

# Programmable AC/DC Power Source

ASR-2000 Series

---

## PROGRAMMING MANUAL

Rev. E



ISO-9001 CERTIFIED MANUFACTURER

**GW INSTEK**

This manual contains proprietary information, which is protected by copyright. All rights are reserved. No part of this manual may be photocopied, reproduced or translated to another language without prior written consent of Good Will company.

The information in this manual was correct at the time of printing. However, Good Will continues to improve products and reserves the rights to change specification, equipment, and maintenance procedures at any time without notice.

**Good Will Instrument Co., Ltd.**  
**No. 7-1, Jhongsing Rd., Tucheng Dist., New Taipei City 236, Taiwan.**

# Table of Contents

|          |                                  |            |
|----------|----------------------------------|------------|
| <b>S</b> | <b>SAFETY INSTRUCTIONS .....</b> | <b>4</b>   |
| <b>G</b> | <b>GETTING STARTED.....</b>      | <b>8</b>   |
|          | ASR-2000 Series Overview .....   | 9          |
|          | Appearance.....                  | 14         |
| <b>R</b> | <b>REMOTE CONTROL.....</b>       | <b>22</b>  |
|          | Interface Configuration .....    | 23         |
|          | Command Syntax .....             | 41         |
|          | Command List .....               | 45         |
|          | Status Register Overview .....   | 140        |
|          | Error List .....                 | 156        |
| <b>A</b> | <b>APPENDIX.....</b>             | <b>164</b> |
|          | Factory Default Settings.....    | 164        |
| <b>I</b> | <b>INDEX.....</b>                | <b>176</b> |

# S SAFETY INSTRUCTIONS

This chapter contains important safety instructions that you must follow during operation and storage. Read the following before any operation to ensure your safety and to keep the instrument in the best possible condition.

## Safety Symbols

These safety symbols may appear in this manual or on the instrument.

---



WARNING

Warning: Identifies conditions or practices that could result in injury or loss of life.



CAUTION

Caution: Identifies conditions or practices that could result in damage to the ASR-2000 or to other properties.



DANGER High Voltage



Attention Refer to the Manual



Protective Conductor Terminal



Earth (ground) Terminal



Do not dispose electronic equipment as unsorted municipal waste. Please use a separate collection facility or contact the supplier from which this instrument was purchased.

## Safety Guidelines

---

### General Guideline



#### CAUTION

- Do not place any heavy object on the ASR-2000.
- Avoid severe impact or rough handling that leads to damaging the ASR-2000.
- Do not discharge static electricity to the ASR-2000.
- Use only mating connectors, not bare wires, for the terminals.
- Do not block the cooling fan opening.
- Do not disassemble the ASR-2000 unless you are qualified.

(Measurement categories) EN 61010-1:2010 specifies the measurement categories and their requirements as follows. The ASR-2000 doesn't fall under category II, III or IV.

- Measurement category IV is for measurement performed at the source of low-voltage installation.
- Measurement category III is for measurement performed in the building installation.
- Measurement category II is for measurement performed on the circuits directly connected to the low voltage installation.
- 0 is for measurements performed on circuits not directly connected to Mains.

---

### Power Supply



#### WARNING

- AC Input voltage range:
    - 100 ~ 240 Vac
    - Frequency: 47 ~ 63 Hz
  - To avoid electrical shock connect the protective grounding conductor of the AC power cord to an earth ground.
-

- Cleaning the ASR-2000
- Disconnect the power cord before cleaning.
  - Use a soft cloth dampened in a solution of mild detergent and water. Do not spray any liquid.
  - Do not use chemicals containing harsh material such as benzene, toluene, xylene, and acetone.

- Operation Environment
- Location: Indoor, no direct sunlight, dust free, almost non-conductive pollution (Note below)
  - Relative Humidity: 20%~ 80%, no condensation
  - Altitude: < 2000m
  - Temperature: 0°C to 40°C

(Pollution Degree) EN 61010-1:2010 specifies the pollution degrees and their requirements as follows. The ASR-2000 falls under degree 2.

Pollution refers to “addition of foreign matter, solid, liquid, or gaseous (ionized gases), that may produce a reduction of dielectric strength or surface resistivity”.

- Pollution degree 1: No pollution or only dry, non-conductive pollution occurs. The pollution has no influence.
- Pollution degree 2: Normally only non-conductive pollution occurs. Occasionally, however, a temporary conductivity caused by condensation must be expected.
- Pollution degree 3: Conductive pollution occurs, or dry, non-conductive pollution occurs which becomes conductive due to condensation which is expected. In such conditions, equipment is normally protected against exposure to direct sunlight, precipitation, and full wind pressure, but neither temperature nor humidity is controlled.

- Storage environment
- Location: Indoor
  - Temperature: -10°C to 70°C
  - Relative Humidity: ≤90%, no condensation

Disposal



Do not dispose this instrument as unsorted municipal waste. Please use a separate collection facility or contact the supplier from which this instrument was purchased. Please make sure discarded electrical waste is properly recycled to reduce environmental impact.

## Power cord for the United Kingdom

When using the instrument in the United Kingdom, make sure the power cord meets the following safety instructions.

NOTE: This lead/appliance must only be wired by competent persons




**WARNING: THIS APPLIANCE MUST BE EARTHED**

IMPORTANT: The wires in this lead are coloured in accordance with the following code:

|                |              |
|----------------|--------------|
| Green/ Yellow: | Earth        |
| Blue:          | Neutral      |
| Brown:         | Live (Phase) |



As the colours of the wires in main leads may not correspond with the coloured marking identified in your plug/appliance, proceed as follows:

The wire which is coloured Green & Yellow must be connected to the Earth terminal marked with either the letter E, the earth symbol  or coloured Green/Green & Yellow.

The wire which is coloured Blue must be connected to the terminal which is marked with the letter N or coloured Blue or Black.

The wire which is coloured Brown must be connected to the terminal marked with the letter L or P or coloured Brown or Red.

If in doubt, consult the instructions provided with the equipment or contact the supplier.

This cable/appliance should be protected by a suitably rated and approved HBC mains fuse: refer to the rating information on the equipment and/or user instructions for details. As a guide, a cable of 0.75mm<sup>2</sup> should be protected by a 3A or 5A fuse. Larger conductors would normally require 13A types, depending on the connection method used.

Any exposed wiring from a cable, plug or connection that is engaged in a live socket is extremely hazardous. If a cable or plug is deemed hazardous, turn off the mains power and remove the cable, any fuses and fuse assemblies. All hazardous wiring must be immediately destroyed and replaced in accordance to the above standard.

# **G**ETTING STARTED

This chapter describes the ASR-2000 power supply in a nutshell, including its main features and front / rear panel introduction.

**ASR-2000**

**• ASR-2000R**



---

|                                |    |
|--------------------------------|----|
| ASR-2000 Series Overview ..... | 9  |
| Series lineup .....            | 9  |
| Operating Area .....           | 9  |
| Main Features .....            | 11 |
| Accessories .....              | 12 |
| Appearance .....               | 14 |
| Front Panel .....              | 14 |
| Rear Panel.....                | 19 |



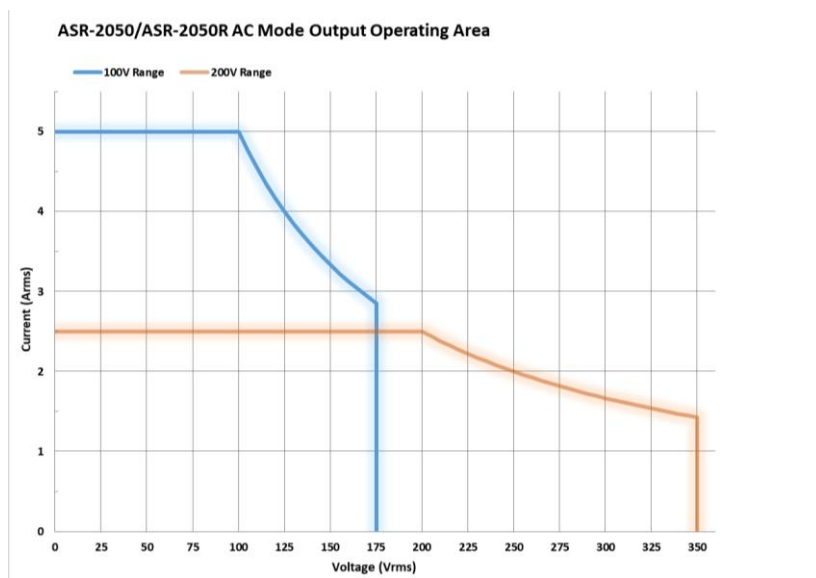
## ASR-2000 Series Overview

### Series lineup

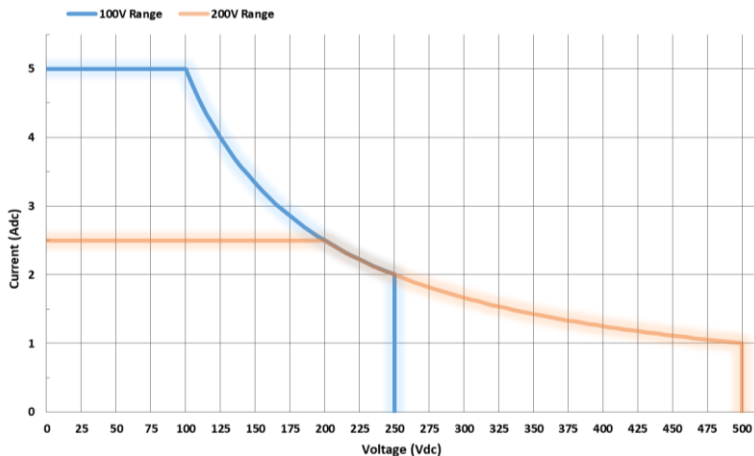
The ASR-2000 series consists of 4 models, the ASR-2050, ASR-2100, ASR-2050R and ASR-2100R, differing only in capacity and front panel output. Note that throughout the programming manual, the term “ASR-2000” refers to any of the models, unless stated otherwise.

| Model Name | Power Rating | Max. Output Current | Max. Output Voltage |
|------------|--------------|---------------------|---------------------|
| ASR-2050   | 500 VA       | 5 / 2.5 A           | 350 Vrms / 500 Vdc  |
| ASR-2100   | 1000 VA      | 10 / 5 A            | 350 Vrms / 500 Vdc  |
| ASR-2050R  | 500 VA       | 5 / 2.5 A           | 350 Vrms / 500 Vdc  |
| ASR-2100R  | 1000 VA      | 10 / 5 A            | 350 Vrms / 500 Vdc  |

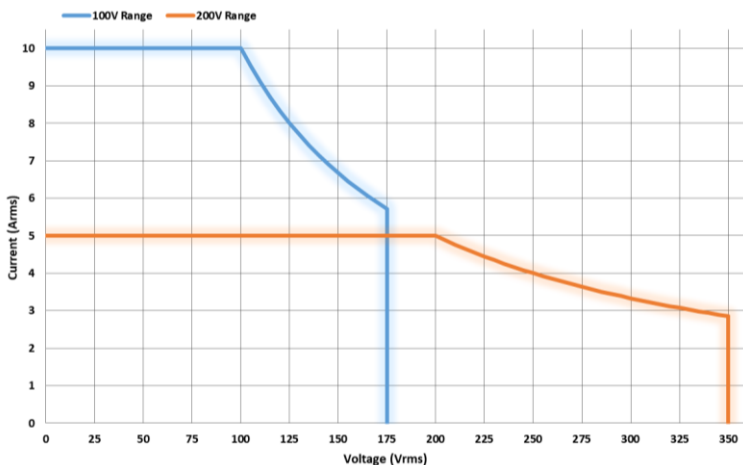
### Operating Area

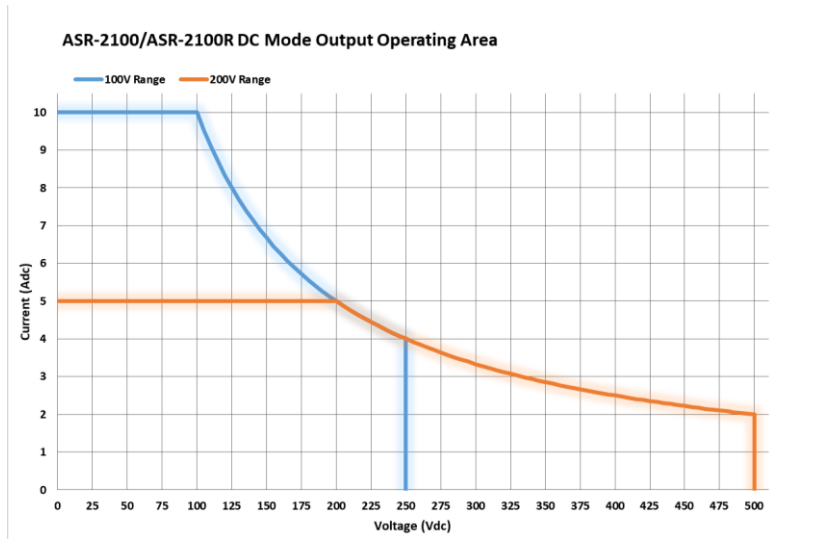


**ASR-2050/ASR-2050R DC Mode Output Operating Area**



**ASR-2100/ASR-2100R AC Mode Output Operating Area**





## Main Features

---

- Performance
- Maximum AC output voltage is 350 Vrms
  - Maximum DC output voltage is 500 Vdc
  - Maximum output frequency is 999.9 Hz in AC mode
  - Supported AC+DC waveform application
  - DC full capacity output ability
  - Output voltage total harmonic distortion is less than 0.5% at all frequency
  - Crest factor reached 4 times high
-

- Features
- Include sine, square, triangle, arbitrary and DC output waveforms
  - Variable voltage, frequency and current limiter
  - Harmonic voltage and current analysis ability
  - Excellent and feature-rich measurement capacity
  - Sequence and simulate function
  - External input amplification
  - AC line synchronized output
  - Preset memory function
  - USB memory support
  - Remote sense
  - OCP, OPP and OTP protection function

- Interface
- Built-in LAN, USB host, USB device and RS232
  - External control I/O
  - External signal input
  - Factory option GPIB interface

### Accessories

Before using the ASR-2000 power source unit, check the package contents to make sure all the standard accessories are included.

| Standard Accessories | Part number       | Description                     |
|----------------------|-------------------|---------------------------------|
|                      | CD ROM            | User manual, programming manual |
|                      | 82GW1SAFE0M*1     | Safety guide                    |
|                      | Region dependent  | Power cord                      |
|                      | 63SC-XF101601 x 1 | Mains terminal cover set        |
|                      | 63SC-XF101701 x 1 | Remote sensing cover set        |

|                           |             |  |
|---------------------------|-------------|--|
|                           | GTL-123     | Test leads: 1x red, 1x black   |
|                           | GTL-246     | USB CABLE (USB 2.0 Type A-Type B Cable, Approx. 1.2M)                        |
| Factory Installed Options | Part number | Description  |
|                           | ASR-GPIB-2K | Optional GPIB Interface for ASR-2000   |
|                           | ASR-UR-2K   | European Output Outlet only for ASR-2000                                     |
| Optional Accessories      | Part number | Description  |
|                           | GET-003     | Universal extended terminal box<br>(ASR-2000R only)                          |
|                           | GET-004     | EURO extended terminal box<br>(ASR-2000R only)                               |
|                           | GRA-439-E   | Rack mount adapter (EIA)   |
|                           | GRA-439-J   | Rack mount adapter (JIS)   |
|                           | GTL-232     | RS232C cable, approx. 2M   |
|                           | GTL-258     | An approximately 2M in length GPIB Cable including 25 pins Micro-D connector |
|                           | ASR-001     | Air inlet filter   |
|                           | ASR-002     | External Three Phase Control Unit  |
|                           | Download    | Name   |
|                           | gw_asr.inf  | USB driver   |

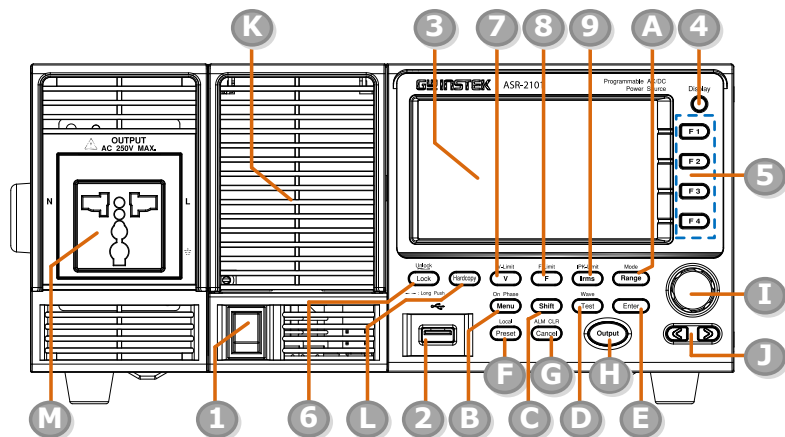


**Declaration**

For the device produced after April 2022, the RS-232 interface will be changed to a standard interface. Please confirm the actual status of supplied products by contacting the business windows of each district.

# Appearance

## Front Panel

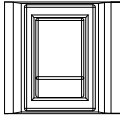


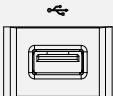
| Item Index | Description |
|------------|-------------|
|------------|-------------|


- 1 Power switch button
- 2 USB interface connector (A Type)
- 3 LCD screen
- 4 Display mode select key
- 5 Function keys (blue zone)
- 6 Lock/Unlock button
- 7 V/V-Limit button
- 8 F/F-Limit button
- 9 Irms/IPK-Limit button
- A Range key/Output mode key
- B Menu key/On phase key
- C Shift key

|   |                                    |
|---|------------------------------------|
| D | Test key/Output waveform key       |
| E | Enter key                          |
| F | Preset key/Local mode key          |
| G | Cancel key/ALM CLR key             |
| H | Output key                         |
| I | Scroll wheel                       |
| J | Arrow keys                         |
| K | Air inlet                          |
| L | Hardcopy key                       |
| M | Output socket (ASR-2100/2050 only) |


| Item | Description |
|------|-------------|
|------|-------------|

|              |   |                         |
|--------------|---|-------------------------|
| Power Switch |  | Turn on the mains power |
|--------------|---|-------------------------|

|            |  |   |
|------------|--|---|
| USB A Port |  | The USB port is used for data transfers and upgrading software. Also, it is available for screenshot hardcopy in association with the Hardcopy key. |
|------------|--|---|

|   |   |
|---|---|
|  <b>Note</b> | It supports FAT32 format with maximum 32G storage only. |
|---|---|

|            |   |
|------------|---|
| LCD Screen | Displays the setting and measured values or menu system |
|------------|---|

|                         |   |   |
|-------------------------|---|---|
| Display Mode Select Key |  | Selects between standard, simple and harmonic analysis mode |
|-------------------------|---|---|

Function Keys



Assigned to the functions displayed on the right side of the screen



Lock/Unlock Key



— : Long Push

Used to lock or unlock the front panel keys except output key. Simply press to lock, whilst long press to unlock.

Shift Key



Turns on the shift state, which enables shortcut operations with an icon **Shift** indicated on the top status bar. The shift state, which allows continuous shortcut operations, is kept until another press on shift key again.



Note

When performing shortcut operations, press shift key followed by another shortcut function key. Do not press both shift key and shortcut function key simultaneously.

V



Used for setting the output voltage

V-Limit



+



Used for setting the output voltage limit value

F



Used for setting the output frequency (DC mode N/A)

F-Limit

















+



Used for setting the output frequency limit value (DC mode N/A)



|                        |  |  |
|------------------------|--|--|
| <p>Irms</p>            | <p>IPK-Limit<br/></p>   | <p>Used for setting the maximum output current</p>   |
| <p>IPK-Limit</p>       | <p> +<br/>IPK-Limit<br/></p> | <p>Used to set the peak output current limit value</p>   |
| <p>Range Key</p>       | <p>Mode<br/></p>  | <p>Switches between the 100V, 200V and AUTO ranges</p>   |
| <p>Output Mode</p>     | <p> +<br/>Mode<br/></p>      | <p>Selects between the AC+DC-INT, AC-INT, DC-INT, AC+DC-EXT, AC-EXT, AC+DC-ADD, AC-ADD, AC+DC-Sync, AC-Sync and AC-VCA modes.</p>  |
| <p>Menu Key</p>        | <p>On Phase<br/></p>  | <p>Enters the Main menu or goes back to one of the display modes.</p>  |
| <p>On Phase</p>        | <p> +<br/>On Phase<br/></p>  | <p>Sets the on phase for the output voltage</p>  |
| <p>Test Key</p>        | <p>Wave<br/></p>  | <p>Puts the instrument into the Sequence and Simulation control mode.</p>  |
| <p>Output Waveform</p> | <p> +<br/>Wave<br/></p>  | <p>Selects between the Sine, Square, Triangle and ARB 1~16 waveforms<br/><br/>(not available for DC-INT, AC+DC-EXT and AC-EXT)</p> |
| <p>Enter Key</p>       | <p></p>   | <p>Confirms selections and settings</p>  |
| <p>Preset Key</p>      | <p>Local<br/></p>   | <p>Puts the instrument into Preset mode</p>  |

Local Mode



+

Switches operation back to local mode from remote mode

Cancel Key



Used to cancel function setting menus or dialogs.

ALM CLR



+

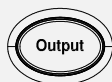
Clears alarms

Hardcopy Key



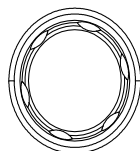
Used to take a screenshot by simply one press on the key. Make sure an USB flash disk in well inserted before the action.

Output Key



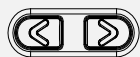
Turns the output on or off.

Scroll Wheel



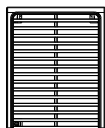
Used to navigate menu items or for increment/decrement values one step at a time.

Arrow Keys



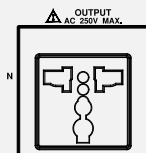
The arrow keys are used to select the digit power of a value that is being edited

Air Inlet



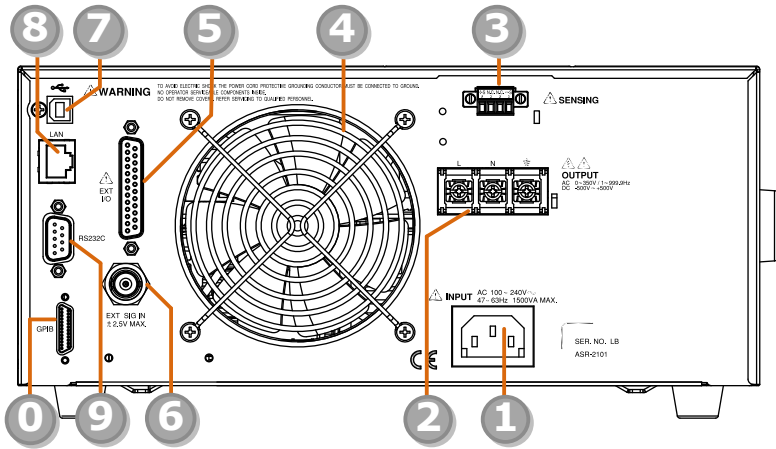
Air inlet for cooling the inside of the ASR-2000 series

Output Socket



Output voltage socket, which has 2 versions in accordance with different regions: Universal and European types, in front panel. (only available for ASR-2100/2050)

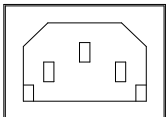
Rear Panel

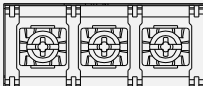



| Item Index | Description |
|------------|-------------|
|------------|-------------|

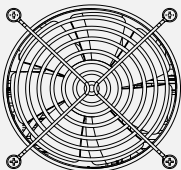
- |   |  |
|---|--|
| 1 | Line input   |
| 2 | Output terminal  |
| 3 | Remote sensing input terminal                                |
| 4 | Exhaust fan  |
| 5 | External I/O connector                                       |
| 6 | External signal input/<br>External synchronized signal input |
| 7 | USB interface connector (B Type)                             |
| 8 | Ethernet (LAN) connector                                     |
| 9 | RS232C connector   |
| 0 | ASR-GPIB-2K interface (GPIB connector)                       |

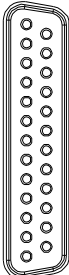
| Item | Description |
|------|-------------|
|------|-------------|

|            |   |          |
|------------|---|----------|
| Line Input |  | AC inlet |
|------------|---|----------|

|                 |   |  |
|-----------------|---|--|
| Output Terminal |  | Output voltage terminal (M3 screw type, 10 ~ 18 AWG) |
|-----------------|---|--|

|                               |   |  |
|-------------------------------|---|--|
| Remote Sensing Input Terminal |  | Compensation of load wire voltage drop. Only +S and -S are available for compensation. N.C. terminals are N/A. Refer to User Manual for details. |
|-------------------------------|---|--|

|             |   |   |
|-------------|---|---|
| Exhaust Fan |  | The exhaust fan is used to expel the heat from the unit. Please ensure there is at least 20 cm distance between any object and the fan. |
|-------------|---|---|

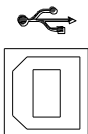
|                                |  |  |
|--------------------------------|--|--|
| External Control I/O Connector |  | Used to control ASR-2000 externally by using the logic signal and monitor Sequence function status |
|--------------------------------|--|--|

External Signal Input Connector



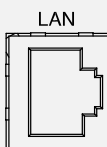
Synchronizing the output frequency with this external input signal for SYNC or outputting the amplified external signal with this external input signal for EXT, ADD and VCA

USB



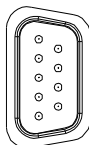
USB port for controlling the ASR-2000 remotely

Ethernet Port



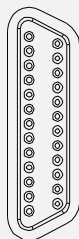
The Ethernet port is used for remote control

RS232C Connector



The RS232C connector for controlling the ASR-2000 remotely

GPIB Connector (25 pins Micro-D)



The optional GPIB connector for controlling the ASR-2000 remotely (Factory Installed ASR-GPIB-2K)

# REMOTE CONTROL

This chapter describes basic configuration of IEEE488.2 based remote control.

---

|  |     |
|--|-----|
| Interface Configuration.....                                 | 23  |
| Configure Ethernet Connection .....                          | 23  |
| USB Remote Interface.....                                    | 24  |
| USB Remote Control Function Check .....                      | 25  |
| RS232 Remote Interface .....                                 | 26  |
| RS232 Remote Control Function Check .....                    | 28  |
| Using Realterm to Establish a Remote Connection .....        | 29  |
| GPIB Remote Interface (ASR-GPIB-2K Option).....              | 32  |
| GPIB Function Check.....                                     | 33  |
| Web Server Remote Control Function Check.....                | 35  |
| Socket Server Function Check .....                           | 36  |
| Command Syntax.....  | 41  |
| Command List.....  | 45  |
| Status Register Overview.....                                | 140 |
| Introduction to the Status Registers.....                    | 140 |
| The Status Registers .....                                   | 141 |
| Questionable Status Register Group.....                      | 142 |
| Operation Status Register Group.....                         | 145 |
| Warning Status Register Group .....                          | 147 |
| System Lock Status Register Group.....                       | 150 |
| Standard Event Status Register Group.....                    | 152 |
| Status Byte Register & Service Request Enable Register ..... | 154 |
| Error List.....  | 156 |
| Command Errors.....  | 156 |
| Execution Errors .....                                       | 159 |
| Device Specific Errors.....                                  | 161 |
| Query Errors .....   | 162 |

## Interface Configuration

### Configure Ethernet Connection



The Ethernet interface can be configured for a number of different applications. Ethernet can be configured for basic remote control or monitoring using a web server or it can be configured as a socket server.

The ASR-2000 supports both DHCP connections so the instrument can be automatically connected to an existing network or alternatively, network settings can be manually configured.

---


|                     |                            |                           |
|---------------------|----------------------------|---------------------------|
| Ethernet Parameters | MAC Address (display only) | DHCP                      |
|                     | IP Address                 | Subnet mask               |
|                     | Gateway                    | DNS address               |
|                     | DNS Server                 | Socket port fixed at 2268 |

---

- |                        |   |  |
|------------------------|---|--|
| Ethernet Configuration | 1. Connect a LAN cable from the PC to the Ethernet port on the rear panel.  |   |
|                        | 2. Press the <i>Menu</i> key. The Menu setting will appear on the display.  |  |
|                        | 3. Use the scroll wheel to go to item 3, <i>LAN</i> and press <i>Enter</i> .  |  |
|                        | 4. If the LAN cable is installed correctly a connection is active, the <i>Connection Status</i> will show <i>Online</i> .                   |  |
|                        | 5. To automatically have the network assign an IP address, set DHCP to ON. Otherwise set DHCP to OFF to manually set the Ethernet settings. |  |





2. Press the *Menu* key. The Menu setting will appear on the display. 
3. Use the scroll wheel to go to item 4, *USB Device*
4. If the connection is successful *Connection Status* will change from Offline to Online.

Exit

5. Press *Exit[F4]* to exit from the rear panel USB settings.



## USB Remote Control Function Check

---

Functionality  
Check

Invoke a terminal application such as Realterm. ASR-2000 will appear as a COM port on the PC.

To check the COM settings in Windows, see the Device Manager. For example, in Win7 go to the Control panel → System → Hardware tab.



Note

If you are not familiar with using a terminal application to send/receive remote commands via a USB connection, please see page 29 for more information.

Run this query command via the terminal after the instrument has been configured for USB remote control (page 24).

\*IDN?

This should return the Manufacturer, Model number, Serial number, and Software version in the following format.

GW-INSTEK,ASR-XXXX,GXXXXXXXXX, XX.XX

Manufacturer: GW-INSTEK

Model number : ASR-XXXX

Serial number : GXXXXXXXXX

Software version : XX.XX



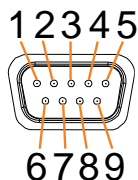
Note

For further details, please see the programming manual, available on the GW Instek web site @ [www.gwinstek.com](http://www.gwinstek.com).

## RS232 Remote Interface

|                     |            |  |
|---------------------|------------|--|
| RS232 Configuration | Connector  | BD-9, male                               |
|                     | Parameters | Baud rate, data bits, parity, stop bits. |

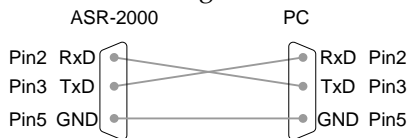
Pin Assignment



- 2: RxD (Receive data)
- 3: TxD (Transmit data)
- 5: GND
- 4, 6 ~ 9: No connection

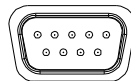
Pin Connection

Use a Null Modem connection (RS232C cable) as shown in the diagram below.



Steps

1. Connect the RS232C cable from the PC to the rear panel RS232 port.



2. Press the *Menu* key. The Menu setting will appear on the display.



3. Use the scroll wheel to go to item 5, RS232C and press *Enter*.
4. Set the RS232C relative settings.

|           |   |
|-----------|---|
| Baud rate | 1200, 2400, 4800,<br>9600(default), 19200,<br>38400, 57600, 115200, |
| Data bits | 7 bits, 8 bits(default)   |
| Parity    | None(default), Odd, Even  |
| Stop bits | 1 bit(default), 2 bits  |

**RS232C Configuration**



Exit

5. Press *Exit*[F4] to exit from the RS232C settings.



 **Note**

The standard accessory does Not include RS232 data cable. Please purchase the additional GTL-232 which will meet your need for RS232 connection.

## RS232 Remote Control Function Check

---

### Functionality Check

Invoke a terminal application such as Realterm.

For RS232, set the COM port, baud rate, stop bit, data bit and parity accordingly.

To check the COM settings in Windows, see the Device Manager. For example, in Win7 go to the Control panel → System → Hardware tab.

---



### Note

If you are not familiar with using a terminal application to send/receive remote commands from the serial port, please see page 29 for more information.

---

Run this query command via the terminal after the instrument has been configured for RS232 remote control (page 26).

\*IDN?

This should return the Manufacturer, Model number, Serial number, and Software version in the following format.

GW-INSTEK,ASR-XXXX,GXXXXXXXX,XX.XX

Manufacturer: GW-INSTEK

Model number : ASR-XXXX

Serial number : GXXXXXXXX

Software version : XX.XX

---



### Note

For further details, please see the programming manual, available on the GW Instek web site @ [www.gwinstek.com](http://www.gwinstek.com).

---

## Using Realterm to Establish a Remote Connection

---

**Background** Realterm is a terminal program that can be used to communicate with a device attached to the serial port of a PC or via an emulated serial port via USB.

The following instructions apply to version 2.0.0.70. Even though Realterm is used as an example to establish a remote connection, any terminal program can be used that has similar functionality.

---



Note

Realterm can be downloaded on Sourceforge.net free of charge.

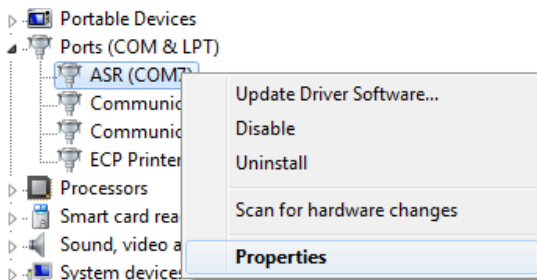
For more information please see  
<http://realterm.sourceforge.net/>

---

- Operation**
1. Download Realterm and install according to the instructions on the Realterm website.
  2. Connect the ASR-2000 via USB (page 23) or via RS232 (page 25).
  3. If using RS232, make note of the configured baud rate, stop bits and parity.
  4. Go to the Windows device manager and find the COM port number for the connection. For example, go to the Start menu > Control Panel > Device Manager.

Double click the *Ports* icon to reveal the connected serial port devices and the COM port for the each connected device.

If using USB, the baud rate, stop bit and parity settings can be viewed by right-clicking the connected device and selecting the *Properties* option.



5. Start Realterm on the PC as an administrator.  
Click:  
Start menu>All Programs>RealTerm>realterm

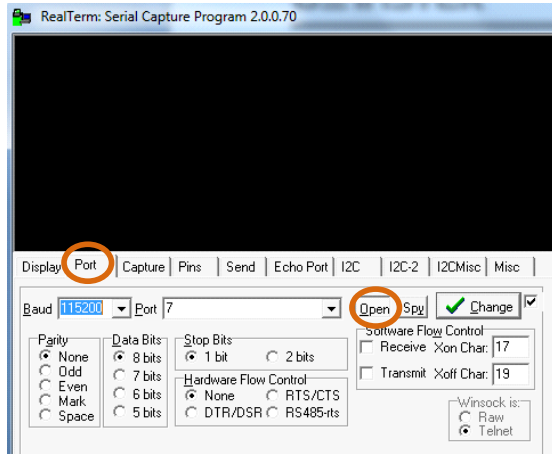
Tip: to run as an administrator, you can right click the Realterm icon in the Windows Start menu and select the *Run as Administrator* option.

6. After Realterm has started, click on the *Port* tab.

Enter the *Baud*, *Parity*, *Data bits*, *Stop bits* and *Port* number configuration for the connection.

The *Hardware Flow Control*, *Software Flow Control* options can be left at the default settings.

7. Press *Open* to connect to the ASR-2000.



Note

For USB, the baud rate should be fixed to 115,200.

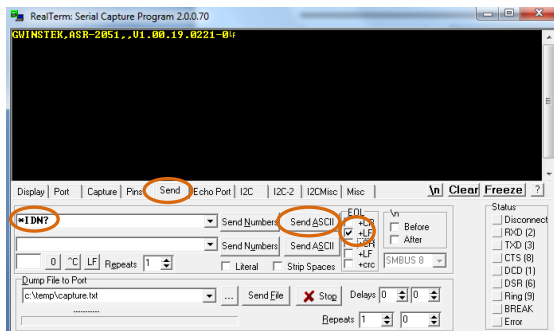
8. Click on the *Send* tab.

In the *EOL* configuration, check on the *+LF* check boxes.

Enter the query:

*\*idn?*

Click on *Send ASCII*.



9. The terminal display will return the following:

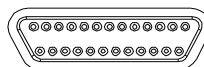
GW-INSTEK,ASR-XXXX,GXXXXXXXXX,XX.XX  
(manufacturer, model, serial number, software version)

10. If Realterm fails to connect to the ASR-2000, please check all the cables and settings and try again.

## GPIB Remote Interface (ASR-GPIB-2K Option)

GPIB Configuration

1. Connect a GPIB cable from the PC to the GPIB port on the rear panel.

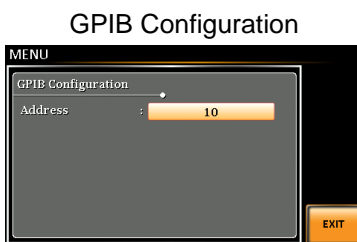


2. Press the *Menu* key. The Menu setting will appear on the display.



3. Use the scroll wheel to go to item 6, *GPIB* and press *Enter*.
4. Set the GPIB address.

GPIB Address            0 ~ 30 (10 by default)



Note

Only one GPIB address can be used at a time.

Exit

5. Press *Exit*[F4] to exit from the GPIB settings.





- GPIB Constraints
- Maximum 15 devices altogether, 20m cable length, 2m between each device
  - Unique address assigned to each device
  - At least 2/3 of the devices turned On
  - No loop or parallel connection



Note

The standard accessory does Not include GPIB data cable. Please purchase the additional GTL-258 which will meet your need for GPIB connection.

## GPIB Function Check

Functionality Check      Please use the National Instruments Measurement & Automation Controller software to confirm GPIB/LAN functionality.

See the National Instrument website,  
<http://www.ni.com> for details.



Note

- For further details, please see the programming manual, available on the GW Instek web site @ [www.gwinstek.com](http://www.gwinstek.com)
- Operating System: Windows XP, 7, 8, 10

Operation

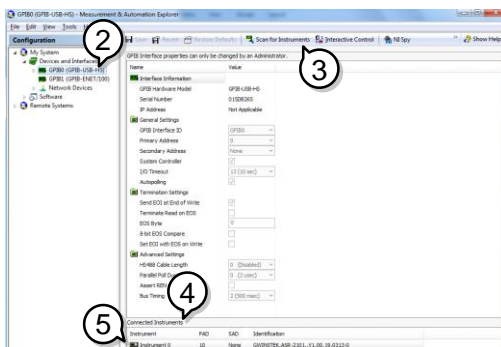
1. Start the NI Measurement and Automation Explorer (MAX) program. Using Windows, press:



*Start>All Programs>NI MAX*



2. From the Configuration panel access;  
My System>Devices and Interfaces>GPIB0
3. Press the *Scan for Instruments* button.
4. In the *Connected Instruments* panel the ASR-2000 should be detected as *Instrument 0* with the address the same as that configured on the ASR-2000.
5. Double click the *Instrument 0* icon.

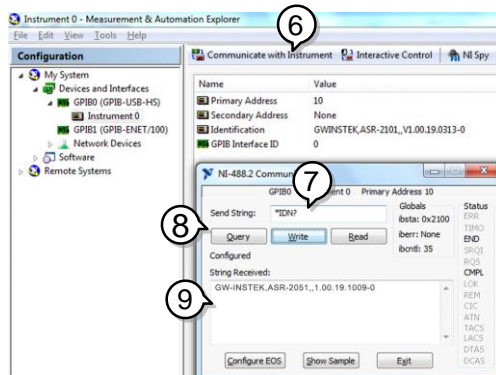


6. Click on *Communicate with Instrument*.
7. Under the Communicator tab, ensure *\*IDN?* is written in the *Send String* text box.

8. Click on the *Query* button to send the \*IDN? query to the instrument.
9. The instrument identification string will be returned to the buffer area:

GW-INSTEK,ASR-XXXX,GXXXXXXXXX,XX.XX

(manufacturer, model, serial number, software version)



10. The function check is complete.

## Web Server Remote Control Function Check

Functionality Check

Enter the IP address of the power supply (for example: [http:// XXX.XXX.XXX.XXX](http://XXX.XXX.XXX.XXX)) in a web browser after the instrument has been configured for LAN (page 23).

The web interface allows you to:

- View the system and information and the network configuration.
- View the analog control pinout.

- View the dimensions of the unit.
- View the operating area

Example:

The screenshot shows the GW INSTEK website header with the logo and navigation links: "Simply Reliable", "Visit Our Site", "Support", and "Contact Us". Below the header is a navigation menu with the following items: "Welcome Page", "Network Configuration", "Analog Control", "Figure of Dimensions", and "Operating Area". The "Network Configuration" section is expanded, showing a form with the following fields: "IP Address" (192.168.0.145), "Subnet Mask" (255.255.255.0), "Gateway" (192.168.0.1), and "DNS" (172.16.1.252). The "Analog Control" section shows "DHCP State" with radio buttons for "ON" and "OFF", where "OFF" is selected. The "Figure of Dimensions" section shows a "Password" field. The "Operating Area" section has a "Submit" button.

## Socket Server Function Check

**Background** To test the socket server functionality, National Instruments Measurement and Automation Explorer can be used. This program is available on the NI website, [www.ni.com](http://www.ni.com), via a search for the VISA Run-time Engine page, or “downloads” at the following URL, <http://www.ni.com/visa/>

**Requirements** Operating System: Windows XP, 7, 8, 10

**Functionality Check**

1. Start the NI Measurement and Automation Explorer (MAX) program. Using Windows, press:

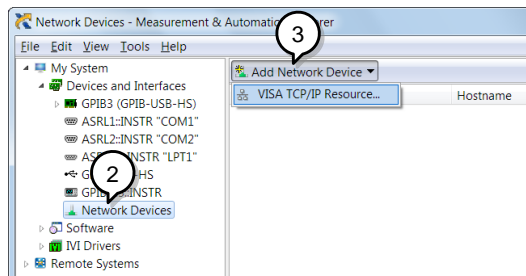
*Start>All Programs>NI MAX*



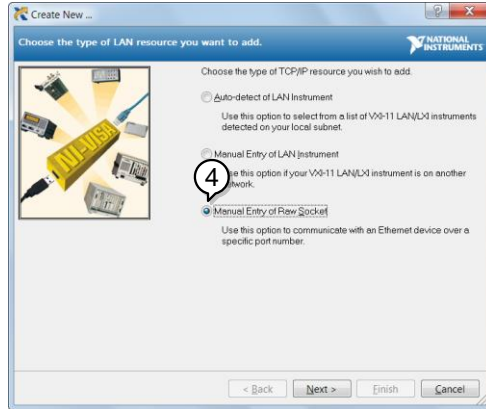
2. From the Configuration panel access;

*My System>Devices and Interfaces>Network Devices*

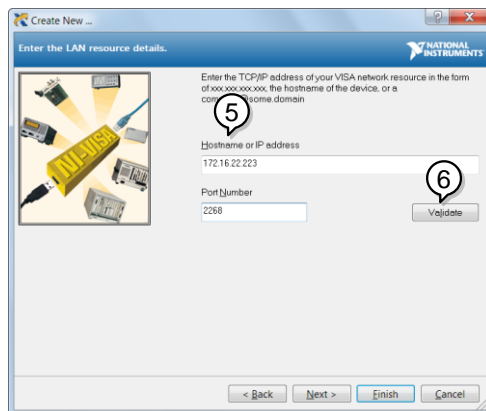
3. Press *Add New Network Device>Visa TCP/IP Resource...*



4. Select *Manual Entry of Raw Socket* from the popup window.

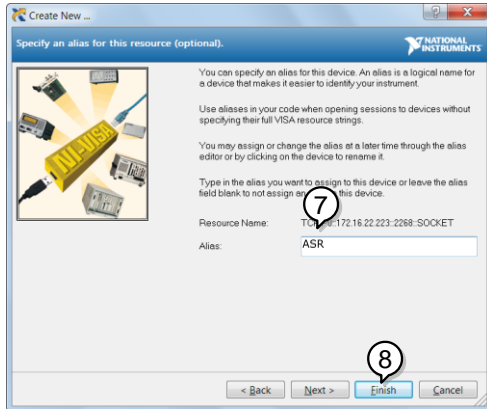


5. Enter the IP address and the port number of the ASR-2000. The port number is fixed at 2268.
6. Double click the Validate button and press *Next*.



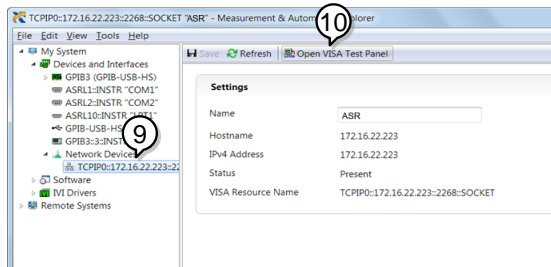
7. Next configure the Alias (name) of the ASR-2000 connection. In this example the Alias is: ASR

8. Click finish.

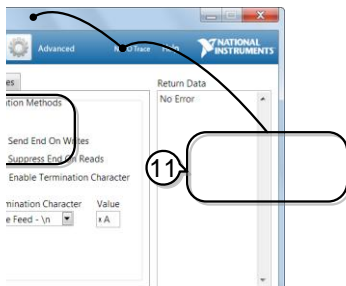


9. The IP address of the power supply will now appear under Network Devices in the configuration panel. Select this icon now.

10. Press *Open VISA Test Panel*.



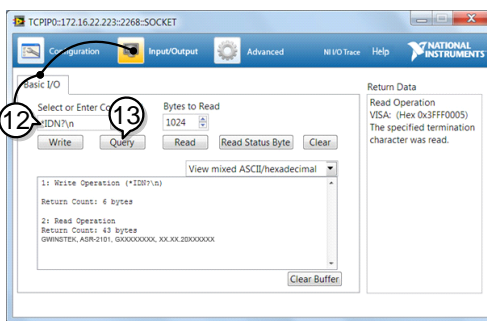
11. Click the *Configuration* Icon. Under the *IO Settings* tab check *Enable Termination Character*. The termination character should be set as *Line Feed - \n*.



12. Click the *Input/Output* icon. Under the *Basic I/O* tab, make sure *\*IDN?\n* is entered in the *Select or Enter Command* drop box.

13. Click *Query*.

The ASR-2000 will return the machine identification string into the buffer area:  
 GW-INSTEK,ASR-XXXX,GXXXXXXXXX,XX.XX



Note

For further details, please see the programming manual, available on the GW Instek web site @ [www.gwinstek.com](http://www.gwinstek.com).

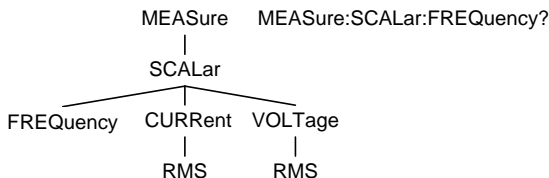


## Command Syntax

|                     |            |                       |
|---------------------|------------|-----------------------|
| Compatible Standard | IEEE488.2  | Partial compatibility |
|                     | SCPI, 1999 | Partial compatibility |

**Command Structure** SCPI commands follow a tree-like structure, organized into nodes. Each level of the command tree is a node. Each keyword in a SCPI command represents each node in the command tree. Each keyword (node) of a SCPI command is separated by a colon (:).

For example, the diagram below shows an SCPI sub-structure and a command example.



**Command types** There are a number of different instrument commands and queries. A command sends instructions or data to the unit and a query receives data or status information from the unit.

### Command types

**Simple** A single command with/without a parameter

**Example** \*IDN?

Query                      A query is a simple or compound command followed by a question mark (?). A parameter (data) is returned.

Example                    meas:curr?


Compound                Two or more commands on the same command line. Compound commands are separated with either a semi-colon (;) or a semi-colon and a colon (;:).

A semi-colon is used to join two related commands, with the caveat that the last command must begin at the last node of the first command.

Example                    meas:volt?;curr?

A semi-colon and colon are used to combine two commands from different nodes.

Example                    meas:volt?;;sour:volt?

 Note  
(Further explanation)

A semi-colon(;) is used to connect two commands. A colon(:) at the start of a command indicates that the command starts from the root node. The first command can ignore that first colon. Any commands after the first command (for compound commands) that do not begin with a colon, must begin at the last node of the first command.

**Command Forms** Commands and queries have two different forms, long and short. The command syntax is written with the short form of the command in capitals and the remainder (long form) in lower case.

The commands can be written in capitals or lower-case, just so long as the short or long forms are complete. An incomplete command will not be recognized.

Below are examples of correctly written commands.

---

Long form :SYSTem:ERRor?  
           :SYSTEM:ERROR?  
           :system:error?

---

Short form SYST:ERR?  
           syst:err?

---

**Square Brackets** Commands that contain square brackets indicate that the contents are optional. The function of the command is the same with or without the square bracketed items, as shown below.

For example the query “:OUTPut[:STATe]?” has two valid forms, “:OUTPut:STATe?” and “:OUTPut?”.

|                |  |   |
|----------------|--|---|
| Command Format |  | <ol style="list-style-type: none"> <li>1. Command header</li> <li>2. Space</li> <li>3. Parameter 1</li> <li>4. Comma (no space before/after comma)</li> <li>5. Parameter 2</li> </ol> |
|----------------|--|---|

---

| Parameters | Type      | Description   | Example |
|------------|-----------|---------------|---------|
|            | <Boolean> | Boolean logic | 0, 1    |

|              |  |                 |
|--------------|--|-----------------|
| <NR1>        | integers   | 0, 1, 2, 3      |
| <NR2>        | decimal numbers  | 0.1, 3.14, 8.5  |
| <NR3>        | floating point   | 4.5e-1, 8.25e+1 |
| <NRf>        | any of NR1, 2, 3   | 1, 1.5, 4.5e-1  |
| <block data> | Definitive length arbitrary block data. A single decimal digit followed by data. The decimal digit specifies how many 8-bit data bytes follow. |                 |

---

|                    |    |                |
|--------------------|----|----------------|
| Message Terminator | LF | Line feed code |
|--------------------|----|----------------|

---

## Command List

---

|            |  |    |
|------------|--|----|
| Common     | *CLS.....                                      | 51 |
| Commands   | *ESE.....                                      | 51 |
|            | *ESR.....                                      | 52 |
|            | *IDN.....                                      | 52 |
|            | *OPC.....                                      | 52 |
|            | *RCL.....                                      | 53 |
|            | *RST.....                                      | 53 |
|            | *SAV.....                                      | 53 |
|            | *SRE.....                                      | 54 |
|            | *STB.....                                      | 54 |
|            | *WAI.....                                      | 54 |
| Data/Trace | :DATA   TRACe:SEQuence:CLear.....              | 55 |
| Commands   | :DATA   TRACe:SEQuence:RECall.....             | 55 |
|            | :DATA   TRACe:SEQuence:STORe.....              | 56 |
|            | :DATA   TRACe:SIMulation:CLear.....            | 56 |
|            | :DATA   TRACe:SIMulation:RECall.....           | 57 |
|            | :DATA   TRACe:SIMulation:STORe.....            | 57 |
|            | :DATA   TRACe:WAVe:CLear.....                  | 57 |
|            | :DATA   TRACe:WAVe[:DATA].....                 | 58 |
| Measure    | :MEASure[:SCALar]:CURRent:CFACtor.....         | 59 |
| Commands   | :MEASure[:SCALar]:CURRent:HIGH.....            | 59 |
|            | :MEASure[:SCALar]:CURRent:LOW.....             | 60 |
|            | :MEASure[:SCALar]:CURRent:PEAK:CLear.....      | 60 |
|            | :MEASure[:SCALar]:CURRent:PEAK:HOLD.....       | 60 |
|            | :MEASure[:SCALar]:CURRent[:RMS].....           | 60 |
|            | :MEASure[:SCALar]:CURRent:AVERAge.....         | 61 |
|            | :MEASure[:SCALar]:CURRent:HARMonic[:RMS]... .. | 61 |
|            | :MEASure[:SCALar]:CURRent:HARMonic:RATio... .. | 61 |
|            | :MEASure[:SCALar]:FREQuency.....               | 61 |
|            | :MEASure[:SCALar]:POWer[:AC]:APParent.....     | 62 |
|            | :MEASure[:SCALar]:POWer[:AC]:PFACTor.....      | 62 |
|            | :MEASure[:SCALar]:POWer[:AC]:REACTive.....     | 62 |

|          |   |    |
|----------|---|----|
|          | :MEASure[:SCALar]:POWER[:AC][:REAL] .....     | 62 |
|          | :MEASure[:SCALar]:VOLTage[:RMS] .....         | 62 |
|          | :MEASure[:SCALar]:VOLTage:AVERage .....       | 63 |
|          | :MEASure[:SCALar]:VOLTage:HIGH .....          | 63 |
|          | :MEASure[:SCALar]:VOLTage:LOW .....           | 63 |
|          | :MEASure[:SCALar]: VOLTage:HARMonic[:RMS].... | 63 |
|          | :MEASure[:SCALar]: VOLTage:HARMonic:RATio ... | 64 |
|          | :MEASure:CONFigure:SENsing .....              | 64 |
|          | :MEASure:AVERage:COUNt.....                   | 65 |
|          | :MEASure:UPDate:RATE .....                    | 65 |
| Memory   | :MEMory:RCL.....                              | 67 |
| Commands | :MEMory:SAV.....                              | 67 |
| Output   | :OUTPut[:STATe] .....                         | 68 |
| Commands | :OUTPut:PON .....                             | 68 |
|          | :OUTPut:PROTection:CLEar .....                | 69 |
|          | :OUTPut:RELAy .....                           | 69 |
| Status   | :STATus:OPERation:CONDition.....              | 70 |
| Commands | :STATus:OPERation:ENABle.....                 | 71 |
|          | :STATus:OPERation[:EVENT] .....               | 71 |
|          | :STATus:OPERation:NTRansition .....           | 71 |
|          | :STATus:OPERation:PTRansition .....           | 71 |
|          | :STATus:QUEStionable[:EVENT].....             | 72 |
|          | :STATus:QUEStionable:CONDition .....          | 72 |
|          | :STATus:QUEStionable:ENABle .....             | 72 |
|          | :STATus:QUEStionable:NTRansition .....        | 73 |
|          | :STATus:QUEStionable:PTRansition .....        | 73 |
|          | :STATus:PRESet.....                           | 73 |
|          | :STATus:WARNIing:CONDition .....              | 74 |
|          | :STATus:WARNIing:ENABle .....                 | 74 |
|          | :STATus:WARNIing[:EVENT].....                 | 75 |
|          | :STATus:WARNIing:NTRansition .....            | 75 |
|          | :STATus:WARNIing:PTRansition .....            | 75 |
|          | :STATus:LOCK:CONDition.....                   | 76 |
|          | :STATus:LOCK:ENABle .....                     | 76 |

|          |   |    |
|----------|---|----|
|          | :STATus:LOCK[:EVENT]                    | 76 |
|          | :STATus:LOCK:NTRansition                | 76 |
|          | :STATus:LOCK:PTRansition                | 77 |
| System   | :SYSTem:ACIN:DETection                  | 80 |
| Commands | :SYSTem:ARBITrary:EDIT:BUIlTin          | 80 |
|          | :SYSTem:ARBITrary:EDIT:SURGe            | 81 |
|          | :SYSTem:ARBITrary:EDIT:STAir            | 81 |
|          | :SYSTem:ARBITrary:EDIT:CFACtor2         | 82 |
|          | :SYSTem:ARBITrary:EDIT:CFACtor1         | 82 |
|          | :SYSTem:ARBITrary:EDIT:CLIP             | 83 |
|          | :SYSTem:ARBITrary:EDIT:STORe            | 83 |
|          | :SYSTem:ARBITrary:EDIT:TRIngle          | 84 |
|          | :SYSTem:ARBITrary:EDIT:RIPple           | 85 |
|          | :SYSTem:ARBITrary:EDIT:STORe:APPLy      | 86 |
|          | :SYSTem:BEEPer:STATe                    | 87 |
|          | :SYSTem:COMMunicate:GPIB[:SELF]:ADDReSS | 87 |
|          | :SYSTem:COMMunicate:LAN:DHCP            | 87 |
|          | :SYSTem:COMMunicate:LAN:DNS             | 88 |
|          | :SYSTem:COMMunicate:LAN:GATeway         | 88 |
|          | :SYSTem:COMMunicate:LAN:IPADdress       | 89 |
|          | :SYSTem:COMMunicate:LAN:MAC             | 89 |
|          | :SYSTem:COMMunicate:LAN:SMASk           | 89 |
|          | :SYSTem:COMMunicate:RLState             | 90 |
|          | :SYSTem:COMMunicate:SERial[:RECeive]    |    |
|          | :TRANsmit:BAUD                          | 90 |
|          | :SYSTem:COMMunicate:SERial[:RECeive]    |    |
|          | :TRANsmit:BITS                          | 91 |
|          | :SYSTem:COMMunicate:SERial[:RECeive]    |    |
|          | :TRANsmit:PARity                        | 92 |
|          | :SYSTem:COMMunicate:SERial[:RECeive]    |    |
|          | :TRANsmit:SBITS                         | 92 |
|          | :SYSTem:COMMunicate:TCPIP:CONTRol       | 93 |
|          | :SYSTem:COMMunicate:USB:FRONT:STATe     | 93 |
|          | :SYSTem:COMMunicate:USB:REAR:STATe      | 94 |
|          | :SYSTem:CONFigure[:MODE]                | 94 |
|          | :SYSTem:CONFigure:EXTio[:STATe]         | 95 |
|          | :SYSTem:CONFigure:TRIGger:OUTPut:WIDTh  |    |

|          |   |     |
|----------|---|-----|
|          | .....   | 95  |
|          | :SYSTem:CONFIgure:TRIGger:OUTPut:SOURce       |     |
|          | .....   | 96  |
|          | :SYSTem:ERRor .....                           | 97  |
|          | :SYSTem:ERRor:ENABle .....                    | 97  |
|          | :SYSTem:HOLD:STATe .....                      | 97  |
|          | :SYSTem:IPKHold:TIME .....                    | 98  |
|          | :SYSTem:KLOCK .....                           | 98  |
|          | :SYSTem:REBooT.....                           | 98  |
|          | :SYSTem:SLEW:MODE.....                        | 99  |
|          | :SYSTem:VUNit .....                           | 99  |
| Source   | [:SOURce]:CURRent:LIMit:PEAK:HIGH .....       | 100 |
| Commands | [:SOURce]:CURRent:LIMit:PEAK:LOW .....        | 101 |
|          | [:SOURce]:CURRent:LIMit:RMS[:AMPLitude] ..... | 101 |
|          | [:SOURce]:CURRent:LIMit:PEAK:MODE .....       | 102 |
|          | [:SOURce]:CURRent:LIMit:RMS:MODE.....         | 102 |
|          | [:SOURce]:FREQuency:LIMit:HIGH .....          | 103 |
|          | [:SOURce]:FREQuency:LIMit:LOW .....           | 103 |
|          | [:SOURce]:FREQuency[:IMMEDIATE] .....         | 104 |
|          | [:SOURce]:FUNctIon[:SHAPE][:IMMEDIATE] ....   | 104 |
|          | [:SOURce]:FUNctIon:THD:FORMat .....           | 105 |
|          | [:SOURce]:MODE .....                          | 106 |
|          | [:SOURce]:PHASe:STARt:STATe .....             | 107 |
|          | [:SOURce]:PHASe:STOP:STATe.....               | 107 |
|          | [:SOURce]:PHASe:STARt[:IMMEDIATE] .....       | 108 |
|          | [:SOURce]:PHASe:STOP[:IMMEDIATE] .....        | 108 |
|          | [:SOURce]:PHASe:SYNC[:IMMEDIATE].....         | 109 |
|          | [:SOURce]:READ .....                          | 109 |
|          | [:SOURce]:VOLTagE:RANGe .....                 | 110 |
|          | [:SOURce]:VOLTagE:LIMit:RMS .....             | 110 |
|          | [:SOURce]:VOLTagE:LIMit:PEAK.....             | 111 |
|          | [:SOURce]:VOLTagE:LIMit:HIGH .....            | 111 |
|          | [:SOURce]:VOLTagE:LIMit:LOW .....             | 112 |
|          | [:SOURce]:VOLTagE[:LEVel][:IMMEDIATE]         |     |
|          | [:AMPLitude] .....                            | 112 |
|          | [:SOURce]:VOLTagE[:LEVel][:IMMEDIATE]:OFFSet  |     |
|          | .....   | 113 |



|   |   |     |
|---|---|-----|
| Sequence                                      | [.:SOURce]:SEQuence:CPARAmeter .....                        | 114 |
| Commands                                      | [.:SOURce]:SEQuence:CSTep .....                             | 115 |
|   | [.:SOURce]:SEQuence:CJUMp:CNT .....                         | 115 |
|   | [.:SOURce]:SEQuence:CTIME .....                             | 116 |
|   | [.:SOURce]:SEQuence:SPARAmeter .....                        | 116 |
|   | [.:SOURce]:SEQuence:STEP .....                              | 117 |
|   | [.:SOURce]:SEQuence:CONDition .....                         | 118 |
|   | :TRIGger:SEQuence:SELEcted:EXECute .....                    | 118 |
| <br>  |   |     |
| Simulate                                      | [.:SOURce]:SIMulation:CONDition .....                       | 120 |
| Commands                                      | [.:SOURce]:SIMulation:ABNormal:CODE .....                   | 120 |
|   | [.:SOURce]:SIMulation:ABNormal:FREQuency .....              | 121 |
|   | [.:SOURce]:SIMulation:ABNormal:PHASe                        |     |
|   | :STARt:ENABle .....   | 121 |
|   | [.:SOURce]:SIMulation:ABNormal:PHASe                        |     |
|   | :STARt[:IMMEdiate] .....                                    | 122 |
|   | [.:SOURce]:SIMulation:ABNormal:PHASe                        |     |
|   | :STOP:ENABle .....  | 122 |
|   | [.:SOURce]:SIMulation:ABNormal:PHASe                        |     |
|   | :STOP[:IMMEdiate] .....                                     | 123 |
|   | [.:SOURce]:SIMulation:ABNormal:TIME .....                   | 123 |
|   | [.:SOURce]:SIMulation:ABNormal:VOLTage .....                | 124 |
|   | [.:SOURce]:SIMulation:CSTep .....                           | 124 |
|   | [.:SOURce]:SIMulation:CREPeat:COUNT .....                   | 125 |
|   | [.:SOURce]:SIMulation:CTIME .....                           | 125 |
|   | [.:SOURce]:SIMulation:INITial:CODE .....                    | 126 |
|   | [.:SOURce]:SIMulation:INITial:FREQuency .....               | 126 |
|   | [.:SOURce]:SIMulation:INITial:PHASe:STARt:ENABle .....      | 127 |
|   | [.:SOURce]:SIMulation:INITial:PHASe:STARt[:IMMEdiate] ..... | 127 |
|   | [.:SOURce]:SIMulation:INITial:PHASe:STOP :ENABle .....      | 128 |
|   | [.:SOURce]:SIMulation:INITial:PHASe:STOP[:IMMEdiate] .....  | 128 |
|   | [.:SOURce]:SIMulation:INITial:VOLTage .....                 | 129 |
|   | [.:SOURce]:SIMulation:NORMAl<1   2>:CODE .....              | 129 |
| [.:SOURce]:SIMulation:NORMAl1:FREQuency ..... | 130   |     |

|                |  |     |
|----------------|--|-----|
|                | :SOURce]:SIMulation:NORMal<1   2>          |     |
|                | :PHASe:STARt:ENABle .....                  | 130 |
|                | :SOURce]:SIMulation:NORMal<1   2>          |     |
|                | :PHASe:STARt[:IMMEdiate] .....             | 131 |
|                | :SOURce]:SIMulation:NORMal<1   2>          |     |
|                | :PHASe:STOP:ENABle .....                   | 131 |
|                | :SOURce]:SIMulation:NORMal<1   2>          |     |
|                | :PHASe:STOP[:IMMEdiate].....               | 132 |
|                | :SOURce]:SIMulation:NORMal<1   2>:TIME.... | 132 |
|                | :SOURce]:SIMulation:NORMal1:VOLTage.....   | 133 |
|                | :SOURce]:SIMulation:REPeat:COUNT .....     | 133 |
|                | :SOURce]:SIMulation:REPeat:ENABle.....     | 134 |
|                | :SOURce]:SIMulation:TRANsition<1   2>:TIME |     |
|                | .....                                      | 134 |
|                | :SOURce]:SIMulation:TRANsition<1   2>:CODE |     |
|                | .....                                      | 135 |
|                | :TRIGger:SIMulation:SELeCted:EXECute.....  | 135 |
| Input Commands | :INPut:GAIN .....                          | 136 |
|                | :INPut:SYNC:SOURce .....                   | 136 |
| Display        | :DISPlay[:WINDow]:DESIgn:MODE .....        | 138 |
| Commands       | :DISPlay[:WINDow]:MEASure                  |     |
|                | :SOURce<1   2   3> .....                   | 138 |

## Common Commands

|           |    |
|-----------|----|
| *CLS..... | 51 |
| *ESE..... | 51 |
| *ESR..... | 52 |
| *IDN..... | 52 |
| *OPC..... | 52 |
| *RCL..... | 53 |
| *RST..... | 53 |
| *SAV..... | 53 |
| *SRE..... | 54 |
| *STB..... | 54 |
| *WAI..... | 54 |

### \*CLS

Set →

**Description**      The \*CLS command clears all the event registers, including the status byte, event status and error queue.

**Syntax**            \*CLS

### \*ESE

Set →  
→ Query

**Description**      Sets or queries the Standard Event Status Enable register.

**Syntax**            \*ESE <NR1>

**Query Syntax**    \*ESE?

**Parameter**        <NR1>    0~255

**Return parameter** <NR1>    Returns the bit sum of the Standard Event Status Enable register.

**\*ESR** → Query

**Description**      Queries the Standard Event Status (Event) register. The Event Status register is cleared after it is read.

**Query Syntax**      \*ESR?

**Return parameter** <NR1>      Returns the bit sum of the Standard Event Status (Event) register and clears the register.

**\*IDN** → Query

**Description**      Queries the manufacturer, model name, serial number, and firmware version of the ASR.

**Query Syntax**      \*IDN?

**Return parameter** <string>      Returns the instrument identification as a string in the following format:  
 GW-INSTEK,ASR-XXXX,GXXXXXXXXX,  
 XX.XX  
 Manufacturer: GW-INSTEK  
 Model number : ASR-XXXX  
 Serial number : GXXXXXXXXX  
 Firmware version : XX.XX

**\*OPC** → Query

**Description**      The \*OPC? Query returns 1 when all the outstanding commands have completed.

**Query Syntax**      \*OPC?

**Return parameter** 1      Returns 1 when all the outstanding commands have completed.

**\*RCL**

Set →

---

|             |   |                                 |
|-------------|---|---------------------------------|
| Description | Recalls the contents stored in memory slot M0 ~ M9. These memory slots are mapped to the preset settings. |                                 |
| Syntax      | *RCL {<NR1> MINimum MAXimum}  |                                 |
| Parameter   | <NR1>   | 0 ~ 9 (as memory M0 ~ M9)       |
|             | MIN   | Recalls the M0 memory contents. |
|             | MAX   | Recalls the M9 memory contents. |

---

**\*RST**

Set →

---

|             |   |  |
|-------------|---|--|
| Description | Performs a device reset. Configures the unit to a known configuration (default settings). This known configuration is independent of the usage history. |  |
| Syntax      | *RST  |  |

---

**\*SAV**

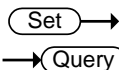
Set →

---

|                  |  |                              |
|------------------|--|------------------------------|
| Description      | Saves the settings into memory slot M0 ~ M9. These memory slots are mapped to the preset settings. |                              |
| Syntax           | *SAV {<NR1> MINimum MAXimum}   |                              |
| Return parameter | <NR1>  | 0 ~ 9 (as memory M0 ~ M9)    |
|                  | MIN  | Saves to the M0 memory slot. |
|                  | MAX  | Saves to the M9 memory slot. |

---

**\*SRE**



**Description** Sets or queries the Service Request Enable register. The Service Request Enable register determines which registers of the Status Byte register are able to generate service requests.

**Syntax** \*SRE <NR1>

**Query Syntax** \*SRE?

**Parameter** <NR1> 0~255

**Return parameter** <NR1> Returns the bit sum of the Service Request Enable register.

**\*STB**



**Description** Queries the bit sum of the Status Byte register with MSS (Master summary Status) replacing the RQS bit (bit 6).

**Query Syntax** \*STB?

**Return parameter** <NR1> Returns the bit sum of the Status Byte register with the MSS bit (bit 6).

**\*WAI**



**Description** Prevents any other commands or queries from being executed until all outstanding commands have completed.

**Syntax** \*WAI

## Trace/Data Commands



Note

The TRACE and DATA node for the following commands are functionally equivalent.

|                                      |    |
|--------------------------------------|----|
| :DATA   TRACe:SEQuence:CLEar.....    | 55 |
| :DATA   TRACe:SEQuence:RECall.....   | 55 |
| :DATA   TRACe:SEQuence:STORE.....    | 56 |
| :DATA   TRACe:SIMulation:CLEar.....  | 56 |
| :DATA   TRACe:SIMulation:RECall..... | 57 |
| :DATA   TRACe:SIMulation:STORE.....  | 57 |
| :DATA   TRACe:WAVE:CLEar.....        | 57 |
| :DATA   TRACe:WAVE[:DATA].....       | 58 |

### :DATA|TRACe:SEQuence:CLEar



**Description** Clears the sequence data for the selected save memory (Seq0 ~ Seq9).

**Syntax** :DATA|TRACe:SEQuence:CLEar  
{<NR1>|MINimum|MAXimum}

|                  |       |     |
|------------------|-------|-----|
| <b>Parameter</b> | <NR1> | 0~9 |
|                  | MIN   | 0   |
|                  | MAX   | 9   |

**Example** :DATA:SEQ:CLE 1  
Clears the sequence data from Seq1.

### :DATA|TRACe:SEQuence:RECall



**Description** Loads the sequence data. This command is the equivalent to recalling a sequence memory in the Sequence mode.

**Syntax** :DATA|TRACe:SEQuence:RECall  
{<NR1>|MINimum|MAXimum}

|           |       |                    |
|-----------|-------|--------------------|
| Parameter | <NR1> | 0~9 (Seq0 ~ Seq9). |
|           | MIN   | 0                  |
|           | MAX   | 9                  |

Example :DATA:SEQ:REC 1  
Loads the data from Seq1.

**:DATA|TRACe:SEQuence:STORe** (Set) →

Description Saves the sequence data. This command is the equivalent to saving a sequence memory in Sequence mode.

Syntax :DATA|TRACe:SEQuence:STORe  
{<NR1>|MINimum|MAXimum}

|           |       |                    |
|-----------|-------|--------------------|
| Parameter | <NR1> | 0~9 (Seq0 ~ Seq9). |
|           | MIN   | 0                  |
|           | MAX   | 9                  |

Example :DATA:SEQ:STOR 1  
Saves the data from Seq1.

**:DATA|TRACe:SIMulation:CLEAr** (Set) →

Description Clears the simulation data for the selected save memory (SIM0 ~ SIM9).

Syntax :DATA|TRACe:SIMulation:CLEAr  
{<NR1>|MINimum|MAXimum}

|           |       |                    |
|-----------|-------|--------------------|
| Parameter | <NR1> | 0~9 (SIM0 ~ SIM9). |
|           | MIN   | 0                  |
|           | MAX   | 9                  |

Example :DATA:SIM:CLE 1  
Clears the simulation data from SIM1.



**:DATA|TRACe:SIMulation:RECall****Set** →

|             |  |
|-------------|--|
| Description | Loads the simulation data. This command is the equivalent to recalling a simulation memory in the Simulation mode (SIM0~SIM9). |
| Syntax      | :DATA TRACe:SIMulation:RECall<br>{<NR1> MINimum MAXimum}   |
| Parameter   | <NR1> 0~9 (SIM0 ~ SIM9).<br>MIN 0<br>MAX 9   |
| Example     | :DATA:SIM:REC 1<br>Loads the data from SIM1.   |

**:DATA|TRACe:SIMulation:STORE****Set** →

|             |  |
|-------------|--|
| Description | Saves the simulation data. This command is the equivalent saving a simulation memory in Simulation mode (SIM0 ~ SIM9). |
| Syntax      | :DATA TRACe:SIMulation:STORE<br>{<NR1> MINimum MAXimum}  |
| Parameter   | <NR1> 0~9 (SIM0 ~ SIM9).<br>MIN 0<br>MAX 9   |
| Example     | :DATA:SIM:STOR 1<br>Saves the data from SIM1.  |

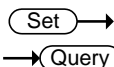
**:DATA|TRACe:WAVE:CLEar****Set** →

|             |   |
|-------------|---|
| Description | Clears the ARB 1-16 data for the selected wave group. |
| Syntax      | :DATA TRACe:WAVE:CLEar<br>{<NR1> MINimum MAXimum}     |

|           |       |                      |
|-----------|-------|----------------------|
| Parameter | <NR1> | 1~16 (ARB1 ~ ARB16). |
|           | MIN   | 1(1ARB1)             |
|           | MAX   | 16(ARB16)            |

Example :DATA:WAV:CLE 13  
Clears the wave data from ARB13.

:DATA|TRACe:WAVe[:DATA]



Description Sets the arbitrary wave.

Syntax :DATA|TRACe:WAVe[:DATA] {<NR1>|<Binary Data>}  
:DATA|TRACe:WAVe[:DATA]? <NR1>

|           |  |  |
|-----------|--|--|
| Parameter | <NR1>  | 1 - 16 (ARB 1 - 16)  |
|           | Binary Data includes the #48192<DAB>...<DAB> |  |
|           | #  | Indicates the block data is sent.  |
|           | 4  | Indicates the number of subsequent numbers.  |
|           | 8192   | Indicates the number of subsequent byte data.  |
|           | <DAB>..<br><DAB>                             | Indicates 16-bit with 4096 words waveform data. Plus, the data format of wave is the big endian in the form of two's complement. |

Example TRAC:WAV 1, #48192<DAB>...<DAB>

## Measure Commands

|  |    |
|--|----|
| :MEASure[:SCALar]:CURRent:CFACtor .....      | 59 |
| :MEASure[:SCALar]:CURRent:HIGH .....         | 59 |
| :MEASure[:SCALar]:CURRent:LOW .....          | 60 |
| :MEASure[:SCALar]:CURRent:PEAK:CLEar .....   | 60 |
| :MEASure[:SCALar]:CURRent:PEAK:HOLD .....    | 60 |
| :MEASure[:SCALar]:CURRent[:RMS] .....        | 60 |
| :MEASure[:SCALar]:CURRent:AVERAge .....      | 61 |
| :MEASure[:SCALar]:CURRent:HARMonic[:RMS] ... | 61 |
| :MEASure[:SCALar]:CURRent:HARMonic:RATio ... | 61 |
| :MEASure[:SCALar]:FREQuency .....            | 61 |
| :MEASure[:SCALar]:POWer[:AC]:APParent .....  | 62 |
| :MEASure[:SCALar]:POWer[:AC]:PFACtor .....   | 62 |
| :MEASure[:SCALar]:POWer[:AC]:REACTive .....  | 62 |
| :MEASure[:SCALar]:POWer[:AC][:REAL] .....    | 62 |
| :MEASure[:SCALar]:VOLTag[:RMS] .....         | 62 |
| :MEASure[:SCALar]:VOLTag:AVERAge .....       | 63 |
| :MEASure[:SCALar]:VOLTag:HIGH .....          | 63 |
| :MEASure[:SCALar]:VOLTag:LOW .....           | 63 |
| :MEASure[:SCALar]:VOLTag:HARMonic[:RMS] ...  | 63 |
| :MEASure[:SCALar]:VOLTag:HARMonic:RATio ...  | 64 |
| :MEASure:CONFIgure:SENSing .....             | 64 |
| :MEASure:AVERAge:COUNT .....                 | 65 |
| :MEASure:UPDate:RATE .....                   | 65 |

**:MEASure[:SCALar]:CURRent:CFACtor** → **Query**

**Description** Returns the output current crest factor (CF).


**Query syntax** :MEASure[:SCALar]:CURRent:CFACtor?

**Return parameter** <NR2> Returns the crest factor.

**:MEASure[:SCALar]:CURRent:HIGH** → **Query**

|                  |   |
|------------------|---|
| Description      | Returns the output current maximum peak value (Imax).                                   |
| Note:            | Current maximum peak value is defined as the highest peak value in the complete period. |
| Query syntax     | :MEASure[:SCALar]:CURRent:HIGH?   |
| Return parameter | <NR2> Returns the Imax value in amps.   |

**:MEASure[:SCALar]:CURRent:LOW → Query**

|  |  |
|--|--|
| Description  | Returns the output current minimum value (Imin).                             |
|  Note | Current minimum value is defined as the lowest value in the complete period. |
| Query syntax   | :MEASure[:SCALar]:CURRent:LOW?   |
| Return parameter   | <NR2> Returns the Imin value in amps.  |

**:MEASure[:SCALar]:CURRent:PEAK:CLEar Set →**

|             |                                      |
|-------------|--------------------------------------|
| Description | Clears the current peak-hold value.  |
| Syntax      | :MEASure[:SCALar]:CURRent:PEAK:CLEar |

**:MEASure[:SCALar]:CURRent:PEAK:HOLD → Query**

|              |   |
|--------------|---|
| Description  | Returns the current peak hold value in amps (IPK Hold). |
| Query syntax | :MEASure[:SCALar]:CURRent:PEAK:HOLD?                    |
| Return       | <NR2> Returns the peak hold value in amps.              |

**:MEASure[:SCALar]:CURRent[:RMS] → Query**

|              |                                    |
|--------------|------------------------------------|
| Description  | Returns the output current (Irms). |
| Query syntax | :MEASure[:SCALar]:CURRent[:RMS]?   |

Return <NR2> Returns the current value in Irms.

**:MEASure[:SCALar]:CURRent:AVEAge** → (Query)

Description Returns the current average value (Iavg).

Query syntax :MEASure[:SCALar]:CURRent:AVEAge?

Return <NR2> Returns the current average value in amps.

**:MEASure[:SCALar]:CURRent:HARMonic[:RMS]** → (Query)

Description Returns 41 values covering Total and order 1 to 40 current (Irms) in harmonic. (Only AC-INT and 50 /60 Hz Active)

Query syntax :MEASure[:SCALar]:CURRent:HARMonic[:RMS]?

Return <NR2>,<NR2 >,<NR2>,<NR2>..., etc. Returns the entire 41 values containing Total and order 1 to 40 current (Irms) in harmonic.

**:MEASure[:SCALar]:CURRent:HARMonic:RATio** → (Query)

Description Returns 41 values covering Total and order 1 to 40 current (Ratio) in harmonic. (Only AC-INT and 50 /60 Hz Active)

Query syntax :MEASure[:SCALar]:CURRent:HARMonic:RATio?

Return <NR2>,<NR2 >,<NR2>,<NR2>..., etc. Returns the entire 41 values containing Total and order 1 to 40 current (Ratio) in harmonic.

**:MEASure[:SCALar]:FREQuency** → (Query)

Description Returns the SYNC signal source frequency in Hz. (Only AC+DC-sync or AC-sync Active)

Query syntax :MEASure[:SCALar]:FREQuency?

Return <NR2> Returns the SYNC frequency in Hz.

**:MEASure[:SCALar]:POWer[:AC]:APParent** → Query

Description Returns the apparent power (S).

Query syntax :MEASure[:SCALar]:POWer[:AC]:APParent?

Return <NR2> Returns the apparent power in VA.

**:MEASure[:SCALar]:POWer[:AC]:PFACTOR** → Query

Description Returns the power factor (PF).

Query syntax :MEASure[:SCALar]:POWer[:AC]:PFACTOR?

Return <NR2> Returns the power factor.

**:MEASure[:SCALar]:POWer[:AC]:REACTIVE** → Query

Description Returns the reactive power (Q).

Query syntax :MEASure[:SCALar]:POWer[:AC]:REACTIVE?

Return <NR2> Returns the reactive power in VAR.

**:MEASure[:SCALar]:POWer[:AC][:REAL]** → Query

Description Returns the active power in Watts (P).

Query syntax :MEASure[:SCALar]:POWer[:AC][:REAL]?

Return <NR2> Returns the power in Watts.

**:MEASure[:SCALar]:VOLTage[:RMS]** → Query

Description Returns the voltage (Vrms).


Query syntax :MEASure[:SCALar]:VOLTage[:RMS]?

Return <NR2> Returns the voltage value in Vrms.

**:MEASure[:SCALar]:VOLTage:AVERage** → **Query**

|              |   |
|--------------|---|
| Description  | Returns the voltage average value (Vavg).         |
| Query syntax | :MEASure[:SCALar]:VOLTage:AVERage?                |
| Return       | <NR2> Returns the voltage average value in volts. |

**:MEASure[:SCALar]:VOLTage:HIGh** → **Query**

|  |   |
|--|---|
| Description  | Returns the output voltage maximum peak value (Vmax).                                   |
|  Note | Voltage maximum peak value is defined as the highest peak value in the complete period. |
| Query syntax   | :MEASure[:SCALar]:VOLTage:HIGh?   |
| Return parameter   | <NR2> Returns the Vmax value in volts.  |

**:MEASure[:SCALar]:VOLTage:LOW** → **Query**

|                  |  |
|------------------|--|
| Description      | Returns the output current minimum value (Vmin).                             |
| Note:            | Voltage minimum value is defined as the lowest value in the complete period. |
| Query syntax     | :MEASure[:SCALar]:VOLTage:LOW?   |
| Return parameter | <NR2> Returns the Vmin value in volts.                                       |

**:MEASure[:SCALar]:VOLTage:HARMonic[:RMS]** → **Query**

|              |  |
|--------------|--|
| Description  | Returns 41 values covering Total and order 1 to 40 voltage (Vrms) in harmonic. (Only AC-INT and 50 / 60 Hz Active) |
| Query syntax | :MEASure[:SCALar]:VOLTage:HARMonic[:RMS]?  |

|        |  |   |
|--------|--|---|
| Return | <code>&lt;NR2&gt;,&lt;NR2 &gt;,&lt;NR2&gt;,&lt;NR2&gt;..., etc.</code> | Returns the entire 41 values containing Total and order 1 to 40 voltage (Vrms) in harmonic. |
|--------|--|---|

**:MEASure[:SCALar]: VOLTage:HARMonic:RATio** → **Query**

|             |  |
|-------------|--|
| Description | Returns 41 values covering Total and order 1 to 40 voltage (Ratio) in harmonic. (Only AC-INT and 50 /60 Hz Active) |
|-------------|--|

|              |  |
|--------------|--|
| Query syntax | :MEASure[:SCALar]: VOLTage:HARMonic:RATio? |
|--------------|--|

|        |  |  |
|--------|--|--|
| Return | <code>&lt;NR2&gt;,&lt;NR2 &gt;,&lt;NR2&gt;,&lt;NR2&gt;..., etc.</code> | Returns the entire 41 values containing Total and order 1 to 40 voltage (Ratio) in harmonic. |
|--------|--|--|

**Set** →

**:MEASure:CONFigure:SENSing** → **Query**

|             |  |
|-------------|--|
| Description | Sets or queries the remote sense configuration. (Only AC-INT, DC-INT, AC-SYNC Mode and 100V, 200V Range and SIN Wave Shape and Time Slew Rate Mode Active) |
|-------------|--|

|        |  |
|--------|--|
| Syntax | :MEASure:CONFigure:SENSing {<bool> OFF ON} |
|--------|--|

|              |                             |
|--------------|-----------------------------|
| Query Syntax | :MEASure:CONFigure:SENSing? |
|--------------|-----------------------------|

|           |         |                             |
|-----------|---------|-----------------------------|
| Parameter | OFF   0 | Turns the remote sense off. |
|-----------|---------|-----------------------------|

|  |        |                            |
|--|--------|----------------------------|
|  | ON   1 | Turns the remote sense on. |
|--|--------|----------------------------|

|                  |        |                                     |
|------------------|--------|-------------------------------------|
| Return parameter | <bool> | Returns the status of remote sense. |
|------------------|--------|-------------------------------------|

|         |   |
|---------|---|
| Example | :MEAS:CONF:SENS 0<br>Sets the remote sense off. |
|---------|---|



**:MEASure:AVERage:COUNT** 


|                        |  |  |
|------------------------|--|--|
| Description            | Sets or queries the averaging count for Measure Function.                        |  |
| Syntax                 | :MEASure:AVERage:COUNT <NR1>   MINimum   MAXimum                                 |  |
| Query Syntax           | :MEASure:AVERage:COUNT? [ MINimum   MAXimum ]                                    |  |
| Parameter              | <NR1>  | 1 ~ 128<br>MINimum 1<br>MAXimum 128              |
| Return parameter       | <NR1>  | Returns the averaging count for Measure Function |
| Example                | :MEASure:AVERage:COUNT?<br>1<br>Returns the averaging count for Measure Function |  |
| Averaged Parameter     | Vrms & Vmax & Vmin & Irms & Imax & Imin & P & S & Q & PF & CF                    |  |
| Not Averaged Parameter | Vavg & Iavg & IpkH & Freq & THDv & THDi  |  |

**:MEASure:UPDate:RATE** 


|                  |  |  |
|------------------|--|--|
| Description      | Sets or queries the data update interval for Measure Function. |  |
| Syntax           | :MEASure:UPDate:RATE <Time(NR2)>   FAST                        |  |
| Query Syntax     | :MEASure:UPDate:RATE?  |  |
| Parameter        | <Time(NR2)>  | 0.1   0.25   0.5   1   2   5   10   20                 |
| Return parameter | FAST   | Select update rate at fast(update as soon as possible) |

|                               |  |
|-------------------------------|--|
| Example                       | :MEASure:UPDate:RATE?<br>Fast<br>Returns the data update interval for Measure Function |
| Use Update Rate Parameter     | Vrms & Vmax & Vmin & Irms & Imax & Imin & P & S & Q & PF & CF                          |
| Not Use Update Rate Parameter | Vavg & Iavg & IpkH & Freq & THDv & THDi(update as soon as possible)                    |

## Memory Commands

|                  |    |
|------------------|----|
| :MEMory:RCL..... | 67 |
| :MEMory:SAV..... | 67 |

### :MEMory:RCL



**Description**      Recalls the settings from memory slot M0~M9. These memory slots are mapped to the preset settings. Equivalent to the \*RCL command.

**Syntax**            :MEMory:RCL {<NR1>|MINimum|MAXimum}

|                  |         |     |
|------------------|---------|-----|
| <b>Parameter</b> | <NR1>   | 0~9 |
|                  | MINimum | 0   |
|                  | MAXimum | 9   |

**Example**            :MEMory:RCL  
Recall the settings to M1.

### :MEMory:SAV



**Description**      Saves the settings into memory slot M0 ~ M9. These memory slots are mapped to the preset settings. Equivalent to the \*SAV command.

**Syntax**            :MEMory:SAV {<NR1>|MINimum|MAXimum}

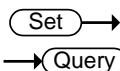
|                  |         |     |
|------------------|---------|-----|
| <b>Parameter</b> | <NR1>   | 0~9 |
|                  | MINimum | 0   |
|                  | MAXimum | 9   |

**Example**            :MEMory:SAV 1  
Save the settings to M1.

## Output Commands

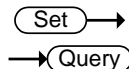
|                          |    |
|--------------------------|----|
| :OUTPut[:STATe]          | 68 |
| :OUTPut:PON              | 68 |
| :OUTPut:PROtection:CLEar | 69 |
| :OUTPut:RELAy            | 69 |

### :OUTPut[:STATe]



|                  |   |  |
|------------------|---|--|
| Description      | Sets or queries the output state of power source. |  |
| Syntax           | :OUTPut[:STATe] {<bool> OFF ON}                   |  |
| Query Syntax     | :OUTPut[:STATe]?                                  |  |
| Parameter        | OFF   0   | Turns the output off.                    |
|                  | ON   1  | Turns the output on.                     |
| Return parameter | <bool>  | Returns output status of the instrument. |
| Example          | :OUTP 0<br>Sets power output off.                 |  |

### :OUTPut:PON



|                  |                                    |  |
|------------------|------------------------------------|--|
| Description      | Sets the output state at power-on. |  |
| Syntax           | :OUTPut:PON {<NR1> OFF ON SEQ SIM} |  |
| Return Syntax    | :OUTPut:PON?                       |  |
| Parameter        | <NR1>                              | 0 ~ 3  |
|                  | OFF   0                            | Disabled   |
|                  | ON   1                             | Enabled  |
|                  | SEQ   2                            | Sequence function  |
|                  | SIM   3                            | Simulate function  |
| Return parameter | <NR1>                              | Returns the selected output state at power-on from 0 to 3. |

Example                   :OUTPut:PON 2  
                               Sets sequence function on at power-on.

**:OUTPut:PROTEction:CLEar** Set →

---

Description            The Command will clear alarms like Over Current, Over Peak Current, Output Over-Power, Output Short, Output Overvoltage, Sensing Voltage Error.

Syntax                   :OUTPut:PROTEction:CLEar

**:OUTPut:RELAy** Set →  
→ Query

---

Description            Sets or queries the output relay of power source.

Syntax                   :OUTPut:RELAy {<bool>|OFF|ON}

Query Syntax           :OUTPut:RELAy?

|           |         |                                 |
|-----------|---------|---------------------------------|
| Parameter | OFF   0 | Turns the output relay Disable. |
|           | ON   1  | Turns the output relay Enable.  |

Return parameter <bool>   Returns output relay of the instrument.

Example                   :OUTP:REL 1  
                               Sets output relay Enable.

## Status Commands

|  |    |
|--|----|
| :STATus:OPERation:CONDition .....      | 70 |
| :STATus:OPERation:ENABle.....          | 71 |
| :STATus:OPERation[:EVENT] .....        | 71 |
| :STATus:OPERation:NTRansition .....    | 71 |
| :STATus:OPERation:PTRansition .....    | 71 |
| :STATus:QUESTionable[:EVENT].....      | 72 |
| :STATus:QUESTionable:CONDition .....   | 72 |
| :STATus:QUESTionable:ENABle .....      | 72 |
| :STATus:QUESTionable:NTRansition ..... | 73 |
| :STATus:QUESTionable:PTRansition.....  | 73 |
| :STATus:PRESet.....                    | 73 |
| :STATus:WARNIng:CONDition .....        | 74 |
| :STATus:WARNIng:ENABle .....           | 74 |
| :STATus:WARNIng[:EVENT].....           | 75 |
| :STATus:WARNIng:NTRansition .....      | 75 |
| :STATus:WARNIng:PTRansition .....      | 75 |
| :STATus:LOCK:CONDition.....            | 76 |
| :STATus:LOCK:ENABle .....              | 76 |
| :STATus:LOCK[:EVENT] .....             | 76 |
| :STATus:LOCK:NTRansition.....          | 76 |
| :STATus:LOCK:PTRansition.....          | 77 |

### :STATus:OPERation:CONDition → Query

|             |  |
|-------------|--|
| Description | Queries the Operation Status register. This query will not clear the register. |
| Syntax      | :STATus:OPERation:CONDition?   |
| Return      | <NR1> Returns the bit sum of the Operation Condition register. (0~32767)       |

 →  
 → 

**:STATus:OPERation:ENABLE**

Description      Sets or queries the bit sum of the Operation Status Enable register.

Syntax             :STATus:OPERation:ENABLE <NR1>

Query Syntax      :STATus:OPERation:ENABLE?

Parameter         <NR1>    0~32767

Return parameter <NR1>    0~32767

→ 

**:STATus:OPERation[:EVENT]**

Description      Queries the Operation Status Event register and clears the contents of the register.

Syntax             :STATus:OPERation[:EVENT]?

Return             <NR1>    Returns the bit sum of the Operation Status Event register.

 →  
 → 

**:STATus:OPERation:NTRansition**

Description      Sets or queries the bit sum of the negative transition filter of the Operation Status register.

Syntax             :STATus:OPERation:NTRansition <NR1>

Query Syntax      :STATus:OPERation:NTRansition?

Parameter         <NR1>    0~32767

Return parameter <NR1>    0~32767

 →  
 → 

**:STATus:OPERation:PTRansition**

Description      Sets or queries the bit sum of the positive transition filter of the Operation Status register.

|                  |   |
|------------------|---|
| Syntax           | :STATus:OPERation:PTRansition <NR1><br>:STATus:OPERation:PTRansition? |
| Parameter        | <NR1> 0~32767   |
| Return parameter | <NR1> 0~32767   |

**:STATus:QUEStionable[:EVENT]** → Query

|                  |   |
|------------------|---|
| Description      | Queries the bit sum of the Questionable Status Event register. This query will also clear the contents of the register. |
| Query Syntax     | :STATus:QUEStionable[:EVENT]?   |
| Return parameter | <NR1> 0~32767   |

**:STATus:QUEStionable:CONDition** → Query

|                  |   |
|------------------|---|
| Description      | Queries the status (bit sum) of the Questionable Status register. This query will not clear the register. |
| Query Syntax     | :STATus:QUEStionable:CONDition?   |
| Return parameter | <NR1> 0~32767   |

**:STATus:QUEStionable:ENABle** Set →  
→ Query

|                  |   |
|------------------|---|
| Description      | Sets or queries the bit sum of the Questionable Status Enable register. |
| Syntax           | :STATus:QUEStionable:ENABle <NR1>                                       |
| Query Syntax     | :STATus:QUEStionable:ENABle?  |
| Parameter        | <NR1> 0~32767   |
| Return parameter | <NR1> 0~32767   |



**:STATus:QUEStionable:NTRansition** (Set) →  
→ (Query)

|                  |  |
|------------------|--|
| Description      | Sets or queries the bit sum of the negative transition filter of the Questionable Status register. |
| Syntax           | :STATus:QUEStionable:NTRansition <NR1>   |
| Query Syntax     | :STATus:QUEStionable:NTRansition?  |
| Parameter        | <NR1> 0~32767  |
| Return parameter | <NR1> 0~32767  |

**:STATus:QUEStionable:PTRansition** (Set) →  
→ (Query)

|                  |  |
|------------------|--|
| Description      | Sets or queries the bit sum of the positive transition filter of the Questionable Status register. |
| Syntax           | :STATus:QUEStionable:PTRansition <NR1>   |
| Return Syntax    | :STATus:QUEStionable:PTRansition?  |
| Parameter        | <NR1> 0~32767  |
| Return parameter | <NR1> 0~32767  |

**:STATus:PRESet** (Set) →

Description This command resets the ENABLE register, the PTRansition filter and NTRansition filter on the Operation Status, Questionable Status, Warning Status and System Lock Status Registers. The registers/filters will be reset to a default value.

| Default Register/Filter Values          | Setting |
|---|---------|
| QUEStionable Status Enable              | 0x0000  |
| QUEStionable Status Positive Transition | 0x7FFF  |
| QUEStionable Status Negative Transition | 0x0000  |
| Operation Status Enable                 | 0x0000  |
| Operation Status Positive Transition    | 0x7FFF  |

|  |        |
|--|--------|
| Operation Status Negative Transition   | 0x0000 |
| WARNING Status Enable                  | 0x0000 |
| WARNING Status Positive Transition     | 0x7FFF |
| WARNING Status Negative Transition     | 0x0000 |
| System Lock Status Enable              | 0x0000 |
| System Lock Status Positive Transition | 0x7FFF |
| System Lock Status Negative Transition | 0x0000 |

Summary: The Questionable Status Enable registers, the Operation Status Enable registers, Warning Status registers and System Lock Status registers are both reset to 0.

The Questionable Status, Operation Status, Warning Status and System Lock Status Positive Transition filters are all set high (0x7FFF) and the Negative Transition filters are all set low (0x0000). I.e., only positive transitions will be recognized for the Questionable Status, Operation Status, Warning Status and System Lock Status registers.

Syntax :STATus:PRESet

:STATus:WARNING:CONDition → **Query**

Description Queries the Warning Status register. This query will not clear the register.

Syntax :STATus:WARNING:CONDition?

Return <NR1> Returns the bit sum of the Warning Condition register. (0~32767)

→ **Set** →

:STATus:WARNING:ENABLE → **Query**

Description Sets or queries the bit sum of the Warning Status Enable register.

Syntax :STATus:WARNing:ENABle <NR1>

Query Syntax :STATus:WARNing:ENABle?

Parameter <NR1> 0~32767

Return parameter <NR1> 0~32767

**:STATus:WARNing[:EVENT]** → Query

Description Queries the Warning Status Event register and clears the contents of the register.

Syntax :STATus:WARNing[:EVENT]?

Return <NR1> Returns the bit sum of the Warning Status Event register.

Set →

**:STATus:WARNing:NTRansition** → Query

Description Sets or queries the bit sum of the negative transition filter of the Warning Status register.

Syntax :STATus:WARNing:NTRansition <NR1>

Query Syntax :STATus:WARNing:NTRansition?

Parameter <NR1> 0~32767

Return parameter <NR1> 0~32767

Set →

**:STATus:WARNing:PTRansition** → Query

Description Sets or queries the bit sum of the positive transition filter of the Warning Status register.

Syntax :STATus:WARNing:PTRansition <NR1>

:STATus:WARNing:PTRansition?

Parameter <NR1> 0~32767

Return parameter <NR1> 0~32767

**:STATus:LOCK:CONDition** → Query

|             |  |
|-------------|--|
| Description | Queries the System Lock Status register. This query will not clear the register. |
| Syntax      | :STATus:LOCK:CONDition?  |
| Return      | <NR1> Returns the bit sum of the System Lock Status register. (0~32767)          |

Set →

**:STATus:LOCK:ENABLE** → Query

|                  |  |
|------------------|--|
| Description      | Sets or queries the bit sum of the System Lock Status Enable register. |
| Syntax           | :STATus:LOCK:ENABLE <NR1>  |
| Query Syntax     | :STATus:LOCK:ENABLE?   |
| Parameter        | <NR1> 0~32767  |
| Return parameter | <NR1> 0~32767  |

**:STATus:LOCK[:EVENT]** → Query

|             |  |
|-------------|--|
| Description | Queries the System Lock Status Event register and clears the contents of the register. |
| Syntax      | :STATus:LOCK [:EVENT]?   |
| Return      | <NR1> Returns the bit sum of the System Lock Status Event register.                    |

Set →

**:STATus:LOCK:NTRansition** → Query

|              |   |
|--------------|---|
| Description  | Sets or queries the bit sum of the negative transition filter of the System Lock Status register. |
| Syntax       | :STATus:LOCK:NTRansition <NR1>  |
| Query Syntax | :STATus:LOCK:NTRansition?   |
| Parameter    | <NR1> 0~32767   |

Return parameter <NR1> 0~32767

:STATus:LOCK:PTRansition

Set →

→ Query

Description Sets or queries the bit sum of the positive transition filter of the System Lock Status register.

Syntax :STATus:LOCK:PTRansition <NR1>  
:STATus:LOCK:PTRansition?

Parameter <NR1> 0~32767

Return parameter <NR1> 0~32767

## System Function Commands

|   |    |
|---|----|
| :SYSTem:ACIN:DETEction.....                                   | 80 |
| :SYSTem:ARBitrary:EDIT:BUILtin.....                           | 80 |
| :SYSTem:ARBitrary:EDIT:SURGe.....                             | 81 |
| :SYSTem:ARBitrary:EDIT:STAir.....                             | 81 |
| :SYSTem:ARBitrary:EDIT:CFACTor2.....                          | 82 |
| :SYSTem:ARBitrary:EDIT:CFACTor1.....                          | 82 |
| :SYSTem:ARBitrary:EDIT:CLIP.....                              | 83 |
| :SYSTem:ARBitrary:EDIT:STORE.....                             | 83 |
| :SYSTem:ARBitrary:EDIT:TRIangle.....                          | 84 |
| :SYSTem:ARBitrary:EDIT:RIPple.....                            | 85 |
| :SYSTem:ARBitrary:EDIT:STORE:APPLY.....                       | 86 |
| :SYSTem:BEEPer:STATe.....                                     | 87 |
| :SYSTem:COMMunicate:GPIB[:SELF]:ADDRESS<br>.....              | 87 |
| :SYSTem:COMMunicate:LAN:DHCP.....                             | 87 |
| :SYSTem:COMMunicate:LAN:DNS.....                              | 88 |
| :SYSTem:COMMunicate:LAN:GATeway.....                          | 88 |
| :SYSTem:COMMunicate:LAN:IPAdDress.....                        | 89 |
| :SYSTem:COMMunicate:LAN:MAC.....                              | 89 |
| :SYSTem:COMMunicate:LAN:SMASK.....                            | 89 |
| :SYSTem:COMMunicate:RLSState.....                             | 90 |
| :SYSTem:COMMunicate:SERial[:RECeive]<br>:TRANsmit:BAUD.....   | 90 |
| :SYSTem:COMMunicate:SERial[:RECeive]<br>:TRANsmit:BITS.....   | 91 |
| :SYSTem:COMMunicate:SERial[:RECeive]<br>:TRANsmit:PARity..... | 92 |
| :SYSTem:COMMunicate:SERial[:RECeive]<br>:TRANsmit:SBITs.....  | 92 |
| :SYSTem:COMMunicate:TCPip:CONTRol.....                        | 93 |
| :SYSTem:COMMunicate:USB:FRONt:STATe .....                     | 93 |
| :SYSTem:COMMunicate:USB:REAR:STATe .....                      | 94 |
| :SYSTem:CONFigure[:MODE].....                                 | 94 |
| :SYSTem:CONFigure:EXTio[:STATe].....                          | 95 |
| :SYSTem:CONFigure:TRIGger:OUTPut:WIDTH<br>.....               | 95 |

|   |    |
|---|----|
| :SYSTem:CONFIgure:TRIGger:OUTPut:SOURce ..... | 96 |
| :SYSTem:ERRor .....                           | 97 |
| :SYSTem:ERRor:ENABle .....                    | 97 |
| :SYSTem:HOLD:STATe .....                      | 97 |
| :SYSTem:IPKHold:TIME .....                    | 98 |
| :SYSTem:KLOCK .....                           | 98 |
| :SYSTem:REBoot .....                          | 98 |
| :SYSTem:SLEW:MODE .....                       | 99 |
| :SYSTem:VUNit .....                           | 99 |

:SYSTem:ACIN:DETection 


|                  |   |   |
|------------------|---|---|
| Description      | Sets or queries the AC input detection on/off.  |   |
| Syntax           | :SYSTem:ACIN:DETection {<bool> OFF ON}          |   |
| Query Syntax     | :SYSTem:ACIN:DETection?                         |   |
| Parameter        | OFF   0   | Turns the AC input detection off.       |
|                  | ON   1  | Turns the AC input detection on.        |
| Return parameter | <bool>  | Returns the AC input detection setting. |
| Example          | :SYST:ACIN:DET 1<br>Sets AC input detection on. |   |

:SYSTem:ARBitrary:EDIT:BUILtin 


|                  |  |                                      |
|------------------|--|--------------------------------------|
| Description      | Sets or queries the built in function of arbitrary edit                        |                                      |
| Syntax           | :SYSTem:ARBitrary:EDIT:BUILtin TRlangle   STAir                                |                                      |
| Query Syntax     | CLIP   CFACtor1   CFACtor2   SURGe   DST<01 22>                                |                                      |
|                  | RIPPLE   |                                      |
|                  | :SYSTem:ARBitrary:EDIT:BUILtin?  |                                      |
| Parameter/Return | TRlangle   | Built In Triangle Wave Function      |
|                  | STAir  | Built In Stair Wave Function         |
|                  | CLIP   | Built In Clip Wave Function          |
|                  | CFACtor1   | Built In CF-1 Wave Function          |
|                  | CFACtor2   | Built In CF-2 Wave Function          |
|                  | SURGe  | Built In Surge Wave Function         |
|                  | DST<01 22>   | Built In DST01 ~ DST22 Wave Function |
|                  | RIPPLE   | Built In DC Ripple Wave Function     |
| Example          | :SYST:ARB:EDIT:BUIL?<br>TRI<br>Returns the built in function of arbitrary edit |                                      |



**:SYSTem:ARBitrary:EDIT:SURGe** 


|                  |   |   |
|------------------|---|---|
| Description      | Sets or queries the type and ACV and site parameter for built in Surge wave function  |   |
| Syntax           | :SYSTem:ARBitrary:EDIT:SURGe <NR1>   SQU   SIN, <NR1>   MINimum   MAXimum, <NR1>   MINimum   MAXimum                        |   |
| Query Syntax     | :SYSTem:ARBitrary:EDIT:SURGe?   |   |
| Parameter<Type>  | SQU   0<br>SIN   1  | Square waveform type<br>Sine waveform type  |
| Parameter<ACV>   | <NR1><br>MINimum<br>MAXimum   | ACV Ratio : 0 ~100(0 ~ 100%)<br>Minimum ACV Ratio : 0 (0%)<br>Maximum ACV Ratio : 100 (100%)    |
| Parameter<Site>  | <NR1><br>MINimum<br>MAXimum   | Site Ratio : 0 ~100(0 ~ 100%)<br>Minimum Site Ratio : 0 (0%)<br>Maximum Site Ratio : 100 (100%) |
| Return parameter | <Type>,<ACV>,<Site>   | Returns the type and ACV and site parameter for built in Surge wave function                    |
| Example          | :SYST:ARB:EDIT:SURG?<br><br>SIN,+50,+25<br><br>Returns the type and ACV and site parameter for built in Surge wave function |   |

**:SYSTem:ARBitrary:EDIT:STAir** 


|                  |  |  |
|------------------|--|--|
| Description      | Sets or queries the stair parameter for built in stair wave function |  |
| Syntax           | :SYSTem:ARBitrary:EDIT:STAir <NR1>   MINimum   MAXimu                |  |
| Query Syntax     | :SYSTem:ARBitrary:EDIT:STAir? [ MINimum   MAXimum ]?                 |  |
| Parameter        | <NR1><br>MINimum<br>MAXimum  | stair : 1 ~ 100<br>Minimum stair : 1<br>Maximum stair : 100  |
| Return parameter | <NR1>  | Returns the stair parameter for built in stair wave function |

Example :SYST:ARB:EDIT:STA?  
+5  
Returns the stair parameter for built in stair wave function

Set →

:SYSTem:ARBitrary:EDIT:CFACtor2 → Query

Description Sets or queries the crest factor parameter for built in CF-2 wave function

Syntax :SYSTem:ARBitrary:EDIT:CFACtor2 <NR2> | MINimum | MAXimum

Query Syntax :SYSTem:ARBitrary:EDIT:CFACtor2? [ MINimum | MAXimum ]?

|                  |         |  |
|------------------|---------|--|
| Parameter        | <NR2>   | crest factor : 1.5 ~ 2.0   |
|                  | MINimum | Minimum crest factor : 1.5   |
|                  | MAXimum | Maximum crest factor : 2.0   |
| Return parameter | <NR2>   | Returns the crest factor parameter for built in CF-2 wave function |

Example :SYST:ARB:EDIT:CFAC2?  
+1.5000  
Returns the crest factor parameter for built in CF-2 wave function

Set →

:SYSTem:ARBitrary:EDIT:CFACtor1 → Query

Description Sets or queries the crest factor parameter for built in CF-1 wave function

Syntax :SYSTem:ARBitrary:EDIT:CFACtor1 <NR2> | MINimum | MAXimum

Query Syntax :SYSTem:ARBitrary:EDIT:CFACtor1? [ MINimum | MAXimum ]?

|                  |         |  |
|------------------|---------|--|
| Parameter        | <NR2>   | crest factor : 1.1 ~ 10.0  |
|                  | MINimum | Minimum crest factor : 1.1   |
|                  | MAXimum | Maximum crest factor : 10.0  |
| Return parameter | <NR2>   | Returns the crest factor parameter for built in CF-1 wave function |

Example :SYST:ARB:EDIT:CFAC1?  
 +2.0000  
 Returns the crest factor parameter for built in CF-1 wave function

**:SYSTem:ARBitary:EDIT:CLIP**

Set →

→ Query

Description Sets or queries the ratio parameter for built in clip wave function

Syntax :SYSTem:ARBitary:EDIT:CLIP <NR2> | MINimum | MAXimum

Query Syntax :SYSTem:ARBitary:EDIT:CLIP? [ MINimum | MAXimum ]?

|                  |         |   |
|------------------|---------|---|
| Parameter        | <NR2>   | clip ratio : 0.00 ~ 1.00                                    |
|                  | MINimum | Minimum clip ratio : 0.00                                   |
|                  | MAXimum | Maximum clip ratio : 1.00                                   |
| Return parameter | <NR2>   | Returns the ratio parameter for built in clip wave function |

Example :SYST:ARB:EDIT:CLIP?  
 +0.5000  
 Returns the ratio parameter for built in clip wave function

**:SYSTem:ARBitary:EDIT:STORE**

Set →

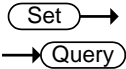
Description Saves the waveform data of built in into ARB1 ~ ARB16

Syntax :SYSTem:ARBitary:EDIT:STORE <NR1> | ARB1 | ARB2 | ARB3 | ARB4 | ARB5 | ARB6 | ARB7 | ARB8 | ARB9 | ARB10 | ARB11 | ARB12 | ARB13 | ARB14 | ARB15 | ARB16

|           |          |   |
|-----------|----------|---|
| Parameter | ARB1   1 | Saves the waveform data of built in into ARB1 |
|           | ARB2   2 | Saves the waveform data of built in into ARB2 |
|           | ARB3   3 | Saves the waveform data of built in into ARB3 |
|           | ARB4   4 | Saves the waveform data of built in into ARB4 |
|           | ARB5   5 | Saves the waveform data of built in into ARB5 |
|           | ARB6   6 | Saves the waveform data of built in into ARB6 |

|       |    |  |
|-------|----|--|
| ARB7  | 7  | Saves the waveform data of built in into ARB7  |
| ARB8  | 8  | Saves the waveform data of built in into ARB8  |
| ARB9  | 9  | Saves the waveform data of built in into ARB9  |
| ARB10 | 10 | Saves the waveform data of built in into ARB10 |
| ARB11 | 11 | Saves the waveform data of built in into ARB11 |
| ARB12 | 12 | Saves the waveform data of built in into ARB12 |
| ARB13 | 13 | Saves the waveform data of built in into ARB13 |
| ARB14 | 14 | Saves the waveform data of built in into ARB14 |
| ARB15 | 15 | Saves the waveform data of built in into ARB15 |
| ARB16 | 16 | Saves the waveform data of built in into ARB16 |

**Example**                   :SYST:ARB:EDIT:STOR ARB1  
                                   Saves the waveform data of built in into ARB1

**:SYSTem:ARBitrary:EDIT:TRlangle**                   

|                         |  |  |
|-------------------------|--|--|
| <b>Description</b>      | Sets or queries the symmetry parameter for built in triangle wave function |  |
| <b>Syntax</b>           | :SYSTem:ARBitrary:EDIT:TRlangle <NR1>   MINimum                            |  |
| <b>Query Syntax</b>     | MAXimum<br>:SYSTem:ARBitrary:EDIT:TRlangle? [ MINimum   MAXimum ]?         |  |
| <b>Parameter</b>        | <NR1>  | Symmetry : 0 ~ 100(0 ~ 100%)                                       |
|                         | MINimum  | Minimum Symmetry : 0 (0%)  |
|                         | MAXimum  | Maximum Symmetry : 100 (100%)                                      |
| <b>Return parameter</b> | <NR1>  | Returns the symmetry parameter for built in triangle wave function |

**Example**                   :SYST:ARB:EDIT:TRI?  
                                   +50  
                                   Returns the symmetry parameter for built in triangle wave function

Set →

→ Query

:SYSTem:ARBitrary:EDIT:RIPPlE

|                   |   |  |
|-------------------|---|--|
| Description       | Sets or queries the Times and VDC and Level parameter for built in DC Ripple wave function                              |  |
| Syntax            | :SYSTem:ARBitrary:EDIT:RIPPlE <NR1>   MINimum   MAXimum,<NR1>   MINimum   MAXimum, <NR1>   MINimum   MAXimum            |  |
| Query Syntax      | :SYSTem:ARBitrary:EDIT:RIPPlE?  |  |
| Parameter <Times> | <NR1>   | Times : 1   2   3   6  |
|                   | MINimum   | MINimum Times : 1  |
|                   | MAXimum   | MAXimum Times : 6  |
| Parameter <VDC>   | <NR1>   | VDC Value : 1 ~ 100  |
|                   | MINimum   | MINimum VDC Value : 1  |
|                   | MAXimum   | MAXimum VDC Value : 100  |
| Parameter <Level> | <NR1>   | Level Ratio : 1 ~ 30(1 ~ 30%)  |
|                   | MINimum   | MINimum Level Ratio : 1(1%)  |
|                   | MAXimum   | MAXimum Level Ratio : 30(30%)  |
| Return parameter  | <Times>, <VDC>, <Level>   | Returns the Times and VDC and Level parameter for built in DC Ripple wave function |
| Example           | :SYST:ARB:EDIT:RIPP?<br>1,+48,+15<br>Returns the Times and VDC and Level parameter for built in DC Ripple wave function |  |

:SYSTem:ARBitrary:EDIT:STORe:APPLy Set →

|             |  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
|-------------|--|----------|---|----------|---|----------|---|----------|---|----------|---|----------|---|----------|---|----------|---|----------|---|------------|--|------------|--|------------|--|------------|--|------------|--|------------|--|------------|--|
| Description | Saves the waveform(into ARB1 ~ ARB16)/Output Mode/ACV/DCV/VPK+ Limit/VPK- Limit/V Unit(TRI, ARB) data(for Built in is RIPple)  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| Syntax      | :SYSTem:ARBitrary:EDIT:STORe:APPLy <NR1>   ARB1   ARB2   ARB3   ARB4   ARB5   ARB6   ARB7   ARB8   ARB9   ARB10   ARB11   ARB12   ARB13   ARB14   ARB15   ARB16  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| Parameter   | <table border="0"> <tr><td>ARB1   1</td><td>Saves the waveform data of built in into ARB1</td></tr> <tr><td>ARB2   2</td><td>Saves the waveform data of built in into ARB2</td></tr> <tr><td>ARB3   3</td><td>Saves the waveform data of built in into ARB3</td></tr> <tr><td>ARB4   4</td><td>Saves the waveform data of built in into ARB4</td></tr> <tr><td>ARB5   5</td><td>Saves the waveform data of built in into ARB5</td></tr> <tr><td>ARB6   6</td><td>Saves the waveform data of built in into ARB6</td></tr> <tr><td>ARB7   7</td><td>Saves the waveform data of built in into ARB7</td></tr> <tr><td>ARB8   8</td><td>Saves the waveform data of built in into ARB8</td></tr> <tr><td>ARB9   9</td><td>Saves the waveform data of built in into ARB9</td></tr> <tr><td>ARB10   10</td><td>Saves the waveform data of built in into ARB10</td></tr> <tr><td>ARB11   11</td><td>Saves the waveform data of built in into ARB11</td></tr> <tr><td>ARB12   12</td><td>Saves the waveform data of built in into ARB12</td></tr> <tr><td>ARB13   13</td><td>Saves the waveform data of built in into ARB13</td></tr> <tr><td>ARB14   14</td><td>Saves the waveform data of built in into ARB14</td></tr> <tr><td>ARB15   15</td><td>Saves the waveform data of built in into ARB15</td></tr> <tr><td>ARB16   16</td><td>Saves the waveform data of built in into ARB16</td></tr> </table> | ARB1   1 | Saves the waveform data of built in into ARB1 | ARB2   2 | Saves the waveform data of built in into ARB2 | ARB3   3 | Saves the waveform data of built in into ARB3 | ARB4   4 | Saves the waveform data of built in into ARB4 | ARB5   5 | Saves the waveform data of built in into ARB5 | ARB6   6 | Saves the waveform data of built in into ARB6 | ARB7   7 | Saves the waveform data of built in into ARB7 | ARB8   8 | Saves the waveform data of built in into ARB8 | ARB9   9 | Saves the waveform data of built in into ARB9 | ARB10   10 | Saves the waveform data of built in into ARB10 | ARB11   11 | Saves the waveform data of built in into ARB11 | ARB12   12 | Saves the waveform data of built in into ARB12 | ARB13   13 | Saves the waveform data of built in into ARB13 | ARB14   14 | Saves the waveform data of built in into ARB14 | ARB15   15 | Saves the waveform data of built in into ARB15 | ARB16   16 | Saves the waveform data of built in into ARB16 |
| ARB1   1    | Saves the waveform data of built in into ARB1  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB2   2    | Saves the waveform data of built in into ARB2  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB3   3    | Saves the waveform data of built in into ARB3  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB4   4    | Saves the waveform data of built in into ARB4  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB5   5    | Saves the waveform data of built in into ARB5  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB6   6    | Saves the waveform data of built in into ARB6  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB7   7    | Saves the waveform data of built in into ARB7  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB8   8    | Saves the waveform data of built in into ARB8  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB9   9    | Saves the waveform data of built in into ARB9  |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB10   10  | Saves the waveform data of built in into ARB10   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB11   11  | Saves the waveform data of built in into ARB11   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB12   12  | Saves the waveform data of built in into ARB12   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB13   13  | Saves the waveform data of built in into ARB13   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB14   14  | Saves the waveform data of built in into ARB14   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB15   15  | Saves the waveform data of built in into ARB15   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |
| ARB16   16  | Saves the waveform data of built in into ARB16   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |          |   |            |  |            |  |            |  |            |  |            |  |            |  |            |  |

**Example**                   :SYST:ARB:EDIT:STOR:APPL ARB2

Saves the waveform(into ARB2) and Output Mode(AC+DC-INT) / ACV / DCV / VPK+ Limit(max) / VPK- Limit(min) / V Unit(TRI, ARB)(p-p) data(for Built in is RIPple)

**:SYSTem:BEEPer:STATe** 



|                  |  |                            |
|------------------|--|----------------------------|
| Description      | Sets or queries the buzzer state on/off. |                            |
| Syntax           | :SYSTem:BEEPer:STATe {<bool> OFF ON}     |                            |
| Query Syntax     | :SYSTem:BEEPer:STATe?                    |                            |
| Parameter        | OFF   0                                  | Turns the buzzer off.      |
|                  | ON   1                                   | Turns the buzzer on.       |
| Return parameter | <bool>                                   | Returns the buzzer status. |

**:SYSTem:COMMunicate:GPIB[:SELF]:ADDRes** 



|                  |   |      |
|------------------|---|------|
| Description      | Sets or queries the GPIB address.                               |      |
| Note:            | The setting will only be valid after the power has been cycled. |      |
| Syntax           | :SYSTem:COMMunicate:GPIB[:SELF]:ADDRes <NR1>                    |      |
| Query Syntax     | :SYSTem:COMMunicate:GPIB[:SELF]:ADDRes?                         |      |
| Parameter/Return | <NR1>   | 0~30 |
| Example          | SYST:COMM:GPIB:ADDR 15<br>Sets the GPIB address to 15.          |      |

**:SYSTem:COMMunicate:LAN:DHCP** 



|              |   |          |
|--------------|---|----------|
| Description  | Turns DHCP on/off. Queries the DHCP status.                     |          |
| Note:        | The setting will only be valid after the power has been cycled. |          |
| Syntax       | :SYSTem:COMMunicate:LAN:DHCP {<bool> OFF ON}                    |          |
| Query Syntax | :SYSTem:COMMunicate:LAN:DHCP?                                   |          |
| Parameter    | OFF   0   | DHCP off |

ON | 1 DHCP on

Return parameter <bool> Returns the DHCP status.

Set →

:SYSTem:COMMunicate:LAN:DNS

→ Query

Description Sets or queries the DNS address.

Note The setting will only be valid after the power has been cycled.

Syntax :SYSTem:COMMunicate:LAN:DNS <string>

Query Syntax :SYSTem:COMMunicate:LAN:DNS?

Parameter/Return <string> DNS in string format ("mask")  
Applicable ASCII characters: 20H to 7EH


Example SYST:COMM:LAN:DNS "172.16.1.252"  
Sets the DNS to 172.16.1.252.

Set →

:SYSTem:COMMunicate:LAN:GATeway

→ Query

Description Sets or queries the Gateway address.

 Note The setting will only be valid after the power has been cycled.

Syntax :SYSTem:COMMunicate:LAN:GATeway <string>

Query Syntax :SYSTem:COMMunicate:LAN:GATeway?


Parameter/Return <string> Gateway address in string format ("address")  
Applicable ASCII characters: 20H to 7EH

Example SYST:COMM:LAN:GAT "172.16.0.254"  
Sets the LAN gateway to 172.16.0.254.



**:SYSTem:COMMunicate:LAN:IPADdress** 




|  |   |
|--|---|
| Description  | Sets or queries LAN IP address.   |
|  Note | The setting will only be valid after the power has been cycled.                                 |
| Syntax   | :SYSTem:COMMunicate:LAN:IPADdress <string>  |
| Query Syntax   | :SYSTem:COMMunicate:LAN:IPADdress?  |
| Parameter/Return   | <string> LAN IP address in string format (“address”)<br>Applicable ASCII characters: 20H to 7EH |
| Example  | SYST:COMM:LAN:IPAD “172.16.5.111”<br>Sets the IP address to 172.16.5.111.                       |

**:SYSTem:COMMunicate:LAN:MAC** 

|                  |  |
|------------------|--|
| Description      | Returns the unit MAC address as a string. The MAC address cannot be changed. |
| Query Syntax     | :SYSTem:COMMunicate:LAN:MAC?   |
| Return parameter | <string> Returns the MAC address in the following format “FF-FF-FF-FF-FF-FF” |
| Example          | SYST:COMM:LAN:MAC?<br>02-80-AD-20-31-B1<br>Returns the MAC address.          |

**:SYSTem:COMMunicate:LAN:SMASK** 



|  |   |
|--|---|
| Description  | Sets or queries the LAN subnet mask.                            |
|  Note | The setting will only be valid after the power has been cycled. |
| Syntax   | :SYSTem:COMMunicate:LAN:SMASk <string>                          |
| Query Syntax   | :SYSTem:COMMunicate:LAN:SMASK?                                  |

Parameter/Return <string> Subnet mask in string format ( "mask")  
 Applicable ASCII characters: 20H to 7EH

Example SYST:COMM:LAN:SMASK "255.255.0.0"  
 Sets the LAN mask to 255.255.0.0.

:SYSTem:COMMunicate:RLState

Set →

→ Query

Description Enables or disables local/remote state of the instrument.

Syntax :SYSTem:COMMunicate:RLState  
 {LOCAL | REMote | RWLock | LREMote}

Query Syntax :SYSTem:COMMunicate:RLState?

|                            |         |   |
|----------------------------|---------|---|
| Parameter/Return parameter | LOCAL   | All keys are valid. This instrument is controlled by the front panel controls.              |
|                            | REMote  | All keys are invalid, except for the [local] key and the ability to turn the output off.    |
|                            | RWLock  | All keys are invalid. The instrument can only be controlled remotely.                       |
|                            | LREMote | All keys are valid. This instrument is controlled by the front panel controls and remotely. |

Example :SYST:COMM:RLST LOCAL  
 Sets the operating mode to local.

:SYSTem:COMMunicate:SERial[:RECEive]  
 :TRANsmit:BAUD

Set →

→ Query

Description Sets or queries the UART baud rate.



Note

The setting will only be valid after the power has been cycled.

Syntax :SYSTem:COMMunicate:SERial[:RECeive]:TRANsmit  
:BAUD <NR1>


Query Syntax :SYSTem:COMMunicate:SERial[:RECeive]:TRANsmit  
:BAUD?

Parameter/Return <NR1> 1200, 2400, 4800, 9600, 19200, 38400,  
57600, 115200

Example SYST:COMM:SER:TRAN:BAUD?  
9600  
Returns the baud rate settings.

:SYSTem:COMMunicate:SERial[:RECeive] (Set) →  
:TRANsmit:BITS → (Query)

Description Sets or queries the UART number of data bits.

 Note The setting will only be valid after the power has been cycled.

Syntax :SYSTem:COMMunicate:SERial[:RECeive]:TRANsmit  
:BITS <NR1>

Query Syntax :SYSTem:COMMunicate:SERial[:RECeive]:TRANsmit  
:BITS?


Parameter 0 7 bits  
1 8 bits

Return parameter +0 7 bits  
+1 8 bits

Example SYST:COMM:SER:TRAN:BITS?  
+1  
Indicates that 8 data bits are used for the UART connection.

:SYSTem:COMMunicate:SERial[:RECeive] (Set) →  
 :TRANsmit:PARity → (Query)

**Description** Sets or queries the parity of the UART connection.

 **Note** The setting will only be valid after the power has been cycled.

**Syntax** :SYSTem:COMMunicate:SERial[:RECeive]:TRANsmit  
 :PARity {NONE|ODD|EVEN}

**Query Syntax** :SYSTem:COMMunicate:SERial[:RECeive]:TRANsmit  
 :PARity?


|                  |      |             |
|------------------|------|-------------|
| <b>Parameter</b> | NONE | No parity   |
|                  | ODD  | Odd parity  |
|                  | EVEN | Even parity |

|                         |    |             |
|-------------------------|----|-------------|
| <b>Return parameter</b> | +0 | No parity   |
|                         | +1 | Odd parity  |
|                         | +2 | Even parity |

**Example** SYST:COMM:SER:TRAN:PARity?  
 +0  
 Indicates that no parity is used for the UART connection.

:SYSTem:COMMunicate:SERial[:RECeive] (Set) →  
 :TRANsmit:SBITs → (Query)

**Description** Sets or queries the number of stop bits used for the UART connection.

 **Note** The setting will only be valid after the power has been cycled.

Syntax :SYSTem:COMMunicate:SERial[:RECeive]:TRANsmit  
:SBITs <NR1>

Query Syntax :SYSTem:COMMunicate:SERial[:RECeive]:TRANsmit  
:SBITs?

|           |   |             |
|-----------|---|-------------|
| Parameter | 0 | 1 stop bit  |
|           | 1 | 2 stop bits |

|                  |    |             |
|------------------|----|-------------|
| Return parameter | +0 | 1 stop bit  |
|                  | +1 | 2 stop bits |

Example SYST:COMM:SER:TRAN:SBITs?  
+1  
Indicates that one stop bit is used for the UART connection.

**:SYSTem:COMMunicate:TCPIp:CONTRol** → **Query**

Description Queries the socket port number.

Query Syntax :SYSTem:COMMunicate:TCPIp:CONTRol?

Return parameter <NR1> 0000 ~ 9999

Example SYST:COMM:TCP:CONTRol?  
2268  
Returns the socket port number.

**:SYSTem:COMMunicate:USB:FRONT:STATe** → **Query**

Description Queries the front panel USB-A port state.

Query Syntax :SYSTem:COMMunicate:USB:FRONT:STATe?

|                  |    |                   |
|------------------|----|-------------------|
| Return parameter | +0 | <NR1>Absent       |
|                  | +1 | <NR1>Mass Storage |



**:SYSTem:COMMunicate:USB:REAR:STATe** → **Query**

|                  |  |                          |
|------------------|--|--------------------------|
| Description      | Queries the rear panel USB-B port state. |                          |
| Query Syntax     | :SYSTem:COMMunicate:USB:REAR:STATe?      |                          |
| Return parameter | +0                                       | <NR1>Absent              |
|                  | +1                                       | <NR1>Connected to the PC |

**Set** →


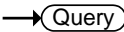
**:SYSTem:CONFigure[:MODE]** → **Query**

|                  |   |   |
|------------------|---|---|
| Description      | Sets or queries the test mode for the power supply.   |   |
| Syntax           | :SYSTem:CONFigure[:MODE]<br>{<NR1> CONTinuous SEQuence SIMulation}<br>(SEQ is available for AC+DC-INT, AC-INT, DC-INT Modes, whilst SIM is available for AC+DC-INT Mode.) |   |
| Query Syntax     | :SYSTem:CONFigure[:MODE]?   |   |
| Parameter        | CONTinuous   0  | Continuous mode (normal operating mode) |
|                  | SEQuence   1  | Sequence mode                           |
|                  | SIMulation   2  | Simulation mode                         |
| Return parameter | <NR1>   |   |
|                  | CONT  | Continuous mode (normal operating mode) |
|                  | SEQ   | Sequence mode                           |
|                  | SIM   | Simulation mode                         |

**:SYSTem:CONFigure:EXTio[:STATe]**

|                  |  |                                      |
|------------------|--|--------------------------------------|
| Description      | Sets or queries the external control state on/off. |                                      |
| Syntax           | :SYSTem:CONFigure:EXTio[:STATe] {<bool> OFF ON}    |                                      |
| Query Syntax     | :SYSTem:CONFigure:EXTio[:STATe]?                   |                                      |
| Parameter        | OFF   0  | Turns the external control off.      |
|                  | ON   1   | Turns the external control on.       |
| Return parameter | <bool>   | Returns the external control status. |

**:SYSTem:CONFigure:TRIGger:OUTPut:WIDTh**

|                  |   |                                   |
|------------------|---|-----------------------------------|
| Description      | Sets or queries the type of trigger output. The trigger output can be set as a user-defined pulse width or as a trigger output level. |                                   |
| Syntax           | :SYSTem:CONFigure:TRIGger:OUTPut:WIDTh {<NR2> MINimum MAXimum}  |                                   |
| Query Syntax     | :SYSTem:CONFigure:TRIGger:OUTPut:WIDTh? [MINimum MAXimum]   |                                   |
| Parameter        | <NR2>   | 0.0001 ~ 0.06                     |
|                  | MINimum   | 0.0001                            |
|                  | MAXimum   | 0.06                              |
| Return parameter | <NR2>   | Returns the trigger output width. |
| Example          | :SYST:CONF:TRIG:OUTP:WIDT 0.005<br>Sets the trigger output width to 5ms.  |                                   |

Set →

→ Query

:SYSTem:CONFIgure:TRIGger:OUTPut:SOURce

|                  |   |   |
|------------------|---|---|
| Description      | Configures the source for the trigger output. Equivalent to the Output Pin>Source settings. |   |
| Syntax           | :SYSTem:CONFIgure:TRIGger:OUTPut:SOURce<br>NONE   ZERO-cross   OUTPut-off                   |   |
| Query Syntax     | :SYSTem:CONFIgure:TRIGger:OUTPut:SOURce?  |   |
| Parameter        | None  | No source is assigned.  |
|                  | ZERO-cross  | When the output waveform at zero cross will generate a trigger. |
|                  | OUTPut-off  | Turning the output off will generate a trigger.                 |
| Return parameter | None  | No source is assigned.  |
|                  | ZERO-cross  | When the output waveform at zero cross will generate a trigger. |
|                  | OUTPut-off  | Turning the output on will generate a trigger.                  |
| Example          | :SYST:CONF:TRIG:OUTP:SOUR?<br>None<br>No trigger source is assigned.                        |   |



**:SYSTem:ERRor** → Query

|             |  |
|-------------|--|
| Description | Queries the error queue. The last error message is returned. A maximum of 32 errors are stored in the error queue. |
|-------------|--|

|              |                |
|--------------|----------------|
| Query Syntax | :SYSTem:ERRor? |
|--------------|----------------|

|                  |          |  |
|------------------|----------|--|
| Return parameter | <string> | Returns an error code followed by an error message as a single string. |
|------------------|----------|--|

|         |  |
|---------|--|
| Example | SYSTem:ERRor?<br>-100, "Command error" |
|---------|--|

**:SYSTem:ERRor:ENABle** Set →

|             |   |
|-------------|---|
| Description | Clears the Error Queue and enables all error messages to be placed in the System Error Queue. |
|-------------|---|

|        |                      |
|--------|----------------------|
| Syntax | :SYSTem:ERRor:ENABle |
|--------|----------------------|

**:SYSTem:HOLD:STATe** Set →  
→ Query

|             |   |
|-------------|---|
| Description | Sets or queries the freeze hold state on/off. |
|-------------|---|

|        |                                    |
|--------|------------------------------------|
| Syntax | :SYSTem:HOLD:STATe {<bool> OFF ON} |
|--------|------------------------------------|

|              |                     |
|--------------|---------------------|
| Query Syntax | :SYSTem:HOLD:STATe? |
|--------------|---------------------|

|           |         |                            |
|-----------|---------|----------------------------|
| Parameter | OFF   0 | Turns the freeze hold off. |
|-----------|---------|----------------------------|

|  |        |                           |
|--|--------|---------------------------|
|  | ON   1 | Turns the freeze hold on. |
|--|--------|---------------------------|

|                  |        |                                 |
|------------------|--------|---------------------------------|
| Return parameter | <bool> | Returns the freeze hold status. |
|------------------|--------|---------------------------------|

Set →  
 → Query

**:SYSTem:IPKHold:TIME**

|              |  |          |
|--------------|--|----------|
| Description  | Sets or queries the Ipeak hold time for peak current measurement when output on. |          |
| Syntax       | :SYSTem:IPKhold:TIME {<NR1>}   |          |
| Query Syntax | :SYSTem:IPKhold:TIME?  |          |
| Parameter    | <NR1>  | 1~60,000 |
| Example      | :SYST:IPKH:TIME 10<br>Sets the Ipeak hold time 10ms to measure when output on.   |          |

Set →  
 → Query

**:SYSTem:KLOCK**

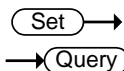
|                  |   |                              |
|------------------|---|------------------------------|
| Description      | Enables or disables the front panel key lock. |                              |
| Syntax           | :SYSTem:KLOCK {<bool> OFF ON}                 |                              |
| Query Syntax     | :SYSTem:KLOCK?                                |                              |
| Parameter        | OFF   0                                       | Panel keys unlocked          |
|                  | ON   1  | Panel keys locked            |
| Return parameter | <bool>  | Returns the key lock status. |

Set →

**:SYSTem:REBoot**

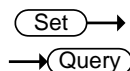
|             |                         |
|-------------|-------------------------|
| Description | Reboots the ASR system. |
| Syntax      | :SYSTem:REBoot          |

**:SYSTem:SLEW:MODE**



|                  |   |                                |
|------------------|---|--------------------------------|
| Description      | Sets or queries slew mode setting.                        |                                |
| Syntax           | :SYSTem:SLEW:MODE {<bool> TIME SLOPe}                     |                                |
| Query Syntax     | :SYSTem:SLEW:MODE?  |                                |
| Parameter        | TIME   0  | Sets the Time mode.            |
|                  | SLOPe   1   | Sets the Slope mode.           |
| Return parameter | <bool>  | Returns the slew mode setting. |
| Example          | :SYST:SLEW:MODE TIME<br>Sets the Time mode for slew mode. |                                |

**:SYSTem:VUNit**



|                  |  |                                       |
|------------------|--|---------------------------------------|
| Description      | Sets or Queries the Unit of Voltage Setting in Specific Wave Shape(TRI or ARB) |                                       |
| Syntax           | :SYSTem:VUNit <bool>   RMS   P-P   |                                       |
| Query Syntax     | :SYSTem:VUNit?   |                                       |
| Parameter        | RMS   0  | Sets V Unit (TRI, ARB) as rms         |
|                  | P-P   1  | Sets V Unit (TRI, ARB) as p-p         |
| Return parameter | <bool>   | Returns the V Unit(TRI, ARB) setting. |
| Example          | :SYSTem:VUNit?<br>+1<br>Returns the V Unit(TRI, ARB) setting.                  |                                       |

## Source Commands

|  |     |
|--|-----|
| [:SOURce]:CURRent:LIMit:PEAK:HIGH.....       | 100 |
| [:SOURce]:CURRent:LIMit:PEAK:LOW .....       | 101 |
| [:SOURce]:CURRent:LIMit:RMS[:AMPLitude].     | 101 |
| [:SOURce]:CURRent:LIMit:PEAK:MODE .....      | 102 |
| [:SOURce]:CURRent:LIMit:RMS:MODE.....        | 102 |
| [:SOURce]:FREQuency:LIMit:HIGH .....         | 103 |
| [:SOURce]:FREQuency:LIMit:LOW .....          | 103 |
| [:SOURce]:FREQuency[:IMMEDIATE] .....        | 104 |
| [:SOURce]:FUNction[:SHAPE][:IMMEDIATE] ....  | 104 |
| [:SOURce]:FUNction:THD:FORMat .....          | 105 |
| [:SOURce]:MODE .....                         | 106 |
| [:SOURce]:PHASe:STARt:STATe .....            | 107 |
| [:SOURce]:PHASe:STOP:STATe .....             | 107 |
| [:SOURce]:PHASe:STARt[:IMMEDIATE] .....      | 108 |
| [:SOURce]:PHASe:STOP[:IMMEDIATE] .....       | 108 |
| [:SOURce]:PHASe:SYNC[:IMMEDIATE] .....       | 109 |
| [:SOURce]:READ .....                         | 109 |
| [:SOURce]:VOLTagE:RANGe .....                | 110 |
| [:SOURce]:VOLTagE:LIMit:RMS.....             | 110 |
| [:SOURce]:VOLTagE:LIMit:PEAK.....            | 111 |
| [:SOURce]:VOLTagE:LIMit:HIGH .....           | 111 |
| [:SOURce]:VOLTagE:LIMit:LOW .....            | 112 |
| [:SOURce]:VOLTagE[:LEVel][:IMMEDIATE]        |     |
| [:AMPLitude] .....                           | 112 |
| [:SOURce]:VOLTagE[:LEVel][:IMMEDIATE]:OFFSet |     |
| .....  | 113 |

[:SOURce]:CURRent:LIMit:PEAK:HIGH 


|              |   |
|--------------|---|
| Description  | Sets or queries the Ipk-High Limit parameter for the continuous operation mode. |
| Syntax       | [:SOURce]:CURRent:LIMit:PEAK:HIGH<br>{<NR2> MINimum MAXimum}                    |
| Query Syntax | [:SOURce]:CURRent:LIMit:PEAK:HIGH?<br>[MINimum MAXimum]                         |

|           |         |  |
|-----------|---------|--|
| Parameter | <NR2>   | Ipk-High Limit in Arms.                  |
|           | MINimum | Minimum settable peak current high limit |
|           | MAXimum | Maximum settable peak current high limit |

Return parameter <NR2> Returns the Ipk-High Limit value

Example CURR:LIM:PEAK:HIGH?  
+42.0000  
Returns the peak current high limit as +42.0A.

→  
 →

**[[:SOURce]:CURRent:LIMit:PEAK:LOW**

Description Sets or queries the Ipk-Low Limit parameter for the continuous operation mode.

Syntax [[:SOURce]:CURRent:LIMit:PEAK:LOW  
{<NR2>|MINimum|MAXimum}

Query Syntax [[:SOURce]:CURRent:LIMit:PEAK:LOW?  
[MINimum|MAXimum]

|           |         |   |
|-----------|---------|---|
| Parameter | <NR2>   | Ipk-Low Limit in Arms.                  |
|           | MINimum | Minimum settable peak current low limit |
|           | MAXimum | Maximum settable peak current low limit |

Return parameter <NR2> Returns the Ipk-Low Limit value

Example :CURR:LIM:PEAK:LOW?  
-42.0000  
Returns the peak current low limit as -42.0A.

→  
 →

**[[:SOURce]:CURRent:LIMit:RMS[:AMPLitude]**

Description Sets or queries the Irms parameter for the continuous operation mode.

Syntax [[:SOURce]:CURRent:LIMit:RMS[:AMPLitude]  
{<NR2>|MINimum|MAXimum}

Query Syntax [[:SOURce]:CURRent:LIMit:RMS[:AMPLitude]?  
[MINimum|MAXimum]

|           |         |                          |
|-----------|---------|--------------------------|
| Parameter | <NR2>   | Irms in A.               |
|           | MINimum | Minimum settable current |
|           | MAXimum | Maximum settable current |

Return parameter <NR2> Returns the Irms.

Example :CURR:LIM:RMS?  
+10.5000  
Returns the Irms setting.

Set →

**[[:SOURce]:CURRent:LIMit:PEAK:MODE**

→ Query

Description Sets or queries Ipk limit enabled or disabled.

Syntax [[:SOURce]:CURRent:LIMit:PEAK:MODE  
{<bool>|OFF|ON}

Query Syntax [[:SOURce]:CURRent:LIMit:PEAK:MODE?

|                                |         |                  |
|--------------------------------|---------|------------------|
| Parameter/<br>Return parameter | <bool>  | OFF (0)   ON (1) |
|                                | OFF   0 | Ipk limit off    |
|                                | ON   1  | Ipk limit on     |

Example :CURR:LIM:PEAK:MODE ON  
Sets Ipk limit enabled.

Set →

**[[:SOURce]:CURRent:LIMit:RMS:MODE**

→ Query

Description Sets or queries IRMS limit status.

Syntax [[:SOURce]:CURRent:LIMit:RMS:MODE  
{<bool>|OFF|ON}

Query Syntax [[:SOURce]:CURRent:LIMit:RMS:MODE?

|                                |         |                  |
|--------------------------------|---------|------------------|
| Parameter/<br>Return parameter | <bool>  | OFF (0)   ON (1) |
|                                | OFF   0 | IRMS limit off   |
|                                | ON   1  | IRMS limit on    |

Example :CURR:LIM:RMS:MODE ON  
Sets IRMS limit enabled.

**[[:SOURce]:FREQuency:LIMit:HIGH** 



|                  |   |                             |
|------------------|---|-----------------------------|
| Description      | Sets or queries the frequency upper limit range. (Only AC+DC-INT or AC-INT or AC+DC-ADD or AC-ADD or AC-VCA Active) |                             |
| Syntax           | [:SOURce]:FREQuency:LIMit:HIGH {<NR2> MINimum MAXimum}  |                             |
| Query Syntax     | [:SOURce]:FREQuency:LIMit:HIGH? [INimum MAXimum]  |                             |
| Parameter        | <NR2>   | Frequency in Hz.            |
|                  | MINimum   | Minimum settable frequency  |
|                  | MAXimum   | Maximum settable frequency  |
| Return parameter | <NR2>   | Returns the frequency limit |
| Example          | FREQ:LIM:HIGH?<br>+999.9000<br>Returns the frequency upper limit.   |                             |

**[[:SOURce]:FREQuency:LIMit:LOW** 



|                  |   |                             |
|------------------|---|-----------------------------|
| Description      | Sets or queries the frequency lower limit range. (Only AC+DC-INT or AC-INT or AC+DC-ADD or AC-ADD or AC-VCA Active) |                             |
| Syntax           | [:SOURce]:FREQuency:LIMit:LOW {<NR2> MINimum MAXimum}   |                             |
| Query Syntax     | [:SOURce]:FREQuency:LIMit:LOW? [INimum MAXimum]   |                             |
| Parameter        | <NR2>   | Frequency in Hz.            |
|                  | MINimum   | Minimum settable frequency  |
|                  | MAXimum   | Maximum settable frequency  |
| Return parameter | <NR2>   | Returns the frequency limit |

Example           FREQ:LIM:LOW?  
                   +1.0000  
                   Returns the frequency lower limit.

Set →  
 → Query

**[[:SOURce]:FREQuency[:IMMediate]**

Description       Sets or queries the frequency for the immediate trigger. (Only AC+DC-INT or AC-INT or AC+DC-ADD or AC-ADD or AC-VCA Active)

Syntax           [:SOURce]:FREQuency[:IMMediate]  
                   {<NR2>(HZ)|MINimum|MAXimum}

Query Syntax     [:SOURce]:FREQuency[:IMMediate]?  
                   [MINimum|MAXimum]

|                            |         |                          |
|----------------------------|---------|--------------------------|
| Parameter/Return parameter | <NR2>   | Frequency setting in Hz. |
|                            | MINimum | Minimum frequency        |
|                            | MAXimum | Maximum frequency        |

Example           :FREQ 60  
                   Sets the frequency of 60Hz.

Set →  
 → Query

**[[:SOURce]:FUNCTion[:SHAPE][:IMMediate]**

Description       Sets or queries the waveforms of power supply. (Not available for DC-INT, AC+DC-EXT and AC-EXT)

Syntax           [:SOURce]:FUNCTion[:SHAPE][:IMMediate]  
                   {<NR1>|ARB1|ARB2|ARB3|ARB4|ARB5|ARB6|ARB7|ARB8|ARB9|ARB10|ARB11|ARB12|ARB13|ARB14|ARB15|ARB16|SIN|SQU|TRI}

Query Syntax     [:SOURce]:FUNCTion[:SHAPE][:IMMediate]?

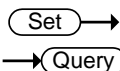
|                              |          |   |
|------------------------------|----------|---|
| Parameter / Return parameter | <NR1>    | From 0 – 18, which represent different waveforms, respectively. |
|                              | ARB1   0 | Arbitrary wave 1  |
|                              | ARB2   1 | Arbitrary wave 2  |



|            |                   |
|------------|-------------------|
| ARB3   2   | Arbitrary wave 3  |
| ARB4   3   | Arbitrary wave 4  |
| ARB5   4   | Arbitrary wave 5  |
| ARB6   5   | Arbitrary wave 6  |
| ARB7   6   | Arbitrary wave 7  |
| ARB8   7   | Arbitrary wave 8  |
| ARB9   8   | Arbitrary wave 9  |
| ARB10   9  | Arbitrary wave 10 |
| ARB11   10 | Arbitrary wave 11 |
| ARB12   11 | Arbitrary wave 12 |
| ARB13   12 | Arbitrary wave 13 |
| ARB14   13 | Arbitrary wave 14 |
| ARB15   14 | Arbitrary wave 15 |
| ARB16   15 | Arbitrary wave 16 |
| SIN   16   | Sin wave          |
| SQU   17   | Square wave       |
| TRI   18   | Triangle wave     |

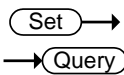
Example :SOUR:FUNC:SHAP:IMM?  
 TRI  
 Returns the waveform as Triangle wave.

**[:SOURce]:FUNctio:n:THD:FORMat**



|                              |  |                   |
|------------------------------|--|-------------------|
| Description                  | Sets or queries the THD format.                    |                   |
| Syntax                       | [:SOURce]:FUNctio:n:THD:FORMat<br>{<bool> IEC CSA} |                   |
| Query Syntax                 | [:SOURce]:FUNctio:n:THD:FORMat?                    |                   |
| Parameter / Return parameter | <bool>   | IEC (0)   CSA (1) |

|         |   |                |
|---------|---|----------------|
|         | IEC   0   | IEC THD format |
|         | CSA   1   | CSA THD format |
| Example | :SOUR:FUNC:THD:FORM?<br>IEC<br>Returns the THD format as IEC. |                |



**[[:SOURce]:MODE**

|                              |   |   |
|------------------------------|---|---|
| Description                  | Sets or queries the output mode of power supply.  |   |
| Syntax                       | [:SOURce]:MODE<br>{<NR1> ACDC-INT AC-INT DC-INT ACDC-EXT AC-EXT ACDC-ADD AC-ADD ACDC-SYNC AC-SYNC AC-VCA} |   |
| Query Syntax                 | [:SOURce]:MODE?   |   |
| Parameter / Return parameter | <NR1>   | From 0 – 9, which represent different output modes, respectively. |
|                              | ACDC-INT   0  | AC+DC-INT   |
|                              | AC-INT   1  | AC-INT  |
|                              | DC-INT   2  | DC-INT  |
|                              | ACDC-EXT   3  | AC+DC-EXT   |
|                              | AC-EXT   4  | AC-EXT  |
|                              | ACDC-ADD   5  | AC+DC-ADD   |
|                              | AC-ADD   6  | AC-ADD  |
|                              | ACDC-SYNC   7   | AC+DC-SYNC  |
|                              | AC-SYNC   8   | AC-SYNC   |
|                              | AC-VCA   9  | AC-VCA  |

|         |   |
|---------|---|
| Example | MODE?<br>AC+DC-INT<br>Returns the output mode as AC+DC-INT. |
|---------|---|

**[[:SOURce]:PHASe:STARt:STATe** 



|                                |  |   |
|--------------------------------|--|---|
| Description                    | Sets or queries state of start phase. (Not available for DC-INT, AC+DC-EXT and AC-EXT) |   |
| Syntax                         | [:SOURce]:PHASe:STARt:STATe<br>{<bool> FREE FIXED}                                     |   |
| Query Syntax                   | [:SOURce]:PHASe:STARt:STATe?   |   |
| Parameter/<br>Return parameter | <bool><br>FREE   0<br>FIXED   1  | FREE (0)   FIXED (1)<br>Start phase Free<br>Start phase Fixed |
| Example                        | :PHAS:STAR:STAT?<br>FREE<br>Returns the state of start phase as Free.                  |   |

**[[:SOURce]:PHASe:STOP:STATe** 



|                                |   |   |
|--------------------------------|---|---|
| Description                    | Sets or queries state of stop phase. (Not available for DC-INT, AC+DC-EXT and AC-EXT) |   |
| Syntax                         | [:SOURce]:PHASe:STOP:STATe<br>{<bool> FREE FIXED}                                     |   |
| Query Syntax                   | [:SOURce]:PHASe:STOP:STATe?   |   |
| Parameter/<br>Return parameter | <bool><br>FREE   0<br>FIXED   1   | FREE (0)   FIXED (1)<br>Start phase Free<br>Start phase Fixed |
| Example                        | :PHAS:STOP:STAT?<br>FIXED<br>Returns the state of stop phase as Fixed.                |   |

**[:SOURce]:PHASe:STARt[:IMMEDIATE]** 


|                            |   |                   |
|----------------------------|---|-------------------|
| Description                | Sets or queries the start phase. (Not available for DC-INT, AC+DC-EXT and AC-EXT) |                   |
| Syntax                     | [:SOURce]:PHASe:STARt[:IMMEDIATE]<br>{<NR2> MINimum MAXimum}                      |                   |
| Query Syntax               | [:SOURce]:PHASe:STARt[:IMMEDIATE]?<br>[MINimum MAXimum]                           |                   |
| Parameter/Return parameter | <NR2>   | Start phase value |
|                            | MINimum   | 0°                |
|                            | MAXimum   | 359.9°            |
| Example                    | :PHAS:STAR 0<br>Sets the starting phase to 0.                                     |                   |

**[:SOURce]:PHASe:STOP[:IMMEDIATE]** 


|                            |   |                  |
|----------------------------|---|------------------|
| Description                | Sets or queries the off phase of the waveform. (Not available for DC-INT, AC+DC-EXT and AC-EXT) |                  |
| Note:                      | Sets the off phase of the waveform after the output has been turned off.                        |                  |
| Syntax                     | [:SOURce]:PHASe:STOP[:IMMEDIATE]<br>{<NR2> MINimum MAXimum}                                     |                  |
| Query Syntax               | [:SOURce]:PHASe:STOP[:IMMEDIATE]?<br>[MINimum MAXimum]  |                  |
| Parameter/Return parameter | <NR2>   | Stop phase value |
|                            | MINimum   | 0°               |
|                            | MAXimum   | 359.9°           |
| Example                    | :PHAS:STOP 60<br>Sets the stop phase to 60.   |                  |

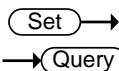
**[[:SOURce]:PHASe:SYNC[:IMMEdiate]]** 



|                            |   |  |
|----------------------------|---|--|
| Description                | Sets or queries the sync delay phase. (Only AC+DC-sync or AC-sync Active) |  |
| Syntax                     | [:SOURce]:PHASe:SYNC[:IMMEdiate]{<NR2> MINimum MAXimum}                   |  |
| Query Syntax               | [:SOURce]:PHASe:SYNC[:IMMEdiate]?[MINimum MAXimum]                        |  |
| Parameter/Return parameter | <NR2><br>MINimum<br>MAXimum   | Sync delay phase value<br>0 °<br>359.9 ° |
| Example                    | :PHAS:SYNC 0<br>Sets the sync delay phase to 0.                           |  |

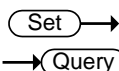
**[[:SOURce]:READ]** 

|                  |  |   |
|------------------|--|---|
| Description      | Returns the measurement readouts.  |   |
| Query Syntax     | [:SOURce]:READ?  |   |
| Return parameter | <Vrms>,<Vavg>,<Vmax>,<Vmin>,<Irms>,<lavg>,<Imax>,<Imin>,<IpkH>,<P>,<S>,<Q>,<PF>,<CF>,<THDv>,<THDi>,<Freq>  | <THDv>,<THDi> returns values in AC-INT mode only, whereas returns Invalid in other modes.<br><S>,<Q>,<PF>,<CF> returns Invalid in DC-INT mode.<br><Freq> returns values in AC+DC-Sync and AC-Sync modes only, whereas returns Invalid in other modes. |
| Example          | :READ?<br>>+0.3204,+0.0306,+0.1879,-0.5809,+0.0121, -0.0007,<br>+0.0030, -0.0060, -0.0201, +0.0013, +0.0039, +0.0037,<br>+0.3400, +1.1500, Invalid, Invalid, Invalid |   |



**[:SOURce]:VOLTage:RANGe**

|                              |   |   |
|------------------------------|---|---|
| Description                  | Sets or queries the voltage range.                            |   |
| Syntax                       | [:SOURce]:VOLTage:RANGe<br>{<NR1> 100 200 AUTO}               |   |
| Query Syntax                 | [:SOURce]:VOLTage:RANGe?                                      |   |
| Parameter / Return parameter | <NR1>   | From 0 - 2, which represent different voltage ranges, respectively.       |
|                              | 100   0   | 100V  |
|                              | 200   1   | 200V  |
|                              | AUTO   2  | AUTO (Only AC+DC-INT or AC-INT or DC-INT or AC+DC-sync or AC-sync Active) |
| Example                      | :SOUR:VOLT:RANG?<br>200<br>Returns the voltage range as 200V. |   |



**[:SOURce]:VOLTage:LIMit:RMS**

|                  |  |                            |
|------------------|--|----------------------------|
| Description      | Sets or queries the voltage limit for the continuous operation mode. (Only AC-INT or AC-ADD or AC-Sync Active) |                            |
| Syntax           | [:SOURce]:VOLTage:LIMit:RMS<br>{<NR2> MINimum MAXimum}   |                            |
| Query Syntax     | [:SOURce]:VOLTage:LIMit:RMS?<br>[MINimum MAXimum]  |                            |
| Parameter        | <NR2>  | Vrms.                      |
|                  | MINimum  | Minimum voltage limit      |
|                  | MAXimum  | Maximum voltage limit      |
| Return parameter | <NR2>  | Returns the voltage limit. |

Example            VOLT:LIM:RMS?  
                       +350.0000  
                       Returns the Vrms limit.

**[[:SOURce]:VOLTage:LIMit:PEAK**

Set →

→ Query

Description       Sets or Queries the Value of Vpp in Specific Mode(AC-INT or AC-ADD or AC-Sync) and Specific Wave Shape(TRI or ARB) and Specific V Unit(p-p)

Syntax            [[:SOURce]:VOLTage:LIMit:PEAK <NR2> | MINimum | MAXimum

Query Syntax     [[:SOURce]:VOLTage:LIMit:PEAK?  
                       [ MINimum | MAXimum ]

|           |         |                   |
|-----------|---------|-------------------|
| Parameter | <NR2>   | Vpp               |
|           | MINimum | Minimum Vpp limit |
|           | MAXimum | Maximum Vpp limit |

Return parameter <NR2>    Returns the Vpp limit.

Example            VOLT:LIM:PEAK?  
                       +500.0000  
                       Returns the Vpp limit.

**[[:SOURce]:VOLTage:LIMit:HIGH**

Set →

→ Query

Description       Sets or queries the voltage high limit. (Only AC+DC-INT or DC-INT or AC+DC-ADD or AC+DC-Sync Active)

Syntax            [[:SOURce]:VOLTage:LIMit:HIGH  
                       {<NR2>|MINimum|MAXimum}

Query Syntax     [[:SOURce]:VOLTage:LIMit:HIGH?  
                       [MINimum|MAXimum]

|           |         |                            |
|-----------|---------|----------------------------|
| Parameter | <NR2>   | Voltage high limit         |
|           | MINimum | Minimum voltage high limit |

|                  |  |                                 |
|------------------|--|---------------------------------|
|                  | <b>MAXimum</b>   | Maximum voltage high limit      |
| Return parameter | <NR2>  | Returns the voltage high limit. |
| Example          | VOLT:LIM:HIGH?<br>+500.0000<br>Returns the voltage high limit. |                                 |

Set →  
 → Query

|                  |   |                                |
|------------------|---|--------------------------------|
| Description      | Sets or queries the voltage low limit. (Only AC+DC-INT or DC-INT or AC+DC-ADD or AC+DC-Sync Active) |                                |
| Syntax           | [:SOURce]:VOLTage:LIMit:LOW {<NR2> MINimum MAXimum}   |                                |
| Query Syntax     | [:SOURce]:VOLTage:LIMit:LOW? [MINimum MAXimum]  |                                |
| Parameter        | <NR2>   | Voltage low limit              |
|                  | MINimum   | Minimum voltage low limit      |
|                  | MAXimum   | Maximum voltage low limit      |
| Return parameter | <NR2>   | Returns the voltage low limit. |
| Example          | VOLT:LIM:LOW?<br>-500.0000<br>Returns the voltage low limit.  |                                |

Set →  
 → Query

|              |  |  |
|--------------|--|--|
| Description  | Sets or queries the RMS voltage for the continuous operation mode. (Not available for DC-INT, AC+DC-EXT and AC-EXT and AC-VCA) |  |
| Syntax       | [:SOURce]:VOLTage[:LEVel][:IMMediate][:AMPLitude] {<NR2>(V) MINimum MAXimum}   |  |
| Query Syntax | [:SOURce]:VOLTage[:LEVel][:IMMediate][:AMPLitude]? [MINimum MAXimum]   |  |



|                            |         |                 |
|----------------------------|---------|-----------------|
| Parameter/Return parameter | <NR2>   | Vrms.           |
|                            | MINimum | Minimum voltage |
|                            | MAXimum | Maximum voltage |

Example :VOLT 150.0  
Sets the voltage to 150.0 ACV.

Set →  
Query ←

**[:SOURce]:VOLTage[:LEVel][:IMMediate]:OFFSet**

Description Sets or queries the voltage offset value. (Only AC+DC-INT or DC-INT or AC+DC-ADD or AC+DC-Sync Active)

Syntax [:SOURce]:VOLTage[:LEVel][:IMMediate]:OFFSet {<NR2>(V)|MINimum|MAXimum}

Query Syntax [:SOURce]:VOLTage[:LEVel][:IMMediate]:OFFSet? [MINimum|MAXimum]

|                            |         |                              |
|----------------------------|---------|------------------------------|
| Parameter/Return parameter | <NR2>   | Voltage offset value         |
|                            | MINimum | Minimum voltage offset value |
|                            | MAXimum | Maximum voltage offset value |

Example :VOLT:OFFS?  
+150.0000  
Returns the voltage offset value as 150.0.

## Sequence Commands

|  |     |
|--|-----|
| [:SOURce]:SEQUence:CPARAmeter .....      | 114 |
| [:SOURce]:SEQUence:CSTep .....           | 115 |
| [:SOURce]:SEQUence:CJUMp:CNT .....       | 115 |
| [:SOURce]:SEQUence:CTIME .....           | 116 |
| [:SOURce]:SEQUence:SPARAmeter .....      | 116 |
| [:SOURce]:SEQUence:STEP .....            | 117 |
| [:SOURce]:SEQUence:CONDition .....       | 118 |
| :TRIGger:SEQUence:SELEcted:EXECute ..... | 118 |

Set →  
 → Query

### [:SOURce]:SEQUence:CPARAmeter

|              |  |  |
|--------------|--|--|
| Description  | Sets the common parameters for the Sequence mode. Please see the user manual for a full description of each parameter (Only Sequence Mode Active).   |  |
| Syntax       | [:SOURce]:SEQUence:CPARAmeter<br>{<NR2>,<NR2>,<bool> OFF ON,<NR2>,<bool> OFF ON,<NR1> CONTInue END HOLD,<NR1>,<bool> OFF ON,<NR1>,<bool> OFF ON,<NR1>,<bool> OFF ON,<NR1>,<bool> OFF ON,<NR1>,<bool> OFF ON} |  |
| Query Syntax | [:SOURce]:SEQUence:CPARAmeter?   |  |
| Parameter    | <NR2>  | Step Time  |
|              | <NR2>  | On phase   |
|              | <bool> OFF ON FREE FIXED   | On Phase settings:<br>on (fixed) (1) / off (free) (0)  |
|              | <NR2>  | Off phase  |
|              | <bool> OFF ON FREE FIXED   | Off Phase settings:<br>on (fixed) (1) / off (free) (0) |
|              | <NR1> CONTInue END HOLD  | Term settings:<br>Continue(0)/End(1)/Hold(2)           |
|              | <NR1>  | Jump step number (0 ~ 999)                             |

|                  |   |  |
|------------------|---|--|
|                  | <bool> OFF ON   | Jump on(1)/off(0)                        |
|                  | <NR1>   | Jump Cnt (0~ 9999)                       |
|                  | <NR1>   | Sync Code: LL(0) / LH(1) / HL(2) / HH(3) |
|                  | <NR1>   | Branch1 (0 ~ 999)                        |
|                  | <bool> OFF ON   | Branch1 on(1)/off(0)                     |
|                  | <NR1>   | Branch2 (0 ~ 999)                        |
|                  | <bool> OFF ON   | Branch2 on(1)/off(0)                     |
|                  | <bool>  | Reserved (Fixed to 0)                    |
| Return parameter | <NR2>,<NR2>,<bool>,<NR2>,<bool>,<NR1>,<NR1>,<bool>,<NR1>,<NR1>,<bool>,<NR1>,<bool>,<bool>   |  |
|                  | Returns the common parameters in the following order:<br>Step time, on phase, on phase on/off, off phase, off phase on/off, term settings, jump step number, jump on/off, jump count, code on/off, branch1, branch1 on/off, branch2, branch2 on/off, reserved on/off. |  |
| Example1         | :SEQ:CPAR 1,1,0,10,1,HOLD,10,1,0,1,0,0,0,0  |  |
| Example2         | :SEQ:CPAR?<br>+1.0000,+1.0,+0,+10.0,+1,HOLD,+10,+1,+0,+1,+0,+0,+0,+0,+0   |  |

## [:SOURce]:SEQuence:CSTep

→ Query

|                  |  |                     |
|------------------|--|---------------------|
| Description      | Returns the currently running step number (Only Sequence Mode Active). |                     |
| Query Syntax     | [:SOURce]:SEQuence:CSTep?  |                     |
| Return parameter | <NR1>  | Current step number |
| Example          | :SEQ:CSTep?<br>+1  |                     |

## [:SOURce]:SEQuence:CJUMp:Cnt

→ Query

|             |  |  |
|-------------|--|--|
| Description | Returns the currently running step & Jump count. (Only Sequence Mode Active) |  |
|-------------|--|--|

|                  |                               |                        |
|------------------|-------------------------------|------------------------|
| Query Syntax     | [:SOURce]:SEQuence:CJUMp:Cnt? |                        |
| Return parameter | <NR1>                         | Current step number    |
|                  | <NR1>                         | Jump count(Inf,1~9999) |
| Example          | :SEQ:CJUM:Cnt?<br>+2,+9       |                        |

**[:SOURce]:SEQuence:CTIME → Query**

|                  |  |                      |
|------------------|--|----------------------|
| Description      | Returns the currently running step & Elapsed time. (Only Sequence Mode Active) |                      |
| Query Syntax     | [:SOURce]:SEQuence:CTIME?  |                      |
| Return parameter | <NR1>  | Current step number  |
|                  | <NR1>  | Elapsed time(Hour)   |
|                  | <NR1>  | Elapsed time(Minute) |
|                  | <NR1>  | Elapsed time(Second) |
| Example          | :SEQ:CTIM?<br>+1,+0,+0,+14   |                      |

Set →

**[:SOURce]:SEQuence:SPARAmeter → Query**

|              |   |  |
|--------------|---|--|
| Description  | Sets or queries the parameters for a specified step (Only Sequence Mode Active).  |  |
| Syntax       | [:SOURce]:SEQuence:SPARAmeter<br>{<NR2>,<NR1> CONSt KEEp SWEEp,<NR2>,<NR1> CONSt KEEp SWEEp,<NR2>,<NR1> CONSt KEEp SWEEp,SIN,<NR1>} |  |
| Query Syntax | [:SOURce]:SEQuence:SPARAmeter?  |  |
| Parameter    | <NR2>   | ACV setting  |
|              | <NR1> CONSt KEEp SWEEp  | ACV mode: Constant(0)   Keep(1)   Sweep(2)           |
|              | <NR2>   | DCV. Not applicable. This parameter will be ignored. |

|                  |   |   |
|------------------|---|---|
|                  | <NR1> CONSt <br>KEEp SWEep  | DCV mode: Constant(0)   Keep(1)<br>  Sweep(2)   |
|                  | <NR2>   | Frequency   |
|                  | <NR1> CONSt <br>KEEp SWEep  | Frequency mode: Constant(0)  <br>Keep(1)   Sweep(2)   |
|                  | Waveform  | ARB1   ARB2   ARB3   ARB4   ARB5<br>  ARB6   ARB7   ARB8   ARB9   ARB<br>10   ARB11   ARB12   ARB13   ARB1<br>4   ARB15   ARB16   SIN   SQU   TRI |
|                  | <NR1>   | Phase angle. Fixed to 0.  |
| Return parameter | <NR2>,<NR1>   CONSt   KEEp   SWEep,<NR2>,<NR1><br>  CONSt   KEEp   SWEep,<NR2>,<NR1>   CONSt   KEEp<br>  SWEep,ARB1   ARB2   ARB3   ARB4   ARB5   ARB6   AR<br>B7   ARB8   ARB9   ARB10   ARB11   ARB12   ARB13   ARB<br>14   ARB15   ARB16   SIN   SQU   TRI,<NR1> |   |
|                  | Returns the step parameters in the following order:<br>ACV, ACV mode, DCV, DCV mode, frequency,<br>frequency mode, wave, phase.   |   |

Example :SEQ:SPAR?  
+0.0,CONST,+0.0,CONST,+50.00,CONST,SIN,0

[[:SOURce]:SEQuence:STEP (Set) →  
→ (Query)

|                            |  |                     |
|----------------------------|--|---------------------|
| Description                | Sets or queries the current step number (Only Sequence Mode Active). |                     |
| Syntax                     | [:SOURce]:SEQuence:STEP<br>{<NR1> MINimum MAXimum}                   |                     |
| Query Syntax               | [:SOURce]:SEQuence:STEP? [MINimum MAXimum]                           |                     |
| Parameter/Return parameter | <NR1>  | Step number         |
|                            | MINimum  | Minimum step number |
|                            | MAXimum  | Maximum step number |

Example :SEQ:STEP 1  
Sets the step number to 1.

**[:SOURce]:SEQuence:CONDition** → (Query)

|                  |   |                         |  |
|------------------|---|-------------------------|--|
| Description      | Returns the sequence status.(Only Sequence Mode Active) |                         |  |
| Query Syntax     | [:SOURce]:SEQuence:CONDition?                           |                         |  |
| Return parameter | <NR1>   | Current sequence status | 0 (Idle mode)<br>1 (Run mode)<br>2 (Hold mode) |
| Example          | :SEQ:COND?<br>1   |                         |  |

**:TRIGger:SEQuence:SELEcted:EXECute** (Set) →

|             |  |                             |  |
|-------------|--|-----------------------------|--|
| Description | Sets to execute actions for sequence mode. (Only Sequence Mode Active) |                             |  |
| Syntax      | :TRIGger:SEQuence:SELEcted:EXECute {STOP STARt HOLD BRAN1 BRAN2}       |                             |  |
| Parameter   | STOP   | Stops sequence execution    |  |
|             | STARt  | Starts sequence execution   |  |
|             | HOLD   | Holds sequence execution    |  |
|             | BRAN1  | Jumps to Branch 1 execution |  |
|             | BRAN2  | Jumps to Branch 2 execution |  |
| Example     | TRIG:SEQ:SEL:EXEC STAR<br>Starts sequence execution.                   |                             |  |

## Simulate Commands

|   |     |
|---|-----|
| [.:SOURce]:SIMulation:CONDition .....                       | 120 |
| [.:SOURce]:SIMulation:ABNormal:CODE .....                   | 120 |
| [.:SOURce]:SIMulation:ABNormal:FREQuency .....              | 121 |
| [.:SOURce]:SIMulation:ABNormal:PHASe                        |     |
| :START:ENABle .....   | 121 |
| [.:SOURce]:SIMulation:ABNormal:PHASe                        |     |
| :START[:IMMEdiate] .....                                    | 122 |
| [.:SOURce]:SIMulation:ABNormal:PHASe                        |     |
| :STOP:ENABle .....  | 122 |
| [.:SOURce]:SIMulation:ABNormal:PHASe                        |     |
| :STOP[:IMMEdiate] .....                                     | 123 |
| [.:SOURce]:SIMulation:ABNormal:TIME .....                   | 123 |
| [.:SOURce]:SIMulation:ABNormal:VOLTage .....                | 124 |
| [.:SOURce]:SIMulation:CSTep .....                           | 124 |
| [.:SOURce]:SIMulation:CREPeat:COUNT .....                   | 125 |
| [.:SOURce]:SIMulation:CTIME .....                           | 125 |
| [.:SOURce]:SIMulation:INITial:CODE .....                    | 126 |
| [.:SOURce]:SIMulation:INITial:FREQuency .....               | 126 |
| [.:SOURce]:SIMulation:INITial:PHASe:START:ENABle .....      | 127 |
| [.:SOURce]:SIMulation:INITial:PHASe:START[:IMMEdiate] ..... | 127 |
| [.:SOURce]:SIMulation:INITial:PHASe:STOP:ENABle .....       | 128 |
| [.:SOURce]:SIMulation:INITial:PHASe:STOP[:IMMEdiate] .....  | 128 |
| [.:SOURce]:SIMulation:INITial:VOLTage .....                 | 129 |
| [.:SOURce]:SIMulation:NORMal<1   2>:CODE .....              | 129 |
| [.:SOURce]:SIMulation:NORMal1:FREQuency .....               | 130 |
| [.:SOURce]:SIMulation:NORMal<1   2>                         |     |
| :PHASe:START:ENABle .....                                   | 130 |
| [.:SOURce]:SIMulation:NORMal<1   2>                         |     |
| :PHASe:START[:IMMEdiate] .....                              | 131 |
| [.:SOURce]:SIMulation:NORMal<1   2>                         |     |
| :PHASe:STOP:ENABle .....                                    | 131 |
| [.:SOURce]:SIMulation:NORMal<1   2>                         |     |

:PHASe:STOP[:IMMediate].....132  
 [:SOURce]:SIMulation:NORMal<1 | 2>:TIME ...132  
 [:SOURce]:SIMulation:NORMal1:VOLTage.....133  
 [:SOURce]:SIMulation:REPeat:COUNt .....133  
 [:SOURce]:SIMulation:REPeat:ENABle.....134  
 [:SOURce]:SIMulation:TRANsition<1 | 2>:TIME  
 .....134  
 [:SOURce]:SIMulation:TRANsition<1 | 2>:CODE  
 .....135  
 :TRIGger:SIMulation:SELEcted:EXECute.....135

**[:SOURce]:SIMulation:CONDition** → Query

**Description** Returns the simulation status. (Only Simulation Mode Active)

**Query Syntax** [:SOURce]:SIMulation:CONDition?

|                         |       |                           |               |
|-------------------------|-------|---------------------------|---------------|
| <b>Return parameter</b> | <NR1> | Current simulation status | 0 (Idle mode) |
|                         |       |                           | 1 (Run mode)  |
|                         |       |                           | 2 (Hold mode) |

**Example** :SIM:COND?  
1

Set →

**[:SOURce]:SIMulation:ABNormal:CODE** → Query

**Description** Sets the external trigger output for the abnormal step parameter. This option is only applicable when in the Simulation mode (Only Simulation Mode Active).

**Syntax** [:SOURce]:SIMulation:ABNormal:CODE {<NR1>|MINimum|MAXimum}

**Query Syntax** [:SOURce]:SIMulation:ABNormal:CODE? [MINimum|MAXimum]

**Parameter/Return** <NR1> External trigger output, 0=LL, 1=LH, 2=HL, 3=HH.



|           |         |        |
|-----------|---------|--------|
| parameter | MINimum | 0 (LL) |
|           | MAXimum | 3 (HH) |

Example SIM:ABN:CODE 1

`[[:SOURce]:SIMulation:ABNormal:FREQuency` (Set) →  
→ (Query)

Description Sets or queries the frequency of the abnormal step of the simulation mode (Only Simulation Mode Active).

Syntax `[[:SOURce]:SIMulation:ABNormal:FREQuency {<NR2>|MINimum|MAXimum}`

Query Syntax `[[:SOURce]:SIMulation:ABNormal:FREQuency? [MINimum|MAXimum]`

|                            |         |                            |
|----------------------------|---------|----------------------------|
| Parameter/Return parameter | <NR2>   | Frequency of abnormal step |
|                            | MINimum | Minimum frequency          |
|                            | MAXimum | Maximum frequency          |

Example `:SIM:ABN:FREQ 55`  
 Sets the frequency to 55Hz.

`[[:SOURce]:SIMulation:ABNormal:PHASe :START:ENABLE` (Set) →  
→ (Query)

Description Enables/Disables (Fixed/Free) the ON Phs parameter of the abnormal step for the Simulation mode (Only Simulation Mode Active).

Syntax `[[:SOURce]:SIMulation:ABNormal:PHASe:START:ENABLE {<bool>|OFF|ON|FREE|FIXED}`

Query Syntax `[[:SOURce]:SIMulation:ABNormal:PHASe:START:ENABLE?`

|                            |                |          |
|----------------------------|----------------|----------|
| Parameter/Return parameter | OFF   0   FREE | Disabled |
|                            | ON   1   FIXED | Enabled  |

Example `:SIM:ABN:PHAS:STAR:ENAB 1`  
 Enable the ON Phs.

`[[:SOURce]:SIMulation:ABNormal:PHASe` (Set) →  
`:START[:IMMediate]` → (Query)

|                            |  |                      |
|----------------------------|--|----------------------|
| Description                | Sets or queries the ON Phs parameter of the abnormal step for the Simulation mode (Only Simulation Mode Active). |                      |
| Syntax                     | <code>[[:SOURce]:SIMulation:ABNormal:PHASe:START[:IMMediate] {&lt;NR2&gt; MINimum MAXimum}</code>                |                      |
| Query Syntax               | <code>[[:SOURce]:SIMulation:ABNormal:PHASe:START[:IMMediate]? [MINimum MAXimum]</code>                           |                      |
| Parameter/Return parameter | <code>&lt;NR2&gt;</code>   | ON Phs (start phase) |
|                            | MINimum  | 0                    |
|                            | MAXimum  | 359.9                |

Example `:SIM:ABN:PHAS:STAR 0`  
 Sets ON Phs to 0.


`[[:SOURce]:SIMulation:ABNormal:PHASe` (Set) →  
`:STOP:ENABLE` → (Query)

|                            |   |          |
|----------------------------|---|----------|
| Description                | Enables/Disables (Fixed/Free) the OFF Phs parameter of the abnormal step for the Simulation mode (Only Simulation Mode Active). |          |
| Syntax                     | <code>[[:SOURce]:SIMulation:ABNormal:PHASe:STOP:ENABLE {&lt;bool&gt; OFF ON FREE FIXED }</code>                                 |          |
| Query Syntax               | <code>[[:SOURce]:SIMulation:ABNormal:PHASe:STOP:ENABLE?</code>  |          |
| Parameter/Return parameter | <code>OFF   0   FREE</code>   | Disabled |
|                            | <code>ON   1   FIXED</code>   | Enabled  |

Example `:SIM:ABN:PHAS:STOP:ENAB 1`  
 Enable the OFF Phs.

`[[:SOURce]:SIMulation:ABNormal:PHASe` (Set) →  
`:STOP[:IMMEDIATE]` → (Query)

**Description**      Sets or queries the OFF Phs parameter of the abnormal step for the Simulation mode (Only Simulation Mode Active).

 **Note**      Sets the off phase of the waveform after the output has been turned off.

**Syntax**            `[[:SOURce]:SIMulation:ABNormal:PHASe:STOP`  
`[:IMMEDIATE] {<NR2>|MINimum|MAXimum}`

**Query Syntax**    `[[:SOURce]:SIMulation:ABNormal:PHASe:STOP`  
`[:IMMEDIATE]? [MINimum|MAXimum]`

|                                   |                    |                      |
|-----------------------------------|--------------------|----------------------|
| <b>Parameter/Return parameter</b> | <b>&lt;NR2&gt;</b> | OFF Phs (Stop phase) |
|                                   | <b>MINimum</b>     | 0                    |
|                                   | <b>MAXimum</b>     | 359.9                |

**Example**            `:SIM:ABN:PHAS:STOP 0`  
 Sets OFF Phs to 0.

`[[:SOURce]:SIMulation:ABNormal:TIME` (Set) →  
 → (Query)

**Description**      Sets or queries the Time parameter of the abnormal step for the Simulation mode (Only Simulation Mode Active).

**Syntax**            `[[:SOURce]:SIMulation:ABNormal:TIME`  
`{<NR2>|MINimum|MAXimum}`

**Query Syntax**    `[[:SOURce]:SIMulation:ABNormal:TIME?`  
`[MINimum|MAXimum]`

|                                   |                    |                                      |
|-----------------------------------|--------------------|--------------------------------------|
| <b>Parameter/Return parameter</b> | <b>&lt;NR2&gt;</b> | Time of the abnormal step in seconds |
|                                   | <b>MINimum</b>     | 0.0001                               |
|                                   | <b>MAXimum</b>     | 999.9999s                            |

**Example**            `:SIM:ABN:TIME 1`  
 Sets the abnormal step time to 1 second.

**[:SOURce]:SIMulation:ABNormal:VOLTage** 



|                            |  |                               |
|----------------------------|--|-------------------------------|
| Description                | Sets or queries the Vset parameter of the abnormal step for the Simulation mode (Only Simulation Mode Active). |                               |
| Syntax                     | [:SOURce]:SIMulation:ABNormal:VOLTage {<NR2> MINimum MAXimum}  |                               |
| Query Syntax               | [:SOURce]:SIMulation:ABNormal:VOLTage? [MINimum MAXimum]   |                               |
| Parameter/Return parameter | <NR2>  | Voltage of the abnormal step. |
|                            | MINimum  | Minimum settable voltage      |
|                            | MAXimum  | Maximum settable voltage      |
| Example                    | :SIM:ABN:VOLT MAX<br>Sets the abnormal step voltage to the maximum.  |                               |

**[:SOURce]:SIMulation:CSTep** 


|                  |   |                       |
|------------------|---|-----------------------|
| Description      | Returns the currently running step (Only Simulation Mode Active). |                       |
| Query Syntax     | [:SOURce]:SIMulation:CSTep?                                       |                       |
| Return parameter | <NR1>   | Current step          |
|                  |   | +0 = Initial step     |
|                  |   | +1 = Normal1 step     |
|                  |   | +2 = Transition1 step |
|                  |   | +3 = Abnormal step    |
|                  |   | +4 = Transition2 step |
|                  | +5 = Normal2 step   |                       |
| Example          | :SIM:CSTep?<br>+1   |                       |

**[[:SOURce]:SIMulation:CREPeat:COUNT** → **Query**

**Description** Returns the currently running step & Repeat count. (Only Simulation Mode Active)

**Query Syntax** [[:SOURce]:SIMulation:CREPeat:COUNT?

|                         |                    |  |
|-------------------------|--------------------|--|
| <b>Return parameter</b> | <b>&lt;NR1&gt;</b> | Current step number<br>+0 = Initial step<br>+1 = Normal1 step<br>+2 = Transition1 step<br>+3 = Abnormal step<br>+4 = Transition2 step<br>+5 = Normal2 step |
|                         | <b>&lt;NR1&gt;</b> | Repeat count (Inf,1~9999)  |

**Example** :SIM:CREP:COUNT?  
+5,+3

**[[:SOURce]:SIMulation:CTIME** → **Query**

**Description** Returns the currently running step & Elapsed time. (Only Simulation Mode Active)

**Query Syntax** [[:SOURce]:SIMulation:CTIME?

|                         |                    |  |
|-------------------------|--------------------|--|
| <b>Return parameter</b> | <b>&lt;NR1&gt;</b> | Current step number<br>+0 = Initial step<br>+1 = Normal1 step<br>+2 = Transition1 step<br>+3 = Abnormal step<br>+4 = Transition2 step<br>+5 = Normal2 step |
|                         | <b>&lt;NR1&gt;</b> | Elapsed time (Hour)  |
|                         | <b>&lt;NR1&gt;</b> | Elapsed time (Minute)  |
|                         | <b>&lt;NR1&gt;</b> | Elapsed time (Second)  |

**Example** :SIM:CTIME?  
+1,+0,+0,+10

Set →  
 → Query

---

**[:SOURce]:SIMulation:INITial:CODE**

|                            |  |                        |
|----------------------------|--|------------------------|
| Description                | Sets the external trigger output for the initial step parameter. This option is only applicable when in the Simulation mode (Only Simulation Mode Active). |                        |
| Syntax                     | [:SOURce]:SIMulation:INITial:CODE {<NR1> MINimum MAXimum}  |                        |
| Query Syntax               | [:SOURce]:SIMulation:INITial:CODE? [MINimum MAXimum]   |                        |
| Parameter/Return parameter | <NR1>  | 0=LL, 1=LH, 2=HL, 3=HH |
|                            | MINimum  | 0 (LL)                 |
|                            | MAXimum  | 3 (HH)                 |
| Example                    | SIM:INIT:CODE 1  |                        |

Set →  
 → Query

---

**[:SOURce]:SIMulation:INITial:FREQuency**



|                            |   |                           |
|----------------------------|---|---------------------------|
| Description                | Sets or queries the frequency of the initial step of the simulation mode (Only Simulation Mode Active). |                           |
| Syntax                     | [:SOURce]:SIMulation:INITial:FREQuency {<NR2> MINimum MAXimum}  |                           |
| Query Syntax               | [:SOURce]:SIMulation:INITial:FREQuency? [MINimum MAXimum]   |                           |
| Parameter/Return parameter | <NR2>   | Frequency of initial step |
|                            | MINimum   | Minimum frequency         |
|                            | MAXimum   | Maximum frequency         |
| Example                    | :SIM:INIT:FREQ 60<br>Sets the frequency of the initial step to 60Hz.                                    |                           |

**[[:SOURce]:SIMulation:INITial:PHASe:STARt:ENABle** (Set) →  
→ (Query)

|                            |   |          |
|----------------------------|---|----------|
| Description                | Enables/Disables (Fixed/Free) the ON Phs parameter of the initial step for the Simulation mode (Only Simulation Mode Active). |          |
| Syntax                     | [:SOURce]:SIMulation:INITial:PHASe:STARt:ENABle {<bool> OFF ON FREE FIXED}  |          |
| Query Syntax               | [:SOURce]:SIMulation:INITial:PHASe:STARt:ENABle?  |          |
| Parameter/Return parameter | OFF   0   FREE  | Disabled |
|                            | ON   1   FIXED  | Enabled  |
| Example                    | :SIM:INIT:PHAS:STAR:ENAB 1<br>Enable the ON Phs.  |          |

**[[:SOURce]:SIMulation:INITial:PHASe:STARt[:IMMediate]** (Set) →  
→ (Query)

|                            |   |                      |
|----------------------------|---|----------------------|
| Description                | Sets or queries the ON Phs parameter of the initial step for the Simulation mode (Only Simulation Mode Active). |                      |
| Syntax                     | [:SOURce]:SIMulation:INITial:PHASe:STARt[:IMMediate] {<NR2> MINimum MAXimum}                                    |                      |
| Query Syntax               | [:SOURce]:SIMulation:INITial:PHASe:STARt[:IMMediate]? [MINimum MAXimum]   |                      |
| Parameter/Return parameter | <NR2>   | ON Phs (start phase) |
|                            | MINimum   | 0                    |
|                            | MAXimum   | 359.9                |
| Example                    | :SIM:INIT:PHAS:STAR 0<br>Sets ON Phs to 0.  |                      |

`[:SOURce]:SIMulation:INITial:PHASe:STOP`    
`:ENABLE` 



**Description** Enables/Disables (Fixed/Free) the OFF Phs parameter of the initial step for the Simulation mode (Only Simulation Mode Active).

**Syntax** `[:SOURce]:SIMulation:INITial:PHASe:STOP:ENABLE {<bool>|OFF|ON|FREE|FIXED}`

**Query Syntax** `[:SOURce]:SIMulation:INITial:PHASe:STOP:ENABLE?`

|                                   |                             |          |
|-----------------------------------|-----------------------------|----------|
| <b>Parameter/Return parameter</b> | <code>OFF   0   FREE</code> | Disabled |
|                                   | <code>ON   1   FIXED</code> | Enabled  |

**Example** `:SIM:INIT:PHAS:STOP:ENAB 1`  
 Enable the OFF Phs.

`[:SOURce]:SIMulation:INITial:PHASe:STOP[:IM`    
`Mediate]` 

**Description** Sets or queries the OFF Pha parameter of the initial step for the Simulation mode (Only Simulation Mode Active).



**Note**

Sets the off phase of the waveform after the output has been turned off.

**Syntax** `[:SOURce]:SIMulation:INITial:PHASe:STOP [:IMMediate] {<NR2>|MINimum|MAXimum}`

**Query Syntax** `[:SOURce]:SIMulation:INITial:PHASe:STOP [:IMMediate]? [MINimum|MAXimum]`

|                                   |                          |                      |
|-----------------------------------|--------------------------|----------------------|
| <b>Parameter/Return parameter</b> | <code>&lt;NR2&gt;</code> | OFF Phs (Stop phase) |
|                                   | <code>MINimum</code>     | 0                    |
|                                   | <code>MAXimum</code>     | 359.9                |

**Example** `:SIM:INIT:PHAS:STOP 0`  
 Sets OFF Phs to 0.



Set →

→ Query

**[[:SOURce]:SIMulation:INITial:VOLTage**

|                            |   |                              |
|----------------------------|---|------------------------------|
| Description                | Sets or queries the Vset parameter of the initial step for the Simulation mode (Only Simulation Mode Active). |                              |
| Syntax                     | [:SOURce]:SIMulation:INITial:VOLTage {<NR2> MINimum MAXimum}  |                              |
| Query Syntax               | [:SOURce]:SIMulation:INITial:VOLTage? [MINimum MAXimum]   |                              |
| Parameter/Return parameter | <NR2>   | Voltage of the initial step. |
|                            | MINimum   | Minimum settable voltage     |
|                            | MAXimum   | Maximum settable voltage     |
| Example                    | :SIM:INIT:VOLT MAX<br>Sets the initial step voltage to the maximum.   |                              |

Set →

→ Query


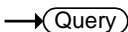
**[[:SOURce]:SIMulation:NORMa|<1|2>:CODE**

|                            |   |                        |
|----------------------------|---|------------------------|
| Description                | Sets the external trigger output for the normal 1 or normal 2 step parameter. This option is only applicable when in the Simulation mode (Only Simulation Mode Active). |                        |
| Syntax                     | [:SOURce]:SIMulation:NORMa <1 2>:CODE {<NR1> MINimum MAXimum}   |                        |
| Query Syntax               | [:SOURce]:SIMulation:NORMa <1 2>:CODE? [MINimum MAXimum]  |                        |
| Parameter/Return parameter | <NR1>   | 0=LL, 1=LH, 2=HL, 3=HH |
|                            | MINimum   | 0 (LL)                 |
|                            | MAXimum   | 3 (HH)                 |
| Example                    | SIM:NORM1:CODE 1  |                        |

`[:SOURce]:SIMulation:NORMal1:FREQuency` 



|                              |   |                            |
|------------------------------|---|----------------------------|
| Description                  | Sets or queries the frequency of the normal1 step of the simulation mode (Only Simulation Mode Active). |                            |
| Syntax                       | [:SOURce]:SIMulation:NORMal1:FREQuency {<NR2> MINimum MAXimum}  |                            |
| Query Syntax                 | [:SOURce]:SIMulation:NORMal1:FREQuency? [MINimum MAXimum]   |                            |
| Parameter/Return 1 parameter |   | Normal 1                   |
|                              | <NR2>   | Frequency of abnormal step |
|                              | MINimum   | Minimum frequency          |
|                              | MAXimum   | Maximum frequency          |
| Example                      | :SIM:NORM1:FREQ 60<br>Sets the frequency to 60Hz.   |                            |

`[:SOURce]:SIMulation:NORMal<1|2>:PHASe:STARt:ENABle` 



|                              |  |                      |
|------------------------------|--|----------------------|
| Description                  | Enables/Disables (Fixed/Free) the ON Phs parameter of the normal1 or normal2 step for the Simulation mode (Only Simulation Mode Active). |                      |
| Syntax                       | [:SOURce]:SIMulation:NORMal<1 2>:PHASe:STARt:ENABle { <bool> OFF ON FREE FIXED}  |                      |
| Query Syntax                 | [:SOURce]:SIMulation:NORMal<1 2>:PHASe:STARt:ENABle?   |                      |
| Parameter/Return 1 parameter | <1 2>  | Normal 1 or Normal 2 |
|                              | OFF   0   FREE   | Disabled             |
|                              | ON   1   FIXED   | Enabled              |
| Example                      | :SIM:NORM1:PHAS:STAR:ENAB 1<br>Enable the ON Phs.  |                      |

`[[:SOURce]:SIMulation:NORMal<1|2>:PHASe:STARt[:IMMEDIATE]]` 
 →  
 →

**Description** Sets or queries the ON Phs parameter of the normal1 or normal2 step for the Simulation mode (Only Simulation Mode Active).

**Syntax** `[[:SOURce]:SIMulation:NORMal<1|2>:PHASe:STARt[:IMMEDIATE]] {<NR2>|MINimum|MAXimum}`

**Query Syntax** `[[:SOURce]:SIMulation:NORMal<1|2>:PHASe:STARt[:IMMEDIATE]]? [MINimum|MAXimum]`

|                                   |                          |                      |
|-----------------------------------|--------------------------|----------------------|
| <b>Parameter/Return parameter</b> | <code>&lt;1 2&gt;</code> | Normal 1 or Normal 2 |
|                                   | <code>&lt;NR2&gt;</code> | ON Phs (start phase) |
|                                   | <code>MINimum</code>     | 0                    |
|                                   | <code>MAXimum</code>     | 359.9                |

**Example** `:SIM:NORM1:PHAS:STAR 0`  
Sets ON Phs to 0.

`[[:SOURce]:SIMulation:NORMal<1|2>:PHASe:STOP:ENABle]` 
 →  
 →

**Description** Enables/Disables (Fixed/Free) the OFF Phs parameter of the normal1 or normal2 step for the Simulation mode (Only Simulation Mode Active).

**Syntax** `[[:SOURce]:SIMulation:NORMal<1|2>:PHASe:STOP:ENABle] {<bool>|OFF|ON|FREE|FIXED}`


**Query Syntax** `[[:SOURce]:SIMulation:NORMal<1|2>:PHASe:STOP:ENABle]?`

|                                   |                             |                      |
|-----------------------------------|-----------------------------|----------------------|
| <b>Parameter/Return parameter</b> | <code>&lt;1 2&gt;</code>    | Normal 1 or Normal 2 |
|                                   | <code>OFF   0   FREE</code> | Disabled             |
|                                   | <code>ON   1   FIXED</code> | Enabled              |

**Example** `:SIM:NORM1:PHAS:STOP:ENAB 1`  
Enable the OFF Phs.

`[[:SOURce]:SIMulation:NORMal<1|2>:PHASe:STOP[:IMMEDIATE]]` 
 →  
 →

**Description** Sets or queries the OFF Phs parameter of the normal1 or normal2 step for the Simulation mode (Only Simulation Mode Active).

 **Note** Sets the off phase of the waveform after the output has been turned off.

**Syntax** `[[:SOURce]:SIMulation:NORMal<1|2>:PHASe:STOP[:IMMEDIATE]] {<NR2>|MINimum|MAXimum}`

**Query Syntax** `[[:SOURce]:SIMulation:NORMal<1|2>:PHASe:STOP[:IMMEDIATE]]? [MINimum|MAXimum]`

|                                   |                          |                      |
|-----------------------------------|--------------------------|----------------------|
| <b>Parameter/Return parameter</b> | <code>&lt;1 2&gt;</code> | Normal 1 or Normal 2 |
|                                   | <code>&lt;NR2&gt;</code> | OFF Phs (Stop phase) |
|                                   | <code>MINimum</code>     | 0                    |
|                                   | <code>MAXimum</code>     | 359.9                |

**Example** `:SIM:NORM1:PHAS:STOP 0`  
Sets OFF Phs to 0.

`[[:SOURce]:SIMulation:NORMal<1|2>:TIME]` 
 →  
 →

**Description** Sets or queries the Time parameter of the normal1 or normal2 step for the Simulation mode (Only Simulation Mode Active).

**Syntax** `[[:SOURce]:SIMulation:NORMal<1|2>:TIME {<NR2>|MINimum|MAXimum}`

**Query Syntax** `[[:SOURce]:SIMulation:NORMal<1|2>:TIME? [MINimum|MAXimum]`

|                                   |                          |                             |
|-----------------------------------|--------------------------|-----------------------------|
| <b>Parameter/Return parameter</b> | <code>&lt;1 2&gt;</code> | Normal 1 or Normal 2        |
|                                   | <code>&lt;NR2&gt;</code> | Time of the step in seconds |
|                                   | <code>MINimum</code>     | 0.0001                      |
|                                   | <code>MAXimum</code>     | 999.9999s                   |

Example                   :SIM:NORM1:TIME 1  
                               Sets the step time to 1 second.

**[[:SOURce]:SIMulation:NORMal1:VOLTage**                   (Set) →  
 → (Query)

Description           Sets or queries the Vset parameter of the normal1 step for the Simulation mode (Only Simulation Mode Active).

Syntax                [:SOURce]:SIMulation:NORMal1:VOLTage  
                           {<NR2>|MINimum|MAXimum}

Query Syntax        [:SOURce]:SIMulation:NORMal1:VOLTage?  
                           [MINimum|MAXimum]

|                            |         |                               |
|----------------------------|---------|-------------------------------|
| Parameter/Return parameter | 1       | Normal 1                      |
|                            | <NR2>   | Voltage of the abnormal step. |
|                            | MINimum | Minimum settable voltage      |
|                            | MAXimum | Maximum settable voltage      |

Example               :SIM:NORM1:VOLT MAX  
                           Sets the normal1 step voltage to the maximum.

**[[:SOURce]:SIMulation:REPeat:COUNT**                   (Set) →  
 → (Query)

Description           Sets or queries the repeat count for the Simulation mode (Only Simulation Mode Active).

Syntax                [:SOURce]:SIMulation:REPeat:COUNT  
                           {<NR1>|MINimum|MAXimum}

Query Syntax        [:SOURce]:SIMulation:REPeat:COUNT?  
                           [MINimum|MAXimum]

|                            |         |                              |
|----------------------------|---------|------------------------------|
| Parameter/Return parameter | <NR1>   | 0 ~ 9999 (0 = infinite loop) |
|                            | MINimum | 0                            |
|                            | MAXimum | 9999                         |

Example               :SIM:REP:COUN 1  
                           Sets the repeat count to 1.

Set →  
 → Query

**[[:SOURce]:SIMulation:REPeat:ENABle**

|                            |  |          |
|----------------------------|--|----------|
| Description                | Turns the repeat function on or off for the Simulation mode (Only Simulation Mode Active). |          |
| Syntax                     | [:SOURce]:SIMulation:REPeat:ENABle {<bool> OFF ON}   |          |
| Query Syntax               | [:SOURce]:SIMulation:REPeat:ENABle?  |          |
| Parameter/Return parameter | OFF   0  | Disabled |
|                            | ON   1   | Enabled  |
| Example                    | :SIM:REP:ENAB 1<br>Enables the repeat function.  |          |

Set →  
 → Query

**[[:SOURce]:SIMulation:TRANSition<1|2>:TIME**

|                            |  |                             |
|----------------------------|--|-----------------------------|
| Description                | Sets or queries the Time parameter of the transition step for the Simulation mode (Only Simulation Mode Active). |                             |
| Syntax                     | [:SOURce]:SIMulation:TRANSition<1 2>:TIME {<NR2> MINimum MAXimum}  |                             |
| Query Syntax               | [:SOURce]:SIMulation:TRANSition<1 2>:TIME? [MINimum MAXimum]   |                             |
| Parameter/Return parameter | <NR2>  | Time of the step in seconds |
|                            | MINimum  | 0                           |
|                            | MAXimum  | 999.9999s                   |
| Example                    | :SIM:TRAN1:TIME 1<br>Sets the step time to 1 second.   |                             |

**[[:SOURce]:SIMulation:TRANSition<1|2>:CODE** (Set) →  
→ (Query)

|                            |   |                        |
|----------------------------|---|------------------------|
| Description                | Sets the external trigger output for the transition step parameter. This option is only applicable when in the Simulation mode (Only Simulation Mode Active). |                        |
| Syntax                     | [:SOURce]:SIMulation:TRANSition<1 2>:CODE {<NR1> MINimum MAXimum}   |                        |
| Query Syntax               | [:SOURce]:SIMulation:TRANSition<1 2>:CODE? [MINimum MAXimum]  |                        |
| Parameter/Return parameter | <NR1>   | 0=LL, 1=LH, 2=HL, 3=HH |
|                            | MINimum   | 0 (LL)                 |
|                            | MAXimum   | 3 (HH)                 |
| Example                    | SIM:TRAN1:CODE 1  |                        |

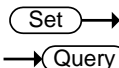
**:TRIGger:SIMulation:SElected:EXECute** (Set) →

|             |  |                           |
|-------------|--|---------------------------|
| Description | Sets to execute actions for simulate mode (Only Simulation Mode Active). |                           |
| Syntax      | :TRIGger:SIMulation:SElected:EXECute {STOP START HOLD}                   |                           |
| Parameter   | STOP   | Stops simulate execution  |
|             | START  | Starts simulate execution |
|             | HOLD   | Holds simulate execution  |
| Example     | TRIG:SIM:SEL:EXEC STAR<br>Starts simulate execution.                     |                           |

## Input Subsystem Command

|                          |     |
|--------------------------|-----|
| :INPut:GAIN .....        | 136 |
| :INPut:SYNC:SOURce ..... | 136 |

### :INPut:GAIN



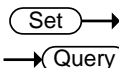
**Description** Sets or queries the input gain value. (Only AC+DC-EXT or AC-EXT or AC+DC-ADD or AC-ADD or AC-VCA Active)

**Syntax** :INPut:GAIN {<NR2>(V)|MINimum|MAXimum}

**Query Syntax** :INPut:GAIN? [MINimum|MAXimum]

|                                   |         |                          |
|-----------------------------------|---------|--------------------------|
| <b>Parameter/Return parameter</b> | <NR2>   | Input gain value         |
|                                   | MINimum | Minimum input gain value |
|                                   | MAXimum | Maximum input gain value |

**Example** :INP:GAIN?  
+150.0000  
Returns the input gain value as 150.0.



### :INPut:SYNC:SOURce

**Description** Sets or queries state of sync source. (Only AC+DC-sync or AC-sync Active)

**Syntax** :INPut:SYNC:SOURce {<NR1>|LINE|EXT}

**Query Syntax** :INPut:SYNC:SOURce?

|                                   |          |                   |
|-----------------------------------|----------|-------------------|
| <b>Parameter/Return parameter</b> | <NR1>    | LINE (0)  EXT (1) |
|                                   | LINE   0 | LINE sync source  |
|                                   | EXT   1  | EXT sync source   |



Example           :INP:SYNC:SOUR?  
                  EXT  
                  Returns the state of sync source as EXT.

## Display Command

|   |     |
|---|-----|
| :DISPlay[:WINDow]:DESIgn:MODE.....                    | 138 |
| :DISPlay[:WINDow]:MEASure:<br>SOURce<1   2   3> ..... | 138 |

### :DISPlay[:WINDow]:DESIgn:MODE (Set) →

|             |  |                                  |
|-------------|--|----------------------------------|
| Description | Sets two display mode.                               |                                  |
| Syntax      | :DISPlay[:WINDow]:DESIgn:MODE{NORMAl SIMPlE}         |                                  |
| Parameter   | NORMAl   | Configure setup and Measurement. |
|             | SIMPlE   | All measurement times.           |
| Example     | :DISP:DES:MODE NORM<br>Sets standard normal display. |                                  |

### :DISPlay[:WINDow]:MEASure:SOURce<1|2|3> (Set) →

|             |  |  |
|-------------|--|--|
| Description | Sets standard normal display to measurement items 1 – 3.   |  |
| Syntax      | :DISPlay[:WINDow]:MEASure:SOURce<1 3><br>{ VRMS VAVG VMAX VMIN IRMS I AVG I MAX IMIN I PKH RPOWER SPOWER QPOWER FREQUENCY PFACTOR CFACtor THDV THDI} |  |
| Parameter   | Item 1   | VRMS , VAVG , VMAX , VMIN , RPOWER , SPOWER*1, QPOWER*1, THDV*2  |
|             | Item 2   | IRMS , IAVG , IMAX , IMIN , IPKH , PFACTOR*1, CFACtor*1, THDI*2  |
|             | Item 3   | RPOWER , SPOWER*1, QPOWER*1, IPKH , PFACTOR*1, CFACtor*1, FREQUENCY*3  |
|             | Note   | *1: Not available for DC-INT<br>*2: Available for AC-INT only<br>*3: Available for AC+DC-Sync & AC-Sync only |

Example           :DISP:MEAS:SOUR1 VRMS  
                  Sets measurement source 1 VRMS display.

## Status Register Overview

To program the ASR power supply effectively, the Status registers need to be understood. This chapter explains in detail how the Status registers are used and how to configure them.

### Introduction to the Status Registers

---

#### Overview

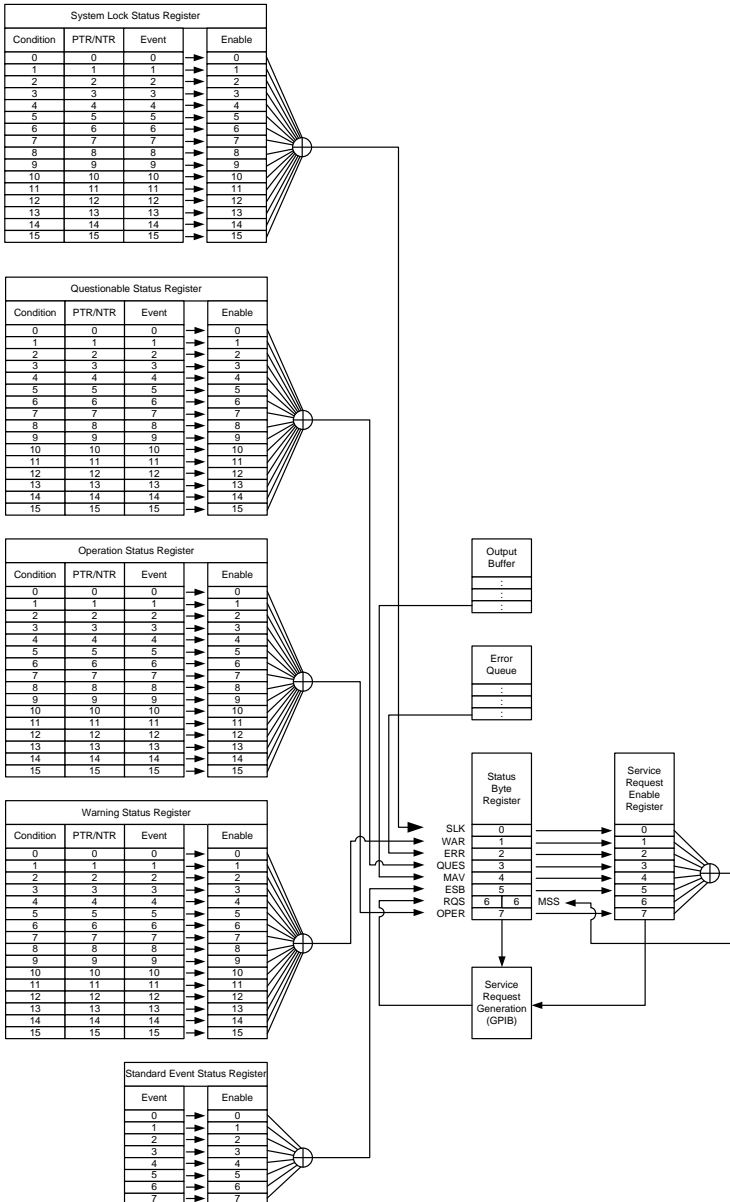
The status registers are used to determine the status of the power supply. The status registers maintain the status of the protection conditions, operation conditions and instrument errors.

The ASR Series have a number of register groups:

- Questionable Status Register Group
- Standard Event Status Register Group
- Operation Status Register Group
- Warning Status Register Group
- System Lock Status Register Group
- Status Byte Register
- Service Request Enable Register
- Service Request Generation
- Error Queue
- Output Buffer

The diagram below shows the structure of the Status registers.

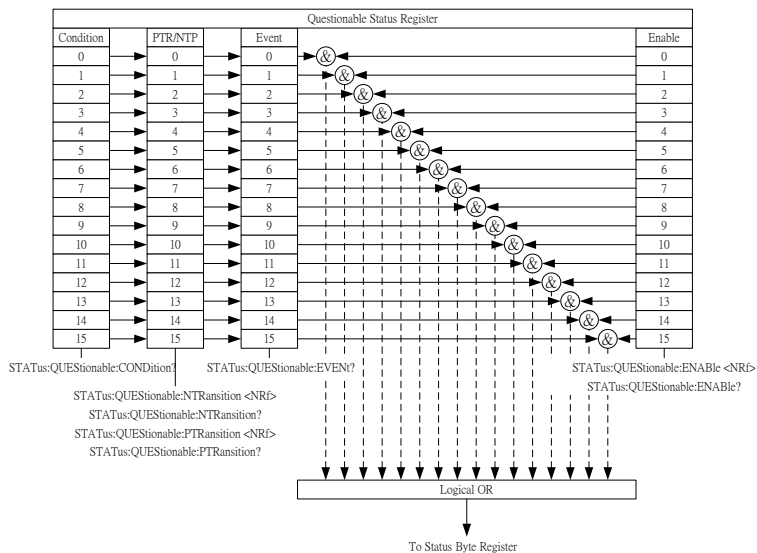
# The Status Registers



## Questionable Status Register Group

### Overview

The Questionable Status Register Group indicates if any protection modes or limits have been tripped.



| Bit Summary | Event   | Bit # | Bit Weight |
|-------------|---|-------|------------|
|             | Output Overvoltage                                      | 0     | 1          |
|             | Over internal maximum voltage (110% of rating voltage). |       |            |
|             | Over Irms Current                                       | 1     | 2          |
|             | Output current RMS value is excessive                   |       |            |
|             | Overheat  | 4     | 16         |
|             | Internal power stage over heat.                         |       |            |

|   |    |       |
|---|----|-------|
| Output Short  | 5  | 32    |
| Call attention to output terminal short status.                 |    |       |
| Over Ipeak+ Current or Over Ipeak- Current                      | 6  | 64    |
| Positive/Negative output current peak value is excessive.       |    |       |
| Fan Failure   | 7  | 128   |
| Fan failure. Contact service center.                            |    |       |
| Calibration Data Error  | 8  | 256   |
| The calibration data is abnormal or out of allowance range.     |    |       |
| Output Over-Power   | 9  | 512   |
| Over internal power stage maximum power (110% of rating power). |    |       |
| IPK Limit   | 10 | 1024  |
| The peak current limiter is activated.                          |    |       |
| Remote Sensing Voltage Out of Range                             | 11 | 2048  |
| The Sensing voltage limiter is activated.                       |    |       |
| IRMS Limit  | 12 | 4096  |
| The RMS current limiter is activated.                           |    |       |
| Always 0  | 15 | 32768 |

Condition Register      The Questionable Status Condition Register indicates the status of the power supply. If a bit is set in the Condition register, it indicates that the event is true. Reading the condition register does not change the state of the condition register.

PTR/NTR Filters      The PTR/NTR (Positive/Negative transition) register determines the type of transition conditions that will set the corresponding bit in the Event Registers. Use the Positive transition filter to view events that change from false to positive, and use the negative transition filter to view events that change from positive to negative.

Positive Transition      0→1

Negative Transition      1→0

Event Register      The PTR/NTR Register will dictate the type of transition conditions will set the corresponding bits in the Event Register. If the Event Register is read, it will be cleared to 0.

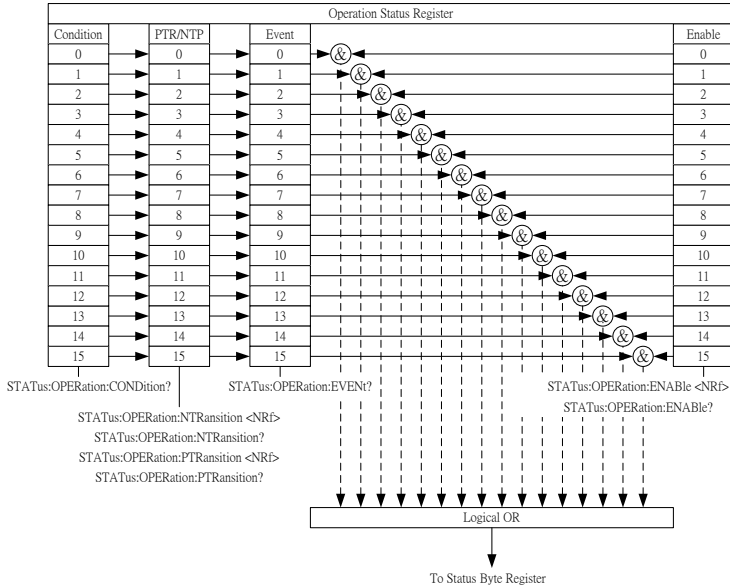
Enable Register      The Enable register determines which Events in the Event Register will be used to set the QUES bit in the Status Byte Register.



## Operation Status Register Group

### Overview

The Operation Status Register Group indicates the operating status of the power supply.



| Bit Summary | Event                     | Bit # | Bit Weight |
|-------------|---------------------------|-------|------------|
|             | Busy Status               | 1     | 2          |
|             | LOCK status (SYNC) status | 8     | 256        |
|             | Hold Status (Sequence)    | 12    | 4096       |
|             | Run Status (Sequence)     | 14    | 16384      |
|             | Always 0                  | 15    | 32768      |

### Condition Register

The Operation Status Condition Register indicates the operating status of the power supply. If a bit is set in the Condition register, it indicates that the event is true. Reading the condition register does not change the state of the condition register.

**PTR/NTR Filters**    The PTR/NTR (Positive/Negative transition) register determines the type of transition conditions that will set the corresponding bit in the Event Registers. Use the Positive transition filter to view events that change from false to positive, and use the negative transition filter to view events that change from positive to negative.

---

Positive Transition                      0→1

Negative Transition                      1→0

---

**Event Register**    The PTR/NTR Register will dictate the type of transition conditions will set the corresponding bits in the Event Register. If the Event Register is read, it will be cleared to 0.

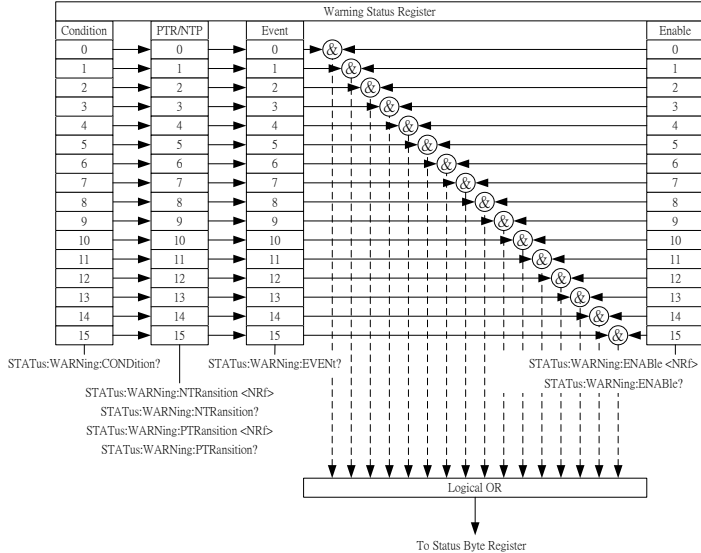
---

**Enable Register**    The Enable register determines which registered Events in the Event Register will be used to set the OPER bit in the Status Byte Register.

## Warning Status Register Group

### Overview

The Warning Status Register Group is a secondary protection status register for the supply output.



| Bit Summary | Event   | Bit # | Bit Weight |
|-------------|---|-------|------------|
|             | Output Overvoltage  | 0     | 1          |
|             | Over internal maximum voltage (110% of rating voltage).   |       |            |
|             | Over Irms Current   | 1     | 2          |
|             | Output current RMS value is excessive                     |       |            |
|             | Over Ipeak+ Current or Over Ipeak- Current                | 3     | 8          |
|             | Positive/Negative output current peak value is excessive. |       |            |
|             | Overheat  | 6     | 64         |
|             | Internal power stage over heat.                           |       |            |

|   |    |       |
|---|----|-------|
| External Sync Frequency Error   | 7  | 128   |
| The external synchronization signal input frequency is out of the allowance range. (40Hz ~ 999.9Hz) |    |       |
| Sensing Voltage Error   | 9  | 512   |
| Remote sense connection wire is abnormal or over maximum compensation voltage.                      |    |       |
| Over Irms Current   | 10 | 1024  |
| Output current RMS value is excessive   |    |       |
| Over Ipeak+ Current or Over Ipeak- Current  | 11 | 2048  |
| Positive/Negative output current peak value is excessive.   |    |       |
| Output Over-Power   | 12 | 4096  |
| Over internal power stage maximum power (110% of rating power)                                      |    |       |
| IRMS Limit  | 13 | 8192  |
| The RMS current limiter is activated.   |    |       |
| IPK Limit   | 14 | 16384 |
| The peak current limiter is activated.  |    |       |
| Always 0  | 15 | 32768 |

Condition Register      The System Lock Status Condition Register indicates the system lock status of the power supply. If a bit is set in the Condition register, it indicates that the event is true. Reading the condition register does not change the state of the condition register.

---

**PTR/NTR Filters**    The PTR/NTR (Positive/Negative transition) register determines the type of transition conditions that will set the corresponding bit in the Event Registers. Use the Positive transition filter to view events that change from false to positive, and use the negative transition filter to view events that change from positive to negative.

---

|                     |     |
|---------------------|-----|
| Positive Transition | 0→1 |
|---------------------|-----|

|                     |     |
|---------------------|-----|
| Negative Transition | 1→0 |
|---------------------|-----|

---

**Event Register**    The PTR/NTR Register will dictate the type of transition conditions will set the corresponding bits in the Event Register. If the Event Register is read, it will be cleared to 0.

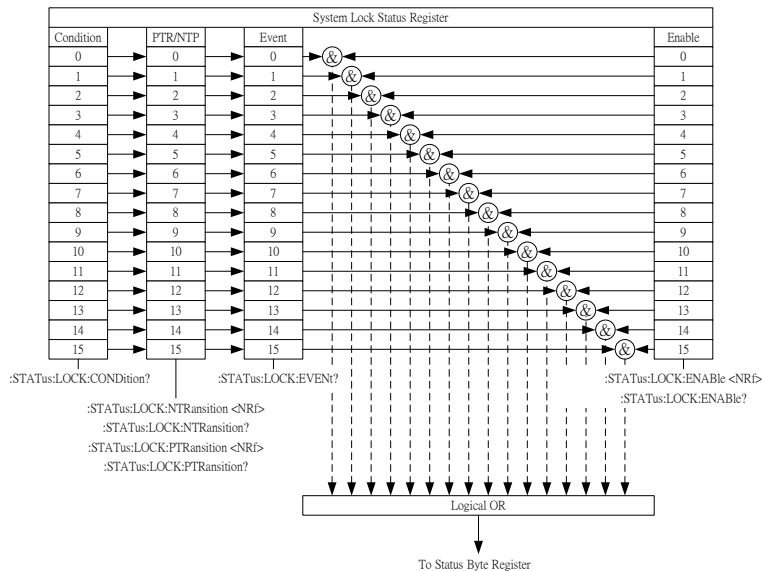
---

**Enable Register**    The Enable register determines which registered Events in the Event Register will be used to set the WAR bit in the Status Byte Register.

## System Lock Status Register Group

### Overview

The System Lock Status Register Group indicates if system lock protection modes have been tripped.



### Bit Summary

| Event  | Bit # | Bit Weight |
|--|-------|------------|
| Power Input Anomaly  | 0     | 1          |
| The power input voltage is insufficient or turning off main power switch. Check input power before rebooting the unit. |       |            |
| Fan Failure  | 7     | 128        |
| Fan failure. Contact service center.   |       |            |
| Startup Anomaly  | 8     | 256        |
| Abnormal startup procedure.  |       |            |

---

|                     |  |                     |     |                     |     |
|---------------------|--|---------------------|-----|---------------------|-----|
| Condition Register  | The System Lock Status Condition Register indicates the system lock status of the power supply. If a bit is set in the Condition register, it indicates that the event is true. Reading the condition register does not change the state of the condition register.  |                     |     |                     |     |
| PTR/NTR Filters     | The PTR/NTR (Positive/Negative transition) register determines the type of transition conditions that will set the corresponding bit in the Event Registers. Use the Positive transition filter to view events that change from false to positive, and use the negative transition filter to view events that change from positive to negative.<br><table><tr><td>Positive Transition</td><td>0→1</td></tr><tr><td>Negative Transition</td><td>1→0</td></tr></table> | Positive Transition | 0→1 | Negative Transition | 1→0 |
| Positive Transition | 0→1  |                     |     |                     |     |
| Negative Transition | 1→0  |                     |     |                     |     |
| Event Register      | The PTR/NTR Register will dictate the type of transition conditions will set the corresponding bits in the Event Register. If the Event Register is read, it will be cleared to 0.   |                     |     |                     |     |
| Enable Register     | The Enable register determines which registered Events in the Event Register will be used to set the SLK bit in the Status Byte Register.  |                     |     |                     |     |

---



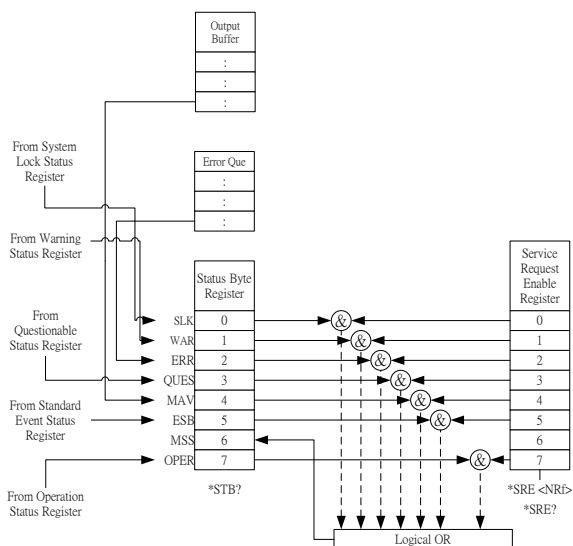


|                 |  |   |     |
|-----------------|--|---|-----|
|                 | DDE (Device Dependent Error)   | 3 | 8   |
|                 | Device specific error.   |   |     |
|                 | EXE (Execution Error)  | 4 | 16  |
|                 | The EXE bit indicates an execution error due to one of the following: illegal command parameter, parameter out of range, invalid parameter, the command didn't execute due to an overriding operation condition. |   |     |
|                 | CME (Command Error)  | 5 | 32  |
|                 | The CME bit is set when a syntax error has occurred. The CME bit can also be set when a <GET> command is received within a program message.  |   |     |
|                 | URQ (User Request)   | 6 | 64  |
|                 | PON (Power On)   | 7 | 128 |
|                 | Indicates the power is turned on.  |   |     |
| Event Register  | Any bits set in the event register indicate that an error has occurred. Reading the Event register will reset the register to 0.   |   |     |
| Enable Register | The Enable register determines which Events in the Event Register will be used to set the ESB bit in the Status Byte Register.   |   |     |

## Status Byte Register & Service Request Enable Register

### Overview

The Status Byte register consolidates the status events of all the status registers. The Status Byte register can be read with the `*STB?` query and can be cleared with the `*CLS` command.



### Bit Summary

| Event                                       | Bit # | Bit Weight |
|---|-------|------------|
| SLK(System Lock Status Register Summary)    | 0     | 1          |
| WAR (Warning Status Register)               | 1     | 2          |
| ERR (Error Queue not empty)                 | 2     | 4          |
| QUES (Questionable Status Register)         | 3     | 8          |
| MAV (Message Available)                     | 4     | 16         |
| ESB(Standard Event Status Register Summary) | 5     | 32         |

---

|  |  |   |     |
|--|--|---|-----|
|  | RQS / MSS(Request Service / Master Summary Status) | 6 | 64  |
|  | OPER (Operation Status Register)                   | 7 | 128 |

---

|                                 |   |
|---------------------------------|---|
| Status Byte Register            | Any bits set in the Status byte register acts as a summary register for all the three other status registers and indicates if there is a service request, an error in the Error Queue or data in the Output Queue. Reading the Status Byte register will reset the register to 0. |
| Service Request Enable Register | The Service Request Enable Register controls which bits in the Status Byte Register are able to generate service requests.  |

---

## Error List

### Command Errors

---

**Overview** An <error/event number> in the range [ -199 , -100 ] indicates that an IEEE 488.2 syntax error has been detected by the instrument's parser. The occurrence of any error in this class shall cause the command error bit (bit 5) in the event status register (IEEE 488.2, section 11.5.1) to be set. One of the following events has occurred:

- An IEEE 488.2 syntax error has been detected by the parser. That is, a controller-to-device message was received which is in violation of the IEEE 488.2 standard. Possible violations include a data element which violates the device listening formats or whose type is unacceptable to the device.
- An unrecognized header was received. Unrecognized headers include incorrect device-specific headers and incorrect or unimplemented IEEE 488.2 common commands.

Events that generate command errors shall not generate execution errors, device-specific errors, or query errors; see the other error definitions in this chapter.

| Error Code         | Description   |
|--------------------|---|
| -100 Command Error | This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that a Command Error as defined in IEEE 488.2,11.5.1.1.4 has occurred. |

- 102 Syntax error An unrecognized command or data type was encountered; for example, a string was received when the device does not accept strings.
- 103 Invalid separator The parser was expecting a separator and encountered an illegal character; for example, the semicolon was omitted after a program message unit, MEAS:VOLT:DC?:MEASCURR:DC?
- 104 Data type error The parser recognized a data element different than one allowed; for example, numeric or string data was expected but block data was encountered.
- 108 Parameter not allowed More parameters were received than expected for the header; for example, the :SYSTem:KLOCK command only accepts one parameter, so receiving SYSTem:KLOCK 1,0 is not allowed.
- 109 Missing parameter Fewer parameters were received than required for the header; for example the :SYSTem:KLOCK command requires one parameter, so receiving :SYSTem:KLOCK is not allowed.
- 111 Header separator error A character which is not a legal header separator was encountered while parsing the header; for example, no white space followed the header, thus \*SRE2 is an error.
- 112 Program mnemonic too long The header contains more than twelve characters (see IEEE 488.2, 7.6.1.4.1).
- 113 Undefined header The header is syntactically correct, but it is undefined for this specific device; for example, \*XYZ is not defined for any device.

- 114 Header suffix out of range      The value of a numeric suffix attached to a program mnemonic, see Syntax and Style section 6.2.5.2, makes the header invalid.
- 115 Unexpected number of parameters      The number of parameters received does not correspond to the number of parameters expected. This is typically due an inconsistency with the number of instruments in the selected group.
- 120 Numeric data error      This error, as well as errors -121 through -129, are generated when parsing a data element which appears to be numeric, including the non-decimal numeric types. This particular error message should be used if the device cannot detect a more specific error.
- 121 Invalid character in number      An invalid character for the data type being parsed was encountered; for example, an alpha in a decimal numeric or a "9" in octal data.
- 128 Numeric data not allowed      A legal numeric data element was received, but the device does not accept one in this position for the header.
- 131 Invalid suffix      The suffix does not follow the syntax described in IEEE 488.2, 7.7.3.2, or the suffix is inappropriate for this device.
- 141 Invalid character data      Either the character data element contains an invalid character or the particular element received is not valid for the header.
- 148 Character data not allowed      A legal character data element was encountered where prohibited by the device.

---

|                                  |   |
|----------------------------------|---|
| -151 Invalid string data         | A string data element was expected, but was invalid for some reason (see IEEE 488.2, 7.7.5.2); for example, an END message was received before the terminal quote character.                      |
| -158 String data not allowed     | A string data element was encountered but was not allowed by the device at this point in parsing.   |
| -160 Block data error            | This error, as well as errors -161 through -169, are generated when parsing a block data element. This particular error message should be used if the device cannot detect a more specific error. |
| -161 Invalid block data          | A block data element was expected, but was invalid for some reason (see IEEE 488.2, 7.7.6.2); for example, an END message was received before the length was satisfied.                           |
| -168 Block data not allowed      | A legal block data element was encountered but was not allowed by the device at this point in parsing.  |
| -178 Expression data not allowed | A legal expression data was encountered but was not allowed by the device at this point in parsing.   |

## Execution Errors

---

|          |   |
|----------|---|
| Overview | <p>An &lt;error/event number&gt; in the range [ -299 , -200 ] indicates that an error has been detected by the instrument's execution control block. The occurrence of any error in this class shall cause the execution error bit (bit 4) in the event status register (IEEE 488.2, section 11.5.1) to be set. One of the following events has occurred:</p> <ul style="list-style-type: none"><li>• A &lt;PROGRAM DATA&gt; element following a header was evaluated by the device as outside of its legal input range or is otherwise</li></ul> |
|----------|---|

---

inconsistent with the device's capabilities.

- A valid program message could not be properly executed due to some device condition.

Execution errors shall be reported by the device after rounding and expression evaluation operations have taken place. Rounding a numeric data element, for example, shall not be reported as an execution error. Events that generate execution errors shall not generate Command Errors, device-specific errors, or Query Errors; see the other error definitions in this section.

| Error Code                  | Description   |
|-----------------------------|---|
| -200 Execution error        | This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that an Execution Error as defined in IEEE 488.2, 11.5.1.1.5 has occurred.   |
| -201 Invalid while in local | Indicates that a command is not executable while the device is in local due to a hard local control (see IEEE 488.2, 5.6.1.5); for example, a device with a rotary switch receives a message which would change the switches state, but the device is in local so the message cannot be executed. |
| -203 Command protected      | Indicates that a legal password-protected program command or query could not be executed because the command was disabled.  |
| -211 Trigger ignored        | Indicates that a GET, *TRG, or triggering signal was received and recognized by the device but was ignored because of device timing considerations; for example, the device was not ready to respond. Note: a DT0 device always ignores GET and treats *TRG as a Command Error.                   |



- 213 Init ignored Indicates that a request for a measurement initiation was ignored as another measurement was already in progress.
- 220 Parameter error Indicates that a program data element related error occurred. This error message should be used when the device cannot detect the more specific errors described for errors -221 through -229.
- 221 Settings conflict Indicates that a legal program data element was parsed but could not be executed due to the current device state (see IEEE 488.2, 6.4.5.3 and 11.5.1.1.5.).
- 222 Data out of range Indicates that a legal program data element was parsed but could not be executed because the interpreted value was outside the legal range as defined by the device (see IEEE 488.2, 11.5.1.1.5.).
- 224 Illegal parameter value Used where exact value, from a list of possible, was expected.

## Device Specific Errors

---

- Overview An <error/event number> in the range [ -399 , -300 ] or [ 1 , 32767 ] indicates that the instrument has detected an error which is not a command error, a query error, or an execution error; some device operations did not properly complete, possibly due to an abnormal hardware or firmware condition. These codes are also used for self-test response errors. The occurrence of any error in this class should cause the device-specific error bit (bit 3) in the event status register (IEEE 488.2, section 11.5.1) to be set. The meaning of positive error codes is device-dependent and may be enumerated or bit mapped; the <error
-

message>string for positive error codes is not defined by SCPI and available to the device designer.

Note that the string is not optional; if the designer does not wish to implement a string for a particular error, the null string should be sent (for example, 42,""). The occurrence of any error in this class should cause the device-specific error bit (bit 3) in the event status register (IEEE 488.2, section 11.5.1) to be set. Events that generate device-specific errors shall not generate command errors, execution errors, or query errors; see the other error definitions in this section.

| Error Code         | Description  |
|--------------------|--|
| -310 System error  | Indicates that some error, termed "system error" by the device, has occurred. This code is device-dependent.   |
| -320 Storage fault | Indicates that the firmware detected a fault when using data storage. This error is not an indication of physical damage or failure of any mass storage element. |

## Query Errors

**Overview** An <error/event number> in the range [ -499 , -400 ] indicates that the output queue control of the instrument has detected a problem with the message exchange protocol described in IEEE 488.2, chapter 6. The occurrence of any error in this class shall cause the query error bit (bit 2) in the event status register (IEEE 488.2, section 11.5.1) to be set. These errors correspond to message exchange protocol errors described in IEEE 488.2, section 6.5. One of the following is true:

- An attempt is being made to read data from the output queue when no output is either present or pending;
- Data in the output queue has been lost.

Events that generate query errors shall not generate command errors, execution errors, or device-specific errors; see the other error definitions in this section.

| Error Code       | Description   |
|------------------|---|
| -400 Query error | This is the generic query error for devices that cannot detect more specific errors. This code indicates only that a Query Error as defined in IEEE 488.2, 11.5.1.1.7 and 6.3 has occurred. |

# A PPENDIX

## Factory Default Settings

The following default settings are the factory configuration settings for the ASR-2000 series. For details on how to return to the factory default settings, please see the user manual.

| AC+DC-INT Mode | ASR-2050      | ASR-2050R | ASR-2100    | ASR-2100R |
|----------------|---------------|-----------|-------------|-----------|
| Range          | 100V          |           |             |           |
| Wave Shape     | SIN           |           |             |           |
| ACV            | 0.0 Vrms      |           |             |           |
| DCV            | +0.0 Vdc      |           |             |           |
| FREQ           | 50.00 Hz      |           |             |           |
| IRMS           | 5.25 A        |           | 10.50 A     |           |
| V Limit        | +/- 250.0 Vdc |           |             |           |
| F Limit Lo     | 1.0 Hz        |           |             |           |
| F Limit Hi     | 999.9 Hz      |           |             |           |
| IPK Limit      | +/- 21.00 A   |           | +/- 42.00 A |           |
| ON Phs         | 0.0°          |           |             |           |
| OFF Phs        | 0.0°          |           |             |           |

| AC-INT Mode | ASR-2050    | ASR-2050R | ASR-2100    | ASR-2100R |
|-------------|-------------|-----------|-------------|-----------|
| Range       | 100V        |           |             |           |
| Wave Shape  | SIN         |           |             |           |
| ACV         | 0.0 Vrms    |           |             |           |
| FREQ        | 50.00 Hz    |           |             |           |
| IRMS        | 5.25 A      |           | 10.50 A     |           |
| V Limit     | 175.0 Vrms  |           |             |           |
| F Limit Lo  | 40.0 Hz     |           |             |           |
| F Limit Hi  | 999.9 Hz    |           |             |           |
| IPK Limit   | +/- 21.00 A |           | +/- 42.00 A |           |
| ON Phs      | 0.0°        |           |             |           |
| OFF Phs     | 0.0°        |           |             |           |

| DC-INT Mode | ASR-2050      | ASR-2050R | ASR-2100    | ASR-2100R |
|-------------|---------------|-----------|-------------|-----------|
| Range       | 100V          |           |             |           |
| DCV         | 0.0 Vdc       |           |             |           |
| I           | 5.25 A        |           | 10.50 A     |           |
| V Limit     | +/- 250.0 Vdc |           |             |           |
| IPK Limit   | +/- 21.00 A   |           | +/- 42.00 A |           |

| AC+DC-EXT Mode | ASR-2050    | ASR-2050R | ASR-2100    | ASR-2100R |
|----------------|-------------|-----------|-------------|-----------|
| Range          | 100V        |           |             |           |
| GAIN           | 100.0       |           |             |           |
| IRMS           | 5.25 A      |           | 10.50 A     |           |
| IPK Limit      | +/- 21.00 A |           | +/- 42.00 A |           |

| AC-EXT Mode | ASR-2050    | ASR-2050R | ASR-2100    | ASR-2100R |
|-------------|-------------|-----------|-------------|-----------|
| Range       | 100V        |           |             |           |
| GAIN        | 100.0       |           |             |           |
| IRMS        | 5.25 A      |           | 10.50 A     |           |
| IPK Limit   | +/- 21.00 A |           | +/- 42.00 A |           |

| AC+DC-ADD Mode | ASR-2050      | ASR-2050R | ASR-2100    | ASR-2100R |
|----------------|---------------|-----------|-------------|-----------|
| Range          | 100V          |           |             |           |
| Wave Shape     | SIN           |           |             |           |
| ACV            | 0.0 Vrms      |           |             |           |
| DCV            | +0.0 Vdc      |           |             |           |
| GAIN           | 100.0         |           |             |           |
| FREQ           | 50.00 Hz      |           |             |           |
| IRMS           | 5.25 A        |           | 10.50 A     |           |
| V Limit        | +/- 250.0 Vdc |           |             |           |
| F Limit Lo     | 1.0 Hz        |           |             |           |
| F Limit Hi     | 999.9 Hz      |           |             |           |
| IPK Limit      | +/- 21.00 A   |           | +/- 42.00 A |           |
| ON Phs         | 0.0°          |           |             |           |
| OFF Phs        | 0.0°          |           |             |           |

| AC-ADD Mode | ASR-2050 | ASR-2050R | ASR-2100 | ASR-2100R |
|-------------|----------|-----------|----------|-----------|
| Range       | 100V     |           |          |           |
| Wave Shape  | SIN      |           |          |           |
| ACV         | 0.0 Vrms |           |          |           |
| GAIN        | 100.0    |           |          |           |

|            |             |             |
|------------|-------------|-------------|
| FREQ       | 50.00 Hz    |             |
| IRMS       | 5.25 A      | 10.50 A     |
| V Limit    | 175.0 Vrms  |             |
| F Limit Lo | 40.0 Hz     |             |
| F Limit Hi | 999.9 Hz    |             |
| IPK Limit  | +/- 21.00 A | +/- 42.00 A |
| ON Phs     | 0.0°        |             |
| OFF Phs    | 0.0°        |             |

| AC+DC-SYNC Mode | ASR-2050      | ASR-2050R | ASR-2100    | ASR-2100R |
|-----------------|---------------|-----------|-------------|-----------|
| Range           | 100V          |           |             |           |
| Wave Shape      | SIN           |           |             |           |
| ACV             | 0.0 Vrms      |           |             |           |
| DCV             | +0.0 Vdc      |           |             |           |
| SIG             | LINE          |           |             |           |
| IRMS            | 5.25 A        |           | 10.50 A     |           |
| V Limit         | +/- 250.0 Vdc |           |             |           |
| F Limit Lo      | 1.0 Hz        |           |             |           |
| F Limit Hi      | 999.9 Hz      |           |             |           |
| IPK Limit       | +/- 21.00 A   |           | +/- 42.00 A |           |
| ON Phs          | 0.0°          |           |             |           |
| OFF Phs         | 0.0°          |           |             |           |
| Syn Phs         | 0.0°          |           |             |           |

| AC-SYNC Mode | ASR-2050 | ASR-2050R   | ASR-2100   | ASR-2100R   |
|--------------|----------|-------------|------------|-------------|
| Range        |          |             | 100V       |             |
| Wave Shape   |          |             | SIN        |             |
| ACV          |          |             | 0.0 Vrms   |             |
| SIG          |          |             | LINE       |             |
| IRMS         |          | 5.25 A      |            | 10.50 A     |
| V Limit      |          |             | 175.0 Vrms |             |
| F Limit Lo   |          |             | 1.0 Hz     |             |
| F Limit Lo   |          |             | 999.9 Hz   |             |
| IPK Limit    |          | +/- 21.00 A |            | +/- 42.00 A |
| ON Phs       |          |             | 0.0°       |             |
| OFF Phs      |          |             | 0.0°       |             |
| Syn Phs      |          |             | 0.0°       |             |

| AC-VCA Mode | ASR-2050 | ASR-2050R   | ASR-2100   | ASR-2100R   |
|-------------|----------|-------------|------------|-------------|
| Range       |          |             | 100V       |             |
| Wave Shape  |          |             | SIN        |             |
| GAIN        |          |             | 100.0      |             |
| IRMS        |          | 5.25 A      |            | 10.50 A     |
| V Limit     |          |             | 175.0 Vrms |             |
| F Limit Lo  |          |             | 1.0 Hz     |             |
| F Limit Hi  |          |             | 999.9 Hz   |             |
| IPK Limit   |          | +/- 21.00 A |            | +/- 42.00 A |
| ON Phs      |          |             | 0.0°       |             |
| OFF Phs     |          |             | 0.0°       |             |

| Menu                | ASR-2000 |
|---------------------|----------|
| T ipeak, hold(msec) | 1 ms     |
| IPK CLR             | EXEC     |
| Power ON            | OFF      |
| Buzzer              | ON       |
| Remote Sense        | OFF      |
| Slew Rate Mode      | Slope    |
| Output Relay        | Enable   |
| THD Format          | IEC      |
| External Control    | OFF      |
| V Unit (TRI, ARB)   | rms      |
| ACin Detection      | ON       |
| TrgOut Width (ms)   | 0.1      |
| Data Average Count  | 1        |
| Data Update Rate    | Fast     |
| TrgOut Source       | None     |

| LAN  | ASR-2000 |
|------|----------|
| DHCP | ON       |

| USB Device | ASR-2000 |
|------------|----------|
| Speed      | Full     |

| Sequence Mode | ASR-2000 |
|---------------|----------|
| Step          | 0        |
| Time          | 0.1000 s |
| ACV           | 0.0, CT  |

---

|           |          |
|-----------|----------|
| DCV       | 0.0, CT  |
| Fset      | 50.0, CT |
| Wave      | SIN      |
| Jump To   | OFF      |
| Jump Cnt  | 1        |
| Branch 1  | OFF      |
| Branch 2  | OFF      |
| Term      | CONTI    |
| Sync Code | LL       |
| ON Phs    | Free     |
| OFF Phs   | Free     |

---

|                        |                 |
|------------------------|-----------------|
| <b>Simulation Mode</b> | <b>ASR-2000</b> |
|------------------------|-----------------|

|         |          |
|---------|----------|
| Step    | Initial  |
| Repeat  | OFF      |
| Time    | 0.1000 s |
| ACV     | 0.0      |
| Fset    | 50.00    |
| ON Phs  | Free     |
| OFF Phs | Free     |
| Wave    | SIN      |
| Code    | LL       |

---

|               |                 |
|---------------|-----------------|
| <b>RS232C</b> | <b>ASR-2000</b> |
|---------------|-----------------|

|          |       |
|----------|-------|
| Baudrate | 9600  |
| Databits | 8bits |
| Parity   | None  |
| Stopbits | 1bit  |

---

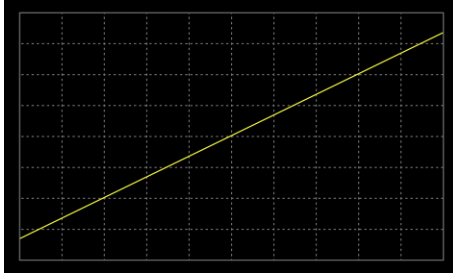
|             |                           |
|-------------|---------------------------|
| <b>GPIB</b> | <b>ASR-GPIB-2K Option</b> |
|-------------|---------------------------|

|         |    |
|---------|----|
| Address | 10 |
|---------|----|

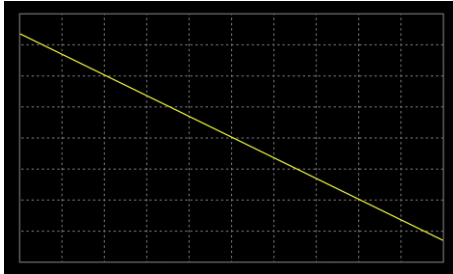


## Default Waveform Setting

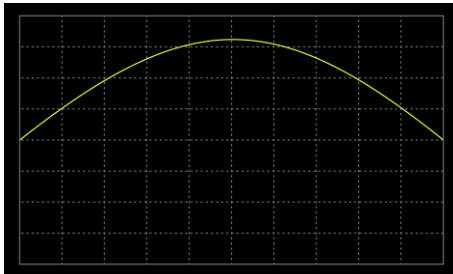
ARB 1 Ramp (rising)



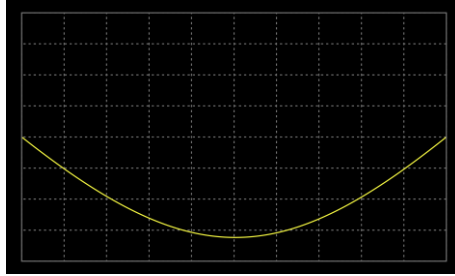
ARB 2 Ramp (falling)



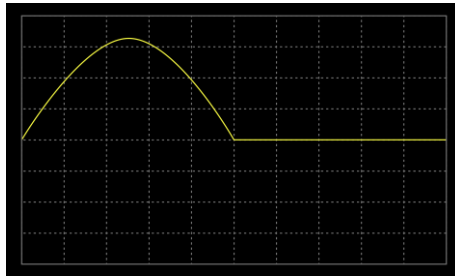
ARB 3 Sine wave, half-cycle(positive pole)



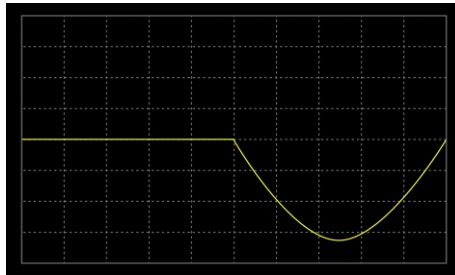
ARB 4 Sine wave, half-cycle(negative pole)



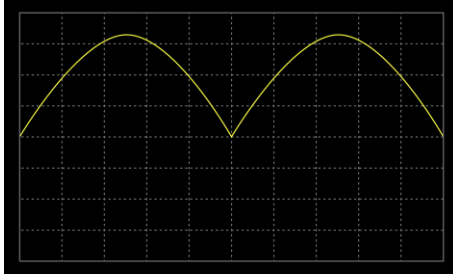
ARB 5 Sine wave, half-wave rectification (positive polarity)



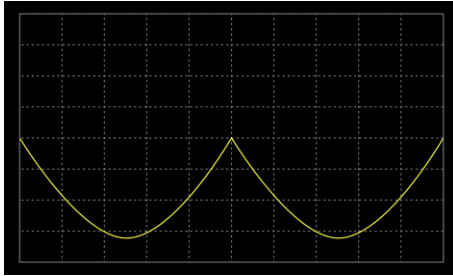
ARB 6 Sine wave, half-wave rectification(negative polarity)



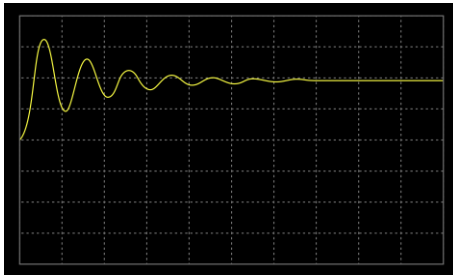
ARB 7 Sine wave, full-wave  
rectification(positive polarity)



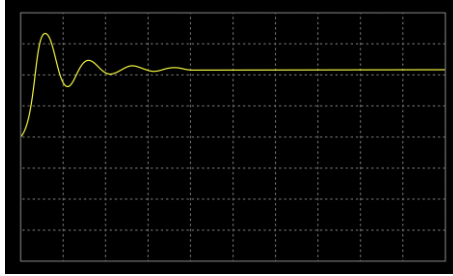
ARB 8 Sine wave, full-wave  
rectification(negative polarity)



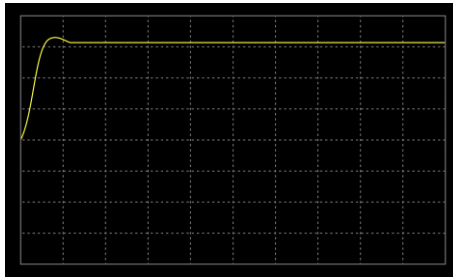
ARB 9 Second order step response(damping  
coefficient 0.1)



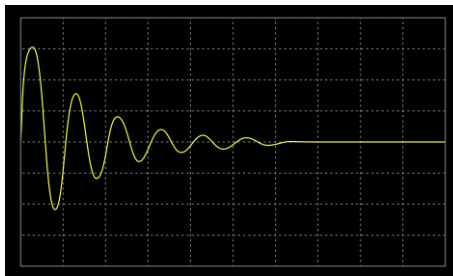
ARB 10 Second order step response(damping coefficient 0.2)



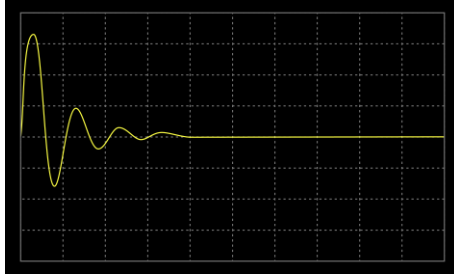
ARB 11 Second order step response(damping coefficient 0.7)



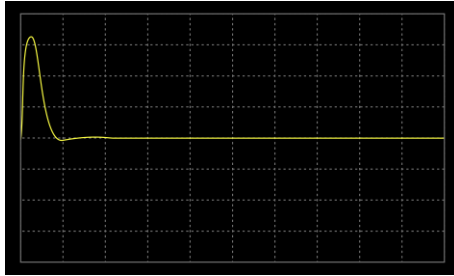
ARB 12 Second order impulse response(damping coefficient 0.1)



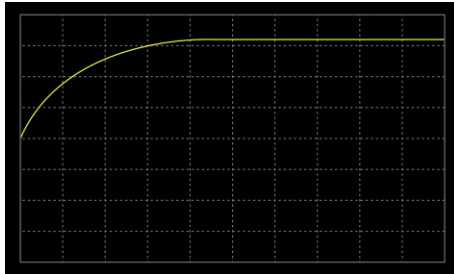
ARB 13 Second order impulse  
response(damping coefficient 0.2)



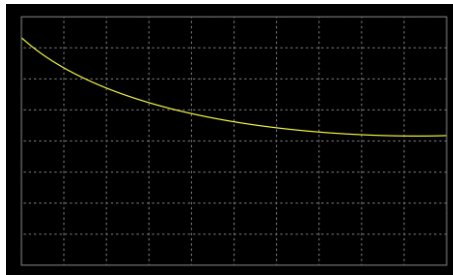
ARB 14 Second order impulse  
response(damping coefficient 0.7)



ARB 15 Exponential (rising)



ARB 16 Exponential (falling)

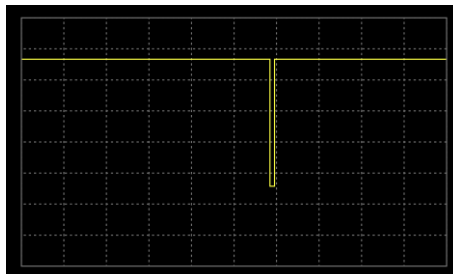


Note

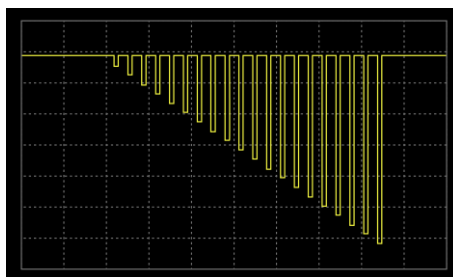
Only FW version V1.12 above can support the function of default waveform Setting in Factory Default Settings.

Default Sequence Setting

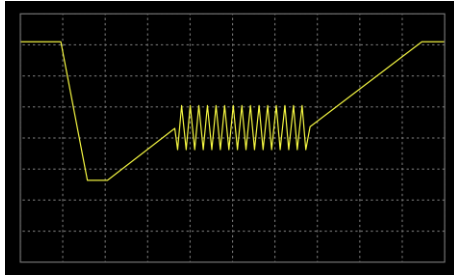
SEQ6 Momentary drop in supply voltage



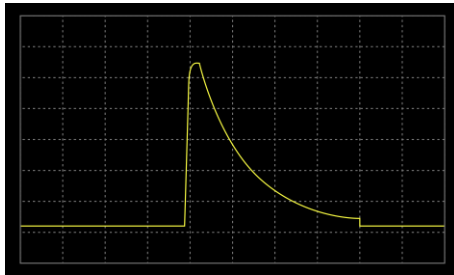
SEQ7 Reset test for Level1 systems with 12V



## SEQ8 Starting Profile



## SEQ9 Test2 Tr: 10ms, Td: 40ms



Note

Only FW version V1.12 above can support the function of Default Sequence Setting in Factory Default Settings.

# INDEX

|                              |     |                                   |        |
|------------------------------|-----|-----------------------------------|--------|
| Accessories .....            | 12  | Command list.....                 | 45     |
| Caution symbol .....         | 4   | Command syntax.....               | 41     |
| Cleaning the instrument..... | 6   | Error list .....                  | 156    |
| Default settings.....        | 164 | Ethernet.....                     | 23     |
| Disposal instructions.....   | 6   | GPIB .....                        | 32     |
| EN61010                      |     | LAN.....                          | 23     |
| measurement category .....   | 5   | RS232.....                        | 26     |
| pollution degree .....       | 6   | Status registers .....            | 140    |
| Environment                  |     | USB.....                          | 24     |
| safety instruction.....      | 6   | Remote control function check     |        |
| Ethernet                     |     | GPIB .....                        | 33     |
| interface .....              | 23  | Realterm.....                     | 29     |
| Front panel diagram .....    | 14  | RS-232.....                       | 25, 28 |
| Ground                       |     | USB.....                          | 25, 28 |
| symbol .....                 | 4   | Service operation                 |        |
| List of features .....       | 11  | about disassembly .....           | 5      |
| Model differences.....       | 9   | Socket server function check .... | 36     |
| Power on/off                 |     | UK power cord.....                | 7      |
| safety instruction.....      | 5   | Warning symbol .....              | 4      |
| Remote control.....          | 22  | Web server function check .....   | 35     |