Arbitrary Waveform Generators

Model 665 – 50 MHz Single Channel
Model 667 – 50 MHz Dual Channel
Model 685 – 80 MHz Single Channel
Model 687 – 80 MHz Dual Channel

OR-X
P.O. Box 2116,
Rehovot, Israel 76121
Tel: (972)-89477771
Fax:(972)-89477772
www.or-x.net

P/N 066-5501-01

Warranty

OR-X warrants only to the original purchaser that this product, as purchased from OR-X or an OR-X distributor or dealer, will conform to the written specifications for a period of one year from the date of purchase. If the product fails to conform to these warranties, OR-X, as its sole and exclusive liability hereunder, will repair or replace the product and/or its components within a reasonable period of time if the product is returned to OR-X's facility, within the warranty period as expressed above. These warranties are made upon the express condition that:

a. The purchaser promptly notify OR-X in writing of any non-conformity with the above warranty including a detailed explanation of the alleged deficiencies.

b. The product is returned to OR-X at the buyer's expense only after obtaining the proper authorization from OR-X.

c. When the product is returned for repair, a copy of the original bill of sale or invoice is sent with the product .

d. OR-X will not be liable for any incidental or consequential damages.

e. In the opinion of OR-X upon inspection, the product has not been misused, altered, or damaged due to abnormal handling and/or operation.

f. Repairs to the product and/or its components have not been made by anyone other than OR-X or one of its authorized repair agents.

g. The product has not been modified, altered, or changed in any manner by anyone other than OR-X or one of its authorized repair agents.

THIS WARRANTY EXCLUDES ALL OTHER WARRANTIES, WHETHER EXPRESSED OR IMPLIED, ORAL OR WRITTEN, INCLUDING WITHOUT LIMITATION WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR A PARTICULAR PURPOSE.

No term, condition, understanding or agreement purporting to modify the terms of this warranty shall have any legal effect unless made in writing and signed by an authorized officer of OR-X and the purchaser.

Safety Summary

The following safety precautions apply to both operating and maintenance personnel and must be observed during all phases of operation, service, and repair of this instrument. Before applying power, follow the installation instructions and become familiar with the operating instructions for this instrument.

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 instrument. OR-X assumes no liability for a customer's failure to comply with these requirements. This is a Safety Class I instrument.

GROUND THE INSTRUMENT

To minimize shock hazard, the instrument chassis and cabinet must be connected to an electrical ground. This instrument is grounded through the ground conductor of the supplied, three-conductor ac power cable. The power cable must be plugged into an approved three-conductor electrical outlet. Do not alter the ground connection. Without the protective ground connection, all accessible conductive parts (including control knobs) can render an electric shock. The power jack and mating plug of the power cable meet IEC safety standards.

DO NOT OPERATE IN AN EXPLOSIVE ATMOSPHERE

Do not operate the instrument in the presence of flammable gases or fumes. Operation of any electrical instrument in such an environment constitutes a definite safety hazard.

KEEP AWAY FROM LIVE CIRCUITS

Instrument covers must not be removed by operating personnel. Component replacement and internal adjustments must be made by qualified maintenance personnel. Disconnect the power cord before removing the instrument covers and replacing components. Under certain conditions, even with the power cable removed, dangerous voltages may exist. To avoid injuries, always disconnect power and discharge circuits before touching them.

DO NOT SERVICE OR ADJUST ALONE

Do not attempt any internal service or adjustment unless another person, capable of rendering first aid and resuscitation, is present.

DO NOT SUBSTITUTE PARTS OR MODIFY THE INSTRUMENT

Do not install substitute parts or perform any unauthorized modifications to this instrument. Return the instrument to OR-X for service and repair to ensure that safety features are maintained.

WARNINGS AND CAUTIONS

WARNING and ***CAUTION*** statements, such as the following examples, denote a hazard and appear throughout this manual. Follow all instructions contained in these statements.

A ***WARNING*** statement calls attention to an operating procedure, practice, or condition, which, if not followed correctly, could result in injury or death to personnel.

A ***CAUTION*** statement calls attention to an operating procedure, practice, or condition, which, if not followed correctly, could result in damage to or destruction of part or all of the product.

WARNING: ***Do not alter the ground connection. Without the protective ground connection, all accessible conductive parts (including control knobs) can render an electric shock. The power jack and mating plug of the power cable meet IEC safety standards.***

WARNING: ***To avoid electrical shock hazard, disconnect power cord before removing covers. Refer servicing to qualified personnel.***

CAUTION: ***Before connecting the line cord to the AC mains, check the rear panel AC line voltage indicator. Applying a line voltage other than the indicated voltage can destroy the AC line fuses. For continued fire protection, replace fuses only with those of the specified voltage and current ratings.***

CAUTION: ***This product uses components which can be damaged by electro-static discharge (ESD). To avoid damage, be sure to follow proper procedures for handling, storing and transporting parts and subassemblies which contain ESD-sensitive components.***

Contents

Section 1

Introduction

1.1 Introduction

This manual contains information required to operate, program and test the Arbitrary Waveform Generator series, including the following models:

- Model 665 - 50 MHz single channel
- Model 667 - 50 MHz dual channel
- Model 685 - 80 MHz single channel
- Model 687 - 80 MHz dual channel

This section covers the instrument general description, instrument specifications and characteristics.

1.2 Description

The MODEL 687 is a versatile high performance arbitrary waveform generator. Arbitrary waveforms can be programmed and generated with 14 bit resolution and up to 16,777,216 points length. Waveforms can be output in continuous, triggered, gated or burst mode with an internal or external reference. AM, FM and FSK modulation combined with versatile Sweep capabilities make the unit suitable for a wide range of applications.
Editing is flexible and easy including auto increment, line draw and predefined waveform facilities. The instrument can be remotely operated via the GPIB (IEEE-488.2) or USB interface bus and it is SCPI compatible.
The unit has two identical channels that can be separately operated on combined for two channel operation.

1.3 Memory Architecture

The waveform generator has two types of memory:

Waveform memory
Execution memory

The waveform memory consists of a flash circuit capable to store 6 full waveforms of 16M points per channel. The user can edit arbitrary waveforms in waveform memory and can specify any data value in the range from -8191 to 8191 for any point in waveform memory.

The following operations can be performed in the waveform memory:

- Insert and scale any of the following predefined waveforms: sine, triangle, square, ramp up, ramp down, noise
- Draw a line between any two points
- Clear (set to zero) any set of points or all points
- Protect (prevent from changing) any set of points or all points
- Copy any set of points to another area
- Set individual point values

Up to 16M continuous points of waveform memory can be executed by specifying a starting address in waveform memory and length.
After specifying a section of waveform memory for execution, the following parameters can be set:

- Point rate (frequency)
- Peak-to-peak amplitude
- Offset voltage

SPECIFICATIONS

GENERAL DESCRIPTION

The MODEL 687 is a programmable dual channel arbitrary waveform generator that can generate arbitrary waveforms and predefined signal as: Sine, Triangle, Square, Pulse, etc. The unit is available with a single channel as MODEL 685. The MODELS 665 and 667 are compatible units with lower frequency range and smaller memory.
The specifications are identical for both channels. The channels can be operated separately or combined with a phase relation. Each channel specs are:

FREQUENCY CHARACTERISTICS (STANDARD WAVEFORMS)

Sine	-	1 uHz to 80 MHz (50 MHz for 665/667)
Square	-	1 uHz to 80 MHz (50 MHz for 665/667)
Triangle	-	1 uHz to 5 MHz, variable symmetry 1%-99% to 500 KHz, 19%-90% to 2 MHZ
Pulse	-	0.5 mHz to 25 MHz
Accuracy	-	0.001 % (10 ppm)
Resolution	-	12 digits or 1uHz

ARBITRARY CHARACTERISTICS

Waveform length	-	2 points to 16,777,216 points (4,194,304 points for 665/667)
Vertical resolution	-	14 bits
Sampling rate	-	5 ns to 100 s
Accuracy	-	0.001 % (10 ppm)
Resolution	-	4 digits or 1 ps

OUTPUT CHARACTERISTICS

Amplitude Range	-	10mV to 10Vp-p into 50 ohms
Resolution	-	4 digits (9999 counts)
Amplitude Accuracy	-	± 1% ± 20mV of the programmed output from 1V- 10V ± 1% ± 1 mV of the programmed output from 500 mV- 999 mV
Flatness	-	± 0.1 dB at 10MHz
	-	± 1 dB at 50 MHz
Offset Range	-	± 4.99V into 50 ohms, depending on the Amplitude setting
Offset Resolution	-	10 mV with 4 digits resolution
Offset Accuracy	-	± 1% ± 10mV into 50 ohms
Output Impedance	-	50 ohms
Output Protection	-	The instrument output is protected against short circuit or accidental voltage practically available in electronic laboratories, applied to the main output connector
Filters	-	9 pole Elliptic and 5 pole Bessel filters

WAVEFORM CHARACTERISTICS

Harmonic Distortion	-	0-100 KHz	-65 dBc
	-	100 KHz-5 MHz	-45 dBc
	-	5 MHz-50 MHz	-35 dBc
Spurious	-	DC-1MHz	<-65 dBc
Square Rise/Fall Time	-	< 5 ns (10% to 90%) at full amplitude into 50 ohms	
Square Variable Duty Cycle	-	20% to 80% to 10 MHz, 40% to 60% up to 30MHz	

Symmetry at 50%	-	< 0.5 % to 1 MHz
Aberrations	-	< 5 % of p-p amplitude ± 50 mV.

OPERATING MODES

Continuous	-	Output continuous at programmed parameters.
Triggered	-	Output quiescent until triggered by an internal or external trigger, then one waveform cycle is generated to programmed parameters. Up to 20MHz trig rate for ARB waveforms and 10 MHz in DDS mode.
Gate	-	Same as triggered mode, except waveform is executed for the duration of the gate signal. The last cycle started is completed.
Burst	-	1- 999,999 cycles
Phase	-	-360° to +360°, 0.1° resolution
Trigger Source	-	Trigger source may be internal, external or manual. Internal trigger rate 0.01Hz-1MHz

MODULATION CHARACTERISTICS

Amplitude Modulation

- Internal:	0.01Hz-20KHz sine , square or triangle waveform Variable modulation from 0% to 100%.
- External:	5 Vp-p for 100% modulation, 10 Kohms input impedance.

Frequency Modulation

- Internal:	0.01Hz-20KHz sine wave, square or triangle
- External:	5 Vp-p for 100% deviation, 10 Kohms input impedance.

FSK

- Internal rate	0.01Hz-1MHz
- External	1MHz max

PWM to 20 KHz with programmable width and modulation rate

SWEEP CHARACTERISTICS

Sweep Shape:	Linear and Logarithmic, up or down
Sweep Time:	10 ms to 500 s.
Sweep trigger	internal, external, continuous or burst

VARIABLE PHASE

Range	+360° to -360°
Resolution	0.1°

Dual channel synchronization capability

INPUTS AND OUTPUTS

Trigger In	-	TTL compatible.
	-	Max. rate 20MHz.
	-	Minimum width 20ns.
	-	Input impedance 10 Kohms nominal.
Sync Out	-	TTL pulse at programmed frequency, 50 ohms source impedance.
Modulation In	-	5 Vp-p for 100% modulation .

	- 10 KΩ input impedance. - DC to >50 KHz minimum bandwidth.
Reference In	- 10 MHz, input , for external unit synchronization. 50 ohms output impedance and 10 Kohm input.
Reference Out	- 10 MHz square wave, TTL compatible
Marker Out	- a positive TTL pulse user programmable in Arbitrary waveform, 50 ohm source impedance. source impedance.

GENERAL

Store memory	50 full panel settings at power-off
Arbitrary memory	8 waveform of up to 16 M points for each hannel
Dimensions	- 8.4 inch (213 mm) wide - 3.5 inch (88 mm) high - 10.8 inch (275 mm) deep
Weight	- 2.5 Kg.
Power	- 100V-240V ± 10% - < 50 VA max
Temperature	
Operating	- 0ºC to +50ºC
Non-operating	- -20ºC to +70ºC
Humidity	- 90 % RH , 0ºC to 30ºC
EMC	- According to EN55011 class B for radiated and conducted emissions.
Electrical Discharge Immunity	- According to EN55082
Safety Specifications	- According to EN61010

CE Labeled

NOTE

Specifications are verified according to the performance check procedures in the technical manual.
Specifications not verified in the manual are either explanatory notes or general performance characteristics only.

Section 2

Installation

2.1 Introduction

This section contains installation information, power requirements, initial inspection and signal connections for all Arbitrary Waveform Generator in this series..

2.2 Mechanical Inspection

This instrument was carefully inspected before shipment. Upon receipt inspect the instrument for damage that might have occurred in transit. If there is damage due to shipping, file a claim with the carrier who transported the unit. The shipping and packing material should be saved if reshipment is required. If the original container is not to be used, then use a heavy carton box. Wrap the unit with plastic and place cardboard strips across the face for protection. Use packing material around all sides of the container and seal it with tape bands. Mark the box "FRAGILE".

2.3 Initial Inspection

After the mechanical inspection, verify the contents of the shipment (accessories and installed options). If the contents are incomplete, or if the instrument does not pass the specification acceptance tests, notify the local OR-X service center.

2.4 Instrument Mounting

The Arbitrary Waveform Generators are intended for bench use. The instruments include a front feet tilt mechanism for optimum panel viewing angle. The instrument does not require special cooling when operated within conventional temperature limits. The unit can be installed in a closed rack or test station if proper air flow is assured for removing about 30 W of power dissipation.

2.5 Power Requirements

The units can be operated from any source of 90V to 264V AC, frequency from 48Hz to 66Hz. The maximum power consumption is 50 VA. Use a slow blow fuse UL/CSA approved of 1A as indicated on the rear panel of the instrument.

WARNING

THE LINE POWER VOLTAGE OF THE INSTRUMENT IS NOTED ON THE AC INPUT PLUG. TO PREVENT DAMAGE TO THE INSTRUMENT, CHECK FOR PROPER MATCH OF LINE VOLTAGE AND PROPER FUSE TYPE AND RATING.

The instrument power fuse is located in the AC input plug. To access the fuse, first disconnect the power cord and then remove the fuse cartridge.

2.6 Grounding Requirements

For the safety of operating personnel, the instrument must be grounded. The central pin on the AC plug grounds the instrument when properly connected to the ground wire and plugged into proper receptacle.

WARNING

TO AVOID PERSONAL INJURY DUE TO SHOCK, THE THIRD WIRE EARTH GROUND MUST BE CONTINUOUS TO THE POWER OUTLET. BEFORE CONNECTION TO THE POWER OUTLET, EXAMINE ALL CABLES AND CONNECTIONS BETWEEN THE UNIT AND THE FACILITY POWER FOR A CONTINUOUS EARTH GROUND PATH.
THE POWER CABLE MUST MEET IEC SAFETY STANDARDS.

2.7 Signal Connections

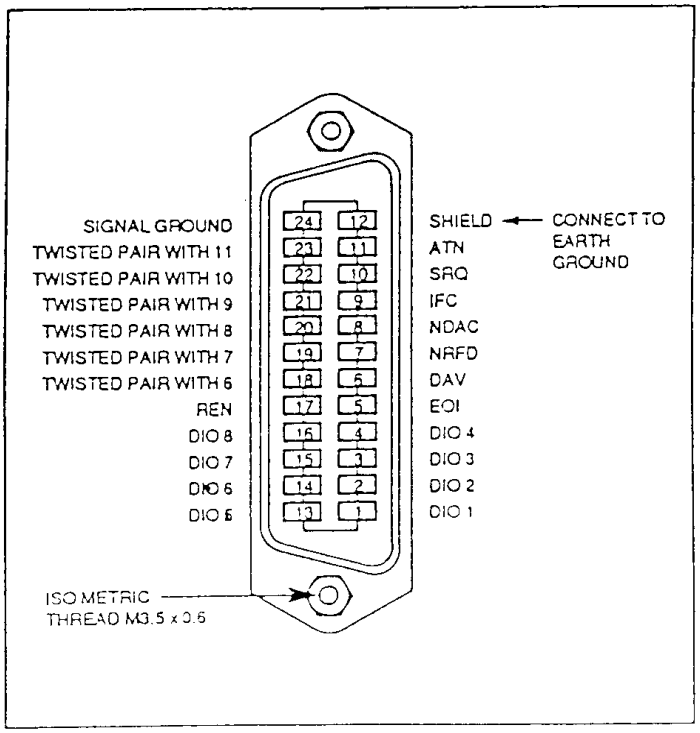
Use RG58U 50 Ohm or equivalent coaxial cables for all input and output signals to and from the instrument.

2.8 GPIB Address

The instrument optional GPIB interface is shipped with the address set to decimal 9. The address can be changed from the front panel by using the "UTILITY" menu (refer to “Special Functions” on the OPERATING INSTRUCTIONS section of this manual).

2.9 GPIB Connections

The rear panel GPIB connector is a AMPHENOL 57-10240 or equivalent, and connects to a standard IEEE-488 bus cable connector. The GPIB line screens are not isolated from chassis and signal ground.



24-Pin Micro-Ribbon (Series 57) Connector

2.10 USB Connection

On the rear panel of the unit a USB type B connector is provided as device interface.

2.11 USB Configuration

Before using the USB, the driver provided with the unit CD must be properly installed.
Follow the install instructions from on the CD.

2.12 LAN Configuration

TBD

Section 3

Operating Instructions

3.1 General Description

This section describes the displays, controls and connectors of the MODEL 687 Dual Chanel Arbitrary Waveform Generator.
All controls for the instrument local operation are located on the front panel. The connectors are located on both front and rear panels. The **CHAN** push button selects the active channel for displaying and changing the operating parameters. The active channel is indicated on the right upper corner of the LCD display.
On single channel units Model 665 and 685 the **CHAN** push button is not available.

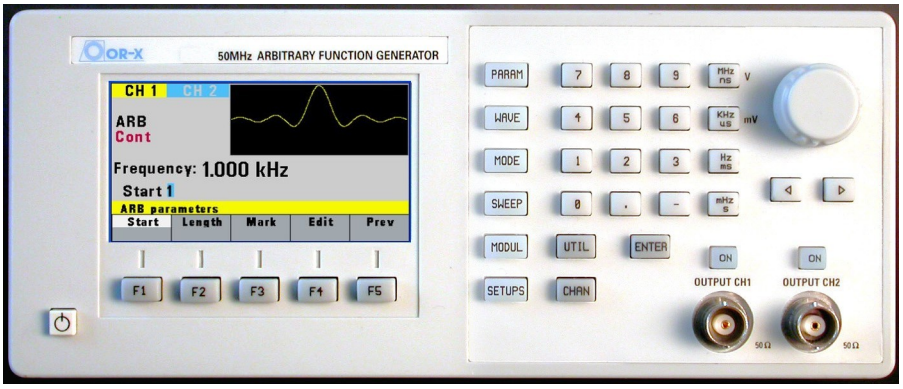


Figure 3.1 - MODEL 667 Front Panel

1. Power ON-OFF -Applies and removes AC power to the unit.

2. Display Window -Displays all instrument data and settings on a LCD.

3. FI-F5 Keys -Select the menu options that appear on the second line of the LCD display. Menus differ depending on the selected parameter, function or mode.

4. MENU Keys -Select parameters, functions or modes whose settings are to be displayed or changed.

5. Rotary Knob -Used to increment/decrement numerical values or to scan through the possible selections.

6. Modify Keys -Used to move the cursor (when visible) to either left or right.

7. Output ON -Controls the main output signal. The output status is displayed on the LCD.

3.2 Display Window

The MODEL 687 has a color graphic LCD display that can display up to 400 x 240 dots. When you power-on the unit a parameter (Frequency) and its current settings appear in the display. The bottom displays a menu that corresponds to the function, parameter or mode displayed selected.

3.3 Front Panel Controls

The front-panel controls select, display, and change parameter, function, and mode settings. They also include the keys you use to program and generate arbitrary waveform output.
Use the rotary input knob and the cursor movement keys to enter data into the waveform generator.

To change a setting:
1. Press the key that leads to a required item.
2. Move cursor using cursor keys to the appropriate position in the numeric field (if applicable).
3. Use the rotary input or the numerical keyboard to change the value of the displayed item. Changes take effect immediately.

The following subsections describe the function of each front panel key and connector.

3.4 Connectors

The function generator has two BNC connectors on the front panel and ten on the rear panel where you can connect coaxial cables. These coaxial cables serve as carrier lines for input and output signals delivered to and from the function generator. Two sets of connectors, one set for each channel are available. The only common connector is the REF IN/OUT.
The rear panel connectors are arranged in two rows: the lower row is for channel 1 and the upper row for channel 2. The REF IN and REF OUT connectors are common to both channels.

Output Connector
Two connectors provided, one for each channel. Use this connector to transfer the main output signal from the function generator.

TRIG IN Connector
Use this connector to apply an external trigger or gate signal, depending on the waveform generator setting, to the generator.

SYNC OUT Connector
Use this connector to output a positive TTL sync pulse generated at each waveform cycle.

MODULATION IN Connector
5V p-p signal for 100% modulation, 10Kohms input impedance with DC - >50KHz bandwidth.

REF IN and REF OUT Connectors
Use this connector to input a 10MHz TTL signal to be used as an reference clock for the unit signal generation.
A 10 MHz TTL level signal from the unit internal clock is available at the REF OUT for synchronization of external units.
The connectors are common to both channels and the reference source is selected for each channel from the main menu.

MARKER OUT

Use this connector to output a positive TTL pulse in Arbitrary waveform. The Marker position and width can be programmed at any desired Arbitrary location

3.5 *Output Connections*

The waveform generator output circuits operate as a 50 ohms voltage source working into a 50 ohms load. At higher frequencies, un-terminated or improperly terminated output cause aberrations on the output waveform. In addition, loads less than 50 ohms reduce the waveform amplitude, while loads more than 50 ohms increase waveform amplitude.

Excessive distortion or aberrations caused by improper termination are less noticeable at lower frequencies, especially with sine and triangle waveforms. To ensure waveform integrity, follow these precautions:

1. Use good quality 50 ohms coaxial cable and connectors.
2. Make all connections tight and as short as possible.
3. Use good quality attenuators if it is necessary to reduce waveform amplitudes applied to sensitive circuits.
4. Use termination or impedance-matching devices to avoid reflections.
5. Ensure that attenuators and terminations have adequate power handling capabilities.

If there is a DC voltage across the output load, use a coupling capacitor in series with the load. The time constant of the coupling capacitor and load must be long enough to maintain pulse flatness.

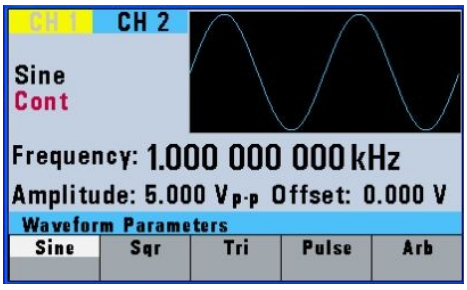
Impedance Matching

If the waveform generator is driving a high impedance, such as the 1 Mohm input impedance (paralleled by a stated capacitance) of an oscilloscope vertical input, connect the transmission line to a 50 ohms attenuator, a 50 ohms termination and to the oscilloscope input. The attenuator isolates the input capacitance of the device and terminates the waveform generator properly.

3.6 MENU Keys

These keys select the main menus for displaying or changing a parameter, function or mode.

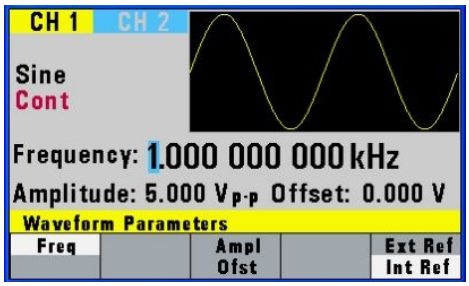
The **CHAN** is changing the operating channel from **CH1** to **CH2**. The active channel is displayed on the screen top side and the Menu Bar is changing the color as per selected channel, yellow for CH1 and blue for CH2.



CH2 Display

3.6.1 PARAMETER Key

This key selects and displays the waveform frequency, amplitude, offset and external reference and allows changing the parameter data.
When the Arbitrary Waveform is selected, the display shows also the waveform rate.



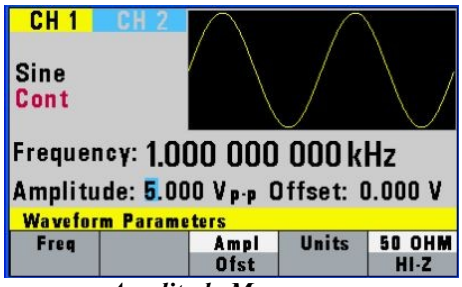
Frequency Menu

F1: Freq - (Frequency) Selects and displays the frequency. Change the frequency setting using the shift keys, rotary knob or numerical keys. If a certain wavelength can't produce the waveform at the desired frequency, the waveform generator displays an error message.

F1: Freq/Rate - Selects and displays the Point Rate (for Arbitrary Waveform only). The Rate parameter governs the rate at which waveform points are executed, and thus the frequency of the waveform output. When you set this parameter, the waveform generator will keep that execution rate for all waveform lengths until it is changed.

F5: Int Ref - Selects internal or external reference source (the external reference must be connected to the rear panel Ref In connector).

F3: Ampl/Ofst - Selects the Amplitude and Offset display and changing of parameters.



Amplitude Menu

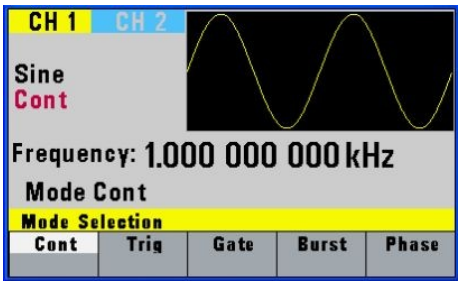
In Arbitrary mode this setting defines the maximum peak-to-peak amplitude of a full-scale waveform. If the waveform does not use the full scale of data (-8091 to +8091), then its actual amplitude will be smaller.

F4: Units - Selects the amplitude units: peak-to-peak, RMS or dBm (the dBm is for sine waves only)

F5: 50 Ohm/ HI-Z - Selects the display value for high impedance connection to an external device or 50 ohms termination

3.6.2 MODE Key

Selects the output mode: CONT (Continuous), TRIG (Triggered), GATE (Gated), BRST (Burst) and PHASE. To select the output mode, press MODE, then press the function *key* that corresponds to the desired Mode menu option, as shown:



Mode Menu

F1: Cont - (Continuous) - Selects continuous output.

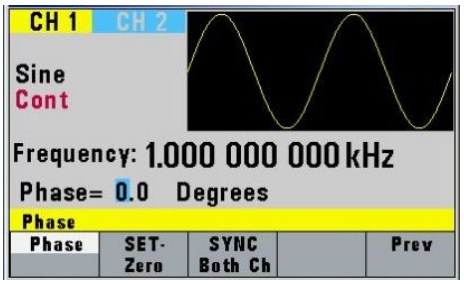
F2: Trig - (Triggered) - Triggers one output cycle of the selected waveform for each trigger event.

F3: Gate - Triggers output cycles as long as the trigger source asserts the gate signal.

F4: Burst - Triggers output N output cycles for each trigger event, where N ranges from 2 to 999999.

F5: Phase - Selects the start phase of the signal in non-continuous modes. The range is from -360˚ to +360˚, with a 0.1˚ resolution**.**

When Phase Menu is selected the following screen is displayed:

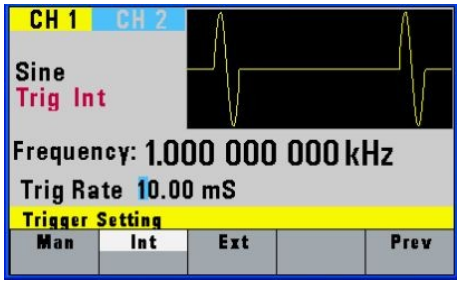


Phase Menu

F2: Set-Zero - Sets the phase reference to zero..

F3: SYNC Both Ch. - For Dual Channel models, pushing this key synchronized both channels in between. with a phase shift as in the Phase parameter entered (or 0 after pushing SET-Zero)

After selecting the TRIG , GATE or BURST menu, the trigger source menu is available:



Trigger Menu

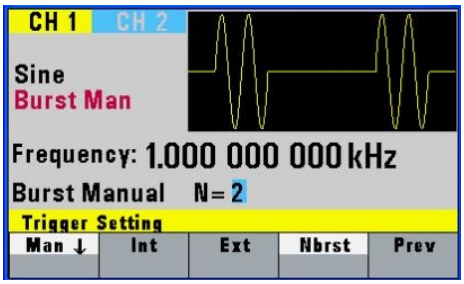
F1: Man - Selects manual as the trigger source. To trigger the waveform generator, press this MAN TRIG again.

F2: Int - (Internal) Selects the internal trigger generator as the trigger source. Change the internal trigger rate displayed with the rotary input knob.

F3: Ext - (External) Selects the external trigger signal as the trigger source. The trigger source is supplied through the TRIG IN connector.

F5: Prev - (Previous) Returns to the previous Menu selection.

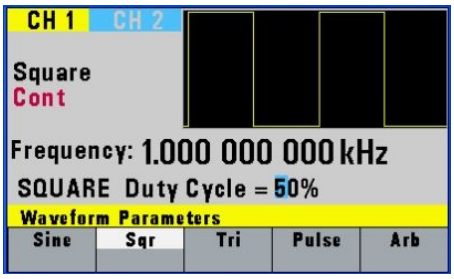
In BURST mode, the **F4** displays **NBRST**, the number of burst pulses to be output with each trigger. The N can be changed from 1 to 999,999.



Burst Menu

3.6.3 WAVEFORM Key

Displays the waveforms available:



Waveform Menu

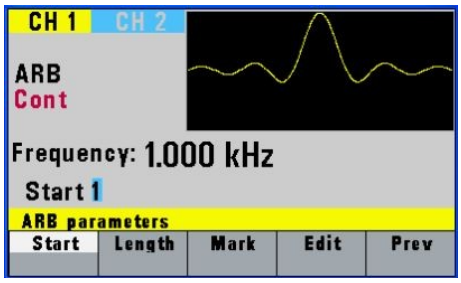
F1: Sine - Selects the sine waveform.

F2: Sqr - (Square) Selects the square waveform and displays the waveform duty cycle that can be changed from 20% to 80% up to 10MHz and 40% to 60% to 30MHz.

F3: Tri - (Triangle) Selects the triangle waveform and displays the waveform duty cycle that can be changed from 1% to 9%. The triangle maximum frequency is 5MHz.

F4: Pulse - Selects the Pulse waveform and then displays the pulse menu

F5: ARB - Selects the arbitrary waveform and then displays the Arbitrary menu:

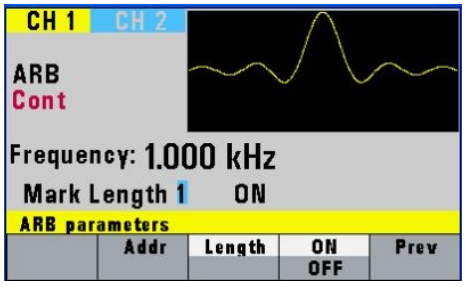


Arbitrary Menu

F1: Start - Selects the arbitrary waveform start address.

F2: Length - Selects the arbitrary waveform length. Use the Start and Length keys to mark a selection of the waveform memory that will be executed.

F3: Mark - (Marker Output) Selects the marker output address of the signal to be available at the Marker Out connector. The **F2:Addr, F3:Length** or **F4:ON/OFF** can be selected and the Marker output signal can be available at any desired location address between the start and stop addresses of the waveform executed.



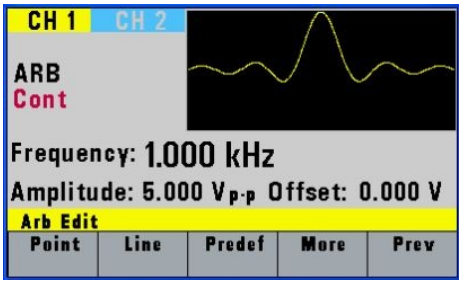
Marker Menu

Changing one of the arbitrary parameters as start, length and marker cause an updating of the output waveform to the new parameters.

3.6.4 Arbitrary EDIT Menu

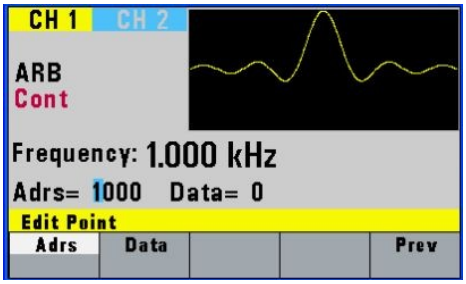
The Arbitrary Edit Menu enters data for creating arbitrary waveforms. You can enter data one point at a time, as a value at an address, draw a line from one point (a value at an address) to another point, create a predefined waveform, or combine these to create complex waveforms. The valid data values range is -8091 to 8091. The valid waveform memory addresses range from 1 to 16,777,216 per channel (to 4,194,304 for the Models 665 and 685) The data value governs the output amplitude of that point of the waveform, scaled to the instrument output amplitude. Therefore, a value of 8191 corresponds to positive peak amplitude, 0 corresponds to the waveform

offset, and -8191 corresponds to the negative peak amplitude.



Edit Menu

F1: Point - This menu allows the point by point waveform editing. When selected, the following menu is displayed:



Point Editing Menu

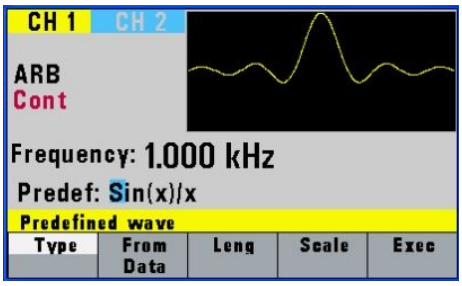
F1: Adrs - Select the current address in the arbitrary waveform memory.

F2: Data - Selects the data point value at the current address. You can change the point value from -8191 to 8191.

F2: Line - This menu allows a line drawing between two selected points.
Displays the following menu:

F1: From - Selects the starting point address.
F2: To - Selects the ending point address.
F4: Exec - Displays the Confirmation menu, **F1:NO** and **F3:YES**

F3: Predef - (Predefined Waveforms) Selects one of the predefined waveforms: Sine, Triangle, Square, Noise and ADD Noise (added to the existing data).
Displays the predefined waveforms menu:



Predefine Waveform Menu

F1: Type - Selects the waveform Sine, Triangle, Square, Sinx/x, Ramp, Exponential, Gaus or Noise. If Noise function is selected, a submenu is displayed to allow adding the noise to an available waveform or to generate it as a new noise waveform.

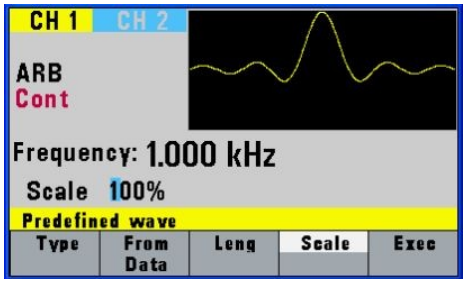
F2: From/Data - Selects the starting point of the generated waveform and the data value.

F3: Leng - Selects the length of the predefined waveform (number of points for a full wave). Different waveforms have different limitations on the length, as shown:

Table 3-1: Waveform Length Limits for Predefined Waveforms

Wave	Minimum length	Divisible by
Sine	16	4
Triangle	16	4
Square	2	2
Noise	16	1

F4: Scale - Selects the scale factor of the waveform. 100% means that the waveform spans the full scale of -8191 to 8191. Scale factors are limited by the point data value of the starting point and automatically calculated by the unit.



Scale Menu

F5: Exec - Prompts you to confirm whether to execute the selected predefined waveform. Press **NO** to abort executing the predefined waveform; press **YES** to execute the predefined waveform. On the NOISE function a menu of **ADD** and **NEW** is prompt to select a new noise waveform or to add noise to the existing waveform.

F4: More - Displays more editing functions, as shown:

F1: Copy - Displays the Copy menu (see the Copy Function later in this section).

F2: Clear - Displays the Clear menu (see the Clear Function later in this section).

F3: Prot - Displays the Protect menu (see the Protect Function later in this section).

F4:Show Wave - Display the Arbitrary waveform on the full LCD display. By pressing any button, the display returns to the MENU selection.

Copy Function. Copies an area of waveform memory to another area of waveform memory.

F1: From - Selects the address of the first point to copy.

F2: Leng - Selects the length (number of points) of the waveform to copy.

F3: To - Selects the destination address where the first point is copied.

F4: Exec - Prompts you to confirm whether to copy. Press **NO** to abort copying, **YES** to copy.

Clear Function. Clears (sets the data values to zero) either a section of or all of waveform memory.

F1: From - Selects the address of the first point to clear.

F2: To - Selects the address of the last point to clear.

F3: All - Clears the whole waveform memory. Equivalent to selecting from 1 to 4,000,000.

F4: Exec - Prompts you to confirm whether to clear. Press **NO** to abort clearing, **YES** to clear.

Protect Function. Protects (makes read-only) a section of waveform memory.

F1: From - Selects the address of the first point to protect.

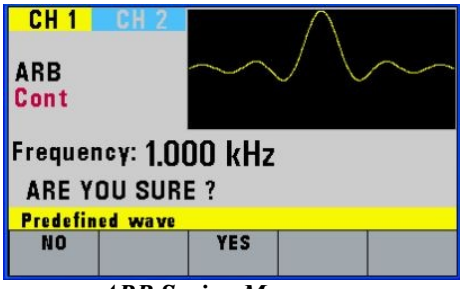
F2: To - Selects the address of the last point to protect.

F3: All - Protects the whole waveform memory. Equivalent to selecting from 1 to 16,000,000.

NOTE: *You can protect only one segment of waveform memory at a time.*

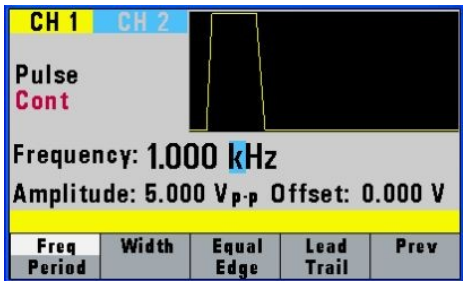
F4: ON/OFF - Selects the unprotect and resets memory protection so that the whole waveform memory can be written to.

NOTE: When exiting the ARB menu to select other waveform and if changes were made to the arbitrary waveform, the **NO-YES** menu is displayed for saving the updated arbitrary waveform. The selected waveform is of the length defined by the Start and Length data on the ARB menu.



ARB Saving Menu

3.6.5 *Pulse Menu*



Pulse Menu

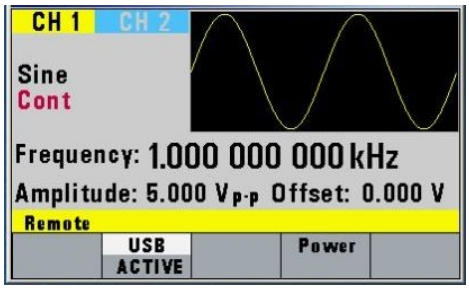
F1: Freq/Period - Selects the parameter definition of the Pulse repetition period.

F2: Width - Selects the Width of the generated pulse.

F3: Equal Edge - Selects equal Rise (Leading edge) and Fall (Trailing edge) times of the Pulse.

F4: Lead-Trail - Selects different Rise and Fall times of the Pulse.

3.6.6 *UTILITY Key*



Utility Menu

F1: Gpib **The F1 selection is available only when the GPIB option is installed**

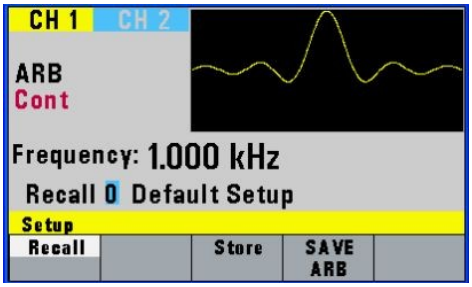
-Selects the GPIB remote mode of operation. After selection the GPIB can be set to any value from 1 to 31 using the rotary knob. The value is kept in a nonvolatile memory and used at power-on. The factory default address is 9. Setting the address to 31 puts the device in the off-bus state (it will not respond to messages on the GPIB bus).

F2: USB -Selects the USB. If selected, the **ACTIVE** message is displayed.

F4: Power - (Power-on default) Selects the power-on default setting. Select a value using the data keys or the rotary input knob. The selection is effective after a few seconds time-out period. Select zero (0) to have the waveform generator power on with the factory default settings. Select 50 to have the waveform generator power-on with the settings it had at the last power-off. Select any other value in the range from 1 to 49 to have the waveform generator power-on with the settings that you have saved with SETUPS STORE in the range 1 to 49.

3.6.7 SETUPS Key

The waveform generator can store the current front-panel settings, called a setup, into one of 50 storage buffers. When you recall a setup, the waveform generator restores the front-panel settings to those that you stored in the selected buffer. All waveform data, except for the waveform memory data, is stored in the setup.

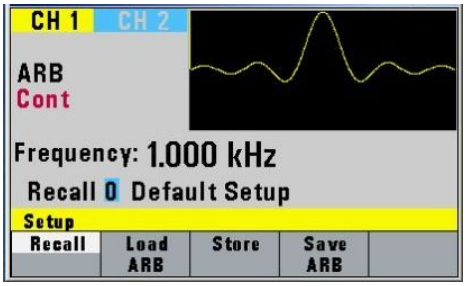


SETUP Menu

F1: Recall - Recalls a previously stored front-panel setup from the selected buffer. Change the buffer number by using the rotary input knob. Valid storage buffer numbers are from 1 to 49. Buffer 0 is the factory default setup; buffer 50 is the last front panel setup before power-off.

F3: Store - Stores the current front-panel setup to the specified storage buffer. Change the buffer number by using the data keys or the rotary input knob. Valid storage buffer numbers range from I to 49.

When the ARB waveform is selected, the set-up menu is:

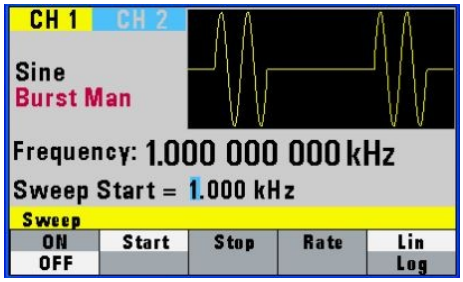


Arbitrary SETUP Menu

This menu allows the Save and Load of arbitrary waveforms. 8 waveforms of full length (16,777,216) can be saved for each channel. The saving is performing only the waveform segment that is defined by the Start and Length defined on the ARB menu. To save a full wave or other waveforms parts, the Start and Length must be changed accordingly.

3.6.8 SWEEP Key

Selects the Sweep Mode and allows the entering of sweep parameters as Sweep Start, Stop and Sweep Rate.
To select the sweep mode, press SWEEP, then press the function key that corresponds to the desired Sweep menu option, as shown:



Sweep Menu

F1: ON/OFF - Operates the sweep function, selecting between Sweep ON or OFF.

F2: Start - Defines the Sweep Start frequency.

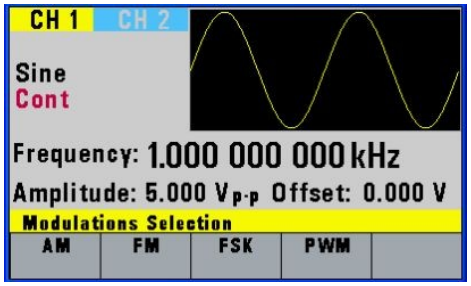
F3: Stop - Defines the Sweep Stop frequency.

F4: Rate - Defines the Sweep Rate.

F5: Lin/Log - Selects the Sweep Shape, LIN or LOG.

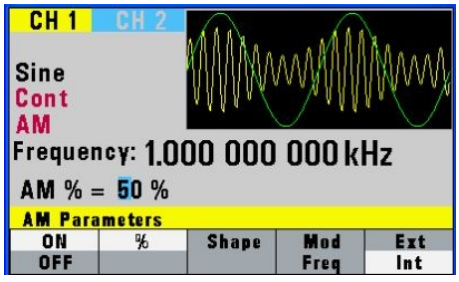
3.6.9 MODULATION Key

Selects the Modulation mode **AM, FM, FSK** or **PWM**.
To select the output mode, press MODUL key, then press the function key that corresponds to the desired menu option, as shown:



Modulation Menu

If the **F1**:**AM** is selected, the following menu is available:



AM Menu

F1: ON/OFF - Selects the Modulation ON or OFF operating mode.

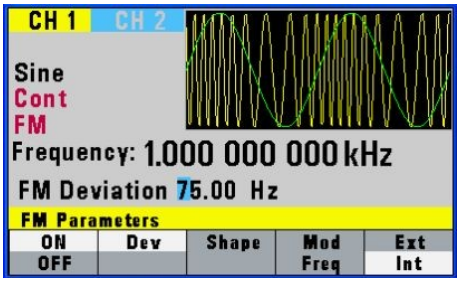
F2: % - Defines the modulation depth, from 0 to 100%.

F3: Shape - Selects the internal modulation shape between SINE, TRIANGLE or SQUARE .

F4: Mod Freq - Selects the internal modulation frequency, from 0.01Hz to 20.00KHz.

F5: Ext/Int - Selects and enables the external modulation by an external signal applied to the Modulation In connector.

If the **F2:FM** is selected, the following menu is available:



FM Menu

F1: ON/OFF - Selects the Modulation ON or OFF operating mode.

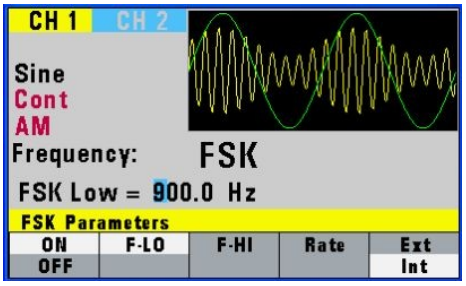
F2: Dev - Defines the FM deviation frequency.

F3: Shape - Selects the internal modulation shape between SINE, TRIANGLE or SQUARE.

F4: Mod Freq - Selects the internal modulation frequency, from 0.01Hz to 20.00KHz.

F5: Ext/Int - Selects and enables the external modulation by the external signal applied to the Modulation In connector.

If the **F3:FSK** is selected, the following menu is available:



FSK Menu

F1: ON/OFF - Selects the FSK ON or OFF operating mode.

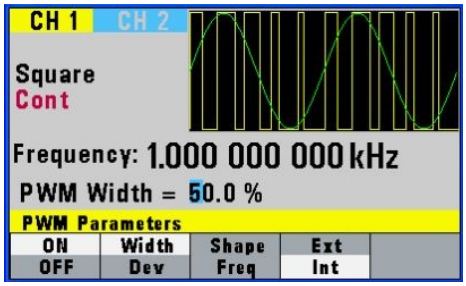
F2: F-LO - Defines the low frequency of the FSK.

F3: F-HI - Defines the high frequency of the FSK.

F4: Rate - Selects the rate of the alternating between the low and high frequencies.

F5: Ext/Int - Selects and enables the external FSK when the unit frequency is alternating between the low and high frequencies by an external signal applied to the Trig In connector.

If the **F4:PWM** is selected, the following menu is available:



FSK Menu

F1: ON/OFF - Selects the FSK ON or OFF operating mode.

F2: Width-Dev - Defines the Pulse Width or the deviation. Both % of period, 50% being the symmetrical pulse.

F3: Shape Freq - Defines the internal shape frequency of the modulating wave.

F5: Ext/Int - Selects and enables the external PWM by an external signal applied to the Modulation In connector.

3.7 ON Key

Use these key to control the main output signal. When the output is active, a build-in LED in the OUT key is illuminated.

3.8 Cursor Movement Keys

Use these keys to move the cursor (when visible) either left or right. They are used in conjunction with the rotary input knob to set the step size of the rotary input knob.

3.9 Rotary Input Knob

Use this knob to increase and decrease numeric values or to scroll through a list. The cursor indicates the low-order position of the displayed value which changes when you rotate the knob (for straight numeric entries only). For other types of data, the whole value changes when you rotate the knob.

3.10 Power-On Settings

At power-on, the waveform generator performs a diagnostic self-test procedure to check itself for errors. If it find an error, and error code and text appear in the display window. Other error codes appear when you enter and invalid

front-panel setting. For more information on error codes, see the Error Indication section.
When the waveform generator finishes the diagnostic self-test routine, it enters the local state (LOGS) and assumes power-on default settings. Table 3-2 lists the factory default settings. You can program the waveform generator for any settings you want at power on, as described earlier in this section.

Table 3-2
Power-on Default Settings

Key Function		Comments
FREQUENCY	1000000 Hz	Wave frequency
RATE(ARB)	1 us	Sample time per point
AMPLITUDE	5.00V	Peak to peak output amplitude
FUNCTION	SINE	Output waveform
OFFSET	0.00V	Zero offset
REPETITION	10mS	Internal trigger rate
MODE	CONT	Waveform mode
N-BURST	2	Waves per burst
START ADRS	1	Start memory address
WAVELENGTH	1000	Number of points per waveform
TRIG SOURCE	EXT	External trigger source
OUTPUT	OFF	Output disabled
SWEEP	OFF	Sweep execution
MODULATION	OFF	Modulation execution

3.11 Memory

The waveform generator uses two forms of memory:
* Nonvolatile FLASH for storing arbitrary waveform data and front panel settings.
* Dynamic RAM for generating arbitrary waveform output.

3.11.1 Nonvolatile Flash (Arbitrary Waveform Memory)

The FLASH stores the arbitrary waveform memory, which the waveform generator uses to store the data points that make up an arbitrary waveform. This memory is a matrix 16383 x 16,000,000 that you use to plot your arbitrary waveforms.
The FLASH also stores the settings storage buffer, which the waveform generator uses to store up to 49 front-panel setups. For more information about the settings storage buffer, see SETUPS key earlier in this section.
Because it is impossible to 100% guarantee against loss of stored data, a back-up copy of the data stored in memory should be kept in an external device or computer, so that can be restored, if necessary.

3.11.2 Dynamic RAM

The DRAM stores waveform output. For arbitrary waveform output, the waveform generator loads the selected contents of the FLASH (arbitrary waveform memory) into the DRAM. In order to save the arbitrary edited data in the flash memory, the instrument will display a message " SAVE DATA" after each modification of the arbitrary waveform in the EDIT menu.

3.12 Displaying Errors

At power-on, the waveform generator performs a diagnostic routine to check itself for problems. If the diagnostic

routine finds an error, an error message is displayed. The waveform generator also displays error messages when front-panel settings are either invalid or may produce unexpected results.

Error messages for MODEL 687

Message Text	Cause
Out of range	Attempt to set variable out of instrument limits.
Setting conflict	Can't have this parameter set with some other.
Trig rate short	Internal trigger rate too short for wave/burst.
Empty location	Attempt to restore non existent setting.
SCALE too high	Attempt to set scale too high for current dot value
Protected RAM	Attempt to write to protected RAM range.
RAM error	Error in testing RAM.
Save RAM	New firmware installed.
Must divide by 4	Predefined wave length must be divisible by 4.
Must divide by 2	Predefined wave length must be divisible by 2.

3.13 Using The MODEL 687

This section explains how to generate various waveforms and modify the output waveform, including:

* Generating a standard waveform
* Creating an arbitrary waveform
* Loading a waveform into execution memory
* Generating a waveform output
* Modifying waveform output
* Storing and recalling a waveform generator setup

The explanation is for one channel only. Since the channels are identical and can be operated separately, the procedures for the second channel operation are similar.

3.13.1 Waveform and Execution Memory

The waveform generator has two types of memory: waveform memory and execution memory. Waveform memory consists of 16,777,216 points at which you can specie a value from -8191 to 8191.
You can scale these points in execution memory through the output amplitude setting; that is, data point 8191 is equal to the positive peak of the output amplitude.

3.13.2 Selecting a Standard Waveform

You can select several standard waveforms as: sine, triangle and square. Creating a standard waveform requires selecting the waveform type, parameters, modes, etc., and their settings that define the waveform.
Generating a standard waveform requires the following:

* Selecting the waveform
* Setting the output frequency
* Setting the output amplitude and offset

3.13.3 Selecting the Waveform

To select a waveform:

1. Press the WAVEFORM key.
2. Select the waveform from the menu available.

3.13.4 Setting the Output Frequency

To set the output frequency:
1. Press PARAMETER to select the Frequency parameter.
2. Use the rotary knob to set the frequency.

3.13.5 Setting the Output Amplitude

To set the output amplitude:
1. Press PARAMETER to display the Parameter menu.
2. Press F3:AMPL to select the Amplitude parameter.
3. Use the rotary knob to set the amplitude.
4. Press F:OFST again to select the Offset parameter.

3.13.6 Setting the Output Mode

To set the output mode:
1. Press MODE to display the Mode menu on the display window.
2. Press the function key (Fl to F5) that corresponds to the desired mode.

3.13.7 Setting the Output

To set the output channel, press the Output ON key.

3.14 Examples

3.14.1 Creating an Arbitrary Waveform

You can create an arbitrary waveform using the following methods:
* Enter individual data points
* Draw lines between data points
* Create a predefined waveform
* Combine any of these methods

NOTE

You can use a software package such as **Wave-X** *to create Arbitrary Waveforms.*
For more information on software packages, contact your local sale office.

No need tot use all 16,000,000 data points for one waveform. You can program any number of waveforms into waveform memory, keeping in mind the addresses where one waveform ends and the other begins.
The waveform's frequency and amplitude are influenced by the number of data points and their value in the waveform. For further information on how the number of data points influence the frequency and amplitude of a waveform in execution memory, see the Setting the Frequency and Setting the Amplitude sections, respectively.

3.14.2 Entering Individual Data Points

The most basic way to program an arbitrary waveform is to enter data points for the waveform, one data point at a time. While this can become tedious the auto-increment function helps this process.
To enter individual data points into waveform memory, follow these steps:
1. Press WAVEFORM to display the waveform selection menu.
2. Press F5 :ARB to display the arbitrary menu.
3. Press F4:EDIT to display the Edit menu.

4. Press F1:POINT, to select the point by point programming mode.
5. Press F1:ADDRESS
6. Use the rotary knob or the numerical keys to enter the address.
7. Press F2:DATA.
8. Use the rotary knob r the numerical keys to enter the value for the data point. Valid entries range from –8191 to 8191.
9. Repeat steps 5 through 9 until you finish programming your arbitrary waveform.

NOTE

Each time you press ENTER to complete a data point entry in numerical mode, the auto-increment address advances the "A= value" by one.

3.14.3 Creating an Arbitrary Waveform

To create a complex arbitrary waveform:

* Load a predefined sine waveform
* Load a scaled sine waveform at the positive peak of the first sine wave
* Draw a straight line between two data points in the waveform
* Add a pulse/glitch to the waveform
* Add a noise signal at the negative peak of the first sine wave

To see the waveform as you build it, connect the waveform generator to an oscilloscope and perform the following steps:

ACTION	**KEYSTROKES**
Loads the first 1000 points of waveform memory into execution memory.	WAVEFORM F5:ARB F1:START 1 F2:LENGTH 1000 F5:PREV MODE F1:CONT OUTPUT ON

The following steps set up the waveform shown in Figure 3-2.

ACTION	**KEYSTROKES**
Step 1. Load a 1000 point, 50% scaled, predefined sine wave into waveform memory starting at address 1	WAVEFORM F5:ARB F4:EDIT F3:PREDEF F1:TYPE (rotate knob for selection) SINE F2:FROM/DATA 1 F3:LENG 1000 F4:SCAL 50 F5:EXEC F3:YES

Step 2.	Load a 5% scaled, 100 point predefined waveform into waveform memory starting at address 200.	WAVEFORM F5:ARB F4:EDIT F3:PREDEF F1:TYPE SINE F2:FROM 200 F3:LENG 100 F4:SCAL 5 F5:EXEC F3:YES
Step 3.	Draw a line between address 251 (the highest point of the sine wave) and address 501 (where the sine wave crosses the origin).	WAVEFORM F5:ARB F4:EDIT F2:LINE F1:FROM 251 F2:TO 501 F4:EXEC F3:YES
Step 4.	Add a negative pulse/glitch (data value -4095) at addresses 600 through 606.	WAVEFORM F5:ARB F4:EDIT F1:PONT F1:ADRS 600 F2:DATA -4095 (repeat -4095 and ENTER for addresses 601-606)
Step 5.	Add a 5% noise signal to addresses 700 through 800.	WAVEFORM F5:ARB F4:EDIT F3:PREDEF F2:FROM 700 F3:LENG 100 F4:SCAL 5 F1:TYPE NOISE F5:EXEC F1:ADD F4:EXEC F3-YES

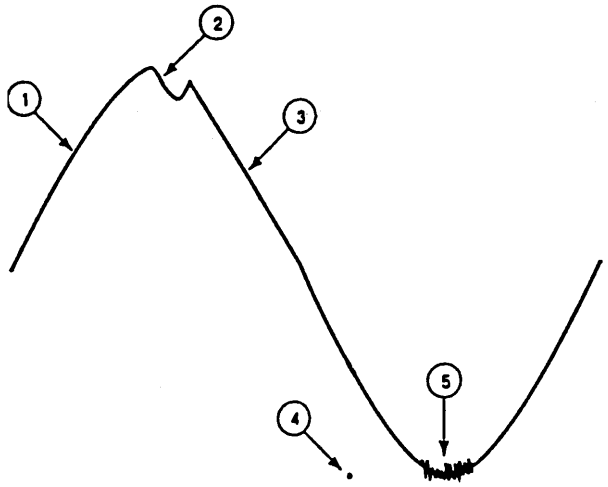


Figure 3-2: Steps to set up an Arbitrary waveform.

3.14.4 Setting the Frequency

The arbitrary waveform frequency is a function of the number of data points used to run the waveform (the length parameter in the ARBITRARY menu) and the waveform execution point rate. The waveform execution point rate is the execution time between each point in the waveform. The total time taken to run one period of the waveform is given by:

number of points X rate

Because the output frequency is a function of the rate and the number of points being executed, the output frequency other waveform is:

$$\text{frequency} = \frac{1}{\text{number of points X rate}}$$

For example, to set the output frequency to 1000Hz, given the number of data points used for the waveform output is 1000, calculate:

$$\text{rate} = \frac{1}{\text{1000 points X 1000Hz}} = \text{lus}$$

EXAMPLE: Setting the Output Frequency
To set the output frequency of a 1000 point waveform in execution memory to 1000Hz, set the rate to 1us:

ACTION	KEYSTROKES
Step 1. Set the output rate to 1 us (equivalent to 1000Hz output frequency)	PARAMETER F1 :RATE 1 KHz/us

Once you have created the waveform, you must specify which memory segment to execute. For information about specking a segment of memory to execute, see the Loading a Waveform into Execution Memory section.

3.14.5 Setting the Amplitude

The following equation represents the relative output amplitude voltage relationship between the front-panel amplitude peak-to-peak setting and the data point values in waveform memory:

$$\text{output voltage} = \frac{\text{amplitude p-p setting} \times \text{data point value}}{16392} + \text{offset}$$

Where 16392 is the data point value range in waveform memory.

Table 3-4: Relative Amplitude for Waveform Output (Examples)

Front-panel Amplitude Setting	Data Point Value	Relative Output Amplitude Voltage
5V peak-to-peak	8191	2.5V positive peak
5V peak-to-peak	0	0V (offset voltage)
10V peak-to-peak	8191	5Vpositive peak

3.14.6 Loading a Waveform into Execution Memory

To load a waveform into execution memory, specify its starting address and length in the ARBITRARY menu.

1. Select the channel to ON.
2. Press WAVEFORM and select the F5:ARB function.
3. Press F1:START to set the address. Valid entries range from 1 to 15,999,999.
4. Press F2:LENGTH to display the length of the waveform. Valid entries range from 2 to 4,000,000.
6. Use the rotary input knob or the numerical keys to enter the length of the waveform.

3.14.7 Generating a Waveform Output

Once you load a waveform into execution memory, you can output the waveform by setting the mode and output parameters (frequency, amplitude, offset).

Setting the Mode
To set the mode:
1. Press MODE to display the Mode menu.
2. Press F1:CONT to select continuous mode, F2:TRIG to select triggered mode, F3:GATE to select gated mode, or F4:BRST to select burst mode.

Setting the Output
To turn on the main output, press Output ON.

Modifying Waveform Output
You can modify the output of a waveform by:
* Using the offset
* Using modulation

3.14.8 Using Voltage Offset

Through the offset parameter you can add a positive or negative DC level to the output waveform.

To set voltage offset:
1. Press PARAMETERS to display the menu.
2. Press F3 :OFST to display the offset setting.
3. Use the rotary input knob or the numerical keys to set the voltage offset.

To turn the voltage offset OFF, repeat the steps above, but set the offset voltage level to 0.

3.14.9 Storing and Recalling a Waveform Generator Setup

You can store up to 49 front-panel setups in a part of nonvolatile RAM known as the settings storage memory. When you recall a stored setup, the front-panel settings change to match the settings in the stored setup. These stored and recalled settings include the starting address and length of the arbitrary memory that is loaded in the execution memory.

NOTE

Storing a waveform generator's setup values stores neither waveform nor the execution memory.

Storing Setups

To store the front-panel setup:
1. Press SETUPS to display the menu.
2. Press F3:STORE to select the Store mode.
3. Use the rotary input knob to select a buffer number. Valid buffer numbers range from 1 to 49. Buffer 0 is a read-only buffer that contains the power-on settings listed in Table 3-3.

The waveform generator does not warn you when you store a setup into a settings buffer that is already occupied.

Recalling Setups. To recall stored front-panel setup:

1. Press SETUPS to display the menu.
2. Press F1:RECALL to select the Recall mode.
3. Use the rotary input knob to select a buffer number. Valid buffers numbers range from 0 to 49. Buffer 0 is a read-only buffer that contains the power-on settings listed in Table 3-3.

Section 4

Programming

4.1 Overview

This section provides detailed information on programming the MODEL 687 via the IEEE 488 bus (referred to from now as the GPIB - General Purpose Interface Bus). The MODEL 687 is programmable over the IEEE 488.1 bus, and its message protocol is compatible with IEEE 488.2. The device command set is compatible with the SCPI 1992.0 standard.
The command syntax as defined by the IEEE 488.2 and SCPI standards is briefly explained in the following sections. Users who have experience programming GPIB instruments may skip these paragraphs, and go directly to where the individual command syntax is given. Considering the relative newness of these standards, it is recommended to all users to read the explanations given here. Users wishing to gain further insight should consult the standards.

4.2 DEVICE STATE

The device may be in one of the four possible states described below. The transition between states is defined by IEEE 488.1.

4.2.1 Local State (LOCS)

In the LOCS the device may be operated from the front panel only. Its settings may be queried over the GPIB, but not changed. Commands that do not affect the signal being output by the instrument are accepted.

4.2.2 Local With Lockout State (LWLS)

In the LWLS the device may be operated from the front panel only. Its settings may be queried over the GPIB, but not changed. Commands that do not affect the signal being output by the instrument are accepted. The difference between the LOCS and the LWLS is that from the LWLS the device may enter the Remote With Lockout State.

4.2.3 Remote State (REMS)

In the REMS the device may be operated from the GPIB. Actuating any front panel key will cause the device state to revert to the LOCS.

4.2.4 Remote With Lockout State (RWLS)

In the RWLS the device is operable only from the GPIB. Front panel operation may be returned by either sending an appropriate IEEE 488.1 command, or by cycling the device power.

4.3 Interface Function Subsets

The following interface function subsets are implemented in the MODEL 687:

SH1, AH1, T6, L4, SR1, RL1, PP0, DC1, DT1, E2, C0

4.4 Device Address

The GPIB address of the device may be set to any value from 0 to 31. The address may be changed from the front panel, using the numeric keypad or the rotary encoder, or via the GPIB itself using the command:

:SYSTem:COMMunicate:GPIB:ADDRess

Setting the device to address 31 puts it in the 'off-bus' state. In this state it will not respond to messages on the GPIB. If the device is in the REMS when set to address 31, an internal 'return-to-local' command will be given, setting the device to the LOCS. If the device is in the RWLS, the 'return-to-local' command is ignored, and the device remains in the RWLS. The only way to then re-establish communication with the device over the GPIB is to cycle the power, and to then change the address to that required from the front panel.

4.5 MESSAGE EXCHANGE PROTOCOL

The device decodes messages using the Message Exchange Protocol defined in IEEE 488.2. The following functions implemented in the MEP must be considered:

4.5.1 The Input Buffer

The device has a 128-byte long cyclic input buffer. Decoding of remote messages is begun as soon as the input buffer is not empty, that is, as soon as the controller has sent at least one byte to the device. Should the input buffer be filled up by the controller faster than the device can remove the bytes and decode them, the bus handshake is not completed until room has been made for more bytes in the buffer. This prevents a fast controller from overrunning the device with data.
If the user has sent part of a Program Message, but not the Program Message Terminator, and he wishes to abort the message decoding and execution, the Device Clear command may be sent, or front panel operation resumed (in REMS only).

4.5.2 The Output Queue

The device has a 100-byte long output queue in which it stores response messages for the controller to read. If at the time a response message is formatted the queue contains previously formatted response messages, such that there is not enough place in the queue for the new message, the device will put off putting the message in the queue until there is place for it.

The Status Byte MAV bit indicates when set that part or all of a response message is ready to be read.

4.5.3 Response Messages

The device sends a Response Message in response to a valid query. All queries return a single Response Message Unit. In only one case is the Response Message generated when the response is read (as opposed to when the response is parsed), and this is when querying Arbitrary Waveform data. All other queries generate the Response Message when they are parsed.

4.5.4 Coupled Commands

Coupled Commands are either commands whose execution validity depends on the value of other parameters, or commands whose execution changes the value of another parameter. The execution of commands designated as being coupled is deferred until all other commands in the same Program Message have been executed. The coupled commands are then grouped together according to their functionality, and executed as a group. These groups of coupled commands are defined in the MODEL 687:

a) The commands to set the amplitude, the offset, and to switch the output on. The output being switched on is included here in order to prevent possible damage to the equipment being driven as a result of the amplitude and offset not being executed as intended by the user, due to an execution error.

b) The commands to set the function, frequency, the point rate, the wavelength, and the waveform start address. In ARB function, setting the frequency or the point rate causes the other to change, keeping the wavelength constant (if it, too, is not specified in the same program message). If the wavelength is specified as well, the frequency or point rate must change in accordance with the new value. The validity of the start address is a function of the wavelength. Please refer to the individual commands for more detail.
The maximum frequency is also dependant upon the waveform, so that changing the waveform may render the current frequency out of range.

c) The commands to set modulation, modulation source and the function are inter-related. FM and FSK are not available for ARB function. External source of modulation can be active for either FM or AM but not both. FSK and FM cannot be active at the same time.

d) Sweep start and sweep stop frequencies must be distanced more than the minimum allowed for sweep to function correctly.

4.6 Block Data

Arbitrary waveform values may be sent to the device in one of three formats:

a) ASCII values,
b) Definite form arbitrary data, and
c) Indefinite form arbitrary data.

Essentially we would like to check the execution validity of all the data of a command before execution. When downloading a long arbitrary waveform, however, it is not possible to check all the data sent before execution, since this would require excessive amounts of memory. The following compromise has therefore been reached:

An arbitrary waveform is limited in length only by the amount of waveform memory. Each point is checked and then written to memory. If an invalid value is detected all subsequent values are discarded, and an execution error is flagged.

Querying arbitrary waveform data will result in a Response Message containing only as many points as the user requested.

4.7 Instrument Identification

The *IDN? common query is used to read the instrument's identification string. The string returned is as follows:

OR-X, Model 687,0,V1.1

4.8 Instrument Reset

The *RST common command effects an instrument reset to the factory default power up state.

4.9 Self Test

The *TST common query causes the device to perform a self test. This self test consists of checking the functionality of the arbitrary waveform memory.

4.10 COMMAND SYNTAX

4.10.1 General Command Structure

The device commands are generally defined by the SCPI standard, with the exception of those instrument functions for which SCPI commands do not as yet exist. The Common Commands and Queries are defined by IEEE 488.2. The command syntax, ie. how a command is structured, is defined by IEEE 488.2.

A Program Message is defined as a string containing one or more Program Message Units, each of which is an instrument command or query. Program Message Units are separated from each other by the Program Message Unit Separator. The Program Message is terminated by the Program Message Terminator.

The Program Message Unit Separator consists of a semicolon (';'), optionally preceded and/or followed by whitespace characters. A whitespace character is defined as the ASCII characters in the ranges 00H-09H, and 0BH-20H. This range includes the ASCII control characters and the space, but excludes the Linefeed character.

The Program Message Terminator consists of optional whitespace characters, followed by one of three options:

a) the Linefeed (LF) character (ASCII 0A);
b) the GPIB EOI bus line being set true on the last byte of the message;
c) LF being sent with EOI true.

The Program Message Unit can be divided into three sections as follows:

a) Program Header

The Program Header represents the operation to be performed, and consists of ASCII character mnemonics. Two types of Program Headers are used in the 687: Instrument-control headers and Common Command and Query headers. A Program Header may consist of more than one mnemonic, in which case the mnemonics are separated from each other by the colon (':'). For instrument control commands, the mnemonics are specified by the SCPI standard, and indicate the tree structure of the command set. The first mnemonic indicates the subsystem being controlled. Common Command and Query Program Headers consist of a single mnemonic prefixed by an asterisk ('*').

The mnemonics consist of upper - or lower-case alpha characters. Mnemonics may be written in either the long form, in which the entire mnemonic is written out, or the short form, in which only a specified portion of the mnemonic is written out. Some mnemonics have only one form due to their short length. Where a command is described, the portion appearing in upper case is the short form. Only the short form or the long form may be used.

Example: The command to set the frequency to 1KHZ may be written in the following ways:

```
SOURCE:FREQUENCY 1KHZ
SOUR:FREQ 1KHZ
SOURCE:FREQ 1KHZ
```

Some mnemonics in a specified Program Header may be optional. This is indicated in the command description by the mnemonic being enclosed in square brackets ([...]). This means it is not necessary to write the mnemonic into the Program Header: it is a default condition. The 'SOURCE' mnemonic, for example, is optional. Not specifying it will cause the device to search for the mnemonics in the Program Header under the Source Subsystem. For example, the frequency may be set by the commands

FREQ:CW 1KHz
FREQ 1KHz

since the 'CW' mnemonic is also optional.

b) Program Header Separator

The Program Header Separator is used to separate the program header from the program data. It consists of one or more whitespace characters, denoted as <ws>. Typically, it is a space.

c) Program Data

The Program Data represent the values of the parameters being set, for example, the '1KHZ' in the above examples. Different forms of program data are accepted, depending on the command. The Program Data types used in the instrument are:

i) Character program data

This form of data is comprised of a mnemonic made up of lower - or upper-case alpha characters. As with Program Header mnemonics, some Character Data mnemonics have short and long forms. Only the short or the long form may be used.

ii) Boolean data

Boolean data indicate that the parameter can take one of two states, ON or OFF. The parameter may be character type
ON or OFF

or numeric. A numeric value is rounded to an integer. A non-zero result is interpreted as 1 (ON), and a zero result as 0 (OFF).
Queries return the values 0 or 1.

iii) NRf

This is a decimal numeric data type, where
NR1 indicates an integer number,
NR2 indicates a fixed point real number, and
NR3 indicates a floating point real number.

iv) Expression data

An expression is contained in parentheses (...). This data type is used only with the STATus:QUEue:ENABle command.

v) Numeric value program data

This data type defines numeric values, as well as special cases of Character Data. Numeric values may be specified in any of Integer, Fixed Point or Floating Point format. All parameters which have associated units accept a suffix, which may be specified using upper or lower-case characters. When the suffix is not specified, the numeric value is accepted in the default units, which are Hertz for frequency, Seconds for time, and Volts for voltage. To set the frequency to 1KHz we can send one of the following commands:

FREQ 1000
FREQ 1E3

The special forms of character data accepted as numbers are

MAXimum: sets the parameter to its maximum value.
MINimum: sets the parameter to its minimum value.

For example, to set the frequency to it's maximum value we can send the command

FREQ MAX

vi) Arbitrary Block Data

The Arbitrary block data type is used to send arbitrary waveform data to the instrument. In this data type, the waveform points are specified in binary format, and each point consists of two bytes. Two types of arbitrary block data are defined (by IEEE 488.2):

Definite Form

The Definite Form has the structure

-#-Byte Count Length - Byte Count - 8-bit byte

The Byte Count Length consists of a single ASCII digit from 1 to 9. It tells the parser how many digits are in the Byte Count.

The Byte Count is a decimal integer made up of the number of digits specified in the Byte Count Length. It tells the parser how many 8-bit bytes are being sent.

Indefinite Form

The Indefinite Form has the structure

- # - 0 – 8-bit byte – LF^EOI

Some Program Message Units either require, or can accept, more than one data element. Program data elements are separated from each other by the Program Data Separator. It is defined as optional whitespace characters followed by a comma (','), which in turn is followed by optional whitespace characters.

There are two types of Program Message Units: Command Message Units and Query Message Units. A Query differs from a Command in that the Program Header is terminated with a question mark ('?'). For example, the frequency might be queried with the following query:

FREQ?

Some Query Message Units accept data, giving the device more specific information about what is being queried. In many cases the Query Message Unit may optionally be supplied with the MIN or MAX mnemonics as data. This tells the device to return the minimum or maximum value to which the parameter may currently be set. For example,

FREQ? MAX

will return the maximum value to which the frequency may currently be set.

Not all Program Message units have query forms (for example, STATUS:PRESET), and some Program Message Units might have only the query form (for example SYSTEM:VERSION?).

The instrument puts the response to the query into the output queue, from where it may be read by the controller. The Status Byte MAV bit is set to indicate to the controller that a response is ready to be read.

4.10.2 SCPI Command Structure

SCPI commands are based on a hierarchical structure. This allows the same instrument-control header to be used several times for different purposes, providing that the mnemonic occurs in a unique position in the hierarchy. Each level in the hierarchy is defined as a node. Mnemonics in the different levels are separated from each other by a colon (':'). The first Program Message Unit, or command, in a Program Message is always referenced to the root node. Subsequent commands are referenced to the same level as the previous command. A Program Message Unit having a colon as its first character causes the reference to return to the root. This process is defined by IEEE 488.1, section A.1.1. Consider the following examples:

a) The following command may be used to set the amplitude and the offset of the signal.

SOURCE:VOLTAGE:AMPLITUDE 5V;OFFSET 2V

Note that the offset command is referenced to the command preceding it: the OFFSET mnemonic resides at the same node as the AMPLITUDE command.

b) This command set the frequency and the amplitude.

SOURCE:FREQUENCY 2KHZ;VOLTAGE:AMPLITUDE 4V

The FREQUENCY and VOLTAGE mnemonics are at the same level.

c) When Program Message Units describe different subsystems, a colon prefix must be used to reset the command reference to the root. Here the frequency and the output state are set.

SOURCE:FREQUENCY 3KHZ;:OUTPUT:STATE ON

Common Commands may be inserted in the Program Message without affecting the instrument-control command reference. For example,

SOURCE:VOLTAGE:AMPLITUDE 4V;*ESE 255;OFFSET 2V

In an instrument with multiple channels, the selection of which channel to use is achieved through the use of a numeric suffix indicating the channel, attached to the root level mnemonic. Four root level mnemonics are channel - dependent, and these are SOURce, TRIGger, OUTPut and ARBitrary. When the channel is not specified, channel 1 is assumed. Program message units that are referred back to a specific root level mnemonic operate on the channel specified in that mnemonic.

Examples:

a) SOUR:FREQ 5KHZ;VOLT:AMPL 3V

Sets the frequency and amplitude of the first channel (default).

b) SOUR2:FREQ 5KHZ;VOLT:AMPL 3V

Sets the frequency and amplitude of channel 2.

c) ARB2:START 100;LENGTH 50

Sets the start address and length of the waveform being output on channel 2.

4.11 STATUS REPORTING

The instrument is capable of reporting status events and errors to the controller, using the IEEE 488.1 Service Request function and the IEEE 488.2 Status Reporting structure.

4.11.1 The Status Byte

Status summary information is communicated from the device to the controller using the Status Byte (STB). The STB is composed of single-bit summary-messages, each summary message summarizing an overlying Status Data Structure. By examining the content of the STB, the controller gains some information concerning the instrument's status.

The STB bits are defined as follows:

Bit 0: Unused
Bit 1: Unused
Bit 2: Error/event queue summary message (EVQ). This bit is set if the queue is not empty.
Bit 3: Questionable Status summary message.
Bit 4: Message Available (MAV) summary message. This bit is set whenever all or part of a message is available for the controller to read. The controller may be ready to read the response message before it is available, in which case it can either wait until this bit is set, or it can start to read. In the second case, the controller timeout must be set so that the read action will not be aborted before the message has been read.
Bit 5: Event Status Bit (ESB) summary message. This bit is set to indicate that one or more of the enabled standard events have occurred.
Bit 6: Request Service (RQS). This bit is set when the device is actively requesting service.
Bit 7: Operation Status summary message. No Operation Status events are defined in the instrument, and so this bit is never set.

The STB is read by the controller during a serial poll. If the RQS bit was set, it is then cleared. The STB may also be read by the *STB? common query.

4.11.2 Service Request Enabling

Service request enabling allows the user to select which Status Byte summary messages may cause the device to actively request service. This is achieved using the Service Request Enable Register, which is an 8-bit register whose bits correspond to those of the STB. The RQS bit in the STB is set when a bit in the STB is set, and its corresponding bit in the service request enable register is set.

The service request enable register is set using the *SRE common command, and read using the *SRE? common query.

4.11.3 Standard Event Status Register

The Standard Event Status Register (SESR) is defined by IEEE 488.2. It is implemented in the instrument as a byte, whose bits have the following definitions:

Bit 0: Operation Complete (OPC). This bit is set in response to the *OPC common command being executed.
Bit 1: Request Control (RQC). Not implemented.

Bit 2: Query Error (QYE). This bit is set when either the controller is attempting to read data from the device when none is available, or when data prepared for the controller to read has been lost.
Bit 3: Device-Specific Error (DDE). This bit is set to indicate that a device operation did not execute due to some device condition. For example, trying to recall an uninitialized device stored setting.
Bit 4: Execution Error (EXE). This bit is set when the device could not execute a command, due to the command being outside of it's capabilities. For example, a parameter being out of range.
Bit 5: Command Error (CME). This bit is set to indicate an error in the command syntax.
Bit 6: User Request (URQ). This bit is not used.
Bit 7: Power On (PON). This bit is set when the device is powered on.

The SESR is queried using the *ESR? common query.

The SESR is paired with an enable register, the Standard Event Status Enable Register (SESER). This register enables one or more events in the SESR to be reflected in the Status Byte ESB summary message bit. The bits of the SESER correspond to those of the SESR. Setting a bit in the SESER enables the corresponding event to set the ESB bit when it occurs. The SESER is set with the *ESE common command and queried with the *ESE? command query.

4.11.4 The Error Queue

The error queue is used to store codes of errors detected in the device. It is implemented as a cyclic buffer of length 10. When the error queue is not empty, bit EVQ in the Status Byte is set. The error queue is read with either one of the following two queries:

:SYSTEM:ERROR?
:STATUS:QUEUE:NEXT?

The first error in the queue is returned, and the queue is advanced.

4.11.5 Error Codes

The negative error codes are defined by SCPI. Positive codes are specific to the instrument.

The error message is returned in the form

<error number>,"<error description>"

A table of error numbers and their descriptions is presented here.

No error reported

0 No error

Command Errors

A command error is in the range -199 to -100, and indicates that a syntax error was detected. This includes the case of an unrecognized header. The occurrence of a command error causes the CME bit (bit 5) of the Standard Event Status Register to be set.

-100	Command Error
-101	Invalid character
-102	Syntax error
-103	Invalid separator
-104	Data type error

-105	GET not allowed
-108	Parameter not allowed More parameters than allowed were received
-109	Missing parameter Fewer parameters than necessary were received
-110	Command header error
-111	Header separator error
-112	Program mnemonic too long The mnemonic must contain no more than 12 characters.
-114	Header suffix out of range; Only suffix of 1 or 2 is valid, indicating the channel being operated.
-113	Undefined header
-120	Numeric data error
-121	Invalid character in number
-123	Exponent too large IEEE 488.2 specifies maximum of 32000
-124	Too many digits IEEE 488.2 specifies maximum of 255 digits in mantissa.
-128	Numeric data not allowed A different data type was expected
-131	Invalid suffix
-134	Suffix too long A maximum of 12 characters are allowed in a suffix.
-138	Suffix not allowed
-140	Character data error.
-141	Invalid character data. Incorrect character data were received.
-144	Character data too long Character data may contain no more than 12 characters.
-148	Character data not allowed
-158	String data not allowed
-161	Invalid block data An error was found in the block data
-168	Block data not allowed
-170	Expression error Only 6 error ranges may be specified.
-171	Invalid expression An error was found in the expression.
-178	Expression data not allowed

Execution Errors

An execution error indicates that the device could not execute a syntactically correct command, either since the data were out of the instrument's range, or due to a device condition. The EXE bit (bit 4) of the Standard Event Status Register is set on occurrence of an execution error.

-200	Execution error An attempt was made to RECALL the contents of an uninitialized stored setting buffer.
-201	Invalid while in local.
-211	Trigger ignored. The GET or *TRG common command was ignored due to the device not being in the correct state to execute the trigger.
-220	Parameter error. A parameter is in the correct range, but conflicts with other parameters.
-221	Settings conflict. The parameter is out of range due to the current instrument state.
-222	Data out of range.
-223	Too much data.

The arbitrary waveform memory limit has been exceeded.

-224 Illegal parameter value.
The parameter value must be selected from a finite list of possibilities.

-241 Hardware missing; A command was sent to operate a non-existent channel.

-258 Media protected.
An attempt was made to write to protected arbitrary waveform memory.

Device-Specific Errors

An error specific to the device occurred. The DDE bit (bit 3) of the Standard Event Status Register is set.

-315 Configuration memory lost.
Device memory has been lost.

-330 Self-test failed.

-350 Queue overflow.
Error codes have been lost due to more than 10 errors being reported without being read.

Query Errors

A query error indicates that the output queue control has detected a problem. This could occur if either an attempt was made to read data from the instrument if none was available, or when data were lost. Data could be lost when a query causes data to be formatted for the controller to be read, and the controller sends more commands without reading the data.

-410 Query INTERRUPTED.
Data were sent before the entire response of a previous query was read.

-420 Query UNTERMINATED.
An attempt was made to read a response before the complete program message meant to generate that response was sent.

-430 Query DEADLOCKED.
The input buffer and output queue are full, and the controller is attempting to send more data. In this case the output queue and input buffers will be cleared. Parsing will resume after the END message is detected.

-440 Query UNTERMINATED after indefinite response.
A query was received in the same program message after a query requiring an indefinite response was formatted. Essentially this means that the *IDN? common query and the :ARB:DATA? query should not be followed by more query messages in the same program message.

System Events

System events have positive valued codes. They are not defined by SCPI, but are specific to the instrument. Sending the :STATus:PRESet command will disable these events from being reported.

401 Power on

402 Operation complete
The *OPC command has been executed.

Warnings

The execution of some commands might cause an undesirable instrument state. The commands are executed, but a warning is issued. Sending the :STATus:PRESet command disables reporting of warnings. The existence of these conditions causes a bit in the Status Questionable Condition register to be set (refer to section 13.5.4).

500 Trigger rate short on channel 1

501 Trigger rate short on channel 2

510 Output overload on channel 1

511 Output overload on channel 2

"Trigger rate short" means that the period of the waveform is larger than the value of the internal trigger rate. Thus not every trigger will generate a cycle (or burst) of the waveform.

4.12 IEEE 488.2 COMMON COMMANDS

4.12.1 System Data Commands

a) ***IDN?** - Identification query

The identification query enables unique identification of the device over the GPIB. This query should always be the last in a program message. It returns a string with four fields:

Manufacturer name
Model name
Serial number (0 if not relevant)
Version number

Command
Type: Common Query
Syntax: *IDN?
Response: OR-X, Model 687,0,V1.0

b) ***OPT?** - Option identification query

The Option Identification Query is used to identify device options over the system interface. This query should always be the last in a program message.

Command
Type: Common Query
Syntax: *OPT?
Response : Dual channel.

4.12.2 Internal Operation Commands

a) ***RST** - Reset command

The Reset command performs a device reset. It causes the device to return to the factory default power up state.

Type: Common Command
Syntax: *RST

b)***TST?** - Self-test query

The self-test query causes an internal self-test to be performed. This test consists of checking the GPIB function.

Type: Common Query

Syntax: *TST?
Response: ASCII 0 if test passes
ASCII 1 if test fails

4.12.3 Synchronization Commands

a) ***OPC** - Operation complete command

The operation complete command causes the device to generate the operation complete message in the Standard Event Status Register, on completion of the selected device operation.

Type: Common Command
Syntax: *OPC
Examples: FREQ 5KHZ;*OPC

The *OPC command (and the *OPC? query described below) find use mainly when commands having relatively long execution times are executed, for example the programming of long predefined waveforms.

b) ***OPC?** - Operation complete query

The operation complete query places an ASCII character 1 in the output queue on completion of the selected device operation.

Type: Common Query
Syntax: *OPC?
Response: ASCII character 1
Example FREQ 1KHz;*OPC?

c) ***WAI** - Wait-to-continue command

This command is intended for use with overlapped commands. No commands in the instrument are overlapped, and so this command has no effect.

Type: Common Command
Syntax: *WAI

4.12.4 Status and Event Commands

a) ***CLS** - Clear status

The clear status command clears the SESR and Error Queue status data structures.

Type: Common Command
Syntax: *CLS

b) ***ESE** - Standard event status enable

This command is used to set the value of the Standard Event Status Enable Register.
Arguments
Type: NRf
Range: 0 to 255. Non integer arguments are rounded before execution.
Type: Common Command or Query
Syntax: *ESE<ws><NRf>
Examples: *ESE 48 (Enables the CME and EXE bits)

*ESE 255 (Enables all standard events)
Query
Syntax: *ESE?
Response: <NR1>

c) ***ESR?** - Standard event status register query

This query is used to read the value of the Standard Event Status Register. Reading the register clears it.

Type: Common Query
Syntax: *ESR?
Response: <NR1>

d) ***PSC** - Power on status clear command

This command is used to control the automatic power-on clearing of certain status functions.

Arguments
Type: Boolean
Type: Common Command or Query
Command
Syntax: *PSC<ws><Boolean>
Examples: *PSC ON or *PSC 1
*PSC OFF or *PSC 0
Query
Syntax: *PSC?
Response: ASCII 0 for OFF
ASCII 1 for ON

When set to ON (1), the Service Request Enable Register and the Standard Event Status Enable Register are cleared on power-on.

e) ***SRE** - Service request enable command

This command sets the Service Request Enable Register bits.

Arguments
Type: NRf
Range: 0 to 255. Non integer arguments are rounded before execution.
The value of bit 6 is ignored, and is set always to zero.

Type: Common Command or Query
Command
Syntax: *SRE<ws><NRf>
Examples: *SRE 48 (Enables reporting of ESB and MAV events)
Query
Syntax: *SRE?
Response: <NR1>

f) ***STB?** - Status byte query

This query is used to read the value of the Status Byte.

Type: Common Query
Syntax: *STB?
Response: <NR1>

The value of the Status Byte read with the *STB? query may differ from that read with the Serial Poll. Bit 6 of the STB will be set as long as a reason for requesting service exists, while bit 6 of the STB as read by the Serial Poll is cleared by the Serial Poll.

4.12.5 Device Trigger Commands

a) ***TRG** - Trigger command

This command is analogous to the IEEE 488.1 Group Execute Trigger interface message, and has the same effect. It is used to trigger the device to output a wave, and is accepted only when the trigger mode is set to Trigger, Gate or Burst, and the trigger source is set to BUS.

Type: Common Command
Syntax: *TRG

4.12.6 Stored Settings Commands

a) ***RCL** - Recall instrument state

This command is used to restore the state of the device to that stored in the specified memory location.

Arguments
Type <NRf>
Range 0 to 50. Non integer values are rounded before execution
Type: Common Command
Syntax: *RCL<ws><NRf>
Example: *RCL 0 (Recall default state)
*RCL 50

Stored setting location 50 stores the last instrument setting before power down.

b) ***SAV** - Save instrument state

This command is used to store the current instrument state in the specified memory location.

Arguments
Type: <NRf>
Range: 1 to 49. Non integer values are rounded before execution
Type: Common Command
Syntax: *SAV<ws><NRf>
Example: *SAV 25

Stored setting location 0 stores the factory defaults, and is a read-only location. Location 50 stores a copy of the current instrument setting, and it, too, is read-only.

4.13 INSTRUMENT CONTROL COMMANDS

Instrument control commands are grouped into logical subsystems according to the SCPI instrument model. The commands are comprised of mnemonics indicating the subsystem to which the command belongs, and the hierarchy within that subsystem. When the command is to be referred to the Root node, it should be prefixed with a colon (:).

Mnemonics appearing in square brackets [...] are optional. The '|' character is used to denote a choice of specifications. The '<ws>' is used to denote a white space character.

All commands except those in the STATus and SYSTem subsystems, and commands specifically indicated, are channel-dependent. Commands are referenced to channel 1 by default. The numeric suffix '2' must be appended to the subsystem mnemonic to refer a command to channel 2. (See paragraph 10.2.)

4.13.1 SOURce Subsystem

The Source Subsystem controls the frequency, voltage, amplitude modulation and clock source. The command structure is as follows:

```
:SOURce
    :FREQuency
        [:CW|:FIXed] <numeric value>
    :FUNCtion
        [:SHAPe] SINusoid|SQUare|TRIangle||ARBitrary||PULSe
    : DCYCle ] <numeric value>
    :VOLTage
        [:AC]
            [:LEVel]
                [:IMMediate]
                    [:AMPLitude] <numeric value>
                    :OFFSet <numeric value>
    :REFerence
        INTernal |EXTernal
    :ROSCillator
        [:SOURCE] INTernal |EXTernal

    :AM
                            [:STATe] <Boolean>
                            :DEPTh <numeric value>
                            :SHAPe SINusoid|SQUare|TRIangle
                            :FREQuency <numeric value>
                            :SOURce INTernal |EXTernal
    :FM
                            [:STATe] <Boolean>
                            :DEViation <numeric value>
                            :SHAPe SINusoid|SQUare|TRIangle
                            :FREQuency <numeric value>
                            :SOURce INTernal |EXTernal
    :FSK
                            [:STATe] <Boolean>
                            :LOWFrequency <numeric value>
                            :HIFrequency <numeric value>
                            :RATE <numeric value>
                            :SOURce INTernal |EXTernal

    :PWM
                            [:STATe] <Boolean>
                            WIDth <numeric value>
                            :DEViation <numeric value>
                            :SHAPe SINusoid|SQUare|TRIangle
```

:FREQuency <numeric value>
:SOURce INTernal |EXTernal

:SWEep
:STATe <Boolean>
:SPACing <LIN|LOG>
:TIME <numeric value>
:STARt <numeric value>
:STOP <numeric value>

:PHAse
[:ADjust] <numeric value >
SYNChronize

:PULSe
:PERiod <numeric value >
:WIDth <numeric value >
:EDGe <numeric value >
:RISe <numeric value >
:FALl <numeric value >

4.13.1.1 Frequency

:SOURce:FREQuency <frequency>

The frequency command controls the frequency of the output waveform.

Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)
Range: Dependent on the Point Rate and Wavelength.
Fmax = 1/(5 ns * Wavelength)
Fmin = 1/(100 S * Wavelength)
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce]:FREQuency[:CW]<ws><frequency>[units]
[:SOURce]:FREQuency<ws>MINimum|MAXimum
Examples: :FREQ 5KHZ
:FREQ 5E3
:FREQ MAXIMUM
:FREQ MIN
Query
Syntax: [:SOURce]:FREQuency[:CW]?[<ws>MAXimum|MINimum]
Examples: :FREQ?
:FREQ? MAX
Response: NR3

Considerations:

1) The MIN | MAX arguments should be used only in a Program Message that does NOT contain Program Message Units specifying Arbitrary Point Rate or Wavelength, since the MAXimum or MINimum value is calculated at the time the command is parsed.
2) The MIN and MAX arguments refer to currently settable minimum or maximum.
3) FIXed is alias for CW.

4.13.1.2 Amplitude

:SOURce:VOLTage[:AMPLitude] <p-p amplitude>

The amplitude command is used to set the peak-to-peak amplitude of the output waveform. Note that the amplitude and the offset are limited by the relation

Peak Amplitude + |Offset| <= 5V

Arguments
Type: Numeric
Units: V, mV, VPP, mVPP
Range: 10mV to 10V
Rounding: 1mV.
Command Type: Setting or Query
Setting
Syntax: [:SOURce]:VOLTage:AMPLitude<ws><amplitude>[units]
[:SOURce]:VOLTage:AMPLitude<ws>MINimum|MAXimum
Examples: :VOLT:AMPL 2.5
:VOLT:AMPL 2.5V
:VOLT:AMPL MAX
Query
Syntax: [:SOURce]:VOLTage:AMPLitude? [<ws>MINimum|MAXimum]
Examples: :VOLT:AMPL?
:VOLT:AMPL? MAX
Response: NR2
Considerations:

1) The MAXimum amplitude is dependent on the offset.
2) The MAX and MIN arguments should not be used in a program message containing an OFFSet command, since these values are evaluated during parsing, based on the current value of the offset.

4.13.1.3 Offset

:SOURce:VOLTage:OFFSet <offset>

The offset command is used to set the DC offset of the output waveform. Note that the amplitude and the offset are limited by the relation

Peak Amplitude + |Offset| ≤ 5V

Arguments
Type: Numeric
Units: V, mV
Range: 10mV to 5V
Rounding: to 1mV
Command Type: Setting or Query

Setting
Syntax: [:SOURce]:VOLTage:OFFSet<ws><offset>[units]
[:SOURce]:VOLTage:OFFSet<ws>MINimum|MAXimum
Examples: :VOLT:OFFS 2.5
:VOLT:OFFS 2.5V
:VOLT:OFFS MAX
Query
Syntax: [:SOURce]:VOLTage:OFFSet?[<ws>MINimum|MAXimum]
Examples: :VOLT:OFFS?
:VOLT:OFFS? MAX
Response: NR2
Considerations:

1) The MAXimum offset is dependent on the amplitude.
2) The MAX and MIN arguments should not be used in a program message containing an AMPLitude command, since these values are evaluated during parsing, based on the current value of the amplitude.

4.13.1.4 Clock Reference Source

:SOURce:REFerence<clock source>

This command is used to select the source of the arbitrary waveform clock. This clock sets the arbitrary waveform point rate.

Arguments
Type: Character
Options: INTernal, EXTernal
Command Type: Setting or Query
Setting
Syntax: [:SOURce]: REFerence <ws><option>
Examples: :REF INT
:REF EXT
Query
Syntax: [:SOURce]: REFerence ?
Response: INT|EXT

4.13.1.5 Function

:SOURce:FUNCtion [:SHAPe]

The function command is used to set the type of waveform to be generated by the instrument.

Command Type: Setting or Query
Setting
Syntax: [:SOURce]:FUNCtion[:SHAPe]<WS><OPTION>
Examples: :FUNC SIN
:FUNC ARB
Query
Syntax: [:SOURce]:FUNCtion[:SHAPe]?
Examples: :FUNC?
Response: SIN|TRI|SQU|ARB||PULS
Considerations:

The following functions are available:

SINusoid,SQUare, TRIangle, ARBitrary,|PULSe

4.13.1.6 AM modulation

The following commands control the AM modulation:

AM STATe

This command activates or deactivates AM modulation:

Arguments

Type:	Boolean
Command Type:	Setting or Query
Setting	
Syntax:	[: SOURce:]AM [:STATe]<ws>ON|1|OFF|0
Examples:	: SOURce:AM :STAT ON AM OFF
Query	
Syntax:	[:SOURce:]AM[:STATe]?
Response:	0|1

AM DEPTh

This command sets the AM modulation depth in %

Arguments	
Type:	Numeric
Units:	none (implied %)
Range:	0 to 100
Rounding:	To integer
Command Type:	Setting or Query
Setting	
Syntax:	:SOURce:AM:DEPTh<ws><percent depth> :SOURce:AM:DEPTh<ws>MINimum|MAXimum
Examples:	AM:DEPTh 50
Query	
Syntax:	AM:DEPTh?[<ws>MINimum|MAXimum]
Response:	NR3

AM SHAPe

This command selects the AM modulating waveform shape

Arguments	
Type:	Character
Options:	SINusoid, TRIangle, SQUare
Command Type:	Setting or Query
Setting	
Syntax:	[:SOURce:]AM:SHAPe<ws><SIN|TRI|SQU>
Examples:	[:SOURce:]AM:SHPE SIN AM:SHAPE TRI
Query	
Syntax:	[:SOURce:]AM:SHAPe?
Response:	SIN|TRI|SQU

AM FREQuency

This command sets the AM modulating waveform frequency

Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)
Range: Fmax = 20 KHz
Fmin = 0.01 Hz
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]AM:FREQuency<ws><frequency>[units]
[:SOURce:]AM:FREQuency<ws>MINimum|MAXimum
Examples: AM:FREQ 5KHZ
AM:FREQ 5E3
AM:FREQ MAXIMUM
AM:FREQ MIN
Query
Syntax: [:SOURce:]AM:FREQuency?[<ws>MAXimum|MINimum]
Examples: AM:FREQ?
AM:FREQ? MAX
Response: NR3

AM SOURce

This command selects the AM modulation source as either internal (then the above settings are effective) or external (and then the external waveform determines depth, shape and frequency of modulation).

Arguments
Type: Character
Options: INTernal, EXTernal
Command Type: Setting or Query
Setting
Syntax: [:SOURce:] AM:SOURce<ws><option>
Examples: AM:SOUR INT
AM:SOUR EXT
Query
Syntax: [:SOURce]:AM:SOURce?
Response: INT|EXT

4.13.1.7 FM modulation

The following commands control the FM modulation:

1) FM STATe

This command activates or deactivates FM modulation:

Arguments
Type: Boolean
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]FM[:STATe]<ws>ON|1|OFF|0
Examples: FM:STAT ON
FM OFF
Query
Syntax: [:SOURce:]FM[:STATe]?
Response: 0|1

1. FM DEViation

This command sets the FM modulation deviation

Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)
Range: Dependent on the carrier frequency.
Fmax = carrier frequency
Fmin = 10 uHz
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce]:FM:DEViation<ws><frequency>[units]
[:SOURce]:FM:DEViation<ws>MINimum|MAXimum
Examples: FM:DEV 5KHZ
FM:DEV 5E3
FM:DEV MAXIMUM
FM:DEV MIN
Query
Syntax: [:SOURce]:FM:DEViation?[<ws>MAXimum|MINimum]
Examples: FM:DEV?
FM:DEV? MAX
Response: NR3

2. FM SHAPe

This command selects the FM modulating waveform shape

Arguments
Type: Character
Options: SINusoid, TRIangle, SQUare
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]FM:SHAPe<ws><SIN|TRI|SQU>
Examples: [:SOURce:]FM:SHPE SIN
FM:SHAPE TRI
Query
Syntax: [:SOURce:]FM:SHAPe?
Response: SIN|TRI|SQU

3. FM FREQuency

This command sets the FM modulating waveform frequency

Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)
Range: Fmax = 20 KHz
Fmin = 0.01 Hz
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]FM:FREQuency<ws><frequency>[units]
[:SOURce:]FM:FREQuency<ws>MINimum|MAXimum
Examples: FM:FREQ 5KHZ

FM:FREQ 5E3
FM:FREQ MAXIMUM
FM:FREQ MIN

Query
Syntax: [:SOURce:]FM:FREQuency?[<ws>MAXimum|MINimum]
Examples: FM:FREQ?
FM:FREQ? MAX
Response: NR3

4. FM SOURce

This command selects the FM modulation source as either internal (then the above settings are effective) or external (and then the external waveform determines deviation, shape and frequency of modulation).

Arguments
Type: Character
Options: INTernal, EXTernal
Command Type: Setting or Query
Setting
Syntax: [:SOURce:] FM:SOURce<ws><option>
Examples: FM:SOUR INT
FM:SOUR EXT
Query
Syntax: [:SOURce]:FM:SOURce?
Response: INT|EXT

4.13.1.8 PWM modulation

The following commands control the PWM modulation:

1) PWM STATe

This command activates or deactivates PWM modulation:

Arguments
Type: Boolean
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]PWM[:STATe]<ws>ON|1|OFF|0
Examples: PWM:STAT ON
PWM OFF
Query
Syntax: [:SOURce:]PWM[:STATe]?
Response: 0|1

2) PWM WIDth

This command sets the PWM modulation width

Arguments
Type: Numeric.
Units: % (default)
Range: Dependent on the carrier frequency.
Rounding: The value is rounded to 1%.
Command Type: Setting or Query
Setting
Syntax: [:SOURce]:PWM:WID <ws><width>
[:SOURce]:PWM:WIDth <ws>MINimum|MAXimum

Examples: PWM:WID 50
PWM:WID 5E1
PWM:WID MAXIMUM
PWM:WID MIN
Query
Syntax: [:SOURce]:PWM:WIDth?[<ws>MAXimum|MINimum]
Examples: PWM:WID?
PWM:WID? MAX
Response: NR3

2. PWM DEViation

This command sets the PWM modulation deviation

Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)
Range: Dependent on the carrier frequency.
PWMax = carrier frequency
PWMin = 10 uHz
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce]:PWM:DEViation<ws><frequency>[units]
[:SOURce]:PWM:DEViation<ws>MINimum|MAXimum
Examples: PWM:DEV 5KHZ
PWM:DEV 5E3
PWM:DEV MAXIMUM
PWM:DEV MIN
Query
Syntax: [:SOURce]:PWM:DEViation?[<ws>MAXimum|MINimum]
Examples: PWM:DEV?
PWM:DEV? MAX
Response: NR3

3. PWM SHAPe

This command selects the PWM modulating waveform shape

Arguments
Type: Character
Options: SINusoid, TRIangle, SQUare
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]PWM:SHAPe<ws><SIN|TRI|SQU>
Examples: [:SOURce:]PWM:SHPE SIN
PWM:SHAPE TRI
Query
Syntax: [:SOURce:]PWM:SHAPe?
Response: SIN|TRI|SQU

4. PWM FREQuency

This command sets the PWM modulating waveform frequency

Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)
Range: PWMax = 20 KHz

PWMin = 0.01 Hz
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]PWM:FREQuency<ws><frequency>[units]
[:SOURce:]PWM:FREQuency<ws>MINimum|MAXimum
Examples: PWM:FREQ 5KHZ
PWM:FREQ 5E3
PWM:FREQ MAXIMUM
PWM:FREQ MIN
Query
Syntax: [:SOURce:]PWM:FREQuency?[<ws>MAXimum|MINimum]
Examples: PWM:FREQ?
PWM:FREQ? MAX
Response: NR3

5. PWM SOURce

This command selects the PWM modulation source as either internal (then the above settings are effective) or external (and then the external waveform determines deviation, shape and frequency of modulation).

Arguments
Type: Character
Options: INTernal, EXTernal
Command Type: Setting or Query
Setting
Syntax: [:SOURce:] PWM:SOURce<ws><option>
Examples: PWM:SOUR INT
PWM:SOUR EXT
Query
Syntax: [:SOURce]:PWM:SOURce?
Response: INT|EXT

4.13.1.9 FSK modulation

The following commands control the FSK modulation:

FSK STATe

This command activates or deactivates FSK modulation:Arguments
Type: Boolean
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]FSK[:STATe]<ws>ON|1|OFF|0
Examples: FSK:STAT ON
FSK OFF
Query
Syntax: [:SOURce:]FSK[:STATe]?
Response: 0|1

FSK LOWFrequency

This command sets the lower of the two frequencies used in FSK modulation.
Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)

Range: The whole frequency range of the current function.
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]FSK:LOWFrequency<ws><frequency>[units]
[:SOURce:]FSK:LOWFrequency<ws>MINimum|MAXimum
Examples: FSK:LOWFrequency 5KHZ
FSK:LOWF 5E3
FSK:LOWF MAXIMUM
FSK:LOWF MIN
Query
Syntax: [:SOURce:]FSK:LOWFrequency?[<ws>MAXimum|MINimum]
Examples: FSK:LOWF?
FSK:LOWF? MAX
Response: NR3

FSK HIFrequency

This command sets the higher of the two frequencies used in FSK modulation.
Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)
Range: The whole frequency range of the current function.
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]FSK:HIFrequency<ws><frequency>[units]
[:SOURce:]FSK:HIFrequency<ws>MINimum|MAXimum
Examples: FSK:HIFrequency 5KHZ
FSK:HIF 5E3
FSK:HIF MAXIMUM
FSK:HIF MIN
Query
Syntax: [:SOURce:]FSK:HIFrequency?[<ws>MAXimum|MINimum]
Examples: FSK:HIF?
FSK:HIF? MAX
Response: NR3

FSK RATE

This command sets the rate of switching between the two frequencies of the modulation.
Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)
Range: Fmax = 100KHz
Fmin = 0.01Hz
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]:FSK:RATE<ws><frequency>[units]
[:SOURce:]:FSK:RATE <ws>MINimum|MAXimum
Examples: FSK:RATE 5KHZ
FSK:RATE 5E3
FSK:RATE MAXIMUM
FSK:RATE MIN
Query
Syntax: [:SOURce]:FSK:RATE ?[<ws>MAXimum|MINimum]

Examples: FSK:RATE ?
FSK:RATE ? MAX
Response: NR3

FSK SOURce

This command selects the FSK modulation source as either internal (then the above settings are effective) or external (and then the external waveform determines the frequency of modulation).

Arguments
Type: Character
Options: INTernal, EXTernal
Command Type: Setting or Query
Setting
Syntax: [:SOURce:] FSK:SOURce<ws><option>
Examples: FSK:SOUR INT
FSK:SOUR EXT
Query
Syntax: [:SOURce]:FSK:SOURce?
Response: INT|EXT

4.13.1.10 .Sweep control

The following commands control the sweep functionality:

Sweep STATe

This command activates or deactivates sweep:

Arguments
Type: Boolean
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]SWEEP[:STATe]<ws>ON|1|OFF|0
Examples: SWEEP:STAT ON
SWEEP OFF
Query
Syntax: [:SOURce:]SWEEP[:STATe]?
Response: 0|1
Note: Sweep cannot be activated in ARB or if either FSK or FM are active.

Sweep SPACing

This command sets the sweep spacing as either LINear or LOGarithmic:

Arguments
Type: Character
Options: LINear,LOGarithmic
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]SWEEP:SPACing<ws><LIN|LOG>
Examples: [:SOURce:] SWEEP:SPACing LIN
SWEEP:SPAC LOG
Query
Syntax: [:SOURce:] SWEEP:SPACing ?
Response: LIN|LOG

Sweep TIME

This command sets the time for one complete sweep:

Arguments
Type: Numeric
Units: S, mS, uS, nS
Range: 10mS to 500S
Rounding: To 4 digits
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]SWEEP:TIME<ws><time>[units]
[:SOURce:]SWEEP:TIME<ws>MINimum|MAXimum
Examples: SWEEP:TIME 50MS
Query
Syntax: [:SOURce:]SWEEP:TIME?[<ws>MINimum|MAXimum]
Response: NR3

Sweep STARt

This command sets the start frequency of the sweep:

Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)
Range: Dependent on the frequency range of the current function.
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]SWEEP:STARt<ws><frequency>[units]
[:SOURce:]SWEEP:STARt<ws>MINimum|MAXimum
Examples: SWEEP:STARt 5KHZ
SWEEP:STARt 5E3
SWEEP:STARt MAXIMUM
SWEEP:STARt MIN
Query
Syntax: [:SOURce:]SWEEP:STARt?[<ws>MAXimum|MINimum]
Examples: SWEEP:STARt ?
SWEEP:STARt ? MAX
Response: NR3

Sweep STOP

This command sets the stop frequency of the sweep:

Arguments
Type: Numeric.
Units: MHz, KHz, Hz (default)
Range: Dependent on the frequency range of the current function.
Rounding: The value is rounded to 4 digits.
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]SWEEP:STOP<ws><frequency>[units]
[:SOURce:]SWEEP:STOP<ws>MINimum|MAXimum
Examples: SWEEP:STOP 5KHZ
SWEEP:STOP 5E3
SWEEP:STOP MAXIMUM
SWEEP:STOP MIN
Query
Syntax: [:SOURce:]SWEEP:STOP?[<ws>MAXimum|MINimum]

Examples: SWEEP:STOP ?
SWEEP:STOP ? MAX
Response: NR3

4.13.1.11 Phase adjust

This command controls the phase adjustment. It is valid only for DDS waveforms (SIN,SQU,TRI).

Arguments
Type: Numeric
Units: none (degrees implied)
Range: -180 to +180 (other values converted to this range)
Rounding: To integer
Command Type: Setting or Query
Setting
Syntax: [:SOURce:]PHASe<ws><phase>
[:SOURce:]PHASe<ws>MINimum|MAXimum
Examples: [:SOURce:]PHASe 500NS
Query
Syntax: [:SOURce:]PHASe?[<ws>MINimum|MAXimum]
Response: NR3

4 Phase Synchronize

This command synchronizes the phases of both channels.

Arguments None
Command Type: Setting only
Setting
Syntax: [:SOURce:]PHASe:SYNC
Examples: [:SOURce:]PHASe:SYNC

4.13.1.12 PULSe setting

The following commands control the pulse function:
Note that *width +0.6*(rise+fall)<period* in order to have valid values.

PULSe PERiod

This command sets the pulse period to the specified value.
NOTE: This also sets the pulse frequency.

Arguments
Type: Numeric
Units: S, mS, uS, nS
Range: 40nS to 2000S
Rounding: To integer
Command Type: Setting or Query
Setting
Syntax: [:SOURce:] PULSe: PERiod <ws><value>
[:SOURce:] PULSe: PERiod <ws>MINimum|MAXimum
Examples: [:SOURce:] PULSe: PERiod 500NS
Query
Syntax: [:SOURce:] PULSe: PERiod?[<ws>MINimum|MAXimum]
Response: NR3

PULSe WIDth

This command pulse width to the specified value.

Arguments
Type: Numeric
Units: S, mS, uS, nS
Range: 20nS to 2000S
Rounding: To integer
Command Type: Setting or Query
Setting
Syntax: [:SOURce:] PULSe: WIDth <ws><value>
[:SOURce:] PULSe: WIDth <ws>MINimum|MAXimum
Examples: [:SOURce:] PULSe: WIDth 500NS
Query
Syntax: [:SOURce:] PULSe: WIDth?[<ws>MINimum|MAXimum]
Response: NR3

PULSe EDGe

This command sets both rising and falling edge of the pulse to the specified value.

Arguments
Type: Numeric
Units: S, mS, uS, nS
Range: 0 (10nS) and 100nS minimum. Maximum depends on pulse width and period
Rounding: To integer
Command Type: Setting or Query
Setting
Syntax: [:SOURce:] PULSe: EDGe <ws><value>
[:SOURce:] PULSe: EDGe <ws>MINimum|MAXimum
Examples: [:SOURce:] PULSe: EDGe 500NS
Query
Syntax: [:SOURce:] PULSe: EDGe?[<ws>MINimum|MAXimum]
Response: NR3

PULSe RISe

This command sets rising edge of the pulse to the specified value.

Arguments
Type: Numeric
Units: S, mS, uS, nS
Range: 0 (10nS) and 100nS minimum. Maximum depends on pulse width and period
Rounding: To integer
Command Type: Setting or Query
Setting
Syntax: [:SOURce:] PULSe: RISe <ws><value>
[:SOURce:] PULSe: RISe <ws>MINimum|MAXimum
Examples: [:SOURce:] PULSe: RISe 500NS
Query
Syntax: [:SOURce:] PULSe: RISe?[<ws>MINimum|MAXimum]
Response: NR3

PULSe FALl

This command sets falling edge of the pulse to the specified value.

Arguments
Type: Numeric
Units: S, mS, uS, nS
Range: 0 (10nS) and 100nS minimum. Maximum depends on pulse width and period
Rounding: To integer
Command Type: Setting or Query

Setting
Syntax: [:SOURce:] PULSe: FALl <ws><value>
[:SOURce:] PULSe: FALl <ws>MINimum|MAXimum
Examples: [:SOURce:] PULSe: FALl 500NS
Query
Syntax: [:SOURce:] PULSe: FALl?[<ws>MINimum|MAXimum]
Response: NR3

Duty Cycle

:SOURce:DCYCle <duty cycle value<

This command is used to set the duty-cycle of the square wave or the symmetry of triangular wave. The value is given in percent and the maximum and minimum are frequency dependent, as per instrument specifications.

Arguments
Type: Numeric
Units: None (percent implied)
Range: 20 to 80 (1 to 99% for triangle)
Rounding: To integer
Command Type: Setting or Query
Setting
Syntax: :SOURce:DCYCle <ws><duty cycle value<
:SOURce:DCYCle <ws>MINimum|MAXimum
Query
Syntax: : SOURce: DCYCle?[<ws>MINimum|MAXimum]
Response: NR3

4.13.2 OUTPut Subsystem

The Output Subsytem controls characteristics of the source's output. Included in this subsystem are the State and Summing commands. The command structure is as follows:

:OUTPut
[:STATe] <Boolean>
:TERminator <Boolean>

Output State

:OUTPut [:STATe] <state 0,1>

This command controls whether the output is ON or OFF.
Arguments
Type: Boolean
Command Type: Setting or Query
Setting
Syntax: :OUTPut[:STATe]<ws>ON|1|OFF|0
Examples: :OUTP:STAT ON
:OUTP OFF
Query

Syntax: :OUTPut[:STATe]?
Response: 0|1

Output Impedance

:OUTPut :TERM <state 0,1>

This command controls whether the 50Ohm terminator is ON or OFF.

Arguments
Type: Boolean
Command Type: Setting or Query
Setting
Syntax: :OUTPut:TERM<ws>ON|1|OFF|0
Examples: :OUTP: TERM ON
:OUTP: TERM OFF
Query
Syntax: :OUTPut:TERM?
Response: 0|1

4.13.3 Trigger Subsystem

The Trigger Subsystem is used to control the waveform triggering. The command structure is as follows:

:TRIGger
:MODE CONTinuous|TRIGger|GATE|BURSt
:BURSt <numeric value>
:SOURce <MANual|INTernal|EXTernal|BUS
:TIMer <numeric value>

Trigger Mode

:TRIGger:MODE <trigger mode>

This command is used to set the trigger mode. It is not a standard SCPI command.

Arguments
Type: Character
Options: CONTinuous
TRIGger
GATE
BURSt
Command Type: Setting or Query
Setting
Syntax: :TRIGger:MODE<ws><option>
Examples: :TRIG:MODE CONT
:TRIG:MODE BURS
Query
Syntax: :TRIGger:MODE?
Response: CONT|TRIG|GATE|BURS

Trigger Source

:TRIGger:SOURce <trigger source>

This command is used to select the trigger source, for use in the Trigger, Gate and Burst trigger modes.

Arguments	
Type:	Character
Options:	MANual - Front panel MAN key
	BUS - GPIB trigger (GET or *TRG)
	INTernal - Internal trigger
	EXTernal - External trigger
Command Type:	Setting or Query
Setting	
Syntax:	:TRIGger:SOURce<ws><option>
Examples:	:TRIG:SOUR BUS
	:TRIG:SOUR INT
Query	
Syntax:	:TRIGger:SOURce?
Response:	MAN|BUS|INT|EXT

Burst Count

:TRIGger:BURSt <burst count>

Used to set the number of cycles to be output in the BURST mode. It is not a standard SCPI command.

Arguments	
Type:	Numeric
Range:	2 to 999999
Rounding:	To integer value
Command Type:	Setting or Query
Setting	
Syntax	:TRIGger:BURSt<ws><value>
Examples	:TRIG:BURS 100
	:TRIG:BURS MAXIMUM
Query	
Syntax:	:TRIGger:BURSt?[<ws>MAXimum|MINimum]
Response:	NR1
Examples:	:TRIG:BURST?
	:TRIG:BURS? MAX

Internal Trigger Rate

:TRIGger:TIMer <trigger rate>

Sets the rate of the internal trigger.

Arguments	
Type:	Numeric
Units:	S, mS, uS, nS
Range:	1E-6S to 100S
Rounding:	to 4 digits

Command Type: Setting or Query
Setting
Syntax: :TRIGger:TIMer<ws><value>[units]
:TRIGger:TIMer<ws>MINimum|MAXimum
Examples: :TRIG:TIM 10E-6
:TRIG:TIM MIN
Query
Syntax: :TRIGger:TIMer?[<ws>MINimum|MAXimum]
Response: NR3
Examples: :TRIG:TIM?
:TRIG:TIM? MIN

4.13.4 Arbitrary Subsystem

The Arbitrary subsystem is not part of the SCPI standard. It was developed to suit the needs of the instrument. Within this subsystem are found commands to:
1) control the point rate, start address, wavelength, marker address, and synchronization pulse address;
2) set values of the arbitrary waveform, either discretely or using predefined, copy or draw functions;
3) protect an area of waveform memory;
4) set the state of the automatic update and increment features;
5) update the waveform.

The unit has 16,777,216 memory locations for Models 685 and 687 and 4,194,304 for Models 665 and 667. The limits on the following text are 16,000,000 or 4000,000 for simplicity purpose. The real numbers are 16,777,216 or 4,194,304.

The following shows the structure of the ARBitrary subsystem:

```
:ARBitrary
    :PRATe <numeric value>
                :ADDRess <numeric value>
                :DATA <numeric value>|<arbitrary block>
                :DRAW <numeric value>,<numeric value>
                :CLEar <numeric value>,<numeric value>
                :COPY <NRf>,<NRf>,<NRf>
                :PROTect
                    [:RANGe] <numeric value>,<numeric value>
                    :STATe <Boolean>
                :PREDefined <shape>,<start address>,<length>,<scale>
                :STARt <numeric value>
                :LENGth <numeric value>
                :MARKer
                    [:ADDRess] <numeric value>
        :STATe <Boolean>
        :LENGth <numeric value>
    :SAVe <numeric value>
    :LOAD <numeric value>
```

Point Rate

ARBitrary:PRATe <point rate>

This command is used to set the point rate. It is coupled with the frequency of the waveform by the relation:

Frequency = 1/(Point Rate * Wavelength)

Thus changing the point rate will result in a change in frequency.

Arguments
Type: Numeric
Units: S, mS, uS, nS
Range: 5nS to 100S
Rounding: to 4 digits
Command Type: Setting or Query
Setting
Syntax: :ARBitrary:PRATe<ws><point rate>[units]
:ARBitrary:PRATe<ws>MINimum|MAXimum
Examples: :ARB:PRAT 100NS
Query
Syntax**:** :ARBitrary:PRATe?[<ws>MINimum|MAXimum]
Response: NR3

Address

:ARBitrary:ADDRess <address>

This command sets the current address of the waveform. It is used to determine to where arbitrary data are to be written.

Arguments
Type: Numeric
Range: 1 to 16,000,000 (4,000,000 for Model 665)
Rounding: to integer value
Command Type: Setting or Query
Setting
Syntax: :ARBitrary:ADDRess<ws><address>
:ARBitrary:ADDRess<ws>MINimum|MAXimum
Examples: :ARB:ADDR 100
Query
Syntax: :ARBitrary:ADDRess?[<ws>MINimum|MAXimum]
Response: NR1

Data

:ARBitrary:DATA <data>

This command is used to set the values of the waveform.

Arguments
Type: Numeric. Definite form arbitrary block. Indefinite form arbitrary block
Numeric Range: -8191 to 8191 ASCII
Rounding: to integer value
Binary Range: 001H to 3FFFH BINARY
Command Type: Setting or Query
Setting Syntax
Numeric: :ARBitrary:DATA<ws><numeric>{[,<numeric]}
Example :ARB:DATA 100,200,1000,2000,-2000

Arbitrary Block: :ARBitrary:DATA<ws>#<arb block data>
See paragraph 4.16
Examples
Definite Form: :ARB:DATA #14\x8\x64\x8\xC8
'\x' indicates that the values are Hexadecimal.
Indefinite: :ARB:DATA #0\x8\x64\x8\xc8\xa^EOI
Query
Syntax: :ARBitrary:DATA?<ws><number of points>, BINary|ASCii
Response: Using the BINary option, data are returned in the Indefinite arbitrary block form.
Using the ASCii option, data are returned in the decimal numeric form.
Considerations:

Data cannot be written to protected memory.
In binary form, each data point consists of two bytes.
The high byte must precede the low byte.

Line Draw

:ARBitrary:DRAW <start address>,<end address>

This command is used to generate a straight line between two points in the arbitrary waveform memory.

Arguments
Type: Numeric.
Range: 1 to 16,000,000
Rounding: to integer value
Command Type: Setting only
Setting
Syntax: :ARBitrary:DRAW<ws><start address>, <end address>
Example: :ARB:DRAW 1,1000
Considerations:

1) The value of the data at the start and end points must first be set by the user, using the :ARB:DATA command.
2) The range of the straight line cannot overlap with protected memory.
3) The end address must be greater than the start address.

Clear

:ARBitrary:CLEar <start address>,<end address>

This command is used to clear all or a portion of waveform memory. The memory is the set to the value zero.

Arguments
Type: Numeric.
Numeric Range: 1 to 16,000,000
Rounding: to integer value
Command Type: Setting only.
Setting
Syntax: :ARBitrary:CLEar<ws><start address>, <end address>
Examples: :ARB:CLE 1,1000
Considerations:

1) The clear range cannot overlap with protected memory.

2) The end address must be greater than the start address.

Copy

:ARBitrary:COPY <start address>,<length>,<destination address>

This command is used to copy a section of the waveform to a different location in waveform memory.

Arguments
Type: NRf
Range: 1 to 16,000,000
Rounding: to integer value
Command Type: Setting only
Setting
Syntax: :ARBitrary:COPY<ws><start>,<length>,<destination>
Example: :ARB:COPY 1,1000,1001
Considerations:

1) The destination range cannot overlap with protected memory.
2) The destination range cannot overlap with the source range.
3) The destination end address may not exceed the maximum address:

Start address + Length - 1 <= 16,000,000
Destination address + Length - 1 <= 16,000,000

Memory Protection Range

:ARBitrary:PROTect [:RANGe] <start address>,<end address>

This command is used to define a range of arbitrary waveform memory to be write-protected. The protection is effective only if the PROTect:STATe is ON.

Arguments
Type: Numeric.
Numeric Range: 1 to 16,000,000
Rounding: to integer value.
Command Type: Setting or Query
Setting
Syntax: :ARBitrary:PROTect[:RANGe]<ws><start>,<end>
Examples: :ARB:PROT 1,1E3
Query
Syntax: :ARBitrary:PROTect[:RANGe]?
Response: <protect start>,<protect end> in NR1 format.

Memory Protection State

:ARBitrary:PROTect:STATe <Boolean>

This command is used to enable or disable arbitrary waveform write-protection.

Arguments
Type: Boolean
Command Type: Setting or Query
Setting

Syntax: :ARBitrary:PROTect:STATe<ws>ON|1|OFF|0
Example: :ARB:PROT:STAT ON
Query
Syntax: :ARBitrary:PROTect:STATe?
Response: 0|1

Predefined waveforms

:ARB:PRED <shape>,<start address>,<length>,<scale>

This command is used to load the waveform memory with a specific type of waveform.

Arguments
Shape
Type: Character
Options: SINusoid
SQUare
TRIangle
NOISe (Pseudo-Random Noise)
ANOise (Noise added to the current waveform)
URAMp (Ramp up)
DRAMp (Ramp down)
SINXx (Sin[x]/x)
EXPUp (Exponent up)
EXPDn (Exponent down)
GAUS
Start Address
Type: Numeric. The MIN and MAX forms both set the address to 1
Range: 1 to 16,000,000
Rounding: to integer value
Length
Type: Numeric.
Range
SIN: 16 to 65,536; divisible by 4
SQU: 2 to 65,536; divisible by 2
TRI: 16 to 65,536; divisible by 4
URAM: 16 to 65,536
DRAM: 16 to 65,536
NOIS: 16 to 65,536
ANO: 16 to 65,536
SINX: 16 to 65,536
EXPU: 16 to 65,536
EXPD: 16 to 65,536
GAUS: 16 to 65,536
Rounding: to integer value.
Scale
Type: Numeric. MIN sets the scale to 1; MAX sets the scale to 100
Range: 1 to 100 (See considerations)
Rounding; to integer value.
Command Type: Setting only
Setting
Syntax: :ARBitrary:PREDefined<ws> <shape>, <start>, <length>,<scale>
Examples: :ARB:PRED SIN,1,1e3,100
:ARB:PRED URAM,1001,1e3,50
Considerations:

1) The start address and the length must meet the specification that. Start address + Length - 1 <= 16,000,000

2) The 'scale' refers to the scaling of the waveform as a percentage of full scale. A scale of 100% will, under the correct conditions, generate a waveform whose data values range from -8191 to +8191. These 'correct conditions' are set by the 'offset' value. This offset is the value of the data at the start address, and determines the maximum scale settable. The following table shows the data values required in order to achieve maximum scale.

SHAPE	DATA
SIN	0
SQU	0
TRI	0
NOIS	0
URAM	-8191
DRAM	+8191
SINX	0
EXPU	-8191
EXPD	+8191
GAUS	0

Start Address

:ARBitrary:STARt <start address>

This command sets the start address of the waveform to be run.

Arguments	
Type:	Numeric
Range:	1 to 15,999,999
Rounding:	to integer value
Command Type:	Setting or Query
Setting	
Syntax:	:ARBitrary:STARt<ws><start address> :ARBitrary:STARt<ws>MINimum|MAXimum
Example:	:ARB:STAR 100
Query	
Syntax:	:ARBitrary:STARt?[<ws>MINimum|MAXimum]
Examples:	:ARB:START? :ARB:STAR? MIN
Response:	NR1
Considerations:	The start address and length must meet the condition:

Start Address + Length - 1 <= 16,000,000

Wavelength

:ARBitrary:LENGth <length>

This command sets the length of the waveform being run.

Arguments	
Type:	Numeric
Range:	2 to 16,000,000
Rounding:	to integer value
Command Type:	Setting or Query

Setting
Syntax: :ARBitrary:LENGth<ws><length>
:ARBitrary:LENGth<ws>MINimum|MAXimum
Example: :ARB:LENG 1E3
Query
Syntax: :ARBitrary:LENGth?[<ws>MINimum|MAXimum]
Example: :ARB:LENG?
Response: NR1
Considerations: 1) Changing the wavelength will change either the frequency.
2) The minimum wavelength is 2.

Marker Address

:ARBitrary:MARKer [:ADDRess] <marker address>

This command is used to set the address of the marker.

Arguments
Type: Numeric.
Range: 1 to 16,000,000
Rounding: to integer values
Setting
Syntax: :ARBitrary:MARKer[:ADDRess]<ws><marker address>
Examples: :ARB:MARK 45
Query
Syntax: :ARBitrary:MARKer[:ADDRess]?
Example: :ARB:MARK?
Response: Marker address in NR1 format.
Considerations: The marker is only output if it's address is within the range of addresses currently being run.

Marker Length

:ARBitrary:MARKer:LENGth <numeric value>

This command is used to set the marker length. The marker length is specified by appending a numeric value, to the MARKer keyword.

Arguments
Type: Numeric
Range: 1 to 4,000
Command Type: Setting or Query
Setting
Syntax: :ARBitrary:MARKer:LENGth<ws><length>
Example: :ARB:MARK:LENG 5
Query
Syntax: :ARBitrary:MARKer:LENGth?
Response: NR1

Marker State

:ARBitrary:MARKer:STATe <Boolean>

This command is used to enable or disable the marker.

Arguments
Type: Boolean
Command Type: Setting or Query
Setting
Syntax: :ARBitrary:MARKer:STATe<ws>ON|1|OFF|0

Query
Syntax: :ARBitrary:MARKer:STATe?
Response: 0/1

Save

:ARBitrary:SAVe

This command is used to save all unsaved arbitrary waveform data into one of the non-volatile memory locations.

Arguments
Type: Numeric
Range: 1 to 4
Command Type: Setting only
Setting
Syntax: :ARBitrary:SAVe <location>

Load

:ARBitrary:LOAD

This command is used to load all arbitrary waveform data from one of the non-volatile memory locations.

Arguments
Type: Numeric
Range: 1 to 4
Command Type: Setting only
Setting
Syntax: :ARBitrary:LOAD <location>

4.13.5 Status Subsystem

This subsystem controls the SCPI-defined status reporting structures, which are the QUEStionable and OPERation status registers, and the error/event queue. The OPERation status registers are mandated by SCPI, and so are implemented, but are not used by the hardware. No status is ever reported through them, and they are not detailed in this manual. The following shows the STATus structure used:

```
:STATus
    :PRESet
        :QUEue
            [:NEXT]?
                :ENABle
        :QUEStionable
            :CONDition?
            :PTRansition <NRf>
            :NTRansition <NRf>
            :EVENt?
            :ENABle <NRf>
```

Status Preset

:STATus:PRESet

This command is used to set certain status values to defined values.

a) The OPERation and QUEstionable enable registers are cleared.
b) The Positive transition filters are set to 32767.
c) The Negative transition filters are set to 0.
d) Only errors in the range (-440:-100) are enabled to be reported in the event queue.

Command Type: Setting only
Setting
Syntax: :STATus:PRESet

Error Queue Read

:STATus:QUEue?

This query returns the first entry in the error queue, and removes that entry from the queue. Its function is identical to that of the :SYSTem:ERRor? query.

Command Type: Query only
Query
Syntax: :STATus:QUEue[:NEXT]?
Response: <Error number>, "<error description>"

Error Queue Enable

:STATus:QUEue:ENABle

This command is used to enable individual errors to be placed in the queue when they occur. Those errors not specified in the :ENABle command are disabled from being reported in the error queue. Errors and events enabled to be reported at power on depends on the Power on Status Clear status (set with the *PSC common command). If PSC is set, the status is cleared on power on, and the range of errors enabled is as set by the STATus:PRESet command, ie -440 to -100. If PSC is cleared, the status is not cleared on power on, and the errors and events enabled are those that were enabled before the last power down.

Type: Expression

The expression data takes the form

(NRf|<event range>[{,NRf|<event range>}])

where NRf represents an error number. Entries are rounded to integer values.

An <event range> is defined as

NRf:NRf

The first number in a range MUST be less than the second.

Up to 6 ranges may be specified using one :ENABle command, representing the 6 ranges of errors/events. The ranges are then separated from each other by Program Data Separators (comma). The entire expression must be enclosed in parentheses(...).

Command Type: Setting or Query
Setting
Syntax: :STATus:QUEue:ENABle<ws><expression>
Example: :STAT:QUE:ENAB (-440:-410,-258:-220,402,-110)
Query
Syntax: :STATus:QUEue:ENABle?
Response: (NRf|<event range>[{,[NRf|event range>]})

Questionable Status

The Questionable status data structure is used to alert the user to instrument conditions that affect the signal quality. Two types of conditions are defined in the AWG, and these are:

1) Frequency - Trigger rate conflict, and
2) Output overload condition.

Each condition is reported separately for each channel. Thus, a total of four conditions may be reported.

The data structure is comprised of five 16-Bit registers. Each bit represents a different status condition. In the AWG, bits 9 and 11 are used as follows:

Bit 9: Frequency - trigger rate conflict.
Bit 11: Output overload.

The existence of these conditions is indicated in the CONDition register. Bit 3 of the status byte is used to indicate the occurrence of a questionable status condition. The conditions cause this bit to be set depending on the values of the other four registers.

The positive transition filter enables a bit in the event register to be set when a condition changes from false to true. The negative transition register enables a bit in the event register to be set when a condition changes from true to false. In order for the bit in the event register to be set, the corresponding bit in the transition register must be set.

Bit 3 in the status byte will be set if a bit in the event register is set while the corresponding bit in the enable register is set.

Questionable Condition

:STATus:QUEstionable:CONDition?

This query is used to read the condition register.

Command Type: Query only
Query
Syntax: :STATus:QUES:COND?

Response: NR1

Positive Transition Filter

:STAT:QUES:PTR

This command is used to set and query the value of the positive transition filter.

Arguments
Type: NRf
Range: 0 to 131,072. Non integer arguments are rounded before execution.
Command Type: Setting or Query
Setting
Syntax: :STAT:QUES:PTR<ws><NRf>
Examples: :STAT:QUES:PTR 2048
Query
Syntax: :STAT:QUES:PTR?
Response: NR1

Negative Transition Filter

:STAT:QUES:NTR

This command is used to set and query the value of the negative transition filter.

Arguments
Type: NRf
Range: 0 to 131,072. Non integer arguments are rounded before execution
Command Type: Setting or Query
Setting
Syntax: :STAT:QUES:NTR<ws><NRf>
Examples: :STAT:QUES:NTR 2048
Query
Syntax: :STAT:QUES:NTR?
Response: NR1

Event Register

:STAT:QUES:EVENt?

This query is used to read the event register. Reading the register clears it.

Command Type: Query only
Query
Syntax: :STATus:QUES:EVEN?
Response: NR1

Event Enable Register

:STAT:QUES:ENABle

This command si used to set and query the value of the enable register.

Arguments
Type: NRf
Range: 0 to 131,072. Non integer arguments are rounded before execution.
Command Type: Setting or Query
Setting
Syntax: :STAT:QUES:ENAB<ws><NRf>
Examples: :STAT:QUES:ENAB 2048
Query
Syntax: :STAT:QUES:ENAB?
Response: NR1

4.13.6 System Subsystem

The SYSTem subsystem collects the functions that are not related to instrument performance. The functions implemented in the AWG are security, GPIB address changing, error queue reading, SCPI version reading, and power-on buffer setting (not SCPI-defined). The command structure is as follows:

:SYSTem
:COMMunicate
:GPIB
:ADDRess <numeric value>
:ERRor?
:VERSion?
:SECurity
[:STATe] <Boolean>
:POBuffer <numeric value>

GPIB Address Change

:SYSTem:COMMunicate:GPIB:ADDRess

This command is used to set the GPIB address.

Arguments
Type: Numeric
Range: 0 to 31
Rounding: to integer value
Command Type: Setting or Query
Setting
Syntax: :SYSTem:COMMunicate:GPIB:ADDRess<ws><address>|MINimum|MAXimum
Example: :SYST:COMM:GPIB:ADDR 20
Query
Syntax: :SYSTem:COMMunicate:GPIB:ADDRess?[<ws>MINimum|MAXimum]
Response: <address> in NR1 format
Considerations: 1) Setting the address to 31 puts the instrument in the off-bus state.
2) Using the MAX option sets the address to 30, not 31.

Default Power-on is address 9.

Error Queue Reading

:SYSTem:ERRor?

This query returns the first entry in the error queue, and removes that entry from the queue. It's function is identical to that of the :STATus:QUEue:NEXT? query.

Command Type: Query only
Query
Syntax: :SYSTem:ERRor?
Response: <Error number>, "<error description>"

SCPI Version

:SYSTem:VERSion?

This query is used to read the SCPI version to which the instrument complies.

Command Type: Query only
Query
Syntax; :SYSTem:VERSion?
Response: 1992.0 (NR2 format)

Security

:SYSTem:SECurity[:STATe] <Boolean>

This command enables the instrument memory to be cleared. The stored settings and the arbitrary waveform memory are cleared when the Security state is changed from ON to OFF. The instrument state is returned to the factory power-on default.

Arguments
Type: Boolean
Command Type: Setting or Query
Setting
Syntax: :SYSTem:SECurity[:STATe]<ws>ON|1|OFF|0
Examples: :SYST:SEC ON
:SYST:SEC OFF
Query
Syntax: :SYSTem:SECurity[:STATe]?
Response: 0|1

Power-on Buffer

:SYSTem:POBuffer <buffer number>

This command is used to set the Power On Buffer setting. The instrument will power-on with the setting stored in that buffer.

Arguments
Type: Numeric
Range: 0 to 49
Rounding: to integer value
Command Type: Setting or Query

Setting
Syntax: :SYSTem:POBuffer<ws><buffer>|MINimum|MAXimum
Example: :SYST:POB 49
Query
Syntax: :SYSTem:POBuffer?[<ws>MINimum|MAXimum]
Response: Power-on buffer in NR1 format

4.14 IEEE 488.1 INTERFACE MESSAGES

GET - Group Execute Trigger

The GET is used by the AWG as a trigger when it is in either the TRIGGER, GATE or BURST modes, with the trigger source set to BUS. It has the same effect as the *TRG common command.

DCL - Device Clear

In response to the DCL, the AWG does the following:

a) Clears the input buffer and the output queue.
b) Resets the Message Processing Functions.

SDC - Selected Device Clear

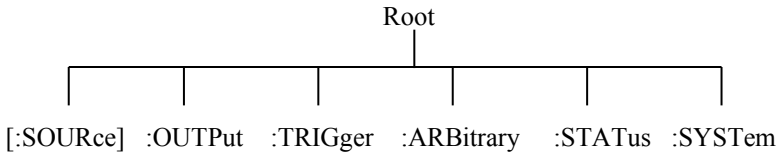
The response is as for the DCL message, when device is addressed to listen.

LLO - Local Lockout

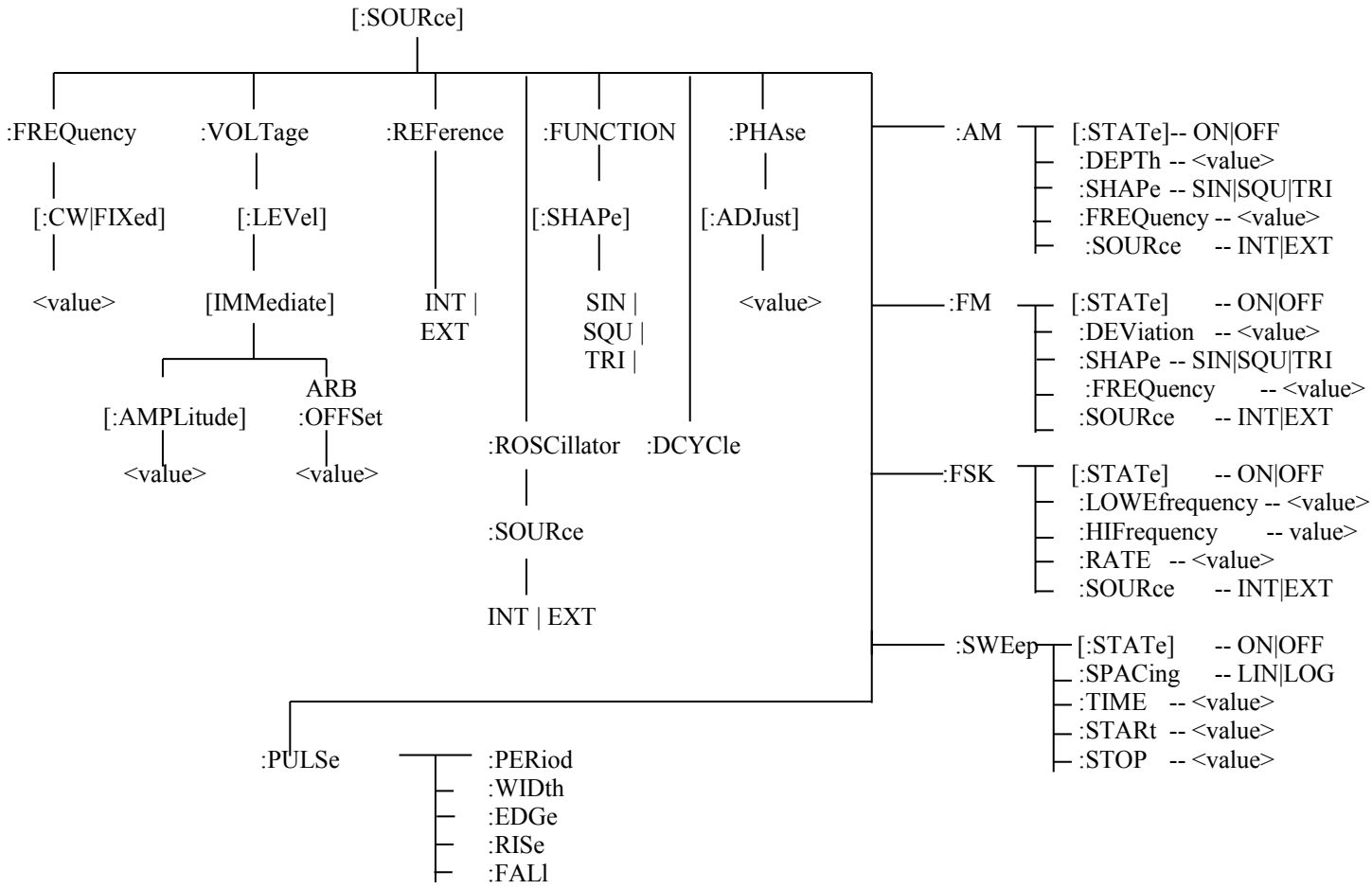
Sending LLO when device is addressed to listen and controller is asserting the REN line will put the device into "Remote with Lock out" state, locking out the front panel.

4.15 SCPI COMMAND TREE

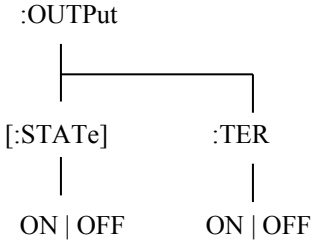
4.15.1 Root Node



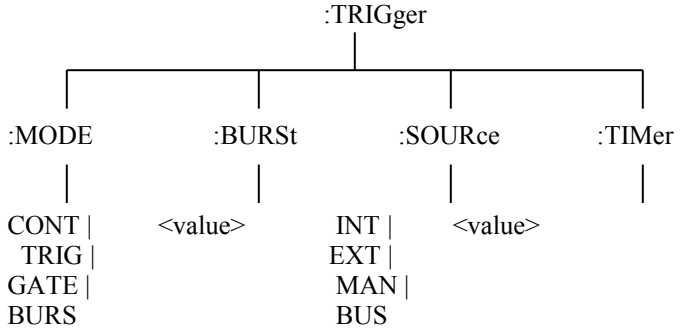
4.15.2 :SOURce Subsystem



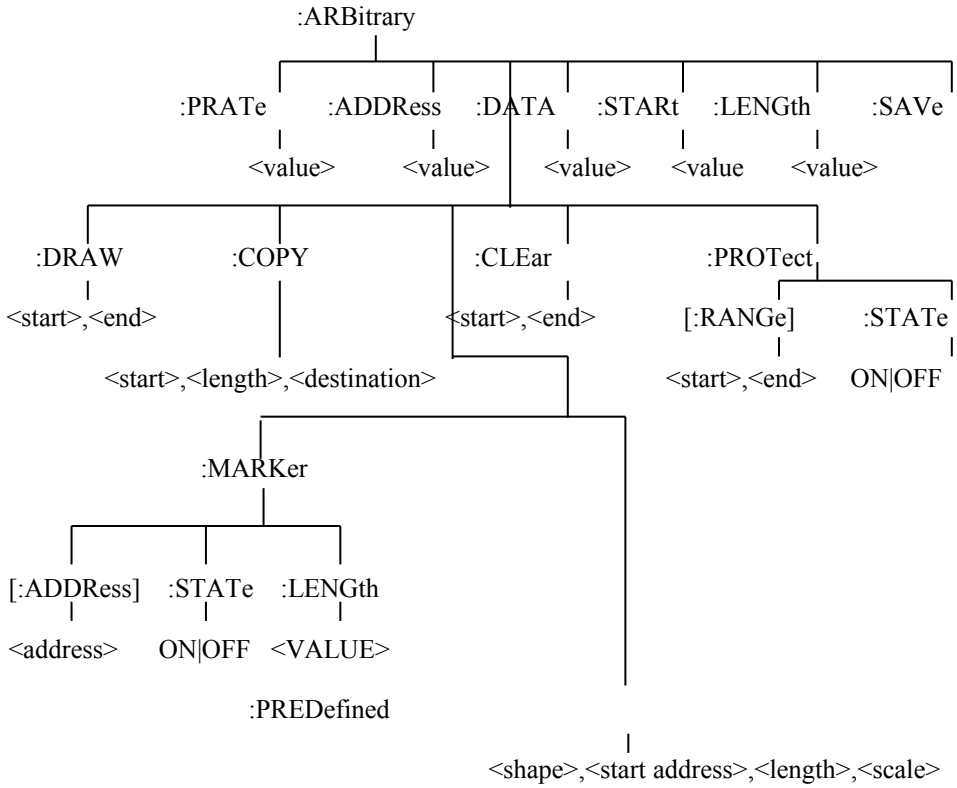
4.15.3 :OUTPut Subsystem



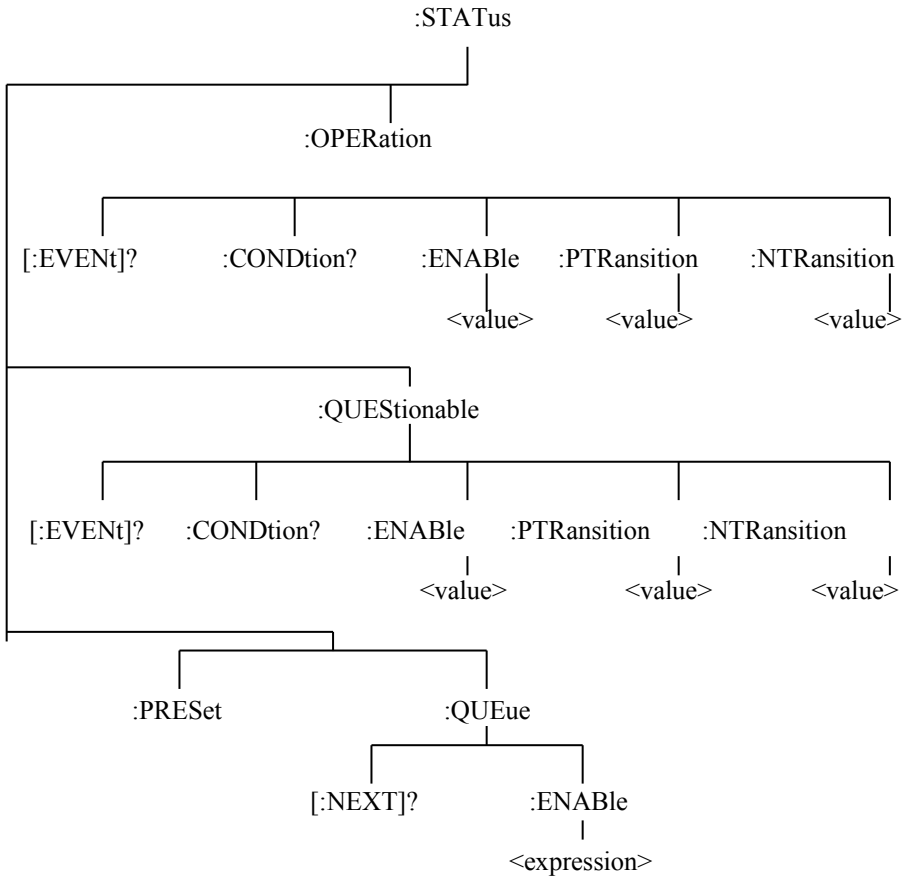
4.15.4 :TRIGger Subsystem



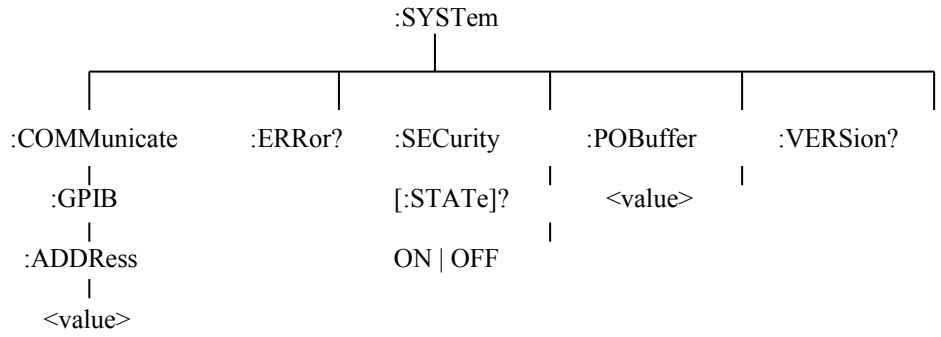
4.15.5 :ARBitrary Subsystem



4.15.6 :STATus Subsystem



4.15.7 :SYSTem Subsystem



ASCII & GPIB CODE CHART

B7 B6 B5 / BITS / B4 B3 B2 B1	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
	CONTROL		NUMBERS SYMBOLS		UPPER CASE		LOWER CASE	
0 0 0 0	0 NUL 0	20 DLE 16	40 SP 32	60 0 48	100 @ 64	120 P 80	140 ` 96	160 p 112
0 0 0 1	1 GTL SOH 1	21 LLO DC1 17	41 ! 33	61 1 49	101 A 65	121 Q 81	141 a 97	161 q 113
0 0 1 0	2 STX 2	22 DC2 18	42 " 34	62 2 50	102 B 66	122 R 82	142 b 98	162 r 114
0 0 1 1	3 ETX 3	23 DC3 19	43 # 35	63 3 51	103 C 67	123 S 83	143 c 99	163 s 115
0 1 0 0	4 SDC EOT 4	24 DCL DC4 20	44 $ 36	64 4 52	104 D 68	124 T 84	144 d 100	164 t 116
0 1 0 1	5 PPC ENQ 5	25 PPU NAK 21	45 % 37	65 5 53	105 E 69	125 U 85	145 e 101	165 u 117
0 1 1 0	6 ACK 6	26 SYN 22	46 & 38	66 6 54	106 F 70	126 V 86	146 f 102	166 v 118
0 1 1 1	7 BEL 7	27 ETB 23	47 ' 39	67 7 55	107 G 71	127 W 87	147 g 103	167 w 119
1 0 0 0	10 GET BS 8	30 SPE CAN 24	50 (40	70 8 56	110 H 72	130 X 88	150 h 104	170 x 120
1 0 0 1	11 TCT HT 9	31 SPD EM 25	51) 41	71 9 57	111 I 73	131 Y 89	151 i 105	171 y 121
1 0 1 0	12 LF 10	32 SUB 26	52 * 42	72 : 58	112 J 74	132 Z 90	152 j 106	172 z 122
1 0 1 1	13 VT 11	33 ESC 27	53 + 43	73 ; 59	113 K 75	133 [91	153 k 107	173 { 123
1 1 0 0	14 FF 12	34 FS 28	54 , 44	74 < 60	114 L 76	134 \ 92	154 l 108	174 | 124
1 1 0 1	15 CR 13	35 GS 29	55 - 45	75 = 61	115 M 77	135] 93	155 m 109	175 } 125
1 1 1 0	16 SO 14	36 RS 30	56 . 46	76 > 62	116 N 78	136 ^ 94	156 n 110	176 ~ 126
1 1 1 1	17 SI 15	37 US 31	57 / 47	77 UNL ? 63	117 O 79	137 UNT _ 95	157 o 111	177 DEL (RUBOUT) 127
	ADDRESSED COMMANDS	UNIVERSAL COMMANDS	LISTEN ADDRESSES		TALK ADDRESSES		SECONDARY ADDRESSES OR COMMANDS (PPE)	(PPD)

octal 25 PPU
NAK
hex 15 21

GPIB code
ASCII character
decimal

4.16 BLOCK TRANSFER

Arbitrary waveform data sent in IEEE488.2 arbitrary block format may take two forms: the definite form and the indefinite form. The essential difference between these forms is that the definite form contains a byte count, while the indefinite form does not. In both cases, the format of the command is

:ARB:DATA<ws><preamble><data><terminator>.

The <data> represents the arbitrary waveform data. This field consists of 8 bit bytes sent in hexadecimal form. Each arbitrary data point consists of two bytes with the high byte being sent first. When sending data in this way, the value of a data point may range from -8191, corresponding to the negative peak, to 8191, corresponding to the positive peak. The value 0 corresponds to zero baseline. Compare the ASCII, or front panel representation, which defines data in the range - 8191to + 8191.

Example: to set a data value to zero, send the Hex bytes

0800

The definite form <preamb4> consists of two fields. The first is a single byte representing the number of digits in the byte count. The byte count is the second field in the preamble, and consists of decimal bytes (0-9), which, when taken together, give the byte count. For example, the preamble

42563

means that the byte count consists of 4 bytes, and that the byte count is 2563 bytes.
The indefinite form preamble consists of a 0 character alone. Both the definite and indefinite forms have a '#' prefix in the preamble.

Another difference is that the indefinite form, since it does not contain a byte count, must be terminated with LF^EOI.
Since each arbitrary data point consists of two bytes, an even number of bytes must be sent. In the following examples, the data is specified in Hex format with each byte being preceded by ' \x', in order to show this.

Example of definite form:

:ARB:DATA #16\x8\x0\x8\x1\x8\x2

Here, the byte count consists of one byte only, and the value is 6.
Example to definite form:

:ARB:DATA #0\x8\x0\x8\x1\x8\x2\x0A

The '\x0A' represents the Linefeed character. EOI must be sent with this character.
Arbitrary data read from the instrument in binary form, as opposed to ASCII, are returned in indefinite form.
Before sending your data enable all errors to be reported, using

:STAT:QUEUE:ENABLE ALL.

Then, if the arb data command causes an error, the error message may be read from the queue using the query.

:SYST:ERR?

4.17 GPIB COMMUNICATION PROTOCOL

4.17.1 General

This appendix describes the effects of interface messages on waveform generator operation and uses abbreviations from the IEEE Standard 488.1-1987.

4.17.2 Responses to IEEE-488.1 Interface Messages

Interface messages and the effects of those messages on the instrument interface functions are defined in IEEE Standard 488.1-1987. Where appropriate, the GPIB code is listed, in decimal.

UNL-Unlisten (63 with ATN)
UNT-Untalk (95 with ATN)

The UNL command places the listener function to its idle (unaddressed) state. In this idle state, the waveform generator cannot accept commands from the GPIB.

The UNT command places the talker function to its idle state. In this idle state, the waveform generator cannot output data through the GPIB.

When the talker and listener functions are idle, the front-panel ADRS indicator is off. If the waveform generator is talk-addressed or listen-addressed, the ADRS indicator lights.

IFC-Interface Clear (Bus pin 9)

The IFC message places both the listener and talk functions to idle states. When the talker and listener functions are idle, the front-panel ADRS indicator is off.

The IFC pulse is >= 100us.

DCL-Device Clear (20 with ATN)

The Device Clear message resets GPIB communication. That is, the DCL message clears all input and output messages, as well as all unexecuted settings.

SDC-Selected Device Clear (4 with ATN)

The SDC message causes the same response as the DCL message. However, the waveform generator only responds if it is listen-addressed.

GET-Group Execute Trigger (8 with ATN)

The waveform generator responds to the Group Execute Trigger message only if it is listen addressed and the device trigger function is enabled. The TRIGger:MODE must be in TRIG, CONTinuous, or BURst and the TRIGger:SOURce must be set to BUS to enable device triggering via GET.

SPE-Serial Poll Enable (24 with ATN)

The SPE message generates output serial poll status bytes when talk-addressed.

SPD-Serial Poll Disable (25 with ATN)

The SPD message switches back to generating output data from the Output Buffer.

MLA-My Listen Address (GPIB Address + 32)
MTA-My Talk Address (GPIB Address + 64)

The instrument GPIB primary address establishes the listen and talk addresses. To see the current GPIB primary address, press SPECIAL and then F1:SYS on the front panel. When the waveform generator is talk-addressed or listen-addressed, the front-panel ADRS indicator lights.

LLO-Local Lockout (17 with ATN)

When the waveform generator is listen addressed, the LLO message changes the waveform generator to the front-panel lockout state.

REN-Remote Enable (GPIB pin 17)

REN is normally held true by the controller and allows the controller to then put the waveform generator into one of the remote states. Pulsing REN false or holding it false forces the device into local state (LOCS).

NOTE

If you disconnect and reconnect the GPIB cable when the controller is holding REN true, the REN goes false and the device got to local state (LOCS).

GTL-Go To Local (1 with ATN)

Listen-addressed instruments respond to GTL by changing to a local state. Remote-to-local transitions caused by GTL do not affect the execution of the message being processed when GTL was received.

Remote-Local Operation

Most front-panel controls cause a transition from REMS to LOCS by asserting a message called return-to-local (rtl). This transition can occur during message execution. However, in contrast to TGL and REN transitions, a transition initiated by rtl affects message execution. In this case, the waveform generator generates an error if there are any unexecuted setting or operational commands.

Front-panel controls that change only the display, do not affect the remote-local states. Only front-panel controls that change settings assert rtl. The rtl message remains asserted when you enter multiple keystroke settings from the front panel, and is unasserted after you execute the settings changes. Since rtl prevents transition to REMS, the waveform generator unasserts rtl if you do not complete a multiple key sequence in a reasonable length of time (about 5 to 10 seconds).

A record of the front-panel settings is in the Current Settings Buffer; however, new settings entered from the front panel or the controller update these recorded settings. In addition, the front panel updates to reflect setting changes

from controller commands. Settings are unaffected by transitions among the 4 remote-local states. The REMOTE indicator lights when the waveform generator is in REMS or RWLS.

Local State (LOCS)

When in a local state (LOCS), you control the settings through the front-panel controls. In addition, only GPIB query commands are executed. All other GPIB commands,setting and operational,prompt and error since those commands are under front-panel (local) control.

NOTE

The waveform generator can be in either Local State (LOCS) or Remote State (REMS) when the it receives the Local Lockout (LLO) interface message. If in LOCS and REN is asserted, the waveform generator enters the Local With Lockout State (LWLS) or, if in REMS, it enters the Remote With Lockout State (RWLS) when it receives LLO. The controller controls the LWLS and RWLS state transitions.

Local Without Lockout State (LWLS)

When the waveform generator is in a local without lockout state (LWLS), it operates the same as it does in LOCS. However, in LWLS rtl does not inhibit a transition to remote state.

Remote State (REMS)

When the waveform generator is in a remote state (REMS), you control its operations from the controller. All settings update when GPIB are executed.

Remote With Lockout State (RWLS)

When in a remote with lockout state (RWLS), the waveform generator operates much the same as it does in LOCS. However, when in RWLS the waveform generator ignores the rtl message, locking out any changes made from the front panel.

4.17.3 IEEE 488.2 Interface Function Subsets

IEEE Standard 488.2 identifies the interface function repertoire of a device on the bus in terms of interface function subsets. These subsets are defined in the standard. Table C-1 lists the subsets that apply to the waveform generator.

NOTE

For more information, refer to IEEE Standard 488.2. The standard is published by the Institute of Electrical and Electronics Engineers, Inc., 345 East 47 Street, New York, New York 10017.

Table C-1: Interface Function Subsets

FUNCTION	SUBSET	CAPABILITY
Source Handshake	SH1	Complete capability
Acceptor Handshake	AH1	Complete capability
Basic Talker	T6	Responds to Serial Poll, Untalk if

		My Listen Address (MLA) is received
Basic Listener	L4	Unlisten if My Talk Address (MTA) is received
Service Request	SR1	Complete capability
Remote-Local	RL1	Complete capability, including Local Lockout (LLO)
Parallel Poll	PP0	Does not respond to Parallel Poll
Device Clear	DC1	Complete capability
Device Trigger	DT1	Complete capability
Controller	C0	No controller functions
Electrical Interface	E2	Three-state drive capability

4.18 USB Programming

4.18.1 General

The INSTALLATION section of this manual describes the USB connection for the instrument. Be sure that you have the Remote Mode set to USB.
The USB standard specifies the electrical characteristics and pin out of the serial bus communication standard for connecting a 'slave' device (your instrument) to a 'master' device (your PC).

4.18.2 USB Operation

The instrument uses the USB-CDC standard, emulating a serial port.

The serial interface implements the same SCPI command set as the GPIB interface. The instrument is programmed by sending ASCII coded characters to the instrument.
When the instrument is in the remote mode remote command input has priority over any front panel control. Therefore, as long as the serial interface is continuously supplied with data, the keyboard will appear to be inoperative to the user.

The instrument accepts a carriage return (CR) as an end of string (EOS) terminator and sends both a CR and LF as the EOS terminator.