



**XDG Dual-Channel  
Arbitrary Waveform Generator  
Programmer Manual**

[www.owon.com.cn](http://www.owon.com.cn)

# Representations and Warranties

**Oct. 2018 edition V1.0**

Copyright © LILLIPUT Company. All rights reserved.

The LILLIPUT's products are under the protection of the patent rights, including ones which have already obtained the patent rights and those which are applied for. The information in this manual will replace all materials published.

The information in this manual was correct at the time of printing. However, LILLIPUT will continue to improve products and reserves the rights to change specification at any time without notice.

**owon**<sup>®</sup> is the registered trademark of the LILLIPUT Company.

**Fujian LILLIPUT Optoelectronics Technology Co., Ltd.**

No. 19, Heming Road

Lantian Industrial Zone, Zhangzhou 363005 P.R. China

**Tel:** +86-596-2130430

**Fax:** +86-596-2109272

**Web:** [www.owon.com.cn](http://www.owon.com.cn)

**E-mail:** [info@owon.com.cn](mailto:info@owon.com.cn)

# Table of Contents

Representations and Warranties.....	ii
1. Introduction to the SCPI Language .....	1
1.1. Command Syntax .....	1
1.2. Symbol Description .....	1
1.3. Programmed Parameter Type .....	1
1.4. Command Abbreviation .....	2
2. Commands .....	3
AFGControl:CSCopy (No Query Form).....	3
*CLS (No Query Form).....	3
COUNter:COUPing.....	3
COUNter:DUTYcycle? (Query Only).....	4
COUNter:FREQ? (Query Only).....	4
COUNter:HFR .....	5
COUNter:PERiod? (Query Only) .....	5
COUNter:PULSewidth? (Query Only) .....	6
COUNter:SENSitivity.....	6
COUNter:TRIGger .....	7
DISPlay:BRIGhtness .....	7
DISPlay:SAVer:DELay .....	8
DISPlay:SAVer:IMMEDIATE (No Query Form).....	8
DISPlay:SAVer[:STATe].....	9
HCOPy:SDUMp:DATA? (Query Only) .....	9
HCOPy:SDUMp[:IMMEDIATE] (No Query Form) .....	10
*IDN? (Query Only) .....	10
MMEMory:CATalog? (Query Only).....	11
MMEMory:CDIRectory .....	11
MMEMory:DELete (No Query Form).....	12
OUTPut[1   2]:IMPedance .....	12
OUTPut[1   2][:STATe] .....	13
*RCL (No Query Form) .....	13
*RST (No Query Form) .....	14
*SAV (No Query Form) .....	14
[SOURce[1   2]]:AM[:DEPTh] .....	14
[SOURce[1   2]]:AM:INTernal:FREQUency .....	15
[SOURce[1   2]]:AM:INTernal:FUNCTion .....	16
[SOURce[1   2]]:AM:INTernal:FUNCTion:EFILe.....	16
[SOURce[1   2]]:AM:SOURce .....	17
[SOURce[1   2]]:AM:STATe .....	17
[SOURce[1   2]]:ASKey[:AMPLitude].....	18
[SOURce[1   2]]:ASKey:INTernal:RATE .....	18

[SOURCE[1   2]]:ASKey:SOURce .....	19
[SOURCE[1   2]]:ASKey:STATe .....	20
[SOURCE[1   2]]:BPSKey:DATA.....	20
[SOURCE[1   2]]:BPSKey:INTernal:RATE .....	21
[SOURCE[1   2]]:BPSKey:PHASe .....	21
[SOURCE[1   2]]:BURSt:GATE:POLarity.....	22
[SOURCE[1   2]]:BURSt:INTernal:PERiod.....	22
[SOURCE[1   2]]:BURSt:MODE .....	23
[SOURCE[1   2]]:BURSt:NCYCles.....	23
[SOURCE[1   2]]:BURSt:SOURce .....	24
[SOURCE[1   2]]:BURSt:STATe.....	24
[SOURCE[1   2]]:FM[:DEViation] .....	25
[SOURCE[1   2]]:FM:INTernal:FREQuency.....	26
[SOURCE[1   2]]:FM:INTernal:FUNcTion.....	26
[SOURCE[1   2]]:FM:INTernal:FUNcTion:EFILe .....	27
[SOURCE[1   2]]:FM:SOURce.....	27
[SOURCE[1   2]]:FM:STATe.....	28
[SOURCE[1   2]]:FREQuency:CENTer .....	28
[SOURCE[1   2]]:FREQuency:CONCurrent .....	29
[SOURCE[1   2]]:FREQuency[:FIXed] .....	30
[SOURCE[1   2]]:FREQuency:SPAN .....	30
[SOURCE[1   2]]:FREQuency:STARt .....	31
[SOURCE[1   2]]:FREQuency:STOP .....	31
[SOURCE[1   2]]:FSKey[:FREQuency].....	32
[SOURCE[1   2]]:FSKey:INTernal:RATE.....	33
[SOURCE[1   2]]:FSKey:SOURce.....	33
[SOURCE[1   2]]:FSKey:STATe .....	34
[SOURCE[1   2]]:3FSKey[:FREQuency].....	34
[SOURCE[1   2]]:3FSKey:INTernal:RATE.....	35
[SOURCE[1   2]]:4FSKey[:FREQuency].....	35
[SOURCE[1   2]]:4FSKey:INTernal:RATE.....	36
[SOURCE[1   2]]:FUNcTion:EFILe .....	36
[SOURCE[1   2]]:FUNcTion:RAMP:SYMMetry .....	37
[SOURCE[1   2]]:FUNcTion[:SHAPE] .....	38
[SOURCE[1   2]]:MOD:STATe .....	39
[SOURCE[1   2]]:OSKey:INTernal:RATE .....	39
[SOURCE[1   2]]:OSKey:TIME .....	40
[SOURCE[1   2]]:PHASe[:ADJust].....	40
[SOURCE[1   2]]:PHASe:INITiate (No Query Form) .....	41
[SOURCE[1   2]]:PM[:DEViation].....	42
[SOURCE[1   2]]:PM:INTernal:FREQuency.....	42
[SOURCE[1   2]]:PM:INTernal:FUNcTion .....	43
[SOURCE[1   2]]:PM:INTernal:FUNcTion:EFILe.....	43
[SOURCE[1   2]]:PM:SOURce .....	44

[SOURce[1   2]]:PM:STATe .....	44
[SOURce[1   2]]:PSKey[:DEViation].....	45
[SOURce[1   2]]:PSKey:INTernal:RATE.....	46
[SOURce[1   2]]:PSKey:SOURce .....	46
[SOURce[1   2]]:PSKey:STATe .....	47
[SOURce[1   2]]:PULSe:DCYCLE .....	47
[SOURce[1   2]]:PWM:INTernal:FREQuency .....	48
[SOURce[1   2]]:PWM:INTernal:FUNCTion.....	48
[SOURce[1   2]]:PWM:INTernal:FUNCTion:EFILe .....	49
[SOURce[1   2]]:PWM:SOURce .....	50
[SOURce[1   2]]:PWM:STATe.....	50
[SOURce[1   2]]:PWM[:DEViation]:DCYCLE.....	51
[SOURce[1   2]]:SWEep:SOURce.....	51
[SOURce[1   2]]:SWEep:SPACing.....	52
[SOURce[1   2]]:SWEep:STATe.....	52
[SOURce[1   2]]:SWEep:TIME .....	53
[SOURce[1   2]]:VOLTage[:LEVel][:IMMEDIATE]:OFFSet .....	53
[SOURce[1   2]]:VOLTage[:LEVel][:IMMEDIATE][:AMPLitude] .....	54
SYSTem:BEEPer[:IMMEDIATE] (No Query Form) .....	55
SYSTem:BEEPer:STATe.....	55
SYSTem:ERRor[:NEXT]? (Query Only) .....	56
SYSTem:KLOCK[:STATe].....	56
SYSTem:LANguage.....	57
SYSTem:POWeron.....	57
SYSTem:RESTART .....	58
SYSTem:VERSion? (Query Only).....	58
TRACe DATA:CATalog? (Query Only).....	59
TRACe DATA:COPI (No Query Form) .....	59
TRACe DATA[:DATA] .....	60
TRACe DATA[:DATA]:VALue.....	60
TRACe DATA:POINts.....	61
*TRG (No Query Form).....	61

# 1. Introduction to the SCPI Language

## 1.1. Command Syntax

The command systems present a hierarchy structure (tree system) and each command consists of a "Root" keyword and one or multiple sub-keywords. The keywords are separated by ":" and are followed by the parameter settings available, "?" is added at the end of the command string to indicate query and the command and parameter are separated by "space".

After one of the complete commands, if the following command has the same first-level keyword or the same first-level and second-level keywords (keywords have 3 levels most), the same keywords in the following command can be omit, the command can be written as the rest parts.

## 1.2. Symbol Description

Following symbols are usually used to assist to explain the parameters contained in a command.

a) Braces { }

The options enclosed in a { } are parameters available in the command. Only one option could be selected every time, and all the options are separated by "|". For example, {ON|OFF} indicates that ON or OFF can be selected.

b) Triangle Brackets < >

The parameter enclosed in < > must be replaced by an effective value.

## 1.3. Programmed Parameter Type

The commands contain 8 kinds of parameters, different parameters have different setting methods.

(1) **arbitrary block**

A specified length of arbitrary data, for example,

#6377512xxxxx . . . where

6 indicates that the following 6 digits (377512) specify the length of the data in bytes;

xxxxx ... indicates the data or #0xxxxx...<LF><&EOI>

(2) **boolean**

Boolean numbers or values, for example,

ON or  $\neq$  0

OFF or 0

**(3) discrete**

A list of specific values, for example,  
MIN, MAX

**(4) NR1 numeric**

Integers, for example,  
0, 2, 30, -5

**(5) NR2 numeric**

Decimal numbers, for example,  
0.6, 3.1415926, -2.6

**(6) NR3 numeric**

Floating point numbers, for example,  
3.1415E-7, -8.2E3

**(7) NRf numeric**

Flexible decimal number that may be type NR1, NR2 or NR3  
See NR1, NR2, and NR3 examples

**(8) string**

Alphanumeric characters (must be within quotation marks)  
"Model, 123456"

### 1.4. Command Abbreviation

All the commands are case-insensitive, so you can use any kind of them. But if abbreviation is used, all the capital letters specified in commands must be written completely. For example,

SOURce1:FREQuency:FIXed 500kHz also can be:

SOUR1:FREQ:FIX 500kHz

## 2. Commands

### **AFGControl:CSCopy (No Query Form)**

This command copies setup parameters for one channel to another channel.

#### **Syntax**

AFGControl:CSCopy {CH1|CH2},{CH1|CH2}

#### **Arguments**

CH1|CH2

#### **Examples**

AFGControl:CSCopy CH1,CH2

copies the CH1 setup parameters into CH2.

### **\*CLS (No Query Form)**

This command clears all the event registers and queues, which are used in the instrument status and event reporting system.

#### **Syntax**

\*CLS

#### **Arguments**

None

#### **Examples**

\*CLS

clears all the event registers and queues.

### **COUNter:COUPing**

This command sets or queries the coupling mode of the counter to AC or DC.

#### **Syntax**



COUNter:COUPing [AC|DC]  
COUNter:COUPing?

**Arguments**

AC means that AC is selected for the coupling mode of the counter.  
DC means that DC is selected for the coupling mode of the counter.

**Returns**

AC|DC

**Example**

COUNter:COUPing DC  
sets the coupling mode of the counter to DC.

**COUNter:DUTYcycle? (Query Only)**

This query-only command returns the measurement results for duty cycle of the counter.

**Syntax**

COUNter:DUTYcycle?

**Arguments**

None

**Returns**

<dutycycle>::=<NR3>

**Examples**

:COUNter:DUTYcycle?  
might returns 2.265700368E+01.

**COUNter:FREQ? (Query Only)**

This query-only command returns the measurement results for frequency of the counter.

**Syntax**

COUNter:FREQ?

**Arguments**

None

**Returns**

<frequency>::=<NR3>

**Examples**

:COUNter:FREQ?

might returns 1.000082563E+02.

**COUNter:HFR**

This command enables or disables the high-frequency reject of the counter. The query returns the state of high-frequency reject of the counter.

**Syntax**

COUNter:HFR {ON|OFF|<NR1>}

COUNter:HFR?

**Arguments**

ON or <NR1>≠0 enables the high-frequency reject of the counter.

OFF or <NR1>=0 disables the high-frequency reject of the counter.

**Returns**

<NR1>

**Examples**

COUNter:HFR ON

enables the high-frequency reject of the counter.

**COUNter:PERiod? (Query Only)**

This query-only command returns the measurement results for period of the counter.

**Syntax**

COUNter:PERiod?

**Arguments**

None

**Returns**

<period>::=<NR3>

**Examples**

:COUNter:PERiod?

might returns 8.545000251E-03.

**COUNter:PULSewidth? (Query Only)**

This query-only command returns the measurement results for pulse width of the counter.

**Syntax**

COUNter:PULSewidth?

**Arguments**

None

**Returns**

<pulsewidth>::=<NR3>

**Examples**

:COUNter:PULSewidth?

might returns 366213017E-03.

**COUNter:SENSitivity**

This command sets or queries the trigger sensitivity of the counter.

**Syntax**

COUNter:SENSitivity [LOW|MIDDLE|HIGH]

COUNter:SENSitivity?

**Arguments**

LOW means that Low is selected for the trigger sensitivity of the counter.

MIDDLE means that Middle is selected for the trigger sensitivity of the counter.

HIGH means that High is selected for the trigger sensitivity of the counter.

**Returns**

LOW|MIDDLE|HIGH

**Example**

COUNter:SENSitivity LOW

sets the trigger sensitivity of the counter to Low.

## **COUNTER:TRIGGER**

This command sets or queries the trigger level of the counter.

### **Syntax**

COUNTER:TRIGGER {<value>|MINimum|MAXimum}

COUNTER:TRIGGER? {MINimum|MAXimum}

### **Arguments**

<value>::=<NR3>[<units>]

where:

<NR3> is the trigger level of the counter.

<units>::=[mV|V]

### **Returns**

<value>

### **Examples**

COUNTER:TRIGGER 500mV

sets the trigger level of the counter to 500 mV.

## **DISPLAY:BRIGHTNESS**

This command sets or queries the brightness of the LCD display.

### **Syntax**

DISPLAY:BRIGHTNESS {<brightness>|MINimum|MAXimum}

DISPLAY:BRIGHTNESS?

### **Arguments**

<brightness>::=<NR1> where:

<NR1> is a range of display brightness from 0 through 100. The larger the value, the higher the screen brightness.

MINimum sets the display to the lowest brightness level.

MAXimum sets the display to the highest brightness level.

### **Returns**

<NR1>

### Examples

DISPLAY:BRIGHTNESS 90

sets the display brightness to 90%.

### DISPlay:SAVer:DELay

This command sets or queries delay time for the screen saver function. The setting range is 1 minute to 999 minutes.

#### Syntax

DISPlay:SAVer:DELay {<minutes>|MINimum|MAXimum}

DISPlay:SAVer:DELay? {MINimum|MAXimum}

#### Arguments

<minutes>::=<NR1>[<units>]

where:

<NR1> is the delay time in minutes.

<units>::=MIN

#### Returns

<minutes>

### Examples

DISPlay:SAVer:DELay 30

sets the delay time for the screen saver function to 30 minutes.

### DISPlay:SAVer:IMMEDIATE (No Query Form)

This command sets the screen saver state to ON, regardless of the DISPlay:SAVer[:STATe]? command setting.

The screen saver is enabled immediately (without waiting for the delay time).

#### Syntax

DISPlay:SAVer:IMMEDIATE

#### Arguments

None

### Examples

DISPLAY:SAVER:IMMEDIATE  
sets the screen saver state to ON.

### **DISPlay:SAVer[:STATe]**

This command sets or queries the screen saver setting of the LCD display. When enabled, the screen saver function starts automatically if no operations are applied to the instrument front panel for the delay time set in DISPlay:SAVer:DELAy.

#### **Syntax**

DISPlay:SAVer[:STATe] {ON|OFF|<NR1>}  
DISPlay:SAVer[:STATe]?

#### **Arguments**

ON or <NR1>≠0 enables the screen saver function.  
OFF or <NR1>=0 disables the screen saver function.

#### **Returns**

<NR1>  
indicating the screen saver state.

#### **Examples**

DISPLAY:SAVER[:STATE] OFF  
disables the screen saver function.

### **HCOPy:SDUMp:DATA? (Query Only)**

This query-only command returns a specified length of binary data which consist a screen image.

#### **Syntax**

HCOPy:SDUMp:DATA?

#### **Arguments**

None

#### **Returns**

<data>::=<arbitrary block>

#### **Examples**

HCOPY:SDUMp:DATA?

might return the following response:

#6377512xxxxx . . . where

6 indicates that the following 6 digits (377512) specify the length of the data in bytes;  
xxxxx ... indicates the image data.

### **HCOPY:SDUMp[:IMMEDIATE] (No Query Form)**

This command copies a screen image and saves the image file to a USB memory. The default file name is n.BMP, where n is a consecutive number from 0. The image files are saved in a folder named Model/IMAGE (Model is the instrument model) in the USB memory.

#### **Syntax**

HCOPY:SDUMp[:IMMEDIATE]

#### **Arguments**

None

#### **Examples**

HCOPY:SDUMP[:IMMEDIATE]

copies the screen image and may create a file 1.BMP in a USB memory.

### **\*IDN? (Query Only)**

This query-only command returns identification information on the instrument.

#### **Syntax**

\*IDN?

#### **Arguments**

None

#### **Returns**

<Manufacturer>,<Model>,<Serial Number>,<Firmware Level>

#### **Examples**

\*IDN?

might return the following response:

OWON,XDG3202,1837001,SCPI:99.0 FV:V1.2.0

## **MMEemory:CATalog? (Query Only)**

This query-only command returns the current state of the mass storage system (USB memory).

### **Syntax**

MMEemory:CATalog?

### **Arguments**

None

### **Returns**

<NR1>,<NR1>[,<file\_name>,<file\_type>,<file\_size>]...

where:

The first <NR1> indicates that the total amount of storage currently used, in bytes. The second <NR1> indicates that the free space of mass storage, in bytes.

<file\_name> is the name of directory or file. If the name exceeds 22 characters in length, it will be shortened to 8 characters (without suffix) in 8.3 name format.

<file\_type> is DIR for directory, otherwise it is blank.

<file\_size> is the size of the file, in bytes. This value will be 0 for directory.

### **Examples**

The USB memory includes the Test\_folder folder, a Firmware.upp file, and a memo.txt file.

MMEemory:CATalog? might return the following response:

```
21973685,16851047,"Test_folder,DIR,0","Firmware.upp,,7791","memo.txt,,2566"
```

## **MMEemory:CDIRectory**

This command changes the current working directory in the mass storage system.

### **Syntax**

MMEemory:CDIRectory [<directory\_name>]

MMEemory:CDIRectory?

### **Arguments**

<directory\_name>::=<string> indicates the current working directory for the mass storage system.



**Returns**

<directory\_name>::=<string>

**Examples**

MMEemory:CDIRectory "/Test\_folder/Case"  
changes the current directory to /Test\_folder/Case.

**MMEemory:DELeTe (No Query Form)**

This command deletes a file or directory from the mass storage system. If a specified file in the mass storage is not allowed to overwrite or delete, this command causes an error. You can delete a directory if it is empty.

**Syntax**

MMEemory:DELeTe <file\_name>

**Arguments**

<file\_name>::=<string> specifies a file to be deleted and should include full path.

**Examples**

MMEemory:DELeTe "/Test\_folder/Case/Firmware.upp"  
deletes the Firmware.upp file from the /Test\_folder/Case directory.

**OUTPut[1|2]:IMPedance**

This command sets the output load impedance for the specified channel. The specified value is used for amplitude, offset, and high/low level settings. You can set the impedance to any value from 1  $\Omega$  to 10 k $\Omega$  with a resolution of 1  $\Omega$ . The default value is 50  $\Omega$ . The query returns the current load impedance setting in ohms. If the load impedance is set to INFINITY, the query returns "9.9E+37".

**Syntax**

OUTPut[1|2]:IMPedance {<ohms>|INFINITY|MINimum|MAXimum}  
OUTPut[1|2]:IMPedance? {MINimum|MAXimum}

**Arguments**

<ohms>::=<NR3>[<units>]

where:

<units>::=OHM

INFINITY sets the load impedance to >10 k $\Omega$ .

MINimum sets the load impedance to 1  $\Omega$ .  
MAXimum sets the load impedance to 10 k $\Omega$ .

### Returns

<ohms>::=<NR3>

### Examples

OUTPut1:IMPedance MAXimum  
sets the CH1 load impedance to 10 k $\Omega$ .

## OUTPut[1|2][:STATe]

This command sets or query the instrument output state for the specified channel.

### Syntax

OUTPut[1|2][:STATe] {ON|OFF|<NR1>}  
OUTPut[1|2][:STATe]?

### Arguments

ON or <NR1> $\neq$ 0 enables the instrument output.  
OFF or <NR1>=0 disables the instrument output.

### Returns

<NR1>

### Examples

OUTPut1:STATe ON  
sets the instrument CH1 output to ON.

## \*RCL (No Query Form)

This command restores the state of the instrument from a copy of the settings stored in the setup memory. The settings are stored using the \*SAV command. If the specified setup memory is deleted, this command causes an error.

### Syntax

\*RCL {0|1|2|...|14|15}

### Arguments

0, 1, 2, ..., 14, or 15 specifies the location of setup memory.

### **Examples**

\*RCL 3

restores the instrument from a copy of the settings stored in memory location 3.

### **\*RST (No Query Form)**

This command resets the instrument to the factory default settings.

### **Syntax**

\*RST

### **Arguments**

None

### **Examples**

\*RST

resets the instrument settings to the factory defaults.

### **\*SAV (No Query Form)**

This command stores the current settings of the arbitrary function generator to a specified setup memory location.

### **Syntax**

\*SAV {0|1|2|...|14|15}

### **Arguments**

0, 1, 2, ... 14, or 15 specifies the location of setup memory.

### **Examples**

\*SAV 3

saves the current instrument state in the memory location 3.

### **[SOURce[1|2]]:AM[:DEPTh]**

This command sets or queries the modulation depth of AM modulation for the specified

channel. Set the modulation depth from 0% to 100% with resolution of 1%.

### Syntax

```
[SOURce[1|2]]:AM[:DEPTh] {<depth>|MINimum|MAXimum}  
[SOURce[1|2]]:AM[:DEPTh]? [MINimum|MAXimum]
```

### Arguments

<depth>::=<NR2>[<units>]

where:

<NR2> is the depth of modulating frequency.

<units>::=PCT

MINimum sets the modulation depth to minimum value. MAXimum sets the modulation depth to maximum value.

### Returns

<depth>

### Examples

```
SOURce1:AM:DEPth MAXimum
```

sets the depth of modulating signal on CH1 to the maximum value.

## [SOURce[1|2]]:AM:INTernal:FREQuency

This command sets or queries the internal modulation frequency of AM modulation for the specified channel. Use this command when the internal modulation source is selected.

Set the internal modulation frequency from 2 mHz to 100.00 kHz with resolution of 1 mHz.

### Syntax

```
[SOURce[1|2]]:AM:INTernal:FREQuency {<frequency>|MINimum|MAXimum}  
[SOURce[1|2]]:AM:INTernal:FREQuency? [MINimum|MAXimum]
```

### Arguments

<frequency>::=<NRf>[<units>]

where:

<NRf> is the modulation frequency.

<units>::=[Hz|kHz|MHz]

### Returns

<frequency>

### Examples

```
SOURce1:AM:INTernal:FREQuency 10kHz
```

sets the CH1 internal modulation frequency to 10 kHz.

## **[SOURCE[1|2]]:AM:INTERNAL:FUNCTION**

This command sets or queries the modulating waveform of AM modulation for the specified channel. Use this command when the internal modulation source is selected. If you specify EFILE when there is no EFILE or the EFILE is not yet defined, this command causes an error.

### **Syntax**

```
[SOURCE[1|2]]:AM:INTERNAL:FUNCTION {SINusoid|SQUare|RAMP|PRNoise  
|USER<NR1>|EMEMory|EFILE}  
[SOURCE[1|2]]:AM:INTERNAL:FUNCTION?
```

### **Arguments**

USER<NR1>|EMEMory

<NR1> specifies the user waveform memory location, can be any number from 0 to 15. A user defined waveform saved in the user waveform memory or the EMEMory can be selected as a modulating signal.

EFILE

EFILE is used as a modulating signal.

### **Returns**

SIN|SQU|RAMP|PRN|USER<NR1>|EMEMory|EFILE

### **Examples**

```
SOURCE1:AM:INTERNAL:FUNCTION SQUare
```

selects Square as the shape of modulating waveform for the CH1 output.

## **[SOURCE[1|2]]:AM:INTERNAL:FUNCTION:EFILE**

This command sets or queries an EFILE name used as a modulating waveform for AM modulation. A file name must be specified in the mass storage system. This command returns " " if there is no file in the mass storage.

### **Syntax**

```
[SOURCE[1|2]]:AM:INTERNAL:FUNCTION:EFILE <file_name>  
[SOURCE[1|2]]:AM:INTERNAL:FUNCTION:EFILE?
```

### **Arguments**

<file\_name>::=<string> specifies a file name in the mass storage system. The <file\_name> includes path. Path separators are forward slashes (/).

### Returns

<file\_name>

### Examples

SOURce1:AM:INteRnal:FUNcTion:EFILe "TEST"  
sets a file named "TEST" in the mass storage.

## [SOURce[1|2]]:AM:SOURce

This command sets or queries the source of modulating signal of AM modulation for the specified channel.

### Syntax

[SOURce[1|2]]:AM:SOURce [INteRnal|EXteRnal]  
[SOURce[1|2]]:AM:SOURce?

### Arguments

INteRnal means that the carrier waveform is modulated with an internal source.  
EXteRnal means that the carrier waveform is modulated with an external source.

### Returns

INT|EXT

### Examples

SOURce1:AM:SOURce INteRnal  
sets the CH1 source of modulating signal to internal.

## [SOURce[1|2]]:AM:STATe

This command enables or disables AM modulation for the specified channel. The query returns the state of AM modulation.

### Syntax

[SOURce[1|2]]:AM:STATe {ON|OFF|<NR1>}  
[SOURce[1|2]]:AM:STATe?

### Arguments

If [SOURCE[1|2]] are omitted, CH1 is specified automatically.  
ON or <NR1>≠0 enables AM modulation.  
OFF or <NR1>=0 disables AM modulation.

#### Returns

<NR1>

#### Examples

SOURCE1:AM:STATE ON  
enables the CH1 AM modulation.

### [SOURCE[1|2]]:ASKKey[:AMPLitude]

This command sets or queries the modulation amplitude of ASK modulation for the specified channel. Set the modulation amplitude from 0 Vpp to the current amplitude of the carrier waveform. The amplitude resolution is 1 mVpp or four digits.

#### Syntax

[SOURCE[1|2]]:ASKKey[:AMPLitude] {<amplitude>|MINimum|MAXimum}  
[SOURCE[1|2]]:ASKKey[:AMPLitude]? [MINimum|MAXimum]

#### Arguments

<amplitude>::=<NRf>[<units>]

where:

<NRf> is the modulation amplitude.

<units>::=[Vpp]

MINimum sets the modulation amplitude to minimum value.

MAXimum sets the modulation amplitude to maximum value.

#### Returns

<amplitude>

#### Examples

SOURCE1:ASKKey:AMPLitude MAXimum  
sets the amplitude of modulating signal on CH1 to the maximum value.

### [SOURCE[1|2]]:ASKKey:INTERNAL:RATE

This command sets or queries the internal modulation rate of ASK modulation for the specified channel. Use this command when the internal modulation source is selected.

**Syntax**

[SOURce[1|2]]:ASKey:INTernal:RATE {<rate>|MINimum|MAXimum}  
[SOURce[1|2]]:ASKey:INTernal:RATE? {MINimum|MAXimum}

**Arguments**

<rate>::=<NRf>[<units>]

where:

<NRf> is the modulation rate.

<units>::=[Hz|kHz|MHz]

**Returns**

<rate>

**Examples**

SOURce1:ASKey:INTernal:RATE 50Hz

sets the CH1 internal modulation rate to 50 Hz.

**[SOURce[1|2]]:ASKey:SOURce**

This command sets or queries the source of modulation signal of ASK modulation for the specified channel.

**Syntax**

[SOURce[1|2]]:ASKey:SOURce [INTernal|EXTernal]  
[SOURce[1|2]]:ASKey:SOURce?

**Arguments**

INTernal means that the carrier waveform is modulated with an internal source.

EXTernal means that the carrier waveform is modulated with an external source.

**Returns**

INT|EXT

**Examples**

SOURce1:ASKey:SOURce INTernal

sets the CH1 source of modulating signal to internal.



## **[SOURce[1|2]]:ASKey:STATe**

This command enables or disables ASK modulation. The query returns the state of ASK modulation. Select a sine, square, ramp, or arbitrary waveform as the carrier waveform.

### **Syntax**

```
[SOURce[1|2]]:ASKey:STATe {ON|OFF|<NR1>}  
[SOURce[1|2]]:ASKey:STATe?
```

### **Arguments**

ON or <NR1>≠0 enables ASK modulation.  
OFF or <NR1>=0 disables ASK modulation.

### **Returns**

<NR1>

### **Examples**

```
SOURce1:ASKey:STATe ON  
enables the CH1 ASK modulation.
```

## **[SOURce[1|2]]:BPSKey:DATA**

This command sets or queries the data source of modulation signal of BPSK modulation for the specified channel.

### **Syntax**

```
[SOURce[1|2]]:BPSKey:DATA [01|10|PN15|PN21]  
[SOURce[1|2]]:BPSKey:DATA?
```

### **Arguments**

01 means that the carrier waveform is modulated with 01 pattern.  
10 means that the carrier waveform is modulated with 10 pattern.  
PN15 means that the carrier waveform is modulated with PN15 pattern.  
PN21 means that the carrier waveform is modulated with PN21 pattern.

### **Returns**

01|10|PN15|PN21

### **Examples**

```
SOURce1:BPSKey:DATA 01  
sets the CH1 source of BPSK modulating signal to 01 pattern.
```

## **[SOURce[1|2]]:BPSKey:INTernal:RATE**

This command sets or queries the internal modulation rate of BPSK modulation for the specified channel.

### **Syntax**

[SOURce[1|2]]:BPSKey:INTernal:RATE {<rate>|MINimum|MAXimum}

[SOURce[1|2]]:BPSKey:INTernal:RATE? {MINimum|MAXimum}

### **Arguments**

<rate>::=<NRf>[<units>]

where:

<NRf> is the modulation rate.

<units>::=[Hz|kHz|MHz]

### **Returns**

<rate>

### **Examples**

SOURce1:BPSKey:INTernal:RATE 50Hz

sets the CH1 internal BPSK modulation rate to 50 Hz.

## **[SOURce[1|2]]:BPSKey:PHASe**

This command sets or queries the phase deviation of BPSK modulation for the specified channel.

### **Syntax**

[SOURce[1|2]]:BPSKey:PHASe {<phase>|MINimum|MAXimum}

[SOURce[1|2]]:BPSKey:PHASe? [MINimum|MAXimum]

### **Arguments**

<phase>::=<NR3>[<units>]

where:

<NR3> is the phase deviation.

<units>::=[RAD|DEG]

If <units> are omitted, RAD is specified automatically. The setting ranges are:

RAD: 0 PI to +1 PI, relative to phase value

DEG: 0 to +180, in 1 degree steps, relative to phase value

**Returns**

<phase>

**Examples**

SOURce1:BPSKey:PHASe MAXimum

sets the maximum value for the CH1 phase deviation of BPSK modulation.

**[SOURce[1|2]]:BURSt:GATE:POLarity**

This command sets the generator to output a burst when the gated signal at the [Ext Trig/Burst/Fsk In] connector at the rear panel is high level or low level.

This command is only available in gated Burst mode.

**Syntax**

[SOURce[1|2]]:BURSt:GATE:POLarity{NORMal|INVerted}

[SOURce[1|2]]:BURSt:GATE:POLarity?

**Arguments**

NORMal sets the polarity to Normal.

INVerted sets the polarity to Inverted.

**Returns**

NORM|INV

**Examples**

[SOURce[1|2]]:BURSt:GATE:POLarity INVerted

sets the polarity to INVerted, means the instrument outputs a burst when the gated signal at the [Ext Trig/Burst/Fsk In] connector at the rear panel is low level.

**[SOURce[1|2]]:BURSt:INTernal:PERiod**

This command sets or queries the Burst period for the specified channel.

**Syntax**

[SOURce[1|2]]:BURSt:INTernal:PERiod {<period>|MINimum|MAXimum}

[SOURce[1|2]]:BURSt:INTernal:PERiod?[MINimum|MAXimum]

**Arguments**

<period>::=<NRf>[<units>]

where  
<NRf> is the Burst period.  
<units>::=[ns|μs|ms|s]

### Returns

<period>  
Examples SOURce1:BURSt:INternal:PERiod 200ns  
sets the CH1 Burst period to 200 ns.

## [SOURce[1|2]]:BURSt:MODE

This command sets or queries the burst mode for the specified channel.

### Syntax

[SOURce[1|2]]:BURSt:MODE {TRIGgered|GATed}  
[SOURce[1|2]]:BURSt:MODE?

### Arguments

TRIGgered means that triggered mode is selected for burst mode.  
GATed means that gated mode is selected for burst mode.

### Returns

TRIG|GAT

### Examples

SOURce1:BURSt:MODE TRIGgered  
selects triggered mode.

## [SOURce[1|2]]:BURSt:NCYCles

This command sets or queries the number of cycles (burst count) to be output in burst mode for the specified channel. The query returns 9.9E+37 if the burst count is set to INFINITY.

### Syntax

[SOURce[1|2]]:BURSt:NCYCles {<cycles>|INFINITY|MINimum|MAXimum}  
[SOURce[1|2]]:BURSt:NCYCles? {MINimum|MAXimum}

### Arguments

<cycles>::=<NRf>

where:

<NRf> is the burst count. The burst count ranges from 1 to 500,000.

INFINITY sets the burst count to infinite count.

MINimum sets the burst count to minimum count.

MAXimum sets the burst count to maximum count.

### Returns

<cycles>

### Examples

SOURce1:BURSt:NCYCles 2

sets the CH1 burst count to 2.

## [SOURce[1|2]]:BURSt:SOURce

This command sets or queries the trigger source in the burst mode for the specified channel. This command is available only in the Triggered burst mode.

### Syntax

[SOURce[1|2]]:BURSt:SOURce [TImEr|MANual|EXternal]

[SOURce[1|2]]:BURSt:SOURce?

### Arguments

TImEr specifies an internal clock as the trigger source.

MANual specifies a manual trigger input as the trigger source.

EXternal specifies an external trigger input as the trigger source.

### Returns

TIM|MAN|EXT

### Examples

SOURce1:BURSt:SOURce EXternal

sets an external trigger input as the trigger source in the burst mode.

## [SOURce[1|2]]:BURSt:STATe

This command enables or disables the burst mode for the specified channel. The query returns the state of burst mode.

### Syntax

[SOURce[1|2]]:BURSt:STATe {ON|OFF|<NR1>}  
[SOURce[1|2]]:BURSt:STATe?

### Arguments

ON or <NR1>≠0 enables the burst mode.  
OFF or <NR1>=0 disables the burst mode.

### Returns

<NR1>

### Examples

SOURce1:BURSt:STATe ON  
enables the burst mode for the CH1.

## [SOURce[1|2]]:FM[:DEViation]

This command sets or queries the peak frequency deviation of FM modulation for the specified channel. The setting range of frequency deviation depends on the waveform selected as the carrier.

### Syntax

[SOURce[1|2]]:FM[:DEViation] {<deviation>|MINimum|MAXimum}  
[SOURce[1|2]]:FM[:DEViation]? [MINimum|MAXimum]

### Arguments

<deviation>::=<NRf>[<units>]

where:

<NRf> is the frequency deviation.

<units>::=[Hz|kHz|MHz]

### Returns

<deviation>

### Examples

SOURce1:FM:DEViation 1.0MHz  
sets the CH1 frequency deviation to 1.0 MHz.

## **[SOURce[1|2]]:FM:INTernal:FREQuency**

This command sets or queries the internal modulation frequency of FM modulation for the specified channel. Use this command when the internal modulation source is selected.

Set the internal modulation frequency from 2 mHz to 100.00 kHz with resolution of 1 mHz.

### **Syntax**

[SOURce[1|2]]:FM:INTernal:FREQuency {<frequency>|MINimum|MAXimum}

[SOURce[1|2]]:FM:INTernal:FREQuency? [MINimum|MAXimum]

### **Arguments**

<frequency>::=<NRf>[<units>]

where:

<NRf> is the modulation frequency.

<units>::=[Hz|kHz|MHz]

### **Returns**

<frequency>

### **Examples**

SOURce1:FM:INTernal:FREQuency 10kHz

sets the CH1 internal modulation frequency to 10 kHz.

## **[SOURce[1|2]]:FM:INTernal:FUNCTION**

This command sets or queries the modulating waveform of FM modulation for the specified channel. Use this command when the internal modulation source is selected.

If you specify EFILE when there is no EFILE or the EFILE is not yet defined, this command causes an error.

### **Syntax**

[SOURce[1|2]]:FM:INTernal:FUNCTION {SINusoid|SQUare|RAMP|PRNoise

|USER<NR1>|EMEMory|EFILE}

[SOURce[1|2]]:FM:INTernal:FUNCTION?

### **Arguments**

USER<NR1>|EMEMory

<NR1> specifies the user waveform memory location, can be any number from 0 to 15.

A user defined waveform saved in the user waveform memory or the EMEMory can be selected as a modulating signal.

EFILe

EFILe is used as a modulating signal.

### Returns

SIN|SQU|RAMP|PRN|USER<NR1>|EMEMory|EFILe

### Examples

SOURce1:FM:INTernal:FUNCTion SQUare

selects Square as the shape of modulating waveform for the CH1 output.

## [SOURce[1|2]]:FM:INTernal:FUNCTion:EFILe

This command sets or queries an EFILe name used as a modulating waveform for FM modulation. A file name must be specified in the mass storage system. This command returns “ ” if there is no file in the mass storage.

### Syntax

[SOURce[1|2]]:FM:INTernal:FUNCTion:EFILe <file\_name>

[SOURce[1|2]]:FM:INTernal:FUNCTion:EFILe?

### Arguments

<file\_name>::=<string> specifies a file name in the mass storage system. The <file\_name> includes path. Path separators are forward slashes (/).

### Returns

<file\_name>

### Examples

SOURce1:FM:INTernal:FUNCTion:EFILe “TEST”

sets a file named “TEST” in the mass storage.

## [SOURce[1|2]]:FM:SOURce

This command sets or queries the source of modulating signal of FM modulation for the specified channel.

### Syntax

[SOURce[1|2]]:FM:SOURce [INTernal|EXTernal]

[SOURce[1|2]]:FM:SOURce?



### Arguments

INTernal means that the carrier waveform is modulated with the internal source.  
EXTernal means that the carrier waveform is modulated with an external source.

### Returns

INT|EXT

### Examples

SOURce1:FM:SOURce INTernal  
sets the CH1 source of modulating signal to internal.

## [SOURce[1|2]]:FM:STATe

This command enables or disables FM modulation. The query returns the state of FM modulation.

### Syntax

[SOURce[1|2]]:FM:STATe {ON|OFF|<NR1>}  
[SOURce[1|2]]:FM:STATe?

### Arguments

ON or <NR1>≠0 enables FM modulation.  
OFF or <NR1>=0 disables FM modulation.

### Returns

<NR1>

### Examples

SOURce1:FM:STATe ON  
enables the CH1 FM modulation.

## [SOURce[1|2]]:FREQUency:CENTer

This command sets or queries the center frequency of sweep for the specified channel. This command is always used with the [SOURce[1|2]]:FREQUency:SPAN command. The setting range of center frequency depends on the waveform selected for sweep.

### Syntax

[SOURce[1|2]]:FREQUency:CENTer {<frequency>|MINimum|MAXimum}  
[SOURce[1|2]]:FREQUency:CENTer? {MINimum|MAXimum}

### Arguments

<frequency>::=<NRf>[<units>]

where:

<NRf> is the center frequency.

<units>::=[Hz|kHz|MHz]

### Returns

<frequency>

### Examples

SOURce1:FREQuency:CENTer 550kHz  
sets the CH1 center frequency to 550 kHz.

## [SOURce[1|2]]:FREQuency:CONCurent

This command enables or disables the function to copy the frequency (or period) of one channel to another channel.

The[SOURce[1|2]]:FREQuency:CONCurent command copies the frequency (or period) of the channel specified by the header suffix to another channel. If you specify CH1 with the header, the CH1 frequency will be copied to CH2.

When the concurrent copy function is enabled, the FreqLock function is also enabled automatically. Use general knob to adjust frequency (or period) of the two channels synchronously.

The[SOURce[1|2]]:FREQuency:CONCurent? command returns "0" (off) or "1" (on).

### Syntax

[SOURce[1|2]]:FREQuency:CONCurent {ON|OFF|<NR1>}

[SOURce[1|2]]:FREQuency:CONCurent?

### Arguments

ON or <NR1>≠0 enables the concurrent copy function.

OFF or <NR1>=0 disables the concurrent copy function.

### Returns

<NR1>

### Examples

SOURce1:FREQuency:CONCurent ON

copies the frequency value of CH1 to CH2.

## **[SOURce[1|2]]:FREQUency[:FIXed]**

This command sets or queries the frequency of output waveform for the specified channel. The setting range of output frequency depends on the type of output waveform. If you change the type of output waveform, it might change the output frequency because changing waveform types impacts on the setting range of output frequency. The resolution is 1  $\mu$ Hz or 12 digits.

### **Syntax**

```
[SOURce[1|2]]:FREQUency[:FIXed] {<frequency>|MINimum|MAXimum}  
[SOURce[1|2]]:FREQUency[:FIXed]? {MINimum|MAXimum}
```

### **Arguments**

<frequency>::=<NRf>[<units>]

where:

<NRf> is the output frequency.

<units>::=[Hz|kHz|MHz]

### **Returns**

<frequency>

### **Examples**

```
SOURce1:FREQUency:FIXed 500kHz  
sets the CH1 output frequency to 500 kHz.
```

## **[SOURce[1|2]]:FREQUency:SPAN**

This command sets or queries the span of frequency sweep for the specified channel. This command is always used with the [SOURce[1|2]]:FREQUency:CENTer command. The setting range of frequency span depends on the waveform selected for sweep.

### **Syntax**

```
[SOURce[1|2]]:FREQUency:SPAN {<frequency>|MINimum|MAXimum}  
[SOURce[1|2]]:FREQUency:SPAN? {MINimum|MAXimum}
```

### **Arguments**

<frequency>::=<NRf>[<units>]

where:

<NRf> is the frequency span.  
<units>::=[Hz|kHz|MHz]

### Returns

<frequency>

### Examples

SOURce1:FREQUENCY:SPAN 900 kHz  
sets the CH1 frequency span to 900 kHz.

## [SOURce[1|2]]:FREQUENCY:START

This command sets or queries the start frequency of sweep for the specified channel. This command is always used with the [SOURce[1|2]]:FREQUENCY:STOP command. The setting range of start frequency depends on the waveform selected for sweep.

### Syntax

[SOURce[1|2]]:FREQUENCY:START {<frequency>|MINimum|MAXimum}  
[SOURce[1|2]]:FREQUENCY:START? {MINimum|MAXimum}

### Arguments

<frequency>::=<NRf>[<units>]

where:

<NRf> is the start frequency.

<units>::=[Hz|kHz|MHz]

### Returns

<frequency>

### Examples

SOURce1:FREQUENCY:START 10kHz  
sets the sweep start frequency of CH1 to 10 kHz.

## [SOURce[1|2]]:FREQUENCY:STOP

This command sets or queries the stop frequency of sweep for the specified channel. This command is always used with the [SOURce[1|2]]:FREQUENCY:START command. The setting range of stop frequency depends on the waveform selected for sweep.

### Syntax

[SOURce[1|2]]:FREQUency:STOP {<frequency>|MINimum|MAXimum}  
[SOURce[1|2]]:FREQUency:STOP? {MINimum|MAXimum}

### Arguments

<frequency>::=<NRf>[<units>]

where:

<NRf> is the stop frequency.

<units>::=[Hz|kHz|MHz]

### Returns

<frequency>

### Examples

SOURce1:FREQUency:STOP 100KHz  
sets the stop frequency of CH1 to 100 kHz.

## [SOURce[1|2]]:FSKey[:FREQUency]

This command sets or queries the hop frequency of FSK modulation for the specified channel.

### Syntax

[SOURce[1|2]]:FSKey[:FREQUency] {<frequency>|MINimum|MAXimum}  
[SOURce[1|2]]:FSKey[:FREQUency]? {MINimum|MAXimum}

### Arguments

<frequency>::=<NRf>[<units>]

where:

<NRf> is the hop frequency.

<units>::=[Hz|kHz|MHz]

### Returns

<frequency>

### Examples

SOURce1:FSKey:FREQUency 1.0MHz  
sets the hop frequency of CH1 FSK modulation to 1.0 MHz.

## **[SOURce[1|2]]:FSKey:INTernal:RATE**

This command sets or queries the internal modulation rate of FSK modulation for the specified channel. Use this command when the internal modulation source is selected.

### **Syntax**

```
[SOURce[1|2]]:FSKey:INTernal:RATE {<rate>|MINimum|MAXimum}  
[SOURce[1|2]]:FSKey:INTernal:RATE? {MINimum|MAXimum}
```

### **Arguments**

<rate>::=<NRf>[<units>]

where:

<NRf> is the modulation rate.

<units>::=[Hz|kHz|MHz]

### **Returns**

<rate>

### **Examples**

```
SOURce1:FSKey:INTernal:RATE 50Hz  
sets the CH1 internal FSK modulation rate to 50 Hz.
```

## **[SOURce[1|2]]:FSKey:SOURce**

This command sets or queries the source of modulation signal of FSK modulation for the specified channel.

### **Syntax**

```
[SOURce[1|2]]:FSKey:SOURce [INTernal|EXTernal]  
[SOURce[1|2]]:FSKey:SOURce?
```

### **Arguments**

INTernal means that the carrier waveform is modulated with an internal source.

EXTernal means that the carrier waveform is modulated with an external source.

### **Returns**

INT|EXT

### **Examples**

```
SOURce1:FSKey:SOURce INTernal  
sets the CH1 source of modulating signal to internal.
```

## **[SOURce[1|2]]:FSKey:STATe**

This command enables or disables FSK modulation. The query returns the state of FSK modulation. Select a sine, square, ramp, or arbitrary waveform as the carrier waveform.

### **Syntax**

```
[SOURce[1|2]]:FSKey:STATe {ON|OFF|<NR1>}  
[SOURce[1|2]]:FSKey:STATe?
```

### **Arguments**

ON or <NR1>≠0 enables FSK modulation.  
OFF or <NR1>=0 disables FSK modulation.

### **Returns**

<NR1>

### **Examples**

```
SOURce1:FSKey:STATe ON  
enables the CH1 FSK modulation.
```

## **[SOURce[1|2]]:3FSKey[:FREQUENCY]**

This command sets or queries the hop frequency of 3FSK modulation for the specified channel.

### **Syntax**

```
[SOURce[1|2]]:3FSKey[:FREQUENCY] <n>,{<frequency>|MINimum|MAXimum}  
[SOURce[1|2]]:3FSKey[:FREQUENCY]? <n>,{MINimum|MAXimum}
```

### **Arguments**

<n>::=<NR1>

where:

<NR1> is the sequence number of hop frequency, which can be 1 or 2.

<frequency>::=<NRf>[<units>]

where:

<NRf> is the hop frequency.

<units>::=[Hz|kHz|MHz]

### **Returns**

<frequency>

### Examples

SOURce1:3FSKey:FREQuency 2,1.0MHz

sets the hop frequency 2 of CH1 3FSK modulation to 1.0 MHz.

## [SOURce[1|2]]:3FSKey:INTernal:RATE

This command sets or queries the internal modulation rate of 3FSK modulation for the specified channel.

### Syntax

[SOURce[1|2]]:3FSKey:INTernal:RATE {<rate>|MINimum|MAXimum}

[SOURce[1|2]]:3FSKey:INTernal:RATE? {MINimum|MAXimum}

### Arguments

<rate>::=<NRf>[<units>]

where:

<NRf> is the modulation rate.

<units>::=[Hz|kHz|MHz]

### Returns

<rate>

### Examples

SOURce1:3FSKey:INTernal:RATE 50Hz

sets the CH1 internal 3FSK modulation rate to 50 Hz.

## [SOURce[1|2]]:4FSKey[:FREQuency]

This command sets or queries the hop frequency of 4FSK modulation for the specified channel.

### Syntax

[SOURce[1|2]]:4FSKey[:FREQuency] <n>,{<frequency>|MINimum|MAXimum}

[SOURce[1|2]]:4FSKey[:FREQuency]? <n>,{MINimum|MAXimum}

### Arguments

<n>::=<NR1>

where:



<NR1> is the sequence number of hop frequency, which can be 1, 2, or 3.

<frequency>::=<NRf>[<units>]

where:

<NRf> is the hop frequency.

<units>::=[Hz|kHz|MHz]

### Returns

<frequency>

### Examples

SOURce1:4FSKey:FREQuency 2,1.0MHz

sets the hop frequency 2 of CH1 4FSK modulation to 1.0 MHz.

## [SOURce[1|2]]:4FSKey:INTernal:RATE

This command sets or queries the internal modulation rate of 4FSK modulation for the specified channel.

### Syntax

[SOURce[1|2]]:4FSKey:INTernal:RATE {<rate>|MINimum|MAXimum}

[SOURce[1|2]]:4FSKey:INTernal:RATE? {MINimum|MAXimum}

### Arguments

<rate>::=<NRf>[<units>]

where:

<NRf> is the modulation rate.

<units>::=[Hz|kHz|MHz]

### Returns

<rate>

### Examples

SOURce1:4FSKey:INTernal:RATE 50Hz

sets the CH1 internal 4FSK modulation rate to 50 Hz.

## [SOURce[1|2]]:FUNCTion:EFILe

This command sets or queries an EFILe name used as an output waveform. A file name must be specified in the mass storage system. This command returns “ ” if there is no file

in the mass storage.

### Syntax

```
[SOURce[1|2]]:FUNCTION:EFILe <file_name>  
[SOURce[1|2]]:FUNCTION:EFILe?
```

### Arguments

<file\_name>::=<string> specifies a file name in the mass storage system. The <file\_name> includes path. Path separators are forward slashes (/).

NOTE: The <file\_name> argument is case sensitive.

### Returns

<file\_name>

### Examples

```
SOURce1:FUNCTION:EFILe "TEST"  
sets a file named "TEST" in the mass storage.
```

## [SOURce[1|2]]:FUNCTION:RAMP:SYMMetry

This command sets or queries the symmetry of ramp waveform for the specified channel. The setting range is 0.0% to 100.0%.

### Syntax

```
[SOURce[1|2]]:FUNCTION:RAMP:SYMMetry {<symmetry>|MINimum|MAXimum}  
[SOURce[1|2]]:FUNCTION:RAMP:SYMMetry?
```

### Arguments

<symmetry>::=<NR2>[<units>]

where:

<NR2> is the symmetry.

<units>::=PCT

### Returns

<symmetry>

### Examples

```
SOURce1:FUNCTION:RAMP:SYMMetry 80.5  
sets the symmetry of the CH1 ramp waveform to 80.5%.
```

## [SOURCE[1|2]]:FUNCTION[:SHAPE]

This command sets or queries the shape of the output waveform. When the specified user memory is deleted, this command causes an error if you select the user memory.

### Syntax

```
[SOURCE[1|2]]:FUNCTION[:SHAPE] {SINusoid|SQUare|PULSe|RAMP  
|PRNoise|<Built_in>|USER<NR1>|EMEMory|EFILE}  
[SOURCE[1|2]]:FUNCTION[:SHAPE]?
```

### Arguments

<Built\_in>::={DC|AbsSine|AbsSineHalf|AmpALT|AttALT|GaussPulse|NegRamp|NPulse|PPulse|SineTra|SineVer|StairDn|StairUD|StairUp|Trapezia|Heart|Cardiac|LFPulse|Tens1|Tens2|Tens3|EOG|EEG|Pulseilogram|ResSpeed|Ignition|TP2A|ISP|VR|TP1|TP2B|P4|TP5A|TP5B|SCR|Surge|Airy|Besselj|Bessely|Cauchy|X^3|Erf|Erfc|ErfcInv|ErfInv|Dirichlet|ExpFall|ExpRise|Laguerre|Laplace|Legend|Gauss|HaverSine|Log|LogNormal|Lorentz|Maxwell|Rayleigh|Versiera|Weibull|Ln(x)|X^2|Round|Chirp|Rhombus|CosH|Cot|CotH|CotHCon|CotHPro|CscCon|Csc|CscPro|CscH|CscHCon|CscHPro|RecipCon|RecipPro|SecCon|SecPro|SecH|Sinc|SinH|Sqrt|Tan|TanH|ACos|ACosH|ACot|ACotCon|ACotPro|ACotH|ACotHCon|ACotHPro|Acsc|ACscCon|ACscPro|AcscH|ACscHCon|ACscHPro|Asec|AsecCon|AsecPro|AsecH|ASin|ASinH|ATan|ATanH|Bartlett|BarthannWin|Blackman|BlackmanH|BohmanWin|Boxcar|ChebWin|FlattopWin|Hamming|Hanning|Kaiser|NuttallWin|ParzenWin|TaylorWin|Triang|TukeyWin|Butterworth|Combin|CPulse|CWPulse|RoundHalf|BandLimited|BlaseiWave|Chebyshev1|Chebyshev2|DampedOsc|DualTone|Gamma|GateVibar|LFMPulse|MCNoise|Discharge|Quake|Radar|Ripple|RoundsPM|StepResp|SwingOsc|TV|Voice|AM|FM|PM|PWM}

NOTE: The arguments defined in <Built\_in> can not be abbreviated, all the upper and lower case letters are needed.

This command selects the output waveform of the instrument when Mod, Sweep and Burst are disabled.

This command selects the carrier waveform corresponding to the function when Mod, Sweep or Burst is currently enabled.

If you specify EFILE when there is no EFILE or the EFILE is not yet defined, this command causes an error.

If you change the type of output waveform, it might change the output frequency because changing waveform types impacts the setting range of output frequency.

USER<NR1>|EMEMory

<NR1> specifies the user waveform memory location, can be any number from 0 to 15.

A user defined waveform saved in the user waveform memory or the EMEMory can be

selected as an output waveform.

EFILe

EFILe is specified as an output waveform.

### Returns

SIN|SQU|PULS|RAMP|PRN|<Built\_in>|USER<NR1>|EMEMory|EFILe

### Examples

SOURce1:FUNCTion:SHAPE SQUare

selects the shape of CH1 output waveform to square waveform.

## [SOURce[1|2]]:MOD:STATE

This command enables or disables the modulation function for the specified channel. The query returns the state of modulation function.

### Syntax

[SOURce[1|2]]:MOD:STATE {ON|OFF|<NR1>}

[SOURce[1|2]]:MOD:STATE?

### Arguments

ON or <NR1>≠0 enables the modulation function.

OFF or <NR1>=0 disables the modulation function.

### Returns

<NR1>

### Examples

SOURce1:MOD:STATE ON

enables the modulation function for the CH1.

## [SOURce[1|2]]:OSKey:INTERNAL:RATE

This command sets or queries the internal modulation rate of OSK modulation for the specified channel.

### Syntax

[SOURce[1|2]]:OSKey:INTERNAL:RATE {<rate>|MINimum|MAXimum}

[SOURce[1|2]]:OSKey:INTERNAL:RATE? {MINimum|MAXimum}

### Arguments

<rate>::=<NRf>[<units>]

where:

<NRf> is the modulation rate.

<units>::=[Hz|kHz|MHz]

### Returns

<rate>

### Examples

```
SOURce1:OSKey:INternal:RATE 50Hz
```

sets the CH1 internal OSK modulation rate to 50 Hz.

## [SOURce[1|2]]:OSKey:TIME

This command sets or queries the oscillate period for the OSK modulation for the specified channel.

### Syntax

```
[SOURce[1|2]]:OSKey:TIME {<seconds>|MINimum|MAXimum}
```

```
[SOURce[1|2]]:OSKey:TIME?
```

### Arguments

<seconds>::=<NRf>[<units>]

where:

<NRf> is the oscillate period in seconds.

<units>::=[ns|μs|ms|s]

### Returns

<seconds>

### Examples

```
SOURce1:OSKey:TIME 100μs
```

sets the CH1 OSK oscillate period to 100 μs.

## [SOURce[1|2]]:PHASe[:ADJust]

This command sets or queries the phase of output waveform for the specified channel. Set the value in radians or degrees. If no units are specified, the default is RAD. The query

returns the value in RAD.

This command is supported when you select a waveform other than DC and Noise.

### **Syntax**

```
[SOURce[1|2]]:PHASe[:ADJust] {<phase>|MINimum|MAXimum}
```

```
[SOURce[1|2]]:PHASe[:ADJust]? {MINimum|MAXimum}
```

### **Arguments**

<phase>::=<NR3>[<units>]

where:

<NR3> is the phase of output waveform.

<units>::=[RAD|DEG]

If <units> are omitted, RAD is specified automatically. The setting ranges are:

RAD: 0 to +2 PI, relative to phase value

DEG: 0 to +360, relative to phase value

### **Returns**

<phase>

### **Examples**

```
SOURce1:PHASe:ADJust MAXimum
```

sets the maximum value for the phase of CH1 output waveform.

## **[SOURce[1|2]]:PHASe:INITiate (No Query Form)**

This command synchronizes the phase of CH1 and CH2 output waveforms. The arbitrary generator performs the same operation if you specify either SOURce1 or SOURce2.

### **Syntax**

```
[SOURce[1|2]]:PHASe:INITiate
```

### **Arguments**

None

### **Examples**

```
SOURce1:PHASe:INITiate
```

synchronizes the phase of CH1 and CH2 output signals.

## **[SOURce[1|2]]:PM[:DEVIation]**

This command sets or queries the phase deviation of PM modulation for the specified channel.

### **Syntax**

```
[SOURce[1|2]]:PM[:DEVIation] {<deviation>|MINimum|MAXimum}  
[SOURce[1|2]]:PM[:DEVIation]? [MINimum|MAXimum]
```

### **Arguments**

<deviation>::=<NR3>[<units>]

where:

<NR3> is the phase deviation.

<units>::=[RAD|DEG]

If <units> are omitted, RAD is specified automatically. The setting ranges are:

RAD: 0 PI to +1 PI, relative to phase value

DEG: 0 to +180, in 1 degree steps, relative to phase value

### **Returns**

<deviation>

### **Examples**

```
SOURce1:PM:DEVIation MAXimum
```

sets the maximum value for the CH1 phase deviation of PM modulation.

## **[SOURce[1|2]]:PM:INTernal:FREQUENCY**

This command sets or queries the internal modulation frequency of PM modulation for the specified channel. Use this command when the internal modulation source is selected.

Set the internal modulation frequency from 2 mHz to 100.00 kHz with resolution of 1 mHz.

### **Syntax**

```
[SOURce[1|2]]:PM:INTernal:FREQUENCY {<frequency>|MINimum|MAXimum}  
[SOURce[1|2]]:PM:INTernal:FREQUENCY? [MINimum|MAXimum]
```

### **Arguments**

<frequency>::=<NRf>[<units>]

where:

<NRf> is the modulation frequency.

<units>::=[Hz|kHz|MHz]

**Returns**

<frequency>

**Examples**

SOURce1:PM:INTernal:FREQuency 10kHz  
sets the CH1 internal modulation frequency to 10 kHz.

**[SOURce[1|2]]:PM:INTernal:FUNctIon**

This command sets or queries the modulating waveform of PM modulation for the specified channel. Use this command when the internal modulation source is selected. If you specify EFILE when there is no EFILE or the EFILE is not yet defined, this command causes an error.

**Syntax**

[SOURce[1|2]]:PM:INTernal:FUNctIon {SINusoid|SQUare|RAMP|PRNoise  
|USER<NR1>|EMEMory|EFILE}  
[SOURce[1|2]]:PM:INTernal:FUNctIon?

**Arguments**

USER<NR1>|EMEMory

<NR1> specifies the user waveform memory location, can be any number from 0 to 15. A user defined waveform saved in the user waveform memory or the EMEMory can be selected as a modulating signal.

EFILE

EFILE is used as a modulating signal.

**Returns**

SIN|SQU|RAMP|PRN|USER<NR1>|EMEMory|EFILE

**Examples**

SOURce1:PM:INTernal:FUNctIon SQUare  
selects Square as the shape of modulating waveform for the CH1 output.

**[SOURce[1|2]]:PM:INTernal:FUNctIon:EFILE**

This command sets or queries an EFILE name used as a modulating waveform for PM modulation. A file name must be specified in the mass storage system. This command returns “ ” if there is no file in the mass storage.



**Syntax**

[SOURce[1|2]]:PM:INTernal:FUNction:EFILe <file\_name>  
[SOURce[1|2]]:PM:INTernal:FUNction:EFILe?

**Arguments**

<file\_name>::=<string> specifies a file name in the mass storage system. The <file\_name> includes path. Path separators are forward slashes (/).

**Returns**

<file\_name>

**Examples**

SOURce1:PM:INTernal:FUNction:EFILe "TEST"  
sets a file named "TEST" in the mass storage.

**[SOURce[1|2]]:PM:SOURce**

This command sets or queries the source of modulation signal of PM modulation for the specified channel.

**Syntax**

[SOURce[1|2]]:PM:SOURce [INTernal|EXTernal]  
[SOURce[1|2]]:PM:SOURce?

**Arguments**

INTernal means that the carrier waveform is modulated with an internal source.  
EXTernal means that the carrier waveform is modulated with an external source.

**Returns**

INT|EXT

**Examples**

SOURce1:PM:SOURce INTernal  
sets the CH1 source of modulating signal to internal.

**[SOURce[1|2]]:PM:STATe**

This command enables or disables PM modulation. The query returns the state of PM modulation. Select a sine, square, ramp, or arbitrary waveform as the carrier waveform.

**Syntax**

[SOURce[1|2]]:PM:STATe {ON|OFF|<NR1>}  
[SOURce[1|2]]:PM:STATe?

**Arguments**

ON or <NR1>≠0 enables PM modulation.  
OFF or <NR1>=0 disables PM modulation.

**Returns**

<NR1>

**Examples**

SOURce1:PM:STATe ON  
enables the CH1 PM modulation.

**[SOURce[1|2]]:PSKey[:DEViation]**

This command sets or queries the phase deviation of PSK modulation for the specified channel.

**Syntax**

[SOURce[1|2]]:PSKey[:DEViation] {<deviation>|MINimum|MAXimum}  
[SOURce[1|2]]:PSKey[:DEViation]? [MINimum|MAXimum]

**Arguments**

<deviation>::=<NR3>[<units>]

where:

<NR3> is the phase deviation.

<units>::=[RAD|DEG]

If <units> are omitted, RAD is specified automatically. The setting ranges are:

RAD: 0 PI to +1 PI, relative to phase value

DEG: 0 to +180, in 1 degree steps, relative to phase value

**Returns**

<deviation>

**Examples**

SOURce1:PSKey:DEViation MAXimum  
sets the maximum value for the CH1 phase deviation of PSK modulation.

## **[SOURce[1|2]]:PSKey:INTernal:RATE**

This command sets or queries the internal modulation rate of PSK modulation for the specified channel. Use this command when the internal modulation source is selected.

### **Syntax**

```
[SOURce[1|2]]:PSKey:INTernal:RATE {<rate>|MINimum|MAXimum}  
[SOURce[1|2]]:PSKey:INTernal:RATE? {MINimum|MAXimum}
```

### **Arguments**

<rate>::=<NRf>[<units>]

where:

<NRf> is the modulation rate.

<units>::=[Hz|kHz|MHz]

### **Returns**

<rate>

### **Examples**

```
SOURce1:PSKey:INTernal:RATE 50Hz  
sets the CH1 internal modulation rate to 50 Hz.
```

## **[SOURce[1|2]]:PSKey:SOURce**

This command sets or queries the source of modulation signal of PSK modulation for the specified channel.

### **Syntax**

```
[SOURce[1|2]]:PSKey:SOURce [INTernal|EXTernal]  
[SOURce[1|2]]:PSKey:SOURce?
```

### **Arguments**

INTernal means that the carrier waveform is modulated with an internal source.

EXTernal means that the carrier waveform is modulated with an external source.

### **Returns**

INT|EXT

### **Examples**

```
SOURce1:PSKey:SOURce INTernal  
sets the CH1 source of modulating signal to internal.
```

## [SOURce[1|2]]:PSKey:STATe

This command enables or disables PSK modulation. The query returns the state of PSK modulation. Select a sine, square, ramp, or arbitrary waveform as the carrier waveform.

### Syntax

```
[SOURce[1|2]]:PSKey:STATe {ON|OFF|<NR1>}  
[SOURce[1|2]]:PSKey:STATe?
```

### Arguments

ON or <NR1>≠0 enables PSK modulation.  
OFF or <NR1>=0 disables PSK modulation.

### Returns

<NR1>

### Examples

```
SOURce1:PSKey:STATe ON  
enables the CH1 PSK modulation.
```

## [SOURce[1|2]]:PULSe:DCYClE

This command sets or queries the duty cycle of the pulse waveform for the specified channel.

This parameter is related to the pulse width and when any of them is changed, the other will be modified automatically.

The pulse duty cycle is limited by the minimum pulse width and pulse period.

**pulse duty cycle  $\geq 100 \times \text{minimum pulse width} \div \text{pulse period}$**

**pulse duty cycle  $\leq 100 \times (1 - 2 \times \text{minimum pulse width} \div \text{pulse period})$**

### Syntax

```
[SOURce[1|2]]:PULSe:DCYClE {<percent>|MINimum|MAXimum}  
[SOURce[1|2]]:PULSe:DCYClE? [MINimum|MAXimum]
```

### Arguments

<percent>::=<NR2>[<units>]

where:

<NR2> is the duty cycle.

<units>::=PCT

**Returns**

<percent>

**Examples**

```
SOURce1:PULSe:DCYClE 80.5
```

sets the duty cycle of the pulse waveform on CH1 to 80.5%.

**[SOURce[1|2]]:PWM:INTernal:FREQuency**

This command sets or queries the internal modulation frequency of PWM modulation for the specified channel. Use this command when the internal modulation source is selected. Set the internal modulation frequency from 2 mHz to 100.00 kHz with resolution of 1 mHz. Select the source of modulating signal by using the [SOURce[1|2]]:PWM:SOURce [INTernal|EXTernal] command.

**Syntax**

```
[SOURce[1|2]]:PWM:INTernal:FREQuency {<frequency>|MINimum|MAXimum}  
[SOURce[1|2]]:PWM:INTernal:FREQuency?
```

**Arguments**

<frequency>::=<NRf>[<units>]

where <NRf> is the modulation frequency.

<units>::=[Hz|kHz|MHz]

**Returns**

<frequency>

**Examples**

```
SOURce1:PWM:INTernal:FREQuency 10kHz
```

sets the CH1 internal modulation frequency of PWM modulation to 10 kHz.

**[SOURce[1|2]]:PWM:INTernal:FUNCTion**

This command sets or queries the modulating waveform of PWM modulation for the specified channel. Use this command when the internal modulation source is selected. If you specify EFILE when there is no EFILE or the EFILE is not yet defined, this command causes an error.

**Syntax**

[SOURce[1|2]]:PWM:INTernal:FUNcTion {SINusoid|SQUare|RAMP|PRNoise  
|USER<NR1>|EMEMory|EFILe}  
[SOURce[1|2]]:PWM:INTernal:FUNcTion?

### Arguments

SINusoid|SQUare|RAMP|PRNoise

One of four types of function waveform can be selected as a modulating signal.

USER<NR1>|EMEMory

<NR1> specifies the user waveform memory location, can be any number from 0 to 15.

A user defined waveform saved in the user waveform memory or the EMEMory can be selected as a modulating signal.

EFILe

EFILe is used as a modulating signal.

### Returns

SIN|SQU|RAMP|PRN|USER<NR1>|EMEMory|EFILe

### Examples

SOURce1:PWM:INTernal:FUNcTion SQUare

selects Square as the shape of modulating waveform of PWM modulation for the CH1 output.

## [SOURce[1|2]]:PWM:INTernal:FUNcTion:EFILe

This command sets or queries an EFILe name used as a modulating waveform for PWM modulation. A file name must be specified in the mass storage system. This command returns “ ” if there is no file in the mass storage.

### Syntax

[SOURce[1|2]]:PWM:INTernal:FUNcTion:EFILe <file\_name>

[SOURce[1|2]]:PWM:INTernal:FUNcTion:EFILe?

### Arguments

<file\_name>::=<string> specifies a file name in the mass storage system. The <file\_name> includes path. Path separators are forward slashes (/).

### Returns

<file\_name>

### Examples

SOURce1:PWM:INTernal:FUNction:EFILe "TEST"  
creates a file named "TEST" in the mass storage.

## **[SOURce[1|2]]:PWM:SOURce**

This command sets or queries the source of modulating signal of PWM modulation for the specified channel.

### **Syntax**

[SOURce[1|2]]:PWM:SOURce [INTernal|EXTernal]  
[SOURce[1|2]]:PWM:SOURce?

### **Arguments**

INTernal means that the carrier waveform is modulated with the internal source.  
EXTernal means that the carrier waveform is modulated with an external source.

### **Returns**

INT|EXT

### **Examples**

SOURce1:PWM:SOURce INTernal  
sets the source of modulating signal on CH1 to internal.

## **[SOURce[1|2]]:PWM:STATe**

This command enables or disables PWM modulation. The query returns the state of PWM modulation. Select only pulse waveform as a carrier waveform for PWM.

### **Syntax**

[SOURce[1|2]]:PWM:STATe {ON|OFF|<NR1>}  
[SOURce[1|2]]:PWM:STATe?

### **Arguments**

ON or <NR1>≠0 enables PWM modulation.  
OFF or <NR1>=0 disables PWM modulation.

### **Returns**

<NR1>

### **Examples**

SOURce1:PWM:STATe ON  
enables the CH1 PWM modulation.

### **[SOURce[1|2]]:PWM[:DEVIation]:DCYCLE**

This command sets or queries the PWM deviation in percent for the specified channel.  
The setting range must meet the following conditions:

$$\text{Deviation} \leq \text{Pulse Width} - \text{PWmin}$$

$$\text{Deviation} \leq \text{Pulse Period} - \text{Pulse Width} - \text{PWmin}$$

$$\text{Deviation} \leq \text{Pulse Width} - 0.8 \times (\text{Leading Edge Time} + \text{Trailing Edge Time})$$

$$\text{Deviation} \leq \text{Pulse Period} - \text{Pulse Width} - 0.8 \times (\text{Leading Edge Time} + \text{Trailing Edge Time})$$

where PWmin is the minimum pulse width.

#### **Syntax**

[SOURce[1|2]]:PWM[:DEVIation]:DCYCLE {<percent>|MINimum|MAXimum}  
[SOURce[1|2]]:PWM[:DEVIation]:DCYCLE?

#### **Arguments**

<percent>::=<NR2>[<units>]

where:

<NR2> is the PWM deviation.

<units>::=PCT

#### **Returns**

<percent>

#### **Examples**

SOURce1:PWM[:DEVIation]:DCYCLE 5.0  
sets the CH1 PWM deviation to 5.0%.

### **[SOURce[1|2]]:SWEep:SOURce**

This command sets or queries the trigger source in the sweep mode for the specified channel.

#### **Syntax**

[SOURce[1|2]]:SWEep:SOURce [TIMER|MANual|EXternal]  
[SOURce[1|2]]:SWEep:SOURce?



### Arguments

TIMer specifies an internal clock as the trigger source.

MANual specifies a manual trigger input as the trigger source.

EXTernal specifies an external trigger input as the trigger source.

### Returns

TIM|MAN|EXT

### Examples

SOURce1:SWEep:SOURce EXTernal

sets an external trigger input as the trigger source in the sweep mode.

## [SOURce[1|2]]:SWEep:SPACing

This command selects linear or logarithmic spacing for the sweep for the specified channel.

The query returns the type for the sweep spacing for the specified channel.

### Syntax

[SOURce[1|2]]:SWEep:SPACing {LINear|LOGarithmic}

[SOURce[1|2]]:SWEep:SPACing?

### Arguments

LINear sets the sweep spacing to linear.

LOGarithmic sets the sweep spacing to logarithmic.

### Returns

LIN|LOG

### Examples

SOURce1:SWEep:SPACing LINear

sets the CH1 sweep spacing to linear.

## [SOURce[1|2]]:SWEep:STATe

This command enables or disables the sweep mode for the specified channel. The query returns the state of sweep mode.

### Syntax

[SOURce[1|2]]:SWEep:STATe {ON|OFF|<NR1>}

[SOURce[1|2]]:SWEep:STATe?

### Arguments

ON or <NR1>≠0 enables the sweep mode.

OFF or <NR1>=0 disables the sweep mode.

### Returns

<NR1>

### Examples

SOURce1:SWEep:STATe ON

enables the sweep mode for the CH1.

## [SOURce[1|2]]:SWEep:TIME

This command sets or queries the sweep time for the sweep for the specified channel.

The sweep time does not include hold time and return time. The setting range is 1 ms to 500 s.

### Syntax

[SOURce[1|2]]:SWEep:TIME {<seconds>|MINimum|MAXimum}

[SOURce[1|2]]:SWEep:TIME?

### Arguments

<seconds>::=<NRf>[<units>]

where:

<NRf> is the sweep time in seconds.

<units>::=[ns|μs|ms|s]

### Returns

<seconds>

### Examples

SOURce1:SWEep:TIME 100ms

sets the CH1 sweep time to 100 ms.

## [SOURce[1|2]]:VOLTage[:LEVel][:IMMediate]:OFFSet

This command sets or queries the offset level for the specified channel.

### Syntax

[SOURce[1|2]]:VOLTage[:LEVel][:IMMediate]:OFFSet {<voltage>|MINimum|MAXimum}  
[SOURce[1|2]]:VOLTage[:LEVel][:IMMediate]:OFFSet? {MINimum|MAXimum}

### Arguments

<voltage>::=<NR3>[<units>]

where:

<NR3> is the offset voltage level.

<units>::=[mV|V]

### Returns

<voltage>

### Examples

SOURce1:VOLTage:LEVel:IMMediate:OFFSet 500mV  
sets the CH1 offset level to 500 mV.

## [SOURce[1|2]]:VOLTage[:LEVel][:IMMediate][:AMPLitude]

This command sets or queries the output amplitude for the specified channel. The amplitude resolution is 1 mVpp or four digits.

### Syntax

[SOURce[1|2]]:VOLTage[:LEVel][:IMMediate][:AMPLitude]  
{<amplitude>|MINimum|MAXimum}  
[SOURce[1|2]]:VOLTage[:LEVel][:IMMediate][:AMPLitude]? {MINimum|MAXimum}

### Arguments

<amplitude>::=<NR3>[<units>]

where:

<NR3> is the output amplitude.

<units>::=[Vpp]

### Returns

<amplitude>

### Examples

SOURce1:VOLTage:LEVel:IMMediate:AMPLitude 1Vpp  
sets the CH1 output amplitude to 1 Vpp.

## **SYSTem:BEEPer[:IMMediate] (No Query Form)**

This command causes the instrument to beep immediately. This command is only available when the beeper is enabled.

### **Syntax**

SYSTem:BEEPer[:IMMediate]

### **Arguments**

None

### **Examples**

SYSTEM:BEEPER[:IMMEDIATE]

causes a beep.

## **SYSTem:BEEPer:STATe**

The SYSTem:BEEPer:STATe command sets the beeper ON or OFF.

The SYSTem:BEEPer:STATe? command returns "0" (OFF) or "1" (ON).

When the beeper is set to ON, the instrument will beep when an error message or a warning message is displayed on the screen. The instrument does not beep when an error or warning caused by remote command execution.

### **Syntax**

SYSTem:BEEPer:STATe {ON|OFF|<NR1>}

SYSTem:BEEPer:STATe?

### **Arguments**

ON or <NR1>≠0 enables the beeper.

OFF or <NR1>=0 disables the beeper.

### **Returns**

<NR1>

### **Examples**

SYSTEM:BEEPER:STATE ON

enables the beeper function.

## **SYSTem:ERRor[:NEXT]? (Query Only)**

This query-only command returns the contents of the Error/Event queue.

### **Syntax**

SYSTem:ERRor[:NEXT]?

### **Arguments**

None

### **Returns**

<Error/event number>::=<NR1>

<Error/event description>::=<string>

### **Examples**

SYSTEM:ERROR:NEXT?

might return the following response:

-201,"Invalid while in local"

If the instrument detects an error or an event occurs, the event number and event message will be returned.

## **SYSTem:KLOCK[:STATe]**

This command locks or unlocks the instrument front panel controls. The query command returns "0" (OFF) or "1" (ON).

### **Syntax**

SYSTem:KLOCK[:STATe] {ON|OFF|<NR1>}

SYSTem:KLOCK[:STATe]?

### **Arguments**

ON or <NR1>≠0 locks front panel controls.

OFF or <NR1>=0 unlocks front panel controls.

### **Returns**

<NR1>

### **Examples**

SYSTEM:KLOCK[:STATE] ON

locks front panel controls.

## **SYSTem:LANguage**

This command sets or queries the language that the instrument uses to display information on the screen.

### **Syntax**

```
SYSTem:LANguage  
{SCHinese|TCHinese|ENGLish|PORTuguese|GERMan|POLish|KORean|JAPAnese}  
SYSTem:LANguage?
```

### **Arguments**

SCHinese|TCHinese|ENGLish|PORTuguese|GERMan|POLish|KORean|JAPAnese specifies which language will be used to display instrument information on the screen.

### **Returns**

```
SCH|TCH|ENGL|PORT|GERM|POL|KOR|JAPA
```

### **Examples**

```
SYSTEM:LANGUAGE ENGLish  
specifies that the instrument displays information in English.
```

## **SYSTem:POWeron**

Set the configuration to be used by the instrument at power-on to DEFault or LAsT.  
Query the configuration to be used by the instrument at power-on.

### **Syntax**

```
SYSTem:POWeron [DEFault|LAsT]  
SYSTem:POWeron?
```

### **Arguments**

DEFault means that set the configuration to be used by the instrument at power-on to default setup.  
LAsT means that set the configuration to be used by the instrument at power-on to last setup.

### **Returns**

```
DEF|LAS
```

### **Examples**

SYSTem:POWeron LAST

sets the configuration to be used by the instrument at power-on to LAST.

SYSTem:POWeron?

returns LAS.

## **SYSTem:RESTART**

This command restarts the instrument.

### **Syntax**

SYSTem:RESTART

### **Arguments**

None

### **Examples**

SYSTem:RESTART

restarts the instrument.

## **SYSTem:VERSion? (Query Only)**

This query-only command returns the conformed SCPI version of the instrument.

### **Syntax**

SYSTem:VERSion?

### **Arguments**

None

### **Returns**

<SCPI Version>::=YYYY.V

where:

YYYY – indicates year.

V – indicates the version number for that year.

### **Examples**

SYSTEM:VERSION?

might return 1999.0.

## TRACe|DATA:CATalog? (Query Only)

This query-only command returns the names of user waveform memory and edit memory.

### Syntax

TRACe|DATA:CATalog?

### Arguments

None

### Returns

<string>

A series of strings separated by commas is returned. Each string is enclosed within quotation marks.

### Examples

TRACe|DATA:CATALOG?

might return "USER0","USER2","EMEM"

## TRACe|DATA:COPY (No Query Form)

This command copies the contents of edit memory (or user waveform memory) to a specified user waveform memory (or edit memory).

### Syntax

TRACe|DATA:COPY <trace\_name>,EMEMory

TRACe|DATA:COPY EMEMory,{USER<NR1>}

### Arguments

<trace\_name>::={USER<NR1>}

<NR1> specifies the user waveform memory location, can be any number from 0 to 15.

### Examples

DATA:COPY USER0,EMEMory

copies the waveform data in the edit memory to the user waveform memory USER0.

DATA:COPY EMEMory,USER0

copies the waveform data in the user waveform memory USER0 to the edit memory.



## TRACe|DATA[:DATA]

This command transfers the waveform data from the external controller to the edit memory in the instrument. The query returns the binary block data.

### Syntax

```
TRACe|DATA[:DATA] EMEMory,<binary_block_data>  
TRACe|DATA[:DATA]? EMEMory
```

### Arguments

<binary\_block\_data>

where <binary\_block\_data> is the waveform data in binary format.

### Returns

<binary\_block\_data>

### Examples

```
DATA:DATA EMEMory,#42000<DAB><DAB>...<DAB>
```

transmits a waveform to the edit memory in the instrument. The block data element #21000 indicates that 2 is the number of digits in 1000 (byte count) and the 1000 bytes of binary data are to be transmitted.

## TRACe|DATA[:DATA]:VALue

This command sets or queries the data value at the specified point in the edit memory.

### Syntax

```
TRACe|DATA[:DATA]:VALue EMEMory,<point>,<data>  
TRACe|DATA[:DATA]:VALue? EMEMory,<point>
```

### Arguments

<point>::=<NR1>

where:

<NR1> is the specified point number in the edit memory.

<data>::=<NRf>

where:

<NRf> is the voltage value for the specified point number.

<units>::=[mV|V]

### Returns

<NRf>

### Examples

DATA:DATA:VALue EMEMory,200,1.5V

sets the voltage value to 1.5V for the point number 200 in the edit memory.

DATA:DATA:VALue? EMEMory,200

might return "1.5000000+e0".

This example indicates that the voltage value of point number 200 is set to 1.5V.

## TRACe|DATA:POINts

This command sets or queries the number of data points for the waveform created in the edit memory.

### Syntax

TRACe|DATA:POINts EMEMory[,<points>|MINimum|MAXimum]

TRACe|DATA:POINts? EMEMory{,MIN|MAX}

### Arguments

<points>::=<NR1>

where

<NR1> sets the number of points for the waveform created in the edit memory, can be any number from 2 to 100,000.

### Returns

<NR1>

### Examples

DATA:POINts EMEMory, 500

sets the waveform data points to 500 in the edit memory.

## \*TRG (No Query Form)

This command generates a trigger event.

### Syntax

\*TRG

### Arguments

None

**Examples**

\*TRG

generates a trigger event.