

# Communication Interface Manual

## BIPOLAR POWER SUPPLY PBZ Series

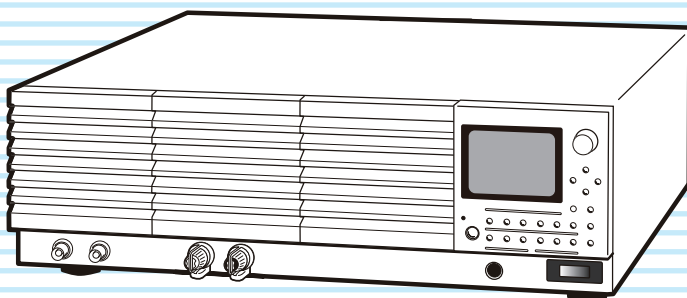
### PBZ20-20

### PBZ20-20A

### PBZ40-10

### PBZ60-6.7

### PBZ80-5



About the Operation Manuals 2

Notations Used in This Manual 2

[Remote Control Overview 3](#)

VISA Library 4

[Interface Setup 5](#)

Using the RS232C Interface 5

Using the GPIB Interface 7

Using the USB Interface 8

Using the LAN Interface (Option) 9

Accessing and Operating the PBZ from a Web Browser (LAN interface) 11

[Message Overview 14](#)

SCPI Command Syntax 14

Parameters 17

Command Description in This Manual 19

[IEEE 488.2 Common Commands 20](#)

[Output Setting Commands 24](#)

CV/CC Mode and Bipolar/Unipolar Mode 24

Selecting the Signal Source (Internal or External) and Setting the External Signal Source 24

DC Signal Setting 26

Output, Trigger Signals, and Screen Contrast 36

[Protection Features Commands 39](#)

Mode 39

Activation points (OCP/OVP) 39

Activation points (I.LIM/ V.LIM) 41

Mode and activation points (legacy command) 42

Clearing Alarms 43

[Measurement Function Commands 44](#)

Trigger Function 45

[Memory Function Commands 53](#)

Preset Memory 53

Setup Memory 53

[Sequence Function Commands 55](#)

Editing Programs and Steps 55

Executing Programs 64

Editing Scripts 65

[System Settings Commands 66](#)

CONFIG Settings 66

Basic System Settings 70

IEEE 488.2 Register Model 74

SCPI Register Model 76

[Default State 81](#)

[Processing time of Commands 86](#)

[List of Messages 87](#)









[List of Errors 96](#)

[Tutorial \(Visual Basic 2008\) 99](#)

[INDEX103](#)

## About the Operation Manuals

There are five PBZ Series Manuals listed as follows.

- **Setup Guide**    
This manual is intended for first-time users of this product. It provides an overview of the product and notes on usage. It also explains how to set up the product for testing the DUT. Always read this manual before using the product.
- **User's Manual**   
This manual is intended for first-time users of this product. It provides an overview of the product and notes on usage. It also explains how to configure the product, operate the product, perform maintenance on the product, and so on.
- **Communication Interface Manual**  (this manual)  
This manual contains details about remotely controlling the tester using SCPI commands.  
The interface manual is written for readers with sufficient basic knowledge of how to control measuring instruments using a PC.
- **Quick Reference**    
This manual explains Panel description and operation briefly.
- **Safety Information**    
This document contains general safety precautions for this product. Keep them in mind and make sure to observe them.

PBZ series manuals is intended for users of the product or persons teaching other users on how to operate the bipolar power supply. PBZ series manuals assumes that the reader has electrical knowledge.

PDF is provided on the included CD-ROM.  
Adobe Acrobat Reader is required to view the PDF file.

### Product firmware versions

This manual applies to products with firmware versions 2.2X (PBZ20-20/ PBZ40-10/ PBZ60-6.7/ PBZ80-5), 1.0X (PBZ20-20A).

When contacting us about the product, please provide us with:

- The model (marked in the top section of the front panel)
- The firmware version (see the user's manual)
- The serial number (marked in the top section of the rear panel)

### Before reading this manual

First read the User's Manual, which includes information on the product's hardware, to avoid connecting or operating the product incorrectly.

## Trademarks

Company names and product names used in this manual are trademarks or registered trademarks of their respective companies.

## Copyrights

The contents of this manual may not be reproduced, in whole or in part, without the prior consent of the copyright holder.

The specifications of this product and the contents of this manual are subject to change without prior notice.

© 2013 Kikusui Electronics Corporation

## Notations Used in This Manual

- The bipolar power supply PBZ series is referred to as the PBZ in this manual.
- The word "PC" used in this manual is a generic term for personal computers and workstations.
- The following markings are used in this manual.

### **WARNING**

**Indicates a potentially hazardous situation which, if ignored, could result in death or serious injury.**

### **CAUTION**

**Indicates a potentially hazardous situation which, if ignored, may result in damage to the product or other property.**

### **NOTE**

Indicates information that you should know.

### **DESCRIPTION**

Explanation of terminology or operation principle.

### **See**

Indicates reference to detailed information.

This chapter provides a general explanation of the remote control function.

In addition to controlling the PBZ from the front panel, you can control it remotely through the following standard-equipped interfaces.

- RS232C interface
- GPIB interface
- USB interface
- LAN interface (factory option)

If the factory option interface board is installed, you can use LAN.

You cannot control the PBZ through RS232C, GPIB, USB, and LAN at the same time.

The remote interfaces comply with IEEE Std 488.2-1992 and SCPI Specification 1999.0.

 p. 14

Familiarize yourself with the syntax of the SCPI commands that are used with the PBZ before you use them to control the product.

The RMT icon is displayed in the screen's status display area when the PBZ is in remote mode. To use the front panel to switch the PBZ back to local mode, press the LOCAL key.

## Measurement instrument interface standards

The PBZ complies with the following standards.

- IEEE Std 488.2-1992 IEEE Standard Codes, Formats, Protocols, and Common Commands For Use With IEEE Std 488.1-1987
- IEEE Std 488.1-1987 IEEE Standard Digital Interface for Programmable Instrumentation
- Standard Commands for Programmable Instruments (SCPI) version 1999.0
- Universal Serial Bus Specification Rev 2.0
- Universal Serial Bus Test and Measurement Class Specification (USBTMC) Rev 1.0
- Universal Serial Bus Test and Measurement Class, Subclass USB488 Specification (USBTMC-USB488) Rev 1.0
- TCP/IP Instrument Protocol Specification VXI-11
- LXI 1.4 Core Device Specification 2011

## VISA Library



To use the VISA library (VISA COM) with the I/O library, the VISA library must be installed on the controller (the host PC).

To use the USB interface to control the PBZ, a driver that supports the USB Test & Measurement Class (USBTMC) must be installed on the controller. The USBTMC driver is installed automatically by the VISA library.

When the product is controlled by the LAN interface, middleware which applies to the VXI-11 protocol is required. Middleware is automatically installed by VISA library.

VISA (Virtual Instrument Software Architecture) was developed by the VXIplug&play Systems Alliance. It is the standard specification for measurement instrument connection software.

You have to install one of the following VISA libraries (driver software that is implemented according to the VISA specifications).

Do not install multiple VISA libraries on the same PC. Doing so may cause errors.

VISA libraries that are older than the versions listed here do not support USB.

- NI-VISA by National Instruments Corporation (version 5.1.1 or later)
- Keysight VISA by Keysight Technologies, Inc. (Keysight IO Libraries Suite 16.0 or later)
- KI-VISA version 5.0..4 or later

KI-VISA is an original VISA library developed by Kikusui Electronics Corporation that supports the VXIplug&play VISA specifications. You can download the most recent version of this library from the Kikusui Electronics Corporation website (<http://www.kikusui.co.jp/download/>). If NI-VISA or Keysight VISA is already installed on your PC, you do not need to install KI-VISA.

You can download the “KI-VISA Library Programming Guide” from the Kikusui Electronics Corporation website.

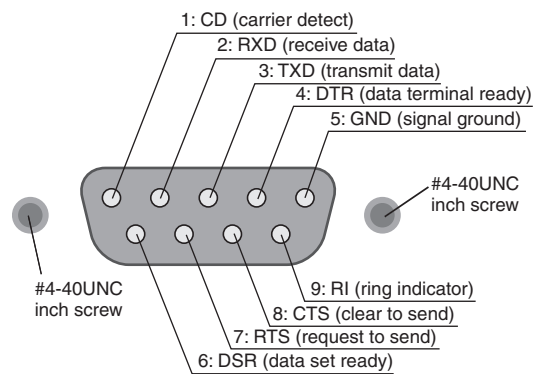
# 2 Interface Setup

## Using the RS232C Interface

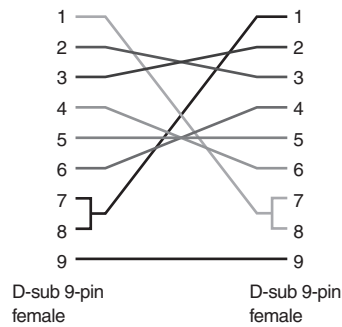
The PBZ RS232C interface is a standard D-sub, 9-pin male connector.

Check that the PBZ and your PC are off, and connect them with a standard cross cable (null-modem cable).

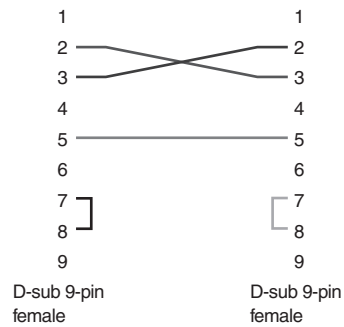
Use a D-sub, 9-pin, female-to-female AT cross cable. The figure below shows the port pinout. The PBZ does not use hardware handshaking (cross cable example 2).



Facing the PBZ rear panel



Cross cable example 1



Cross cable example 2

## RS232C Settings

Press Config key to set the interface type and communication parameters using CONFIG[6].

## Protocol

The RS232C protocol. In the following table, the underlined values are the factory default settings.

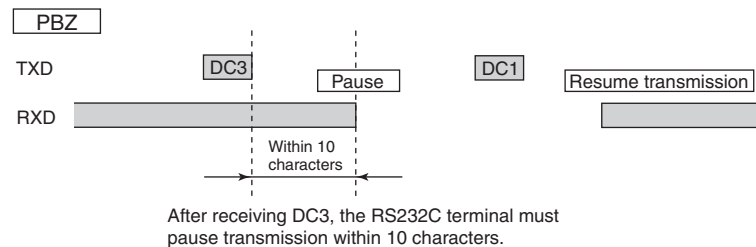
Item	Setting
Connector	Rear panel, D-sub, 9-pin connector
Baud rate	1200, 2400, 4800, 9600, <u>19200</u> , or 38400 bps
Data length	7 or <u>8</u>
Stop bits	<u>1</u> or 2
Parity	None (fixed)
Flow control	<u>ON</u> or OFF

## Flow control

Use flow control for RS232C transmission and reception. Device Control (DC) codes are used for this purpose.

Data may not be received properly if flow control is not used.

Code	Function	ASCII code
DC1 (Xon)	Request to send	11H
DC3 (Xoff)	Transmission stop request	13H



## Break signal

The break signal is used as a substitute for the IEEE488.1 dcl/sdc (Device Clear, Selected Device Clear) message.

## Using the GPIB Interface

### Setting the GPIB address

The factory default GPIB address is "1." You can set the address to a number from 1 to 30. Press Config key to set the interface type and GPIB address using CONFIG[6].

### GPIB feature

Function	Subset	Description
Source handshaking	SH1	Full capability
Acceptor handshaking	AH1	Full capability
Talker	T6	Function available
Listener	L4	Function available
Service request	SR1	Full capability
Remote local	RL1	Full capability
Parallel polling	PP0	No capability
Device clear	DC1	Full capability
Device trigger	DT1	Full capability
Controller	C0	No capability
Electrical interface	E1	Open-collector driver

### Service request

The PBZ is equipped with service request and serial polling functions.

## Using the USB Interface

---

To use the USB interface to control the PBZ, a driver that supports the USB Test & Measurement class (USBTMC) must be installed on the controller. The USBTMC driver is installed automatically by the VISA library.

### USB settings

Press Config key to set the interface type using CONFIG[6].

### USB feature

- Complies with USB specification 2.0
- Complies with USBTMC specification 1.0 and USBTMC-USB488 specification 1.0
- Baud rate: 12 Mbps maximum (full speed)
- VID (vendor ID): 0x0B3E
- PID (product ID): 0x1012

### Service request

The PBZ is equipped with service request and serial polling functions.





## Using the LAN Interface (Option)

The LAN interface is a factory option.

To use the LAN interface to control the PBZ, middleware that supports the VXI-11 protocol must be installed on the controller. The middleware is installed automatically by the VISA library.

There is a Web browser interface to the PBZ embedded in the LAN interface board. You can configure the LAN interface settings from your PC's Web browser.

For information on topics such as connecting to your corporate LAN, your IP address, your host name, and security, contact your network administrator.

### LAN connections

Use a standard LAN cable (category 5 and straight) to connect the PBZ to a network hub or router. Use a crossover cable when making a direct connection.

### LAN settings

Usually, LAN settings are configured using a DHCP server. If you want to use a fixed IP address, you cannot set it from the panel. Set it from a Web browser.

### CONFIG settings

Press Config key to set the interface type and communication parameters using CONFIG[6]. Normally, you should set DHCP and AUTO IP to ON to set the IP address automatically.

For direct connections, set DHCP to OFF and AUTO IP to ON and set the IP address automatically. You cannot set the IP address from the panel. To set a fixed IP address, first set the IP address automatically, and then set a fixed address by accessing the PBZ through a Web browser.

When you are going to use a fixed IP address over a long period of time or at intervals, we recommend that you specify a host name. To set a host name, access the PBZ through a Web browser.

### Service request

The PBZ is equipped with service request and serial polling functions.

### LAN feature

Depending on the operation that you perform through your Web browser, the PBZ may need to connect to the Internet.

Complies with the LXI 1.4 Core 2011

Complies with the VXI-11/ SCPI-RAW protocol

Baud rate: 100 Mbps maximum (auto negotiation)

DHCP client feature

AUTO IP feature

From your Web browser, you can (Internet Explorer; Mozilla Firefox; Safari/ Mobile Safari; Google Chrome):

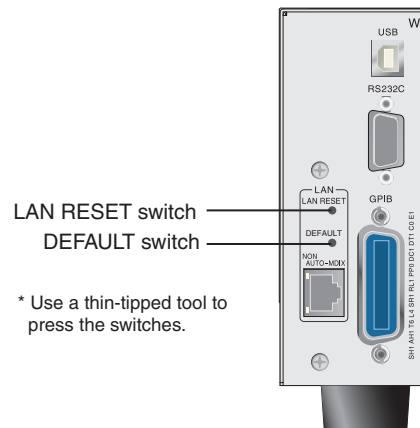
- Change LAN and security settings.

- Use simple control applications.

- Update the LAN board

 p. 11

## LAN RESET switch and DEFAULT switch



### LAN RESET switch

Holding down this switch for 3 seconds resets the LAN interface settings, except for Hostname and Hostname Description, to their factory default values. You can use this switch if you forget the security password.

### DEFAULT switch

Hold down this switch for 3 seconds or more to reset the LAN interface settings to the factory default values. You can use this switch if you forget the security password or IP address.

Neither of these switches affects the PBZ's panel settings. If the PBZ is in remote mode, pressing one of these switches will change the PBZ back into local mode (panel operation).



- **The LAN interface can be accessed from anywhere on the network that the PBZ is connected to. Change the security settings if necessary. Refer to " Accessing and Operating the PBZ from a Web Browser (LAN interface)" on p. 11 for setting procedure.**



- Do not use the DEFAULT and LAN RESET switches carelessly. This may cause the LAN interface to malfunction.
- The LAN interface is different from the other remote interfaces in that you have to use the "SYSTEM:REMOte" command to switch the PBZ to remote mode. When you are carrying out remote programming, send this command at the beginning of the program.

## Accessing and Operating the PBZ from a Web Browser (LAN interface)

For LAN interface, the detailed setting can be operated from the browser on the PC.

The URL of Web site is defined as adding "http://" in front of the IP address.

The URL can be entered directly on the address bar of the browser by confirming the IP address of the config setting (CONFIG[6] > LAN > IP ADDRESS).

(Example) When the IP address is 169.254.7.8

http://169.254.7.8

When VISA library is used, there is the function to retrieve the VXI-11 measuring instrument by the application program provided by VISA vendors (National Instruments NI-MAX, Agilent Connection Expert, and Kikusui KI-VISA Instrument Explorer, etc.). You can access the PBZ by clicking on the Web link from the retrieval results.

The following browsers are applied.

- Internet Explorer
- Mozilla Firefox
- Google Chrome
- Safari/ Mobile Safari

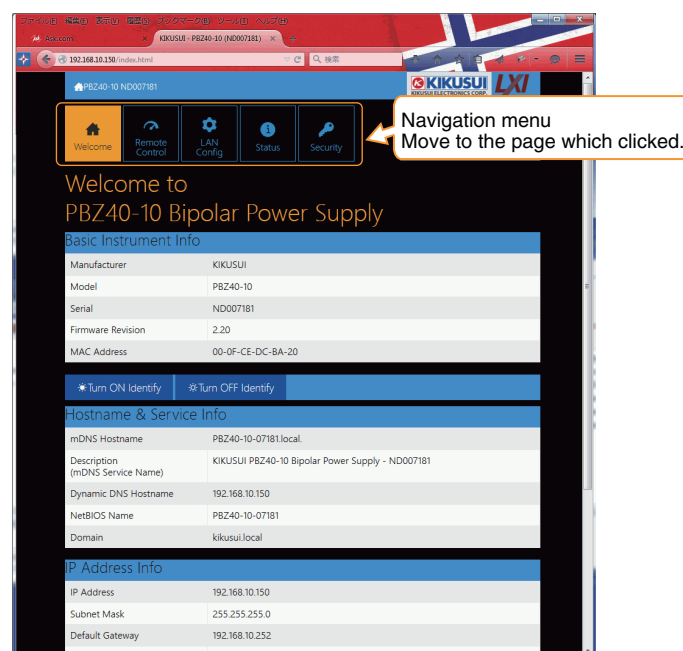
### WELCOME page

When you access the PBZ from a Web browser, the WELCOME page is displayed first.

The instrument information, network information, and VISA resource (I/O resource) information appear on the display.

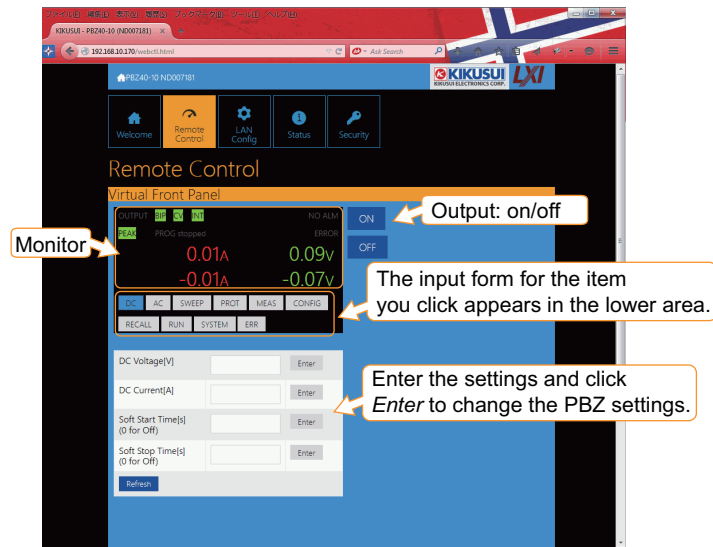
Clicking *Turn ON Identify* displays "Device Identify" on the PBZ series front panel display. This enables you to check which PBZ series is being controlled through the LAN interface.

Click the navigation menu to move to the other page.



## Remote Control page

You can set and control voltage, current, and output settings; view measured values; configure the protection feature.



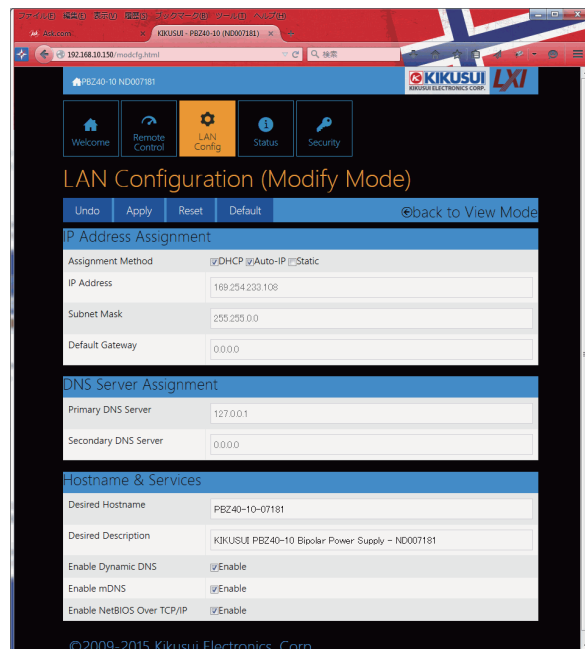
## LAN config page

Config page for the network settings.

Click *Modify Now* to assign the IP address, set the host name, and so on.

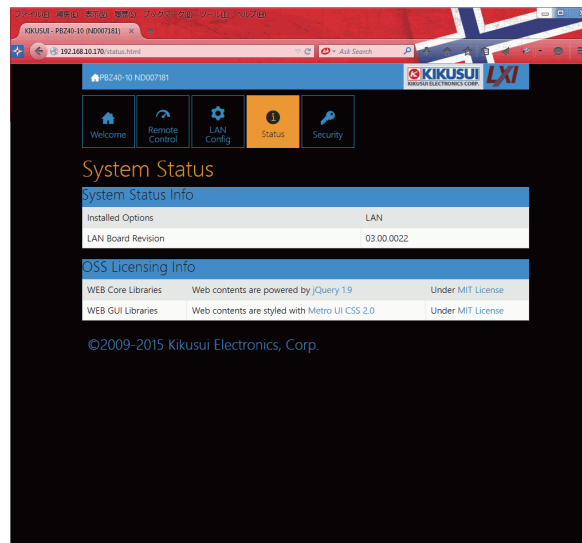
If you set the host name, you can use it in place of the IP address to access the LAN interface.

After you enter the settings, click *Apply* to apply the settings.



## STATUS page

This page shows the LAN board revision and OSS license information.

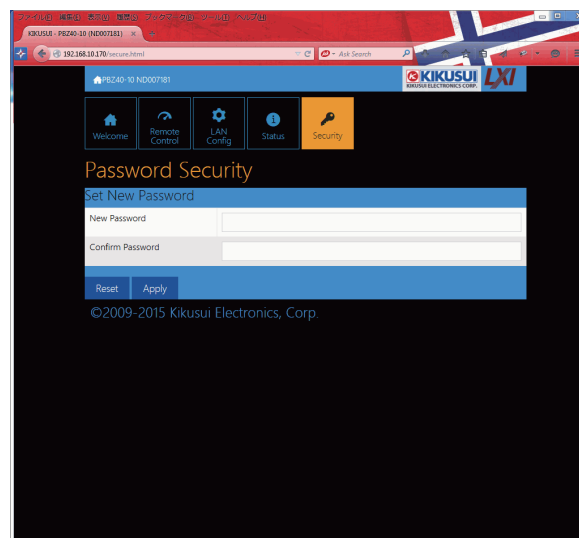


## SECURITY page

This page is for the security setting.

This page can allow you to set the change of the password protection.

The password protection is an effective security features to the Web site. It prevents from being changed inadvertently. The password can be used for any alphanumeric characters, the hyphen, and the underscore.



# 3

## Message Overview

The information that is transferred between the controller (the host PC) and the PBZ is referred to as “messages.”

The PBZ uses the SCPI language for these messages.

The messages that the PC sends to the PBZ are commands. The messages that the PBZ sends to the PC are responses.

Commands are used to execute functions or change settings on the PBZ or to query the PBZ’s settings or status. Responses are used to return the PBZ’s settings or status.

### SCPI Command Syntax

#### Command hierarchy

SCPI is an ASCII-based command language that was designed for test and measuring equipment. The command structure is composed of the common roots and nodes that are the building blocks of the SCPI subsystem. A command consists of a program header, parameters, and punctuation marks.

The following table uses the SOURce subsystem as an example to explain the hierarchy.

Program header	Parameter	Node level
SOURce		Root node
:SWEep		2nd level
:FREQuency		3rd level
:START	<numeric>	4th level
:STOP	<numeric>	4th level
:FUNCTion	{SIN SQE TRI USER[1]... USER[16]}	2nd level

- A colon (:) separates a higher node from a lower node.
- If the program header starts with a colon, the first node is a root node.

## Command syntax

### Format

In this manual, SCPI commands are expressed in the following format.

Example:

```
[SOURce:]FUNction[:SHApe][:IMMediate]
{SINusoid|SQUare|TRIangle|USER[1]..USER16}
```

- SCPI commands can be written in long form (with all the characters) or in short form (omitting the lowercase characters).  
SCPI commands can be transmitted in either long form or short form.
- SCPI commands are not case sensitive. VOLT, Volt, and volt are all received as the short form of the VOLTage command.  
VOLUME, Volume, and volume are all received as the long form of the VOLume command.
- A space separates a program header and its parameters.
- Multiple parameters are separated by commas.

### Compound commands

Compound commands can be created by concatenating two commands with a semicolon.

Example (compound command):

```
CURRent:PROTection:LOWer -5.0;UPPer 10.0
```

You can use a compound command to send a command that is the same as the two following commands.

Example (individual commands):

```
CURRent:PROTection:LOWer -5.0
CURRent:PROTection:UPPer 10.0
```

In the first command, CURRent:PROTection:LOWer -5.0, the path is set to CURRent:PROTection. Therefore, in the second command, CURRent:PROTection can be omitted.

If you specify a node that is not defined in the current path (except for OVER, UND, UPP, LOW, STAT, and PASS), an error will occur.

### Compound command (clearing the specified path)

Example:

```
SENSE:FUNction DC::INITiate
```

There are two root nodes in this compound command: SENSE and INITiate. When the second command or later begins with a colon, the path that was specified by the previous command is cleared.

- Program headers are separated by colons.
- By using colons and semicolons, you can concatenate commands of different subsystems.
- The maximum length of a command that you can transmit on a single line is 128 bytes.

## Special symbols and characters

The special symbols and characters that are used in this manual for the SCPI command syntax are explained below.

Symbol or character	Description
< >	Character strings inside the < and > symbols indicate program data. Do not include the < and > symbols in the actual program.
{ }	Characters and numbers delimited by “ ” inside the { and } symbols indicate that one of the delimited items is to be selected. Do not include the { and } symbols in the actual program.
[ ]	Character strings inside [ and ] indicate optional data. When optional data is not sent with the program, the default value is sent. Do not include the [ and ] symbols in the actual program.

## Queries

You can query the PBZ settings and status.

To make a query, append a question mark to the end of the program header section. If the query has parameters, insert a space after the question mark, and then write the parameters.

Example:

**VOLTAGE:AC? MIN**

### NOTE

If you want to send two queries on separate lines, send the second query after you have received the response to the first one. If you send query commands on two lines at the same time, you may receive an incomplete response.

## Terminating character strings

All commands must be terminated with a valid terminator.

The available terminators are <line feed> (ASCII 0x0A) and EOI (end-or-identify). You can use any one of these terminators to terminate a command.

EOI does not exist in the RS232C specification, so be sure to use <line feed>.

When you terminate a command string, the path is reset to the root level.

### NOTE

CR (ASCII 0x0D) is not a terminator.

## Common commands

See p. 20

There are commands that are common to the IEEE-488.2 and SCPI standards for functions such as resetting devices and performing self-diagnoses. These common commands start with an asterisk (“\*”). These commands may have one or multiple parameters.



## Parameters

The SCPI parameter format is derived from the program parameter format that is defined in IEEE 488.2.

The program data expression format that the PBZ uses is shown below.

### Non-numeric parameters

The PBZ uses the following three parameter types.

Symbol or character	Description
String data (String)	Used when a series of ASCII characters are requested. Be sure to enclose strings in single or double quotation marks. The opening and closing quotation marks must match (you cannot mix single and double quotation marks). Example: PROGram NAME "PBZ" If you want to include a quotation mark as part of the string, enter consecutive quotation marks (with no characters between them). ASCII codes 20H to 7EH can be used in strings.
Character data (Character)	Used when only a limited number of values are available for a program setting. Responses are returned in short form. Example: TRIGger:SOURce {BUS IMMediate}
Boolean data (Boolean)	Used to express a condition of 1 or 0, or ON or OFF. Responses are returned as 1 or 0. Example: OUTPut {ON OFF 1 0}

### Numeric parameters

The PBZ uses the following five parameter types.

Symbol or character	Description
NR1	Represents an integer value. <sup>1</sup>
NR2	Represents a real number in floating-point format. <sup>1</sup>
NR3	Represents a real number in scientific notation. <sup>1</sup> Response data is normally returned with six decimal places. If the 380 is returned in the response data, it is returned as +3.800000+E02. This product has queries that return data with five decimal places.
NRf	NRf is a generic term that includes NR1, NR2, and NR3.
Numeric	Represents values such as the decimal point, optional prefixes, and measurement units. Numbers are expressed the same as NRf. MINimum and MAXimum are available as substitutes for declaring certain values. You can also use units such as V, A, and S in numeric parameters. If a value that cannot be assigned is entered, the PBZ rounds the value to the closest possible value. Example: SYSTem:CONFigure:BEEPer:VOLume 0.56 The buzzer volume must be set to a value from 0.1 to 1.0, so if you send the query SYST:CONF:BEEP:VOL?, 0.6 will be returned.

<sup>1</sup> Details are given in the "IEEE 488.2 Standard Digital Interface for Programmable Instrumentation."

## Special form numeric parameters

The special form numeric parameters MINimum and MAXimum can be used as substitutes for the actual maximum and minimum values when the parameter is numeric.

The following example sets the measurement time to the minimum value.

**SENSe:APERTure MINimum**

You can query the minimum and maximum values for most parameters.

**SENSe:APERTure? MIN**

**SENSe:APERTure? MAX**

## Measurement units

The default measurement units are listed below. Commands are accepted even if measurement units are not specified.

- A (current)
- V (voltage)
- A/V (current/voltage)
- V/V (voltage/voltage)
- W (power)
- S (seconds)
- PCT (%)
- HZ (frequency)
- DEG (degrees)

The following optional prefixes are supported. To enter “μ” in the parameter, use “U.”

- M (milli)
- K (kilo)
- U (micro)

---

### NOTE

The unit symbols in the International System of Units contain lowercase characters. The IEEE standard uses uppercase characters. SCPI commands are not case sensitive.

---

# Command Description in This Manual

In this manual, commands are described in the following manner.

\* RST \* RCL

**CURR:PROT:OVER**

Sets the positive OCP's current activation point.

**Command** [SOURCE:] CURRENT:PROTECTION[:LEVEL]:OVER {<numeric>|MIN|MAX}  
[SOURCE:] CURRENT:PROTECTION[:LEVEL]:OVER?

**Parameter** Value: -1 % to 110 % of the rated output current  
 (The default value is 110 % of the rated output current.)  
Unit: A

**Response** Returns the positive current activation point in <NR3> format.

See

Commands that have these marks are affected when an \*RST or \*RCL command is sent. The settings for the command are changed to the values that are shown in the "Default State" section.

The parameters are listed. In this command, the parameter is numeric. In addition to specifying the desired value, you can specify the minimum or maximum value.

The commands are listed in the long form. The lowercase characters can be omitted. Sections that are enclosed in braces ([]) can also be omitted.

The selectable range is given. Optional symbols such as m and μ can also be used.

The unit for the value being set. The unit can be omitted.

Specify MAX to set the maximum value.

Specify MIN to set the minimum value.

The format of the value that is returned when a query is sent.

## References to command descriptions

Item	Page
Command syntax	14
Parameters	17
Units	18
Default state <span style="border: 1px solid black; border-radius: 5px; padding: 2px;">* RST</span> <span style="border: 1px solid black; border-radius: 5px; padding: 2px;">* RCL</span>	81
Queries	16
List of messages	87
List of errors	96

# 4

## IEEE 488.2 Common Commands

### \*CLS

 p. 72

Clears all event registers including the status byte, event status, and error queue.

**Command** \*CLS

### \*ESE

 p. 76

Sets the event status enable register that is counted by the event summary bit (ESB) of the status byte.

**Command** \*ESE <NR1>  
\*ESE?

**Parameter** Value: 0 to 255  
An SCPI error (-222, "Data out of range") occurs if the specified value is outside the range.

**Example** When \*ESE 16 is transmitted, bit 4 of the event status enable register is set. Each time the execution error bit (bit 4) of the event status register is set, the summary bit (ESB) of the status byte is set.

**Response** Returns the value of the event status enable register in <NR1> format.

### \*ESR

 p. 76

Queries the event status register. Registers that are read are cleared.

**Command** \*ESR?

**Response** Returns the value of the event status register in <NR1> format and clears the register.

### \*IDN

Queries the model name, serial number, and firmware version of the PBZ.

**Command** \*IDN?

**Response** The response to \*IDN? is indicated below.

**Example** For a PBZ20-20 with serial number AB123456 and firmware version 1.00, \*IDN? returns:

KIKUSUI,PBZ20-20,AB123456,1.00.

**\*OPC**

IEEE 488.2-1992  
Section 10.18

Sets the OPC bit (bit 0) of the event status register when all the commands that are in standby have been processed.

**Command** \*OPC  
\*OPC?

**Response** Returns "1" when all the commands that are in standby have been processed.

**\*OPT**

Queries the options that are installed in the PBZ.

**Command** \*OPT?

**Response** If the factory option LAN interface board is installed, "LAN" is returned. Returns "0" if no options are installed.

**\*PSC**

IEEE 488.2-1992  
Section 10.25

When the result of rounding the specified value is a number other than zero, it is interpreted as one, and this command sets the service request enable and event status enable registers to their initial values when the PBZ restarts.

**Command** \*PSC <NR1>  
\*PSC?

Parameter Value: -32767 to +32767

**Response** Returns "1" when the result of rounding the specified value is a number other than zero.

**\*RCL**

p. 81

Aborts measurement and loads the settings that have been saved to setup memory. This command performs the same function as the MEM:SET:RCL command.

For the commands that are affected by \*RCL, see "Default State".

**Command** \*RCL <NR1>

Parameter Value: 0 to 9 Memory number  
An SCPI error (-222, "Data out of range") occurs if the specified value is outside the range.

**\*RST**

p. 81

Aborts measurement and initializes the PBZ to the factory default settings.

For the commands that are affected by \*RST, see "Default State".

**Command** \*RST

**\*SAV**

Saves the current settings to setup memory. This command performs the same function as the MEM:SET:SAV command.

 p. 81

For the commands that are affected by \*RCL, see "Default State".

**Command \*SAV <NR1>**

Parameter Value: 0 to 9            Memory number  
An SCPI error (-222, "Data out of range") occurs if the specified value is outside the range.

**\*SRE**

Sets the service request enable register.

The service request enable register can be used to select which summary messages in the status byte register will perform service requests.

To clear the service request enable register, send \*SRE 0. If the register is cleared, service requests cannot be generated by status information.

**Command \*SRE <NR1>  
\*SRE?**

Parameter Value: 0 to 255  
An SCPI error (-222, "Data out of range") occurs if the specified value is outside the range.

Example Sending \*SRE 8 sets bit 3 of the service request enable register. Each time the summary bit (bit 3) of the QUESTIONABLE status register in the status byte is set, a service request message is generated.

**Response** Returns the value of the service request enable register in <NR1> format.

**\*STB**

 p. 74


Queries the contents of the status byte register and the MSS (master summary status) message.

The response is the same as serial polling only with the exception that the MSS message appears in place of the RQS message in bit 6.

**Command \*STB?**

**Response** Returns the value of the status byte register and the MSS message (bit 6) in <NR1> format.

**\*TRG**

  
IEEE 488.2-1992  
Section 10.37

Trigger command.

This is a substitute command for the IEEE 488.1 get message (Group Execute Trigger). If the PBZ is in a state in which it does not accept triggers, an SCPI error (-211, "Trigger ignored") occurs.

**Command \*TRG**

**\*TST**

IEEE 488.2-1992  
Section 10.38

Executes a self-test. You can query which error occurred by sending the SYST:ERR? command.

**Command** \*TST?

**Response** Returns "0" if no errors are detected. Returns the error code if an error is detected.

**\*WAI**

Prevents the PBZ from executing subsequent commands until all operations that are in standby have completed.

**Command** \*WAI

# 5

## Output Setting Commands

### CV/CC Mode and Bipolar/Unipolar Mode

#### FUNC:MODE

\* RST \* RCL

Sets the CV/CC mode setting.

**Command** [SOURCE:]FUNCTION:MODE {CC|CV}  
[SOURCE:]FUNCTION:MODE?

Parameter	Value:	CC	CC mode
		CV	CV mode (default)

**Response** Returns the setting in <character> format in response to the FUNC:MODE? query.

#### FUNC:POL

\* RST \* RCL

Sets the bipolar/unipolar mode setting.

**Command** [SOURCE:]FUNCTION:POLarity {BIPolar|UNIPolar}  
[SOURCE:]FUNCTION:POLarity?

Parameter	Value:	BIPolar	Bipolar mode (default)
		UNIPolar	Unipolar mode

**Response** Returns the setting in <character> format in response to the FUNC:POL? query.

### Selecting the Signal Source (Internal or External) and Setting the External Signal Source

#### FUNC:SOUR

\* RST \* RCL

Sets the signal source (internal, external, or both).

**Command** [SOURCE:]FUNCTION:SOURce {INTernal|EXTernal|BOTH}  
[SOURCE:]FUNCTION:SOURce?

Parameter	Value:	INTernal	Internal signal source (default)
		EXTernal	External signal source
		BOTH	Internal signal source and external signal source are added (ADD)

**Response** Returns the setting in <character> format in response to the FUNC:SOUR? query.



**FUNC:EXT:TERM**

\* RST \* RCL

Selects the external signal source terminal.

**Command** [SOURCE:]FUNCTION:EXTERNAL:TERMINAL {BNC|J1|BOTH}  
[SOURCE:]FUNCTION:EXTERNAL:TERMINAL?

Parameter	Value:	BNC	BNC terminal (default)
		J1	J1 connector
		BOTH	BNC terminal + J1 connector

**Response** Returns the setting in <character> format in response to the FUNC:EXT:TERM? query.

**FUNC:EXT:CURR:GAIN**  
**FUNC:EXT:VOLT:GAIN**

\* RST \* RCL

Sets the gain of the external signal source BNC terminal input.

**Command** [SOURCE:]FUNCTION:EXTERNAL:CURRENT:GAIN {<numeric>|MIN|MAX}  
[SOURCE:]FUNCTION:EXTERNAL:VOLTAGE:GAIN {<numeric>|MIN|MAX}

**Command** [SOURCE:]FUNCTION:EXTERNAL:CURRENT:GAIN? [{MIN|MAX}]  
[SOURCE:]FUNCTION:EXTERNAL:VOLTAGE:GAIN? [{MIN|MAX}]

Parameter	Value:	PBZ20-20/ PBZ20-20A: 0 to 20 (CV or CC mode)
		PBZ40-10: 0 to 40 (CV mode), 0 to 10 (CC mode)
		PBZ60-6.7: 0 to 60 (CV mode), 0 to 6.7 (CC mode)
		PBZ80-5: 0 to 80 (CV mode), 0 to 5 (CC mode)
		(The default value is 0.)
Unit:		A/V or V/V

**Response** Returns the gain in <NR3> format in response to the FUNC:EXT:CURR:GAIN? and FUNC:EXT:VOLT:GAIN? queries.

## DC Signal Setting

### VOLT CURR

\* RST \* RCL

Sets the DC signal voltage and current.

**Command** [SOURCE:]VOLTage[:LEVel][:IMMediate][:AMPLitude]  
{<numeric>|MIN|MAX}  
[SOURCE:]CURRent[:LEVel][:IMMediate][:AMPLitude]  
{<numeric>|MIN|MAX}

**Command** [SOURCE:]VOLTage[:LEVel][:IMMediate][:AMPLitude]? [{MIN|MAX}]  
[SOURCE:]CURRent[:LEVel][:IMMediate][:AMPLitude]? [{MIN|MAX}]

Parameter Value: 0 % to 105 % of the rated output (The default value is 0 % of the rated output.)  
Unit: V or A

**Response** Returns the voltage setting in <NR3> format in response to the VOLT? and CURR? queries.

### VOLT:TRIG CURR:TRIG

Sets the DC signal voltage and current when software triggers are received.

**Command** [SOURCE:]VOLTage[:LEVel]:TRIGgered[:AMPLitude]  
{<numeric>|MIN|MAX}  
[SOURCE:]CURRent[:LEVel]:TRIGgered[:AMPLitude]  
{<numeric>|MIN|MAX}

**Command** [SOURCE:]VOLTage[:LEVel]:TRIGgered[:AMPLitude]? [{MIN|MAX}]  
[SOURCE:]CURRent[:LEVel]:TRIGgered[:AMPLitude]? [{MIN|MAX}]

Parameter Value: 0 % to 105 % of the rated output  
Unit: V or A

**Response** Returns the setting in <NR3> format in response to the VOLT:TRIG? and CURR:TRIG? queries.

## AC Signal Command

### Setting the AC Signal On/Off State

#### AC:STAT

\* RST \* RCL

Sets the AC signal on/off state.

**Command** [SOURCE:]AC:STATE {ON|OFF|1|0}  
[SOURCE:]AC:STATE?

Parameter	Value:	ON (1)	On
		OFF (0)	Off (default)

**Response** Returns the AC signal on/off state in <NR1> format in response to the AC:STAT? query.

#### AC:STAT:TRIG

Sets the AC signal on/off state when software triggers are received.

**Command** [SOURCE:]AC:STATE:TRIGGERED {ON|OFF|1|0}  
[SOURCE:]AC:STATE:TRIGGERED?

Parameter	Value:	ON (1)	On
		OFF (0)	Off

**Response** Returns the AC signal on/off state in <NR1> format in response to the AC:STAT:TRIG? query.

### Fixed Waveforms and User-defined Arbitrary Waveforms

#### FUNC

\* RST \* RCL

Selects the AC signal waveform.

**Command** [SOURCE:]FUNCTION[:SHAPE][:IMMEDIATE]  
{SINUSOID|SQUARE|TRIANGLE|USER[1]..USER16}  
[SOURCE:]FUNCTION[:SHAPE][:IMMEDIATE]?

Parameter	Value:	SINUSOID	Sine wave(default)
		SQUARE	Square wave
		TRIANGLE	Triangle wave
		USER[1] to USER16	User-defined arbitrary waveform (ARB1 to ARB16)

**Response** Returns the AC signal waveform type in <character> format in response to the FUNC? query.

## FUNC:TRIG

Selects the AC signal waveform when software triggers are received. The settable ranges for the VOLT:TRIG, CURR:TRIG, VOLT:AC:TRIG, and CURR:AC:TRIG commands differ based on the FUNC:TRIG setting. When this command is received, the PBZ first checks the relationship between these settings. To ensure proper operation, send the FUNC:TRIG command first. When the FUNC:TRIG command is received, the PBZ immediately sets the VOLT:TRIG, CURR:TRIG, VOLT:AC:TRIG, and CURR:AC:TRIG values to the current values.

**Command** [SOURCE:]FUNCTION[:SHAPE]:TRIGGERED  
{SINusoid|SQUare|TRIangle|USER[1]..USER16}  
[SOURCE:]FUNCTION[:SHAPE]:TRIGGERED?

Parameter	Value:	SINusoid	Sine wave
		SQUare	Square wave
		TRIangle	Triangle wave
		USER[1] to USER16	User-defined arbitrary waveform

**Response** Returns the AC signal waveform type in <character> format.

## USER{[1]|2|...|16}:TITL

Sets the user-defined arbitrary waveform's title to a string of 15 characters in length. A space character(0x20) is inserted automatically in case of a string of less than 15 characters in length. The user-defined arbitrary waveforms are from USER1 to USER16. You can write USER1 as USER (omitting the "1").

**Command** [SOURCE:]USER{[1]|2|...|16}:TITLE <"string">  
[SOURCE:]USER{[1]|2|...|16}:TITLE?

Parameter	Value:	"string"
	Example	"My Waveform"

**Response** Returns the user-defined arbitrary waveform's title in <"string"> format.

## USER{[1]|2|...|16}:DATA:FORM:BORD

Sets the binary transmission data byte order of the user-defined arbitrary waveform. This command cannot be used with the RS232C interface.

**Command** [SOURCE:]USER{[1]|2|...|16}:DATA:FORM:BOARD  
{NORMal|SWAPped}  
[SOURCE:]USER{[1]|2|...|16}:DATA:FORM:BOARD?

Parameter	Value:	NORMal	Big endian (most significant byte first)
		SWAPped	Little endian (least significant byte first)

**Response** Returns the binary transmission data byte order of the user-defined arbitrary waveform in <character> format.

### USER{[1]|2|...|16}:DATA

Sets the user-defined arbitrary waveform pattern. The user-defined arbitrary waveforms are from USER1 to USER16. You can write USER1 as USER (omitting the "1"). This command cannot be used with the RS232C interface or LAN interface.

To use the RS232C or LAN interface to set a user-defined arbitrary waveform pattern, use USER{[1]|2|...|16}:DATA:VAL.

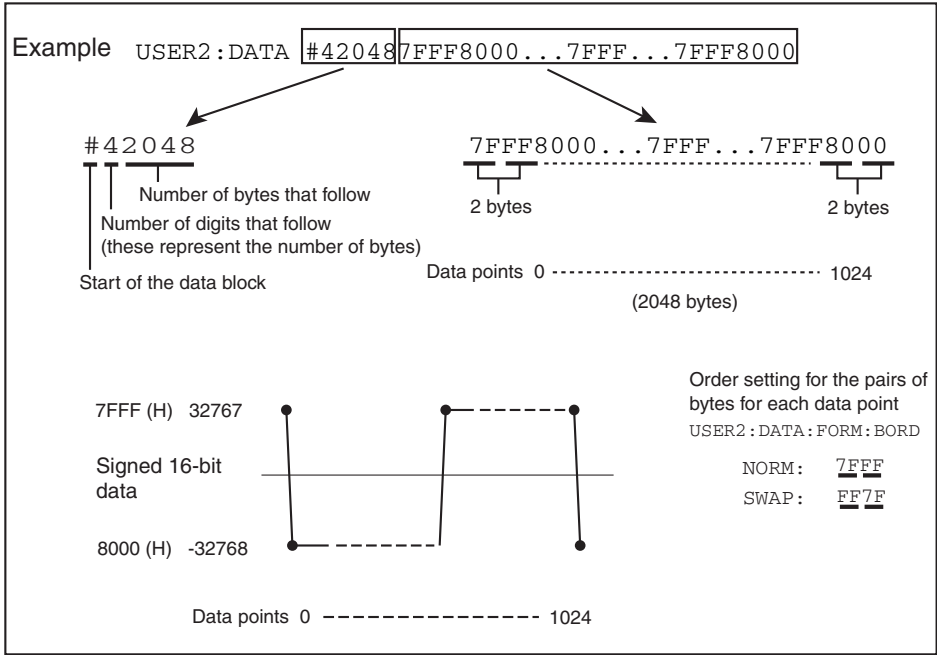
**Command** [SOURCE:]USER{ [1] | 2 | .. | 16 } : DATA <arbitrary block data>  
[SOURCE:]USER{ [1] | 2 | .. | 16 } : DATA?

**Parameter Value:** Arbitrary block data (in binary block format)

**Response** Returns the user-defined arbitrary waveform pattern in <arbitrary block data> format.

### Binary block format data

Waveform data is represented as integer values that are composed of 2 bytes (16 bits) of binary data. Therefore, the total number of bytes used for a waveform's data is the number of data points in the waveform times two. This number is always even. For example, if the number of waveform data points is 1024, the total number of bytes in the waveform data is 2048.



**USER{[1]|2|...|16}:DATA:VAL**

Sets the waveform data relative to the waveform peak value for any of 1024 divisions of one period of a user-defined arbitrary waveform. These waveform divisions are referred to as “addresses.” The user-defined arbitrary waveforms are from USER1 to USER16. You can write USER1 as USER (omitting the “1”).

**Command** [SOURCE:]USER{[1]|2|...|16}:DATA:VALue <addr\_NR1>,<data\_NR1>  
[SOURCE:]USER{[1]|2|...|16}:DATA:VALue? <addr\_NR1>

Parameter <addr\_NR1>Address

Value: 0 to 1023

<data\_NR1>Waveform data

Value: -32768 to +32767

Example USER10:DATA:VAL 0,32767

Sets address 0 of the user-defined arbitrary waveform pattern to 32767.

To enter data for a square wave, set addresses 0 to 511 to +32767, and set addresses 512 to 1023 to -32768.

**Response** Returns the value at an address in a user-defined arbitrary waveform pattern in <NR1> format.

**Signal Amplitude, Frequency, and Start Phase, and Square Wave Duty Cycle****VOLT:AC**

\* RST \* RCL

Sets the AC signal voltage.

**Command** [SOURCE:]VOLTage:AC[:IMMEDIATE][:AMPLITUDE]  
{<numeric>|MIN|MAX}  
[SOURCE:]VOLTage:AC[:IMMEDIATE][:AMPLITUDE]? [{MIN|MAX}]

Parameter Value: 0 % to 105 % of the rated output voltage  
(The default value is 0 % of the rated output voltage.)

Unit: VPP

**Response** Returns the setting in <NR3> format in response to the VOLT:AC? query.

**CURR:AC**

\* RST \* RCL

Sets the AC signal current.

**Command** [SOURCE:]CURRENT:AC[:IMMEDIATE][:AMPLITUDE]  
{<numeric>|MIN|MAX}  
[SOURCE:]CURRENT:AC[:IMMEDIATE][:AMPLITUDE]? [{MIN|MAX}]

Parameter Value: 0 % to 105 % of the rated output  
(The default value is 0 % of the rated output current.)

Unit: APP

**Response** Returns the setting in <NR3> format in response to the CURR:AC? query.

## VOLT:AC:TRIG CURR:AC:TRIG

Sets the AC signal voltage and current when software triggers are received.

**Command** [SOURCE:]VOLTage:AC:TRIGgered[:AMPLitude]{<numeric>|MIN|MAX}  
[SOURCE:]CURRent:AC:TRIGgered[:AMPLitude]{<numeric>|MIN|MAX}

**Command** [SOURCE:]VOLTage:AC:TRIGgered[:AMPLitude]? [{MIN|MAX}]  
[SOURCE:]CURRent:AC:TRIGgered[:AMPLitude]? [{MIN|MAX}]

Parameter Value: 0 % to 105 % of the rated output  
Unit: VPP or APP

**Response** Returns the setting in <NR3> format in response to the VOLT:AC:TRIG? and CURR:AC:TRIG? queries.

## FREQ

\* RST \* RCL

Sets the AC signal frequency.

**Command** [SOURCE:]FREQuency[:IMMediate]{<numeric>|MIN|MAX}  
[SOURCE:]FREQuency[:IMMediate]? [{MIN|MAX}]

Parameter Value: PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5:  
+1.0E-02 Hz to +1.0E+05 Hz (The default value is 1 kHz.)  
Maximum CV mode: +1.0E+05 (100 kHz)  
CC mode: +1.0E+05 (100 kHz)

PBZ20-20A:  
+1.0E-02 Hz to +2.0E+05 Hz (The default value is 1 kHz.)  
Maximum CV mode: +2.0E+05 (200 kHz)  
CC mode: +2.0E+05 (200 kHz)

Unit: HZ

**Response** Returns the setting in <NR3> format in response to the FREQ? query.

## FREQ:TRIG

Sets the AC signal frequency when software triggers are received.

**Command** [SOURCE:]FREQuency:TRIGgered{<numeric>|MIN|MAX}  
[SOURCE:]FREQuency:TRIGgered? [{MIN|MAX}]

Parameter Value: PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5:  
+1.0E-02 Hz to +1.0E+05 Hz  
Maximum CV mode: +1.0E+05 (100 kHz)  
CC mode: +1.0E+05 (100 kHz)

PBZ20-20A:  
+1.0E-02 Hz to +2.0E+05 Hz  
Maximum CV mode: +2.0E+05 (200 kHz)  
CC mode: +2.0E+05 (200 kHz)

Unit: HZ

**Response** Returns the setting in <NR3> format in response to the FREQ:TRIG? query.

**PHAS**

\* RST \* RCL

Sets the AC signal start phase angle.

**Command** [SOURCE:] PHASe {<numeric>|MIN|MAX} , {ON|OFF|1|0}  
[SOURCE:] PHASe?

Parameter Value: 0 to 359 (The default value is 0.)  
Unit: DEG (degrees)

Parameter Value: ON (1) Set the start phase angle (default)  
OFF (0) Do not set the start phase angle

**Response** Returns the start phase angle setting and whether the start phase angle is set or not in <NR3>,<NR1> format in response to the PHAS? query.

**SQU:DCYC**

\* RST \* RCL

Sets the square wave signal duty cycle.

**Command** [SOURCE:] SQUare:DCYCLe {<numeric>|MIN|MAX}  
[SOURCE:] SQUare:DCYCLe? [ {MIN|MAX} ]

Parameter Value: 0.1 % to 99.9 % (The default value is 50.0 %.)  
The settable range depends on the settings of the AC signal frequency.  
Unit: PCT

**Response** Returns the setting in <NR3> format in response to the SQU:DCYC? query.

**Frequency Sweep****SWE:STAT**

\* RST \* RCL

Turns the AC signal frequency sweep on and off.

**Command** [SOURCE:] SWEep:STATe {ON|OFF|1|0}  
[SOURCE:] SWEep:STATe?

Parameter Value: ON (1) Frequency sweeps are executed.  
OFF (0) Frequency sweeps are not executed (default).

**Response** Returns whether frequency sweeps are on or off in <NR1> format in response to the SWE:STAT? query.



**SWE:SPAC**

\* RST \* RCL

Sets the AC signal frequency sweep mode.

**Command** [SOURCE:]SWEep:SPACing {LOGarithmic|LINear}  
[SOURCE:]SWEep:SPACing?

Parameter Value: LOGarithmic Logarithmic sweep (default)  
LINear Linear sweep

**Response** Returns the AC signal frequency sweep mode in <character> format in response to the SWE:SPAC? query.

**SWE:TIME**

\* RST \* RCL

Sets the AC signal frequency sweep time.

**Command** [SOURCE:]SWEep:TIME {<numeric>|MIN|MAX}  
[SOURCE:]SWEep:TIME? [{MIN|MAX}]

Parameter Value: 0.0001 to 1000 (The default value is 1.)  
Unit: S

**Response** Returns the sweep time in <NR3> format in response to the SWE:TIME? query.

**SWE:FREQ:STAR  
SWE:FREQ:STOP**

\* RST \* RCL

Sets the AC signal frequency sweep start and stop frequencies.

**Command** [SOURCE:]SWEep:FREQuency:STARt[:IMMediate]  
{<numeric>|MIN|MAX}  
[SOURCE:]SWEep:FREQuency:STOP[:IMMediate] {<numeric>|MIN|MAX}

**Command** [SOURCE:]SWEep:FREQuency:STARt[:IMMediate]? [{MIN|MAX}]  
[SOURCE:]SWEep:FREQuency:STOP[:IMMediate]? [{MIN|MAX}]

Parameter Value: PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5:  
CV mode +1.0E-2 Hz to +1.0E+05 Hz  
CC mode +1.0E-2 Hz to +1.0E+05 Hz  
The default start frequency is 100 Hz.  
The default stop frequency is 1 kHz.

PBZ20-20A:  
CV mode +1.0E-2 Hz to +2.0E+05 Hz  
CC mode +1.0E-2 Hz to +2.0E+05 Hz  
The default start frequency is 100 Hz.  
The default stop frequency is 1 kHz.

Unit: HZ

**Response** Returns the start or stop frequency in <NR3> format in response to the SWE:FREQ:STAR? or SWE:FREQ:STOP? query.

**MARK**

\* RST \* RCL

Sets the type of output that the TRIG OUT terminal generates. Select MARKER signals or SYNC signals. The setting of the type of the TRIG OUT terminal output is valid when trigger signal output setting is on.

**Command** [SOURCE:]MARKer {ON|OFF|1|0}  
[SOURCE:]MARKer?

Parameter Value: ON (1) MARKER signals  
OFF (0) SYNC signals (default)

**Response** Returns the type of output that the TRIG OUT terminal generates in <NR1> format in response to the MARK? query.

**MARK:FPO**

\* RST \* RCL

Sets the marker frequency of the AC signal frequency sweep.

**Command** [SOURCE:]MARKer:FPoint {<numeric>|MIN|MAX}  
[SOURCE:]MARKer:FPoint? [{MIN|MAX}]

Parameter Value: PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5:  
CV mode +1.0E-2 Hz to +1.0E+05 Hz (The default value is 500 Hz.)  
CC mode +1.0E-2 Hz to +1.0E+05 Hz (The default value is 500 Hz.)  
The settable frequency range is greater than the start frequency and less than the stop frequency.  
MIN is the sweep start or stop frequency, whichever is lower.  
MAX is the sweep start or stop frequency, whichever is higher.

PBZ20-20A:

CV mode +1.0E-2 Hz to +2.0E+05 Hz (The default value is 500 Hz.)  
CC mode +1.0E-2 Hz to +2.0E+05 Hz (The default value is 500 Hz.)  
The settable frequency range is greater than the start frequency and less than the stop frequency.  
MIN is the sweep start or stop frequency, whichever is lower.  
MAX is the sweep start or stop frequency, whichever is higher.

Unit: HZ

**Response** Returns the marker frequency in <NR3> format in response to the MARK:FPO? query.

## Collective Settings (Waveform, Frequency, Signal Amplitude, and DC Signal)

**APPL:DC**  
**APPL:SIN**  
**APPL:SQU**  
**APPL:TRI**  
**APPL:USER{[1]|2|...|16}**  
**APPL**

\* RST \* RCL

Collectively sets the signal type, AC signal frequency, AC signal amplitude, and DC signal setting.

- Command** DC signal. AC signal frequency and AC signal amplitude are void.  
**[SOURCE:]APPLY:DC** <freq\_numeric>,<ac\_numeric>,<dc\_numeric>
- Command** AC signal (sine wave)  
**[SOURCE:]APPLY:SIN**  
 <freq\_numeric>[,<ac\_numeric>[,<dc\_numeric>]]
- Command** AC signal (square wave)  
**[SOURCE:]APPLY:SQU**  
 <freq\_numeric>[,<ac\_numeric>[,<dc\_numeric>]]
- Command** AC signal (triangular wave)  
**[SOURCE:]APPLY:TRI**  
 <freq\_numeric>[,<ac\_numeric>[,<dc\_numeric>]]
- Command** Arbitrary waveform (USER{[1]|2|...|16})  
**[SOURCE:]APPLY:USER{[1]|2|...|16}** <freq\_numeric>[,<ac\_numeric>[,<dc\_numeric>]]
- Command** **[SOURCE:]APPLY?**

Parameter <freq\_numeric> AC signal frequency

Value: PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5:

CV mode +1.0E-2 Hz to +1.0E+05 Hz (AC signal)

CC mode +1.0E-2 Hz to +1.0E+05 Hz (AC signal)

PBZ20-20A:

CV mode +1.0E-2 Hz to +2.0E+05 Hz (AC signal)

CC mode +1.0E-2 Hz to +2.0E+05 Hz (AC signal)

Unit: HZ

Parameter <ac\_numeric> AC signal amplitude

Value: 0 % to 105 % of the rated output (AC signal)

Unit: VPP or APP

Parameter <dc\_numeric> DC signal setting

Value: 0 % to 105 % of the rated output (DC signal)

Unit: V or A

When you are setting the DC signal, frequency and amplitude settings for the AC signal are invalid.

- Response** Returns the signal type (DC, SIN, SQU, TRI, USER{1|2|...|16}), AC signal frequency, AC signal amplitude, and DC signal setting in the following order in response to APPLY?.  
 <character>,<NR3>,<NR3>,<NR3>

## Amplifier Responses

### CURR:RESP VOLT:RESP

\* RST \* RCL

Sets the amplifier's response. Set this separately for CV mode and CC mode.

**Command** [SOURce:]CURRent:RESPOuse {<numeric>|MIN|MAX}  
[SOURce:]VOLTagE:RESPOuse {<numeric>|MIN|MAX}

**Command** [SOURce:]CURRent:RESPOuse? [{MIN|MAX}]  
[SOURce:]VOLTagE:RESPOuse? [{MIN|MAX}]

Parameter Value: CC mode (PBZ20-20/ PBZ60-6.7/ PBZ80-5)  
35 US 35  $\mu$ s (default)  
100 US 100  $\mu$ s  
350 US 350  $\mu$ s  
1 MS 1 ms

Parameter Value: CC mode (PBZ40-10)  
70 US 70  $\mu$ s (default)  
100 US 100  $\mu$ s  
350 US 350  $\mu$ s  
1 MS 1 ms

Parameter Value: CV mode (PBZ models)  
3.5 US 3.5  $\mu$ s (default)  
10 US 10  $\mu$ s  
35 US 35  $\mu$ s  
100 US 100  $\mu$ s

Unit: US and MS

**Response** Returns the response in <NR3> format in response to the CURR:RESP? and VOLT:RESP? queries.

## Output, Trigger Signals, and Screen Contrast

### Turning Output On and Off

#### OUTP

\* RST \* RCL

Turns output on and off.

**Command** OUTPut[:STATe][:IMMediate] {ON|OFF|1|0}  
OUTPut[:STATe][:IMMediate]?

Parameter Value: ON (1) Turns output on  
OFF (0) Turns output off (default)

**Response** Returns whether output is on or off in <NR1> format in response to the OUTP? query.

## OUTP:TRIG

Turns output on and off when software triggers are received.

**Command** `OUTPut[:STATe]:TRIGgered {ON|OFF|1|0}`  
`OUTPut[:STATe]:TRIGgered?`

Parameter Value: ON (1) Turns output on  
OFF (0) Turns output off

**Response** Returns whether output is on or off in <NR1> format in response to the OUTP:TRIG? query.

## OUTP:PON:STAT

\* RST \* RCL

Sets whether output turns on or off when the PBZ turns on.

**Command** `OUTPut:PON:STATe {RST|AUTO}`  
`OUTPut:PON:STATe?`

Parameter Value: RST Output is off when the PBZ turns on (default).  
AUTO Output turns on when the PBZ turns on.

**Response** Returns whether output turns on or off when the PBZ turns on in <character> format in response to the OUTP:PON:STAT? query.

## OUTP:EXT

\* RST \* RCL

Sets the polarity of the external control signal that is used to turn output on.

**Command** `OUTPut:EXTernal {NORMal|INVerted}`  
`OUTPut:EXTernal?`

Parameter Value: NORMal A high level signal turns the output on (default).  
INVerted A low level signal turns the output on.

**Response** Returns the polarity of the external control signal in <character> format in response to the OUTP:EXT? query.

## Option Output

## OUTP:PORT

\* RST \* RCL

Turns signal output from the J1 connector and option terminal on and off.

**Command** `OUTPut:PORT {ON|OFF|1|0}`  
`OUTPut:PORT?`

Parameter Value: ON (1) Generate signal output  
OFF (0) Do not generate signal output (default)

**Response** Returns whether output is on or off in <NR1> format in response to the OUTP:PORT? query.

## Trigger Signals (TRIG IN and TRIG OUT)

### OUTP:TRIG:STAT

\* RST \* RCL

Turns trigger signal output (TRIG OUT) on and off. When trigger signal output setting is on, the TRIG OUT terminal can transmit MARKER or SYNC signals.

**Command** `OUTPut:TRIGger:STATe {ON|OFF|1|0}`  
`OUTPut:TRIGger:STATe?`

Parameter	Value:	ON (1)	Generate trigger signal output
		OFF (0)	Do not generate trigger signal output (default)

**Response** Returns whether trigger signal output (TRIG OUT) is on or off in <NR1> format in response to the OUTP:TRIG:STAT? query.

### OUTP:TRIG:POL

\* RST \* RCL

Sets the polarity of the trigger signal output (TRIG OUT).

**Command** `OUTPut:TRIGger:POLarity {POSitive|NEGative}`  
`OUTPut:TRIGger:POLarity?`

Parameter	Value:	POSitive	Rising (default)
		NEGative	Falling

**Response** Returns the polarity of the trigger signal output (TRIG OUT) in <character> format in response to the OUTP:TRIG:POL? query.

### INP:TRIG:POL

\* RST \* RCL

Sets the polarity of the trigger signal input (TRIG IN).

**Command** `INPut:TRIGger:POLarity {POSitive|NEGative}`  
`INPut:TRIGger:POLarity?`

Parameter	Value:	POSitive	High level (default)
		NEGative	Low level

**Response** Returns the polarity of the trigger signal input (TRIG IN) in <character> format in response to the INP:TRIG:POL? query.

## Screen Contrast

### DISP:CONT

\* RST \* RCL

Sets the screen contrast.

**Command** `DISPlay:CONTRast {<numeric>|MIN|MAX}`  
`DISPlay:CONTRast? [{MIN|MAX}]`

Parameter	Value:	0.0 to 1.0 (The default value is 0.7.)	
	Unit:	None	

**Response** Returns the screen contrast in <NR3> format in response to the DISP:CONT? query.

# 6

## Protection Features Commands

### Mode

#### CURR:PROT:STAT VOLT:PROT:STAT

\* RST \* RCL

Sets the mode (I/V-LIMIT or OCP/OVP) of the overvoltage and overcurrent protection features.

**Command** [SOURCE:]CURRENT:PROTECTION:STATE {LIMIT|TRIP}  
[SOURCE:]VOLTAGE:PROTECTION:STATE {LIMIT|TRIP}

**Command** [SOURCE:]CURRENT:PROTECTION:STATE?  
[SOURCE:]VOLTAGE:PROTECTION:STATE?

Parameter Value:   LIMIT           Voltage or current limit is used.(default)  
                  TRIP            OVP/OCP is used.

Returns the mode in <character> format in response to the {CURR|VOLT}:PROT:STAT? query.

### Activation points (OCP/OVP)

Use SYST:CONF:BTR:PROT to set the protection operation to perform (power off or output off) when OCP or OVP is activated.

#### CURR:PROT:OVER VOLT:PROT:OVER

\* RST \* RCL

Sets the value of the positive OCP or OVP trip point for the current or voltage.

**Command** [SOURCE:]CURRENT:PROTECTION[:LEVEL]:OVER {<numeric>|MIN|MAX}  
[SOURCE:]VOLTAGE:PROTECTION[:LEVEL]:OVER {<numeric>|MIN|MAX}

**Command** [SOURCE:]CURRENT:PROTECTION[:LEVEL]:OVER? [{MIN|MAX}]  
[SOURCE:]VOLTAGE:PROTECTION[:LEVEL]:OVER? [{MIN|MAX}]

Parameter Value:   +1 % to +110 % of the rated value (The default value is +110 % of the rated value.)  
Unit:            A or V

**Response** Returns the setting in <NR3> format in response to the {CURR|VOLT}:PROT:OVER? query.

## **CURR:PROT:UND** **VOLT:PROT:UND**

\* RST \* RCL

Sets the value of the negative OCP or OVP trip point for the current or voltage.

**Command** [SOURCE:]CURRENT:PROTECTION[:LEVEL]:UNDER {<numeric>|MIN|MAX}  
[SOURCE:]VOLTAGE:PROTECTION[:LEVEL]:UNDER {<numeric>|MIN|MAX}

**Command** [SOURCE:]CURRENT:PROTECTION[:LEVEL]:UNDER? [{MIN|MAX}]  
[SOURCE:]VOLTAGE:PROTECTION[:LEVEL]:UNDER? [{MIN|MAX}]

Parameter Value: -110 % to -1 % of the rated value (The default value is -110 % of the rated value.)  
Unit: A or V

**Response** Returns the setting in <NR3> format in response to the {CURR|VOLT}:PROT:UND? query.





## Activation points (I.LIM/ V.LIM)

### CURR:LIM:UPP VOLT:LIM:UPP

\* RST \* RCL

Sets the value of the positive I.LIM or V.LIM trip point for the current or voltage.

**Command** [SOURCE:]CURRENT:LIMit[:LEVel]:UPPer {<numeric>|MIN|MAX}  
[SOURCE:]VOLTage:LIMit[:LEVel]:UPPer {<numeric>|MIN|MAX}

**Command** [SOURCE:]CURRENT:LIMit[:LEVel]:UPPer? [{MIN|MAX}]  
[SOURCE:]VOLTage:LIMit[:LEVel]:UPPer? [{MIN|MAX}]

Parameter Value: +1 % of rated value  $\leq$  +I.LIM  $\leq$  +110 % of the rated value  
-110 % of rated value  $\leq$  -V.LIM  $\leq$  +V.LIM  $\leq$  +110 % of the rated value  
(The default value is +110 % of the rated value.)

Unit: A or V

**Response** Returns the setting in <NR3> format in response to the {CURR|VOLT}:LIM:UPP? query.

### CURR:LIM:LOW VOLT:LIM:LOW

\* RST \* RCL

Sets the value of the negative I.LIM or V.LIM trip point for the current or voltage.

**Command** [SOURCE:]CURRENT:LIMit[:LEVel]:LOWer {<numeric>|MIN|MAX}  
[SOURCE:]VOLTage:LIMit[:LEVel]:LOWer {<numeric>|MIN|MAX}

**Command** [SOURCE:]CURRENT:LIMit[:LEVel]:LOWer? [{MIN|MAX}]  
[SOURCE:]VOLTage:LIMit[:LEVel]:LOWer? [{MIN|MAX}]

Parameter Value: -110 % of rated value  $\leq$  -I.LIM  $\leq$  -1 % of the rated value  
-110 % of rated value  $\leq$  -V.LIM  $\leq$  +V.LIM  $\leq$  +110 % of the rated value  
(The default value is -110 % of the rated value.)

Unit: A or V

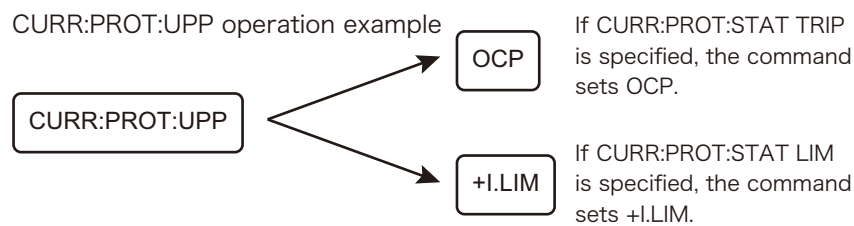
**Response** Returns the setting in <NR3> format in response to the {CURR|VOLT}:LIM:LOW? query.

## Mode and activation points (legacy command)

The following commands were used up to firmware version 2.19. When creating a new program, use the commands listed under the aforementioned OVP or OCP trip point or V.LIM or I.LIM trip point.

The CURR:PROT:UPP and CURR:PROT:LOW commands apply to the protection mode (OCP or current limit) selected with CURR:PROT:STAT. The VOLT:PROT:UPP and VOLT:PROT:LOW commands apply to the protection mode (OVP or voltage limit) selected with VOLT:PROT:STAT. To use a program that you were using for firmware version 2.19 or earlier, check that the protection mode is set before the trip points are set.

Use SYST:CONF:BTR:PROT to set the protection operation to perform (power off or output off) when OVP or OCP is activated.



### CURR:PROT:UPP

\* RST \* RCL

If CURR:PROT:STAT LIM is specified, this command sets the value of the positive I.LIM trip point.  
 If CURR:PROT:STAT TRIP is specified, this command sets the value of the positive OCP trip point.

**Command** [SOURCE:]CURRENT:PROTECTION[:LEVEL]:UPPER {<numeric>|MIN|MAX}  
 [SOURCE:]CURRENT:PROTECTION[:LEVEL]:UPPER? [{MIN|MAX}]

Parameter Value: +1 % to +110 % of the rated value (The default value is +110 % of the rated value.)  
 Unit: A

**Response** Returns the setting in <NR3> format in response to the CURR:PROT:UPP? query.

### VOLT:PROT:UPP

\* RST \* RCL

If VOLT:PROT:STAT LIM is specified, this command sets the value of the positive V.LIM trip point.  
 If VOLT:PROT:STAT TRIP is specified, this command sets the value of the positive OVP trip point.

**Command** [SOURCE:]VOLTAGE:PROTECTION[:LEVEL]:UPPER {<numeric>|MIN|MAX}  
 [SOURCE:]VOLTAGE:PROTECTION[:LEVEL]:UPPER? [{MIN|MAX}]

Parameter Value: -110 % of rated value ≤ -V.LIM ≤ +V.LIM ≤ +110 % of the rated value%  
 (when VOLT:PROT:STAT LIM is specified)  
 +1 % to +110 % of the rated value (when VOLT:PROT:STAT TRIP is specified)  
 (The default value is +110 % of the rated value.)  
 Unit: V

**Response** Returns the setting in <NR3> format in response to the VOLT:PROT:UPP? query.

**CURR:PROT:LOW**

\* RST \* RCL

If CURR:PROT:STAT LIM is specified, this command sets the value of the negative I.LIM trip point.

If CURR:PROT:STAT TRIP is specified, this command sets the value of the negative OCP trip point.

**Command** [SOURCE:]CURRENT:PROTECTION[:LEVEL]:LOWER {<numeric>|MIN|MAX}  
[SOURCE:]CURRENT:PROTECTION[:LEVEL]:LOWER? [{MIN|MAX}]

Parameter Value: -110 % to -1 % of the rated value (The default value is -110 % of the rated value.)  
Unit: A

**Response** Returns the setting in <NR3> format in response to the CURR:PROT:LOW? query.

**VOLT:PROT:LOW**

\* RST \* RCL

If VOLT:PROT:STAT LIM is specified, this command sets the value of the negative V.LIM trip point.

If VOLT:PROT:STAT TRIP is specified, this command sets the value of the negative OVP trip point.

**Command** [SOURCE:]VOLTAGE:PROTECTION[:LEVEL]:LOWER {<numeric>|MIN|MAX}  
[SOURCE:]VOLTAGE:PROTECTION[:LEVEL]:LOWER? [{MIN|MAX}]

Parameter Value: -110 % of rated value ≤ -V.LIM ≤ +V.LIM ≤ +110 % of the rated value%  
(when VOLT:PROT:STAT LIM is specified)  
-110 % to -1 % of the rated value (when VOLT:PROT:STAT TRIP is specified)  
(The default value is -110 % of the rated value.)  
Unit: V

**Response** Returns the setting in <NR3> format in response to the VOLT:PROT:LOW? query.

**Clearing Alarms****OUTP:PROT:CLEAR**

Clears alarms.

**Command** OUTPut:PROTECTION:CLEAR

# 7

## Measurement Function Commands

### SENS:FUNC

\* RST \* RCL

Sets the measurement function.

**Command** `SENSe:FUNCTION {DC|AC|DCAC|PEAK}`  
`SENSe:FUNCTION?`

Parameter	Value:	DC	DC (default)
		AC	AC RMS
		DCAC	DC + AC RMS
		PEAK	DC + the peak AC value

**Response** Returns the measurement function in <character> format in response to the SENS:FUNC? query.

### SENS:APER

\* RST \* RCL

Sets the measurement time.

**Command** `SENSe:APERTure {<numeric>|MIN|MAX}`  
`SENSe:APERTure? [{MIN|MAX}]`

Parameter	Value:	0.0001 to 3600 (The default value is 0.1.)
	Unit:	S

**Response** Returns the measurement time in <NR3> format in response to the SENS:APER? query.

### SENS:TRIG:SOUR

\* RST \* RCL

Sets the measurement start trigger source.

**Command** `SENSe:TRIGger:SOURce {AUTO|INTernal|EXTPOS|EXTNEG}`  
`SENSe:TRIGger:SOURce?`

Parameter	Value:	AUTO	Start measuring automatically (default).
		INTernal	Start measuring whenever a setting is changed.
		EXTPOS	Start measuring when a rising edge is applied to the TRIG IN terminal.
		EXTNEG	Start measuring when a falling edge is applied to the TRIG IN terminal.

**Response** Returns the measurement start trigger source in <character> format in response to the SENS:TRIG:SOUR? query.

**SENS:TRIG:DEL**

\* RST

\* RCL

Sets the trigger delay.

**Command** `SENSe:TRIGger:DELay {<numeric>|MIN|MAX}`  
`SENSe:TRIGger:DELay? [{MIN|MAX}]`

Parameter Value: 0.00000 to 3600 (The default value is 0.00000)  
 Unit: S

**Response** Returns the delay time in <NR3> format in response to the SENS:TRIG:DEL? query.

## Trigger Function


The PBZ has the following four trigger functions.

- **SEquence1 (TRANsient)**  
Specify FUNCtion, FREQuency, CURRent (DC and AC), and VOLTage (DC and AC) settings in advance, and use trigger timing to set them.
- **SEquence2 (OUTPut)**  
Specify OUTPUT ON/OFF settings in advance, and use trigger timing to set them.
- **SEquence3 (ACQuire)**  
Use trigger timing to start measuring or query the measured values.
- **SEquence4 (PROGram)**  
Use trigger timing to carry out PROGram or SCRipt execution. You have to specify the execution conditions in advance.


### SEquence1 (TRANsient)

The following table shows items that you can use trigger timing to set (<transient\_item>). You can set the items immediately after a trigger occurs or wait for a software trigger and then set the items.


#### CV mode

<transient_item>	Description	 Page
[SOURce:]VOLTage[:LEVel]:TRIGgered[:AMPLitude]	DC signal voltage	26
[SOURce:]VOLTage:AC:TRIGgered[:AMPLitude]	AC signal voltage	31

#### CC mode

<transient_item>	Description	 Page
[SOURce:]CURRent[:LEVel]:TRIGgered[:AMPLitude]	DC signal current	26
[SOURce:]CURRent:AC:TRIGgered[:AMPLitude]	AC signal current	31

**CV mode and CC mode**

<transient_item>	Description	 Page
[SOURce:]AC:STAT:TRIGgered	AC signal: on/off	27
[SOURce:]FUNCTion[:SHAPE]:TRIGgered	AC signal waveform	28
[SOURce:]FREQuency:TRIGgered	AC signal frequency	31

**INIT  
INIT:NAME TRAN**

Starts the TRANSient trigger function.

**Command** INITiate[:IMMediate][:SEQuence[1]]  
INITiate[:IMMediate]:NAME TRANsient

**TRIG  
TRIG:TRAN**

Executes a software trigger for the SEQuence1 (TRANSient) group.

**Command** TRIGger[:SEQuence[1]][:IMMediate]  
TRIGger:TRANsient[:IMMediate]

**TRIG:SOUR  
TRIG:TRAN:SOUR**

\* RST \* RCL

Sets an applicable trigger source for the SEQuence1 (TRANSient) group.

**Command** TRIGger[:SEQuence[1]]:SOURce {BUS|IMMediate}  
TRIGger[:SEQuence[1]]:SOURce?

**Command** TRIGger:TRANsient:SOURce {BUS|IMMediate}  
TRIGger:TRANsient:SOURce?

Parameter Value: BUS Software trigger  
Wait for a \*TRG or IEEE 488.1 get (Group Execute Trigger) command, and then begin execution.  
IMMediate Immediately begin execution (default)

**Response** Returns the current trigger source setting in <character> format.

## SEQuence2 (OUTPut)

The following table shows the item that you can use trigger timing to set (<output\_item>). You can set the item immediately after a trigger occurs or wait for a software trigger and then set the item.

<output_item>	Description	See Page
OUTput[:STATE]:TRIGgered	Output: on/off	37

## INIT:SEQ2 INIT:NAME OUTP

Starts the OUTPut trigger function.

**Command** INITiate[:IMMediate]:SEQuence2  
INITiate[:IMMediate]:NAME OUTPut

## TRIG:SEQ2 TRIG:OUTP

Executes a software trigger for the SEQuence2 (OUTPut) group.

**Command** TRIGger:SEQuence2[:IMMediate]  
TRIGger:OUTPut[:IMMediate]

## TRIG:SEQ2:SOUR TRIG:OUTP:SOUR

\*RST \*RCL

Sets an applicable trigger source for the SEQuence2 (OUTPut) group.

**Command** TRIGger:SEQuence2:SOURce {BUS|IMMediate}  
TRIGger:SEQuence2:SOURce?

**Command** TRIGger:OUTPut:SOURce {BUS|IMMediate}  
TRIGger:OUTPut:SOURce?

Parameter Value:	BUS	Software trigger Wait for a *TRG or IEEE 488.1 get (Group Execute Trigger) command, and then begin execution.
	IMMediate	Immediately begin execution (default)

**Response** Returns the current trigger source setting in <character> format.

## TRIG:SEQ2:SST:RISE

### TRIG:OUTP:SST:RISE \* RST \* RCL

Sets the soft start time for the SEquence2 (OUTPut) group.

**Command** TRIGger:SEquence2:SStart:RISE {<numeric>|MIN|MAX}  
TRIGger:SEquence2:SStart:RISE? [{MIN|MAX}]

**Command** TRIGger:OUTPut:SStart:RISE {<numeric>|MIN|MAX}  
TRIGger:OUTPut:SStart:RISE? [{MIN|MAX}]

Parameter Value: 0.0001 to 1000 (The default value is 0.0000.)  
Unit: S

**Response** Returns the soft start time in <NR3> format.

## TRIG:SEQ2:SST:FALL

### TRIG:OUTP:SST:FALL

\* RST \* RCL

Sets the soft stop time for the SEquence2 (OUTPut) group.

**Command** TRIGger:SEquence2:SStart:FALL {<numeric>|MIN|MAX}  
TRIGger:SEquence2:SStart:FALL? [{MIN|MAX}]

**Command** TRIGger:OUTPut:SStart:FALL {<numeric>|MIN|MAX}  
TRIGger:OUTPut:SStart:FALL? [{MIN|MAX}]

Parameter Value: 0.0001 to 1000 (The default value is 0.0000.)  
Unit: S


**Response** Returns the soft stop time in <NR3> format.





## SEquence3 (ACQuire)

The following table shows items that you can use trigger timing to measure (<meas\_item>). You can measure the items immediately after a trigger occurs or wait for a software trigger and then measure the items.

<meas_item>	Measurement description	 Page
VOLTage[:DC]	DC voltage	26
VOLTage:AC	AC voltage	30
VOLTage:{MINimum   MAXimum}	Peak voltage, minimum, and maximum	–
CURRent[:DC]	DC current	26
CURRent:AC	AC current	30
CURRent:{MINimum   MAXimum}	Peak current, minimum, and maximum	–

The following table shows the different measurement operations that are available (FETC, READ, and MEAS). READ and MEASure operate exactly the same, as they are aliases.

Measurement operation and command syntax	Measurement operation description
:FETCh[:SCALar]:<meas_item>?	Query the measured value without starting a new measurement.
:READ[:SCALar]:<meas_item>? :MEASure[:SCALar]:<meas_item>?	Start a new measurement, and query the measured value.

### FETC:<meas\_item>

Queries the measured value that is specified by <meas\_item> without starting a new measurement.

**Command** FETCh[:SCALar]:<meas\_item>?

**Response** Returns the measured value in <NR3> (number of digits to the right of the decimal is 5) format.

### READ:<meas\_item> MEAS:<meas\_item>

Starts a new measurement, and queries the measured value that is specified by <meas\_item>.

**Command** READ[:SCALar]:<meas\_item>?  
MEASure[:SCALar]:<meas\_item>?

**Response** Returns the measured value in <NR3> (number of digits to the right of the decimal is 5) format.

## INIT:SEQ3 INIT:NAME ACQ

Starts the ACQUIRE trigger function. When an INIT command is sent, the measured value that is currently saved is deleted and invalidated. If you send a FETC? query immediately after you send an INIT command, the PBZ returns the measured value after it completes the measurement.

**Command** INITiate[:IMMEDIATE]:SEQUENCE3  
INITiate[:IMMEDIATE]:NAME ACQUIRE

## TRIG:SEQ3 TRIG:ACQ

Executes a software trigger for the SEQUENCE3 (ACQUIRE) group.

**Command** TRIGGER:SEQUENCE3[:IMMEDIATE]  
TRIGGER:ACQUIRE[:IMMEDIATE]

## TRIG:SEQ3:SOUR TRIG:ACQ:SOUR

\* RST \* RCL

Sets an applicable trigger source for the SEQUENCE3 (ACQUIRE) group.

**Command** TRIGGER:SEQUENCE3:SOURCE {BUS|IMMEDIATE}  
TRIGGER:SEQUENCE3:SOURCE?

**Command** TRIGGER:ACQUIRE:SOURCE {BUS|IMMEDIATE}  
TRIGGER:ACQUIRE:SOURCE?

Parameter	Value:	BUS	Software trigger Wait for a *TRG or IEEE 488.1 get (Group Execute Trigger) command, and then begin measuring.
		IMMEDIATE	Immediately begin measuring (default)

**Response** Returns the current trigger source setting in <character> format.

## SEQuence4 (PROGram)

This trigger function is used to carry out PROGram or SCRipt execution. You have to specify the execution conditions in advance. You can execute the program or script immediately after a trigger occurs or wait for a software trigger and then carry out execution.

### INIT:SEQ4 INIT:NAME PROG

Starts the PROGram trigger function.

**Command** `INITiate[:IMMediate]:SEQuence4`  
`INITiate[:IMMediate]:NAME PROGRAM`

### TRIG:SEQ4 TRIG:PROG

Executes a software trigger for the SEQuence4 (PROgram) group.

**Command** `TRIGger:SEQuence4[:IMMediate]`  
`TRIGger:PROGram[:IMMediate]`

### TRIG:SEQ4:SOUR TRIG:PROG:SOUR

\* RST \* RCL

Sets an applicable trigger source for the SEQuence4 (PROgram) group.

**Command** `TRIGger:SEQuence4:SOURce {BUS|IMMediate|EXTernal}`  
`TRIGger:SEQuence4:SOURce?`

**Command** `TRIGger:PROGram:SOURce {BUS|IMMediate}`  
`TRIGger:PROGram:SOURce?`

Parameter	Value:	BUS	Software trigger Wait for a *TRG or IEEE 488.1 get (Group Execute Trigger) command, and then begin execution.
		IMMediate	Immediately begin execution (default)
		EXTernal	Begin execution when an edge is applied to the TRIG IN terminal

**Response** Returns the current trigger source setting in <character> format.

## SEquence1 to SEquence4 Shared Command

### ABOR

Aborts measurement.

The PBZ's trigger state immediately after it turns on is the same as its trigger state after it receives an ABOR command.

If you send an ABOR command while the PBZ is executing measurements, the measured data is discarded.

If you send an ABOR command without first sending an INIT command and if the measurement data that is held in the PBZ is valid, the measured data is not discarded.

**Command** `ABORt`

# 8

# Memory Function Commands

## Preset Memory

### MEM:PRES:RCL

Loads preset memory entries.

**Command** `MEMory:PRESet:RCL {1|2|3}`

Parameter	Value:	1	Preset memory A
		2	Preset memory B
		3	Preset memory C

### MEM:PRES:SAV

Saves the current settings to preset memory.

**Command** `MEMory:PRESet:SAV {1|2|3}`

Parameter	Value:	1	Preset memory A
		2	Preset memory B
		3	Preset memory C

## Setup Memory

### MEM:SET:RCL

Loads the setup memory entry that you specify by its memory number.

**Command** `MEMory:SETup:RCL {0|1|2|3|4|5|6|7|8|9}`

Parameter	Value:	0 to 9	Memory number
-----------	--------	--------	---------------

## MEM:SET:SAV

Saves the current settings to the setup memory entry that you specify by its memory number.

**Command** `MEMory:SETup:SAV {0|1|2|3|4|5|6|7|8|9}`

Parameter Value:

Parameter Value: 0 to 9          Memory number

## MEM:SET:TITL

Specify the setup memory number and sets the name to a string of 15 characters in length. A space character(0x20) is inserted automatically in case of a string of less than 15 characters in length.

**Command** `MEMory:SETup:TITLe {0|1|2|3|4|5|6|7|8|9},<"string">`  
`MEMory:SETup:TITLe? {0|1|2|3|4|5|6|7|8|9}`

Parameter Value: 0 to 9          Memory number

Parameter Value: <"string">      Character string

**Response** Returns the name of the setup memory entry in <"string"> format in response to the `MEMory:SETup:TITLe? {0|1|2|3|4|5|6|7|8|9}` query.



# 9

# Sequence Function Commands

## Editing Programs and Steps

### Selecting Programs

#### PROG:NAME

\* RST \* RCL

Selects the program that corresponds to the specified program number, or clears the current selection.

**Command** `PROG:NAME <"string">`  
`PROG:NAME?`

Parameter Value: "1" to "16" Program number  
" " Blank string (clears the current selection; default)

**Response** Returns the name of the program in <"string"> format in response to the PROG:NAME? query.

### Editing Programs

#### PROG:EDIT:ADD

Adds the specified number of steps to the selected program. Steps are added after the program's final step.

**Command** `PROG:EDIT:ADD <NRf>`

Parameter Value: 1 to 1024 Number of steps to add

#### PROG:EDIT:COUN

Queries the number of steps in the selected program.

**Command** `PROG:EDIT:COUN?`

**Response** Returns the number of steps in the program in <NR1> format in response to the PROG:EDIT:COUN? query.

## PROG:EDIT:COUN:FREE

Queries the number of steps that can be added to the selected program.

**Command** `PROG:EDIT:COUNT:FREE?`

**Response** Returns the number of steps that can be added to the program in <NR1> format in response to the PROG:EDIT:COUN:FREE? query.

## PROG:EDIT:DEL

Deletes all of the selected program's steps.

**Command** `PROG:EDIT:DELETE`

## PROG:EDIT:TITL

Sets the name of the selected program to a string of 15 characters in length. A space character(0x20) is inserted automatically in case of a string of less than 15 characters in length.

**Command** `PROG:EDIT:TITLE <"string">`  
`PROG:EDIT:TITLE?`

Parameter Value: "string"  
Example "My Program"

**Response** Returns the selected program's name in <"string"> format.

## PROG:EDIT:FUNC:MODE

Sets the selected program's CV/CC mode setting.

**Command** `PROG:EDIT:FUNCTION:MODE {CC|CV}`  
`PROG:EDIT:FUNCTION:MODE?`

Parameter Value: CC CC mode  
CV CV mode

**Response** Returns the setting in <character> format in response to the PROG:EDIT:FUNC:MODE? query.

## PROG:EDIT:FUNC:POL

Sets the selected program's bipolar/unipolar mode setting.

**Command** `PROG:EDIT:FUNCTION:POLARITY {BIPolar|UNIPolar}`  
`PROG:EDIT:FUNCTION:POLARITY?`

Parameter Value: BIPolar Bipolar mode (default)  
UNIPolar Unipolar mode

**Response** Returns the setting in <character> format in response to the PROG:EDIT:FUNC:POL? query.



## PROG:EDIT:LOOP

Sets the number of times that the selected program will repeat.

**Command** `PROG:EDIT:LOOP <NRf>`  
`PROG:EDIT:LOOP?`

Parameter Value: 1 to 10000  
10001 (infinity)

**Response** Returns the number of times that the program will repeat in <NR3> format in response to the PROG:EDIT:LOOP? query.

## Selecting Steps

### PROG:EDIT:STEP:SEL

Selects the selected program's step that corresponds to the specified step number.

**Command** `PROG:EDIT:STEP:SElect <NRf>`  
`PROG:EDIT:STEP:SElect?`

Parameter Value: 1 to 1024 Step number: The maximum value is the number of steps that have been added to the program.

**Response** Returns the step number in <NR1> format in response to the PROG:EDIT:STEP:SEL? query. Returns <-1> if there are no steps in the selected program.

## Editing Steps (DC Signal Settings)

### PROG:EDIT:STEP:CURR PROG:EDIT:STEP:VOLT

Sets the DC signal voltage, current, and how the signal will change over time for the selected step. If you specify IMMEDIATE, the DC signal voltage and current are set to the specified values immediately at the start of the step. If you specify RAMP, the DC signal voltage and current increase or decrease gradually from the start of the step and are set to the specified values at the end of the step.

If the step time exceeds 1000 seconds, the DC signal level after 1000 seconds is the same as the DC signal stop setting (STOP). If you want to perform a ramp transition that takes longer than 1000 seconds, you have to combine multiple steps.

**Command** `PROG:EDIT:STEP:VOLTage[:LEVel]`  
`{<numeric>|MIN|MAX} [, <character>]`  
`PROG:EDIT:STEP:VOLTage[:LEVel]? [{MIN|MAX}]`

**Command** `PROG:EDIT:STEP:CURREnt[:LEVel]`  
`{<numeric>|MIN|MAX} [, <character>]`  
`PROG:EDIT:STEP:CURREnt[:LEVel]? [{MIN|MAX}]`

Parameter Value: 0 % to 105 % of the rated output  
 Unit: V or A

Parameter Value: IMMEDIATE Change levels immediately  
 RAMP Change levels gradually  
 The default is IMMEDIATE.

**Response** Returns the settings in <NR3>, <character> format in response to the PROG:EDIT:STEP:VOLT? or PROG:EDIT:STEP:CURR? query.

### PROG:EDIT:STEP:CURR:RAMP PROG:EDIT:STEP:VOLT:RAMP

Sets the value of the DC signal current and voltage at the start of the step when the selected step is set to RAMP (change gradually).

**Command** `PROG:EDIT:STEP:VOLTage:RAMP {<numeric>|MIN|MAX}`  
`PROG:EDIT:STEP:VOLTage:RAMP? [{MIN|MAX}]`

**Command** `PROG:EDIT:STEP:CURREnt:RAMP {<numeric>|MIN|MAX}`  
`PROG:EDIT:STEP:CURREnt:RAMP? [{MIN|MAX}]`

Parameter Value: 0 % to 105 % of the rated output  
 Unit: V or A

**Response** Returns the setting in <NR3> format in response to the PROG:EDIT:STEP:VOLT:RAMP? or PROG:EDIT:STEP:CURR:RAMP? query.

## Editing Steps (AC Signal Settings)

### PROG:EDIT:STEP:FUNC

Sets the AC signal waveform of the selected step. There are three types of fixed waveforms (sine, square, and triangle waves), and you can store up to 16 user-defined arbitrary waveforms.

**Command** `PROG:EDIT:STEP:FUNCTION[:SHAPE]`  
`{SINusoid|SQUare|TRIangle|USER[1]..USER16}`  
`PROG:EDIT:STEP:FUNCTION[:SHAPE]?`

Parameter	Value:	SINusoid	Sine wave
		SQUare	Square wave
		TRIangle	Triangle wave
		USER[1] to USER16	User-defined arbitrary waveform

**Response** Returns the AC signal waveform type in <character> format in response to the PROG:EDIT:STEP:FUNC? query.

### PROG:EDIT:STEP:CURR:AC PROG:EDIT:STEP:VOLT:AC

Sets the AC signal voltage, current, and how the signal will change over time for the selected step. If you specify IMMEDIATE, the AC signal voltage and current are set to the specified values immediately at the start of the step. If you specify SWEep, the AC signal voltage and current increase or decrease gradually from the start of the step and are set to the specified values at the end of the step.

**Command** `PROG:EDIT:STEP:VOLTage:AC`  
`{<numeric>|MIN|MAX} [, <character>]`  
`PROG:EDIT:STEP:VOLTage:AC? [{MIN|MAX}]`

**Command** `PROG:EDIT:STEP:CURRent:AC`  
`{<numeric>|MIN|MAX} [, <character>]`  
`PROG:EDIT:STEP:CURRent:AC? [{MIN|MAX}]`

Parameter	Value:	0 % to 105 % of the rated output
	Unit:	VPP or APP

Parameter	Value:	IMMEDIATE	Change levels immediately
		SWEep	Change levels gradually
		The default is IMMEDIATE.	

**Response** Returns the settings in <NR3>, <character> format in response to the PROG:EDIT:STEP:VOLT:AC? or PROG:EDIT:STEP:CURR:AC? query.

## PROG:EDIT:STEP:CURR:AC:SWE PROG:EDIT:STEP:VOLT:AC:SWE

Sets the value of the AC signal current and voltage at the start of the step when the selected step is set to SWEep (change gradually).

If the step time exceeds 1000 seconds, the AC signal amplitude level after 1000 seconds is the same as the stop amplitude setting (STOP). If you want to perform an AC signal amplitude sweep that takes longer than 1000 seconds, you have to combine multiple steps.

**Command** `PROG:EDIT:STEP:VOLTage:AC:SWEep {<numeric>|MIN|MAX}`  
`PROG:EDIT:STEP:VOLTage:AC:SWEep? [ {MIN|MAX} ]`

**Command** `PROG:EDIT:STEP:CURRent:AC:SWEep {<numeric>|MIN|MAX}`  
`PROG:EDIT:STEP:CURRent:AC:SWEep? [ {MIN|MAX} ]`

Parameter Value: 0 % to 105 % of the rated output  
Unit: VPP or APP

**Response** Returns the setting in <NR3> format in response to the PROG:EDIT:STEP:VOLT:AC:SWE? or PROG:EDIT:STEP:CURR:AC:SWE? query.

## PROG:EDIT:STEP:AC:STAT

Sets the selected step's AC signal on/off state.

**Command** `PROG:EDIT:STEP:AC:STATe {ON|OFF|1|0}`  
`PROG:EDIT:STEP:AC:STATe?`

Parameter Value: ON (1) On  
OFF (0) Off

**Response** Returns the selected step's AC signal on/off state in <NR1> format in response to the PROG:EDIT:STEP:AC:STAT? query.

**PROG:EDIT:STEP:FREQ**

Sets the AC signal frequency and how the frequency will change over time for the selected step. If you specify IMMEDIATE, the AC signal frequency is set to the specified value immediately at the start of the step. If you specify SWEep, the AC signal frequency increases or decreases gradually from the start of the step and it is set to the specified value at the end of the step.

**Command** `PROG:EDIT:STEP:FREQuency {<numeric>|MIN|MAX} [, <character>]`  
`PROG:EDIT:STEP:FREQuency? [ {MIN|MAX} ]`

Parameter Value: PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5:  
+1.0E-02 Hz to +1.0E+05 Hz  
Maximum CV mode: +1.0E+05 (100 kHz)  
CC mode: +1.0E+05 (100 kHz)

PBZ20-20A:  
+1.0E-02 Hz to +2.0E+05 Hz  
Maximum CV mode: +2.0E+05 (200 kHz)  
CC mode: +2.0E+05 (200 kHz)

Unit: HZ

Parameter Value: IMMEDIATE Change levels immediately  
SWEep Change levels gradually  
The default is IMMEDIATE.

**Response** Returns the settings in <NR3>, <character> format in response to the PROG:EDIT:STEP:FREQ? query.

**PROG:EDIT:STEP:FREQ:SWE**

Sets the AC signal frequency sweep mode and the frequency at the start of the step for the selected step.

If the step time exceeds 1000 seconds, the frequency sweep from the specified start frequency to the specified stop frequency is repeated after 1000 seconds. If you want to perform an AC signal frequency sweep that takes longer than 1000 seconds, you have to combine multiple steps.

**Command** `PROG:EDIT:STEP:FREQuency:SWEep`  
`[ <character> ] , { <numeric> | MIN | MAX }`  
`PROG:EDIT:STEP:FREQuency:SWEep? <character> [ , { MIN | MAX } ]`

Parameter Value: LOG Logarithmic sweep  
LINear Linear sweep

Parameter Value: PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5:  
+1.0E-02 Hz to +1.0E+05 Hz (start frequency)  
Maximum CV mode: +1.0E+05 (100 kHz)  
CC mode: +1.0E+05 (100 kHz)

PBZ20-20A:  
+1.0E-02 Hz to +2.0E+05 Hz (start frequency)  
Maximum CV mode: +2.0E+05 (200 kHz)  
CC mode: +2.0E+05 (200 kHz)

Unit: HZ

**Response** Returns the settings in <character>, <NR3> format in response to the PROG:EDIT:STEP:FREQ:SWE? query.

## PROG:EDIT:STEP:PHAS

Sets AC signal start phase angle and whether to set the start phase angle or not for the selected step.

**Command** `PROG:EDIT:STEP:PHAS {<numeric>|MIN|MAX}, {ON|OFF|1|0}`  
`PROG:EDIT:STEP:PHAS?`

Parameter Value: 0 to 359  
Unit: DEG (degrees)

Parameter Value: ON (1) Set the start phase angle  
OFF (0) Do not set the start phase angle

**Response** Returns the start phase angle setting and whether the start phase angle is set or not in <NR3>,<NR1> format in response to the PROG:EDIT:STEP:PHAS? query.

## PROG:EDIT:STEP:SQU:DCYC

Sets the square wave signal duty cycle for the selected step.

**Command** `PROG:EDIT:STEP:SQUare:DCYCLE {<numeric>|MIN|MAX}`  
`PROG:EDIT:STEP:SQUare:DCYCLE? [ {MIN|MAX} ]`

Parameter Value: 0.1 % to 99.9 % (The default value is 50.0 %.)  
Unit: PCT

**Response** Returns the setting in <NR3> format in response to the PROG:EDIT:STEP:SQU:DCYC? query.

## Editing Steps (Execution Time, Turning Output On and Off, and Trigger I/O)

### PROG:EDIT:STEP:TIME

Sets the execution time for the selected step.

The DC signal ramp and AC signal amplitude sweep both stop after 1000 seconds. The AC signal frequency sweep repeats once every 1000 seconds. If you want to perform these functions for lengths of time exceeding 1000 seconds, you have to combine multiple steps.

**Command** `PROG:EDIT:STEP:TIME {<numeric>|MIN|MAX}`  
`PROG:EDIT:STEP:TIME? [ {MIN|MAX} ]`

Parameter Value: 0.0001 to 3600000  
 Unit: S

**Response** Returns the setting in <NR3> (number of digits to the right of the decimal is 5) format in response to the PROG:EDIT:STEP:TIME? query.

### PROG:EDIT:STEP:STAT

Sets the selected step to turn output on or off and sets the selected step's trigger signal I/O.

**Command** `PROG:EDIT:STEP:STATe`  
`{ON|OFF|1|0}, {ON|OFF|1|0}, {ON|OFF|1|0}`  
`PROG:EDIT:STEP:STATe?`

Parameter Value: ON (1) Turns output on  
 OFF (0) Turns output off

Parameter Value: ON (1) Generate trigger signal output  
 OFF (0) Do not generate trigger signal output

Parameter Value: ON (1) Wait for trigger signal input  
 OFF (0) Ignore trigger signal input

**Response** Returns the settings in <NR1>,<NR1>,<NR1> format in response to the PROG:EDIT:STEP:STAT? query.

## Executing Programs

### PROG:EXEC:STAT

Changes the execution state of the selected program.

**Command** `PROGRAM:EXECute:STATe {RUN|PAUSE|STOP|CONTInue}`

Parameter	Value:	RUN	Execute the program
		PAUSE	Pause execution
		STOP	Stop execution
		CONTInue	Continue execution

### PROG:EXEC

Queries the execution state of the selected program.

**Command** `PROGRAM:EXECuting?`

**Response** Returns the execution state, time that has passed since execution began, the number of times that the program will repeat, the current step number, and the program number in <character>, <NR3> (number of digits to the right of the decimal is 5), <NR1>, <NR1>, <"string"> format in response to the PROG:EXEC? query.

Response format <character>: The execution state (RUN, PAUSE, STOP, or WAIT).  
 <NR3>: The time that has passed since execution began.  
 <NR1>: The number of times that the program will repeat.  
 <NR1>: The current step number.  
 <"string">: The program number.

Example When the execution mode is RUN: RUN,+1.00000E+00,1,1,"1".

Example When the execution mode is STOP: STOP,0,0,-1,"".



## Editing Scripts

### SCR:NAME

\* RST \* RCL

Selects the script or clears the current selection.

**Command** `SCRipt:NAME <"string">`  
`SCRipt:NAME?`

Parameter	Value:	"CC"	CC script
		"CV"	CV script
		" "	Blank string (clears the current selection; default)

**Response** Returns the name of the script in <"string"> format in response to the SCR:NAME? query.

### SCR:EDIT

Sets the contents of the selected script.

**Command** `SCRipt:EDIT <"string">`  
`SCRipt:EDIT?`

Parameter	Value:	"DO Pn,Rn"	Selects a program number. Pn: Program number (1 to 16). Rn: The number of times Pn will repeat (1 to 10000; 10001 for infinity).
		"DO LOOP Ln"	Loop start command. Ln: The number of times the loop will repeat (1 to 10000; 10001 for infinity).
		"END LOOP"	Loop end command.

Example `SCR:EDIT "DO P1,R1;DO P2,R1;DO LOOP L2;DO P3,R1;END LOOP"`

Parameters are separated by semicolons.

**Response** Returns the script editing string in <"string"> format in response to the SCR:EDIT? query.

### SCR:EDIT:TITL

Sets the name of the selected script to a string of 15 characters in length. A space character(0x20) is inserted automatically in case of a string of less than 15 characters in length.

**Command** `SCRipt:EDIT:TITLe <"string">`  
`SCRipt:EDIT:TITLe?`

Parameter	Value:	"string"
Example		"My SCRipt"

**Response** Returns the selected script's name in <"string"> format.

# 10 System Settings Commands

## CONFIG Settings

---

### SYST:CONF:BTR

Trips the breaker (turns the POWER switch off).

**Command** `SYSTem:CONFIgure:BTrip[:IMMediate]`

### SYST:CONF:BTR:PROT

Sets whether to trip the breaker or not when overvoltage protection (OVP) and overcurrent protection (OCP) are activated.

**Command** `SYSTem:CONFIgure:BTrip:PROTection {ON|OFF|1|0}`  
`SYSTem:CONFIgure:BTrip:PROTection?`

Parameter	Value:	ON (1)	Trip the breaker
		OFF (0)	Do not trip the breaker

**Response** Returns whether the breaker trips (the POWER switch turns off) or not in <NR1> format in response to the SYST:CONF:BTR:PROT? query.

### SYST:CONF:PAR

Sets the number of units that are operating in parallel. This can only be set on the master unit. This cannot be set when the output is on. This setting is changed after the PBZ is reset.

**Command** `SYSTem:CONFIgure:PARAllel {1|2|3|4|5}`  
`SYSTem:CONFIgure:PARAllel?`

Parameter	Value:	1 to 5 (The default value is 1.)
-----------	--------	----------------------------------

**Response** Returns the setting for the number of units that are operating in parallel in <NR1> format in response to the SYST:CONF:PAR? query.

## SYST:CONF:MSL

Sets the PBZ's parallel operation mode. This setting is changed after the PBZ is reset.

**Command** `SYSTem:CONFIgure:MSLave {MASTER|PARAllel}`  
`SYSTem:CONFIgure:MSLave?`

Parameter	Value:	PARAllel	Slave unit
		MASTer	Master unit or independent operation

**Response** Returns the parallel operation mode in <character> format in response to the SYST:CONF:MSL? query.

## SYST:CONF:SYNC:OPER

\* RST \* RCL

Sets the PBZ's synchronous operation mode.

**Command** `SYSTem:CONFIgure:SYNC:OPERation`  
`{OFF|MASTer|SLAVe|STERmination}`  
`SYSTem:CONFIgure:SYNC:OPERation?`

Parameter	Value:	OFF	Do not perform synchronous operation (default)
		MASTer	Master unit
		SLAVe	Slave unit
		STERmination	Slave end unit

**Response** Returns the synchronous operation mode in <character> format in response to the SYST:CONF:SYNC:OPER? query.

## SYST:CONF:TRAC

Executes a debug trace and sets whether communication errors are displayed or not. If you turn the debug trace feature on, error numbers, such as Err-100, are displayed on the screen.

**Command** `SYSTem:CONFIgure:TRACe {ON|OFF|1|0}`  
`SYSTem:CONFIgure:TRACe?`

Parameter	Value:	ON (1)	Communication errors are displayed.
		OFF (0)	Communication errors are not displayed.

**Response** Returns whether communication errors are displayed or not in <NR1> format in response to the SYST:CONF:TRAC? query.

## SYST:CONF:PRES:RCL:MOD

\* RST \* RCL

Sets the method for loading preset memory entries.

**Command** `SYSTem:CONFIgure:PRESet:RCL:MODE {<numeric>|MIN|MAX}`  
`SYSTem:CONFIgure:PRESet:RCL:MODE?`

Parameter	Value:	1	1KEY mode
		2	2KEY mode (default)

**Response** Returns the method for loading preset memory entries in <NR1> format in response to the SYST:CONF:PRES:RCL:MOD? query.

## SYST:CONF:RSEN

Sets whether to perform remote sensing or not.

**Command** `SYSTem:CONFigure:RSEnsing {ON|OFF|1|0}`  
`SYSTem:CONFigure:RSEnsing?`

Parameter Value: ON (1) Perform remote sensing  
OFF (0) Do not perform remote sensing

**Response** Returns whether remote sensing is performed or not in <NR1> format in response to the SYST:CONF:RSEN? query.

## SYST:CONF:KLOC:LEV

\* RST \* RCL

Sets the key lock security level for the front panel keys.

**Command** `SYSTem:CONFigure:KLOCk:LEVel {1|2|3|MIN|MAX}`  
`SYSTem:CONFigure:KLOCk:LEVel?`

Parameter Value: 1 Level 1  
2 Level 2  
3 Level 3 (default)  
Unit: None

**Response** Returns the key lock security level for the front panel keys in <NR1> format in response to the SYST:CONF:KLOC:LEV? query.

## SYST:CONF:BEEP:STAT

\* RST \* RCL

Turns the buzzer on and off.

**Command** `SYSTem:CONFigure:BEEPer:STATe {ON|OFF|1|0}`  
`SYSTem:CONFigure:BEEPer:STATe?`

Parameter Value: ON (1) On (default)  
OFF (0) Off

**Response** Returns whether the buzzer is on or off in <NR1> format in response to the SYST:CONF:BEEP:STAT? query.

## SYST:CONF:BEEP:VOL

\* RST \* RCL

Sets the buzzer volume.

**Command** `SYSTem:CONFigure:BEEPer:VOLume {<numeric>|MIN|MAX}`  
`SYSTem:CONFigure:BEEPer:VOLume? [ {MIN|MAX} ]`

Parameter Value: 0.0 to 1.0 (The default value is 1.0.)  
Unit: None.

**Response** Returns the buzzer volume setting in <NR3> format.

**SYST:CONF:PON:STAT**

\* RST \* RCL

Sets the power-on state.

**Command** `SYSTem:CONFigure:PON:STATe {DEFault|LAST}`  
`SYSTem:CONFigure:PON:STATe?`

Parameter	Value:	LAST	The settings that were in use immediately before the POWER switch was turned off (default).
		DEFault	The factory default settings.

**Response** Returns the power-on state in <character> format in response to the SYST:CONF:PON:STAT? query.

**SYST:CONF:SEQ:PON:STAT**

\* RST \* RCL

Sets the program or script to execute when the PBZ turns on.

**Command** `SYSTem:CONFigure:SEQuence:PON:STATe <"string">`  
`SYSTem:CONFigure:SEQuence:PON:STATe?`

Parameter	Value:	"1" to "16"	Program number
		"CV"	CV script
		"CC"	CC script
		" "	Blank string (clears the current selection; default)

**Response** Returns the name of the program or script to execute when the PBZ turns on in <"string"> format in response to the SYST:CONF:SEQ:PON:STAT query.

## Basic System Settings

### SYST:ERR

Retrieves the oldest error from the error queue. The error queue can hold up to 255 errors. The error queue is cleared if a \*CLS command is sent.

**Command** `SYSTem:ERRor[:NEXT]?`

**Response** Returns the oldest error held in the error queue in <NR1>,<"string"> format.

**Example** If there are no errors:

```
0, "No error"
```

**Example** If there is a command error:

```
-100, "Command error"
```

### SYST:KLOC

Locks and unlocks the panel keys.

**Command** `SYSTem:KLOCK {ON|OFF|1|0}`  
`SYSTem:KLOCK?`

Parameter	Value:	ON (1)	Lock the panel keys
		OFF (0)	Unlock the panel keys

**Response** Returns whether the panel keys are locked or not in <NR1> format.

### SYST:LOC

Sets the PBZ to local mode (panel operation). This is a substitute command for the IEEE 488.1 REN message (Remote Enable). You can switch the PBZ back to remote mode by sending the SYST:REM or SYST:RWL command.

If the PBZ is using the GPIB interface, an SCPI error (-200, "Execution error") occurs.

**Command** `SYSTem:LOCal`

### SYST:REM

Sets the PBZ to remote mode. All panel keys, except the LOCAL key, are locked. This is a substitute command for the IEEE 488.1 REN message (Remote Enable). This is also the substitute command for address specification.

You can switch the PBZ back to local mode by sending the SYST:LOC command.

If the PBZ is using the GPIB interface, an SCPI error (-200, "Execution error") occurs.

**Command** `SYSTem:REMOte`

## SYST:RWL

Sets the PBZ to remote mode. All panel keys (including the LOCAL key) are locked. This is a substitute command for the IEEE 488.1 llo message (Local Lock Out).

You can switch the PBZ back to local mode by sending the SYST:LOC command.

If the PBZ is using the GPIB interface, an SCPI error (-200, "Execution error") occurs.

**Command** `SYSTem:RWLock`

## SYST:OPT

Queries the options that are installed in the PBZ. This command performs the same function as the \*OPT? command.

**Command** `SYSTem:OPTion?`

**Response** If the factory option LAN interface board is installed, "LAN" is returned. Returns "0" if no options are installed.

## SYST:VERS

Queries the version of the SCPI specifications that the PBZ complies with.

**Command** `SYSTem:VERSion?`

**Response** Always returns 1999.0.

# 11 Status Register and Status Report Function

IEEE 488.2 and SCPI registers are used for status reports.

In each SCPI status register, there are the following sub registers: the CONDition register, the EVENT register, the ENABLE register, the PTRansition filter, and the NTRansition filter.

The figure on page 73 shows the SCPI status register structure. A “+” represents the logical OR of the register bits.

The tables on pages 74 to 78 provide the bit numbers, bit weights, bit names, and the meaning of each bit.

## CONDition register

The CONDition register transmits automatically and reflects the condition of the PBZ in real time. Reading this register does not affect its contents.

## EVENT register

The EVENT register bits are automatically set according to the changes in the CONDition register. The rule for setting the bits varies depending on the positive and negative transition filters (PTRansition and NTRansition). The EVENT register is reset when it is read.

## ENABLE register

The ENABLE register enables reports to the summary bit or status bit of the event bits.

## Transition filter

Use the PTRansition (positive transition) filter to report events when the condition changes from false to true.

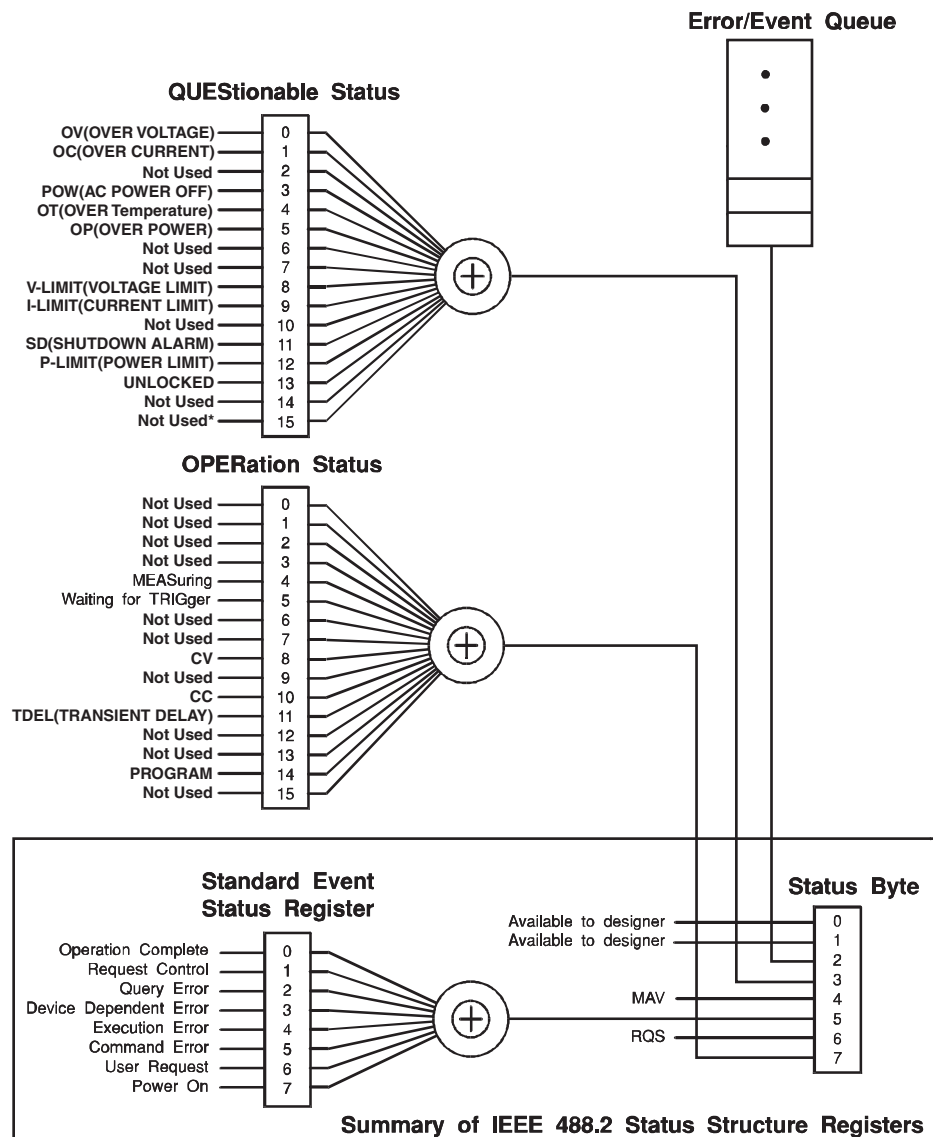
Use the NTRansition (negative transition) filter to report events when the condition changes from true to false.

If both the positive filter and negative filter are set to true, events can be reported each time the status changes.

If both filters are cleared, event reporting is disabled.



### SCPI status registers



\* The use of Bit 15 is not allowed since some controllers may have difficulty reading a 16 bit unsigned integer. The value of this bit shall always be 0.

Partially changed SCPI Standard 1999.0 Volume 1 fig. 9-1

## IEEE 488.2 Register Model

### Status Byte Register

The status byte register stores STB and RQS (MSS) messages as defined by the IEEE 488.1 standard. The status byte register can be read by using IEEE 488.1 serial polling or the IEEE 488.2 common command \*STB?.

When the controller executes serial polling, bit 6 responds with request service (RQS). The status byte value is not changed by serial polling.

\*STB? makes the PBZ transmit the contents of the status byte register and the master status summary (MSS) message.

\*STB? does not change the status byte, MSS, and RQS.

Bit	Bit weight	Bit name	Description
0	1	Reserved	Reserved for future use by IEEE 488. The bit value is notified as zero.
1	2	Reserved	
2	4	Error/Event Queue	If data exists in the error or event queue, this bit is set to true.
3	8	Questionable Status Register (QUES)	This bit is set to true when a bit is set in the QUESTIONABLE event status register and the corresponding bit in the QUESTIONABLE status enable register is true.
4	16	Message Available (MAV)	This bit is set to true when a request is received from the digital programming interface and the PBZ is ready to generate the data byte.
5	32	Standard Event Status Bit Summary (ESB)	This bit is set to true when a bit is set in the event status register.
6	64	Request Service (RQS)	This bit is set to true when a bit is set in the service request enable register and the corresponding bit exists in the status byte. The SRQ line of the GPIB is set.
		Master Status Summary (MSS)	This bit is set to true when any bit in the status byte register is set to 1 and the corresponding bit in the service request enable register is set to 1.
7	128	Operation Status Register (OPER)	This bit is set to true when a bit is set in the OPERATION event status register and the corresponding bit in the OPERATION status enable register is set.
8-15	—	NOT USED	—

## Event Status Register (Standard Event Status Register)

The event status register bits are set when certain events occur during PBZ operation. All the event status register bits are set by the error event queue.

This register is defined by the IEEE 488.2 standard and is controlled by the IEEE 488.2 common commands \*ESE, \*ESE?, and \*ESR?.

Bit	Bit weight	Bit name	Description
0	1	Operation Complete (OPC)	Set when an *OPC command is received and all operations in standby complete.
1	2	Request Control (RQC)	—
2	4	Query Error (QYE)	Set when an attempt is made to read data from the output queue when there is no data or when the output queue is not in the wait state. This indicates that there is no data in the output queue.
3	8	Device Dependent Error (DDE)	Set when there is a device-specific error.
4	16	Execution Error (EXE)	Set when the PBZ evaluates that the program data after the header is outside the formal input range or does not match the specifications of the PBZ. This indicates that a valid SCPI command may not be executed correctly depending on the state of the PBZ.
5	32	Command Error (CME)	Set when an IEEE 488.2 syntax error is detected by the parser, when an unidentifiable header is received, or when a group execution trigger enters the internal IEEE 488.2 SCPI command input buffer.
6	64	User Request (URQ)	—
7	128	Power ON (PON)	Set when the power is turned on.
8-15	—	Reserved	—

## SCPI Register Model

### OPERation Status Register (STATus:OPERation)

The OPERation status register is a 16-bit register that contains information about conditions that are part of normal PBZ operation.

Bit	Bit weight	Bit name	Description
0	1	NOT USED	—
1	2	NOT USED	—
2	4	NOT USED	—
3	8	NOT USED	—
4	16	MEASuring	Indicates whether measurement is in progress on the PBZ.
5	32	Waiting for TRIGger	Indicates whether the PBZ is waiting for a trigger (TRIG).
6	64	NOT USED	—
7	128	NOT USED	—
8	256	Constant Voltage	Indicates whether the PBZ is in CV mode.
9	512	NOT USED	—
10	1024	Constant Current	Indicates whether the PBZ is in CC mode.
11	2048	TRANSient Delay	Indicates whether the PBZ is delaying execution.
12	4096	NOT USED	—
13	8192	NOT USED	—
14	16384	PROGram	Indicates whether the PBZ is executing a program.
15	32768	NOT USED	—

### STAT:OPER

Queries the event of the OPERation status register.  
A query clears the contents of the register.

**Command** STATus:OPERation[:EVENT]?

**Response** Returns the event of the OPERation status register in <NR1> format.

### STAT:OPER:COND

Queries the condition of the OPERation status register.  
A query does not clear the contents of the register.

**Command** STATus:OPERation:CONDition?

**Response** Returns the condition of the OPERation status register in <NR1> format.

## STAT:OPER:ENAB

Sets the enable register of the OPERation status register.

**Command** `STATus:OPERation:ENABle <NRf>`  
`STATus:OPERation:ENABle?`

Parameter Value: 0 to 32767

**Response** Returns the enable register of the OPERation status register in <NR1> format.

## STAT:OPER:PTR

Sets the positive transition filter of the OPERation status register.

**Command** `STATus:OPERation:PTRansition <NRf>`  
`STATus:OPERation:PTRansition?`

Parameter Value: 0 to 32767

**Response** Returns the positive transition filter of the OPERation status register in <NR1> format.

## STAT:OPER:NTR

Sets the negative transition filter of the OPERation status register.

**Command** `STATus:OPERation:NTRansition <NRf>`  
`STATus:OPERation:NTRansition?`

Parameter Value: 0 to 32767

**Response** Returns the negative transition filter of the OPERation status register in <NR1> format.

### QUEStionable Status Register (STATus:QUEStionable)

The QUEStionable status register is a 16-bit register that stores information related to the PBZ's status and the questionable events that occur during PBZ operation.

These register bits may indicate that there are problems with the PBZ's measured data.

Bit	Bit weight	Bit name	Description
0	1	Over Voltage	Overvoltage protection has been activated.
1	2	Over Current	Overcurrent protection has been activated.
2	4	Not Used	—
3	8	AC Power OFF	The POWER switch is off.
4	16	Over Temp	The overheat sensor has been activated.
5	32	Over Power	The power limit protection has been activated.
6	64	Not Used	—
7	128	Not Used	—
8	256	V-Limit	The voltage limit has been activated.
9	512	I-Limit	The current limit has been activated.
10	1024	Not Used	—
11	2048	Shutdown Alarm	The POWER switch was turned off.
12	4096	Power Limit	The power limit has been activated.
13	8192	UNLOCKED	The PBZ is unsynchronized.
14	16384	Not Used	—
15	32768	Not Used	—

### STAT:QUES

Queries the event of the QUEStionable status register.

A query clears the contents of the register.

**Command** `STATus:QUEStionable[:EVENT]?`

**Response** Returns the event of the QUEStionable status register in <NR1> format.

### STAT:QUES:COND

Queries the condition of the QUEStionable status register.

A query does not clear the contents of the register.

**Command** `STATus:QUEStionable:CONDition?`

**Response** Returns the condition of the QUEStionable status register in <NR1> format.

## STAT:QUES:ENAB

Sets the enable register of the QUEStionable status register.

**Command** `STATus:QUEStionable:ENABle <NRf>`  
`STATus:QUEStionable:ENABle?`

Parameter Value: 0 to 32767

**Response** Returns the enable register of the QUEStionable status register in <NR1> format.

## STAT:QUES:PTR

Sets the positive transition filter of the QUEStionable status register.

**Command** `STATus:QUEStionable:PTRansition <NRf>`  
`STATus:QUEStionable:PTRansition?`

Parameter Value: 0 to 32767

**Response** Returns the positive transition filter of the QUEStionable status register in <NR1> format.

## STAT:QUES:NTR

Sets the negative transition filter of the QUEStionable status register.

**Command** `STATus:QUEStionable:NTRansition <NRf>`  
`STATus:QUEStionable:NTRansition?`

Parameter Value: 0 to 32767

**Response** Returns the negative transition filter of the QUEStionable status register in <NR1> format.

## Preset Status

### STAT:PRES

Resets the ENABLE, PTRansition, and NTRansition filter registers of all status registers (including sub registers) to their default values.

Default values:

STATus:ENABle = 0x0000

STATus:PTRansition = 0x7FFF

STATus:NTRansition = 0x0000

#### Command STATus : PRESet

Configures the status data, and enables certain events to be reported at a high level by the status reporting mechanism. These events are summarized in the OPERation status register and the QUEStionable status register that are part of the required structure.

STAT:PRES only affects the transition filter registers of the ENABle register and the status data structure.

STAT:PRES does not clear any event registers, nor does it clear any items from the error or event queues.

To reset all event registers and the queues in the device's status reporting mechanism, send the \*CLS command.

For status data that SCPI requires, STAT:PRES sets the transition filters so that they recognize only positive transitions and sets the ENABle register to all 0. The service request enable register, parallel polling enable register, memory registers that are related to the \*SAV command, the PBZ's address, the output queue, and the power-on status clear flag are not affected by this command.

Register preset values that can be set by the user

Register	Filter or enable	Preset value
QUEStionable	Enable register	All zeros
OPERation	Positive transition filter	All ones
	Negative transition filter	All zeros



# 12 Default State

The following tables show the PBZ settings (default state) when you send a \*RST, \*RCL, or MEAS? command or when you turn PBZ on. In each table, “MEAS?” represents the MEAS:<meas\_item> command.

## SYSTem subsystem

Command	Setting				Unit	Function
	*RST	*RCL	MEAS?	Power on		
SYST:KLOC	–	–	–		–	Panel key lock
SYST:CONF:BTR:PROT	–	–	–		–	Breaker tripping when a protection feature is activated
SYST:CONF:PAR	–	–	–		–	Number of units in parallel operation
SYST:CONF:MSL	–	–	–		–	Parallel operation: master/slave
SYST:CONF:SYNC:OPER	OFF	Depends on the settings in the memory	–	The settings that were in use immediately before the POWER switch was turned off.	–	Synchronous operation: master/slave
SYST:CONF:TRAC	–	–	–		–	Communication error display
SYST:CONF:PRES:RCL:MOD	2		–		–	Preset memory loading method
SYST:CONF:RSEN	–		–		–	Remote sensing
SYST:CONF:KLOC:LEV	3	Depends on the settings in the memory	–		–	Panel key lock and key lock level
SYST:CONF:BEEP:STAT	1 (ON)		–		–	Buzzer: on/off
SYST:CONF:BEEP:VOL	1.0		–		–	Buzzer volume
SYST:CONF:PON:STAT	LAST		–		–	Power-on state
SYST:CONF:SEQ:PON:STAT	“” (Blank string)		–		–	Program or script to execute when the PBZ turns on

## DISPlay subsystem

Command	Setting				Unit	Function
	*RST	*RCL	MEAS?	Power on		
DISP:CONT	0.7	–	–	The setting that was in use immediately before the POWER switch was turned off.	–	Screen contrast

### OUTPut subsystem

Command	Setting				Unit	Function	
	*RST	*RCL	MEAS?	Power on			
OUTP	0 (OFF)	0 (OFF)	–		–	Output: on/off	
OUTP:TRIG	IMMEDIATE	IMMEDIATE	–		–	Output: on/off according to software triggers	
OUTP:PON:STAT	RST		–	The settings that were in use immediately before the POWER switch was turned off.	–	Output: on/off when the PBZ turns on	
OUTP:EXT	NORMAL	Depends on the settings in the memory	–		–	–	Output-on control signal polarity
OUTP:TRIG:POL	POSITIVE		–		–	–	Trigger signal output polarity
OUTP:TRIG:STAT	0 (OFF)		–		–	–	Trigger signal output: on/off
OUTP:PORT	0 (OFF)	–	–		–	J1 connector and option terminal output: on/off	

### SENSe subsystem

Command	Setting				Unit	Function	
	*RST	*RCL	MEAS?	Power on			
SENS:APER	0.1	Depends on the settings in the memory	–	The settings that were in use immediately before the POWER switch was turned off.	S	Measurement time	
SENS:FUNC	DC		–		–	–	Measurement function
SENS:TRIG:DEL	0		–		–	S	Trigger delay
SENS:TRIG:SOUR	AUTO		–		–	–	Measurement start trigger source

## [SOURce] subsystem

Command	Setting			Power on	Unit	Function
	*RST	*RCL	MEAS?			
FUNC:MODE	CV		–		–	CV/CC mode
FUNC:POL	BIPolar		–		–	Bipolar/unipolar mode
FUNC:SOUR	INTernal	Depends on the settings in the memory	–		–	Signal source selection
FUNC:EXT:TERM	BNC		–		–	External signal source terminal selection
FUNC:EXT:CURR:GAIN	0		–		A/V	External signal source gain
FUNC:EXT:VOLT:GAIN	0		–		V/V	External signal source gain
FUNC	SIN		–		–	AC signal selection
FUNC:TRIG	IMMEDIATE	IMMEDIATE	–		–	AC signal source selection according to software triggers
AC:STAT	0 (OFF)	Depends on the settings in the memory	–		–	AC signal: on/off
AC:STAT:TRIG	IMMEDIATE	IMMEDIATE	–		–	AC signal: on/off according to software triggers
VOLT	0	Depends on the settings in the memory	–		V	DC signal voltage value
VOLT:TRIG	IMMEDIATE		–		V	DC signal voltage value according to software triggers
VOLT:AC	0		–		VPP	AC signal voltage value
VOLT:AC:TRIG	IMMEDIATE	IMMEDIATE	–		VPP	AC signal voltage value according to software triggers
CURR	0	Depends on the settings in the memory	–		A	DC signal current value
CURR:TRIG	IMMEDIATE		–		A	DC signal current value according to software triggers
CURR:AC	0		–		APP	AC signal current value
CURR:AC:TRIG	IMMEDIATE	IMMEDIATE	–		APP	AC signal current value according to software triggers
FREQ	1 kHz	Depends on the settings in the memory	–		HZ	AC signal frequency value
FREQ:TRIG	IMMEDIATE	IMMEDIATE			HZ	AC signal frequency value according to software triggers
PHAS	0 or 1 (ON)		–		DEG	AC signal start phase angle
SQU:DCYC	50.0		–		PCT	Square wave signal duty cycle
SWE:STAT	0 (OFF)		–		–	Frequency sweep: on/off
SWE:SPAC	LINear	Depends on the settings in the memory	–		–	Frequency sweep mode
SWE:TIME	1		–		S	Frequency sweep time
SWE:FREQ:STAR	100 Hz		–		HZ	Frequency sweep start frequency
SWE:FREQ:STOP	1 kHz		–		HZ	Frequency sweep stop frequency
MARK	0 (OFF)		–		–	TRIG OUT terminal output type
MARK:FPO	500 Hz		–		HZ	Marker frequency
APPL:DC	–	–	–		HZ	
APPL:SIN					VPP	Collective settings (AC signal waveform, frequency, amplitude, and DC signal)
APPL:SQU					APP	
APPL:TRI					V	
APPL:USER{[1]...16}					V	
					A	

The settings that were in use immediately before the POWER switch was turned off.

### [SOURce] subsystem

Command	Setting				Unit	Function
	*RST	*RCL	MEAS?	Power on		
CURR:RESP	35		–		US	CC mode response (PBZ20-20/ PBZ60-67/ PBZ80-5)
	70		–		US	CC mode response (PBZ40-10)
VOLT:RESP	3.5		–		US	CV mode response
CURR:PROT:OVER CURR:LIM:UPP CURR:PROT:UPP	110 % of the rated value	Depends on the settings in the memory	–	The settings that were in use immediately before the POWER switch was turned off.	A	Positive current activation point
VOLT:PROT:OVER VOLT:LIM:UPP VOLT:PROT:UPP	110 % of the rated value		–		V	Positive voltage activation point
CURR:PROT:UND CURR:LIM:LOW CURR:PROT:LOW	110 % of the rated value		–		A	Negative current activation point
VOLT:PROT:UND VOLT:LIM:LOW VOLT:PROT:LOW	110 % of the rated value		–		V	Negative voltage activation point
CURR:PROT:STAT	LIMIT		–		–	OCP mode
VOLT:PROT:STAT	LIMIT		–		–	OVP mode

### MEASure & TRIGger subsystem

Command	Setting				Unit	Function
	*RST	*RCL	MEAS?	Power on		
TRIG:SOUR TRIG:TRAN:SOUR	IMMEDIATE	–	IMMEDIATE	IMMEDIATE	–	SEQUENCE1 trigger source
TRIG:SEQ2:SOUR TRIG:OUTP:SOUR	IMMEDIATE	–	IMMEDIATE	IMMEDIATE	–	SEQUENCE2 trigger source
TRIG:SEQ2:SST:RISE TRIG:OUTP:SST:RISE	0.0	Depends on the settings in the memory	No change	No change	S	SEQUENCE2 soft start time
TRIG:SEQ2:SST:FALL TRIG:OUTP:SST:FALL	0.0		No change	No change	S	SEQUENCE2 soft stop time
TRIG:SEQ3:SOUR TRIG:ACQ:SOUR	IMMEDIATE	–	IMMEDIATE	IMMEDIATE	–	SEQUENCE3 trigger source
TRIG:SEQ4:SOUR TRIG:PROG:SOUR	IMMEDIATE	–	IMMEDIATE	IMMEDIATE	–	SEQUENCE4 trigger source

### INPut subsystem

Command	Setting				Unit	Function
	*RST	*RCL	MEAS?	Power on		
INP:TRIG:POL	POSITIVE	Depends on the settings in the memory	–	The setting that was in use immediately before the POWER switch was turned off.	–	Trigger signal input polarity

## PROGram subsystem

Command	Setting				Unit	Function	
	*RST	*RCL	MEAS?	Power on			
PROG:NAME	"" (Blank string)	"" (Blank string)	-		-	Program selection and selection clearing	
PROG:EDIT:FUNC:MODE	-	-	-		-	Program: CV/CC mode	
PROG:EDIT:FUNC:POL	-	-	-		-	Program: bipolar/unipolar mode	
PROG:EDIT:LOOP	-	-	-		-	Number of times that the program will repeat	
PROG:EDIT:STEP:CURR	-	-	-		V	Step's DC signal value; value at the end of step if RAMP is specified	
PROG:EDIT:STEP:VOLT	-	-	-		A		
	-	-	-		-	Step's signal change	
PROG:EDIT:STEP:CURR:RAMP	-	-	-		V	Step's DC signal RAMP start value	
PROG:EDIT:STEP:VOLT:RAMP	-	-	-		A		
PROG:EDIT:STEP:FUNC	-	-	-		-	Step's AC signal waveform	
PROG:EDIT:STEP:CURR:AC	-	-	-		VPP	Step's AC signal value; value at the end of step if SWE is specified	
PROG:EDIT:STEP:VOLT:AC	-	-	-		APP		
PROG:EDIT:STEP:CURR:AC:SWE	-	-	-	The settings that were in use immediately before the POWER switch was turned off.	VPP	Step's AC signal SWE start value	
PROG:EDIT:STEP:VOLT:AC:SWE	-	-	-		APP		
PROG:EDIT:STEP:AC:STAT	-	-	-		-	-	Step's AC signal on/off
PROG:EDIT:STEP:FREQ	-	-	-		-	-	Step's AC signal frequency change
PROG:EDIT:STEP:FREQ:SWE	-	-	-		-	Step's AC signal frequency sweep mode	
	-	-	-		HZ	Step's AC signal frequency sweep start frequency	
	-	-	-		HZ	Step's AC signal frequency sweep stop frequency	
PROG:EDIT:STEP:PHAS	-	-	-		DEG	Step's AC signal start phase angle	
	-	-	-		-		
PROG:EDIT:STEP:SQU:DCYC	50 %	-	-		PCT	Step's square wave duty cycle	
PROG:EDIT:STEP:TIME	-	-	-		S	Step execution time	
PROG:EDIT:STEP:STAT	-	-	-		-	Output: on/off	
	-	-	-		-	Trigger signal output	
	-	-	-		-	Trigger signal input	

## SCRipt subsystem

Command	Setting				Unit	Function
	*RST	*RCL	MEAS?	Power on		
SCR:NAME	"" (Blank string)	"" (Blank string)	-	The setting that was in use immediately before the POWER switch was turned off.	-	Script selection and selection clearing

# 13 Processing time of Commands

The command processing time is the time until the next command is accepted. The processing times indicated here are typical values. They are not warranted. The processing times vary depending on the settings and the measurement conditions. It does not include the response time of the hardware.

Command	GPIB <sup>1</sup> Processing Time (ms)	USB Processing Time (ms)	RS232C <sup>2</sup> Processing Time (ms)	LAN <sup>3</sup> Processing Time (ms)	Description
VOLT 20	47	47	47	47	Sets the DC signal voltage value.
MEAS:VOLT? <sup>4</sup>	145	145	145	145	Queries the DC voltage measurement.
CURR 20	47	47	47	47	Sets the DC signal current value.
MEAS:CURR? <sup>4</sup>	145	145	145	145	Queries the DC current measurement.
*RST	260	260	260	260	Performs a device reset.

- 1 Using GPIB-USB-B by National Instruments.
- 2 Data rate setting: 38400 bps. Flow control: On
- 3 100BASE-TX Ethernet
- 4 Measurement time: 0.1 s

# 14 List of Messages

SCPI Command: Command name in the short form.

Effect: "Yes" for commands that are affected by the \*RST and \*RCL commands.

R/W: "R" for query commands and "W" for set commands.

†: "1" for SCPI standard commands, "2" for commands that are being reviewed, and "3" for original commands developed by Kikusui Electronics Corporation.

## SYSTEM subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
SYST									
:CONF									
:BTR	-	-	-	-	-	-	Breaker tripping	W	3
:BTR:PROT	bool	ON   OFF	-	-	NR1	-	Breaker tripping when a protection feature is activated	R/W	3
:PAR	NRf	1 to 5	-	-	NR1	-	Number of units in parallel operation	R/W	3
:MSL	character	MAST   PAR	-	-	char	-	Parallel operation: master/slave	R/W	3
:SYNC:OPER	character	OFF   MAST   SLAV   ATER	-	-	char	-	Synchronous operation: master/slave	R/W	3
:TRAC	bool	ON   OFF	-	-	NR1	-	Communication error display	R/W	3
:PRES:RCL:MOD	numeric	1   2	-	2	NR1	-	Preset memory loading method	R/W	3
:RSEN	bool	ON   OFF	-	-	NR1	-	Remote sensing	R/W	3
:KLOC:LEV	numeric	1   2   3	-	3	NR1	-	Panel key lock and key lock level	R/W	3
:BEEP									
:STAT	bool	ON   OFF	-	ON	NR1	-	Buzzer: on/off	R/W	3
:VOL	numeric	0.0 to 1.0	-	1.0	NR3	-	Buzzer volume	R/W	3
:PON:STAT	character	DEF   LAST	-	LAST	char	-	Power-on state	R/W	3
:SEQ:PON:STAT	string	"1" to "16", "CV", "CC", or ""	-	"" (Blank string)	string	-	Program or script to execute when the PBZ turns on	R/W	3
:ERR[:NEXT]	-	-	-	-	NR1, string	-	Queries error and event information	R	1
:KLOC	bool	ON   OFF	-	-	NR1	-	Panel key lock	R/W	3
:LOC	-	-	-	-	-	-	Switches to local mode	W	2
:REM	-	-	-	-	-	-	Switches to remote mode; locks all keys other than LOCAL and STOP	W	2
:RWL	-	-	-	-	-	-	Switches to remote mode; locks all keys other than STOP	W	2
:OPT	-	-	-	-	-	-	Queries options	R	3
:VERS	-	-	-	-	1999.0	-	Queries the version of the compliant SCPI specifications	R	1

## DISPlay subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
DISP									
:CONT	numeric	0.0 to 1.0	-	0.7	NR3	-	Screen contrast	R/W	3

## MEMory subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
MEM									
:PRES									
:RCL	NRf	1 to 3	–	–	–	–	Preset memory loading	W	3
:SAV	NRf	1 to 3	–	–	–	–	Preset memory saving	W	3
:SET									
:RCL	NRf	0 to 9	–	–	–	–	Setup memory loading	W	3
:SAV	NRf	0 to 9	–	–	–	–	Setup memory saving	W	3
:TITL	NRf string	0 to 9 string	–	–	NR1 string	–	Setup memory name and memory number	R/W	3

## OUTPut subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
OUTP									
[:STAT]	bool	ON   OFF	–	OFF	NR1	–	Output: on/off	R/W	3
:TRIG	bool	ON   OFF	–	IMM	NR1	–	Output: on/off according to software triggers	R/W	3
:PON:STAT	character	RST   AUTO	–	RST	char	–	Output: on/off when the PBZ turns on	R/W	3
:EXT	character	NORM   INV	–	NORM	char	–	Output-on external control signal polarity	R/W	3
:TRIG:POL	character	POS   NEG	–	POS	char	–	Trigger signal output polarity	R/W	3
:TRIG:STAT	bool	ON   OFF	–	OFF	NR1	–	Trigger signal output: on/off	R/W	3
:PORT	bool	ON   OFF	–	OFF	NR1	–	Option output: on/off	R/W	3
:PROT:CLE	–	–	–	–	–	–	Alarm clearing	W	3

## INPut subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
INP									
:TRIG:POL	character	POS   NEG	–	POS	char	–	Trigger signal input polarity	R/W	3

## SENSe subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
SENS									
:APER	numeric	0.0001 to 3600	–	0.1	NR3	Yes	Measurement time	R/W	3
:FUNC	character	DC   AC   DCAC   PEAK	–	DC	char	–	Measurement function	R/W	3
:TRIG:DEL	numeric	0.00000 to 3600	–	0	NR3	Yes	Trigger delay	R/W	3
:TRIG:SOUR	character	AUTO   INT   EXTPOS   EXTNEG	–	AUTO	char	–	Measurement start trigger source	R/W	3



## [SOURce] subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
[SOUR:]									
FUNC	character	SIN   SQU   TRI   USER[1] to 16	–	SIN	char	Yes	AC signal selection	R/W	3
:MODE	character	CV   CC	–	CV	char	Yes	CV/CC mode	R/W	3
:POL	character	BIP   UNIP	–	BIP	char	Yes	Bipolar/unipolar mode	R/W	3
:SOUR	character	INT   EXT   BOTH	–	INT	char	Yes	Signal source selection	R/W	3
TRIG	character	SIN   SQU   TRI   USER[1] to 16	–	IMM	char	Yes	AC signal source selection according to software triggers	R/W	3
:EXT:TERM	character	BNC   J1   BOTH	–	BNC	char	Yes	External signal source terminal selection	R/W	3
:EXT:CURR:GAIN	numeric	0 to 20 or 0 to 10	A/V	0	NR3	Yes	External signal source gain	R/W	3
:EXT:VOLT:GAIN	numeric	0 to 20 or 0 to 40	V/V	0	NR3	Yes	External signal source gain	R/W	3
USERx:TITL	string	–	–	–	string	–	User-defined arbitrary waveform's name	R/W	3
USERx:DATA	string	–	–	–	string	–	User-defined arbitrary waveform's pattern	R/W	3
USERx:DATA:FORM:BOARD	character	NORM   SWAP	–	–	char	Yes	User defined arbitrary waveform's binary transmission data byte order	R/W	3
USERx:DATA:VAL	numeric	addr:0 to 1023 data:-32768 to +32767	–	–	NR1	–	Sets the value at an address in a user-defined arbitrary waveform pattern.	R/W	3
VOLT	numeric	0 % to 105 % of the rated value	V	0	NR3	Yes	DC signal voltage value	R/W	3
:TRIG	numeric	0 % to 105 % of the rated value	V	IMM	NR3	Yes	DC signal voltage value according to software triggers	R/W	3
:AC	numeric	0 % to 105 % of the rated value	VPP	0	NR3	Yes	AC signal voltage value	R/W	3
:TRIG	numeric	0 % to 105 % of the rated value	VPP	IMM	NR3	Yes	AC signal voltage value according to software triggers	R/W	3
:PROT									
:OVER	numeric	+1 % to +110 % of the rated value	V	+110 %	NR3	Yes	Positive OVP activation point	R/W	3
:UND	numeric	-110 % to -1 % of the rated value	V	-110 %	NR3	Yes	Negative OVP activation point	R/W	3
:UPP	numeric	+OVP: +1 % to +110 % of the rated value -110 % ≤ -V.LIM ≤ +V.LIM ≤ +110 % of the rated value	V	+110 %	NR3	Yes	Positive OVP/V.LIM activation point	R/W	3
:LOW	numeric	-OVP: -110 % to -1 % of the rated value -110 % ≤ -V.LIM ≤ +V.LIM ≤ +110 % of the rated value	V	-110 %	NR3	Yes	Negative OVP/V.LIM activation point	R/W	3
:STAT	character	LIMIT   TRIP	–	LIMIT	char	Yes	OVP mode	R/W	3
:LIM									
:UPP	numeric	-110 % ≤ -V.LIM ≤ +V.LIM ≤ +110 % of the rated value	V	+110 %	NR3	Yes	Positive V.LIM activation point	R/W	3
:LOW	numeric	-110 % ≤ -V.LIM ≤ +V.LIM ≤ +110 % of the rated value	V	-110 %	NR3	Yes	Negative V.LIM activation point	R/W	3

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
[SOUR:]									
CURR	numeric	0 % to 105 % of the rated value	A	0	NR3	Yes	DC signal current value	R/W	3
:TRIG	numeric	0 % to 105 % of the rated value	A	IMM	NR3	Yes	DC signal current value according to software triggers	R/W	3
:AC	numeric	0 % to 105 % of the rated value	APP	0	NR3	Yes	AC signal current value	R/W	3
:TRIG	numeric	0 % to 105 % of the rated value	APP	IMM	NR3	Yes	AC signal current value according to software triggers	R/W	3
:PROT									
:OVER	numeric	+1 % to +110 % of the rated value	A	-110 %	NR3	Yes	Positive OCP activation point	R/W	3
:UND	numeric	-110 % to -1 % of the rated value	A	-110 %	NR3	Yes	Negative OCP activation point	R/W	3
:UPP	numeric	+1 % to +110 % of the rated value	A	+110 %	NR3	Yes	Positive OCP/I.LIM activation point	R/W	3
:LOW	numeric	-110 % to -1 % of the rated value	A	-110 %	NR3	Yes	Negative OCP/I.LIM activation point	R/W	3
:STAT	character	LIMIT   TRIP	-	LIMIT	char	Yes	OCP mode	R/W	3
:PROT									
:UPP	numeric	+1 % to +110 % of the rated value	A	+110 %	NR3	Yes	Positive I.LIM activation point	R/W	3
:LOW	numeric	-110 % to -1 % of the rated value	A	-110 %	NR3	Yes	Negative I.LIM activation point	R/W	3
FREQ									
	numeric	CV:1.0E-02 to 1.0E+05 CC:1.0E-02 to 1.0E+05	HZ	1.0E+03	NR3	Yes	AC signal frequency value (PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5)	R/W	3
	numeric	CV:1.0E-02 to 2.0E+05 CC:1.0E-02 to 2.0E+05	HZ	1.0E+03	NR3	Yes	AC signal frequency value (PBZ20-20A)	R/W	3
:TRIG	numeric	CV:1.0E-02 to 1.0E+05 CC:1.0E-02 to 1.0E+05	HZ	IMM	NR3	Yes	AC signal frequency value according to software triggers (PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5)	R/W	3
	numeric	CV:1.0E-02 to 2.0E+05 CC:1.0E-02 to 2.0E+05	HZ	IMM	NR3	Yes	AC signal frequency value according to software triggers (PBZ20-20A)	R/W	3
PHAS									
	numeric	0 to 359	DEG	0	NR3	Yes	AC signal start phase angle	R/W	3
	bool	ON   OFF	-	ON	NR1	Yes		R/W	3
SQU:DCYC									
	numeric	0.1 to 99.9	PCT	50.0	NR3	Yes	Square wave signal duty cycle	R/W	3
SWE									
:STAT	bool	ON   OFF	-	OFF	NR1	Yes	Frequency sweep: on/off	R/W	3
:SPAC	character	LOG   LIN	-	LIN	char	Yes	Frequency sweep mode	R/W	3
:TIME	numeric	0.0001 to 1000	S	1	NR3	Yes	Frequency sweep time	R/W	3
:FREQ									
:STAR	numeric	CV:1.0E-02 to 1.0E+05 CC:1.0E-02 to 1.0E+05	HZ	1.0E+02	NR3	Yes	Frequency sweep start frequency (PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5)	R/W	3
	numeric	CV:1.0E-02 to 2.0E+05 CC:1.0E-02 to 2.0E+05	HZ	1.0E+02	NR3	Yes	Frequency sweep start frequency (PBZ20-20A)	R/W	3
:STOP	numeric	CV:1.0E-02 to 1.0E+05 CC:1.0E-02 to 1.0E+05	HZ	1.0E+03	NR3	Yes	Frequency sweep stop frequency (PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5)	R/W	3
	numeric	CV:1.0E-02 to 2.0E+05 CC:1.0E-02 to 2.0E+05	HZ	1.0E+03	NR3	Yes	Frequency sweep stop frequency (PBZ20-20A)	R/W	3
MARK									
	bool	ON   OFF	-	OFF	NR1	Yes	TRIG OUT terminal output type	R/W	3
:FPO	numeric	CV:1.0E-02 to 1.0E+05 CC:1.0E-02 to 1.0E+05	HZ	500	NR3	Yes	Marker frequency (PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5)	R/W	3
	numeric	CV:1.0E-02 to 2.0E+05 CC:1.0E-02 to 2.0E+05	HZ	500	NR3	Yes	Marker frequency (PBZ20-20A)	R/W	3

SCPI command		Setting		Default	Response	Effect	Description	R/W	†	
Program header	Parameter		Unit							
[SOUR:]										
APPL										
	:DC	–	–	–	–	char	–	Collective settings (AC signal waveform)	R/W	3
	:SIN									
	:SQU									
	:TRI	numeric	CV:1.0E-02 to 1.0E+05 CC:1.0E-02 to 1.0E+05	HZ	–	NR3	Yes	Collective settings (AC signal frequency) (PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5)	R/W	3
	:USERx	numeric	CV:1.0E-02 to 2.0E+05 CC:1.0E-02 to 2.0E+05	HZ	–	NR3	Yes	Collective settings (AC signal frequency) (PBZ20-20A)	R/W	3
		numeric	0 % to 105 % of the rated value	APP	–	NR3	Yes	Collective settings (AC signal amplitude)	R/W	3
		numeric	0 % to 105 % of the rated value	V	–	NR3	Yes	Collective settings (DC signal)	R/W	3
	CURR:RESP	numeric	35   100   350   1000	US	35	NR3	Yes	CC mode response (PBZ20-20/PBZ20-20/PBZ80-5)	R/W	3
		numeric	70   100   350   1000	US	70	NR3	Yes	CC mode response (PBZ40-10)	R/W	3
	VOLT:RESP	numeric	3.5   10   35   100	US	3.5	NR3	Yes	CV mode response	R/W	3
:AC										
	:STAT	bool	ON   OFF	–	OFF	NR1	Yes	AC signal: on/off	R/W	3
	:STAT:TRIG	bool	ON   OFF	–	IMM	NR1	Yes	AC signal: on/off according to software triggers	R/W	3

### MEASure/TRIGger subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†	
Program header	Parameter		Unit							
FETC[:SCAL]   READ[:SCAL]   MEAS[:SCAL]										
:CURR										
	[:DC]	–	–	A	–	NR3	–	DC current measurement	R	3
	:AC	–	–	A	–	NR3	–	AC current measurement (RMS)	R	3
	:MIN	–	–	APP	–	NR3	–	Minimum current measurement (bottom value)	R	3
	:MAX	–	–	APP	–	NR3	–	Maximum current measurement (peak value)	R	3
:VOLT										
	[:DC]	–	–	V	–	NR3	–	DC voltage measurement	R	3
	:AC	–	–	V	–	NR3	–	AC voltage measurement (RMS)	R	3
	:MIN	–	–	VPP	–	NR3	–	Minimum voltage measurement (bottom value)	R	3
	:MAX	–	–	VPP	–	NR3	–	Maximum voltage measurement (peak value)	R	3
	ABOR	–	–	–	–	–	–	Measurement aborting	W	1
INIT[:IMM]										

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
[[:SEQ[1]]]	-	-	-	-	-	-	Starts the trigger function that sets the voltage, current, AC waveform, and frequency	W	1
:SEQ2	-	-	-	-	-	-	Starts the trigger function that turns output on and off	W	1
:SEQ3	-	-	-	-	-	-	Starts the trigger function that starts measurements and queries measured results	W	1
:SEQ4	-	-	-	-	-	-	Starts the trigger function that executes programs and scripts	W	1
[[:IMM]:NAME TRAN	-	-	-	-	-	-	Starts the trigger function that sets the voltage, current, AC waveform, and frequency	W	1
[[:IMM]:NAME OUTP	-	-	-	-	-	-	Starts the trigger function that turns output on and off	W	1
[[:IMM]:NAME ACQ	-	-	-	-	-	-	Starts the trigger function that starts measurements and queries measured results	W	1
[[:IMM]:NAME PROG	-	-	-	-	-	-	Starts the trigger function that executes programs and scripts	W	1
TRIG									
[[:SEQ[1]]]	-	-	-	-	-	-	Executes SEQUENCE1 software triggers	W	1
[[:SOUR	character	IMM   BUS	-	IMM	char	Yes	SEQUENCE1 trigger source selection	R/W	3
:SEQ2	-	-	-	-	-	-	Executes SEQUENCE2 software triggers	W	1
[[:SOUR	character	IMM   BUS	-	IMM	char	Yes	SEQUENCE2 trigger source selection	R/W	3
[[:SST:RISE	numeric	0.0001 to 1000	S	0.0000	NR3	Yes	SEQUENCE2 soft start time	R/W	3
[[:SST:FALL	numeric	0.0001 to 1000	S	0.0000	NR3	Yes	SEQUENCE2 soft stop time	R/W	3
:SEQ3	-	-	-	-	-	-	Executes SEQUENCE3 software triggers	W	1
[[:SOUR	character	IMM   BUS	-	IMM	char	Yes	SEQUENCE3 trigger source selection	R/W	3
:SEQ4	-	-	-	-	-	-	Executes SEQUENCE4 software triggers	W	1
[[:SOUR	character	IMM   BUS   EXT	-	IMM	char	Yes	SEQUENCE4 trigger source selection	R/W	3
:TRAN	-	-	-	-	-	-	Executes SEQUENCE1 software triggers	W	1
[[:SOUR	character	IMM   BUS	-	IMM	char	Yes	SEQUENCE1 trigger source selection	R/W	3
:OUTP	-	-	-	-	-	-	Executes SEQUENCE2 software triggers	W	1
[[:SOUR	character	IMM   BUS	-	IMM	char	Yes	SEQUENCE2 trigger source selection	R/W	3
[[:OUTP:SST:RISE	numeric	0.0001 to 1000	S	0.0000	NR3	Yes	SEQUENCE2 soft start time	R/W	3
[[:OUTP:SST:FALL	numeric	0.0001 to 1000	S	0.0000	NR3	Yes	SEQUENCE2 soft stop time	R/W	3
:ACQ	-	-	-	-	-	-	Executes SEQUENCE3 software triggers	W	1
[[:SOUR	character	IMM   BUS	-	IMM	char	Yes	SEQUENCE3 trigger source selection	R/W	3
:PROG	-	-	-	-	-	-	Executes SEQUENCE4 software triggers	W	1
[[:SOUR	character	IMM   BUS   EXT	-	IMM	char	Yes	SEQUENCE4 trigger source selection	R/W	3

## PROGram subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
PROG									
:NAME	string	1 to 16   ""	–	"" (Blank string)	string	–	Program selection and selection clearing	R/W	3
:EDIT									
:ADD	NRf	–	–	–	–	–	Number of steps to add	W	3
:COUN	–	–	–	–	NR1	–	Queries the number of steps	R	3
:FREE	–	–	–	–	NR1	–	Queries the number of steps that can be added	R	3
:DEL	–	–	–	–	–	–	Deletes all steps from the selected program	W	3
:TITL	string	–	–	–	string	–	Selected program's name	R/W	3
:FUNC:MODE	character	CC   CV	–	–	char	–	Selected program's CV/CC mode	R/W	3
:FUNC:POL	character	BIP   UNIP	–	–	char	–	Selected program's bipolar/unipolar mode	R/W	3
:LOOP	NRf	1 to 10000   10001	–	–	NR3	–	Number of times that the selected program will repeat	R/W	3
:STEP									
:SEL	NRf	–	–	–	NR1	–	Selected program's step number selection	R/W	3
:FUNC	character	SIN   SQU   TRI   USER[1] to 16	–	–	char	–	Selected step's AC signal waveform selection	R/W	3
:CURR	numeric	0 % to 105 % of the rated value	A	–	NR3	Yes	Selected step's DC signal current value; value at the end of step if RAMP is specified	R/W	3
	character	IMM   RAMP	–	–	char	–	Selected step's DC signal current change	R/W	3
:CURR:RAMP	numeric	0 % to 105 % of the rated value	A	–	NR3	Yes	Selected step's DC signal RAMP start value	R/W	3
:CURR:AC	numeric	0 % to 105 % of the rated value	APP	–	NR3	Yes	Selected step's AC signal current value; value at the end of step if SWE is specified	R/W	3
	character	IMM   SWE	–	–	char	–	Selected step's AC signal current change	R/W	3
:VOLT	numeric	0 % to 105 % of the rated value	V	–	NR3	Yes	Selected step's DC signal voltage value; value at the end of step if RAMP is specified	R/W	3
	character	IMM   RAMP	–	–	char	–	Selected step's DC signal voltage change	R/W	3
:VOLT:RAMP	numeric	0 % to 105 % of the rated value	V	–	NR3	Yes	Selected step's DC signal RAMP start value	R/W	3
:VOLT:AC	numeric	0 % to 105 % of the rated value	VPP	–	NR3	Yes	Selected step's AC signal voltage value; value at the end of step if SWE is specified	R/W	3
	character	IMM   SWE	–	–	char	–	Selected step's AC signal voltage change	R/W	3
:FREQ	numeric	CV:1.0E-02 to 1.0E+05 CC:1.0E-02 to 1.0E+05	HZ	–	NR3	Yes	Selected step's AC signal frequency (PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5)	R/W	3
	numeric	CV:1.0E-02 to 2.0E+05 CC:1.0E-02 to 2.0E+05	HZ	–	NR3	Yes	Selected step's AC signal frequency (PBZ20-20A)	R/W	3
	character	IMM   SWE	–	–	char	–	Selected step's AC signal frequency change	R/W	3

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
PROG:EDIT:STEP									
:FREQ:SW	character	LOG   LIN	–	–	char	Yes	Selected step's AC signal frequency sweep mode	R/W	3
	numeric	CV:1.0E-02 to 1.0E+05 CC:1.0E-02 to 1.0E+05	HZ	–	NR3	Yes	Selected step's AC signal frequency sweep start frequency (PBZ20-20/PBZ40-10/PBZ60-6.7/PBZ80-5)	R/W	3
	numeric	CV:1.0E-02 to 2.0E+05 CC:1.0E-02 to 2.0E+05	HZ	–	NR3	Yes	Selected step's AC signal frequency sweep stop frequency (PBZ20-20A)	R/W	3
:PHAS	numeric	0 to 359	DEG	–	NR3	Yes	Selected step's AC signal start phase angle	R/W	3
	bool	ON   OFF	–	–	NR1	Yes		R/W	3
:SQU:DCYC	numeric	0.1 to 99.9	PCT	50	NR3	Yes	Selected step's square wave signal duty cycle	R/W	3
:AC:STAT	bool	ON   OFF	–	–	NR1	Yes	Selected step's AC signal on/off	R/W	3
:TIME	numeric	0.0001 to 3600000	S	–	NR3	Yes	Selected step's frequency sweep time	R/W	3
:STAT	bool	ON   OFF	–	–	NR1	Yes	Selected step's output on/off and trigger signal I/O	R/W	3
	bool	ON   OFF	–	–	NR1	Yes		R/W	3
	bool	ON   OFF	–	–	NR1	Yes		R/W	3

### PROG:EXECute subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
PROG									
:EXEC:STAT	character	RUN   PAUSE   STOP   CONT	–	–	–	–	Selected program's execution state	W	3
:EXEC	–	–	–	–	char	–	Selected program's execution state (RUN, PAUSE, STOP, or WAIT), time that has passed since execution began, number of times that program will repeat, current step number, and program number.	R	3
	–	–	–	–	NR3	–		R	3
	–	–	–	–	NR1	–		R	3
	–	–	–	–	NR1	–		R	3
	–	–	–	–	string	–		R	3

### SCRipt subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
SCR									
:NAME	string	"CV", "CC", "" (Blank string)	–	"" (Blank string)	string	–	Script selection and selection clearing	R/W	3
:EDIT	string	"DO Pn,R1"	–	–	string	–	Selected script's contents: program numbers, repetitions, loop start commands, number of times script will repeat, and loop end commands	R/W	3
	string	"DO LOOP Ln"	–	–	string	–		R/W	3
	string	"END LOOP"	–	–	string	–		R/W	3
:EDIT:TITL	string	–	–	–	string	–	Selected script's name	R/W	3

## STATus subsystem

SCPI command		Setting		Default	Response	Effect	Description	R/W	†
Program header	Parameter		Unit						
STAT									
:OPER									
	[:EVEN]	–	–	–	NR1	–	Queries events <sup>1</sup>	R	1
	:COND	–	–	–	NR1	–	Queries the condition of the register <sup>1</sup>	R	1
	:ENAB	NRf	0 to 32767	–	NR1	–	Enable register <sup>1</sup>	R/W	1
	:NTR	NRf	0 to 32767	–	NR1	–	Negative transition filter <sup>1</sup>	R/W	1
	:PTR	NRf	0 to 32767	–	NR1	–	Positive transition filter <sup>1</sup>	R/W	1
	:PRES	–	–	–	–	–	Resets filter registers to default values	W	1
:QUES									
	[:EVEN]	–	–	–	NR1	–	Queries events <sup>2</sup>	R	1
	:COND	–	–	–	NR1	–	Queries the condition of the register <sup>2</sup>	R	1
	:ENAB	NRf	0 to 32767	–	NR1	–	Enable register <sup>2</sup>	R/W	1
	:NTR	NRf	0 to 32767	–	NR1	–	Positive transition filter <sup>2</sup>	R/W	1
	:PTR	NRf	0 to 32767	–	NR1	–	Negative transition filter <sup>2</sup>	R/W	1

- 1 OPERation status register
- 2 QUESTionable status register

## IEEE 488.2 Common Commands

IEEE 488.2 common commands	Parameter	Description	R/W
*CLS	–	Clears all the event registers.	W
*ESE	NR1	Sets or queries the event status enable register bits.	R/W
*ESR	–	Queries the event status register.	R
*IDN	–	Queries the identification string (manufacturer information).	R
*OPC	–	Causes the PBZ to generate the operation complete message in the event status register when all pending device operations have finished.	R/W
*OPT	–	Queries the options that are installed in the PBZ.	R
*PSC	NR1	Sets or queries whether the service request enable register and the event status enable register are reset to their default values when the PBZ's power is reset.	R/W
*RCL	–	Aborts measurement and loads the settings that have been saved to setup memory.	W
*RST	–	Performs a device reset. Configures the PBZ to a known condition independent from the usage history of the device.	W
*SAV	NR1	Saves the current settings to setup memory.	W
*SRE	NR1	Sets the service request enable register.	R/W
*STB	–	Queries the contents of the status byte register and the master summary status message.	R
*TRG	–	Trigger command.	W
*TST	–	Executes a self-test.	R
*WAI	–	Prevents the PBZ from executing subsequent commands or queries until all operations that are in standby have completed.	W

# 15 List of Errors

## Command errors

An error 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 causes the Command Error bit (bit 5) in the event status register to be set.

Error code	Error message description
-100 Command error	Generic syntax error.
-101 Invalid character	A syntactic element contains an invalid character.
-102 Syntax error	An unrecognized command or data type was encountered.
-103 Invalid separator	The parser was expecting a separator and encountered an illegal character.
-104 Data type error	The parser recognized a data element different than one allowed.
-105 GET not allowed	A Group Execute Trigger was received in a program message.
-108 Parameter not allowed	More parameters were received than expected for the header.
-109 Missing parameter	Fewer parameters were received than required for the header.
-110 Command header error	An error was detected in the header.
-120 Numeric data error	Generated when parsing a data element which appears to be numeric, including the nondecimal numeric types.
-130 Suffix error	Generated when parsing a suffix.
-131 Invalid suffix	The suffix does not follow the syntax, or the suffix is inappropriate for the PBZ.
-134 Suffix too long	The suffix contained more than twelve characters.
-138 Suffix not allowed	A suffix was encountered after a numeric element that does not allow suffixes.
-140 Character data error	Generated when parsing a character data element.
-141 Invalid character data	Either the character data element contains an invalid character, or the element is not valid.
-144 Character data too Long	The character data element contains more than twelve characters.
-148 Character data not allowed	A legal character data element was encountered where prohibited by the PBZ.
-150 String data error	Generated when parsing a string data element.
-160 Block data error	Generated when parsing a block data element.
-170 Expression error	Generated when parsing an expression data element.
-180 Macro error	Generated when defining a macro or executing a macro.



## Execution errors

An error 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 causes the Execution Error bit (bit 4) in the event status register to be set.

Error code	Error message description
-200 Execution error (generic)	A generic PBZ error.
-203 Command protected	Password protected program or query command cannot be executed.
-210 Trigger error	A trigger error.
-211 Trigger ignored	A trigger was received but ignored.
-213 Init ignored	A measurement initiate operation was ignored because measurement is in progress.
-214 Trigger deadlock	A deadlock occurred because a query was received before the software trigger.
-220 Parameter error	A program data element related error occurred.
-221 Settings conflict	A command was received that the PBZ cannot execute in its current condition.
-222 Data out of range	Program value was outside the legal range.
-223 Too much data	Too many parameters were received for the requirements.
-224 Illegal parameter value	Received invalid parameter data.
-230 Data corrupt or stale	Received a data query before the measurement completed.
-241 Hardware missing	Cannot be executed because the optional hardware is not installed.

## Device-specific errors

Error code	Error message description
-330 Self-test failed	The self-test failed.
-360 Communication error	A communication error occurred.
-362 Framing error in program message	The stop bit cannot be detected.
-363 Input buffer overrun	An overrun occurred in the input buffer.
-365 Time out error	A time out error occurred.

## Query errors

An error 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 causes the Query Error bit (bit 2) in the event status register to be set.

Error code	Error message description
-400 Query error (generic)	A generic PBZ error.
-410 Query INTERRUPTED	Received a new command before the response was read.
-420 Query UNTERMINATED	The controller attempted to read the response after the device received an unsupported query or has not received a query. The -100 "Command error" and this error are stored in the error queue. The controller will time out.
-430 Query DEADLOCKED	The error queue, input buffer, and output buffer are full when sending large binary data as a response, and the transmission timing is off.
-440 Query UNTERMINATED after indefinite response	Received a separate query in semicolon-delimited format after a query that returns a response in an indefinite form. Example: *IDN?;SYST:ERR?

## Operation complete event errors

An error in the range [-899, -800] is used when the PBZ wants to report an IEEE 488.2 operation complete event. This event occurs when the instrument's synchronization protocol, having been enabled by an \*OPC command, completes all selected pending operations.

The occurrence of this event causes the Operation Complete bit (bit 0) in the event status register to be set.

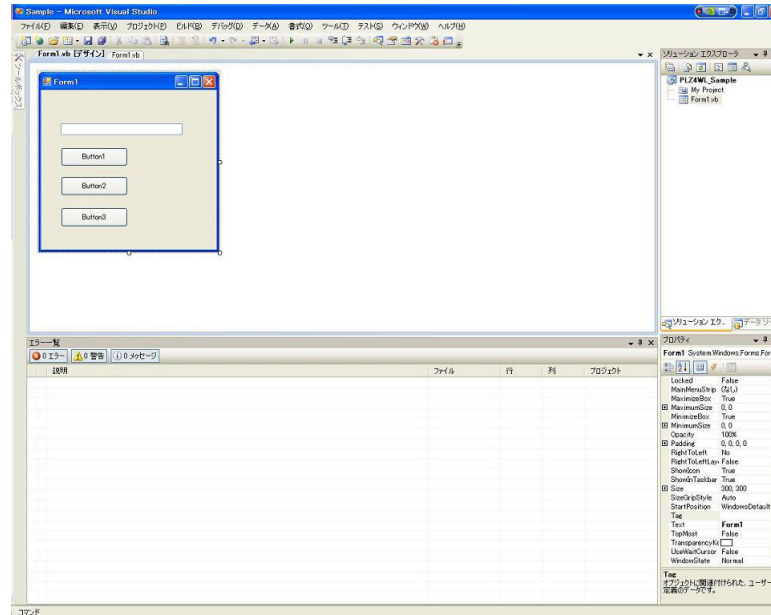
Error code	Error message description
-800 Operation complete	All selected pending operations in accordance with the IEEE 488.2, 12.5.2 synchronization protocol have completed.

## PBZ-specific errors

Error code	Error message description
101 Operation denied while in LOCAL state	The operation was denied because the PBZ is in local mode.
102 Operation denied while in OUTPUT ON state	The operation was denied because the output is on.
103 Operation denied while in PROTECTION state	The operation was denied because a protection feature is activated.
104 Operation denied while in SLAVE mode	The operation was denied because the PBZ is set to slave mode in parallel operation.
105 Operation denied due to incompatible FUNCTION SOURCE	The operation was denied because the signal source is incompatible.
106 Operation denied due to PROGRAM RUNNING	The operation was denied because a sequence program is being executed.
107 Operation denied due to Empty PROGRAM	The operation was denied because there is no sequence program.

# 16 Tutorial (Visual Basic 2008)

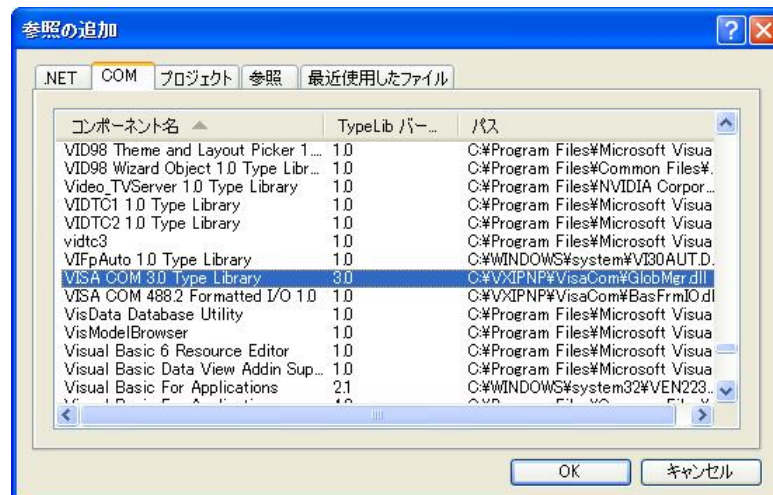
Communicate with the PBZ series using VISA through GPIB, RS232C, USB or LAN.



## Setting the "Project"

At first, set the communication driver (VISA library) for the "Project".

Click "Project" on the menu bar, then select "Add Reference" to refer "VISA COM 3.0 Type Library".



## Communication through GPIB, RS232C, USB, or LAN

After setting the communication driver, it enables to communicate by GPIB, RS232C, USB, or LAN.

The following describes the communication procedure for via VISA.

### Open the VISA

To communicate with the device of GPIB, RS232C, USB, or LAN via VISA, it requires to open the VISA first. When opening the VISA, specify the I/O resource.

Example : To open VISA by using USB

```
Set rm = CreateObject("VISA.GlobalRM")
Set msg = rm.Open("USB::0x0B3E::0x1012::00000001::INSTR", NO_LOCK, 0, "")
```

"USB::0x0B3E::0x1014::00000001::INSTR" is the I/O resource.

The I/O resource is specified by the following constructions. The part indicated with [ ] can be abbreviated. Enter the appropriate value in the part specified in oblique characters.

GPIB		GPIB[ <i>board</i> ]:: <i>PrimaryAddress</i> :: <i>SecondaryAddress</i> ][:INSTR] Example : The primary address 3 of the measuring instrument connected to GPIB0. GPIB0::3::INSTR
Serial (RS232C)		ASRL[ <i>board</i> ][:INSTR] Example : The measuring instrument connected to the serial port COM1. ASRL1::INSTR
USB		USB[ <i>board</i> ]:: <i>VendorID</i> :: <i>ProductID</i> :: <i>SerialNumber</i> [: <i>InterfaceNumber</i> ][:INSTR] Example : The USNTMC measuring instrument having with the vendor ID (VID)2878, Product ID(PID)4114 and serial number "00000001". USB0::0x0B3E::0x1012::00000001::INSTR
LAN <sup>1</sup>	VXI-11	TCPIP[ <i>board</i> ]:: <i>LAN device name</i> [: <i>inst0</i> ][:INSTR] The measuring instrument whose IP address (LAN device name) is 169.254.7.8. TCPIP::169.254.7.8::INSTR You can also set the LAN device name using the host name.
	SCPI-RAW	TCPIP[ <i>board</i> ]:: <i>LAN device name</i> :: <i>portno</i> ::SOCKET Example :The measuring instrument whose IP address (LAN device name) is 169.254.7.8. (The "portno" setting of the PCR is normally 5025.) TCPIP::169.254.7.8::5025::SOCKET You can also set the LAN device name using the host name.

 See p. 11

- 1 The hostname must be a valid mDNS hostname (a Bonjour hostname that ends in ".local") or a DNS hostname that is managed by an external DNS server (a full-qualified domain name—FQDN). If you are using an mDNS hostname, Apple Bonjour (alternatively, iTunes or Safari) must be installed on your PC.

For VISA, the alias can be used for the I/O resource.

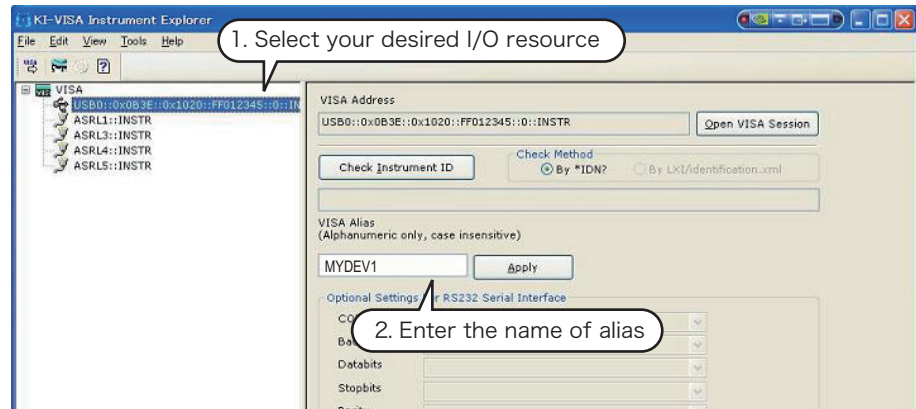
When using the alias for the I/O resource, even the alias name to be hard coating directly in the application, it can be easily converted to the appropriate I/O resource name.

Example : When using the alias (MYPIA) for the I/O resource.

```
Set msg = rm.Open("MYPIA", NO_LOCK, 0, "")
```

When the alias is used, the actual I/O resource is specified by such an external configuration table.

When using the USB (example for KI-VISA)



In case of using VISA other than KI-VISA, please refer to the applied VISA manual.

### Controlling the devices

Next, using such a "Read", "Write" to control the devices.

Example:

```
msg.WriteString ("FUNC:MODE CV")      'Specify a CV mode.
msg.WriteString ("VOLT 10.0")         'Set 10.0 V.
msg.WriteString ("OUTP 1")           'Output ON.
```

### Closing the VISA.

Close the VISA at the end.

A command for "Open" and "Close" of the VISA is required only once in the program.

```
msg.Close
```

## Sample program

```
Imports Ivi.Visa.Interop

Public Class Form1

    Dim rm As ResourceManager
    Dim msg As IMessage

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        rm = CreateObject("VISA.GlobalRM")
        'msg = rm.Open("GPIB0::1::INSTR", AccessMode.NO_LOCK, 0, "") 'Example: GPIB
        'msg = rm.Open("MYDEV1", AccessMode.NO_LOCK, 0, "") 'Example: Using a alias
        'msg = rm.Open("TCPIP::169.254.178.141::INSTR", AccessMode.NO_LOCK, 0, "") 'Example: LAN
        msg = rm.Open("USB0::0x0B3E::0x1012::FF012345::INSTR", AccessMode.NO_LOCK, 0, "") 'Example: USB
    End Sub

    'Query the instrument identity
    Private Sub cmdIdn_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdIdn.Click
        msg.WriteString("*IDN?")
        TextBox1.Text = msg.ReadString(256)
    End Sub

    'Set the operation mode and voltage
    Private Sub cmdCurr_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdCurr.Click
        msg.WriteString("OUTP 0") 'Output OFF
        msg.WriteString("FUNC:MODE CV") 'Specify a CV mode
        msg.WriteString("VOLT 10.0") 'Set 10.0 V.
        msg.WriteString("OUTP 1") 'Output ON
    End Sub

    'Query the instrument identity
    Private Sub cmdMeas_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdMeas.Click
        msg.WriteString("MEAS:CURREN?")
        TextBox1.Text = msg.ReadString(256)
    End Sub

    Private Sub Form1_Disposed(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Disposed
        msg.Close()
    End Sub
End Class
```

# INDEX

<b>B</b>	
boolean data .....	17
<b>C</b>	
character data .....	17
command .....	
description .....	19
hierarchy .....	14
syntax .....	15
units .....	18
common commands .....	16, 20
compound commands .....	15
CONDition register .....	72
<b>D</b>	
default state .....	81
DEFAULT switch .....	10
<b>E</b>	
ENABLE register .....	72
EVENT register .....	72
event status register .....	75
<b>F</b>	
firmware version .....	2
<b>G</b>	
GPIO address, setting .....	7
GPIO interface .....	7
<b>I</b>	
I/O library .....	4
IEEE 488.1 serial polling .....	74
IEEE 488.2 common command .....	74
IEEE Std 488.2 .....	3
interface .....	3
<b>K</b>	
KI-VISA .....	4
<b>L</b>	
LAN Interface .....	9
LAN RESET switch .....	10
LAN settings .....	9
list of errors .....	96
<b>M</b>	
measurement instrument interface standards .....	3
messages .....	14
list .....	87
<b>N</b>	
non-numeric parameters .....	17
NRf .....	17
Numeric .....	17
numeric parameters .....	17
<b>O</b>	
OPERation status register .....	76
<b>P</b>	
parameters .....	17
<b>Q</b>	
queries .....	16
QUESTionable status register .....	78
<b>R</b>	
remote control .....	
list of errors .....	96
remote interfaces .....	3
RS232C interface .....	5
RS232C protocol .....	6
RS232C settings .....	6
<b>S</b>	
SCPI .....	3, 14
SCPI command syntax .....	14
SCPI status registers .....	73
SEQUence1 .....	45
SEQUence2 .....	47
SEQUence3 .....	49
SEQUence4 .....	51
status byte register .....	74
status register .....	72
string data .....	17
<b>T</b>	
terminator .....	16
transition filter .....	72
<b>U</b>	
USB interface .....	8
USB settings .....	8
<b>V</b>	
version .....	2
VISA library .....	4

If you find any misplaced or missing pages in the manuals, they will be replaced. If the manual gets lost or soiled, a new copy can be provided for a fee. In either case, please contact your Kikusui agent or distributor. At that time, inform your agent or distributor of the "PART NO." written on the front cover of this manual.

Every effort has been made to ensure the accuracy of this manual. However, if you have any questions or find any errors or omissions, please contact your Kikusui agent or distributor.

After you have finished reading this manual, store it so that you can use it for reference at any time.

## KIKUSUI ELECTRONICS CORP.

---

1-1-3 Higashiyamata, Tsuzuki-ku, Yokohama,  
224-0023, Japan  
Tel: +81-45-482-6353 Fax: +81-45-482-6261



Website

<http://www.kikusui.co.jp/en>