CLOCk[:SOURce] INTernal|EXTernal [:SOURce]:RADio[1]:ARB:VCO:CLOCk[:SOURce]?	✓	
[:SOURce]:RADio[1]:ARB:WAVeform "WFM1:filename"|"SEQ:filename" [:SOURce]:RADio[1]:ARB:WAVeform?	✓	
[:SOURce]:RADio[1]:ARB:WAVeform:NHEaders "WFM1:filename"|"SEQ:filename" [:SOURce]:RADio[1]:ARB:WAVeform:NHEaders?	✓	
[:SOURce]:RADio[1]:ARB[:STATe] ON|OFF|1|0 [:SOURce]:RADio[1]:ARB[:STATe]?	✓	
Edge Subsystem		
[:SOURce]:RADio[1]:EDGE:...	–	*This subsystem is not supported.*
GSM Subsystem		
[:SOURce]:RADio[1]:GSM:...	–	*This subsystem is not supported.*
Input Subsystem		
:INPut:BERT[:BASeband]:...	–	*This subsystem is not supported.*
Measure Subsystem		
:MEASure:[SCALar]:BERT:BTS:LOOPback:EDGE:MCS5[:SENSitivity]?	–	*This subsystem is not supported.*

Table 6-2 E4428C/38C Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**N51xxA**	**Remarks**
Multi-Tone Subsystem		
[:SOURce]:RADio[1]:MTONe:ARB:...	–	*This subsystem is not supported.*
NADC Subsystem		
[:SOURce]:RADio[1][:NADC]:...	–	*This subsystem is not supported.*
PDC Subsystem		
[:SOURce]:RADio[1]:PDC:...	–	*This subsystem is not supported.*
PHS Subsystem		
[:SOURce]:RADio[1]:PHS:...	–	*This subsystem is not supported.*
Sense Subsystem		
:SENSe:BERT:...	–	*This subsystem is not supported.*
Tetra Subsystem		
[:SOURce]:RADio[1]:TETRa:...	–	*This subsystem is not supported.*
Wideband CDMA ARB Subsystem		
[:SOURce]:RADio[1]:WCDMa:TGPP:ARB:...	–	*This subsystem is not supported.*

8648A/B/C/D Compatible Commands

Selecting the Programming Language

NOTE Compatibility is provided for GPIB only; USB and LAN are *not* supported.

The Agilent MXG has only one AM path; and only one internal and one external source. If executed, the "2" path or "2" internal or external source commands will result in a "`ERROR: -113, Undefined Header`" to be generated in the signal generator.

When using the programming codes in this section, you must set the remote programming language to the correct language format.

- On the front-panel, press the following keys:

  Utility > **I/O Config** > **GPIB Setup** > **Remote Language** > **8648A/B/C/D**

  or

- Execute the SCPI command :SYSTem:LANGuage found on page 168.

To keep the remote language choice so that it does not reset with either preset, instrument power cycle, or *RST, perform the following.

- On the front-panel, press the following keys:

  Utility > **Power On/Preset** > **Preset Language** > **8648A/B/C/D**

  or

- Execute the SCPI command :PRESet:LANGuage found on page 168.

To set the *IDN? response to match the remote language setting, use the command :SYSTem:IDN located in "Changing the Signal Generator Identification String" on page 167.

Table 6-3 8648A/B/C/D Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
System Function Commands		
IEEE Common Commands		
*CLS?	✓	
*ESE <dec. num. data> *ESE?	✓	
*IDN?	✓	
*OPC *OPC?	✓	
*RCL <reg_num>[,<seq_num>]	✓	

Table 6-3 8648A/B/C/D Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
*RST?	✓	
*SAV <reg_num>[,<seq_num>]	✓	
*SRE <dec. num. data> *SRE?	✓	
*STB?	✓	
*TST?	✓	
*WAI?	✓	
Status Subsystem		
[:SOURce]:STATus:QUEStionable:PAGing:CONDition?	–	
[:SOURce]:STATus:QUEStionable:PAGing:ENABle <NR1> [:SOURce]:STATus:QUEStionable:PAGing:ENABle?	–	
[:SOURce]:STATus:QUEStionable:PAGing:EVENt?	–	
[:SOURce]:STATus:QUEStionable:POWer:CONDition?	✓	
[:SOURce]:STATus:QUEStionable:POWer:ENABle <NR1> [:SOURce]:STATus:QUEStionable:POWer:ENABle?	✓	
[:SOURce]:STATus:QUEStionable:POWer:EVENt?	✓	
[:SOURce]:STATus:QUEStionable:MODulation:CONDition?	–	
[:SOURce]:STATus:QUEStionable:MODulation:ENABle <NR1> [:SOURce]:STATus:QUEStionable:MODulation:ENABle?	–	
[:SOURce]:STATus:QUEStionable:MODulation:EVENt?	–	
[:SOURce]:STATus:QUEStionable:CALibration:FEXTension[:EVENt]?	✓	
[:SOURce]:STATus:QUEStionable:CALibration:FEXTension:CONDition?	✓	

Table 6-3 8648A/B/C/D Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:STATus:QUEStionable:CALibration:ENABle <NR1> [:SOURce]:STATus:QUEStionable:CALibration:ENABle?	✓	
System Subsystem		
[:SOURce]:SYSTem:LANGuage "COMP"|"SCPI" [:SOURce]:SYSTem:LANGuage?	✓	
[:SOURce]:SYSTem:ERRor?	✓	
[:SOURce]:SYSTem:VERSion?	✓	
Analog Function Commands		
Amplitude Subsystem		
[:SOURce]:OUTPut:STATe ON|OFF [:SOURce]:OUTPut:STATe?	✓	
[:SOURce]:POWer:AMPLitude <value><units> [:SOURce]:POWer:AMPLitude?	✓	
[:SOURce]:POWer:ATTenuation:AUTO ON|OFF [:SOURce]:POWer:ATTenuation:AUTO?	✓	
[:SOURce]:POWer:REFerence <value><units> [:SOURce]:POWer:REFerence?	✓	
[:SOURce]:POWer:REFerence:STATe ON|OFF [:SOURce]:POWer:REFerence:STATe?	✓	
Frequency Subsystem		
[:SOURce]:FREQuency:CW <value><units> [:SOURce]:FREQuency:CW?	✓	
[:SOURce]:FREQuency:REFerence <value><units> [:SOURce]:FREQuency:REFerence?	✓	
[:SOURce]:FREQuency:REFerence:STATe ON|OFF [:SOURce]:FREQuency:REFerence:STATe?	✓	
Amplitude Modulation Subsystem		
[:SOURce]:AM:DEPTh <value>PCT [:SOURce]:AM:DEPTh?	✓	

Table 6-3 8648A/B/C/D Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:AM:INTernal2:FREQuency <value><units> [:SOURce]:AM:INTernal2:FREQuency?	–	
[:SOURce]:AM:INTernal2:FUNCtion:SHAPe SINe|TRIangle|SQUare|SAW [:SOURce]:AM:INTernal2:FUNCtion:SHAPe?	–	
[:SOURce]:AM:STATe ON|OFF [:SOURce]:AM:STATe?	✓	
[:SOURce]:AM:SOURce INTernal[l]	✓	*Supported but the following parameters are not supported:* INTernal2
[:SOURce]:AM:SOURce INTernal|EXTernal [:SOURce]:AM:SOURce?	✓	
[:SOURce]:AM:INTernal:FREQuency 1kHz [:SOURce]:AM:INTernal:FREQuency 400Hz [:SOURce]:AM:INTernal:FREQuency?	✓	
[:SOURce]:AM:EXTernal:COUPling AC|DC [:SOURce]:AM:EXTernal:COUPling?	✓	
Frequency Modulation Subsystem		
[:SOURce]:CALibration:DCFM	✓	
[:SOURce]:FM:DEViation <value>kHz [:SOURce]:FM:DEViation?	✓	
[:SOURce]:FM:STATe ON|OFF [:SOURce]:FM:STATe?	✓	
[:SOURce]:FM:SOURce INTernal[1]	✓	Supported but the following parameters are not supported: |2|
[:SOURce]:FM:SOURce INTernal|EXTernal [:SOURce]:FM:SOURce?	✓	
[:SOURce]:FM:INTernal:FREQuency 1kHz [:SOURce]:FM:INTernal:FREQuency 400Hz [:SOURce]:FM:INTernal:FREQuency?	✓	

Table 6-3 8648A/B/C/D Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:FM:INTernal2:FREQuency <value><units> [:SOURce]:FM:INTernal2:FREQuency?	–	
[:SOURce]:FM:INTernal2:FUNCtion:SHAPe SINe [:SOURce]:FM:INTernal2:FUNCtion:SHAPe?	–	
[:SOURce]:FM:EXTernal:COUPling AC|DC [:SOURce]:FM:EXTernal:COUPling?	✓	
[:SOURce]:FM:INTernal2:FUNCtion:SHAPe TRIangle|SQUare|SAW [:SOURce]:FM:INTernal2:FUNCtion:SHAPe?	–	
Phase Modulation Subsystem		
[:SOURce]:PM:DEViation <value>RAD [:SOURce]:PM:DEViation?	✓	
[:SOURce]:PM:STATe ON|OFF [:SOURce]:PM:STATe?	✓	
[:SOURce]:PM:SOURce INTernal[1]	✓	Supported but the following parameters are not supported: |2|
[:SOURce]:PM:SOURce INTernal|EXTernal	✓	
[:SOURce]:PM:SOURce?	✓	
[:SOURce]:PM:INTernal:FREQuency 1kHz [:SOURce]:PM:INTernal:FREQuency 400Hz [:SOURce]:PM:INTernal:FREQuency?	✓	
[:SOURce]:PM:INTernal2:FREQuency <value><units> [:SOURce]:PM:INTernal2:FREQuency?	–	
[:SOURce]:PM:INTernal2:FUNCtion:SHAPe SINe|TRIangle|SQUare|SAW [:SOURce]:PM:INTernal2:FUNCtion:SHAPe?	–	
[:SOURce]:PM:EXTernal:COUPling AC|DC [:SOURce]:PM:EXTernal:COUPling?	✓	
Pulse Modulation Subsystem		

Table 6-3 8648A/B/C/D Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:PULM:STATe ON|OFF [:SOURce]:PULM:STATe?	✓	
[:SOURce]:INITiate:IMMediate	✓	Supported but without the [:SOURce] command. Including :SOUR will generate an "Error –113: Undefined header" on the Agilent MXG.
[:SOURce]:ABORt	✓	Supported but without the [:SOURce] command. Including :SOUR will generate an "Error –113: Undefined header" on the Agilent MXG.
[:SOURce]:TRIGger:COUNt <value> [:SOURce]:TRIGger:COUNt?	–	
[:SOURce]:DM:FORMat FSK2|FSK4 [:SOURce]:DM:FORMat?	–	
[:SOURce]:DM:STATe ON|OFF [:SOURce]:DM:STATe?	–	
[:SOURce]:DM:DEViation <value> [:SOURce]:DM:DEViation?	–	
[:SOURce]:DM:POLarity NORMal|INVert [:SOURce]:DM:POLarity?	–	
[:SOURce]:DM:FILTer:STATe ON|OFF [:SOURce]:DM:FILTer:STATe?	–	
[:SOURce]:PAGing:SELect POCS|FLEX|FTD|RESY|PN15 [:SOURce]:PAGing:SELect?	–	
[:SOURce]:PAGing:{POCS|FLEX|FTD|PN15}:RATE <value> [:SOURce]:PAGing:{POCS|FLEX|FTD|PN15}:RATE?	–	
[:SOURce]:PAGing:{POCS|FLEX|FTD}:MESSage:SELect <value> [:SOURce]:PAGing:{POCS|FLEX|FTD}:MESSage:SELect?	–	

Table 6-3 8648A/B/C/D Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:PAGing:{POCS|FLEX|FTD}:MESSage:DEFine "string" [:SOURce]:PAGing:{POCS|FLEX|FTD}:MESSage:DEFine?	–	
[:SOURce]:PAGing:{POCS|FLEX|FTD}:MESSage:LENGth <value> [:SOURce]:PAGing:{POCS|FLEX|FTD}:MESSage:LENGth?	–	
[:SOURce]:PAGing:{POCS|FLEX|FTD}:ARBitrary:DEFine <value>,...<val(n)>	–	
[:SOURce]:PAGing:{POCS|FLEX|FTD}:ARBitrary:STARt <value> [:SOURce]:PAGing:{POCS|FLEX|FTD}:ARBitrary:STARt?	–	
[:SOURce]:PAGing:{POCS|FLEX|FTD}:ARBitrary:STOP <value> [:SOURce]:PAGing:{POCS|FLEX|FTD}:ARBitrary:STOP?	–	
[:SOURce]:PAGing:{FLEX|FTD}:TYPE TONE|NUMeric|ALPHanumeric|HBINary [:SOURce]:PAGing:{FLEX|FTD}:TYPE?	–	
[:SOURce]:PAGing:{FLEX|FTD}:VECTor STANdard|SPECial|NUMBered [:SOURce]:PAGing:{FLEX|FTD}:VECTor?	–	
[:SOURce]:PAGing:{FLEX|FTD}:NUMBered <value> [:SOURce]:PAGing:{FLEX|FTD}:NUMBered?	–	
[:SOURce]:PAGing:{FLEX|FTD}:CYCLe <value> [:SOURce]:PAGing:{FLEX|FTD}:CYCLe?	–	
[:SOURce]:PAGing:{FLEX|FTD}:FRAMe <value> [:SOURce]:PAGing:{FLEX|FTD}:FRAMe?	–	
[:SOURce]:PAGing:{FLEX|FTD}:CCOunt?	–	
[:SOURce]:PAGing:{FLEX|FTD}:FCOunt?	–	
[:SOURce]:PAGing:{FLEX|FTD}:PHASe A|B|C|D [:SOURce]:PAGing:{FLEX|FTD}:PHASe?	–	

Table 6-3 8648A/B/C/D Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:PAGing:{FLEX|FTD}:COLLapse <value> [:SOURce]:PAGing:{FLEX|FTD}:COLLapse?	–	
[:SOURce]:PAGing:{FLEX|FTD}:CODE <value> [:SOURce]:PAGing:{FLEX|FTD}:CODE?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ATYPe SHORt|LONG [:SOURce]:PAGing:{FLEX|FTD}:ATYPe?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ADDRess{1|2} <value> [:SOURce]:PAGing:{FLEX|FTD}:ADDRess{1|2}?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ISTop:STATe ON|OFF [:SOURce]:PAGing:{FLEX|FTD}:ISTop:STATe?	–	
[:SOURce]:PAGing:{FLEX|FTD}:HEADer:STATe ON|OFF [:SOURce]:PAGing:{FLEX|FTD}:HEADer:STATe?	–	
[:SOURce]:PAGing:{FLEX|FTD}:TERMinator:STATe ON|OFF [:SOURce]:PAGing:{FLEX|FTD}:TERMinator:STATe?	–	
[:SOURce]:PAGing:{FLEX|FTD}:VECTor STANdard|SPECial|NUMBered [:SOURce]:PAGing:{FLEX|FTD}:VECTor?	–	
[:SOURce]:PAGing:{FLEX|FTD}:HBINary BIT1|BIT7|BIT8|BIT14|BIT16 [:SOURce]:PAGing:{FLEX|FTD}:HBINary?	–	
[:SOURce]:PAGing:{FLEX|FTD}:DCAL:STATe ON|OFF [:SOURce]:PAGing:{FLEX|FTD}:DCAL:STATe?	–	
[:SOURce]:PAGing:{FLEX|FTD}:DCAL:ADDress{1|2} <value> [:SOURce]:PAGing:{FLEX|FTD}:DCAL:ADDress{1|2}?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SELect NONE|SSID|NID [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SELect?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SSID:LID <value> [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SSID:LID?	–	

Table 6-3 8648A/B/C/D Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SSID:CZONe <value> [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SSID:CZONe?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SSID:CCODe <value> [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SSID:CCODe?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SSID:TMF <value> [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SSID:TMF?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SSID:FOFF <value> [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:SSID:FOFF?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:NID:ADDRes s <value> [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:NID:ADDRes s?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:NID:AREA <value> [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:NID:AREA?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:NID:MULTip lier <value> [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:NID:MULTip lier?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:NID:TMF <value> [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:NID:TMF?	–	
[:SOURce]:PAGing:{FLEX|FTD}:ROAMing:NID:FOFF <value> [:SOURce]:PAGing:{FLEX|FTD}:ROAMing:NID:FOFF?	–	
[:SOURce]:PAGing:FTD:REFerence <value> [:SOURce]:PAGing:FTD:REFerence?	–	
[:SOURce]:PAGing:FTD:RCO?	–	

Table 6-3 8648A/B/C/D Program Codes and Equivalent SCPI Sequences

3 = Supported by Agilent MXG **– = Not supported by Agilent MXG**	**E44xxB**	**Remarks**
[:SOURce]:PAGing:POCS:TYPE TONE|NUMeric|ALPHanumeric|ALPH7|ALPH8 [:SOURce]:PAGing:POCS:TYPE?	–	
[:SOURce]:PAGing:POCS:CODE <value> [:SOURce]:PAGing:POCS:CODE?	–	
[:SOURce]:PAGing:POCS:FUNCtion 0|1|2|3 [:SOURce]:PAGing:POCS:FUNCtion?	–	

8656B, 8657A/B/D/J Programming Codes

Programming Codes

NOTE Compatibility is provided for GPIB only; USB and LAN are *not* supported.

When using the programming codes in this section, you must set the remote programming language to the correct language format.

- On the front-panel, press the following keys:

  Utility > **I/O Config** > **GPIB Setup** > **Remote Language** > **8656B, 8657A/B**

  or

- Execute the SCPI command :SYSTem:LANGuage found on page 168.

To keep the remote language choice so that it does not reset with either preset, instrument power cycle, or *RST, perform the following.

- On the front-panel, press the following keys:

  Utility > **Power On/Preset** > **Preset Language** > **8656B, 8657A/B**

  or

- Execute the SCPI command :PRESet:LANGuage found on page 168.

To set the *IDN? response to match the remote language setting, use the command :SYSTem:IDN located in "Changing the Signal Generator Identification String" on page 167.

Compatible Codes

8656B, 8657A/B/D/J Codes[a]	Description	Equivalent SCPI Command Syntax
AM	Amplitude Modulation	`[:SOURce]:AM[1][:DEPTh][:LINear] <value><unit>|UP|DOWN` For additional commands, refer to, "S1, S2, or S3 used with AM" on page 240
AO	Amplitude Offset	`[:SOURce]:POWer[:LEVel][IMMediate]: OFFSet <value><unit>`
AP	Amplitude (carrier)	`[:SOURce]:POWer[:LEVel][:IMMediate] [:AMPLitude] <value><unit>`
DB	Unit used with the power command	DB
DF	Unit used with the power command	DB
DM	Unit used with the power command	DBM
DN	Step Down	No equivalent SCPI command
EM	Unit used with the power command	EMF
FM	Frequency Modulation	`[:SOURce]:FM[1][:DEViation] <value><unit>` For additional commands, refer to, "S1, S2, or S3 used with FM" on page 241.
FR	Frequency (carrier)	`[:SOURce]:FREQuency[:CW] <value><unit>`
GT	Flexible Sequence	No equivalent SCPI command
Hz	Unit used with the frequency command	Hz
IS[b]	Increment Set	No equivalent SCPI command
KZ	Unit used with the frequency command	kHz
MV	Unit used with the power command	mV

8656B, 8657A/B/D/J Codes[a]	Description	Equivalent SCPI Command Syntax
MZ	Unit used with the frequency command	MHz
P0[c]	Digital Modulation Off	*The Agilent MXG does not support this feature.*
P4[c]	Digital Modulation On	*The Agilent MXG does not support this feature.*
PC[d]	Unit used with the modulation command	PCT
PD	Phase Decrement	`[:SOURce]:PHASE[:ADJust] <value><RAD>`
PF	Pulse Modulation (Fast Mode)	Refer to, "PF (Pulse Modulation- Fast Mode) or PM (Pulse Modulation)" on page 240.
PI	Phase Increment	`[:SOURce]:PHASE[:ADJust] <value><RAD>`
PM	Pulse Modulation	`[:SOURce]:PULM:SOURce EXT2` `[:SOURce]:PULM:STATe ON`
QS	Reverse Sequence	`*RCL <reg>`
RC	Recall (0–9)	`*RCL <reg>`
RL	Recall (0–99)	`*RCL <reg>`
RP[e]	Reverse Power Protection Reset	No equivalent SCPI command
R2	RF Off	`OUTPut[:STATe] OFF`
R3	RF On	`OUTPut[:STATe] ON`
R5	RF Dead (Full Attenuator)	`OUTPut[:STATe] OFF`
SQ	Sequence	`*RCL <reg>`
ST	Save (0–9)	`*SAV <reg>`
SV	Save (0–99)	`*RCL <reg>`
S1	External Modulation Source	Refer to, "S1, S2, or S3 used with AM" on page 240 or "S1, S2, or S3 used with FM" on page 241.
S2	Internal 400 Hz Modulation Source	Refer to, "S1, S2, or S3 used with AM" on page 240 or "S1, S2, or S3 used with FM" on page 241.
S3	Internal 1 kHz Modulation Source	Refer to, "S1, S2, or S3 used with AM" on page 240 or "S1, S2, or S3 used with FM" on page 241.

8656B, 8657A/B/D/J Codes[a]	Description	Equivalent SCPI Command Syntax
S4	Modulation Source Off	Refer to, "S4 (Modulation Source Off)" on page 242.
S5	DC FM	Refer to, "S5 (DC FM)" on page 242.
UP	Step Up	No equivalent SCPI command
UV	Unit used with the power command	UV
VL	Unit used with the power command	V
0-9	Numerals 0–9	0–9
–	Minus Sign	–
.	Decimal Point	.
%[d]	Unit used with the modulation command	PCT

a. Program codes are either upper or lower case.
b. Increment Set is implemented for frequency (FR) and amplitude (AP) only.
c. This code is used with the NADC, PDC, and PHS digital modulation.
d. Either PC or % can be used.
e. The source of reverse power must be removed.

Non-Compatible Codes

8656B, 8657A/B/D/J Codes	Description
HI	HI ALC
LO	LO ALC
R0	Standby
R1	On

Command Mapping

When using the 8656B, 8657A/B/D/J-compatible programming codes, the N5181A/82A internally maps these codes to an equivalent SCPI response. In addition, the modulation source selections for the 8656B, 8657A/B/D/J differ from those available in the N5181A/82A and therefore, are mapped to a

valid selection. (Refer to Table 6-4.)

Table 6-4

Modulation Sources	
8656B, 8657A/B/D/J	**Agilent MXG Signal Generators**
AM, Internal	AM1, Internal 1
AM, External	AM1, External 1
FM, Internal	FM1, Internal 1
FM, External	FM1, External 1
AM, Internal and External	AM1, Internal 1, External 1
FM, Internal and External	FM1, Internal 1, External 1

NOTE The 8656, 8657A/B/D/J signal generators allow multiple modulations to use the same input; the N5181A/82A does not. If you configure multiple modulations on the same input, the N5181A/82A automatically disables the modulations.

The mapping between the 8656B, 8657A/B/D/J-compatible programming codes and the SCPI commands changes depending on the programming codes being executed. Refer to the following sections for explanations of the codes that are affected.

PF (Pulse Modulation-Fast Mode) or PM (Pulse Modulation)

The N5181A/82A supports only one input selection for pulse which is EXTernal 1 (PULSE connector). This is a DC-coupled input. Internal pulse modulation, therefore, is not supported in the 8656B, 8657A/B/D/J-compatible language modes. The PF or PM code is mapped to the following SCPI commands:

- `[:SOURce]:PULM:SOURce EXTernal1`
- `[:SOURce]:PULM:STATe ON`

S1, S2, or S3 used with AM

When the AM code is executed, the following occurs:

- AM becomes the active function.

NOTE The N5182A has only one AM channel: [1]. If AM2 is used in a SCPI command, it will be ignored and AM1 will be selected.

If AM is on, or there is no active modulation, a sequence of SCPI commands are implemented when an AM code is executed with a modulation source code. Table 6-5 shows the sequence of SCPI

commands that are implemented.

Table 6-5

	AM On	No Active Modulation
S1	[:SOURce]:AM[1]:EXTernal[1]: COUPling AC [:SOURce]:AM[1]:SOURce EXTernal1	[:SOURce]:AM[1]:EXTernal[1]: COUPling AC [:SOURce]:AM[1]:SOURce EXTernal1 [:SOURce]:AM[1]:STATe ON
S2	[:SOURce]:AM[1]:SOURce INT[1] [:SOURce]:AM[1]:INTernal[1]: FREQuency 400 Hz	[:SOURce]:AM[1]:SOURce INT[1] [:SOURce]:AM[1]:INTernal[1]: FREQuency 400 Hz [:SOURce]:AM[1]:STATe ON
S3	[:SOURce]:AM[1]:SOURce INT[1] [:SOURce]:AM[1]:INTernal[1]: FREQuency 1 kHz	[:SOURce]:AM[1]:SOURce INT[1] [:SOURce]:AM[1]:INTernal[1]: FREQuency 1 kHz [:SOURce]:AM[1]:STATe ON

- If FM or pulse modulation is on, the signal generator attempts to set up AM with the same settings and turns off the other modulation.

S1, S2, or S3 used with FM

When the FM code is executed, the following occurs:

- FM becomes the active function.

If FM is on, or there is no active modulation, a sequence of SCPI commands are implemented when an FM code is executed with a modulation source code. Table 6-6 shows the sequence of SCPI commands that are implemented.

Table 6-6

	FM On	No Active Modulation
S1	[:SOURce]:FM[1]:EXTernal[1]: COUPling AC [:SOURce]:FM[1]:SOURce EXTernal1	[:SOURce]:FM[1]:EXTernal[1]: COUPling AC [:SOURce]:FM[1]:SOURce EXTernal1 [:SOURce]:FM[1]:STATe ON
S2	[:SOURce]:FM[1]:SOURce INT[1] [:SOURce]:FM[1]:INTernal[1]: FREQuency 400 Hz	[:SOURce]:FM[1]:SOURce INT[1] [:SOURce]:FM[1]:INTernal[1]: FREQuency 400 Hz [:SOURce]:FM[1]:STATe ON
S3	[:SOURce]:FM[1]:SOURce INT[1] [:SOURce]:FM[1]:INTernal[1]: FREQuency 1 kHz	[:SOURce]:FM[1]:SOURce INT[1] [:SOURce]:FM[1]:INTernal[1]: FREQuency 1 kHz [:SOURce]:FM[1]:STATe ON

- If AM or pulse modulation is on, the signal generator attempts to set up FM with the same settings and turns off the other modulation.

S4 (Modulation Source Off)

- If PM is the current active function, pulse modulation is disabled by mapping to the following command:

  ```
  [:SOURce]:PULM:STATe OFF
  ```

- If the last code executed is S2 or S3, internal modulation is turned off for the AM and FM:

  ```
  [:SOURce]:AM[1]:STATe OFF
  [:SOURce]:FM[1]:STATe OFF
  ```

- If the last code executed is S1, external modulation is turned off for the AM and FM:

  ```
  [:SOURce]:AM[1]:STATe OFF
  [:SOURce]:FM[1]:STATe OFF
  ```

- If the current active function is AM or FM, the appropriate modulation is turned off:

  ```
  [:SOURce]:AM[1]:STATe OFF
  [:SOURce]:FM[1]:STATe OFF
  ```

- If S4 is executed with S1, S2, or S3, it will turn off the current modulation.

S5 (DC FM)

- FM becomes the active function.
- In addition, the following commands are mapped:

  ```
  [:SOURce]:FM[1]:SOURce EXTernal1
  [:SOURce]:PULM:STATe OFF
  [:SOURce]:AM[1]:STATe OFF
  [:SOURce]:FM[1]:EXTernal[1]:COUPling DC
  [:SOURce]:FM[1]:STATe ON
  ```

Symbols

Numerics

A

D

E

F

G

Index

N

O

P

Q

R

S

T

U

V

W

Index