PROGRAMMING MANUAL

Function/Arbitrary Waveform Generator

MODEL: 4060 Series (4063, 4064, 4065)

Table of Contents

1.1. 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 for each command or query, with syntax and other information, begins on a new page. The name (header) is given in both long and short form, and the subject is indicated as a command or query or both. Queries perform actions such as obtaining information, and are recognized by the question mark (?) following the header.

1.1.1. How they are listed

The descriptions are listed in alphabetical order according to their short form.

1.1.2. 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.

1.1.3. When can they be used?

The commands and queries listed here can be used for 4060 Series arbitrary/function waveform generators.

1.1.4. Command Notation

The following notation is used in the commands:
< > Angular brackets enclose words that are used placeholders, of which there are two types: the header path and 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.

1.2. Table of Commands & Queries

Short	Long Form	Subsystem	What Command/Query dose
*IDN	*IDN	SYSTEM	Get identification from device.
*RST	*RST	SYSTEM	Resets instrument parameters to default values.
*OPC	*OPC	SYSTEM	Get or set the OPC bit (0) in the Event Status Register (ESR).
CHDR	COMM_HEADER		Set the format of return data (Long, short, off)
BSWV	BASIC_WAVE	SIGNAL	Set or get basic wave parameters. Turns on or off channel signal.
ARWV	ARBWAVE	Data SYSTEM	Change arbitrary wave type.
BUZZ	BUZZER	SYSTEM	Set or get buzzer State.
S_CFG	SYSTEM_CONFIG	SYSTEM	Set or get power on initializing parameter way
ROSC	ROSCILLATOR	SIGNAL	Set or get clock source.
MOD	MODULATION	SIGNAL	Set or get modulated wave parameters.
OUTP	OUTPUT	SIGNAL	Set or get output state.
CHCP	CHANNEL_COPY	SIGNAL	Copy parameters from channel one to channel two, or from channel two to channel one.
INVT	INVERT	SIGNAL	Set or get output signal phase state.
SCSV	SCREEN_SAVE	SYSTEM	Set or get screen save State.
SWE	SWEEP	SIGNAL	Set or get sweep wave.
SYNC	SYNC	SIGNAL	Set or get in-phase signal.
BTWV	BURSTWAVE	SIGNAL	Set or get burst wave parameters.
MDWV	MODULATEWAVE	SIGNAL	Set or get modulate wave parameters.
STL	STORE_LIST	SIGNAL	Get the list of store wave.
WVDT	WAVE_DATA	SIGNAL	Get the wave data of store.
VKEY	VIRTUALKEY	SYSTEM	Set the virtual key.

1.3. IEEE 488.2 Common Command Introduction

IEEE standard defines the common commands used for querying the basic information 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.

1.3.1.   CHDR

DESCRIPTION	This Command is used to change query command return format. SHORT parameter is return short format. LONG parameter is return long format. Off is that command header and parameter unit will not return.
COMMAND SYNTAX	Comm_HeaDeR <parameter> <parameter>:= {SHORT,LONG,OFF}
QUERY SYNTAX	Comm_HeaDeR?
RESPONSE FORMAT	SYNC <parameter>
EXAMPLE 1	Set query command format to long. CHDR LONG
EXAMPLE 2	Read query command format. CHDR? Return: COMM_HEADER LONG

1.3.2.   OPC

DESCRIPTION	The *OPC (OPeration Complete) command sets to true the OPC bit (bit 0) in the standard Event Status Register (ESR). The *OPC? query always responds with the ASCII character 1 because the device only responds to the query when the previous command has been entirely executed.
QUERY SYNTAX	*OPC?
RESPONSE FORMAT	*OPC 1

1.3.3.   IDN

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

QUERY SYNTAX   *IDN?

RESPONSE FORMAT   *IDN , <device id>, <model>, <serial number>, <software>, <version>, <firmware version>
<device id> : ="BK Precision" is used to identify instrument.
<model>:= A model identifier less than 14 characters.
<serial number>:= A nine- or 10-digit decimal code .
<software version>:= A serial numbers about software version.
<firmware version>:= two digits giving the major release level followed by a period, then one digit giving the minor release level followed by a period and a single-digit update level (xx.y.z).

EXAMPLE 1   Reads version information.
*IDN?
Return:
*IDN BK Precision, 4065, 00-00-00-13-22, 5.01.01.10R1, 20.2.3.

1.3.4.   RST

DESCRIPTION   The *RST causes the instrument to reset all settings to default values.

SYNTAX   *RST

1.4. Output Command

DESCRIPTION   Enable or disable the output of the [Output] connector at the front panel corresponding to the channel.
The query returns ON or OFF.

COMMAND SYNTAX   <channel>: OUTPut
<channel>:={C1,C2}
:= {a parameter from the table below}

Parameters	Value	Description
ON	---	Turn on channel
OFF	---	Turn off channel
Load	<load>	Value of load {50 (default unit is Ω), HZ (High-Z)}

QUERY SYNTAX   <channel>: OUTPut?

RESPONSE FORMAT   <channel>:OUTPut <load>

EXAMPLE 1   Turns on channel one.
                   C1:OUTP ON

EXAMPLE 2   Reads channel one output state.
                   C1:OUTP?
                   return:
                   C1:OUTP ON,LOAD,HZ

EXAMPLE 3   Set the load to 50Ω
                   C1:OUTP LOAD,50

1.5. Basic Wave Command

DESCRIPTION   Set or get basic wave parameters.

COMMAND SYNTAX   <channel>:BaSicWaVe
                   <channel>:={C1, C2}
                   := {a parameter from the table below}

Parameters	Value	Description
WVTP	<type>	Type of wave
FRQ	<frequency>	Value of frequency. This parameter cannot be set for noise wave.
AMP	<amplifier>	Value of amplifier. This parameter cannot be set for noise wave.
OFST	<offset>	Value of offset. This parameter cannot be set for noise wave.
SYM	<symmetry>	Value of symmetry. This parameter is for ramp wave only.
DUTY	<duty>	Value of duty cycle. Only Pulse and Duty can set this parameter.
PHSE	<phase>	Value of phase. This parameter cannot be set for noise wave.
STDEV	<Stdev>	Value of noise wave Stdev. This parameter can be set for noise wave only.
MEAN	<mean>	Value of noise wave mean. This parameter can be set for noise wave only.

WIDTH	<width>	Value of width. Parameter is valid only when WVTP is PULSE
RISE	<rise>	Value of rise. Parameter is valid only when WVTP is PULSE
FALL	<fall>	Value of fall. Parameter is valid only when WVTP is PULSE
DLY	<delay>	Value of delay. This parameter can be set for pulse wave only.

Note: If the command doesn't set basic wave type, the parameter will set parameters to current device wave type by default.

where:
<type>:={SINE, SQUARE, RAMP, PULSE, NOISE, ARB ,DC}
<frequency>:= { Default unit is "HZ". Minimum value is 1e-6 HZ, maximum value depends on the 4060 model.}
<amplifier>:= {Default unit is "V". Minimum value is 0.001V (50Ω) 0.002(HiZ), Maximum is 10V(50Ω) 20V(HiZ). }
<offset>:= { Default unit is "V". maximum value depends on the amplifier setting.}
<duty>:= {If wave type is square, range is from 20% to 80%. if wave type is pulse, range is from 0.1% to 99.9%}
<symmetry> :={ 0% to 100%}
<phase>:= {-360° to 360°}
<stdev>:= Maximum is .799V, minimum value is .001V. The default unit is "V".
<mean>:= The range depends on Stdev . The default unit is "V".
<delay>:= Maximum is Pulse Period, minimum value is 0.Unit is s.

QUERY SYNTAX
<channel>:BaSicWaVe?
<channel>:={C1, C2}

RESPONSE
<channel>:BSWV <type>, <frequency>, <amplifier>, <offset>, <duty>, <symmetry>, <phase>

EXAMPLE 1
change channel one current wave type to ramp.
C1:BSWV WVTP,RAMP

EXAMPLE 2
Changes current signal frequency of channel one to 2000 Hz.
C1: BSWV FRQ, 2000HZ

EXAMPLE 3
Set current signal amplifier of channel one.
C1: BSWV AMP, 3V

EXAMPLE 4   Reads channel basic wave parameters from device.
                   C1:BSWV?
                   Return:
                   C1: BSWV WVTP,SINE,FRQ,1000,AMP,3,OFST,3,PHSE,0

RELATED CMDS   ARWV, BTWV, CFG, CPL, MDWV, SWWV

1.6. Modulation Wave Command

DESCRIPTION   Set or get modulated wave parameters.

COMMAND SYNTAX   <channel>:MoDulateWaVe
                   <channel>:={C1, C2}
                   := {a parameter from the table below. }

Parameters	Value	Description
STATE	<state>	Enable or disable modulation. Note: if you want to set or read modulating waveform parameters modulation must be enabled.
AM,SRC	<src>	AM signal source.
AM,MDSP	<mod wave shape>	AM modulation wave. Only AM signal source is set to INT.
AM,FRQ	<am frequency>	AM frequency. Only AM signal source is set to INT.
AM,DEPTH	<depth>	AM deep. Only AM signal source is set to INT.
DSBAM,SRC	<src>	DSBAM signal source
DSBAM,MDSP	<mod wave shape>	DSBAM modulation wave. Only AM signal source is set to INT.
DSBAM,FRQ	<dsbam frequency>	DSBAM frequency. Only AM signal source is set to INT.
FM,SRC	<src>	FM signal source
FM, MDSP	<mod wave shape>	FM modulation wave. Only FM signal source is set to INT.
FM,FRQ	<fm frequency>	FM frequency. Only FM signal source is set to INT.
FM,DEVI	<fm frequency offset>	FM frequency offset. Only FM signal source is set to INT.
PM,SRC	<src>	PM signal source
PM,MDSP	<mod wave shape>	PM modulation wave. Only PM signal source is set to INT.
PM,FRQ	<pm frequency>	PM frequency. Only PM signal source is set to INT.
PWM,FRQ	<pwm frequency>	PWM frequency. Only carry

		wave is PULSE wave.
PWM,DEVI	<pwm devi>	Duty cycle deviation. Only carry wave is Pulse Wave.
PWM,MDSP	<mod wave shape>	PWM modulation wave. Only carry wave is PULSE wave.
PWM,SRC	<src>	PWM signal source.
PM,DEVI	<pm phase offset>	PM phase offset. Only PM signal source is set to INT.
ASK,SRC	<src>	ASK signal source.
ASK,KFRQ	<ask key frequency>	ASK key frequency. Only ASK signal source is set to INT.
FSK,KFRQ	<fsk frequency>	FSK frequency. Only FSK signal source is set to INT.
FSK,HFRQ	<fsk hop frequency>	FSK jump frequency
FSK,SRC	<src>	FSK signal source
CARR,WVTP	<wave type>	Value of carrier wave type.
CARR,FRQ	<frequency>	Value of frequency.
CARR,AMP	<amplifier>	Value of amplifier.
CARR,OFST	<offset>	Value of offset.
CARR,SYM	<symmetry>	Value of symmetry.
CARR,DUTY	<duty>	Value of duty cycle. Only Square can set this parameter.
CARR,PHSE	<phase>	Value of phase.
CARR DLY	<delay>	Value of delay.

Note: If Carrier wave is Pulse or Noise you cannot set the modulation waveform. Also, modulation parameters and carrier parameters cannot be combined into a command. They must be sent separately. See example #8 below.

Where: <state>:={ON,OFF}
<src>:= {INT,EXT}
<mod wave shape>:={SINE, SQUARE, TRIANGLE, UPRAMP, DNRAMP, NOISE, ARB}
<am frequency>:= {0.001Hz to 50000Hz}
<depth>:= {0% to 120%}
<fm frequency>:= {0.001Hz to 50000Hz}
<pm frequency> :={ 0.001Hz to 50000Hz}
<pm phase offset>:= {0° to 360°}
<pwm frequency>:= {0.001Hz to 50kHz }
<pwm devi>:= {depends on carry wave duty}
<ask key frequency>:= {0.002Hz to 20000Hz}
<fsk frequency>:= {0.001Hz to 1000000Hz}
<fsk jump frequency>:= { the same as basic wave frequency}
<wave type>:={SINE ,SQUARE, RAMP, ARB, PULSE }

<frequency>:= { Default unit is "HZ". Minimum value is 1e-6 HZ, maximum value depends on the 4060 model.}

<amplifier>:={ Default unit is "V". Minimum value is 0.001V (50Ω) 0.002(HiZ), Maximum is 10V(50Ω) 20V(HiZ). }

<offset>:={ Default unit is "V".}

<duty>:={ If wave type is square, range is from 20% to 80%. If wave type is pulse, range is from 0.1% to 99.9%.}

<symmetry>:={ 0% to 100%}

<delay>:={the maximum value is 2ks}

QUERY SYNTAX	<channel>:MoDulateWaVe? <channel>:={C1, C2}
RESPONSE FORMAT	<channel>:MoDulateWaVe <parameter> <parameter>:={return all parameter of the current modulate wave parameters, including carrier wave.}
EXAMPLE 1	Set channel one modulation type to AM. C1:MDWV AM
EXAMPLE 2	Set modulation shape to AM, and set AM modulating wave shape to sine wave. C1:MDWV AM, MDSP, SINE
EXAMPLE 3	Reads channel one modulate wave parameters that STATE is ON. C1:MDWV? return: C1:MDWV STATE, ON, AM, MDSP, SINE, SRC, INT, FRQ, 100HZ, DEPTH, 100, CARR, WVTP, RAMP, FRQ, 1000HZ, AMP, 4V, OFST, 0V, SYM, 50
EXAMPLE 4	Reads channel one modulate wave parameters that STATE is OFF. C1:MDWV? return: C1:MDWV STATE,OFF
EXAMPLE 5	Set channel one FM frequency to 1000HZ C1:MDWV FM, FRQ, 1000HZ
EXAMPLE 6	Set the Value of channel one carrier wave shape to SINE. C1:MDWV CARR,WVTP,SINE
EXAMPLE 7	Set the Value of channel one carrier wave frequency to 1000HZ. C1:MDWV CARR,FRQ,1000HZ

EXAMPLE 8   Setup a modulated signal with various parameters.
                   C1:MDWV STATE,ON
                   C1:MDWV
                   CARR,WVTP,SQUARE,FRQ,100000HZ,AMP,5V,OFST,2.5V,PHSE,0,D
                   UTY,50
                   C1:MDWV FM,MDSP,TRIANGLE,SRC,INT,FRQ,1000HZ,DEVI,500HZ

RELATED CMDS   ARWV, BTWV, CFG, CPL, SWWV, BSWV

1.7. Sweep Wave Command

DESCRIPTION   Set or get sweep wave parameters.

COMMAND SYNTAX   <channel>:SWeepWaVe
                     <channel>:={C1, C2}
                     := {a parameter from the table below. }

Parameters	Value	Description
STATE	<state>	Turn on or off sweep wave. Note if you want to set or read sweep wave parameters, you must first enable sweep mode.
TIME	<time>	Value of sweep time
STOP	<stop frequency>	Value of stop frequency
START	<start frequency>	Value of start frequency
TRSR	<trigger src>	Trigger source
TRMD	<trigger mode>	Value of trigger output. If TRSR is EXT, the parameter is invalid.
SWMD	<sweep mode >	Sweep way
DIR	<direction>	Sweep direction
EDGE	<edge>	Value of edge. Only TRSR is EXT, the parameter is valid.
MTRIG	<manual trigger>	Make the device once manual trigger. The parameter is valid only when TRSR is set to MAN.
CARR,WVTP	<wave type>	Value of carrier wave type.
CARR,FRQ	<frequency>	Value of frequency.
CARR,AMP	<amplifier>	Value of amplifier.
CARR,OFST	<offset>	Value of offset.
CARR,SYM	<symmetry>	Value of symmetry.
CARR,DUTY	<duty>	Value of duty cycle. Only Square can set this parameter.

CARR,PHSE	<phase>	Value of phase.

Note: If Carrier wave is Pulse or Noise, enabling sweep is not allowed.

where:

state>:= {ON|OFF}

<time>:= {0.001S to 500S}

<stop frequency> :={ the same with basic wave frequency}

<start frequency> :={ the same with basic wave frequency}

<trigger src>:= {EXT,INT,MAN}

<trigger mode>:= {ON,OFF}

<sweep way>:= {LINE,LOG}

<direction>:= {UP,DOWN}

<edge>:={ON, OFF}

<wave type>:={SINE ,SQUARE, RAMP, ARB}

<frequency>:= { Default unit is "HZ". Minimum value is 1e-6 HZ, maximum value depends on the 4060 model.}

<amplifier>:={ Default unit is "V". Minimum value is 0.001V (50Ω) 0.002(HiZ), Maximum is 10V(50Ω) 20V(HiZ). }

<offset>:={ Default unit is "V".}

<duty>:={ 20% to 80%. }

<symmetry>:={ 0% to 100%}

QUERY SYNTAX   <channel>:SWeepWaVe?

<channel>:={C1, C2}

RESPONSE FORMAT   <parameter>:={return all parameter of the current sweep wave parameters.}

EXAMPLE 1   Set channel one sweep time to 1 s.

C1:SWWV TIME, 1S

EXAMPLE 2   Set channel one sweep stop frequency to 1000Hz.

C1: SWWV STOP, 1000HZ

EXAMPLE 3   Reads channel one modulate wave parameters that STATE is ON.

C2:SWWV?

Return:

C2:SWWV STATE, ON, TIME, 1S, STOP, 100HZ, START, 100HZ, TRSR, MAN, TRMD, OFF, SWMD, LINE, DIR, UP, CARR, WVTP, SQUARE, FRQ, 1000HZ, AMP, 4V, OFST, 0V, DUTY, 50

EXAMPLE 4   Reads channel two modulate wave parameters that STATE is OFF.

C2:SWWV?

Return:

C2:SWWV STATE,OFF

1.8. Burst Wave Command

DESCRIPTION   Set or get burst wave parameters.

COMMAND SYNTAX   <channel>:BursTWaVe
                     <channel>:={C1, C2}
                     := {a parameter from the table below.}

Parameters	Value	Description
STATE	<state>	Enable or disable burst wave. Note if you want to set or read burst wave parameters you must first enable burst mode.
PRD	<period>	When carrier wave is NOISE wave, this cannot be set. When GATE is selected, you cannot set this. This can be set only when trig source is IN (internal).
STPS	<start phase>	When carrier wave is NOISE or PULSE wave, you can't set it.
GATE_NCYC	<gate ncycle>	When carrier wave is NOISE, you can't set it.
TRSR	<trigger>	When carrier wave is NOISE wave, you can't set it. When NCYC was chosen you can set it.
DLAY	<delay>	When carrier wave is NOISE wave, you can't set it. When NCYC was chosen you can't set it.
PLRT	<polarity>	When GATE was chosen you can set it. When carrier wave is NOISE, it is the only parameter.
TRMD	<trig mode>	When carrier wave is NOISE wave, you can't set it. When NCYC was chosen you can set it. When TRSR is set to EXT, you can't set is.
EDGE	<edge>	When carrier wave is NOISE wave, you can't set it. When NCYC is selected and TRSR is set to EXT, you can set it.
TIME	<circle time>	When carrier wave is NOISE wave, you can't set it. When NCYC is selected, you can set it.
MTRIG		When TRSR's parameter be chosen to MAN, that it can be set.
CARR,WVTP	<wave type>	Value of carrier wave type.
CARR,FRQ	<frequency>	Value of frequency.
CARR,AMP	<amplifier>	Value of amplifier.

CARR,OFST	<offset>	Value of offset.
CARR,SYM	<symmetry>	Value of symmetry.
CARR,DUTY	<duty>	Value of duty cycle. Only Square can set this parameter.
CARR,PHSE	<phase>	Value of phase.
CARR,DLY	<carr delay>	Value of carrier wave delay. This is valid only when the carrier wave is pulse.
CARR VAR	<stdev>	Value of carrier wave stdev. This is valid only when the carrier wave is noise.
CARR MEAN	<mean>	Value of carrier wave mean. This is valid only when the carrier wave is noise.

where:

<state>:= {ON,OFF}

<period>:= { Default unit is "S ". 1us to 500s }

<start phase>:= {0 to 360}

<gate ncycle>:= {GATE,NCYC}

<trigger>:= {EXT,INT,MAN}

<delay>:= { Default unit is "S". 0s to 500s }

polarity>:= {NEG,POS}

<trig mode >:= {RISE,FALL,OFF}

<edge>:= { RISE,FALL}

<circle time> :={ 1cycle to 1000000 cycle}

<wave type>:={SINE ,SQUARE, RAMP,PULSE,NOISE, ARB}

<frequency>:= { Default unit is "HZ". Minimum value is 1e-6 HZ, maximum value depends on the 4060 model.}

<amplifier>:={ Default unit is "V". Minimum value is 0.001V (50Ω) 0.002(HiZ), Maximum is 10V(50Ω) 20V(HiZ). }

<offset>:={ Default unit is "V}

<duty>:={ If wave type is Square, range is from 20% to 80%. If wave type is pulse, range is from 0.1% to 99.9%}

<symmetry>:={ 0% to 100%}

<carr delay>:= {Maximum is Pulse Period, minimum valve is 0. Unit is S.}

<stdev>:= Maximum is .799V, minimum value is .001V. The default unit is "V".

<mean>:= The range depends on Stdev . The default unit is "V".

QUERY SYNTAX   <channel>:BursTWaVe?
:=<period>

RESPONSE FORMAT   <channel>:BursTWaVe <type>|<state>|<period>

EXAMPLE 1   Set channel one burst wave period to 1S.
C1:BTWV PRD, 1S

EXAMPLE 2   Set channel one burst wave delay to 0S
C1:BTWV DLAY, 0S

EXAMPLE 3   Reads channel two burst wave parameters that STATE is ON.
C2: BTWV?
Return:
C2:BTWV STATE, ON, PRD, 0.01S, STPS, 0, TRSR, INT, TRMD, OFF, TIME, 1, DLAY, 2.4e 07S, GATE_NCYC, NCYC, CARR, WVTP, SINE, FRQ, 1000HZ, AMP, 4V, OFST, 0V, PHSE, 0

EXAMPLE 4   Reads channel two modulate wave parameters that STATE is OFF.
C2: BTWV?
Return:
C2: BTWV STATE,OFF

1.9. Parameter Copy Command

DESCRIPTION   Copy channel data.

COMMAND SYNTAX PAraCoPy <destination channel>, <src channel>
<destination channel>:= {C1, C2}
<src channel>:= {C1, C2}
Note: The parameters C1 and C2 must be set to device together. C1 is destination channel, C2 is source channel.

EXAMPLE 1   Copy parameters from channel one to channel two.
PACP C2,C1

RELATED CMDS ARWV, BTWV, CFG, CPL, MDWV, SWWV, BSWV

1.10. Arbitrary Wave Command

DESCRIPTION   Change arbitrary wave type.

COMMAND SYNTAX <channel>:ARbWaVe {INDEX, NAME}
<channel>:={C1, C2}
<index>: 0 to 67 (see table below for index information.)
<name>: see table below.

Index	Name	Index	Name	Index	Name	Index	Name
0	StairUp	10	Sinc	20	SNR	30	Sec
1	StairDn	11	Gaussian	21	Hamming	31	Csc

2	StarUD	12	Dlorentz	22	Hanning	32	Asin
3	Trapezia	13	Haversine	23	Kaiser	33	Acos
4	ExpFall	14	Lorentz	24	Blackman	34	Atan
5	ExpRise	15	Gauspuls	25	GaussiWin	35	Acot
6	LogFall	16	Gmonopuls	26	Harris	36~59	(16K Point Waveform Mem)
7	LogRise	17	Cardiac	27	Bartlett		
8	Sqrt	18	Quake	28	Tan	60~67	(512K Point Waveform Mem)
9	X^2	19	TwoTone	29	Cot		

QUERY SYNTAX		\<channel>:ARbWaVe?	
		\<channel>:={C1, C2}	
RESPONSE FORMAT		\<channel>:ARbWaVe \<index>	
EXAMPLE 1		Set StairUD arbitrary wave output by index.	
		ARWV INDEX, 2	
EXAMPLE 2		Reads system current wave.	
		ARWV?	
		Return:	
		ARWV INDEX,2,NAME,StairUD	
EXAMPLE 3		Set Atan arbitrary wave output by name.	
		ARWV NAME, ATAN	
RELATED CMDS		BSWV	

1.11. Phase Command

DESCRIPTION	Set or get phase parameters.
COMMAND SYNTAX	INVerT \<parameter>
	\<parameter>:= {OFF, ON}
QUERY SYNTAX	INVerT?
RESPONSE FORMAT	INVERT \<parameter>
EXAMPLE 1	Set current channel load to invert.
	INVT ON
EXAMPLE 2	Set current channel load to invert.
	INVT OFF
EXAMPLE 3	Set channel 2 load to invert.

C2: INV ON

EXAMPLE 4   Set channel 1 load to normal.
                   C1: INV OFF

1.12. Sync Command

DESCRIPTION   Set signal output from backward panel in phase with forward.

COMMAND SYNTAX   <channel>: SYNC
                   <channel>:={C1,C2}
                   := {ON,OFF}

QUERY SYNTAX   <channel>:SYNC?

RESPONSE FORMAT   <channel>:SYNC

EXAMPLE 1   Sync function on defend of channel one
                   C1:SYNC ON

EXAMPLE 2   Reads channel one sync state.
                   C1:SYNC?
                   Return:
                   C1:SYNC OFF\n

1.13. Configuration Command

DESCRIPTION   Changes system load data of power on.

COMMAND SYNTAX   Sys_CFG
                   := {DEFAULT,LAST}

QUERY SYNTAX   Sys_CFG?

RESPONSE FORMAT   Sys_CFG

EXAMPLE 1   Set system load data of power on to last time data.
                   SCFG LAST

1.14. Buzzer Command

DESCRIPTION   Turns on or off buzzer.

COMMAND SYNTAX   BUZZer
                 := {ON,OFF}

QUERY SYNTAX   BUZZer?

RESPONSE FORMAT BUZZer

EXAMPLE 1   Turns on buzzer.
                 BUZZ ON

1.15. Screen Saver Command

DESCRIPTION   Turns on or off Screen Saver.

COMMAND SYNTAX   SCreen_SaVe
                 := {OFF,1,5,15,30,60,120,300, Unit is minute}

QUERY SYNTAX   SCreen_SaVe?

RESPONSE FORMAT SCreen_SaVe

EXAMPLE 1   Set screen saver time 5 minutes.
                 SCSV 5

1.16. Clock Source Command

DESCRIPTION   Set or get signal oscillator resource .

COMMAND SYNTAX   ROSCillator
                 := {INT, EXT }

QUERY SYNTAX   ROSCillator?

RESPONSE FORMAT ROSC

EXAMPLE 1   Uses system clock source.
                 ROSC INT

1.17. Frequency Counter

DESCRIPTION   Set or get frequency counter.

COMMAND SYNTAX   FreqCouNTer {TRG,<value>,MODE, <value>,HFR,<value>,DEF ,<value>}

<value> = {see table below.}

Parameters	Value	Description
STATE	<state>	Turn on or off frequency counter
FRQ	<frequency>	Input signal frequency.
DUTY	<duty>	Input signal duty.
TRG	<trig level>	Input signal trig level.
PW	<positive width>	Input signal positive width.
NW	<negative width>	Input signal negative width.
MODE	<mode>	Frequency counter mode.
DEF	<default>	Set configuration to default.
HFR	<hfr>	Turn HFR on or off

Note: To use this function, you must turn on the frequency counter. You can only set the mode, trigger level, def and hfr from the above list. The rest of the parameters are for query only.

where: <state>:= {ON|OFF}
                   <frequency>:= {Input signal frequency.}
                   <duty>:={ Input signal duty.}
                   <trig level>:= { Input signal trig level. 1.8V Maximum}
                   <positive width>:= { Input signal positive width.}
                   <negative width>:= { Input signal negative width.}
                   <mode>:={AC|DC}
                   <default>:= { Set configuration to default.}
                   <hfr>:= {ON|OFF}

QUERY SYNTAX   FreqCouNTer? {FRQ, DUTY, TRG, PW, NW, MODE, HFR}

RESPONSE FORMAT   FreqCouNTer

EXAMPLE 1   set trig level to 1.8v.
                   FCNT TRG, 1.8v

EXAMPLE 2   get signal frequency.
                   FCNT?
                   Return:
                   FCNT STATE, ON, FRQ, 0.01HZ, DUTY, 0, TRG, 0V, PW, 0, NW, 0, MODE, AC, HFR, OFF, FRQ, 0.01HZ\n

1.18. Store List Command

DESCRIPTION   This command is used to read the device wave data name. If the store unit is empty, the command will return "EMPTY" string.

4060 Series Programming Manual

Note: M36~ M67 is user defined memory. The name will return what you defined, if it's not defined, the name will "EMPTY".

QUERY SYNTAX   SToreList?

RESPONSE FORMAT   STL M0, StairUp, M1, StairDn, M2, StairUD, M3, Trapezia, M4, ExpFall, M5, ExpRise, M6, LogFall, M7, LogRise, M8, Sqrt, M9, X^2, M10, Sinc, M11, Gaussian, M12, Dlorentz, M13, Haversine, M14, Lorentz, M15, Gauspuls, M16, Gmonopuls, M17, Cardiac, M18, Quake, M19, TwoTone, M20, SNR, M21, Hamming, M22, Hanning, M23, Kaiser, M24, Blackman, M25, GaussiWin, M26, Harris, M27, Bartlett, M28, Tan, M29, Cot, M30, Sec, M31, Csc, M32, Asin, M33, Acos, M34, Atan, M35, ACot, M36, EMPTY, M37, EMPTY, M38, EMPTY, M39, EMPTY, M40, EMPTY, M41, EMPTY, M42, EMPTY, M43, EMPTY, M44, EMPTY, M45, EMPTY, M46, EMPTY, M47, EMPTY, M48, EMPTY, M49, EMPTY, M50, EMPTY, M51, EMPTY, M52, EMPTY, M53, EMPTY, M54, EMPTY, M55, EMPTY, M56, EMPTY, M57, EMPTY, M58, EMPTY, M59, EMPTY, M60, EMPTY, M61, EMPTY, M62, EMPTY, M63, EMPTY, M64, EMPTY, M65, EMPTY, M66, EMPTY, M67, EMPTY

EXAMPLE 1   Read device memory saved arbitrary data.
                    STL?
                    return:
                    STL M0, StairUp, M1, StairDn, M2, StairUD, M3, Trapezia, M4, ExpFall, M5, ExpRise, M6, LogFall, M7, LogRise, M8, Sqrt, M9, X^2, M10, Sinc, M11, Gaussian, M12, Dlorentz, M13, Haversine, M14, Lorentz, M15, Gauspuls, M16, Gmonopuls, M17, Cardiac, M18, Quake, M19, TwoTone, M20, SNR, M21, Hamming, M22, Hanning, M23, Kaiser, M24, Blackman, M25, GaussiWin, M26, Harris, M27, Bartlett, M28, Tan, M29, Cot, M30, Sec, M31, Csc, M32, Asin, M33, Acos, M34, Atan, M35, ACot, M36, EMPTY, M37, EMPTY, M38, EMPTY, M39, EMPTY, M40, EMPTY, M41, EMPTY, M42, EMPTY, M43, EMPTY, M44, EMPTY, M45, EMPTY, M46, EMPTY, M47, EMPTY, M48, EMPTY, M49, EMPTY, M50, EMPTY, M51, EMPTY, M52, EMPTY, M53, EMPTY, M54, EMPTY, M55, EMPTY, M56, EMPTY, M57, EMPTY, M58, EMPTY, M59, EMPTY, M60, EMPTY, M61, EMPTY, M62, EMPTY, M63, EMPTY, M64, EMPTY, M65, EMPTY, M66, EMPTY, M67, EMPTY

1.19. Get or Send Arbitrary Wave Data Command

DESCRIPTION   This command changes the user defined memory unit arbitrary wave data.

COMMAND SYNTAX   WaVe_DaTa <address>,
<address>:= {M36~M67}
:= {a parameter from the table below. }

Parameters	Value	Description
WVNM	<wavename>	Arbitrary wave name
TYPE	<type>	Arbitrary wave type .Note the value has to be set to 5.
LENGTH	<length>	Arbitrary wave data Length. It must be set to "32KB" or "1024KB"
FREQ	<frequency>	Arbitrary wave frequency.
AMPL	<amplifier>	Value of amplify.
OFST	<offset>	Value of offset.
PHASE	<phase>	Value of phase.
WAVEDATA	<wavedata>	Wave data is a 14-bit signed little-endian 2 byte number. Negative values are in two's complement.

Note: All parameters must be sent in one command. If not, the command will not execute successfully. User memory locations, M36~M59 are for 32KB length only. M60~M67 are for 1024KB length only.

QUERY SYNTAX   WaVe_DaTa

RESPONSE FORMAT   WaVe_DaTa <parameter>?

EXAMPLE 1   Read device memory saved arbitrary data.
WVDT M3?
return:
WVDT POS, M9, WVNM, Trapezia, LENGTH, 32KB, TYPE, 5,
WAVEDATA,\x01\x00\x05\x00\t\x00\r\x00\x11\x00\x15\x00\x19\x00\x1D\x00!\x00%\x00)\x00\x001\x005\x009\x00=\x00A\x00E\x00I\x00M\x00Q\x00U\x00Y\x00]\x00a\x00e\x00i\x00m\x00q\x00u\x00y\x00}\x00\x81\x00\x85\x00\x89\x00\x8D\x00\x91\x00\x95\x00\x99\x00\x9D\x00\xA1\x00\xA5\x00\xA9\x00\xAD\x00\xB1\x00\xB5\x00\xB9\x00\xBD\x00\xC1\x00\xC5\x00\xC9\x00\xCD\x00\xD1\x00\xD5\x00\xD9\x00\xDD\x00\xE1\x00\xE5\x00\xE9\x00\xED\x00\xF1\x00\xF5\x00\xF9\x00\xFD\x00\x01\x01\x05\x01\t\x01\r\x01\x11\x01\x15\x01\x19\x01\x1D\x01!\x01%\x01)\x01\x011\x015\x019\x01=\x01A\x01E\x01I\x01M\x01Q\x01U\x01Y\x01]\x01a\x01e\x01i\x01m\x01q\x01u\x01y\x01}\x01\x81\x01\x85\x01\x89\x01\x8D\x01\x91\x01\x95\x01\x99\x01\x9D\x01\xA1\x01\xA5\x01\xA9\x01\xAD\x01\xB1\x01\xB5\x01\xB9\x01\xBD\x01\xC1\x01\xC5\x01\xC9\x01\xCD\x01\xD1\x01\xD5\x01\xD9\x01\xDD\x01\

xE1\x01\xE5\x01\xE9\x01\xED\x01\xF1\x01\xF5\x01\xF9\x01\xFD\x01\
x01\x02\x05\x02…

EXAMPLE 2   Send arbitrary data to saved memory.

WVDT M37, WVNM, SQUAREWAVE1, TYPE, 5, LENGTH, 32KB,
FREQ, 1000, AMPL, 2, OFST, 0, PHASE, 0, WAVEDATA,
\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF
\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F
\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF
\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F
\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF
\x1F\xFF\x1F\xFF\x1F\…
…
x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\xFF\x1F\
x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00
\x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00
\x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00
\x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00
\x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00
\x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00
\x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00
\x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00
\x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00
\x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00 \x00

Note: The Examples of the wave data above are only partial of the return syntax. The generators are 14 bit, one wave point needs 2 bytes, 32KB can store 16K points and 1024KB can store 512K points.

Follow the steps below to convert data values to signed decimal values received from the generator.

1. Obtain 14-bit signed little-endian 2 byte data.
2. Reverse byte order.
3. Determine if 14 bit value is positive or negative. If positive, skip step 4. If negative, proceed to step 4.
4. Takes two's complement of the value based on 14-bit value.
5. The values obtained should be between -8192 and 8191 (decimal)

Conversion of received Waveform Data Examples:

FF 1F (hex) → 1F FF (hex) → positive 14 bit number → +8191 (decimal)
05 00 (hex) → 00 05 (hex) → positive 14 bit number → +5 (decimal)
FF 3F (hex) → 3F FF (hex) → negative 14 bit number → two's complement → -1 (decimal)
00 20 (hex) → 20 00 (hex) → negative 14 bit number → two's complement → -8192 (decimal)

Note: Hexadecimal values in **bold** is data received from the 4060 generators. They are in little-endian format.

Before sending wave data to the 4060 generators, the user must convert the 14-bit signed 2-byte number into little endian format. Negative values are represented in two's complement form.

The following figure shows a triangle waveform with wave data (not little-endian format) points for reference.
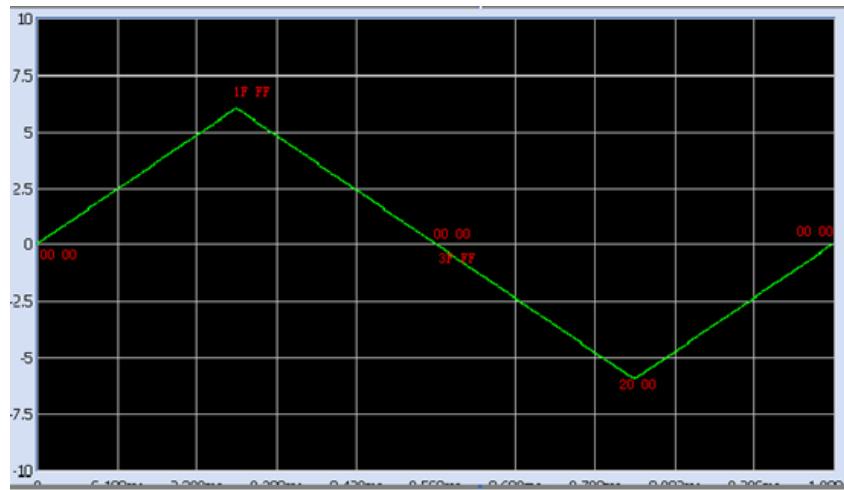
Conversion to send wave data to 4060 generators.
+8191 (decimal) → positive 14 bit number → 1F FF (hex) → **FF 1F** (hex)
0 (decimal) → positive 14 bit number → 00 00 (hex) → **00 00** (hex)
-1 (decimal) → negative 14 bit number → 3F FF (hex) → **FF 3F** (hex)
-8192 (decimal) → negative 14 bit number → 20 00 (hex) → **00 20** (hex)

Note: Hexadecimal values in **bold** is data to be sent to the 4060 generators. They are in little-endian format.

1.20. Virtual Key Command

DESCRIPTION   This sends a virtual key command to the device. The keys are representations of the front panel buttons.

COMMAND SYNTAX   VirtualKEY VALUE, <value>,STATE, 1
                               <value>:= {a parameter from the table below. }

Name	#	Name	#
KB_FUNC1	28	KB_NEGATIVE	43
KB_FUNC2	23	KB_POINT	46

KB_FUNC3	18	KB_WAVES	4
KB_FUNC4	13	KB_PARAMETER	5
KB_FUNC5	8	KB_MOD	15
KB_FUNC6	3	KB_UTILITY	11
KB_NUMBER_0	48	KB_SWEEP	16
KB_NUMBER_1	49	KB_BURST	17
KB_NUMBER_2	50	KB_LEFT	44
KB_NUMBER_3	51	KB_RIGHT	40
KB_NUMBER_4	52	KB_OUTPUT1	153
KB_NUMBER_5	53	KB_OUTPUT2	152
KB_NUMBER_6	54	KB_KNOB_RIGHT	175
KB_NUMBER_7	55	KB_KNOB_DOWN	176
KB_NUMBER_8	56	KB_KNOB_LEFT	177
KB_NUMBER_9	57		

B+K PRECISION®

22820 Savi Ranch Parkway

Yorba Linda, CA 92887

www.bkprecision.com