Programming Manual

4060B Series
Dual Channel Function/Arbitrary Waveform Generators

bkprecision.com

BK PRECISION

Contents

About Commands & Queries

This section lists and describes the remote control commands and queries recognized by the instrument. All commands and queries can be executed in either local or remote state.

The description, command syntax, query syntax, example and respond can be found in a section. The commands are given in both long and short form. All examples are shown in short form. Queries perform actions such as obtaining information, and are recognized by the question mark (?) following the header.

1.1 How They are Listed

The commands are listed by subsystem and alphabetical order according to their short form.

1.2 How They are Described

In the descriptions themselves, a brief explanation of the function performed is given. This is followed by a presentation of the formal syntax, with the header given in Upper-and-Lower-Case characters and the short form derived from it in ALL UPPER-CASE characters. Where applicable, the syntax of the query is given with the format of its response.

1.3 When can they be used?

The commands and queries listed here can be used for 4050 Series arbitrary/function waveform generators.

1.4 Command Notation

The following notation is used in the commands:

< > Angular brackets enclose words that are used as placeholders, of which there are two types: the header pathand the data parameter of a command.

:= A colon followed by an equals sign separates a placeholder from the description of the type and range of values that may be used in a command instead of the placeholder.

{ } Braces enclose a list of choices, one of which one must be made.

[ ] Square brackets enclose optional items.

… An ellipsis indicates that the items both to its left and right may be repeated a number of times.

Common Command Introduction

IEEE standard defines the common commands used for querying the basic inSyntaxion of the instrument or executing basic operations. These commands usually start with "*" and the length of the keywords of the command is usually 3 characters.

Short	Long	Subsystem	Description
***IDN**	*IDN	SYSTEM	Returns a string that uniquely identifies the instrument.
***OPC**	*OPC	SYSTEM	Generates the OPC message in the standard event status register when all pending overlapped operations have been completed.
***OPC?**	*OPC?	SYSTEM	Returns an ASCII "+1" when all pending overlapped operations have been completed.
***RST**	*RST	SYSTEM	Initiates a device reset.

Table 2.1 Common Commands

2.1 *IDN?

Description The *IDN? query causes the instrument to identify itself. The response comprises manufacturer, model, serial number, software version and firmware version.

Query *IDN?

Response *IDN, <device id>,<model>,<serial number>, <firmware, hardware versions>.

<manufacturer>:=“BK” is used to identify the manufacturer.

<model>:= A model identifier less than 14 characters will contain the model number.

<serial number>:= Number that uniquely identifies the instrument.

<version>:= Contains the firmware version and hardware level.

Example *IDN?

Returns: BK,4063B,*********,2.01.01.35R3B2

2.2 *OPC

Description Sets the **Operation Complete** bit (bit 0) in the Standard Event Register after all of the previous commands have been completed. Other commands may be executed before the bit is set.

This command is used to stops the controller until all pending commands are completed. ***OPC?** returns **"1"** to the output buffer after the previous commands have been completed.

Other commands cannot be executed until this command completes.

Syntax *OPC

Query *OPC?

Example *OPC

Response <bool>

2.3 *RST

Description Reset the instrument to its factory default state. *RST does not affect stored instrument states, or the I/O settings, which are stored in non-volatile memory.

Syntax *RST

Example *RST

Output Commands

3.1 Output

Description Enables or disables the output of the selected channel.

Syntax <channel>:OUTPut <bool>,LOAD,<load>,PLRT, <polarity>
<channel> := {C1,C2}
<bool> := {ON | OFF}
<load> := {50 to 100000, or HZ(high impedance)}
<polarity> := {NOR(normal) | INVT(inverted)}

Query <channel>:OUTPut?

Example C1:OUTP OFF,LOAD,75,PLRT,NOR

Response <string>, C1:OUTP OFF,LOAD,75,PLRT,NOR

3.2 Add Noise

Description Adds the specified signal-to-noise ratio to the output of the selected channel.

Syntax <channel>:NOISE_ADD STATE,<bool>,RATIO,<S/N>
<channel> := {C1,C2}
<bool> := {ON | OFF}
<S/N> := {2.1-100000000}
<S/N (dB)>:= {3.24886-80}

Query <channel>:NOISE_ADD?

Example C1:NOISE_ADD STATE,ON,RATIO,120

Response string, C1:NOISE_ADD STATE,OFF,RATIO,100,RATIO_DB,20dB

Waveform Commands

The waveform commands provide remote access to the parameter configuration of all basic waveforms. Basic waveforms consist of: SINE, SQUARE, RAMP, PULSE, NOISE, ARB, DC

Waveform

Frequency

Period

Amplitude

Offset

Symmetry

Duty

Phase

Standard Deviation

Mean

Width

Rise

Fall

Delay

High Level

Low Level

Bandwidth Switch

Bandwidth Value

Vrms Amplitude

Sampling Rate

Harmonic

Waveform Query

The parameters of the waveforms can be combined to set all parameters at once:
C1:BSWV WVTP,SINE,FRQ,1000HZ,PERI,0.001S,AMP,4V,AMPVRMS, 1.414Vrms, MAX_OUTPUT_AMP,20V,OFST,0V,HLEV,2V,LLEV,-2V,PHSE,0

4.1 Waveform

Description Sets the waveform type.

Syntax <channel>:BaSic_WaVe WVTP,<type>
<type> := {SINE | SQUARE | RAMP | PULSE | NOISE | ARB | DC}

Example C1:BaSic_WaVe WVTP,RAMP

4.2 Frequency

Description Sets the frequency of the waveform.

Syntax <channel>:BaSic_WaVe FRQ,<frequency>
<frequency> := {1μ to 120 MHz}**See table 4.1 for more information.**

	4062B	4063B	4064B
Sine	1 μHz to 40 MHz	1 μHz to 80 MHz	1 μHz to 120 MHz
Square	1 μHz to 25 MHz		
Triangle/Ramp	1 μHz to 1 MHz		
Pulse	1 μHz to 25 MHz		
Gaussian Noise (-3dB)	> 120 MHz		
Arbitrary	1 μHz to 20 MHz		

Table 4.1 Frequency Characteristics

Example C1:BaSic_WaVe FREQ,1000

4.3 Period

Description Sets the period of the waveform. This command is not valied when the waveform type is set to **Noise** or **DC**. The value is set in seconds(s).

Syntax <channel>:BaSic_WaVe PERI,<period>
<period> := {8.3 ns to 1 Ms} **See table 4.1 for more information.**

Example C1:BaSic_WaVe PERI,1e-3

4.4 Amplitude

Description Sets the amplitude of the waveform. This command is not valied when the waveform type is set to **Noise** or **DC**. The value is set in volts peak-to-peak(Vpp).

Syntax <channel>:BaSic_WaVe AMP,<amplitude>
<amplitude> := {2 mVpp to 20 Vpp}

Note:

2 mVpp to 20 Vpp (≤ 20 MHz)
2 mVpp to 10 Vpp (> 20 MHz)
This specification will be divided by 2 while applied to a 50 Ω load.

Example C1:BaSic_WaVe AMP,3

4.5 Offset

Description Sets the offset of the waveform. This command is not valied when the waveform type is set to **Noise**. The value is set in volts (V).

Syntax <channel>:BaSic_WaVe OFST,<offset>
<amplitude> := {± 10 V}

Note:

± 5 V (into 50 Ω load)
± 10 V (into open circuit)

Example C1:BaSic_WaVe OFST,0

4.6 Symmetry

Description Sets the symmetry of the ramp waveform. This command is only valid when the waveform type is set to **RAMP**. The value is set in percentage (%).

Syntax <channel>:BaSic_WaVe SYM,<symmetry>
<symmetry> := {0 to 100%}

Example C1:BaSic_WaVe SYM,50

4.7 Duty

Description Sets the duty cycle of the square waveform. This command is only valid when the waveform type is set to **SUARE** or **Pulse**. The value is set in percentage (%).

Syntax <channel>:BaSic_WaVe DUTY,<duty>
<duty> := {0 to 100%}

Example C1:BaSic_WaVe DUTY,50

4.8 Phase

Description Sets the phase of the waveform. This command is not valid when the waveform type is set to **NOISE**, **Pulse**, **DC**. The value is set in degrees.

Syntax <channel>:BaSic_WaVe PHSE,<phase>
<phase> := {0 to 360}

Example C1:BaSic_WaVe PHSE,90

4.9 Standard Deviation

Description Sets the standard deviation of the waveform. This command is only valid when the waveform type is set to **NOISE**. The value is set in volts(V).

Syntax <channel>:BaSic_WaVe STDEV,<standard deviation>
<standard deviation> := {2mV to 1.15 V}

Example C1:BaSic_WaVe STDEV,0.575

4.10 Mean

Description Sets the mean of the waveform. This command is only valid when the waveform type is set to **NOISE**. The value is set in volts(V).

Syntax <channel>:BaSic_WaVe MEAN,<mean>
<mean> := {± 10 V}

Example C1:BaSic_WaVe MEAN,0

4.11 Width

Description Sets the width of the pulse waveform. This command is only valid when the waveform type is set to **PULSE**. The value is set in seconds (S).

Syntax <channel>:BaSic_WaVe WIDTH,<width>
<width> := {16.3 ns to 1 Ms}

Example C1:BaSic_WaVe WIDTH,1e6

4.12 Rise

Description Sets the rise of the pulse waveform (10 to 90%). This command is only valid when the waveform type is set to **PULSE**. The value is set in seconds (S).

Syntax <channel>:BaSic_WaVe RISE,<rise>
<rise> := {8.4 ns to 122.4 s}

Note:

10% to 90%, 1 Vpp, 50Ω Load, Subject to pulse width limits.

Example C1:BaSic_WaVe RISE,8.4e-9

4.13 Fall

Description Sets the fall of the pulse waveform (10 to 90%). This command is only valid when the waveform type is set to **PULSE**. The value is set in seconds (S).

Syntax <channel>:BaSic_WaVe FALL,<fall>
<fall> := {8.4 ns to 122.4 s}

Note:

10% to 90%, 1 Vpp, 50Ω Load, Subject to pulse width limits.

Example C1:BaSic_WaVe FALL,8.4e-9

4.14 Delay

Description Sets the delay of the pulse waveform. This command is only valid when the waveform type is set to **PULSE**. The value is set in seconds (S).

Syntax <channel>:BaSic_WaVe DLY,<delay>
<delay> := {-1 Ms to 1 Ms}

Example C1:BaSic_WaVe DLY,8.4e-9

4.15 High Level

Description Sets the high level of the pulse waveform. This command is not valid when the waveform type is set to **NOISE** or **DC**. The value is set in volts (V).

Syntax <channel>:BaSic_WaVe HLEV,<high level>
<high level> := {-9.998 to 10 V}

Example C1:BaSic_WaVe HLEV,-9.998

4.16 Low Level

Description Sets the low level of the pulse waveform. This command is not valid when the waveform type is set to **NOISE** or **DC**. The value is set in volts (V).

Syntax <channel>:BaSic_WaVe LLEV,<low level>
<low level> := {-10 V to 9.998 }

Example C1:BaSic_WaVe LLEV,-9.998

4.17 Bandwidth Switch

Description Sets the bandwidth switch of the noise waveform. This command is only valid when the waveform type is set to **NOISE**.

Syntax <channel>:BaSic_WaVe BANDSTATE,<bandwidth switch>
<bandwidth switch> := {ON | OFF }

Example C1:BaSic_WaVe BANDSTATE,ON

4.18 Bandwidth Value

Description Sets the bandwidth value of the noise waveform. This command is only valid when the waveform type is set to **NOISE**. The value is set in mega hertz (MHz).

Syntax <channel>:BaSic_WaVe BANDWIDTH,<bandwidth value>
<bandwidth value> := {20 MHz to 120 MHz}

Example C1:BaSic_WaVe BANDWIDTH,ON

4.19 Vrms Amplitude

Description Sets the amplitude value of waveform. The value is set in volts (Vrms).

Syntax <channel>:BaSic_WaVe AMPVRMS,<amplitude>
<amplitude> := {.7 mVrms to 7.07 Vrms}

Note:

.7 mVrms to 7.07 Vrms(≤ 20 MHz)
.7 mVrms to 3.5355 Vrms (> 20 MHz)
This specification will be divided by 2 while applied to a 50 Ω load.

Example C1:BaSic_WaVe AMPVRMS,0.707e-3

4.20 Sampling Rate

Description Writes or reads the sampling rate and interpolation method for the **True ARB** mode.

Synxtax <channel>:SampleRATE MODE,TARB,VALUE,<sample rate>,
<sample rate> := {0 to 70 MSa/s}

Query <channel>:SampleRATE?

Example C1:SRATE MODE,TARB,VALUE,75e6
C1:SRATE?

Response C1:SRATE MODE,TARB,VALUE,75000000Sa/s

4.21 Harmonic

Description Writes or reads the harmonic parameters. This command is only valid when the basic waveform is **SINE**.

Synxtax <channel>:HARMonic HARMSTATE,<bool>,HARMTYPE,<type>,HARMORDER,<order>,<unit>,<va
HARMPHASE,<phase>
<bool > := {ON mid OFF}
<type> := {EVEN | ODD | ALL}
<order> := {1 | 2 |... | M}, where M is the supported maximum order.
<unit> := {HARMAMP| HARMDBC}
<value> := amplitude of specified harmonic. The range of valid values depends on the model. When <unit>= HARMAMP, the unit is volts, peak-to-peak “Vpp”, and when <unit>= HARMDBC, the unit is "dBc".
<phase> := {0 to 360}, the unit is "degree"

Query <channel>:HARMonic

Example C1:HARM HARMORDER,2,HARMDBC,-6
C1:HARM?

Response C1:HARM HARMSTATE,ON,HARMTYPE,EVEN,HARMORDER,2,HARMAMP,0V,
HARMDBC,-80dBc,HARMPHASE,0

4.22 Waveform Query

Description Reads all the parameters of the set basic waveform.

Query <channel>:BaSic_WaVe?

Example C1:WVTP?

Response <string>, C1:BSWV WVTP,RAMP,FRQ,1000HZ,PERI,0.001S,AMP,1.9999V,AMPVRMS,
0.706965Vrms,AMPDBM,9.99825dBm,MAX_OUTPUT_AMP,20V,OFST,0V,HLEV,
0.99995V,LLEV,-0.99995V,PHSE,0

Modulation

The modulation commands provide remote access to all the modulation parameters.

Modulation State

Modulation Type

Modulation Wave Shape

Modulation Source

Frequency

Modulation Depth

Modulation Deviation

Key Frequency

Hop Frequency

The parameters of the modulating waveform can be combined to set all parameters at once:
C1:MDWV STATE,ON,AM,MDSP,SINE,SRC,INT,FRQ,100HZ,DEPTH,100,CARR, WVTP,SQUARE, FRQ,1000HZ,AMP,4V,AMPVRMS,2Vrms,OFST,0V,PHSE,0,DUTY, 50

5.1 Modulation State

Description Enable/disable modulation. Modulation must be enabled before setting or reading any of the modulation parameters.

Synxtax <channel>:MoDulateWaVe STATE,<bool> <bool> := {ON | OFF}

Example C1:MDWV STATE,ON

5.2 Modulation Type

Description Sets the type of modulation.

Synxtax <channel>:MoDulateWaVe <Type>
<type> := {AM | FM | PM | FSK | ASK | PSK | PWM | DSBAM}

Note:

The carrier of PWM can only be **Pulse**.

Example C1:MDWV AM

5.3 Modulation Wave Shape

Description Sets the shape of the modulating waveform for the selected type of modulation. This function is only available when the modulation source is set to internal (INT).

Synxtax <channel>:MoDulateWaVe <modulation type>,MDSP,<modulation wave shape>
<modulation type> := {AM | FM | PM | DSBAM}
<modulation wave shape> := {SINE | SQUARE | TRIANGLE | UPRAMP | DNRAMP | NOISE | ARB}

Example C1:MDWV AM,MDSP,UPRAMP

5.4 Modulation Source

Description Sets the modulation source.

Synxtax <channel>:MoDulateWaVe <modulation type>,SRC,<source>
<type> := {AM | FM | PM | FSK | ASK | PSK | PWM | DSBAM}
<source> := {INT | EXT}

Example C1:MDWV FM,SRC,INT

5.5 Frequency

Description Sets the frequency of the modulating signal. This command is only valid when the modulation source is set to internal.

Synxtax <channel>:MoDulateWaVe <modulation type>,FRQ,<frequency>
<type> := {AM | FM | PM | FSK | ASK | PSK | PWM | DSBAM}
<frequency> := {1 mHz to 1 MHz}

Example C1:MDWV AM,FRQ,1e3

5.6 Modulation Depth

Description Sets the depth of the amplitude modulation signal. This command is onlty valid when the modulation source is set to internal.

Synxtax <channel>:MoDulateWaVe AM,DEPTH,<depth>
<depth> := {0 to 120%}

Example C1:MDWV

5.7 Modulation Deviation

Description Sets the deviation of the modulating signal. This commands is only valid when the modulation source is set to internal.

Synxtax <channel>:MoDulateWaVe <modulation type>,DEVI,<deviation>
<type> := {FM | PM | PWM}
<deviation> := {See **table 5.1**}

Deviation Ranges	
Frequency Deviation	0 to carrier frequency
Phase Deviation	0 to 360°
Pulse Width Deviation	Depends on the carrier duty cycle

Table 5.1 Deviation Ranges

Example C1:MDWV PM,DEVI,180

5.8 Key Frequency

Description Sets the key frequency of the modulating signal. This commands is only valid when the modulation source is set to internal. The value is in Hz.

Synxtax <channel>:MoDulateWaVe <modulation type>,KFRQ,<key frequency>
<type> := {FSK | ASK | PSK }
<key frequency> := {1m to 1MHz}

Example C1:MDWV ASK,KFRQ,1e3

5.9 Hop Frequency

Description Sets the hop frequency of the modulating signal. This commands is only valid when the modulation source is set to internal. The value is in Hz.

Synxtax <channel>:MoDulateWaVe FSK,HFRQ,<hop frequency>
<hop frequency> := {1m to max rated frequency}

Example C1:MDWV FSK,HFRQ,1e3

5.10 Modulate Waveform Query

Description Reads all parameters of the set modulating waveform. If modulation is disabled the query will return "**C1:MDWV STATE,OFF**".

Query <channel>:MoDulateWaVe?

Example C1:MDWV?

Response C1:MDWV STATE,ON,AM,MDSP,SINE,SRC,INT,FRQ,100HZ,DEPTH,100,CARR,
WVTP,SQUARE,FRQ,1000HZ,AMP,4V,AMPVRMS,2Vrms,OFST,0V,PHSE,0,DUTY, 50

Carrier Waveform

The Carrier waveform commands provide remote access to modify the parameters of the selected carrier waveform when modulation is enabled.

6.1 Carrier Waveform

Description Sets the waveform of the carrier signal.

Synxtax <channel>:MoDulateWaVe CARR,WVTP,<waveform>
<waveform> := {SINE | SQUARE | RAMP | ARB | PULSE}

Example C1:MDWV CARR,WVTP,SINE

6.2 Carrier Frequency

Description Sets the frequency of the carrier signal. The value is in Hz.

Synxtax <channel>:MoDulateWaVe CARR,FRQ,<frequency>
<frequency> := {See **table 6.1**}

Example C1:MDWV CARR,FRQ,1e3

	4062B	4063B	4064B
Sine	1 µHz to 40 MHz	1 µHz to 80 MHz	1 µHz to 120 MHz
Square	1 µHz to 25 MHz		
Triangle/Ramp	1 µHz to 1 MHz		
Pulse	1 µHz to 25 MHz		
Gaussian Noise (-3dB)	> 120 MHz		
Arbitrary	1 µHz to 20 MHz		

Table 6.1 Frequency Characteristics

6.3 Carrier Phase

Description Sets the phase of the carrier signal. The value is in degrees.

Synxtax <channel>:MoDulateWaVe CARR,PHSE,<phase>
<phase> := {0 to 360°}

Example C1:MDWV CARR,PHSE,0

6.4 Carrier Amplitude

Description Sets the amplitude of the carrier signal. The value is in volts, peak-to-peak (Vpp).

Synxtax <channel>:MoDulateWaVe CARR,AMP,<amplitude>
<amplitude> := {2 mVpp to 20 Vpp}

Note:

2 mVpp to 20 Vpp (≤ 20 MHz)
2 mVpp to 10 Vpp (> 20 MHz)
This specification will be divided by 2 while applied to a 50 Ω load.

Example C1:MDWV CARR,AMP,4

6.5 Carrier Offset

Description Sets the offset of the carrier signal. The value is in volts (V).

Synxtax <channel>:MoDulateWaVe CARR,OFST,<offset>
<amplitude> := {± 10 V}

Note:

± 5 V (into 50 Ω load)
± 10 V (into open circuit)

Example C1:MDWV CARR,OFST,1

6.6 Carrier Symmetry

Description Sets the symmetry of the carrier signal. This command is only valid when the carrier is **RAMP**. The value is in percentage.

Synxtax <channel>:MoDulateWaVe CARR,SYM,<symmetry>
<symmetry> := {0 to 100%}

Example C1:MDWV CARR,SYM,50

6.7 Carrier Duty

Description Sets the duty cycle of the carrier signal. This command is only valied when the carrier is **SQUARE** or **PULSE**. The value is in percentage.

Synxtax <channel>:MoDulateWaVe CARR,DUTY,<duty>
<duty> := {0 to 100%}

Example C1:MDWV CARR,DUTY,50

6.8 Carrier Rise

Description Sets the rise time of the carrier signal. This commands is only valid when the carrier is **PULSE**. The value is in seconds (S).

Synxtax <channel>:MoDulateWaVe CARR,RISE,<rise time>
<rise time> := {8.4 ns to 122.4 s}

Note:

10% to 90%, 1 Vpp, 50Ω Load, Subject to pulse width limits.

Example C1:MDWV CARR,RISE,8.4e-9

6.9 Carrier Fall

Description Sets the fall time of the carrier signal. This commands is only valid when the carrier is **PULSE**. The value is in seconds (S).

Synxtax <channel>:MoDulateWaVe CARR,FALL,<fall time>
<fall time> := {8.4 ns to 122.4 s}

Note:

10% to 90%, 1 Vpp, 50Ω Load, Subject to pulse width limits.

Example C1:MDWV CARR,FALL,8.4e-9

6.10 Carrier Delay

Description Sets a delay on the carrier signal. This command is only valid when the carrier is **PULSE**.

Synxtax <channel>:MoDulateWaVe CARR,DLY,<delay>
<delay> := {-1 Ms to 1 Ms}

Example C1:MDWV CARR,DLY,0

Sweep Commands

The sweep commands provide remote access to the parameters of the sweep function.

Sweep State

Sweep Time

Sweep Start

Sweep Stop

Sweep Mode

Sweep Direction

Sweep Trigger Source

Sweep Manual Trigger

Sweep Trigger Mode

Sweep Edge

Mark State

Mark Frequency

Sweep Query

The parameters of the sweep can be combined in one command to set all parameters at once:
C1:SWWV STATE,ON,TIME,1S,STOP,1500HZ,START,500HZ,TRSR,INT,TRMD, OFF,SWMD,LINE, DIR,UP,SYM,5288208.01170765,MARK_STATE,OFF, MARK_FREQ,0HZ,CARR,WVTP,SQUARE, FRQ,1000HZ,AMP,4V,AMPVRMS,2Vrms, OFST,0V,PHSE,0,DUTY,50

7.1 Sweep State

Description Enable/disable the state of the sweep function.

Synxtax <channel>:SweepWaVe STATE,<bool>
<bool> := {ON | OFF}

Example C1:SWWV STATE,ON

7.2 Sweep Time

Description Sets the sweep elapse time. The value is set in seconds (s).

Synxtax <channel>:SweepWaVe TIME,<time>
<time> := {1m to 500 s}

Example C1:SWWV TIME,1

7.3 Sweep Start

Description Sets the start frequency of the sweep. The value is set in hertz (Hz).

Synxtax <channel>:SweepWaVe START,<start frequency>
<start frequency> := {See **table 7.1**}

	4062B	4063B	4064B
Sine	1 µHz to 40 MHz	1 µHz to 80 MHz	1 µHz to 120 MHz
Square	1 µHz to 25 MHz		
Triangle/Ramp	1 µHz to 1 MHz		
Pulse	1 µHz to 25 MHz		
Gaussian Noise (-3dB)	> 120 MHz		
Arbitrary	1 µHz to 20 MHz		

Table 7.1 Sweep Frequency Characteristics

Example C1:SWWV START,1e3

7.4 Sweep Stop

Description Sets the stop frequency of the sweep. The value is set in hertz (Hz).

Synxtax <channel>:SweepWaVe STOP,<stop frequency>
<stop frequency> := {See **table 7.2**}

Example C1:SWWV STOP,10e3

	4062B	4063B	4064B
Sine	1 µHz to 40 MHz	1 µHz to 80 MHz	1 µHz to 120 MHz
Square	1 µHz to 25 MHz		
Triangle/Ramp	1 µHz to 1 MHz		
Pulse	1 µHz to 25 MHz		
Gaussian Noise (-3dB)	> 120 MHz		
Arbitrary	1 µHz to 20 MHz		

Table 7.2 Sweep Frequency Characteristics

7.5 Sweep Mode

Description Sets the sweep mode.

Synxtax <channel>:SweepWaVe SWMD,<mode>
<mode> := {LINE (linear) | LOG}

Example C1:SWWV SWMD,LINE

7.6 Sweep Direction

Description Sets the direction of the sweep.

Synxtax <channel>:SweepWaVe DIR,<direction>
<direction> := {UP | DOWN}

Example C1:SWWV DIR,DOWN

7.7 Sweep Trigger Source

Description Sets the trigger source for the sweep function.

Synxtax <channel>:SweepWaVe TRSR,<trigger source>
<trigger source> := {EXT | INT | MAN}

Example C1:SWWV TRSR,INT

7.8 Sweep Manual Trigger

Description Sends a trigger to manually trigger the sweep. This command is only valid when the **Sweep Trigger Source** is set to **MAN**.

Synxtax <channel>:SweepWaVe MTRIG

Example C1:SWWV MTRIG

7.9 Sweep Trigger Mode

Description Enable/disable the trigger out function of the sweep.

Synxtax <channel>:SweepWaVe TRMD,<mode>
<mode> := {ON | OFF}

Example C1:SWWV TRMD,ON

7.10 Sweep Edge

Description Sets the trigger edge of the trigger. This command is only valid when the **Trigger Source** is **EXT** or **MAN**.

Synxtax <channel>:SweepWaVe EDGE,<trigger edge>
<trigger edge> := {RISE | FALL}

Example C1:SWWV EDGE,FALL

7.11 Mark State

Description Enable/disable the mark state.

Synxtax <channel>:SweepWaVe MARK_STATE,<state>
<state> := {ON | OFF}

Example C1:SWWV MARK_STATE,ON

7.12 Mark Frequency

Description Sets the mark frequency. The value is in hertz (Hz).

Synxtax <channel>:SweepWaVe MARK_FREQ,<frequency>
<frequency> := { See **table 7.3** }

	4062B	4063B	4064B
Sine	1 µHz to 40 MHz	1 µHz to 80 MHz	1 µHz to 120 MHz
Square	1 µHz to 25 MHz		
Triangle/Ramp	1 µHz to 1 MHz		
Pulse	1 µHz to 25 MHz		
Gaussian Noise (-3dB)	> 120 MHz		
Arbitrary	1 µHz to 20 MHz		

Table 7.3 Sweep Frequency Characteristics

Note:

The mark frequency is limited by the sweep's start and stop frequency.

Example C1:SWWV MARK_FREQ,1e3

7.13 Sweep Query

Description Reads all the parameters of the sweep. If sweep is disabled the query will return "**C1:SWWV STATE,OFF**.

Query <channel>:SweepWaVe?

Example C1:SWWV?

Response C1:SWWV STATE,ON,TIME,1S,STOP,1500HZ,START,500HZ,TRSR,INT,TRMD,
OFF,SWMD,LINE,DIR,UP,SYM,5288208.01170765,MARK_STATE,OFF,
MARK_FREQ,0HZ,CARR,WVTP,SQUARE,FRQ,1000HZ,AMP,4V,AMPVRMS,2Vrms,
OFST,0V,PHSE,0,DUTY,50

Burst Commands

This chapter provides the commadns used to remote configure the parameters of the Burst function.

The burst parameters can be combined in one command to set all parameters at once:
C1:BTWV STATE,ON,PRD,0.01S,STPS,0,TRSR,INT,TRMD,OFF,TIME,1,DLAY, 5.76e-07S, GATE_NCYC,NCYC,CARR,WVTP,SQUARE,FRQ,1000HZ,AMP,4V, AMPVRMS,2Vrms,OFST,0V, PHSE,0,DUTY,50

8.1 Busrt State

Description Enable/disable the burst function. Burst must be enabled to read its parameters.

Synxtax <channel>:BursTWave STATE,<bool>
<bool> := {ON | OFF}

Example C1:BTWV STATE,ON

8.2 Burst Period

Description Sets the burst period. The value is entered in seconds (s). This command is not valid when the carrier is **NOISE** or trigger source is **EXT**.

Synxtax <channel>:BursTWave PRD,<period>
<period> := {1μ to 1000s}

Example C1:BTWV PRD,1

8.3 Start Phase

Description Sets the start phaseof the burst. This command is not valid when the carrier is **NOISE**.

Synxtax <channel>:BursTWave STPS,<phase>
<phase> := {0 to 360°}

Example C1:BTWV STPS,0

8.4 Burst Mode

Description Sets the burst mode. This command is not valid when the carrier is**NOISE**.

Synxtax <channel>:BursTWave GATE_NCYC,<mode>
<mode> := {GATE | NCYC}

Example C1:BTWV GATE_NCYC,GATE

8.5 Burst Trigger Source

Description Sets the trigger source of the burst function.

Synxtax <channel>:BursTWave TRSR,<source>
<source> := {EXT | INT | MAN}

Example C1:BTWV TRSR,INT

8.6 Burst Manual Trigger

Description Sends a trigger to trigger burst when the trigger source is **MAN**

Synxtax <channel>:BursTWave MTRIG

Example C1:BTWV MTRIG

8.7 Burst Delay

Description Sets the delay time before the burst starts.The command is valid when Burst Mode is **NCYC**. The command is not valid if the carrier is **NOISE**. The value is entered in secods (s).

Synxtax <channel>:BursTWave DLAY,<delay>
<delay> := { 576n to 100s }

Example C1:BTWV DLAY,1

8.8 Burst Polarity

Description Sets the polarity of the gated signal.

Synxtax <channel>:BursTWave PLRT,<polarity>
<polarity> := {NEG | POS}

Example C1:BTWV PLRT,POS

8.9 Burst Trigger Mode

Description Sets the trigger out mode. This command is valid when **Burst Mode** is **NCYC**. The command is not valid when the carrier is **NOISE**.

Synxtax <channel>:BursTWave TRMD,<trigger out mode>
<trigger out mode> := {RISE | FALL | OFF}

Example C1:BTWV TRMD,RISE

8.10 Burst Edge

Description Sets the trigger edge. This command is valid when **Burst Mode** is **NCYC** and **Burst Trigger Source** is **EXT**. The command is not valid if the carrier is **NOISE**.

Synxtax <channel>:BursTWave EDGE,<edge>
<edge> := {RISE | FALL}

Example C1:BTWV EDGE,FALL

8.11 Burst Cycles

Description Sets the number of Ncycles. This command is valid when **Burst Mode** is **NCYC**. The command is not valid when the carrier is **NOISE**.

Synxtax <channel>:BursTWave TIME,<Ncycles>
<Ncycles> := {INF | 1 to 1M}

Example C1:BTWV TIME,1e6

8.12 Burst Query

Description Reads the set parameters of the burst function.

Query <channel>:BursTWave?

Example C1:BTWV?

Response C1:BTWV STATE,ON,PRD,0.01S,STPS,0,TRSR,INT,TRMD,OFF,TIME,1,DLAY,
5.76e-07S,GATE_NCYC,NCYC,CARR,WVTP,SQUARE,FRQ,1000HZ,
AMP,4V, AMPVRMS,2Vrms,OFST,0V,PHSE,0,DUTY,50

Arbitrary Wave Commands

The arbitrary wave commands provide access to edit, creating, saving, and recalling arbitrary waveforms.

9.1 Recall Arbitrary Waveforms

Description Recalls the specified build in waveform. The waveform is called by stating the memory index and the waveform's name.

Synxtax <channel>:MoDulateWaVe ArbWaVe INDEX,<index>,NAME,<name>
<index> := {2 to 198}
<name> := {See **tables 9.1** and **9.2**}

Query <channel>:ARbWaVe?

Example C1:ARWV INDEX,10,NAME,ExpFal

Response C1:ARWV INDEX,10,NAME,ExpFal

Note:

If the loaded waveform is not saved in the index the instrument will return the default index 2.
C1:ARWV INDEX,2,NAME,StairUp

Index	Name	Index	Name	Index	Name	Index	Name
2	StairUp	22	Lorentz	42	Csc	62	Chebyshev1
3	StairDn	23	Gauspuls	43	Asin	63	Chebyshev2
4	StairUD	24	G onopuls	44	Acos	64	TV
5	Ppulse	25	Tripuls	45	Atan	65	Voice
6	Npulse	26	Cardiac	46	Acot	66	Surge
7	Trapezia	27	Quake	47	Square	67	Radar
8	Upra p	28	Chirp	48	SineTra	68	Ripple
9	Dnra p	29	Twotone	49	SineVer	69	Ga a
10	ExpFal	30	SNR	50	A pALT	70	StepResp
11	ExpRise	31	Ha ing	51	AttALT	71	BandLi ited
12	LogFall	32	Hanning	52	RoundHalf	72	CPulse
13	LogRise	33	kaiser	53	RoundsP	73	CWPulse
14	Sqrtl	34	Black an	54	BlaseiWave	74	GateVibr
15	Root3	35	Gausswin	55	Da pedOsc	75	LF Pulse
16	X^2	36	Triangle	56	SwingOsc	76	CNoise
17	X^3	37	Black anH	57	Discharge	77	A
18	Sinc	38	Bartlett-Hann	58	Pahcur	78	F
19	Gaussian	39	Tan	59	Co bin	79	PF
20	Dlorentz	40	Cot	60	SCR	80	P
21	Haversine	41	Sec	61	Butterworth	81	PW

Table 9.1 Built-In ARB Waveforms

Index	Name	Index	Name	Index	Name	Index	Name
82	EOG	112	ErfcInv	141	ParzenWin	170	Duty52
83	EEG	113	ErfInv	142	TaylorWin	171	Duty54
84	EG	114	Laguerre	143	TukeyWin	172	Duty56
85	Pulseilogra	115	Legend	144	Duty01	173	Duty58
86	ResSpeed	116	Versiera	145	Duty02	174	Duty60
87	ECG1	117	Weibull	146	Duty04	175	Duty62
88	ECG2	118	LogNoral	147	Duty06	176	Duty64
89	ECG3	119	Laplace	148	Duty08	177	Duty66
90	ECG4	120	axwell	149	Duty10	178	Duty68
91	ECG5	121	Rayleigh	150	Duty12	179	Duty70
92	ECG6	122	Cauchy	151	Duty14	180	Duty72
93	ECG7	123	CosH	152	Duty16	181	Duty74
94	ECG8	124	CosInt	153	Duty18	182	Duty76
95	ECG9	125	CotH	154	Duty20	183	Duty78
96	ECG10	126	CscH	155	Duty22	184	Duty80
97	ECG11	127	SecH	156	Duty24	185	Duty82
98	ECG12	128	SinH	157	Duty26	186	Duty84
99	ECG13	129	SinInt	158	Duty28	187	Duty86
100	ECG14	130	TanH	159	Duty30	188	Duty88
101	ECG15	131	ACosH	160	Duty32	189	Duty90
102	LFPulse	132	ASecH	161	Duty34	190	Duty92
103	Tens1	133	ASinH	162	Duty36	191	Duty94
104	Tens2	134	ATanH	163	Duty38	192	Duty96
105	Tens3	135	ACsch	164	Duty40	193	Duty98
106	Airy	136	ACoth	165	Duty42	194	Duty99
107	Besselj	137	Bartlett	166	Duty44	195	deo1_375
108	Bessely	138	BohanWin	167	Duty46	196	deo1_16k
109	Dirichlet	139	ChebWin	168	Duty48	197	deo2_3k
110	Erf	140	FlattopWin	169	Duty50	198	deo2_16k
111	Erfc						

Table 9.2 Built-In ARB Waveforms Cont.

9.2 Read Stored Waveforms

Description Reads the stored waveforms list. The query returns the indexes and names of the waveforms.

Query STL?

Response STL M10, ExpFal, M100, ECG14, M101, ECG15, M102, LFPulse, M103, Tens1, M104, Tens2, M105, Tens3, M106, Airy, M107, Besselj, M108, Bessely, M109, Dirichlet, M11, ExpRise, M110, Erf, M111, Erfc, M112, ErfcInv, M113, ErfInv, M114, Laguerre, M115, Legend, M116, Versiera, M117, Weibull, M118, LogNormal, M119, Laplace, M12, LogFall, M120, Maxwell, M121, Rayleigh, M122, Cauchy, M123, CosH, M124, CosInt, M125, CotH, M126, CscH, M127, SecH, M128, SinH, M129, SinInt, M13, LogRise, M130, TanH, M131, ACosH, M132, ASecH, M133, ASinH, M134, ATanH, M135, ACsch, M136, ACoth, M137, Bartlett, M138, BohmanWin, M139, ChebWin, M14, Sqrt, M140, FlattopWin, M141, ParzenWin, M142, TaylorWin, M143, TukeyWin, M144, Duty01, M145, Duty02, M146, Duty04, M147, Duty06, M148, Duty08, M149, Duty10, M15, Root3, M150, Duty12, M151, Duty14, M152, Duty16, M153, Duty18, M154, Duty20, M155, Duty22, M156, Duty24, M157, Duty26, M158, Duty28, M159, Duty30, M16, X^2, M160, Duty32, M161, Duty34, M162, Duty36, M163, Duty38, M164, Duty40, M165, Duty42, M166, Duty44, M167, Duty46, M168, Duty48, M169, Duty50, M17, X^3, M170, Duty52, M171, Duty54, M172, Duty56, M173, Duty58, M174, Duty60, M175, Duty62, M176, Duty64, M177, Duty66, M178, Duty68, M179, Duty70, M18, Sinc, M180, Duty72, M181, Duty74, M182, Duty76, M183, Duty78, M184, Duty80, M185, Duty82, M186, Duty84, M187, Duty86, M188, Duty88, M189, Duty90, M19, Gaussian, M190, Duty92, M191, Duty94, M192, Duty96, M193, Duty98, M194, Duty99, M195, demo1_375, M196, demo1_16k, M197, demo2_3k, M198, demo2_16k, M2, StairUp, M20, Dlorentz, M21, Haversine, M22, Lorentz, M23, Gauspuls, M24, Gmonopuls, M25, Tripuls, M26, Cardiac, M27, Quake, M28, Chirp, M29, Twotone, M3, StairDn, M30, SNR, M31, Hamming, M32, Hanning, M33, kaiser, M34, Blackman, M35, Gausswin, M36, Triangle, M37, BlackmanH, M38, Bartlett-Hann, M39, Tan, M4, StairUD, M40, Cot, M41, Sec, M42, Csc, M43, Asin, M44, Acos, M45, Atan, M46, Acot, M47, Square, M48, SineTra, M49, SineVer, M5, Ppulse, M50, AmpALT, M51, AttALT, M52, RoundHalf, M53, RoundsPM, M54, BlaseiWave, M55, DampedOsc, M56, SwingOsc, M57, Discharge, M58, Pahcur, M59, Combin, M6, Npulse, M60, SCR, M61, Butterworth, M62, Chebyshev1, M63, Chebyshev2, M64, TV, M65, Voice, M66, Surge, M67, Radar, M68, Ripple, M69, Gamma, M7, Trapezia, M70, StepResp, M71, BandLimited, M72, CPulse, M73, CWPulse, M74, GateVibr, M75, LFMPulse, M76, MCNoise, M77, AM, M78, FM, M79, PFM, M8, Upramp, M80, PM, M81, PWM, M82, EOG, M83, EEG, M84, EMG, M85, Pulseilogram, M86, ResSpeed, M87, ECG1, M88, ECG2, M89, ECG3, M9, Dnramp, M90, ECG4, M91, ECG5, M92, ECG6, M93, ECG7, M94, ECG8, M95, ECG9, M96, ECG10, M97, ECG11, M98, ECG12, M99, ECG13

9.3 Read Stored User Waveforms

Description Reads the user defined waveforms stored in internal memory. If no user waveforms have been stored the instrument will return the string **"EMPTY"**.

Query STL? USER

Response STL WVNM,ARB1,wave1,wave5,wave6,wave8

9.4 Arbitrary Waveform Data

Description Writes and read the data of the specified arbritary waveform. The data must be sent and read in hexadecimal with the little endian Byte order.

To determine the amplitude of each point use the following equation:

$$Output\ \ Voltage = \frac{Amplitudeppsetting \quad \times \quad datapointvalue}{65,536} + Offset$$

65,536 is used since the 4060B series is a 16 bit waveform generator.

This command has two formats; one which edits/creates waveforms in the **Built-in** memory, and user defined waveforms sotred in intrenal memory**Stored Waveforms**.

Synxtax **Index Waveforms:**
<channel>:WVDT POS,<index>WVNM,<name>,TYPE,<type>,LENGTH,<length>,
FREQ,<frequency>,AMPL,<amplitude>,OFST,<offset>,PHASE,<phase>,WAVEDATA,<data>

User Defined Waveforms:
<channel>:WVDT WVNM,<name>,TYPE,5,LENGTH,<length>,FREQ,<frequency>,
AMPL,<amplitude>,OFST,<offset>,PHASE,<phase>,WAVEDATA,<data>

Query **Index Waveforms:**
WVDT? M<index>

User Defined Waveforms:
WVDT? USER,<name>

Example C1:WVDT WVNM,ARB1,TYPE,5,LENGTH,10B,FREQ,1000.000000,AMPL,2.000,OFST,
0.000,PHASE,0.0,WAVEDATA,1F40232827102AF82EE032C836B0008000FF0E10

Response WVDT POS, /Local, WVNM, ARB1, LENGTH, 20B, TYPE, 6, WAVEDATA,@ 1F (# 10’ F8* E0. C82 B06 80 00 FF 00 10 0E

Note:

If the returned string is read in ASCII + Hexadecimal there may be some confusion as to which values are being read as ASCII and which are Hexadeximal. It is recommend the returned data be read in Hexadecimal.

Parameters	Description
index	:=index in **Build-in** memory.(See **tables 9.1** and **9.2**)
name	:= waveform name (See **tables 9.1** and **9.2** for Index Waveforms)
type	:= 0 to 5 0 - common 1 - math 2 - engineering 3 - window 4 - triangle function 5 - user defined
length	4B to 16MB
frequency	:= frequency. The unit is Hertz “Hz”.
amplitude	:= amplitude. The unit is volts, peak-to-peak “Vpp”.
offset	:= offset. The unit is volts “V”.
phase	:= phase. The unit is "degree"
wavedata	:= waveform data. The data must be in hexadecimal

Table 9.3 Sweep Frequency Characteristics

System Commands

The system commands provide remote access to configure all settings under the **System** menu.

10.1 Number Format

Description This command writes and read the number format. The number format consist of the delimiter and the seperator.

Synxtax NumBer_ForMat PNT,<delimiter>,SEPT,<separator>
<delimiter> := {Dot | Comma}
<separator> := {SPACE | OFF | ON}

Query NumBer_ForMat?

Example NBFM PNT,DOT,SEPT,SPACE
NBFM?

Response NBFM PNT,DOT,SEPT,SPACE

10.2 Language

Description This command writes and read the system language. The instrument supports English and Chinese.

Synxtax LAnGuaGe <language>
<language> := {EN | CH }

Query LAnGuagGe?

Example LAGG EN
LAGG?

Response LAGG EN

10.3 Power-On Settings

Description This command writes and readsthe power-on system settings. The system supports default, loading all default values at power on, and last, loading the settings before the last power down.

Synxtax Sys_CFG <mode>
<mode> := {DEFAULT | LAST}

Query Sys_CFG?

Example SCFG LAST
SCFG?

Response SCFG DEFAULT

10.4 Buzzer

Description This command writes and reads the state of the buzzer.

Synxtax BUZZer <state>
<state> := {ON | OFF}

Query BUZZer?

Example BUZZ ON
BUZZ?

Response BUZZ OFF

10.5 Display Sleep Timer

Description This command writes and reads the the display sleep timer state.

Synxtax SCreen_SaVe <state>
<state> := {OFF | 1 | 5 | 15 | 30 | 60 | 120 |300}

Query SCreen_SaVe?

Example SCSV OFF
SCSV?

Response SCSV OFF

Counter Commands

The counter commands provide remote access to configure the settings under the counter menu.

Frequency Counter State

Frequency Counter Reference Frequency

Frequency Counter Trigger

Frequency Counter Mode

High Frequency Rejection State

Frequency Counter Query

The commands can be joint to set all parameters at once:
FCNT STATE,ON,REFQ,2e+08HZ,TRG,1.5V,MODE,AC,HFR,OFF

11.1 Frequency Counter State

Description Sets the state of the frequency counter.

Synxtax FreqCouNTer STATE,<bool>
<bool> := {ON | OFF}

Example FCNT STATE,ON

11.2 Frequency Counter Reference Frequency

Description Sets the reference frequency of the frequency counter. This value is in hertz (Hz).

Synxtax FreqCouNTer REFQ,<reference frequency>
<reference frequency> := {10 Hz to 200 Mhz}

Example FCNT REFQ,1e3

11.3 Frequency Counter Trigger

Description Sets the triger level of the frequency counter. The value is in volts (V).

Synxtax FreqCouNTer TRG,<trigger level>
<trigger level> := {-3.000 to 1.500 V}

Example FCNT TRG 0.000

11.4 Frequency Counter Mode

Description Sets the coupling mode of the frequency counter.

Synxtax FreqCouNTer MODE,<mode>
<mode> := {AC | DC}

Example FCNT MODE,DC

11.5 High Frequency Rejection State

Description Sets high frequency rejection state.

Synxtax FreqCouNTer HFR,<bool>
<bool> := {ON | OFF}

Example FCNT HFR,ON

11.6 Frequency Counter Query

Description Reads the set parameters of the frequency counter. Frequency counter must be enabled, if disabled "**FCNT STATE,OFF**" will be returned.

Query FreqCouNTer?

Example FCNT?

Response FCNT STATE,ON,FRQ,0HZ,DUTY,0,REFQ,2e+08HZ,TRG,1.5V,PW,0S,NW,0S,
FRQDEV,0ppm,MODE,AC,HFR,OFF

Coupling Commands

The coupling commands provide remote access to configure all parameters under the channel copy/coupling menu.

Channel Tracking

Frequency Coupling

Frequency Coupling Deviation

Phase Coupling

Phase Coupling Deviation

Amplitude Coupling

Amplitude Coupling Deviation

The commands can be joint to set all parameters at once:
COUP TRACE,OFF,FCOUP,ON,PCOUP,ON,ACOUP,ON,FRAT,1e-06,PRAT,100, ARAT,1000

12.1 Channel Tracking

Description Enables/disables channel tracking.If tracking is enabled all waveform parameters will track will track each other. Therefore coupling of individual parameters will not be available.

Synxtax COUPling TRACE,<bool>
<bool> := {ON | OFF}

Example COUP TRACE ON

12.2 Frequency Coupling

Description Enables/disables frequency coupling.

Synxtax COUPling FCOUP,<bool>
<bool> := {ON | OFF}

Example COUP FCOUP,ON

12.3 Frequency Coupling Deviation

Description Sets the frequency deviation between the 2 channels. Coupling can be set in a ratio or by a specific deviation. The value is set in hertz (Hz).

Synxtax COUPling <coupling mode>,<deviation>
<coupling mode> := {FDEV | FRAT}

<deviation> := {-80 MHz to 80 MHz} for FDEV
{1μ to 1,000,000} for FRAT

Example COUP FDEV 1e3

12.4 Phase Coupling

Description Enables/disables phase coupling.

Synxtax COUPling PCOUP,<bool>
<bool> := {ON | OFF}

Example COUP PCOUP,ON

12.5 Phase Coupling Deviation

Description Sets the phase deviation between the 2 channels. Coupling can be set in a ratio or by a specific deviation. The value is set in degree.

Synxtax COUPling <coupling mode>,<deviation>
<coupling mode> := {PDEV | PRAT}

<deviation> := {-720 to 720°} for PDEV
{10m to 100} for PRAT

Example COUP PRAT 1e-2

12.6 Amplitude Coupling

Description Enables/disables amplitude coupling.

Synxtax COUPling ACOUP,<bool>
<bool> := {ON | OFF}

Example COUP ACOUP,ON

12.7 Amplitude Coupling Deviation

Description Sets the amplitude deviation between the 2 channels. Coupling can be set in a ratio or by a specific deviation. The value is set in peak-to-peak (Vpp).

Synxtax COUPling <coupling mode>,<deviation>
<coupling mode> := {ADEV | ARAT}

<deviation> := {-19.998 to 19.998 Vpp} for ADEV
{1m to 1000 Vpp} for ARAT

Example COUP ADEV 1e-3

12.8 Coupling Query

Description Reads the set copupling parameters. If **Track** is enbabled only the TRACE state will be returned **COUP TRACE,ON**. When **Track** is disabled the coupling states will be returned. The deviation values will only be returned for the enabled coupling parameters.

Query COUPling?

Example COUP?

Response COUP TRACE,OFF,FCOUP,ON,PCOUP,ON,ACOUP,ON,FRAT,1e-06,PDEV,-720,ARAT,1000

Virtual Keys

13.1 Virtual Keys

Description This command is used to simulate presing a key on the front panel.

Synxtax VirtualKEY VALUE,<index>,STATE,1
<index> := {See **table 13.1**}

Name	Index	Name	Index	Name	Index
KB_FUNC1	28	KB_PARAMETER	5	KB_NUMBER_3	51
KB_FUNC2	23	KB_CHANNEL	72	KB_NUMBER_4	52
KB_FUNC3	18	KB_STORE_RECALL	70	KB_NUMBER_5	53
KB_FUNC4	13	KB_POINT	46	KB_NUMBER_6	54
KB_FUNC5	8	KB_LEFT	44	KB_NUMBER_7	55
KB_FUNC6	3	KB_RIGHT	40	KB_NUMBER_8	56
KB_MOD	15	KB_OUTPUT1	153	KB_NUMBER_9	57
KB_SWEEP	16	KB_NEGATIVE	43	KB_OUTPUT2	152
KB_BURST	17	KB_NUMBER_0	48	KB_KNOB_RIGHT	175
KB_WAVES	4	kB_NUMBER_1	49	KB_KNOB_LEFT	177
KB_UTILITY	11	KB_NUMBER_2	50	KB_KNOB_DOWN	176

Table 13.1 Virtual Keys

Example VKEY VALUE,152,STATE,1

LAN Configuration

The LAN configuration commands provide remote access to the LAN parameters.

IP Address

Subnet Mask

Gateway

14.1 IP Address

Description Writes or reads the LAN's IP Address.

Synxtax SYSTem:COMMunicate:LAN:IPADdress "<integer1>.<integer2>.<integer3>.<integer4>"
<integer 1> := {1 to 223}
<integer 2> := {0 to 225}
<integer 3> := {0 to 225}
<integer 4> := {0 to 225}

Query SYSTem:COMMunicate:LAN:IPADdress?

Example SYST:COMM:LAN:IPAD "10.10.11.213"
SYST:COMM:LAN:IPAD?

Response "10.10.11.213"

14.2 Subnet Mask

Description Writes or reads the LAN's subnet mask.

Synxtax SYSTem:COMMunicate:LAN:SMASk "<integer1>.<integer2>.<integer3>.<integer4>"
<integer 1> := {0 to 225}
<integer 2> := {0 to 225}
<integer 3> := {0 to 225}
<integer 4> := {0 to 225}

Query SYSTem:COMMunicate:LAN:SMASk?

Example SYST:COMM:LAN:SMAS "255.255.255.254"
SYST:COMM:LAN:SMAS?

Response "255.255.255.254"

14.3 Gateway

Description Writes or reads the LAN's gateway.

Synxtax SYSTem:COMMunicate:LAN:GATeway "<integer1>.<integer2>.<integer3>.<integer4>"
<integer 1> := {0 to 223}
<integer 2> := {0 to 225}
<integer 3> := {0 to 225}
<integer 4> := {0 to 225}

Query SYSTem:COMMunicate:LAN:GATeway?

Example SYST:COMM:LAN:GAT "10.10.11.254"
SYST:COMM:LAN:GAT?

Response "10.10.11.254"

Sync Commands

The sync commands provide remote access to synchronize channel 1 and 2 as well as multiple devices.

15.1 Channel Sync

Description Sets the synchronization signal.The generator provides Sync output through the [Aux In/Out] connector on the rear panel. When the synchronization is on, the port can output a CMOS signal with the same frequency as basic waveforms (except Noise and DC), arbitrary waveforms, and modulated waveforms (except external modulation).

Synxtax <channel>:SYNC <bool>,TYPE,<type>
<bool> := {ON | OFF}
<type> := {CH1,CH2,MOD_CH1,MOD_CH2}

Query <channel>:SYNC?

Example C1:SYNC ON,TYPE,MOD_CH1

Response C1:SYNC ON,TYPE,MOD_CH1

15.2 Multi-Device Sync

Description This command set up synchronization between two or more instruments and achieve in-phase output. The delay is set in seconds, and is only available for the slave units.

Synxtax CASCADE STATE,<bool>,MODE,<mode>,DELAY,<delay>
<bool> := {ON | OFF}
<mode> := {MASTER | SLAVE}
<delay> := {1 ns to 25 μs}

Query CASCADE?

Example CASCADE STATE,ON,MODE,SLAVE,DELAY,1e-9

Response CASCADE STATE,ON,MODE,SLAVE,DELAY,1e-09s

Utility Commands

The utility commands provide remote access to the settings located in the Utility menu.

16.1 Clock Source

Description Writes and read the clock source. The source can be set to internal or external. Internal provides a10 MHz source. The internal source can also be outputed from the [10 MHz In/Out] connector for other devices.

Synxtax ROSCillator <src>,10MOUT,<bool>
<src> := {INT, EXT}
<bool> := {ON | OFF}

Query ROSCillator?

Example ROSCillator INT,OFF
ROSC?

Response ROSC INT,10MOUT,OFF

16.2 Waveform Polarity

Description Writes or reads the polarity of the specified channel. The waveform's inversion is relative to the offset voltage

Synxtax <channel>:INVerT <state>
<bool> := {ON | OFF}.

Query <channel>:INVerT?

Example C1:INV 0

Response C1:INVT OFF

16.3 Over-Voltage Protection

Description Writes or reads the state of the over-voltage protection.

Synxtax VOLTPRT <bool> <bool>:= ON | OFF

Query VOLTPRT?

Example VOLTPRT ON
VOLTPRT?

Response ON

16.4 Waveform Combining

Description Writes or reads the waveform combining parameters. When the waveforms combining function is enabled, the load of two channels will be set to the same automatically, default using the load value of the currently operated channel.

Synxtax <channel>:CoMBiNe <bool>
<bool> := {ON | OFF}

Query <channel>:CoMBiNe?

Example C1:CMBN ON C1:CMBN?

Response C1:CMBN ON

16.5 Phase Mode

Description Writes or reads the phase mode for the selected channel.
Phase-Locked Mode: When changing the frequency, the DDSs of both channels reset, and the phase deviation between CH1 and CH2 is maintained.

Independent Mode: When changing the frequency, neither channels' DDS resets and the phase deviation between CH1 and CH2 changes at random. When the independent mode is enabled, the phase parameter cannot be modified and the menu Phase is hidden

Synxtax MODE <parameter>
<parameter> := {PHASE-LOCKED | INDEPENDENT}

Query MODE?

Example MODE PHASE-LOCKED
MODE?

Response MODE PHASE-LOCKED

16.6 Parameter Copy

Description Copies the parameters from one channel to the other.

Synxtax ParaCoPy <destination_channel>,<src_channel>
< destination_channel> := {C1 | C2}.
<src_channel> := {C1 | C2}.

Example PACP C2,C1