ARTISAN®
TECHNOLOGY GROUP

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

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

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

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

Custom engineering
so your equipment works exactly as you specify.

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

Expert team | Trust guarantee | 100% satisfaction

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

Find the *Ametek / VTI Instruments / VXI Technology VM3640A* at our website: *Click HERE*

VM3640A

ARBITRARY WAVEFORM GENERATOR

USER'S MANUAL

P/N: 82-0061-000
Rev. November 19, 2011

VXI Technology, Inc.

2031 Main Street
Irvine, CA 92614-6509
(949) 955-1894

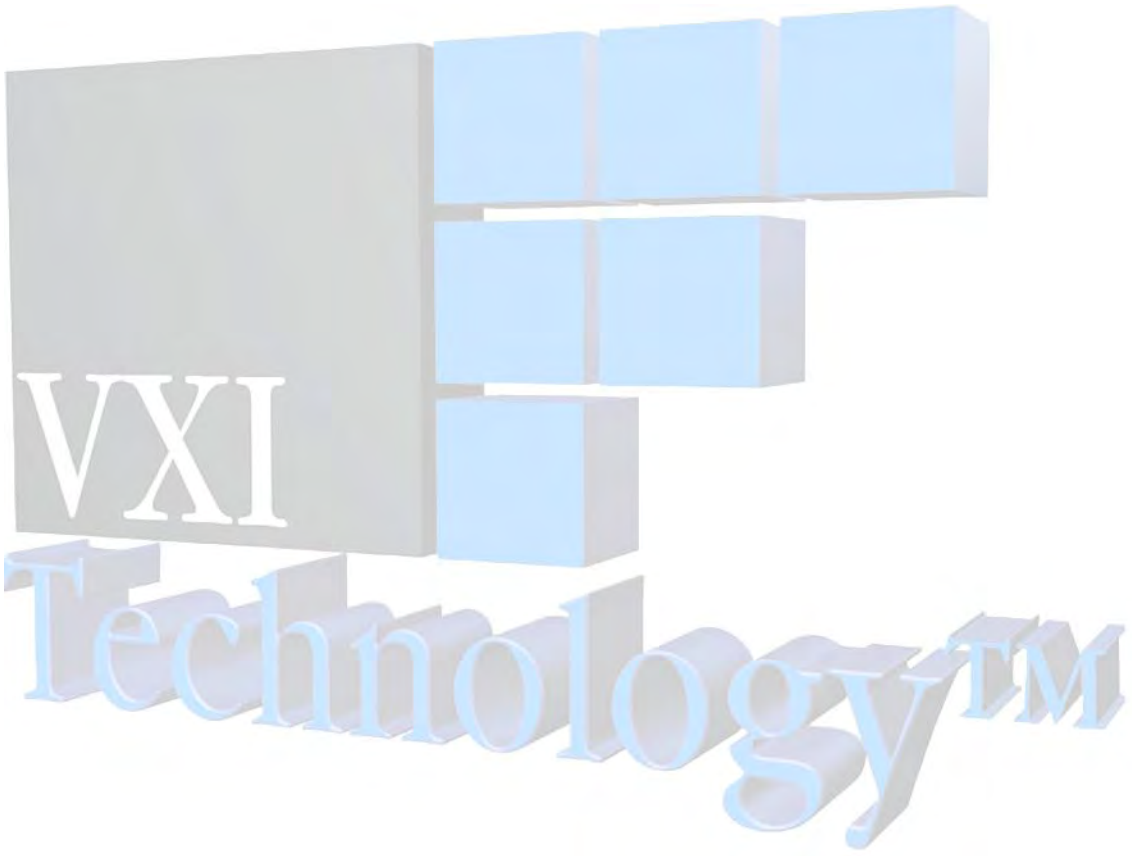

VXI
Technology™

TABLE OF CONTENTS

INTRODUCTION
Certification 6
Warranty 6
Limitation of Warranty 6
Restricted Rights Legend 6
DECLARATION OF CONFORMITY 7
GENERAL SAFETY INSTRUCTIONS 9
Terms and Symbols 9
Warnings 9
SUPPORT RESOURCES 11
SECTION 1 13
INTRODUCTION 13
Overview 13
Programming 14
Calibration 14
Front Panel Connectors 15
Clock In/Out 15
Trigger Input 15
Marker Output 15
Modulation Input 15
Horizontal Sweep Output 15
Main Output 15
SECTION 2 17
PREPARATION FOR USE 17
Installation 17
Calculating System Power and Cooling Requirements 17
Setting the Chassis Backplane Jumpers 18
Setting the Logical Address 18
SECTION 3 19
PROGRAMMING 19
Introduction 19
Notation 20
REGISTER ACCESS 21
Loading and Running a User-Defined Waveform 22
SECTION 4 23
COMMAND DICTIONARY 23
Introduction 23
Alphabetical Command Listing 23
Command Dictionary 28
COMMON IEEE 488.2 COMMANDS 30
*CLS 30
*ESE 31
*ESR? 32
*IDN? 33
*OPC 34
*RST 35
*SRE 36
*STB? 37
*TRG 38
*TST? 39
*WAI 40

INSTRUMENT SPECIFIC SCPI COMMANDS 41
AM[:STATe] 41
CALibration:COUNt? 42
CALibration[:DATA]:AFCorrection 43
CALibration[:DATA]:AMPLitude[:GAIN] 44
CALibration[:DATA]:AMPLitude:OFFSet 45
CALibration[:DATA]:DUTY 46
CALibration[:DATA]:OFFSet[:GAIN] 47
CALibration[:DATA]:OFFSet:OFFSet 48
CALibration[:DATA]:SQHI 49
CALibration[:DATA]:SQLO 50
CALibration[:DATA]:SQUare:OFFSet:OFFSet 51
CALibration[:DATA]:ZERO 52
CALibration:DC:LEVel 53
CALibration:SECure:CODE 54
CALibration:SECure:STATe 55
CALibration:STATe 56
CALibration:STORe 57
CONTrol:IPOWer 58
FSK[:STATe] 59
PSK[:STATe] 60
RESet 61
SENSe:CORRection:IMPedance 62
[SOURce:]BURSt:COUNt 63
[SOURce:]BURSt:PHASe 64
[SOURce:]BURSt:RATE 65
[SOURce:]BURSt[:STATe] 66
[SOURce:]CLOCk:CONFigure 67
[SOURce:]FREQuency[:CW] 68
[SOURce:]FUNCtion:DC:VOLTage 69
[SOURce:]FUNCtion:PULSe:WIDTh 70
[SOURce:]FUNCtion[:SHAPe] 71
[SOURce:]FUNCtion:SQUare[:DUTY] 72
[SOURce:]FUNCtion:USER:DIRectory? 73
[SOURce:]FUNCtion:USER:FREE? 74
[SOURce:]FUNCtion:USER[:NAME] 75
[SOURce:]FUNCtion:USER:WAVE 76
[SOURce:]MARKer:POLarity 77
[SOURce:]MARKer:POSition 78
[SOURce:]MARKer:SOURce 79
[SOURce:]MARKer[:STATe] 80
[SOURce:]OUTPut:FILTer[:LPASs][:STATe] 81
[SOURce:]OUTPut:FILTer[:LPASs]:TYPe 82
[SOURce:]OUTPut[:STATe] 83
[SOURce:]ROSC:SOURce 84
[SOURce:]SWEep:COUNt 85
[SOURce:]SWEep:DIRection 86
[SOURce:]SWEep:MODE 87
[SOURce:]SWEep:POINts 88
[SOURce:]SWEep:SPACing 89
[SOURce:]SWEep:STARt 90
[SOURce:]SWEep:STOP 91
[SOURce:]SWEep[:STATe] 92
[SOURce:]SWEep:TIME 93
[SOURce:]SYNC:MODE 94
[SOURce:]SYNC[:STATe] 95

[SOURce:]TRIGger:GATE 96
[SOURce:]TRIGger[:MODE] 97
[SOURce:]TRIGger:POLarity 98
[SOURce:]TRIGger:RATE 99
[SOURce:]TRIGger:TTLT 100
[SOURce:]VOLTage[:LEVel][:AMPLitude] 101
[SOURce:]VOLTage[:LEVel]:OFFSet 102
TEST[:ALL]? 103
TEST:RAM? 104
REQUIRED SCPI COMMANDS 105
STATus:OPERation:CONDition? 105
STATus:OPERation:ENABle 106
STATus:OPERation[:EVENt]? 107
STATus:PRESet 108
STATus:QUEStionable:CONDition? 109
STATus:QUEStionable:ENABle 110
STATus:QUEStionable[:EVENt]? 111
SYSTem:ERRor? 112
SYSTem:VERSion? 113
INDEX 114

CERTIFICATION

VXI Technology, Inc. (VTI) certifies that this product met its published specifications at the time of shipment from the factory. VTI further certifies that its calibration measurements are traceable to the United States National Institute of Standards and Technology (formerly National Bureau of Standards), to the extent allowed by that organization's calibration facility, and to the calibration facilities of other International Standards Organization members.

WARRANTY

The product referred to herein is warranted against defects in material and workmanship for a period of three years from the receipt date of the product at customer's facility. The sole and exclusive remedy for breach of any warranty concerning these goods shall be repair or replacement of defective parts, or a refund of the purchase price, to be determined at the option of VTI.

For warranty service or repair, this product must be returned to a VXI Technology authorized service center. The product shall be shipped prepaid to VTI and VTI shall prepay all returns of the product to the buyer. However, the buyer shall pay all shipping charges, duties, and taxes for products returned to VTI from another country.

VTI warrants that its software and firmware designated by VTI for use with a product will execute its programming when properly installed on that product. VTI does not however warrant that the operation of the product, or software or firmware will be uninterrupted or error free.

LIMITATION OF WARRANTY

The warranty shall not apply to defects resulting from improper or inadequate maintenance by the buyer, buyer-supplied products or interfacing, unauthorized modification or misuse, operation outside the environmental specifications for the product, or improper site preparation or maintenance.

VXI Technology, Inc. shall not be liable for injury to property other than the goods themselves. Other than the limited warranty stated above, VXI Technology, Inc. makes no other warranties, express or implied, with respect to the quality of product beyond the description of the goods on the face of the contract. VTI specifically disclaims the implied warranties of merchantability and fitness for a particular purpose.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subdivision (b)(3)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

VXI Technology, Inc.
2031 Main Street
Irvine, CA 92614-6509 U.S.A.

DECLARATION OF CONFORMITY

Declaration of Conformity According to ISO/IEC Guide 22 and EN 45014

MANUFACTURER'S NAME	VXI Technology, Inc.
MANUFACTURER'S ADDRESS	2031 Main Street Irvine, California 92614-6509
PRODUCT NAME	50 MSample/s Arbitrary Waveform Generator
MODEL NUMBER(S)	VM3640A
PRODUCT OPTIONS	All
PRODUCT CONFIGURATIONS	All

VXI Technology, Inc. declares that the aforementioned product conforms to the requirements of the Low Voltage Directive 73/23/EEC *and the* EMC Directive 89/366/EEC *(inclusive* 93/68/EEC*) and carries the "CE" mark accordingly. The product has been designed and manufactured according to the following specifications:*

SAFETY	EN61010 (2001)
EMC	EN61326 (1997 w/A1:98) Class A CISPR 22 (1997) Class A VCCI (April 2000) Class A ICES-003 Class A (ANSI C63.4 1992) AS/NZS 3548 (w/A1 & A2:97) Class A FCC Part 15 Subpart B Class A EN 61010-1:2001

The product was installed into a C-size VXI mainframe chassis and tested in a typical configuration.

I hereby declare that the aforementioned product has been designed to be in compliance with the relevant sections of the specifications listed above as well as complying with all essential requirements of the Low Voltage Directive.

October 2004

CE



Steve Mauga, QA Manager

GENERAL SAFETY INSTRUCTIONS

Review the following safety precautions to avoid bodily injury and/or damage to the product. These precautions must be observed during all phases of operation or service of this product. Failure to comply with these precautions, or with specific warnings elsewhere in this manual, violates safety standards of design, manufacture, and intended use of the product.

Service should only be performed by qualified personnel.

TERMS AND SYMBOLS

These terms may appear in this manual:

WARNING Indicates that a procedure or condition may cause bodily injury or death.

CAUTION Indicates that a procedure or condition could possibly cause damage to equipment or loss of data.

These symbols may appear on the product:



ATTENTION - Important safety instructions



Frame or chassis ground

WARNINGS

Follow these precautions to avoid injury or damage to the product:

Use Proper Power Cord To avoid hazard, only use the power cord specified for this product.

Use Proper Power Source To avoid electrical overload, electric shock, or fire hazard, do not use a power source that applies other than the specified voltage.

Use Proper Fuse To avoid fire hazard, only use the type and rating fuse specified for this product.

WARNINGS (CONT.)

Avoid Electric Shock

To avoid electric shock or fire hazard, do not operate this product with the covers removed. Do not connect or disconnect any cable, probes, test leads, etc. while they are connected to a voltage source. Remove all power and unplug unit before performing any service. ***Service should only be performed by qualified personnel.***

Ground the Product

This product is grounded through the grounding conductor of the power cord. To avoid electric shock, the grounding conductor must be connected to earth ground.

Operating Conditions

To avoid injury, electric shock or fire hazard:

- Do not operate in wet or damp conditions.
- Do not operate in an explosive atmosphere.
- Operate or store only in specified temperature range.
- Provide proper clearance for product ventilation to prevent overheating.
- DO NOT operate if you suspect there is any damage to this product. ***Product should be inspected or serviced only by qualified personnel.***

SUPPORT RESOURCES

Support resources for this product are available on the Internet and at VTI Instruments customer support centers.

VTI Instruments Corp.
World Headquarters

VTI Instruments Corp.
2031 Main Street
Irvine, CA 92614-6509

Phone: (949) 955-1894
Fax: (949) 955-3041



VTI Instruments
Cleveland Instrument Division

5425 Warner Road
Suite 13
Valley View, OH 44125

Phone: (216) 447-8950
Fax: (216) 447-8951



VTI Instruments, Pvt. Ltd.
Bangalore Instrument Division

135, II & III Floors
Infantry Road
Bangalore – 560 001
India

Phone: +91 80 4040 7900
Phone: +91 80 4162 0200
Fax: +91 80 4170 0200



Technical Support

Phone: (949) 955-1894
Fax: (949) 955-3041
E-mail: support@vtiinstruments.com

Visit http://www.vtiinstruments.com for worldwide support sites and service plan information.

SECTION 1

INTRODUCTION

OVERVIEW

The VM3640A is a high-performance VXIbus arbitrary waveform generator that is ideal for applications requiring standard function generator capability, or the need to generate customer-defined waveforms. Direct digital synthesis is used to provide outstanding functionality, with standard sine and square waves available to 20 MHz.

The most powerful feature of the VM3640A is that it is part of the VMIP™ family of VXIbus products. This gives the user the added flexibility of combining the VM3640A with two additional VM3640As, providing a multi-channel arbitrary waveform generator. The VM3640A can also be combined with other instruments, such as digital multimeters, digitizers or counters, to create a multi-function, C-size card.

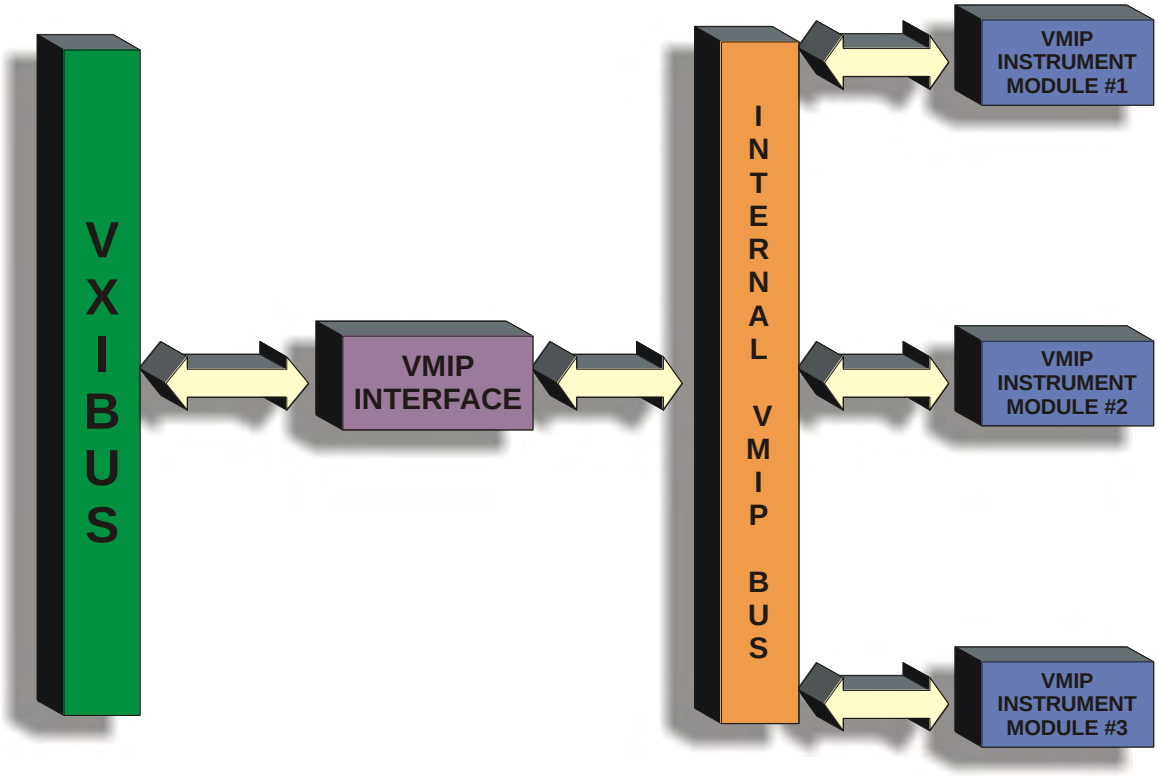


FIGURE 1-1: VMIP PLATFORM

Each arbitrary waveform generator is treated as an independent instrument in the VXIbus chassis. Each has its own Unique Logical Address and its own front panel FAIL and ACCESS indicators. The ACC (ACCESS) LED flashes when read/write commands are being sent to the module. The FAIL LED glows green to indicate that the board is receiving power. This LED glows red when a fail condition has occurred.

PROGRAMMING

The VM3640A is programmed using message-based word serial protocol. The commands are SCPI and IEEE STD 488.2 compatible. VXI*plug&play* drivers are also provided to further ease programming.

CALIBRATION

The calibration constants used to correct the data values are stored in non-volatile memory and are password protected for security. These constants are determined when the instrument is calibrated and can be changed as necessary. These constants may also be queried at any time via a word-serial query and altered via a word-serial command (with the password). All calibration is done using calibration DACs to adjust the gain and offset of each channel. This eliminates the need for removing covers from the unit and allows for automated calibration.

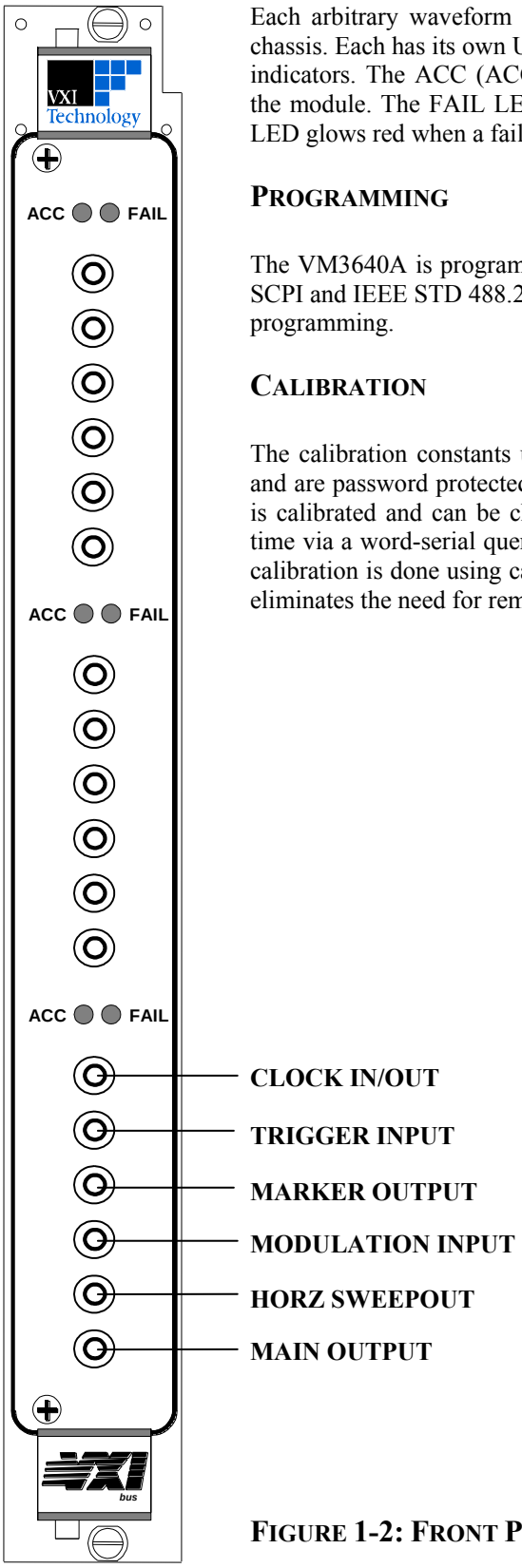


FIGURE 1-2: FRONT PANEL LAYOUT

FRONT PANEL CONNECTORS

The VM3640A has six male SMB connector located on the front panel that require SMB female connectors for instrument interface. The connectors are defined, from top to bottom, as follows:

Clock In/Out

The Clock In/Out connector serves as both an input and an output and provides access to the reference clock of the instrument. If the internal reference source is used, it can be accessed using the *[SOURce:]CLOCk:CONFigure* command. If an externally produced signal is to be used as a reference source, the source must be selected using the *[SOURce:]TRIGger[:MODE]* command.

Trigger Input

The Trigger Input connector is used to synchronize the VM3640A with other instruments. To select an external trigger source, use the *[SOURce:]TRIGger[:MODE]* command.

Marker Output

The Marker Output connector is used to synchronize to the output of the generator. The marker can be enabled or disabled using the *[SOURce:]MARKer[:STATe]* command. The output is about 5 V and can source or sink 40 mA. The *[SOURce:]MARKer:SOURce* command can be used to select the marker source.

Modulation Input

This connector input is used during amplitude modulation, frequency shift keying and phase shift keying. During amplitude modulation, the modulation input is added to the output signal. In frequency and phase shift keying, the modulation input acts as the keying signal.

Horizontal Sweep Output

This output is only available in sweep mode. The horizontal sweep output ramps from 0 V at the beginning of a sweep cycle to 5 V at the end of a sweep cycle. It is intended to provide the user with a representation of the progress through the sweep. The slope of the horizontal sweep output is linear, regardless of the sweep mode selected.

Main Output

This is the main output from the function generator.

VM3640A General Specifications

General Specifications	
Connectors	
	six, male SMB
Waveforms	
Built-In	sine square triangle noise pulse sinc cardiac dc positive/negative ramp exponential rise/fall
Length	
Standard Patterns	4 k (4096) samples
User Patterns	minimum of 8 samples maximum limited to available memory (122,880 samples maximum) (Up to sixteen user defined patterns may be defined depending on the amount of pattern RAM)
Resolution	
	12 bits (including sign)
Sample Rate	
	50 MSamples/s
Frequency	
Sine	0.1 Hz to 20 MHz total harmonic distortion on sine wave no greater than 0.5%
Square	0.1 Hz to 20 MHz rise/fall time no greater than 20 ns with maximum aberrations < 5%
All Other Waveforms	0.1 Hz to 1 MHz
Resolution	8 digits limited by 0.1 Hz for sine/square 5 digits for all other waveforms
Accuracy	same as VXICLK10 (±0.01% typical, can be improved by using an external reference)
Amplitude	
Range	
Voltage (dc)	
Into 50 Ω	-6 V to +6 V
Into ≥ 200 Ω	-10 V to +10 V
Pulsed dc	-6 Vp to +6 Vp
Sine/square	10 Vpp into 50 Ω
All other waveforms	10 Vpp into 50 Ω
Output impedance	50 Ω
Resolution	3.5 digits
Accuracy	±1%
Output Offset	
Range	
Voltage (dc)	-3 V to +3 V, offset + amplitude cannot exceed ±6 V
Pulsed dc	-3 V to +3 V, offset + (amplitude / 2) cannot exceed ±3.5 V
Sine/square	-3 V to +3 V, offset + amplitude cannot exceed 6 V
All other waveforms	-3 V to +3 V, offset + amplitude cannot exceed 6 V

SECTION 2

PREPARATION FOR USE

INSTALLATION

When the VM3640A is unpacked from its shipping carton, the contents should include the following items:

(1) VM3640A VXIbus module
(1) VM3640A Arbitrary Waveform Generator User's Manual (this manual)

All components should be immediately inspected for damage upon receipt of the unit.

Once the VM3640A is assessed to be in good condition, it may be installed into an appropriate C-size or D-size VXIbus chassis in any slot other than slot zero. The chassis should be checked to ensure that it is capable of providing adequate power and cooling for the VM3640A. Once the chassis is found adequate, the VM3640A's logical address and the backplane jumpers of the chassis should be configured before the VM3640A's installation.

CALCULATING SYSTEM POWER AND COOLING REQUIREMENTS

It is imperative that the chassis provide adequate power and cooling for this module. Referring to the chassis user's manual, confirm that the power budget for the system (the chassis and all modules installed therein) is not exceeded and that the cooling system can provide adequate airflow at the specified backpressure.



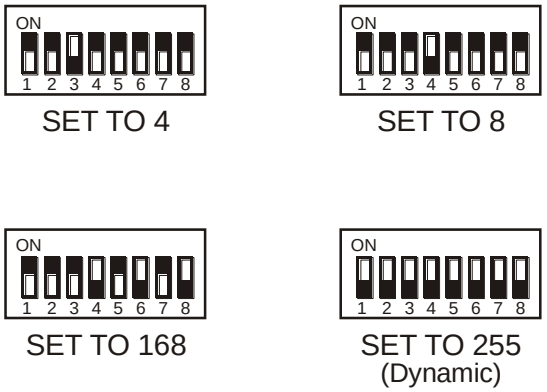
It should be noted that if the chassis cannot provide adequate power to the module, the instrument may not perform to specification or possibly not operate at all. In addition, if adequate cooling is not provided, the reliability of the instrument will be jeopardized and permanent damage may occur. Damage found to have occurred due to inadequate cooling would also void the warranty of the module.

SETTING THE CHASSIS BACKPLANE JUMPERS

Please refer to the chassis user's manual for further details on setting the backplane jumpers.

SETTING THE LOGICAL ADDRESS

The logical address of the VM3640A is set by a single 8-position DIP switch located near the VMIP module's backplane connectors (this is the only switch on the module). The switch is labeled with positions 1 through 8 and with an ON position. A switch pushed toward the ON legend signifies a logic 1; switches pushed away from the ON legend signify a logic 0. The switch located at position 1 is the least significant bit, while the switch located at position 8 is the most significant bit. See Figure 2-1 for examples of setting the logical address switch.



Switch Position	Switch Value
1	1
2	2
3	4
4	8
5	16
6	32
7	64
8	128

FIGURE 2-1: LOGICAL ADDRESS SWITCH SETTING EXAMPLES

The VMIP may contain three separate instruments and will allocate logical addresses as required by the VXIbus specification (revisions 1.3 and 1.4). The logical address of the instrument is set on the VMIP carrier. The VMIP logical addresses must be set to an even multiple of 4 *unless dynamic addressing is used*. Switch positions 1 and 2 must always be set to the OFF position. Therefore, only addresses of 4, 8, 12, 16, ...252 are allowed. The address switch should be set for one of these legal addresses and the address for the second instrument (the instrument in the center position) will automatically be set to the switch set address plus one; while the third instrument (the instrument in the lowest position) will automatically be set to the switch set address plus two. If dynamic address configuration is desired, the address switch should be set for a value of 255 (All switches set to ON). Upon power-up, the slot 0 resource manager will assign the first available logical addresses to each instrument in the VMIP module.

If dynamic address configuration is desired, the address switch should be set for a value of 255. (All switches set to ON). Upon power-up, the slot 0 resource manager will assign the first available logical addresses to each instrument in the VMIP module.

SECTION 3

PROGRAMMING

INTRODUCTION

The VM3640A is a VXIbus message-based device whose command set is compliant with the Standard Commands for Programmable Instruments (*SCPI*) programming language. All module commands are sent over the VXIbus backplane to the module. Commands may be in upper, lower or mixed case. All numbers are sent in ASCII decimal unless otherwise noted.

The SCPI programming language is a tree-structured language based on IEEE STD 488.2 Specifications. It utilizes the IEEE STD 488.2 Standard commands and the device dependent commands are structured to allow multiple branches off the same trunk to be used without repeating the trunk. To use this facility, terminate each branch with a semicolon. For example, **OFFSet:GAIN** and **OFFSet:OFFSet** are both branches off the **CALibration:** trunk and can be combined as follows:

```
CALibration:OFFSet:GAIN <value>;OFFSet <value>
```

The above command is the same as these two commands:

```
CALibration:OFFSet:GAIN <value>
CALibration:OFFSet:OFFSet <value>
```

See the *Standard Commands for Programmable Instruments (SCPI)* Manual, Volume 1: Syntax & Style, Section 6, for more information.

The SCPI commands in this section are listed in upper and lower case. Character case is used to indicate different forms of the same command. Keywords can have both a short form and a long form (some commands only have one form). The short form uses just the keyword characters in uppercase. The long form uses the keyword characters in uppercase plus the keyword characters in lowercase. Either form is acceptable. Note that there are no intermediate forms. All characters of the short form or all characters of the long form must be used. Short forms and long forms may be freely intermixed. The actual commands sent can be in upper case, lower case or mixed case (case is only used to distinguish short and long form for the user). As an example, these commands are all correct and all have the same effect:

```
CALibration:OFFSet:GAIN <value>
calibration:offset:gain <value>
CALIBRATION:OFFSET:GAIN <value>
CAL:OFFSet:GAIN <value>
CAL:OFFS:GAIN <value>
cal:offs:gain <value>
```

The following command is **not** correct because it uses part of the long form of **CALibration**, but not all the characters of the long form:

calib:offs:gain <value>

Incorrect syntax - extra "ib"- only cal or calibration is correct

All of the SCPI commands also have a query form unless otherwise noted. Query forms contain a question mark (?). The query form allows the system to ask what the current setting of a parameter is. The query form of the command generally replaces the parameter with a question mark (?). Query responses do not include the command header. This means only the parameter is returned: no part of the command or "question" is returned.

NOTATION

Keywords or parameters enclosed in square brackets ([ ]) are optional. If the optional part is a keyword, the keyword can be included or left out. Omitting an optional parameter will cause its default to be used.

Parameters are enclosed by angle brackets (< >). Braces ({ }), or curly brackets, are used to enclose one or more parameters that may be included zero or more times. A vertical bar (|), read as "or", is used to separate parameter alternatives.

REGISTER ACCESS

The VM3640A module supports direct register access for high-speed data transfer. The following table shows the A16 Memory Map.

TABLE 3-1: A16 MEMORY MAP

3E	
3C	
3A	
38	
36	
34	
32	
30	
2E	
2C	
2A	
28	
26	
24	
22	
20	Data Register - 16 Bit
1E	
1C	
1A	
18	
16	[A32 Pointer Low]
14	[A32 Pointer High]
12	[A24 Pointer Low]
10	[A24 Pointer High]
E	Data Low
C	Data High
A	Response [/Data Extended]
8	Protocol [/Signal] Register
6	[Offset Register]
4	Status / Control Register
2	Device Type
0	ID Register

LOADING AND RUNNING A USER-DEFINED WAVEFORM

There are four main steps to setting up and running a User-Defined Waveform:

1. Define the waveform name and size by word serial command.

   Example: `FUNC:USER:WAVE #213wavename,1024`

   Defines a user waveform named 'wavename' that is 1024 samples long

2. Download the waveform data using bus access/out register data transfer.

   Example: `USER OFFSET 0x20`

3. Select the downloaded waveform by word serial command.

   Example: `FUNC:USER:NAME #18wavename`

4. Select the 'user' waveform by word serial command.

   Example:
   ```
   OUTPUT ON
   FUNC USER
   ```

SECTION 4

COMMAND DICTIONARY

INTRODUCTION

This section presents the instrument command set. It begins with three alphabetical listings of all the commands supported by the VM3640A: IEEE 488.2 commands, the instrument specific or device dependent SCPI commands and the required SCPI commands. Each listing includes a brief description of each command function.

The remainder of this section is devoted to describing each command, one per page, in detail. The description is presented in a way to assist the user in the use of each command. Every command entry describes the exact command and/or query syntax, the use and range of parameters and a description of the command's purpose.

ALPHABETICAL COMMAND LISTING

The following tables provide an alphabetical listing of each command supported by the VM3640A along with a brief definition. If an X is found in the column titled ***RST**, then the value or setting controlled by this command is possibly changed by the execution of the ***RST** command. If no X is found, then ***RST** has no effect. The **Reset Value** column gives the value of each command's setting when the unit is powered up or when a ***RST** command is executed.

The following table lists the IEEE 488.2 Common (*) Commands. See the *IEEE Standard 488.2 for more information on these commands.*

TABLE 4-1: IEEE 488.2 COMMON COMMANDS

Command	Description	*RST	Reset Value
*CLS	Clears the Status Register		N/A
*ESE	Sets the Standard Event Status Register		N/A
*ESR?	Query the Standard Event Status Register		N/A
*IDN?	Query the module identification string		N/A
*OPC	Set the OPC bit in the Event Status Register		N/A
*RST	Resets the module to a known state		N/A
*SRE	Set the service request enable register		N/A
*STB?	Query the Status Byte Register		N/A
*TRG	Causes a trigger event to occur		N/A
*TST?	Starts and reports a self-test procedure		N/A
*WAI	Halts execution and queries		N/A

TABLE 4-2: INSTRUMENT SPECIFIC SCPI COMMANDS

Command	Description	*RST	Reset Value
AM[:STATe]	Enables or disables amplitude modulation	X	0
CALibration:COUNt?	Tracks the number of times calibration has been performed		Assumes the value last stored in non-volatile memory
CALibration[:DATA]	Sets the calibration constants		
CALibration[:DATA]:AFCorrection	Sets the amplitude gain correction		
CALibration[:DATA]:AMPLitude[:GAIN]	Directly sets the amplitude gain correction		
CALibration[:DATA]:AMPLitude:OFFSet	Directly sets the offset of the amplitude control		
CALibration[:DATA]:DUTY	Calibration for square wave duty cycle		
CALibration[:DATA]:OFFSet[:GAIN]	Directly sets the offset gain correction		
CALibration[:DATA]:OFFSet:OFFSet	Directly sets the output offset voltage control offset		
CALibration[:DATA]:SQHI	Set the positive limit of the square wave output prior to the gain stage		
CALibration[:DATA]:SQLO	Set the negative limit of the square wave output prior to the gain stage		
CALibration[:DATA]:SQUare:OFFSet:OFFSet	Square wave offset adjustment for calibration		
CALibration[:DATA]:ZERO	Zero adjust for the output amplifier		
CALibration:DC:LEVel	Directly sets the dc level calibration point specified in the command		
CALibration:SECure:CODE	Sets the code required to disable calibration security		
CALibration:SECure:STATe	Enables or disables the calibration security		
CALibration:STATe	Enables or disables corrections using calibration data		
CALibration:STORe	Stores calibration data into non-volatile memory		
CONTrol:IPOWer	This command applies power to or removes power from the instrument.	X	1
FSK[:STATe]	Enables or disables Frequency Shift Key modulation input	X	0
PSK[:STATe]	Enables or disables Phase Shift Key modulation input	X	0
RESet	Resets all parameters to their default state		N/A
SENSe:CORRection:IMPedance	Allows the user to set the value of the load resistor connected to the output	X	50 Ω
[SOURce:]BURSt:COUNt	Sets the number of cycles per burst	X	1
[SOURce:]BURSt:PHASe	Sets the phase of the output waveform	X	0
[SOURce:]BURSt:RATE	Sets the burst repetition rate	X	100 Hz
[SOURce:]BURSt[:STATe]	Enables or disables burst mode	X	0
[SOURce:]CLOCk:CONFigure	Sets front panel clock as an input or output	X	INP
[SOURce:]FREQuency[:CW]	Controls the frequency of the output signal	X	1000 Hz
[SOURce:]FUNCtion:DC:VOLTage	Sets the dc output voltage	X	1 V

Command	Description	*RST	Reset Value
[SOURce:]FUNCtion:PULSe:WIDTh	Sets the pulse width	X	1 μs, 1e3 Hz
[SOURce:]FUNCtion[:SHAPe]	Selects the output waveform shape	X	SIN
[SOURce:]FUNCtion:SQUare[:DUTY]	Sets the duty cycle of the square wave function	X	50.0%
[SOURce:]FUNCtion:USER:DIRectory?	Lists the user-defined waveforms		N/A
[SOURce:]FUNCtion:USER:FREE?	Queries the amount of free space for user-defined waveforms		N/A
[SOURce:]FUNCtion:USER[:NAME]	Sets the output waveform shape to a user-defined waveform		N/A
[SOURce:]FUNCtion:USER:WAVE	Sets the name and stores a user-defined waveform		N/A
[SOURce:]MARKer:POLarity	Sets the polarity for the output marker	X	POS
[SOURce:]MARKer:POSition	Sets the marker to a specified point	X	0
[SOURce:]MARKer:SOURce	Selects the source to generate the output marker	X	ZCRO
[SOURce:]MARKer[:STATe]	Enables or disables the marker output	X	1
[SOURce:]OUTPut:FILTer[:LPASs][:STATe]	Enables or bypasses the output filter	X	1
[SOURce:]OUTPut:FILTer[:LPASs]:TYPe	Selects the output filter type	X	ELL
[SOURce:]OUTPut[:STATe]	Enables or disables the MAIN OUT output	X	0
[SOURce:]ROSC:SOURce	Selects the source for the reference oscillator	X	PLL
[SOURce:]SWEep:COUNt	Sets the number of sweeps enabled per trigger event	X	0
[SOURce:]SWEep:DIRection	Sets the sweep direction	X	UP
[SOURce:]SWEep:MODE	Sets the sweep mode	X	CRES
[SOURce:]SWEep:POINts	Sets the sweep points	X	100
[SOURce:]SWEep:SPACing	Sets the sweep spacing	X	LIN
[SOURce:]SWEep:STARt	Sets the starting sweep frequency	X	1000 Hz
[SOURce:]SWEep:STOP	Sets the ending sweep frequency	X	2000 Hz
[SOURce:]SWEep[:STATe]	Enables or disables the sweep function	X	0
[SOURce:]SWEep:TIME	Sets the sweep duration time	X	0.100 s
[SOURce:]SYNC:MODE	Selects the sync mode	X	SLAV
[SOURce:]SYNC[:STATe]	Enables/disables the simultaneous synchronization mode	X	0
[SOURce:]TRIGger:GATE	Selects gate mode	X	0
[SOURce:]TRIGger[:MODE]	Selects the trigger source	X	IMM
[SOURce:]TRIGger:POLarity	Selects the polarity of the trigger signal	X	POS
[SOURce:]TRIGger:RATE	Sets the period for the internal trigger	X	100 Hz
[SOURce:]TRIGger:TTLT	Selects the backplane trigger line	X	0
[SOURce:]VOLTage[:LEVel][:AMPLitude]	Sets the signal output voltage	X	1.0 V_{P-P}
[SOURce:]VOLTage[:LEVel]:OFFSet	Sets the output voltage offset	X	0.0
TEST[:ALL]?	Performs a non-destructive test of the hardware		N/A
TEST:RAM?	Performs a destructive test of the Trace Memory		N/A

TABLE 4-3: REQUIRED SCPI COMMANDS

Command	Description	*RST	Reset Value
STATus:OPERation:CONDition?	Queries the Operation Status Condition Register		N/A
STATus:OPERation:ENABle	Sets the Operation Status Enable Register		N/A
STATus:OPERation[:EVENt]?	Queries the Operation Status Event Register		N/A
STATus:PRESet	Presets the Status Register		N/A
STATus:QUEStionable:CONDition?	Queries the Questionable Status Condition Register		N/A
STATus:QUEStionable:ENABle	Sets the Questionable Status Enable Register		N/A
STATus:QUEStionable[:EVENt]?	Queries the Questionable Status Event Register		N/A
SYSTem:ERRor?	Queries the Error Queue	X	Clears queue
SYSTem:VERSion?	Queries which version of the SCPI standard the module complies with		N/A

COMMAND DICTIONARY

The remainder of this section is devoted to the actual command dictionary. Each command is fully described on its own page. In defining how each command is used, the following items are described:

Purpose	Describes the purpose of the command.
Type	Describes the type of command such as an event or setting.
Command Syntax	Details the exact command format.
Command Parameters	Describes the parameters sent with the command and their legal range.
Reset Value	Describes the values assumed when the *RST command is sent.
Query Syntax	Details the exact query form of the command.
Query Parameters	Describes the parameters sent with the command and their legal range. The default parameter values are assumed the same as in the command form unless described otherwise.
Query Response	Describes the format of the query response and the valid range of output.
Description	Describes in detail what the command does and refers to additional sources.
Examples	Present the proper use of each command and its query (when available).
Related Commands	Lists commands that affect the use of this command or commands that are affected by this command.

COMMON IEEE 488.2 COMMANDS

*CLS

Purpose	Clears all status and event registers	
Type	IEEE 488.2 Common Command	
Command Syntax	*CLS	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	N/A	
Query Parameters	N/A	
Query Response	N/A	
Description	This command clears the Status Event Register, Operation Status Register and the Questionable Data/Signal Register. It also clears the OPC flag and clears all queues (except the output queue).	
Examples	**Command / Query**	**Response / Descriptions**
	*CLS	*(Clears all status and event registers)*
Related Commands	N/A	

*ESE

Purpose	Sets the bits of the Event Status Enable Register
Type	IEEE 488.2 Common Command
Command Syntax	*ESE <mask>
Command Parameters	<mask> = numeric ASCII value
***RST Value**	N/A, the parameter is required
Query Syntax	*ESE?
Query Parameters	N/A
Query Response	Numeric ASCII value from 0 to 255
Description	The Event Status Enable (ESE) command is used to set the bits of the Event Status Enable Register. See ANSI/IEEE 488.2-1987 section 11.5.1 for a complete description of the ESE register. A value of 1 in a bit position of the ESE register enables generation of the Event Status Bit (ESB) in the Status Byte by the corresponding bit in the Event Status Register (ESR). If the ESB is set in the Service Request Enable (SRE) register, then an interrupt will be generated. See the *ESR? query for details regarding the individual bits. The ESE register layout is: Bit 0 - Operation Complete Bit 1 - Request Control Bit 2 - Query Error Bit 3 - Device Dependent Error Bit 4 - Execution Error Bit 5 - Command Error Bit 6 - User Request Bit 7 - Power On The Event Status Enable query reports the current contents of the Event Status Enable Register.
Examples	**Command / Query** / **Response *(Description)***
	*ESE 36 *ESE? / 36 *(Returns the value of the event status enable register)*
Related Commands	*ESR?

*ESR?

Purpose	Queries and clears the Standard Event Status Register	
Type	IEEE 488.2 Common Command	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	ESR?	
Query Parameters	N/A	
Query Response	Numeric ASCII value from 0 to 255	
Description	The Event Status Register (ESR) query queries and clears the contents of the Standard Event Status Register. This register is used in conjunction with the ESE register to generate the Event Status Bit (ESB) in the Status Byte. The layout of the ESR is: Bit 0 - Operation Complete Bit 1 - Request Control Bit 2 - Query Error Bit 3 - Device Dependent Error Bit 4 - Execution Error Bit 5 - Command Error Bit 6 - User Request Bit 7 - Power On The Operation Complete bit is set when it receives an *OPC command. The Query Error bit is set when data is over-written in the output queue. This could occur if one query is followed by another without reading the data from the first query. The Execution Error bit is set when an execution error is detected. Errors that range from -200 to -299 are execution errors. The Command Error bit is set when a command error is detected. Errors that range from -100 to -199 are command errors. The Power On bit is set when the module is first powered on or after it receives a reset via the VXI Control Register. Once the bit is cleared (by executing the *ESR? command) it will remain cleared.	
Examples	**Command / Query**	**Response *(Description)***
	*ESR?	4
Related Commands	*ESE	

*IDN?

Purpose	Queries the module for its identification string	
Type	IEEE 488.2 Common Command	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	*IDN?	
Query Parameters	N/A	
Query Response	ASCII character string	
Description	The Identification (IDN) query returns the identification string of the module. The response is divided into four fields separated by commas. The first field is the manufacturer's name, the second field is the model number, the third field is an optional serial number and the fourth field is the firmware revision number. If a serial number is not supplied, the third field is set to 0 (zero).	
Examples	**Command / Query**	**Response *(Description)***
	*IDN	VXI Technology, Inc.,VM3640A,0,1.0 *(The revision listed here is for reference only; the response will always be the current revision of the instrument.)*
Related Commands	N/A	

*OPC

Purpose	Sets the OPC bit in the Event Status Register	
Type	IEEE 488.2 Common Command	
Command Syntax	*OPC	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	*OPC?	
Query Parameters	N/A	
Query Response	1	
Description	The Operation Complete (OPC) command sets the OPC bit in the Event Status Register when all pending operations have completed. The OPC query will return a 1 to the output queue when all pending operations have completed.	
Examples	**Command / Query**	**Response *(Description)***
	*OPC *OPC?	*(Sets the OPC bit in the Event Status Register)* 1 *(Returns the value of the Event Status Register)*
Related Commands	*WAI	

*RST

Purpose	Resets the module's hardware and software to a known state	
Type	IEEE 488.2 Common Command	
Command Syntax	*RST	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	N/A	
Query Parameters	N/A	
Query Response	N/A	
Description	The Reset (RST) command resets the module's hardware and software to a known state. See the command index at the beginning of this chapter for the default parameter values used with this command.	
Examples	**Command / Query**	**Response *(Description)***
	*RST	*(Resets the module)*
Related Commands	N/A	

*SRE

Purpose	Sets the service request enable register	
Type	IEEE 488.2 Common Command	
Command Syntax	*SRE <mask>	
Command Parameters	<mask> = Numeric ASCII value from 0 to 255	
***RST Value**	None – Required Parameter	
Query Syntax	*SRE?	
Query Parameters	N/A	
Query Response	Numeric ASCII value from 0 to 255	
Description	The Service Request Enable (SRE) mask is used to control which bits in the status byte generate back plane interrupts. If a bit is set in the mask that newly enables a bit set in the status byte and interrupts are enabled, the module will generate a REQUEST TRUE event via an interrupt. See the *STB? Command for the layout of bits. **Note:** Bit 6 is always internally cleared to zero as required by IEEE 488.2 section 11.3.2.3. The layout of the Service Request Enable Register is: Bit 0 – Unused Bit 1 – Unused Bit 2 – Error Queue Has Data Bit 3 – Questionable Status Summary (Not Used) Bit 4 – Message Available Bit 5 – Event Status Summary Bit 6 – 0 (per IEEE 488.2 section 11.3.2.3) Bit 7 – Operation Status Summary	
Examples	**Command / Query**	**Response *(Description)***
	*SRE 4 *SRE?	*(Sets the service request enable register)* 4 *(Returns the value of the SRE register)*
Related Commands	N/A	

*STB?

Purpose	Queries the Status Byte Register	
Type	IEEE 488.2 Common Command	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	*STB?	
Query Parameters	N/A	
Query Response	Numeric ASCII value from 0 to 255	
Description	The Read Status Byte (STB) query fetches the current contents of the Status Byte Register. See the IEEE 488.2 specification for additional information regarding the Status byte Register and its use. The layout of the Status Register is: Bit 0 – Unused Bit 1 – Unused Bit 2 – Error Queue Has Data Bit 4 – Questionable Status Summary (not used) Bit 5 – Message Available Bit 6 – Master Summary Status Bit 7 – Operation Status Summary	
Examples	**Command / Query**	**Response *(Description)***
	*STB?	16 (*Queries the Status Byte Register)*
Related Commands	N/A	

*TRG

Purpose	Causes a trigger event to occur	
Type	IEEE 488.2 Common Command	
Command Syntax	*TRG	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	N/A	
Query Parameters	N/A	
Query Response	N/A	
Description	The Trigger command causes a trigger event to occur.	
Examples	**Command / Query**	**Response *(Description)***
	*TRG	*(Triggers an event)*
Related Commands	N/A	

*TST?

Purpose	Causes a self-test procedure to occur and queries the results	
Type	IEEE 488.2 Common Command	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	*TST?	
Query Parameters	N/A	
Query Response	Numeric ASCII value from 0 to 143	
Description	The Self-Test query causes the VM3640A to run its self-test procedures and report on the results.	
Examples	**Command / Query**	**Response *(Description)***
	*TST	0 (*Begins the self-test procedure returns the result)*
Related Commands	N/A	

*WAI

Purpose	Halts execution of additional commands and queries until the No Operation Pending message is true	
Type	IEEE 488.2 Common Command	
Command Syntax	*WAI	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	N/A	
Query Parameters	N/A	
Query Response	N/A	
Description	The Wait to Continue command halts the execution of commands and queries until the No Operation Pending message is true. This command makes sure that all previous commands have been executed before proceeding. It provides a way of synchronizing the module with its commander.	
Examples	**Command / Query**	**Response *(Description)***
	*WAI	*(Pauses the execution of additional commands until the No Operation Pending message is true.)*
Related Commands	*OPC	

INSTRUMENT SPECIFIC SCPI COMMANDS

AM[:STATe]

Purpose	Enables or disables amplitude modulation	
Type	Setting	
Command Syntax	AM[:STATe] <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	0	
Query Syntax	AM[:STATe]?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	If amplitude modulation is enabled, the signal present at the modulation input connector is added to the signal generated by the VM3640A.	
Examples	**Command / Query**	**Response *(Description)***
	AM 1 AM?	*(Enables amplitude modulation input)* 1 *(Verifies that the amplitude modulation input is enabled)*
Related Commands	FSK[:STATe] PSK[:STATe]	

CALibration:COUNt?

Purpose	Tracks the number of times calibration has been performed
Type	Query
Command Syntax	N/A
Command Parameters	N/A
***RST Value**	N/A
Query Syntax	CALibration:COUNt?
Query Parameters	N/A
Query Response	Numeric value from 0 to 16,777,215 (after the maximum value, it will wrap to 0)
Description	The Calibration Count query returns the number of times the CALibration:STORe operation has been performed. This provides some indication of FLASH prom wear.  **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:COUN?	24 *(Returns the number of times the CAL:STOR command has been performed.)*

Related Commands	CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe

CALibration[:DATA]:AFCorrection

Purpose	Sets the amplitude gain correction
Type	Setting
Command Syntax	CALibration[:DATA]:AFCorrection <point>,<freq>,<gain>
Command Parameters	<point> = 0 to 63 (an index into the table of gain corrections) <freq> = frequency value <gain> = gain value (should be set to 1 for no correction)
***RST Value**	N/A
Query Syntax	CALibration[:DATA]:AFCorrection? <point>
Query Parameters	<point> = 0 to 63 (an index into the table of gain corrections)
Query Response	Returns the set values for the <freq> and <gain> parameters the following format: <freq>,<gain>
Description	The Calibration Data AFCorrection command sets an amplitude gain correction for a specified frequency point. Frequency/Gain correction factor should only be changed by qualified calibration personnel.  **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:DATA:AFC 3,17,1.1	*(Sets point 3 to 17 MHz with a gain of 1.1)*
	CAL:DATA:AFC? 5	18126524.216, 1.709 *(Queries the frequency and gain setting at point five and returns a value 18.126 MHz and a gain of 1.709)*

Related Commands	CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe

CALibration[:DATA]:AMPLitude[:GAIN]

Purpose	Directly sets the amplitude gain correction
Type	Setting
Command Syntax	CALibration[:DATA]:AMPLitude[:GAIN] <numeric_value>
Command Parameters	<numeric_value> = numeric value from 0 to 1000
***RST Value**	N/A
Query Syntax	CALibration[:DATA]:AMPLitude[:GAIN]?
Query Parameters	N/A
Query Response	Returns the value of the set <numeric_value> parameter for this command
Description	The Calibration Data Amplitude Gain command directly sets the amplitude gain correction. This setting is usually calculated by calibration.  **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:DATA:AMPL:GAIN 2.5 CAL:DATA:AMPL?	*(Sets the amplitude gain correction to 2.5)* 2.5 *(Returns the value of the amplitude gain correction)*

Related Commands	CALibration[:DATA]:AMPLitude:OFFSet CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe

CALibration[:DATA]:AMPLitude:OFFSet

Purpose	Directly sets the offset of the amplitude control
Type	Setting
Command Syntax	CALibration[:DATA]:AMPLitude:OFFSet <numeric_value>
Command Parameters	<numeric_value> = numeric value from 0 to 4095
***RST Value**	N/A
Query Syntax	CALibration[:DATA]:AMPLitude:OFFSet?
Query Parameters	N/A
Query Response	Returns the set value of the <numeric_value> parameter for this command
Description	The Calibration Data Amplitude Offset command directly sets the offset of the amplitude control. This setting is usually calculated by calibration. **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:DATA:AMPL:OFFS 0	*(Sets the offset to 0)*
	CAL:DATA:AMPL:OFFS?	0 *(Returns the set value for the amplitude offset)*

Related Commands	CALibration[:DATA]:AMPLitude[:GAIN] CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe

CALibration[:DATA]:DUTY

Purpose	Calibration for square wave duty cycle
Type	Setting
Command Syntax	CALibration[:DATA]:DUTY <value>
Command Parameters	<value> = 0 to 4095 (DAC value for square wave duty-cycle adjustment)
***RST Value**	N/A
Query Syntax	CALibration[:DATA]:DUTY?
Query Parameters	N/A
Query Response	Returns the set value of the <value> parameter for this command
Description	The Calibration Data Duty command sets the duty cycle of the square wave by inputting a voltage level cutoff-point on the sine wave input. **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:DATA:DUTY 2	*(Enables amplitude modulation input)*
	CAL:DATA:DUTY?	2 *(Returns the value set for square wave duty-cycle adjustment)*

Related Commands	CALibration[:DATA]:SQHI CALibration[:DATA]:SQUare:OFFSet:OFFSet CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe

CALibration[:DATA]:OFFSet[:GAIN]

Purpose	Directly set the offset gain correction	
Type	Setting	
Command Syntax	CALibration[:DATA]:OFFSet[:GAIN] <numeric_value>	
Command Parameters	<numeric_value> = numeric value from 0 to 1000	
***RST Value**	N/A	
Query Syntax	CALibration[:DATA]:OFFSet[:GAIN]?	
Query Parameters	N/A	
Query Response	Returns the set value of the <numeric_value> parameter for this command	
Description	The Calibration Data Offset Gain command directly sets the offset gain correction. This setting is usually calculated by calibration. **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**	
Examples	**Command / Query**	**Response *(Description)***
	CAL:DATA:OFFS 5 CAL:DATA:OFFS?	*(Sets the offset gain correction to 5)* *5 (Returns the offset gain correction value)*
Related Commands	CALibration[:DATA]:OFFSet:OFFSet CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe	

CALibration[:DATA]:OFFSet:OFFSet

Purpose	Directly set the output offset voltage control offset
Type	Setting
Command Syntax	CALibration[:DATA]:OFFSet:OFFSet <numeric_value>
Command Parameters	<numeric_value> = numeric value from 0 to 4095
***RST Value**	N/A
Query Syntax	CALibration[:DATA]:OFFSet:OFFSet?
Query Parameters	N/A
Query Response	Returns the set value of the <numeric_value> parameter for this command
Description	The Calibration Data Offset Offset command directly sets the output offset voltage control offset. This setting is usually calculated by calibration. **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:DATA:OFFS:OFFS 5	*(Sets the output offset voltage and control offset to 5)*
	CAL:DATA:OFFS:OFFS?	5 *(Returns the set value for the output offset voltage control offset)*

Related Commands	CALibration[:DATA]:OFFSet[:GAIN] CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe

CALibration[:DATA]:SQHI

Purpose	Set the positive limit of the square wave output prior to the gain stage
Type	Setting
Command Syntax	CALibration[:DATA]:SQHI <value>
Command Parameters	<value> = number
***RST Value**	N/A
Query Syntax	CALibration[:DATA]:SQHI?
Query Parameters	N/A
Query Response	Returns the set value of the <value> parameter for this command
Description	Used to calibrate the positive peak of the square wave's amplitude to the same as the sine wave at a given gain setting. **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:DATA:SQHI 5	*(Sets the square wave amplitude to 5 V)*
	CAL:DATA:SQHI?	*5 (Returns the set value for the square wave amplitude correction)*

Related Commands	CALibration[:DATA]:DUTY CALibration[:DATA]:SQUare:OFFSet:OFFSet CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe

CALibration[:DATA]:SQLO

Purpose	Set the negative limit of the square wave output prior to the gain stage
Type	Setting
Command Syntax	CALibration[:DATA]:SQLO <value>
Command Parameters	<value> = number
***RST Value**	N/A
Query Syntax	CALibration[:DATA]:SQLO?
Query Parameters	N/A
Query Response	Returns the set value of the <value> parameter for this command
Description	Used to calibrate the negative peak of the square wave's amplitude to the same as the sine wave at a given gain setting.  **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:DATA:SQLO –5 CAL:DATA:SQLO?	*(Sets the square wave amplitude to -5 V)* *-5 (Returns the set value for the square wave amplitude correction)*

Related Commands	CALibration[:DATA]:DUTY CALibration[:DATA]:SQUare:OFFSet:OFFSet CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe

CALibration[:DATA]:SQUare:OFFSet:OFFSet

Purpose	Square wave offset adjustment for calibration
Type	Setting
Command Syntax	CALibration:DATA:SQUare:OFFSet:OFFSet <value>
Command Parameters	<value> = numeric value from 0 to 4095
***RST Value**	N/A
Query Syntax	CALibration:DATA:SQUare:OFFSet:OFFSet?
Query Parameters	N/A
Query Response	Returns the set value of the <value> parameter for this command
Description	The CALibration[:DATA]:SQUare:OFFSet:OFFSet command sets the offset value for the square wave. This number becomes the new "zero" value. **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:DATA:SQU:OFFS:OFFS 2047	*(Sets the "zero" level at 2047)*
	CAL:DATA:SQU:OFFS:OFFS?	2047 *(Returns the set "zero" level)*

Related Commands	CALibration[:DATA]:DUTY CALibration[:DATA]:SQHI CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe

CALibration[:DATA]:ZERO

Purpose	Zero adjust for the output amplifier
Type	Setting
Command Syntax	CALibration:DATA:ZERO <value>
Command Parameters	<value> = numeric value from 0 to 4095 (DAC offset value)
***RST Value**	N/A
Query Syntax	CALibration:DATA:ZERO?
Query Parameters	N/A
Query Response	Returns the set value of the <value> parameter for this command
Description	The Calibration Data Zero command inputs the offset to adjust "zero" for the arbitrary waveforms during calibration.  **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:DATA:ZERO 2047 CAL:DATA:ZERO?	 2047
Related Commands	CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe	

CALibration:DC:LEVel

Purpose	Directly sets the dc level calibration point specified in the command
Type	Setting
Command Syntax	CALibration:DC:LEVel <point>,<dcv_level>,<gain>
Command Parameters	<point> = 0 to 31, index into the calibration data structure <dcv_level> = -10.0 to +10.0, the dc voltage level adjusted <gain> = -5.0 to +5.0, the amount of adjustment
***RST Value**	N/A
Query Syntax	CALibration:DC:LEVel? <point>
Query Parameters	<point> = 0 to 31, index into the calibration data structure
Query Response	Returns the set value of the <dcv_level> and <gain> parameters in the following format: <dcv_level>,<gain>
Description	The Calibration dc Level command sets the gain correction for a specified dc voltage output level.  **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:DC:LEV 3,-8.7,0.9886	*(Sets point 3 to -8.7 V with a gain of 0.9886)*
	CAL:DC:LEV? 3	-8.7, 0.9886 *(Returns the set value for point 3 of –8.7 V with a gain of 0.9886)*

Related Commands	[SOURce:]FUNCtion:DC:VOLTage [SOURce:]VOLTage[:LEVel][:AMPLitude] [SOURce:]VOLTage[:LEVel]:OFFSet

CALibration:SECure:CODE

Purpose	Sets the code required to disable calibration security	
Type	Setting	
Command Syntax	CALibration:SECure:CODE <string>	
Command Parameters	<string> = the code string can be from 1 to 12 ASCII characters in length entered in IEEE 488.2 definite or indefinite length arbitrary block format	
***RST Value**	N/A	
Query Syntax	CALibration:SECure:CODE <string>?	
Query Parameters	N/A	
Query Response	Returns the set value of the <string> parameter	
Description	The Calibration Security Code command sets the code required to disable calibration security. Calibration security must first be disabled before the code can be changed. Before shipping the instrument, the factory code setting is VM3640.  **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**	
Examples	**Command / Query**	**Response *(Description)***
	CAL:SEC:CODE #16VM3640	*(Sets the factory code setting of VM3640)*
Related Commands	CALibration:SECure:STATe CALibration:STORe	

CALibration:SECure:STATe

Purpose	Enable or disable the calibration security
Type	Setting
Command Syntax	CALibration SECure:STATe <boolean>,<string>
Command Parameters	<boolean> = 0 | 1 | OFF | ON <string> = the security code must be entered to disable the security
***RST Value**	<boolean> = 1
Query Syntax	CALibration:SECure:STATe?
Query Parameters	N/A
Query Response	0 | 1
Description	The Calibration Security State enables or disables the calibration security. While security is on, no stores to the non-volatile memory are allowed. In order to disable the security state, the security code must be supplied. If this code is not provided, a "Missing Parameter" error will be returned. To enable the security, the code does not need to be supplied. The security state is enabled by default. **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**

Examples	**Command / Query**	**Response *(Description)***
	CAL:SEC:STAT OFF,#16VM3640	*(Disables calibration security)*
	CAL:SEC:STAT 1	*(Turns the calibration security back on)*
	CAL:SEC:STAT?	1 *(Indicates that calibration security is enabled)*
Related Commands	CALibration:SECure:CODE CALibration:STORe	

CALibration:STATe

Purpose	Enables or disables corrections using calibration data	
Type	Setting	
Command Syntax	CALibration:STATe <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	1	
Query Syntax	CALibration:STATe?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	The Calibration State command enables or disables corrections that were stored from the calibration data. If the Calibration State is enabled, it will allow corrections of the output amplitude and offset voltage levels that are stored in non-volatile memory. If the Calibration State function is disabled, then the default corrections are used. **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**	
Examples	**Command / Query**	**Response *(Description)***
	CAL:STAT ON CAL:STAT?	*(Enables corrections using the calibration data)* 1 *(Indicates that calibration data correction has been enabled)*
Related Commands	CALibration:SECure:CODE CALibration:SECure:STATe CALibration:STORe	

CALibration:STORe

Purpose	Stores calibration data into non-volatile memory	
Type	Event	
Command Syntax	CALibration:STORe	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	N/A	
Query Parameters	N/A	
Query Response	N/A	
Description	The Calibration Store command stores correction data into non-volatile memory. The correction data is calibration data that has been downloaded via the program messages in the CALibration:DATA subsystem. The Calibration Store should only be performed after all the correction data has been finalized.  **Calibration commands should only be executed by qualified personnel. Changing these values incorrectly can cause the instrument to perform improperly.**	
Examples	**Command / Query**	**Response *(Description)***
	CAL:STOR	*(Stores correction data into non-volatile memory)*
Related Commands	CALibration[:DATA] CALibration:SECure:CODE CALibration:SECure:STATe	

CONTrol:IPOWer

Purpose	This command applies power to or removes power from the instrument	
Type	Setting	
Command Syntax	CONTrol:IPOWer <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	ON	
Query Syntax	CONTrol:IPOWer?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	This command applies power to or removes power from the instrument. Note that, following a power off/power on cycle, a five second period of time exists where the instrument cannot be used. This interval allows the instrument to complete the process of initialization.	
Examples	**Command / Query**	**Response *(Description)***
	CONT:IPOW 1 COUNT:IPOW?	*(Turns the instrument power ON)* 1 *(Indicates that instrument power is ON)*
Related Commands	N/A	

FSK[:STATe]

Purpose	Enables or disables Frequency Shift Key modulation input	
Type	Setting	
Command Syntax	FSK[:STATe] <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	0	
Query Syntax	FSK[:STATe]?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	The FSK State command enables or disables frequency shift key modulation input. The default state is disabled.	
Examples	**Command / Query**	**Response *(Description)***
	FSK 1	*(Enables frequency shift key modulation input)*
	FSK?	1 *(Indicates that the frequency shift key has been enabled)*
Related Commands	AM[:STATe] PSK[:STATe]	

PSK[:STATe]

Purpose	Enables or disables Phase Shift Key modulation input	
Type	Setting	
Command Syntax	PSK[:STATe] <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	0	
Query Syntax	PSK[:STATe]?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	The PSK State command enables or disable phase shift key modulation input. The default state is disabled.	
Examples	**Command / Query**	**Response *(Description)***
	PSK 1 PSK?	*(Enables the phase shift key modulation input)* 1 *(Indicates that the shift key modulation input is enabled)*
Related Commands	AM[:STATe] FSK[:STATe]	

RESet

Purpose	Resets all parameters to their default state	
Type	Event	
Command Syntax	RESet	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	N/A	
Query Parameters	N/A	
Query Response	N/A	
Description	The Reset command resets all the parameters to their default state.	
Examples	**Command / Query**	**Response *(Description)***
	RES	*(Disables calibration security)*
Related Commands	N/A	

SENSe:CORRection:IMPedance

Purpose	Allows the user to set the value of the load resistor connected to the output	
Type	Setting	
Command Syntax	SENSe:CORRection:IMPedance <resistor_value>	
Command Parameters	<resistor_value> = 10.0 to 100e6 Ω	
***RST Value**	50.0 Ω	
Query Syntax	SENSe:CORRection:IMPedance?	
Query Parameters	N/A	
Query Response	Returns the set value of the <resistor_value> parameter	
Description	The VM3640A is designed to drive a 50 Ω load. This command allows the user to input a different value for use in the gain calculations.	
Examples	**Command / Query**	**Response *(Description)***
	SENS:CORR:IMP 200.57 SENS:CORR:IMP?	*(Sets the load resistance value to 200.57 Ω)* 200.57 *(Indicates that the load resistance value was set to 200.57 Ω)*
Related Commands	[SOURce:]FUNCtion:DC:VOLTage [SOURce:]VOLTage[:LEVel][:AMPLitude] [SOURce:]VOLTage[:LEVel]:OFFSet	

[SOURce:]BURSt:COUNt

Purpose	Sets the number of cycles per burst	
Type	Setting	
Command Syntax	[SOURce:]BURSt:COUNt <count>	
Command Parameters	<count> = integer from 1 to 65535	
***RST Value**	1	
Query Syntax	[SOURce:]BURSt:COUNt?	
Query Parameters	N/A	
Query Response	Returns the set value of the <count> parameter for this command	
Description	The Source Burst Count command sets the number of output cycles per burst. The default value is 1 cycle.	
Examples	**Command / Query**	**Response *(Description)***
	BURS:COUN 12 BURS:COUN?	*(Sets the burst count to twelve cycles)* 12 *(Returns the set value for the burst count)*
Related Commands	[SOURce:]BURSt:PHASe [SOURce:]BURSt:RATE [SOURce:]BURSt[:STATe]	

[SOURce:]BURSt:PHASe

Purpose	Sets the phase of the output waveform	
Type	Setting	
Command Syntax	[SOURce:]BURSt:PHASe <phase>	
Command Parameters	<phase> = value between 0 and +359 degrees	
***RST Value**	0	
Query Syntax	[SOURce:]BURSt:PHASe?	
Query Parameters	N/A	
Query Response	Returns the set value of the <phase> parameter for this command	
Description	The Source Burst Phase command set the starting phase for the burst. The default setting is 0 degrees.	
Examples	**Command / Query**	**Response *(Description)***
	BURS:PHAS 30 BURS:PHAS?	*(Sets the burst-starting phase to 30 degrees)* 30 *(Returns the set starting phase value)*
Related Commands	[SOURce:]BURSt:COUNt [SOURce:]BURSt:RATE [SOURce:]BURSt[:STATe]	

[SOURce:]BURSt:RATE

Purpose	Sets the burst repetition rate	
Type	Setting	
Command Syntax	[SOURce:]BURSt:RATE <rate>	
Command Parameters	<rate> = value from 0.10 to 100000	
***RST Value**	100	
Query Syntax	[SOURce:]BURSt:RATE?	
Query Parameters	N/A	
Query Response	Returns the set value of the <rate> parameter for this command	
Description	The Source Burst Rate command sets the burst repetition or frequency of the burst. The default value is 10 Hz. **Note**: Burst Rate must ≤ Frequency Setting.	
Examples	**Command / Query**	**Response *(Description)***
	BURS:RATE 10 BURS:RATE?	*(Sets the burst rate to 10 Hz)* 10 *(Indicates that the burst rate is set to 10 Hz)*
Related Commands	[SOURce:]BURSt:COUNt [SOURce:]BURSt:PHASe [SOURce:]BURSt[:STATe]	

[SOURce:]BURSt[:STATe]

Purpose	Enables or disables burst mode	
Type	Setting	
Command Syntax	[SOURce:]BURSt[:STATe] <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	0	
Query Syntax	[SOURce:]BURSt[:STATe]?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	The Source Burst State command enables or disables the burst mode. The default setting is disabled.	
Examples	**Command / Query**	**Response *(Description)***
	BURS 1	*(Enables burst mode)*
	BURS?	1 *(Indicates that burst mode is enabled)*
Related Commands	[SOURce:]BURSt:COUNt [SOURce:]BURSt:PHASe [SOURce:]BURSt:RATE	

[SOURce:]CLOCk:CONFigure

Purpose	Sets front panel clock as an input or output	
Type	Setting	
Command Syntax	[SOURce:]CLOCk:CONFigure <clock_dir>	
Command Parameters	<clock_dir> = INPut | OUTPut	
***RST Value**	INP	
Query Syntax	[SOURce:]CLOCk:CONFigure?	
Query Parameters	N/A	
Query Response	INP | OUTPut	
Description	The Source Clock Configure command configures the front-panel CLOCK IN/OUT as an input or output. The default setting is INPut.	
Examples	**Command / Query**	**Response *(Description)***
	CLOC:CONF OUTP	*(Sets the front panel CLOCK IN/OUT connector as an output)*
	CLOC:CONF?	OUTP *(Indicates that the CLOCK IN/OUT connector is set as an output)*
Related Commands	SOURce:ROSC:SOURce	

[SOURce:]FREQuency[:CW]

Purpose	Controls the frequency of the output signal	
Type	Setting	
Command Syntax	[SOURce:]FREQuency[:CW] <numeric_value>	
Command Parameters	<numeric_value> = 0.1 to 20e6 for sine and square waves <numeric_value> = 0.1 to 1.0e6 for all other waveforms <numeric_value> = 0.1 to 1.0e6 for all waveforms in Burst Mode	
***RST Value**	1000	
Query Syntax	[SOURce:]FREQuency[:CW]?	
Query Parameters	N/A	
Query Response	Returns the set value of the <numeric_value> parameter for this command	
Description	The Source Frequency CW command controls the frequency of the output signal.	
Examples	**Command / Query**	**Response *(Description)***
	FREQ 1.0	
Related Commands	[SOURce:]FUNCtion[:SHAPe]	

[SOURce:]FUNCtion:DC:VOLTage

Purpose	Sets the dc output voltage	
Type	Setting	
Command Syntax	[SOURce:]FUNCtion:DC:VOLTage <dc_volts>	
Command Parameters	<dc_volts> = number from -6 to +6	
***RST Value**	1	
Query Syntax	[SOURce:]FUNCtion:DC:VOLTage?	
Query Parameters	N/A	
Query Response	Returns the set value of the <dc_volts> parameter	
Description	The Source Function dc Voltage command sets the dc voltage output. The default setting is 1 V dc.	
Examples	**Command / Query**	**Response *(Description)***
	FUNC:DC:VOLT 2.5	*(Sets the output voltage to +2.5 V dc)*
	FUNC:DC:VOLT?	2.5 *(Indicates that the output voltage is set to +2.5 V dc)*
Related Commands	[SOURce:]VOLTage[:LEVel][:AMPLitude] [SOURce:]VOLTage[:LEVel]:OFFSet	

[SOURce:]FUNCtion:PULSe:WIDTh

Purpose	Sets the pulse width
Type	Setting
Command Syntax	[SOURce:]FUNCtion:PULSe:WIDTh <pulse_width>[,<pulse_freq>]
Command Parameters	<pulse_width> = 100e-9 to 1.0 <pulse_freq> = 0.1 to 1.0e6
***RST Value**	<pulse_width> = 1 μs <pulse_freq> = 1000 Hz
Query Syntax	[SOURce:]FUNCtion:PULSe:WIDTh?
Query Parameters	N/A
Query Response	Returns the set value of the <pulse_width> parameter for this command
Description	The Source Function Pulse Width command sets the pulse width of the output waveform. The default setting is 1 µs. Pulse width must be ≥ 0.001/Fout Pulse width must be ≤ 0.999/Fout For short pulses at low repetition rates (e.g. 1 μs @ 1 Hz): 1. `FREQ 10E3` *Sets the frequency to 10 kHz* 2. `FUNC:PULSE:WIDTH 1e-6` *Sets the pulse width to 1 μs* 3. `TRIG:MODE INT` *Sets the trigger to internal clock* 4. `TRIG:RATE 1` *Sets the repetition rate to 1 Hz* The optional pulse frequency parameter adds another route to set the frequency in addition to the [SOURce:]FREQuency[:CW] command.

Examples	**Command / Query**	**Response *(Description)***
	FUNC:PULS:WIDT 30e-6	*(Sets the pulse width to 30 μs)*
	FUNC:PULS:WIDT?	30 *(Indicates that the pulse width is 30 μs)*
	FUNC:PULS:WIDT 30e-6,10e3	*(Sets the pulse width to 30 μs and the pulse frequency to 10 kHz)*

Related Commands	[SOURce:]FREQuency[:CW] [SOURce:]FUNCtion[:SHAPe]

[SOURce:]FUNCtion[:SHAPe]

Purpose	Selects the output waveform shape	
Type	Setting	
Command Syntax	[SOURce:]FUNCtion[:SHAPe] <shape>	
Command Parameters	<shape> = SINusoid | SQUare | TRIangle | PRNoise | PRAMp | NRAMp | EXPRise | EXPFall | CARDiac | DC | PULSe | SINC	
***RST Value**	SIN	
Query Syntax	[SOURce:]FUNCtion[:SHAPe]?	
Query Parameters	N/A	
Query Response	Returns the set value of the <shape> parameter	
Description	The Source Function Shape command selects the shape of the output waveform. The default setting is a sinusoid waveform.	
Examples	**Command / Query**	**Response *(Description)***
	FUNC SQU FUNC?	*(Sets the output waveform shape to a square wave)* SQU *(Indicates that the waveform shape is a square wave)*
Related Commands	[SOURce:]FUNCtion:PULSe:WIDTh [SOURce:]FREQuency[:CW]	

[SOURce:]FUNCtion:SQUare[:DUTY]

Purpose	Sets the duty cycle of the square wave function	
Type	Command	
Command Syntax	[SOURce:]FUNCtion:SQUare[:DUTY] <duty_cycle>	
Command Parameters	<duty_cycle> = 20.0 to 80.0	
***RST Value**	50.0	
Query Syntax	[SOURce:]FUNCtion:SQUare[:DUTY]?	
Query Parameters	N/A	
Query Response	Returns the set value of the <duty_cycle> parameter	
Description	The Source Function Square Duty command sets the duty cycle of the square wave output. The default setting is 50.0.	
Examples	**Command / Query**	**Response *(Description)***
	FUNC:SQU 20 FUNC:SQU?	*(Sets the square wave duty-cycle to 20.0)* 20.0 *(Indicates that the square wave duty-cycle is set to 20.0)*
Related Commands	[SOURce:]FUNCtion[:SHAPe]	

[SOURce:]FUNCtion:USER:DIRectory?

Purpose	List the user-defined waveforms	
Type	Query	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	[SOURce:]FUNCtion:USER:DIRectory?	
Query Parameters	N/A	
Query Response	Returns the stored user-defined waveforms	
Description	The Source Function User Directory query lists the user-defined waveforms that have been stored.	
Examples	**Command / Query**	**Response *(Description)***
	FUNC:USER:DIR?	*(Lists the stored user-defined waveforms)*
Related Commands	[SOURce:]FUNCtion:USER[:NAME]	

[SOURce:]FUNCtion:USER:FREE?

Purpose	Queries the amount of free space for user-defined waveforms	
Type	Query	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	[SOURce:]FUNCtion:USER:FREE?	
Query Parameters	N/A	
Query Response	Returns the number of samples of free space	
Description	The Source Function User Free query shows how much user-defined space for waveforms is available. Samples are 12 bits/2 bytes.	
Examples	**Command / Query**	**Response *(Description)***
	FUNC:USER:FREE?	126976 *(Shows that the sample space available is nearly 128 k Samples)*
Related Commands	[SOURce:]FUNCtion:USER:DIRectory	

[SOURce:]FUNCtion:USER[:NAME]

Purpose	Sets the output waveform shape to a user-defined waveform	
Type	Setting	
Command Syntax	[SOURce:]FUNCtion:USER[:NAME] <block_data>	
Command Parameters	<block_data> = IEEE 488.2 definite or indefinite length arbitrary block format	
***RST Value**	N/A	
Query Syntax	[SOURce:]FUNCtion:USER[:NAME]?	
Query Parameters	N/A	
Query Response	IEEE 488.2 definite or indefinite length arbitrary block format	
Description	The Source Function User Name command sets the user-defined waveform in an IEEE 488.2 definite or indefinite length arbitrary block format. For example, #18username Defines the name of the user-defined waveform (username). This sets which user-defined waveform will be output. Once set, the user need only type in "FUNC:USER" to use this waveform, which will remain unchanged until another user-defined waveform is selected by the user.	
Examples	**Command / Query**	**Response *(Description)***
	FUNC:USER #18username	*(Sets the user-defined output wave form to "username")*
	FUNC:USER?	#18username *(Returns the user-defined waveform #18username)*
	FUNC:PULS:WIDT 30e-6,10e3	*(Sets the pulse width to 30 µs and the pulse frequency to 10 kHz)*
Related Commands	[SOURce:]FUNCtion:USER:FREE? [SOURce:]FUNCtion:USER:WAVE	

[SOURce:]FUNCtion:USER:WAVE

Purpose	Sets the name and stores a user-defined waveform	
Type	Setting	
Command Syntax	[SOURce:]FUNCtion:USER:WAVE <block_data>	
Command Parameters	<block_data> = IEEE 488.2 definite or indefinite length arbitrary block format	
***RST Value**	N/A	
Query Syntax	N/A	
Query Parameters	N/A	
Query Response	N/A	
Description	The Source Function User Wave command sets the name and size of a user-defined waveform in an IEEE 488.2 definite or indefinite length arbitrary block format. For example: #213username,4095 Defines the name of the user-defined waveform (username) and the size of the waveform (4095). The waveform name and the waveform size are separated by a comma. **Note**: Waveform data must be transferred in "A16 Register Mode."	
Examples	**Command / Query**	**Response *(Description)***
	FUNC:USER #213username,4095	*(Sets the waveform name to "username" and sets its size to 4 k)*
Related Commands	[SOURce:]FUNCtion:USER:FREE? [SOURce:]FUNCtion:USER[:NAME]	

[SOURce:]MARKer:POLarity

Purpose	Sets the polarity for the output marker	
Type	Setting	
Command Syntax	[SOURce:]MARKer:POLarity <mark_pol>	
Command Parameters	<mark_pol> = POSitive | NEGative	
***RST Value**	POS	
Query Syntax	[SOURce:]MARKer:POLarity?	
Query Parameters	N/A	
Query Response	POS | NEG	
Description	The Source Marker Polarity sets the polarity for the output marker. The default setting is positive.	
Examples	**Command / Query**	**Response *(Description)***
	SOUR:MARK:POS NEG	*(Sets the output marker polarity to negative)*
Related Commands	[SOURce:]MARKer:POSition [SOURce:]MARKer[:STATe]	

[SOURce:]MARKer:POSition

Purpose	Sets the marker to a specified point
Type	Setting
Command Syntax	[SOURce:]MARKer:POSition <mark_pos>
Command Parameters	<mark_pos> = 0 to 4095 *for standard waveforms* 0 to pattern_length – 1 *for user waveforms* If the marker is set to position zero (0) for a standard waveform, it will actually be placed at position one (1). This is to avoid having the marker output active while the waveform is waiting to be triggered.
***RST Value**	0
Query Syntax	[SOURce:]MARKer:POSition?
Query Parameters	N/A
Query Response	Returns the set value of the <mark_pos> parameter
Description	The Source Marker Position command sets the position of the output marker to a specific point, when the Marker Source is set to 'BBIT' (arbitrary waveform only). The default setting is 0.
Examples	**Command / Query** — **Response *(Description)***
	MARK:POS 100 — *(Sets the output marker position to 100)*
Related Commands	[SOURce:]MARKer:POLarity [SOURce:]MARKer:SOURce [SOURce:]MARKer[:STATe]

[SOURce:]MARKer:SOURce

Purpose	Selects the source to generate the output marker	
Type	Setting	
Command Syntax	[SOURce:]MARKer:SOURce <mark_src>	
Command Parameters	<mark_src> = ZCROss | BBITs	
***RST Value**	ZCRO	
Query Syntax	[SOURce:]MARKer:SOURce?	
Query Parameters	N/A	
Query Response	ZCRO | BBIT	
Description	The Source Marker Source command sets the source to generate the output marker: ZCROss: Selects a comparator output. The comparator output is high if the instantaneous signal is above mid-scale and low if the signal is below mid-scale. The Zero CROssing marker is valid for all waveforms. However, waveforms with multiple zero crossings, such as cardiac or sinc, will generate multiple markers per cycle. BBITs: With this setting, the Marker comes from a bit in Trace RAM. The marker is generated from pattern RAM and is active for one sample time during the generation of waveforms generated by the DAC. For this reason, the BBITs marker cannot be used with SINe and SQUare waveforms. The location of the marker with respect to the beginning of the waveform is set by the MARK:POS command. The default setting is ZCROss.	
Examples	**Command / Query**	**Response *(Description)***
	MARK:SOUR BBIT MARK:SOUR?	*(Selects BBITs are the output marker)* BBIT *(Indicates that BBIT is selected as the output source)*
Related Commands	[SOURce:]MARKer:POLarity [SOURce:]MARKer:POSition [SOURce:]MARKer[:STATe]	

[SOURce:]MARKer[:STATe]

Purpose	Enables or disables the marker output	
Type	Setting	
Command Syntax	[SOURce:]MARKer[:STATe] <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	1	
Query Syntax	[SOURce:]MARKer[:STATe]?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	The Source Marker State command enables or disables the output marker. The default setting is enabled.	
Examples	**Command / Query**	**Response *(Description)***
	MARK 0	*(Disables the output marker)*
Related Commands	[SOURce:]MARKer:POLarity [SOURce:]MARKer:POSition	

[SOURce:]OUTPut:FILTer[:LPASs][:STATe]

Purpose	Enables or bypasses the output filter	
Type	Setting	
Command Syntax	[SOURce:]OUTPut:FILTer[:LPASs][:STATe] <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	1	
Query Syntax	[SOURce:]OUTPut:FILTer[:LPASs][:STATe]?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	The Source Output Filter Low-Pass State command enables or disables the output filter between the output of the waveform DAC and the preamplifier. The default setting is enabled.	
Examples	**Command / Query**	**Response *(Description)***
	OUTP:FILT 0	*(Bypasses the output filter)*
	OUTP:FILT?	0 *(Indicates that the output filter is bypassed or disabled)*
Related Commands	[SOURce:]OUTPut:FILTer[:LPASs]:TYPe	

[SOURce:]OUTPut:FILTer[:LPASs]:TYPe

Purpose	Selects the output filter type	
Type	Setting	
Command Syntax	[SOURce:]OUTPut:FILTer[:LPASs]:TYPe <filter_type>	
Command Parameters	<filter_type> = BESSel | ELLiptical	
***RST Value**	ELLiptical	
Query Syntax	[SOURce:]OUTPut:FILTer[:LPASs]:TYPe?	
Query Parameters	N/A	
Query Response	BESS | ELL	
Description	The Source Output Filter Low-Pass Type command selects the output filter type. The Bessel type has the flattest response with constant group delay. The Elliptical has the steepest cut-off. The default type is Elliptical.	
Examples	**Command / Query**	**Response *(Description)***
	OUTP:FILT:TYP ELL OUTP:FILT:TYP?	*(Sets the output filter type to elliptical)* ELL *(Indicates that the output filter type is elliptical)*
Related Commands	[SOURce:]OUTPut:FILTer[:LPASs][:STATe]	

[SOURce:]OUTPut[:STATe]

Purpose	Enables or disables the MAIN OUT output	
Type	Setting	
Command Syntax	[SOURce:]OUTPut[:STATe] <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	0	
Query Syntax	[SOURce:]OUTPut[:STATe]?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	The Source Output State command controls the MAIN OUT output relay. The default setting is disabled.	
Examples	**Command / Query**	**Response *(Description)***
	FUNC:PULS:WIDT 30e-6	*(Sets the pulse width to 30 µs)*
	FUNC:PULS:WIDT?	30 *(Indicates that the pulse width is 30 µs)*
	FUNC:PULS:WIDT 30e-6,10e3	*(Sets the pulse width to 30 µs and the pulse frequency to 10 kHz)*
Related Commands	N/A	

[SOURce:]ROSC:SOURce

Purpose	Selects the source for the reference oscillator
Type	Setting
Command Syntax	[SOURce:]ROSC:SOURce <rosc_src>
Command Parameters	<rosc_src> = PLL | EXT
***RST Value**	PLL
Query Syntax	[SOURce:]ROSC:SOURce?
Query Parameters	N/A
Query Response	PLL | EXT

Description	The Source ROSC Source command selects the source for the reference oscillator:
	
	PLL: *Phase Look-Loop*
	EXT: *Signal from the CLOCK IN/OUT – must be set as input*
	The default setting is PLL.
	
	The external clock source connects directly to the pattern RAM address counter and to the 'DDS' chip. For waveforms from pattern RAM, the output frequency will be Ext_Clock_Freq / Number_of_Samples. The 'standard' DAC waveforms are 4096 samples. For DDS waveforms, sine & square, the last value in the DDS frequency register determines the output frequency.
	
	Setting the DDS clock if the clock source is NOT External:
	dPhase = (long)(Fout * (2^32/50MHz))
	While(1) // loop til Mclk is <= 50MHz
	{ Mclk = Fout * (2^32) / (double)dPhase;
	if(Mclk <= 50.0e6)
	break;
	else
	dPhase += 1;
	}
	
	At this point, a long integer dPhase exists that yields a DDS clock ≤ 50 MHz. dPhase is placed in to the DDS Frequency register (32 bits).
	
	Now: Fout = dPhase * Mclk / (2^32)
	
	If at this point, the Reference Oscillator is switched to EXT,
	Fout = dPhase * Fext_ref / (2^32)

Examples	**Command / Query**	**Response *(Description)***
	ROSC:SOUR EXT	*(Sets the source to the CLOCK IN/OUT input)*
	ROSC:SOUR?	EXT *(Indicates that the source is set to the CLOCK IN/OUT input)*
Related Commands	[SOURce:]CLOCk:CONFigure	

[SOURce:]SWEep:COUNt

Purpose	Sets the number of sweeps enabled per trigger event	
Type	Setting	
Command Syntax	[SOURce:]SWEep:COUNt <count>	
Command Parameters	<count> = 0 to 65535	
***RST Value**	0 (continuous)	
Query Syntax	[SOURce:]SWEep:COUNt?	
Query Parameters	N/A	
Query Response	Returns the set value of the <count> parameter for this command	
Description	The Source Sweep Count command sets the number of sweeps per trigger event. The default value is 0, which is a continuous sweep.	
Examples	**Command / Query**	**Response *(Description)***
	SWE:COUN 10	*(Sets the number of sweeps to 10 per trigger)*
Related Commands	[SOURce:]SWEep:DIRection [SOURce:]SWEep:MODE [SOURce:]SWEep:POINts [SOURce:]SWEep:SPACing [SOURce:]SWEep:STARt [SOURce:]SWEep:STOP [SOURce:]SWEep[:STATe] [SOURce:]SWEep:TIME	

[SOURce:]SWEep:DIRection

Purpose	Sets the sweep direction	
Type	Setting	
Command Syntax	[SOURce:]SWEep:DIRection <sweep_direction>	
Command Parameters	<sweep_direction> = UP for low to high, DOWN for high to low	
***RST Value**	UP	
Query Syntax	[SOURce:]SWEep:DIRection?	
Query Parameters	N/A	
Query Response	UP | DOWN	
Description	The Source Sweep Direction command sets the sweep direction from low to high (UP), or from high to low (DOWN). The default setting is from low to high (UP).	
Examples	**Command / Query**	**Response *(Description)***
	SWE:DIR DOWN	*(Sets the sweep direction to sweep from high to low [down])*
Related Commands	[SOURce:]SWEep:COUNt [SOURce:]SWEep:MODE [SOURce:]SWEep:POINts [SOURce:]SWEep:SPACing [SOURce:]SWEep:STARt [SOURce:]SWEep:STOP [SOURce:]SWEep[:STATe] [SOURce:]SWEep:TIME	

[SOURce:]SWEep:MODE

Purpose	Sets the sweep mode	
Type	Setting	
Command Syntax	[SOURce:]SWEep:MODE <sweep_mode>	
Command Parameters	<sweep_mode> = CRESet | CREVerse	
***RST Value**	CRESet	
Query Syntax	[SOURce:]SWEep:MODE?	
Query Parameters	N/A	
Query Response	CRES | CREV	
Description	The Source Sweep Mode command sets the sweep mode: CRESet: Sweeps from the starting sweep frequency to the stop sweep frequency, then returns back to the starting sweep frequency. This is a continuous loop. CREVerse: Sweeps from the starting sweep frequency to the stop sweep frequency, then sweeps back to the start sweep frequency. This is a continuous loop. Horizontal Sweep Output is available only while in sweep mode. The horizontal sweep output ramps from 0 V, at the beginning of a sweep cycle, to 5 V, at the end of a sweep cycle. It is intended to provide the user with a representation of the progress through the sweep. The slope of the horizontal sweep output is linear, regardless of the sweep mode selected.	
Examples	**Command / Query**	**Response *(Description)***
	SWE:MODE CREV	*(Sets the sweep mode to creverse)*
Related Commands	[SOURce:]SWEep:COUNt [SOURce:]SWEep:DIRection [SOURce:]SWEep:POINts [SOURce:]SWEep:SPACing [SOURce:]SWEep:STARt [SOURce:]SWEep:STOP [SOURce:]SWEep[:STATe] [SOURce:]SWEep:TIME	

[SOURce:]SWEep:POINts

Purpose	Sets the sweep points	
Type	Setting	
Command Syntax	[SOURce:]SWEep:POINts <swp_pts>	
Command Parameters	<swp_pts> = 2 to 4096	
***RST Value**	100	
Query Syntax	[SOURce:]SWEep:POINts?	
Query Parameters	N/A	
Query Response	Returns the set value of the <swp_pts> parameter	
Description	The Source Sweep Points command sets the number of discrete frequency steps between the sweep start and sweep stop frequencies. Sweep Time/Sweep Points must be > 500 μs.	
Examples	**Command / Query**	**Response *(Description)***
	SWE:POIN 10	*(Sets the sweep points to 10)*
Related Commands	[SOURce:]SWEep:COUNt [SOURce:]SWEep:DIRection [SOURce:]SWEep:MODE [SOURce:]SWEep:SPACing [SOURce:]SWEep:STARt [SOURce:]SWEep:STOP [SOURce:]SWEep[:STATe] [SOURce:]SWEep:TIME	

[SOURce:]SWEep:SPACing

Purpose	Sets the sweep spacing	
Type	Setting	
Command Syntax	[SOURce:]SWEep:SPACing <sweep_space>	
Command Parameters	<sweep_space> = LINear | LOGarithmic	
***RST Value**	LIN	
Query Syntax	[SOURce:]SWEep:SPACing?	
Query Parameters	N/A	
Query Response	LIN | LOG	
Description	The Source Sweep Spacing command sets the sweep spacing, or frequency vs. time, as follows: Linear: *The output frequency is swept linearly between the start and stop frequencies.* Logarithmic: *The output frequency is swept on a logarithmic curve between the start and stop frequencies.* The default setting is LINear.	
Examples	**Command / Query**	**Response *(Description)***
	SWE:SPAC LOG	*(Sets the sweep spacing to be logarithmic)*
Related Commands	[SOURce:]SWEep:COUNt [SOURce:]SWEep:DIRection [SOURce:]SWEep:MODE [SOURce:]SWEep:POINts [SOURce:]SWEep:STARt [SOURce:]SWEep:STOP [SOURce:]SWEep[:STATe] [SOURce:]SWEep:TIME	

[SOURce:]SWEep:STARt

Purpose	Sets the starting sweep frequency	
Type	Setting	
Command Syntax	[SOURce:]SWEep:STARt <freq>	
Command Parameters	<freq> = 1e-1 to 2e7	
***RST Value**	1000 (1e3)	
Query Syntax	[SOURce:]SWEep:STARt?	
Query Parameters	N/A	
Query Response	Returns the set value of the <freq> parameter for this command	
Description	The Source Sweep Start command sets the sweep starting frequency. The default is set to 1000 (1e3).	
Examples	**Command / Query**	**Response *(Description)***
	SWE:STAR 2000	*(Sets the sweep starting frequency to 2000 Hz)*
Related Commands	[SOURce:]SWEep:COUNt [SOURce:]SWEep:DIRection [SOURce:]SWEep:MODE [SOURce:]SWEep:POINts [SOURce:]SWEep:SPACing [SOURce:]SWEep:STOP [SOURce:]SWEep[:STATe] [SOURce:]SWEep:TIME	

[SOURce:]SWEep:STOP

Purpose	Sets the ending sweep frequency	
Type	Setting	
Command Syntax	[SOURce:]SWEep:STOP <freq>	
Command Parameters	<freq> = 1e-1 to 2e7	
***RST Value**	2000 (2e3)	
Query Syntax	[SOURce:]SWEep:STOP?	
Query Parameters	N/A	
Query Response	Returns the set value of the <freq> parameter for this command	
Description	The Source Sweep Stop command sets the sweep stop frequency. The default is set to 2000 (2e3).	
Examples	**Command / Query**	**Response *(Description)***
	SWE:STOP 3000	*(Sets the sweep stop frequency to 3000 Hz)*
Related Commands	[SOURce:]SWEep:COUNt [SOURce:]SWEep:DIRection [SOURce:]SWEep:MODE [SOURce:]SWEep:POINts [SOURce:]SWEep:SPACing [SOURce:]SWEep:STARt [SOURce:]SWEep[:STATe] [SOURce:]SWEep:TIME	

[SOURce:]SWEep[:STATe]

Purpose	Enables or disables the sweep function	
Type	Setting	
Command Syntax	[SOURce:]SWEep[:STATe] <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	0	
Query Syntax	[SOURce:]SWEep[:STATe]?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	The Source Sweep State command enables or disables the sweep mode. The default setting is disabled. Sweep Mode is only valid for sine and square waveforms.	
Examples	**Command / Query**	**Response *(Description)***
	SWE:STAT 1	*(Enables sweep mode)*
Related Commands	[SOURce:]SWEep:COUNt [SOURce:]SWEep:DIRection [SOURce:]SWEep:MODE [SOURce:]SWEep:POINts [SOURce:]SWEep:SPACing [SOURce:]SWEep:STARt [SOURce:]SWEep:STOP [SOURce:]SWEep:TIME	

[SOURce:]SWEep:TIME

Purpose	Sets the sweep duration time	
Type	Setting	
Command Syntax	[SOURce:]SWEep:TIME <time_val>	
Command Parameters	<time_val> = 0.0005 s to 348160 s	
***RST Value**	0.100 s	
Query Syntax	[SOURce:]SWEep:TIME?	
Query Parameters	N/A	
Query Response	Returns the set value of the <time_val> parameter for this command	
Description	The Source Sweep Time command sets the sweep duration time in seconds. time: *depends on the number of points in the sweep* time/points: *must be ≥ 250e-6* Time must not exceed 348160 seconds. The default setting is 0.100 seconds.	
Examples	**Command / Query**	**Response *(Description)***
	SWE:TIME 0.030	*(Sets the sweep time to 30 ms)*
Related Commands	[SOURce:]SWEep:COUNt [SOURce:]SWEep:DIRection [SOURce:]SWEep:MODE [SOURce:]SWEep:POINts [SOURce:]SWEep:SPACing [SOURce:]SWEep:STARt [SOURce:]SWEep:STOP [SOURce:]SWEep[:STATe]	

[SOURce:]SYNC:MODE

Purpose	Selects the sync mode	
Type	Setting	
Command Syntax	[SOURce:]SYNC:MODE <sync_mode>	
Command Parameters	<sync_mode> = MAST | SLAV	
***RST Value**	SLAV	
Query Syntax	[SOURce:]SYNC:MODE?	
Query Parameters	N/A	
Query Response	MAST | SLAV	
Description	The Source Sync Mode command is used to configure each module as a master or slave, when configuring two or more VM3640A modules to sync simultaneously. The default setting is master.	
Examples	**Command / Query**	**Response *(Description)***
	SYNC:MODE SLAV	*(Configures the module to be a slave)*
	SYNC:MODE?	SLAV *(Indicates that the module is configured as a slave)*
Related Commands	[SOURce:]SYNC[STATe:]	

[SOURce:]SYNC[:STATe]

Purpose	Enables or disables simultaneous sync mode	
Type	Setting	
Command Syntax	[SOURce:]SYNC[:STATe] <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	0	
Query Syntax	[SOURce:]SYNC[:STATe]?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	The Source Sync State command enables or disables sync mode, when configuring two or more VM3640A modules to sync simultaneously. The default setting is disabled. Note that each module must first be configured as a master or slave before using this function.	
Examples	**Command / Query**	**Response *(Description)***
	SYNC 1 SYNC?	*(Enables simultaneous synchronize mode)* 1 *(Indicates that simultaneous synchronize mode is enabled)*
Related Commands	[SOURce:]SYNC:MODE	

[SOURce:]TRIGger:GATE

Purpose	Selects gate mode	
Type	Setting	
Command Syntax	[SOURce:]TRIGger:GATE <boolean>	
Command Parameters	<boolean> = 0 | 1 | OFF | ON	
***RST Value**	0	
Query Syntax	[SOURce:]TRIGger:GATE?	
Query Parameters	N/A	
Query Response	0 | 1	
Description	The Source Trigger Gate command selects the gated mode of operation, when the trigger source is external. The default value is disabled.	
Examples	**Command / Query**	**Response *(Description)***
	TRIG:GATE 1	*(Selects trigger gate mode)*
Related Commands	[SOURce:]TRIGger[:MODE] [SOURce:]TRIGger:POLarity [SOURce:]TRIGger:RATE [SOURce:]TRIGger:TTLT	

[SOURce:]TRIGger[:MODE]

Purpose	Selects the trigger source	
Type	Setting	
Command Syntax	[SOURce:]TRIGger[:MODE] <trg_mode>	
Command Parameters	<trg_mode> = BUS | EXTernal | IMMediate | INTernal	
***RST Value**	IMM	
Query Syntax	[SOURce:]TRIGger[:MODE]?	
Query Parameters	N/A	
Query Response	BUS | EXT | IMM | INT	
Description	The Source Trigger Mode command selects the trigger source. The default selection is IMMediate.	
Examples	**Command / Query**	**Response *(Description)***
	TRIG:MODE EXT	*(Selects external mode of trigger)*
Related Commands	[SOURce:]TRIGger:GATE [SOURce:]TRIGger:POLarity [SOURce:]TRIGger:RATE [SOURce:]TRIGger:TTLT	

[SOURce:]TRIGger:POLarity

Purpose	Selects the polarity of the trigger signal	
Type	Setting	
Command Syntax	[SOURce:]TRIGger:POLarity <trig_pol>	
Command Parameters	<trig_pol> = POSitive | NEGative	
***RST Value**	POS	
Query Syntax	[SOURce:]TRIGger:POLarity?	
Query Parameters	N/A	
Query Response	POS | NEG	
Description	The Source Trigger Polarity command set the trigger polarity to positive or negative. The default setting is positive.	
Examples	**Command / Query**	**Response *(Description)***
	TRIG:POL NEG	*(Sets the trigger negative)*
Related Commands	[SOURce:]TRIGger:GATE [SOURce:]TRIGger[:MODE] [SOURce:]TRIGger:RATE [SOURce:]TRIGger:TTLT	

[SOURce:]TRIGger:RATE

Purpose	Sets the period for the internal trigger	
Type	Setting	
Command Syntax	[SOURce:]TRIGger:RATE <rate>	
Command Parameters	<rate> = 0.1 to 100000	
***RST Value**	100	
Query Syntax	[SOURce:]TRIGger:RATE?	
Query Parameters	N/A	
Query Response	Returns the set value of the <rate> parameter for this command	
Description	The Source Trigger Rate command sets the period for the internal trigger. The default setting is 100.	
Examples	**Command / Query**	**Response *(Description)***
	TRIG:RATE 10	*(Sets the internal trigger period to 10)*
Related Commands	[SOURce:]TRIGger:GATE [SOURce:]TRIGger[:MODE] [SOURce:]TRIGger:POLarity [SOURce:]TRIGger:TTLT	

[SOURce:]TRIGger:TTLT

Purpose	Selects the backplane trigger line	
Type	Setting	
Command Syntax	[SOURce:]TRIGger:TTLT <trg_ttlt>	
Command Parameters	<trg_ttlt> = 0 to 7	
***RST Value**	0	
Query Syntax	[SOURce:]TRIGger:TTLT?	
Query Parameters	N/A	
Query Response	Returns the set value of the <trg_ttlt> parameter	
Description	The Source Trigger TTLT command selects the backplane trigger line. The default setting is trigger line 0.	
Examples	**Command / Query**	**Response *(Description)***
	TRIG:TTLT 3	*(Selects the back plane trigger line 3)*
Related Commands	[SOURce:]TRIGger:GATE [SOURce:]TRIGger[:MODE] [SOURce:]TRIGger:POLarity [SOURce:]TRIGger:RATE	

[SOURce:]VOLTage[:LEVel][:AMPLitude]

Purpose	Sets the signal output peak-to-peak voltage
Type	Setting
Command Syntax	[SOURce:]VOLTage[:LEVel][:AMPLitude] <numeric_value>
Command Parameters	<numeric_value> = Ranges stated below Range = dc Voltage : -6 V to +6 V into 50 Ω -10 V to -10 V into 200 Ω Pulsed dc : -6 V_{PEAK} to +6 V_{PEAK} into 50 Ω Sine/Square: 10 V_{P-P} into 50 Ω All other waveforms: 10 V_{P-P} into 50 Ω
***RST Value**	1.0
Query Syntax	[SOURce:]VOLTage[;LEVel][:AMPLitude]?
Query Parameters	N/A
Query Response	Returns the set value of the <numeric_value> parameter for this command
Description	The Source Voltage Level Amplitude command sets the peak-to-peak amplitude value of the signal output voltage. The default setting is 1.0.

Examples	**Command / Query**	**Response *(Description)***
	VOLT 2.5	*(Sets the voltage amplitude to 2.5 V_{P-P})*
	VOLT?	2.5 *(Verifies that the voltage amplitude is set to V_{P-P})*
Related Commands	[SOURce:]VOLTage[:LEVel]:OFFSet	

[SOURce:]VOLTage[:LEVel]:OFFSet

Purpose	Sets the output voltage offset	
Type	Setting	
Command Syntax	[SOURce:]VOLTage[:LEVel]:OFFSet <numeric_value>	
Command Parameters	<numeric_value> = Ranges stated below: Range = dc Voltage: -3 V to +3 V, Offset + Amplitude cannot exceed ±6 V Pulsed dc: -3 V to +3 V, Offset + (Amplitude / 2) cannot exceed ±3.5 V Sine/Square: -3 V to +3 V, Offset + Amplitude cannot exceed 6 V All other waveforms: -3 V to +3 V, Offset + Amplitude cannot exceed 6 V	
***RST Value**	0.0	
Query Syntax	[SOURce:]VOLTage[:LEVel]:OFFSet?	
Query Parameters	N/A	
Query Response	Returns the set value of the <numeric_value> parameter for this command	
Description	The Source Voltage Level Offset command controls the level of the output offset voltage. The default setting is 0.0.	
Examples	**Command / Query**	**Response *(Description)***
	VOLT:OFFS 1 VOLT:OFFS?	*(Sets the level of output offset-voltage to 1)* 1 *(Verifies that the level of the output offset-voltage is 1)*
Related Commands	[SOURce:]VOLTage[:LEVel][:AMPLitude]	

TEST[:ALL]?

Purpose	Performs a non-destructive test of the hardware	
Type	Query	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	TEST[:ALL]?	
Query Parameters	N/A	
Query Response	Numeric value	
Description	The Test All query performs a non-destructive test of the hardware. A result of 0 indicates that the test has passed; a non-zero value indicates failure.	
Examples	**Command / Query**	**Response *(Description)***
	TEST?	0
Related Commands	TEST:RAM?	

TEST:RAM?

Purpose	Performs a destructive test of the Trace Memory	
Type	Query	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	TEST:RAM?	
Query Parameters	N/A	
Query Response	Numeric value	
Description	The Test Ram query runs a destructive self-test of the Trace Memory. A return of 0 indicates the test has passed.	
Examples	**Command / Query**	**Response *(Description)***
	TEST:RAM?	*(Runs Trace Memory self-test)*
Related Commands	TEST[:ALL]?	

REQUIRED SCPI COMMANDS

STATus:OPERation:CONDition?

Purpose	Queries the Operation Status Register's condition register	
Type	Required SCPI command	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	STATus:OPERation:CONDition?	
Query Parameters	N/A	
Query Response	0 | 32	
Description	Reports the bits set in the Operation Status Register's condition register. Returns a decimal number of 0 (no bits set) or 32 (bit 5 set).	
Examples	**Command / Query**	**Response *(Description)***
	STAT:OPER:COND?	0 *(Indicates that no bits are set in the Operation Status register)*
Related Commands	STATus:OPERation:ENABle STATus:OPERation[:EVENt]	

STATus:OPERation:ENABle

Purpose	Sets the Operation Status Register's enable register	
Type	Required SCPI command	
Command Syntax	STATus:OPERation:ENABle <NRf>	
Command Parameters	<NRf> = numeric ASCII value from 0 to 32767	
***RST Value**	<NRf> must be specified	
Query Syntax	STATus:OPERation:ENABle?	
Query Parameters	N/A	
Query Response	<NRf> = Numeric ASCII value from 0 to 32767	
Description	This command enables bits in the Operation Status Register's enable register to report to the summary bit; sets Status Bytes register bit 7 to true. The query reports the bits enabled in the Operation Status Register's enable register, then clears the register contents and enters the value into the computer.	
Examples	**Command / Query**	**Response *(Description)***
	STAT:OPER ENAB 33 STAT:OPER:ENAB?	*(Enables bit 0 and bit 5)* 33 *(Indicates that bit 0 and 5 are enabled)*
Related Commands	STATus:OPERation:CONDition? STATus:OPERation[:EVENt]	

STATus:OPERation[:EVENt]?

Purpose	Queries the Operation Status Register's event register	
Type	Required SCPI command	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	STATus:OPERation[:EVENt]?	
Query Parameters	N/A	
Query Response	0	
Description	Queries the bits set in the event register of the Operation Status Register. This command clears all bits in the event register.	
Examples	**Command / Query**	**Response *(Description)***
	STAT:OPER?	0
Related Commands	STATus:OPERation:CONDition? STATus:OPERation:ENABle?	

STATus:PRESet

Purpose	Presets the Status Registers	
Type	Required SCPI command	
Command Syntax	STATus:PRESet	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	N/A	
Query Parameters	N/A	
Query Response	N/A	
Description	The Status Preset command presets the Status Registers. The Operational Status Enable Register is set to 0 and the Questionable Status Enable Register is set to 0. This command is provided for SCPI compliance only.	
Examples	**Command / Query**	**Response *(Description)***
	STAT:PRES	
Related Commands	N/A	

STATus:QUEStionable:CONDition?

Purpose	Queries the Questionable Status Condition Register	
Type	Required SCPI command	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	STATus:QUEStionable:CONDition?	
Query Parameters	N/A	
Query Response	0	
Description	The Questionable Status Condition Register query is provided for SCPI compliance only. The VM3640A does not alter any bits in this register and a query always reports a 0.	
Examples	**Command / Query**	**Response *(Description)***
	STAT:QUES:COND?	0
Related Commands	N/A	

STATus:QUEStionable:ENABle

Purpose	Sets the Questionable Status Enable Register	
Type	Required SCPI command	
Command Syntax	STATus:QUEStionable:ENABle <NRf>	
Command Parameters	<NRf> = numeric ASCII value from 0 to 32767	
***RST Value**	<NRf> must be supplied	
Query Syntax	STATus:QUEStionable:ENABle?	
Query Parameters	N/A	
Query Response	<NRf> = Numeric ASCII value from 0 to 32767	
Description	The command sets the bits in the Questionable Data/Signal Register's enable register to be reported to the summary bit (sets Status Byte Register bit 3 to true). The Status Questionable Enable query reports the contents of the Questionable Data/Signal Register's enable register, then clears the register contents and enters the value into the computer	
Examples	**Command / Query**	**Response *(Description)***
	STAT:QUES:ENAB 64 STAT:QUES:ENAB?	 64
Related Commands	N/A	

STATus:QUEStionable[:EVENt]?

Purpose	Queries the Questionable Status Event Register	
Type	Required SCPI command	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	STATus:QUEStionable[:EVENt]?	
Query Parameters	N/A	
Query Response	Decimal number	
Description	The query reports the bits set in the event register of the Questionable Data/Signal register. This command reads the event register, then clears all bits in the event register and enters the value into the computer.	
Examples	**Command / Query**	**Response *(Description)***
	STAT:QUES?	0
Related Commands	N/A	

SYSTem:ERRor?

Purpose	Queries the Error Queue	
Type	Required SCPI command	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	Clears queue	
Query Syntax	SYSTem:ERRor?	
Query Parameters	N/A	
Query Response	ASCII string	
Description	The System Error query is used to retrieve error messages from the error queue. The error queue will maintain two error messages. If additional errors occur, the queue will overflow and the subsequent error messages will be lost. In the case of an overflow, an overflow message will replace the second error message. See the SCPI standard Volume 2: Command Reference for details on errors and reporting them.	
Examples	**Command / Query**	**Response *(Description)***
	SYST:ERR?	-350, “Queue overflow”
Related Commands	N/A	

SYSTem:VERSion?

Purpose	Queries the SCPI version number with which the VM3640A complies	
Type	Required SCPI command	
Command Syntax	N/A	
Command Parameters	N/A	
***RST Value**	N/A	
Query Syntax	SYSTem:VERSion?	
Query Parameters	N/A	
Query Response	Numeric ASCII value	
Description	The System Version query reports version of the SCPI standard with which the VM3640A complies.	
Examples	**Command / Query**	**Response *(Description)***
	SYST:VERS?	1994.0
Related Commands	N/A	

INDEX

Symbols

*CLS....24, 30
*ESE....24, 31
*ESR?....24, 32
*IDN?....24, 33
*OPC....24, 34
*RST....24, 35
*SRE....24, 36
*STB?....24, 37
*TRG....24, 38
*TST?....24, 39
*WAI....24, 40

A

AM[:STATe]....25, 41

B

backplane....18, 19
backplane jumpers....17, 18

C

CALibration:COUNt?....25, 42
CALibration:DC:LEVel....25, 53
CALibration:SECurity:CODE....25, 54
CALibration:SECurity:STATe....25, 55
CALibration:STATe....25, 56
CALibration:STORe....25, 57
CALibration[:DATA]....25
CALibration[:DATA]:AFCorrection....25, 43
CALibration[:DATA]:AMPLitude:OFFSet....25, 45
CALibration[:DATA]:AMPLitude[:GAIN]....25, 44
CALibration[:DATA]:DUTY....25, 46
CALibration[:DATA]:OFFSet:OFFSet....25, 48
CALibration[:DATA]:OFFSet[:GAIN]....25, 47
CALibration[:DATA]:SQHI....25, 49, 50
CALibration[:DATA]:SQUare:OFFSet:OFFSet....25, 51
CALibration[:DATA]:ZERO....25, 52
cooling....17

D

dynamic address configuration....18

F

FSK[:STATe]....25, 58, 59

K

keyword....19, 20

L

logical address....17, 18

M

message-based....14, 19

P

parameter....20
Port....21
power....17, 18, 23, 32
programming language....19
PSK[:STATe]....25, 60

R

RESet....25, 61

S

security code....54
SENSe:CORRection:IMPedance....25, 62
[SOURce:]BURSt:COUNt....25, 63
[SOURce:]BURSt:PHASe....25, 64
[SOURce:]BURSt:RATE....25, 65
[SOURce:]BURSt[:STATe]....25, 66
[SOURce:]CLOCk:CONFigure....25, 67
[SOURce:]FREQuency[:CW]....25, 68
[SOURce:]FUNCtion:DC:VOLTage....25, 69
[SOURce:]FUNCtion:PULSe:WIDTh....26, 70
[SOURce:]FUNCtion:SQUare[:DUTY]....26, 72
[SOURce:]FUNCtion:USER:DIRectory?....26, 73
[SOURce:]FUNCtion:USER:FREE?....26, 74
[SOURce:]FUNCtion:USER:WAVE....26, 76
[SOURce:]FUNCtion:USER[:NAME]....26, 75
[SOURce:]FUNCtion[:SHAPe]....26, 71
[SOURce:]MARKer:POLarity....26, 78
[SOURce:]MARKer:POSition....26, 77
[SOURce:]MARKer:SOURce....26, 79
[SOURce:]MARKer[:STATe]....26, 80
[SOURce:]OUTPut:FILTer[:LPASs]:TYPe....26, 82
[SOURce:]OUTPut:FILTer[:LPASs][:STATe]....26, 81
[SOURce:]OUTPut[:STATe]....26, 83
[SOURce:]ROSC:SOURce....26, 84
[SOURce:]SWEep:COUNt....26, 85
[SOURce:]SWEep:DIRection....26, 86
[SOURce:]SWEep:MODE....26, 87
[SOURce:]SWEep:POINts....26, 88
[SOURce:]SWEep:SPACing....26, 89
[SOURce:]SWEep:STARt....26, 90
[SOURce:]SWEep:STOP....26, 91
[SOURce:]SWEep:TIME....26, 93
[SOURce:]SWEep[:STATe]....26, 92
[SOURce:]SYNC:MODE....26, 94
[SOURce:]SYNC[:STATe]....26, 95
[SOURce:]TRIGger:GATE....26, 96
[SOURce:]TRIGger:POLarity....26, 98
[SOURce:]TRIGger:RATE....26, 99
[SOURce:]TRIGger:TTLT....26, 100
[SOURce:]TRIGger[:MODE]....26, 97
[SOURce:]VOLTage[:LEVel]:OFFSet....26, 102
[SOURce:]VOLTage[:LEVel][:AMPLitude]....26, 101
STATus:OPERation:CONDition?....27, 105
STATus:OPERation:ENABle....27, 106

STATus:OPERation[:EVENt]? 27, 107
STATus:PRESet 27, 108
STATus:QUEStionable:CONDition? 27, 109
STATus:QUEStionable:ENABle.....27, 110
STATus:QUEStionable[:EVENt]? 27, 111
syntax.....19, 23
SYSTem:ERRor?.....27, 112
SYSTem:VERSion? 27, 113

T

TEST:RAM?.....26, 104
TEST[:ALL]? 26, 103
tree-structured language.....19

U

user-defined waveform 22, 26, 75, 76

V

VMIP 13, 18
VXIbus 14, 17, 19

W

word serial protocol 14

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

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

ARTISAN® TECHNOLOGY GROUP