

# Function Generators GX 310P GX 320E

## Remote Programming Manual



**metrix**

CHAUVIN-ARNOUX

Parc des Glaisins

6, avenue du Pré de Challes

F - 74940 ANNECY-LE-VIEUX

Tél. +33 (0)4.50.64.22.22 - Fax +33 (0)4.50.64.22.00



|  |           |
|--|-----------|
| <b>Introduction.....</b>   | <b>5</b>  |
| Presentation .....   | 5         |
| Connection .....   | 5         |
| <i>Connection through « USB ».....</i>                           | <i>5</i>  |
| <b>Introduction (cont'd).....</b>                                | <b>6</b>  |
| "CP210x USB to UART Bridge Controller" driver installation ..... | 6         |
| Port COM Identification.....                                     | 6         |
| Using TeraTerm to drive the instrument.....                      | 6         |
| Connecting several GX 3x0 .....                                  | 8         |
| Programming an IP address for the device.....                    | 10        |
| Use TeraTerm to program your GX 320E via Ethernet .....          | 12        |
| Rear panel.....  | 13        |
| GX 310P.....   | 13        |
| GX 320E.....   | 13        |
| Drivers LabView and LabWindows .....                             | 14        |
| <b>SCPI Protocol .....</b>                                       | <b>14</b> |
| Programming convention.....                                      | 14        |
| <i>Tree structure .....</i>                                      | <i>14</i> |
| Command syntax.....  | 14        |
| <i>Common commands .....</i>                                     | <i>14</i> |
| <i>Specific commands .....</i>                                   | <i>14</i> |
| <i>Keywords.....</i>   | <i>14</i> |
| <b>SCPI Protocol (cont'd).....</b>                               | <b>15</b> |
| <i>Parameter format.....</i>                                     | <i>15</i> |
| <i>Recall.....</i>   | <i>15</i> |
| <b>SCPI Protocol (cont'd).....</b>                               | <b>16</b> |
| <i>Terminator .....</i>  | <i>16</i> |
| <i>Response .....</i>  | <i>16</i> |
| <b>Main Commands .....</b>                                       | <b>17</b> |
| Instrument power off.....  | 17        |
| Activation of the MAIN OUT output.....                           | 17        |
| Generator function selection .....                               | 17        |
| Display contrast setting.....                                    | 18        |
| List of used system files (GX 320) .....                         | 18        |
| Configuration saving (GX 320).....                               | 18        |
| Configuration recall (GX 320) .....                              | 18        |
| Cancellation of a config. file (GX 320) .....                    | 18        |
| <b>CONTinuous Function .....</b>                                 | <b>19</b> |
| Selection of the signal wave shape .....                         | 19        |
| Setting the signal frequency .....                               | 19        |
| Amplitude unit selection .....                                   | 20        |
| Setting the offset signal .....                                  | 20        |
| Setting the high logic level .....                               | 21        |
| Setting the low logic level.....                                 | 21        |
| <b>SWEEP Function .....</b>                                      | <b>22</b> |
| SWEEP source selection.....                                      | 22        |
| SWEEP mode selection.....  | 22        |
| SWEEP type selection.....  | 22        |
| Setting the start/end frequencies .....                          | 22        |
| Setting the SWEEP time.....                                      | 23        |
| <b>MODUL Function (GX 320) .....</b>                             | <b>24</b> |
| Setting AM rate .....  | 24        |
| Selection of AM source.....                                      | 24        |

|  |           |
|--|-----------|
| Selection of FM source .....                                     | 24        |
| Setting start/end frequencies in FM .....                        | 24        |
| <b>SHIFT Key Function (GX 320) .....</b>                         | <b>25</b> |
| Selection of the SHIFT KEYING source .....                       | 25        |
| Setting start/end phases in PSK.....                             | 25        |
| Setting start/end frequencies in FSK .....                       | 25        |
| <b>BURST Function (GX 320).....</b>                              | <b>26</b> |
| BURST source selection.....                                      | 26        |
| Setting the pulse number to be generated .....                   | 26        |
| Setting the delay between pulse trains .....                     | 26        |
| Manual launching of a pulse train .....                          | 26        |
| <b>SYNChronization Function (GX 320).....</b>                    | <b>27</b> |
| Setting the dephasing .....                                      | 27        |
| <b>FREQ Frequencymeter .....</b>                                 | <b>28</b> |
| Reading the frequency value .....                                | 28        |
| <b>GATE mode management (GX 320).....</b>                        | <b>29</b> |
| Activating GATE mode .....                                       | 29        |
| <b>Help and Error .....</b>                                      | <b>30</b> |
| Help .....   | 30        |
| Error .....  | 30        |
| * Command error:               (-199 to -100) .....              | 30        |
| * Execution errors:       (-299 to -200) .....                   | 31        |
| * Specific instrument errors:               (-399 to -300) ..... | 31        |
| * Query errors:               (-499 to -400) .....               | 31        |
| <b>IEEE 488.2 common commands.....</b>                           | <b>32</b> |
| Introduction.....  | 32        |
| Events and status management .....                               | 32        |
| The Registers .....  | 32        |
| <i>Status registers.....</i>                                     | <i>33</i> |
| <i>Service request mask register.....</i>                        | <i>33</i> |
| <i>Event register .....</i>                                      | <i>33</i> |
| <i>Event mask register.....</i>                                  | <i>34</i> |
| <i>Event mask register.....</i>                                  | <i>35</i> |
| <i>Event register .....</i>                                      | <i>35</i> |
| <i>Service demand mask register.....</i>                         | <i>36</i> |
| <i>Status register .....</i>                                     | <i>37</i> |
| <b>SCPI commands .....</b>                                       | <b>39</b> |

## Introduction

### Presentation

The programming instructions comply with standard IEEE488.2, and the SCPI protocol (Standard Commands for Programmable Instruments). They provide the user with the possibility of checking the instrument remotely from simple standard controls.

Communication between a controller and a generator enables users :

- to configure the instrument
- to measure frequencies

### Connection

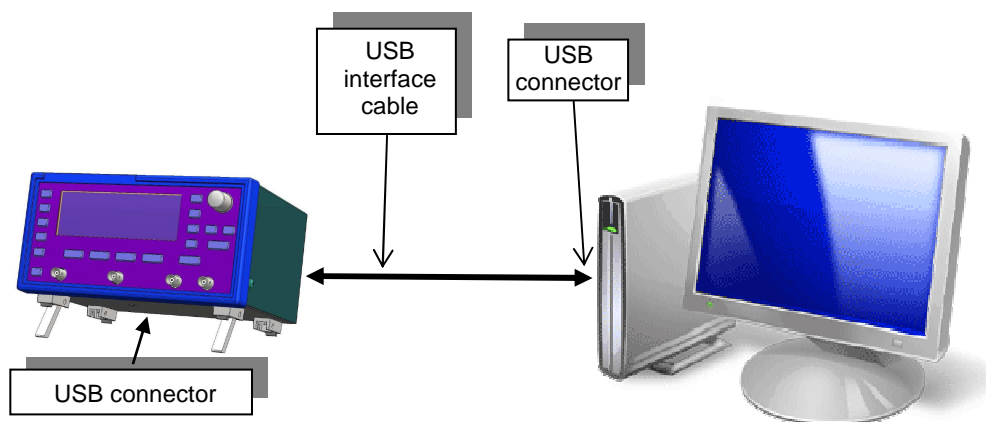
#### Connection through « USB »

The dialog between the instrument and the PC can be realized through an USB link.

This dialog is only possible if the CP210x driver is correctly installed, the USB peripheral is recognized and a new COM port appears in the system parameters of the PC.

On the PC, configure this COM port with following parameters:

- 19 200 baud speed
- 8 data bits
- no parity
- 1 stop bit
- hardware flow control (RTS / CTS)



***The CP210x driver must be installed before connecting the GX 310P or GX 320E generator.***

***If the USB device is not recognized when connecting the cable :***

- ***give up searching the Windows driver,***
- ***disconnect the generator,***
- ***install the driver as described hereafter.***

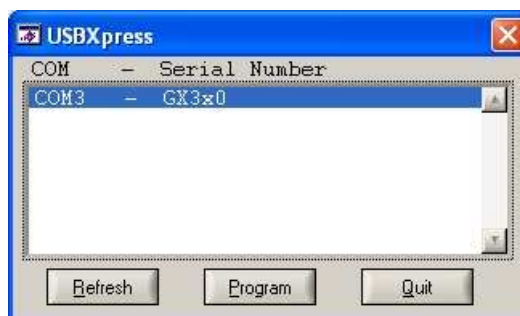
## Introduction (cont'd)

### "CP210x USB to UART Bridge Controller" driver installation

- Do not connect the generator to the USB port of the PC before installing the "CP210x USB to UART Bridge Controller" driver.
- The driver is automatically installed when you set up the utility USBXpress. This application allows among other things :
  - to identify the COM port created for the instrument
  - to change the identifier of the generator port in order to differentiate both generators you wish to connect to a PC (see section §. Connecting Multiple GX3x0).
- Insert the CD-ROM supplied. In the presentation page (readme.html), select the subject 'Windows Installer' of the line USBXpress.
- Follow the install instructions.
- You now can connect your instrument.

### Port COM Identification

- Run the USBXpress utility, or, if needed, press the 'Refresh' key. A new line appears with the COM port number and identifier of USB port (here, the instrument is connected to the PC via the COM3 port. His identifier is 'GX3x0').

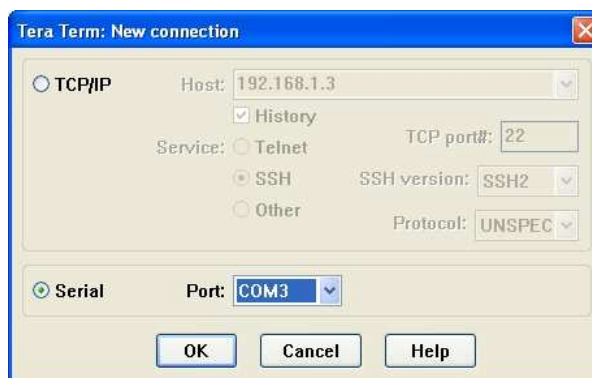


### Using TeraTerm to drive the instrument

Once the driver CP210x is installed and the new COM port identified, you may drive your instrument with TeraTerm or any terminal emulator whatsoever.

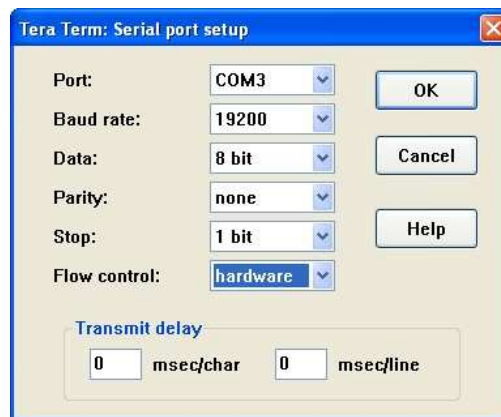
TeraTerm is a free software available on the CD ROM.

- Insert the CD-ROM supplied.
- In the presentation page (readme.html), select 'Windows Installer' in the line TeraTermPro V4.52, and follow the install .
- RunTeraTerm and create a new connection (File/New menu connection...).
- Select the serial port identified above (here COM3).

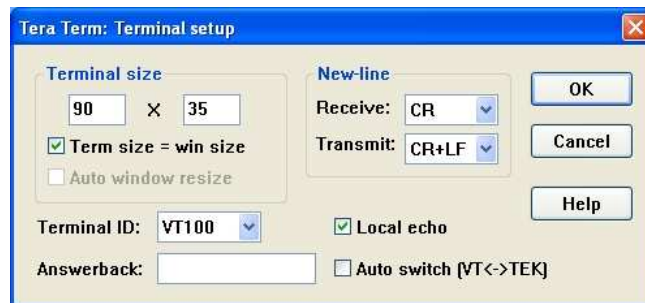


## Introduction (cont'd)

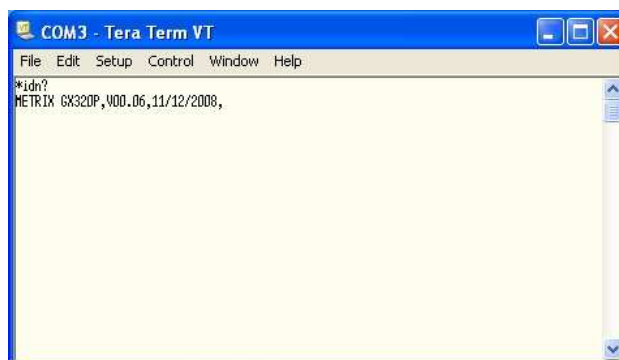
- Set the serial link (Setup menu / Serial ports...) as below :



- Set the Terminal as follows (Setup menu / Terminal ...) :



- TeraTerm is now set up.  
This configuration can be saved in the Setup / Save Setup menu ...
- You can now enter the commands in the window.  
For instance : to the query '\*idn?' the instrument must give its identification and its software version :



## Introduction (cont'd)

### Connecting several GX 3x0

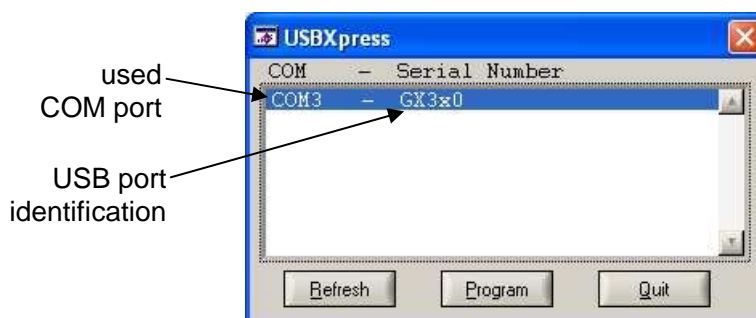
By default, all the instruments are set up with the same "GX3x0" identifier. Thus, on one PC, **GX 320E** and **GX 310P** generators are always associated with the same COM port.

However, connecting two **GX** with the same identifier on a PC leads to a resource conflict.

If you wish to connect several **GX** on the same PC, you must differentiate the instruments using the utility USBXpress.exe.

Be careful : the USB driver must be previously installed (refer to section §. "CP210x USB to UART Bridge Controller" driver installation).

Connect **only one GX** on a PC's USB port. Run 'USBXpress.exe' :



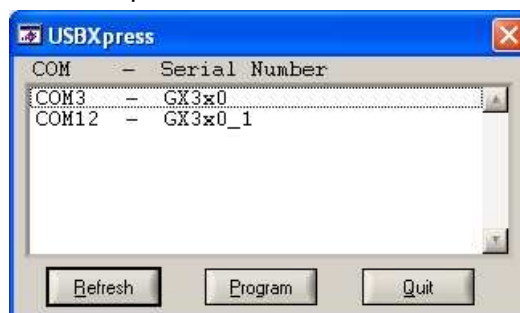
The identifier to be modified is : 'Serial Number'. Enter a different value by pressing "Program" or double-clicking on the line :



For instance : enter 'GX3x0\_1' and press 'OK' key to be programmed in the instrument. You have 20 characters to identify the USB port on the instrument.

Unplug the programmed generator and start the process again with another instrument. Press 'Refresh' key to update. For each instrument, enter a different value.

After programming, each instrument connected to a PC's USB port will be assigned a different COM port.





## Introduction (cont'd)

through  
"ETHERNET"

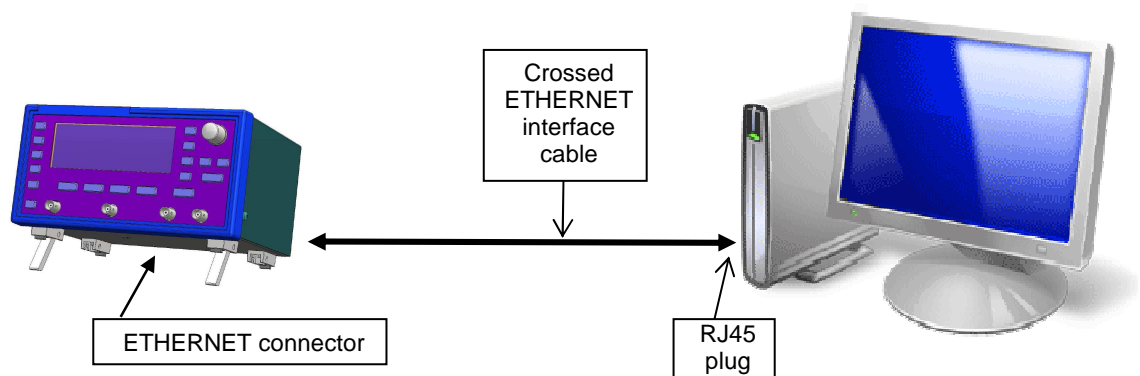


**Warning ! To use the ETHERNET link, the USB cable must be disconnected.**

The dialogue between the generator and the PC is realized via a serial Ethernet adapter. This is a Lantronix XPort module.

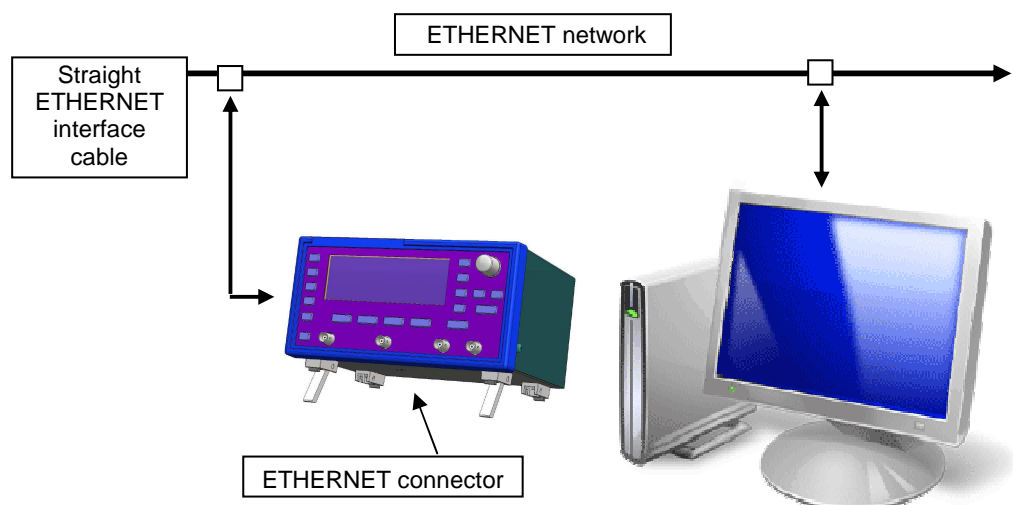
### crossed cable

- Connect the crossed ETHERNET interface cable directly to the PC.
- Set the connection with a terminal (Port TELNET : 23) to the IP address which has been defined in the generator.



### straight cable

- Connect the generator to the PC network through a Hub with the straight ETHERNET interface cable.
- Connect a terminal (TELNET Port : 23) to the IP address defined on the generator.



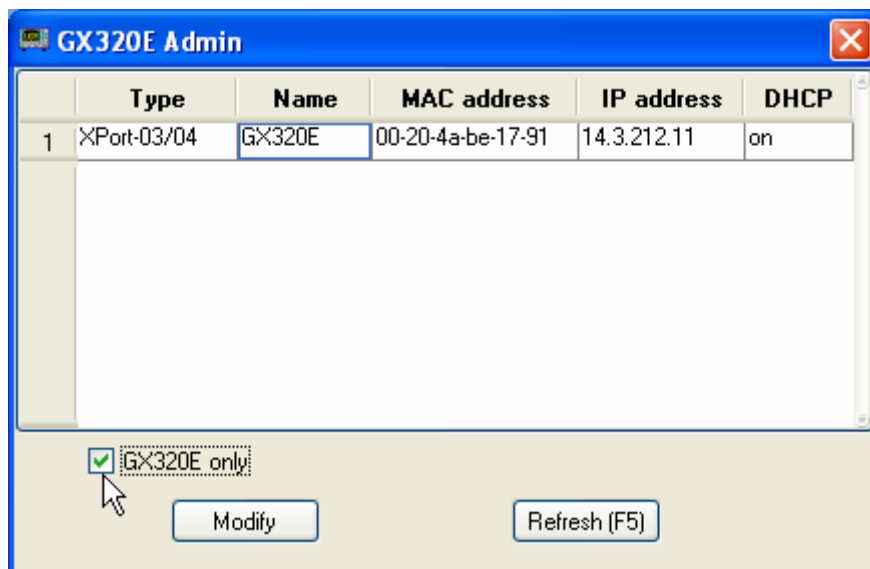
## Introduction (cont'd)

### *Programming an IP address for the device*

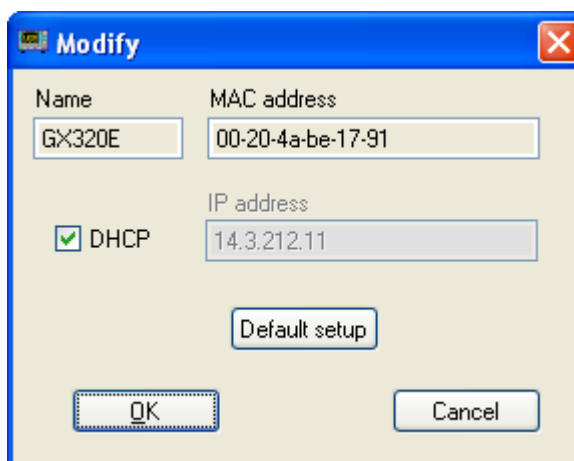
- Insert the supplied CD-ROM and, in the readme.html page presentation, select the item "Windows Installer" of the GX320E-Admin line.
- Follow the installation instructions.

Connect the generator to the power supply (  red button lit).

Launch the "GX320E-Admin.exe" utility, the next window opens:



- Check 'GX320E only' to let GX320E appear only.  
The 'Refresh (F5)' button lets re-examine the network to find the devices.
- Select an item from the line of the instrument to be programmed (the identification of the instrument is done with the MAC address whose value is displayed at the back of the instrument) and click 'Modify'  
The following window opens:



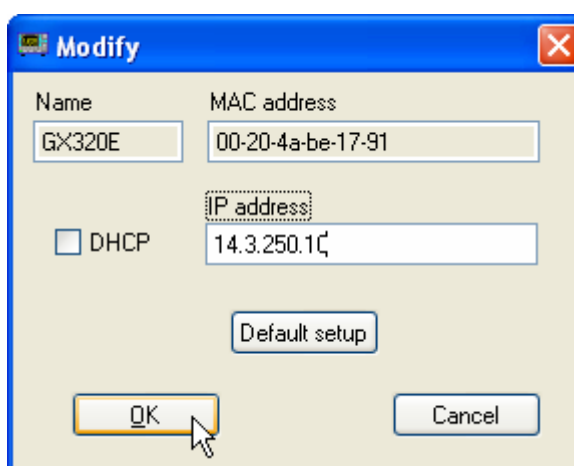
## Introduction (cont'd)

Here the address is automatically provided via DHCP or AutoIP.

**Note: Clicking the Default setup 'window allows to reprogram the default configuration in the generator.**

If you want to enter an address, uncheck the 'DHCP' and type your new address.

**Note: Make sure that address is not already used on the network, as it might disrupt the latter.**



Modify

Name: GX320E

MAC address: 00-20-4a-be-17-91

☐ DHCP

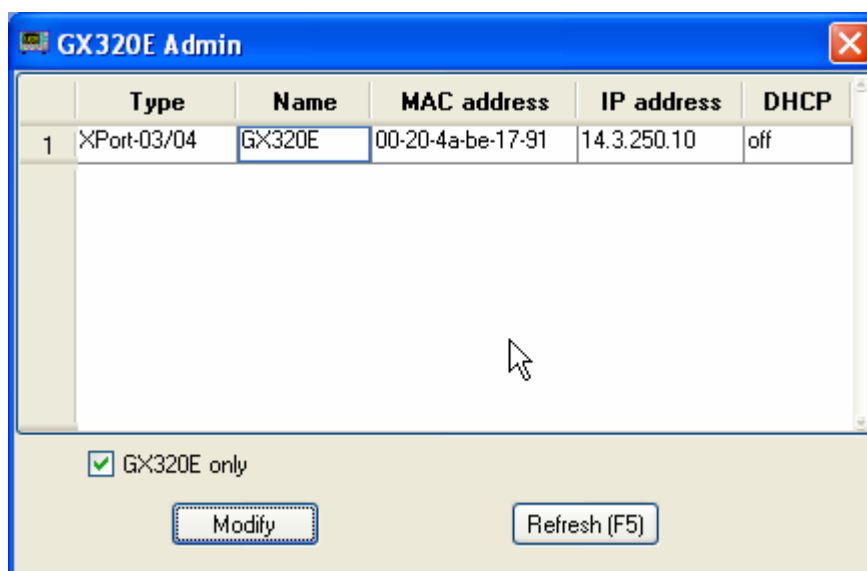
IP address: 14.3.250.10

Default setup

OK Cancel

- Press OK to validate your entry.

After programming, the window closes and you return to the main window (note the change of address).



GX320E Admin

|   | Type        | Name   | MAC address       | IP address  | DHCP |
|---|-------------|--------|-------------------|-------------|------|
| 1 | XPort-03/04 | GX320E | 00-20-4a-be-17-91 | 14.3.250.10 | off  |

☒ GX320E only

Modify Refresh (F5)

## Introduction (*cont'd*)

*Use TeraTerm to  
program your  
GX 320E  
via Ethernet*

Once your IP address is correctly programmed on the instrument, you can start Tera Term and create a new connection (see above ...).

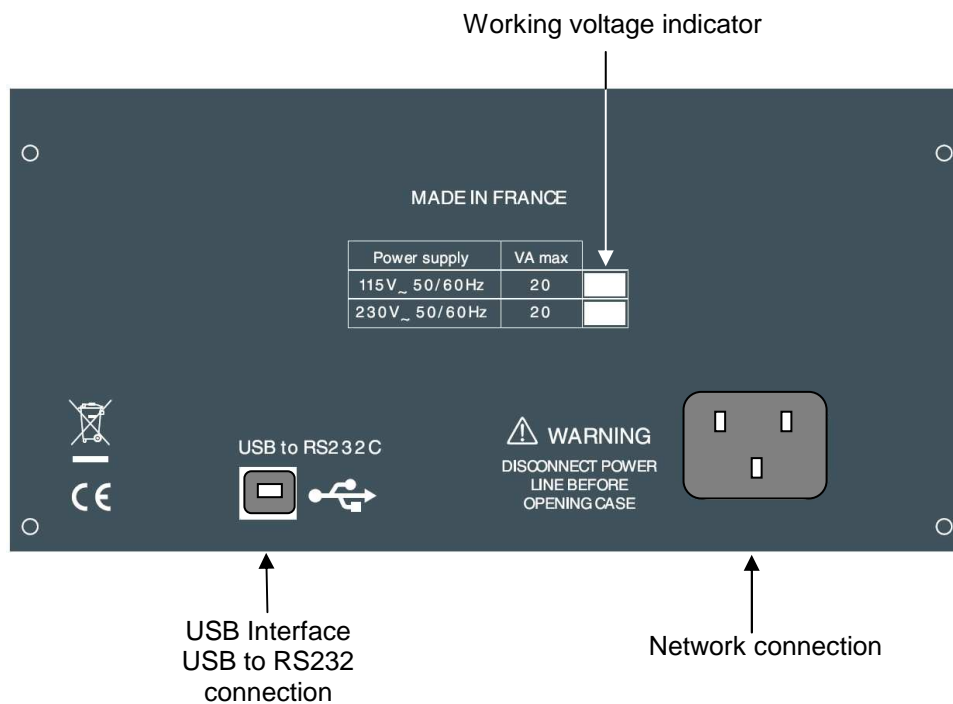
Select TCP / IP, enter the IP address of your instrument (14.3.250.10 here), and select Telnet port 23 :



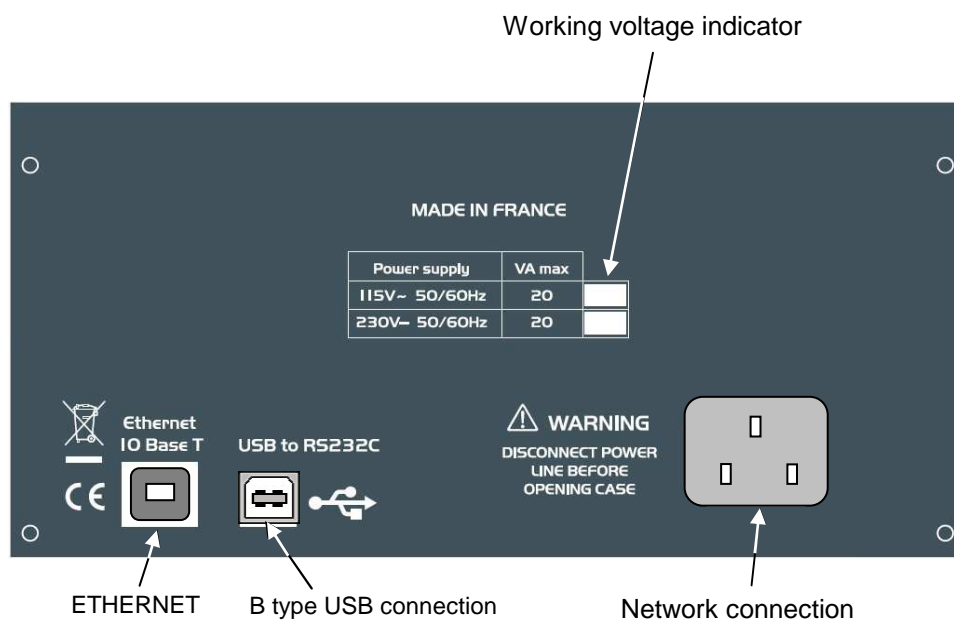
## Introduction (cont'd)

### Rear panel

#### GX 310P



#### GX 320E



## Drivers LabView and LabWindows

These drivers are on the CD ROM, they provide an SCPI commands interface to enable the realization of software with both development environments.

## SCPI Protocol

### Programming convention

#### Tree structure

The command tree diagram includes all the commands specific to the instrument.

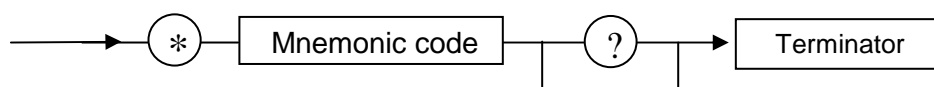
Common commands (standard IEEE 488.2) are listed separately, since they do not affect the position of the syntax analyzer in the tree.

When the terminator **<NL>** is sent to the instrument, the analyzer is positioned at root level.

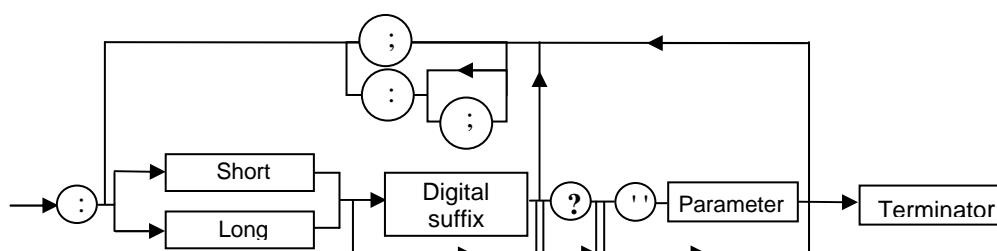
When the analyzer is in a directory, both separators ' ; ' are necessary to return to the root.

### Command syntax

#### Common commands



#### Specific commands



#### Keywords

The brackets ([ ]) frame a keyword which is optional during programming; i.e. the instrument will execute the command whether the keyword is optional or not. Uppercase and lowercase are used to differentiate the short form of the keyword (uppercase letters) and the long form (whole word).

The instrument accepts the uppercase or lowercase letters without distinction.



**SWE:SOUR INT** is equivalent to **SOURce:SWEep:SOURce INTernal**

#### Parameters

The parameters, if any, are separated from the keyword by a space (' '). A command can accept parameters of a defined type, a literal expression or a combination of both.

- < > The defined-type parameters are marked by the characters.
- ( [ ] ) The brackets mean that the parameters are optional.
- { } The accolades define the list of parameters allowed.

- ( | ) The vertical bar may be read as an "or", it is used to separate the various possible parameters.

## SCPI Protocol (*cont'd*)

### Separators

- ' : ' descends in the next directory or returns under the root, if preceded by a ' ; '.
- ' ; ' separates 2 commands in the same directory or marks the end of a directory command, enabling a return to the root level by adding the separator ' : '.
- ' ' (space) separates the keyword from the following parameter.
- ' , ' separates one parameter from the other.

### Parameter format

The parameters can be key words, numeric values, character chains or numeric expressions.

### Key words

These parameters have two forms of call, as for the instructions : the shortened form (in capital letter) and the whole form (shortened form plus complement into small letter).

Thus, for certain commands, the parameters are the following :

- **ON, OFF** corresponding to the boolean values (1,0)
- **CONTinuous, FSK, PSK, SWEep, AM, FM, FREQuencymeter, SYNCMaster, SYNCSlave and BURST** for the generator function

### Numeric values

They have several possible formats :

### NRF (flexible Numeric Representation)

In the case of physical quantity, these numbers can be followed of a sub-multiple or a multiple of its unit.

### Recall

Interpreter does not make any difference between capital and small letters.

 *Example* : 1 milli second can be written either 1ms, 0.001, 1e-6s, 1e-3s

This parameter can also be replaced by the following key words :

- **MAXimum, MINimum** to get extreme values of the parameter
- **UP, DOWN** to get the value following or preceding the current status of the parameter

### Units

- V** volt (Voltage)
- S** second (Time)
- PCT** percent (Percentage)
- Hz** hertz (Frequency)

### Multiples and sub-multiples

- MA** mega:  $10^{+6}$
- K** kilo:  $10^{+3}$
- M** milli:  $10^{-3}$
- U** micro:  $10^{-6}$
- N** nano:  $10^{-9}$
- P** pico:  $10^{-12}$

- NR1** The parameter is a signed integer number.  
✎ *Example* : 10
- NR2** The parameter is a signed real without exponent.  
✎ *Example* : 10.1
- NR3** The parameter is a signed real expressed with a mantissa and a signed exponent.  
✎ *Example* : 10.1e-3

## SCPI Protocol (*cont'd*)

**Chains of Characters** They are continuations of letters and figures framed by quotation marks " ".

### Terminator

**<NL>** **<NL>** is a general term for a terminator.  
NL is the character CR (code ASCII 13 or 0x0D).  
A line of command should not exceed 80 characters ; it ends with a terminator.

### Response

The response can be made up of several elements separated between them by a comma ', '. The last element is followed by the terminator **<NL>**.

The data are of several natures :

**Key words** They are the same ones as those used in parameter, but here, only the shortened form is returned.

**Numeric values** They have three possible formats : NR1, NR2 and NR3.

**Chains of characters** There is no difference compared to the parameters. If the chain contains a key word, it is returned in shortened form.



# Detailed description of commands

## Main Commands

---

### Instrument power off

SYSTem:POWer: (command/query)

The command **SYST:POW {1|0|ON|OFF}** switches the instrument from ON mode to STANDBY mode, and reciprocally.

To the question **SYST:POW?** the instrument returns its operating mode (1 if ON, 0 if STANDBY).

---

### Activation of the MAIN OUT output

OUTPut[:STATe] (command/query)

The command **OUTPut {0|1|ON|OFF}** activates or not the MAIN OUT output.

- **ON|1** : the output is activated, the signals come from the MAIN OUT BNC.
- **OFF|0** : the output is locked.

To the question **OUTPut?** the apparatus returns the current state of the MAIN OUT output (1 if activated, 0 if not).

---

### Generator function selection

DEVice:MODE (command/query)

The command

**DEV:MOD{CONTinuous|FSK|PSK|SWEep|AM|FM|FREQuencymeter|SYNCMaster|SYNCSlave|BURST}** selects the generator function to be used.

To the query **DEV:MOD?** the apparatus returns the current generator function.

☞ *Only the **CONT**, **SWEEP** and **FREQ** functions are available on the GX 310.*

## Display contrast setting

DISPlay:CONTRast (command/query)

The command **DISP:CONT <level>** sets the contrast value of the LCD screen.

**<level>** is the contrast value. It is a decimal value without exponent, between 0 and 1 (1 being the max. contrast).

To the query **DISP:CONT?** the apparatus returns the current contrast value in NR2 format.

## List of used system files (GX 320)

MMEMory:CATalog (query)

To the query **MMEM:CAT?** the instrument returns the list of the configurations which are used in memory.

Response format : **<number of used files>,0,<list of used files>**

**<list of used files>** is the list of the system file numbers - not virgin - which are memorized, separated by commas.

## Configuration saving (GX 320)

MMEMory:STORe :STATe <num config> (command)

The command **MMEM:STOR:STAT <num config>** memorizes the current configuration in the file of configuration number **<num config>**.

The preceding file **<num config>** is erased without notice by the current configuration data.

**<num config>** is a number between 1 and 15.

## Configuration recall (GX 320)

MMEMory:LOAD :STATe (command)

The command **MMEM:LOAD:STAT <num config>** replaces the current configuration of the apparatus by that contained in the file of configuration number. If the file is virgin, the command is without effect.

**<num config>** is a number between 0 et 15, the file 0 corresponds to the default configuration of the instrument.



**MMEM:LOAD:STAT 0** is equivalent to \*RST.

## Cancellation of a config. file (GX 320)

MMEMory:DELeTe (command)

The command **MMEM:DEL <num file>** erases the configuration **<num file>** memorized and resets the file in question to the virgin state.

## CONTinuous Function

---

### Selection of the signal wave shape

[SOURce:]FUNCtion  
[:SHAPE] (command/query)

The command **FUNC {SINusoid|SQUare|LOGICal|TRIangle|DC}** selects the shape of the signal wave.

To the question **FUNC?** the apparatus returns the current wave shape of the instrument.

---

### Setting the signal frequency

[SOURce:]FREQuency  
[:STARt] (command/query)

The command **FREQ {<frequency>|MIN|MAX}** enters the signal main frequency or the frequency  $\text{Freq}_{\text{START}}$  in the modes **SWEEP**, **FSK** and **MODUL FM**.

**<frequency>** is a value in the format **<NRf>**, it can be followed or not by a sub-multiple of the unit (Hz). By default, its value is expressed in Hz.

- **MIN** : sets the min. value at frequency.
- **MAX** : sets the max. value allowed by the instrument

To the question **FREQ?** the apparatus returns the signal frequency or  $\text{Freq}_{\text{START}}$  for the functions **SWEEP**, **FSK** and **MODUL FM**. The value is expressed in format **<NR3>** in Hz.

### Setting the signal duty cycle

[SOURce:]PULSe  
:DCYClE (command/query)

The command **PULS:DCYC <duty cycle>** sets the duty cycle of the square and triangular signals.

**<duty cycle >** is an integer number from 10 to 90 which expresses the duty cycle in %.

To the question **PULS:DCYC?** the apparatus returns the current value of the duty cycle. The default value is 50, it is the returned value if the chosen wave signal is sinusoidal.

## Setting the signal amplitude

[SOURce:]VOLTage  
[:LEVel][:IMMediate]  
[:AMPLitude] (command/query)

The command **VOLT <amplitude>** sets the signal amplitude to be generated (except LOGIC signal).

**<amplitude>** is a value in format **<NRf>**, it can be followed or not by a sub-multiple and by the unit (V). By default, the value is expressed in volt, it is the amplitude peak-to-peak value.

To the question **VOLT?** the apparatus returns the value of the Vpp programmed amplitude. It is a value in format **<NR3>** expressed in volt.

---

## Amplitude unit selection

UNIT:VOLTage  
:AMPLitude (command/query)

The command **UNIT:VOLT:AMPL <PTPeak|RMSsquare>** selects the unit of the amplitude value.

- **PTPeak** : displays the amplitude in Vpp.
- **RMSsquare** : displays the amplitude in Vrms.

To the question **UNIT:VOLT:AMPL?** the apparatus returns the current display mode of the amplitude.

---

## Setting the offset signal

[SOURce:]VOLTage  
[:LEVel][:IMMediate]  
:OFFSet (command/query)

The command **VOLT:OFFS <offset>** sets the offset of the signal to be generated (except signal LOGIC).

**<offset>** is a value in format **<NRf>**, it can be followed or not by a sub-multiple and by the unit (V). By default, the value is expressed in volt.

To the question **VOLT:OFFS?** the apparatus returns the value of the programmed offset. It is a value in format **<NR3>** expressed in volt.

### Setting the high logic level

[SOURce:]VOLTage (command/query)  
[:LEVel][:IMMediate]  
:HIGH

The command **VOLT:HIGH <level>** sets the high level of the LOGIC signal to be generated.

**<level>** is a value in format **<NRf>**, it can be followed or not by a sub-multiple and by the unit (V). By default, the value is expressed in volt.

To the question **VOLT:HIGH?** the apparatus returns the value of the programmed low LOGIC level. It is a value in format **<NR3>** expressed in volt.

---

### Setting the low logic level

[SOURce:]VOLTage (command/query)  
[:LEVel][:IMMediate]  
:LOW

The command **VOLT:LOW <level>** sets the low level of the LOGIC signal to be generated.

**<level>** is a value in format **<NRf>**, it can be followed or not by a sub-multiple and by the unit (V). By default, the value is expressed in volt.

To the question **VOLT:LOW?** the apparatus returns the value of the programmed low LOGIC level. It is a value in format **<NR3>** expressed in volt.

## SWEEP Function

### SWEEP source selection

[SOURce:]SWEep (command/query)  
:SOURce

The command **SWE:SOUR {INTERNAL|EXTERNAL}** selects the source of the internal or external sweep.

To the question **SWE:SOUR?** the apparatus returns the current source used for the sweep.

### SWEEP mode selection

[SOURce:]SWEep (command/query)  
:SPACing

The command **SWE:SPAC {LINEar|LOGarithmic}** selects the linear or logarithmic sweep.

To the question **SWE:SPAC?** the apparatus returns the current sweep mode.

### SWEEP type selection

[SOURce:]SWEep (command/query)  
:TYPE

The command **SWE:TYP {TRIangular|SAW}** selects the type of **SWEEP** : triangular or in sawtooth, in case of INTERNAL source.

To the question **SWE:TYP?** the apparatus returns the current sweep type.

### Setting the start/end frequencies

[SOURce:]FREQuency (See §. Setting the signal frequency, **CONTinuous** Function).  
[:START]

[SOURce:]FREQuency (command/query)  
:STOP

The command **FREQ:STOP {<frequency>|MIN|MAX}** adjusts the end frequency  $Freq_{END}$  used for the **SWEEP**, **FSK** and **MODUL FM**.

**<frequency>** is a value in format **<NRf>**, it can be followed or not by a sub-multiple of the unit (Hz). By default, its value is expressed in Hz.

- **MIN** : sets the min value to frequency.
- **MAX** : sets the max value allowed by the apparatus.

To the question **FREQ:STOP?** the apparatus returns the end frequency  $Freq_{END}$  used for the functions **SWEEP**, **FSK** and **MODUL FM**.

The value is expressed in format **<NR3>** en Hz.

## Setting the SWEEP time

[SOURce:]SWEep (command/query)  
:TIME

The command **SWE:TIME <time>** sets the sweep time used in internal sweep.

**<time>** is a value in format **<NRf>**, it can be followed or not by a sub-multiple of the unit (s). By default, its value is expressed in seconds.

To the question **SWE:TIME?** the apparatus returns the current value of the sweep time.

## MODUL Function (GX 320)

### Setting AM rate

[SOURce:]AM[:DEPT~~h~~] (command/query)

The command **AM {20|80}** selects, in case of internal modulation, the selected amplitude modulation rate, i. e. 20 or 80 % of the non-modulated signal total amplitude.

To the question **AM?** the apparatus returns the selected modulation rate (20 or 80 %).

### Selection of AM source

[SOURce:]AM:SOURce (command/query)

The command **AM:SOUR {INTernal|EXTernal}** selects the internal or external modulation source.

To the question **AM:SOUR?** the apparatus returns the current source used for the AM modulation.

### Selection of FM source

[SOURce:]FM:SOURce (command/query)

The command **FM:SOUR {INTernal|EXTernal}** selects the internal or external modulation source.

To the question **FM:SOUR?** the apparatus returns the current source used for the FM modulation.

### Setting start/end frequencies in FM

[SOURce:]FREQuency  
[:START] (See p. 15).

[SOURce:]FREQuency  
:STOP (See p. 19).



## SHIFT Key Function (GX 320)

---

### Selection of the SHIFT KEYING source

[SOURce:]SHIFT (command/query)  
:SOURce

The command **SHIFT:SOUR {INTernal|EXTernal}** selects the internal or external SHIFT KEYING source.

To the question **SHIFT:SOUR?** the apparatus returns the current source used for the SHIFT KEYING.

---

### Setting start/end phases in PSK

[SOURce:]SHIFT (command/query)  
:PHASe[:START]

The command **SHIFT:PHAS <phase>** adjusts the start phase Phase<sub>START</sub> used for the PSK.

**<phase>** is a value in format **<NR1>** expressed in degree °. Its value field is from -180 to +180.

To the question **SHIFT:PHAS?** the apparatus returns the value of the start phase Phase<sub>START</sub> used for the **PSK**.

[SOURce:]SHIFT (command/query)  
:PHASe:STOP

The command **SHIFT:PHAS:STOP <phase>** adjusts the end phase Phase<sub>END</sub> used for the PSK.

**<phase>** is a value in format **<NR1>** expressed in degree °. Its value field is from -180 to +180.

To the question **SHIFT:PHAS:STOP?** the apparatus returns the value of the end phase Phase<sub>END</sub> used for the **PSK**.

---

### Setting start/end frequencies in FSK

[SOURce:]FREQuency (See p. 15).  
[:START]

[SOURce:]FREQuency (See p. 19).  
:STOP

## BURST Function (GX 320)

### BURST source selection

[SOURce:]PULSe (command/query)  
:SOURce

The command **PULS:SOUR {INTernal|EXTernal}** selects the internal or external source.

To the question **PULS:SOUR?** the apparatus returns the current source used for the **BURST**.

### Setting the pulse number to be generated

[SOURce:]PULSe (command/query)  
:COUNT

The command **PULS:COUN <number>** defines the pulse number (signal period) to generate in the next pulse train.

**<number>** is a number in format **<NR1>**, the value field of which is [1 ;65535].

To the question **PULS:COUN?** the apparatus returns the pulse number generated in BURST mode.

### Setting the delay between pulse trains

[SOURce:]PULSe (command/query)  
:DElay

The command **PULS:DEL <delay>** defines the delay of pulse train generation used in INTernal source **BURST**.

**<delay>** is a value in format **<NRf>**, it can be followed or not by a sub-multiple of the unit (s). By default, its value is expressed in seconds.

To the question **PULS:DEL?** the apparatus returns the generation delay of the pulse train in mode **BURST**.

### Manual launching of a pulse train

[SOURce:]PULSe (command)  
:START

The command **PULS:START** launches the generation of a pulse train. This function is available only in EXTernal source **BURST**.

## SYNChronization Function (GX 320)

---

### Setting the dephasing

[SOURce:]PHASe (command/query)  
[:ADJust]

When synchronizing several instruments, the **PHAS** <phase> adjusts :

- on the master : the start-up phase of its output MAIN OUT
- on the slaves : the dephasing of their outputs (MAIN OUT and TTL OUT) over the master.

<phase> is a value in format <NR1> expressed in degrees °.

Its value area is -180 ° to +180 °.

To the question **PHAS?** the device returns the value of the dephasing applied to the output signal.



*On the slave, **PHAS**, **FREQ** and **FUNC** commands are taken into account only if the generation of signals is inactivated by the master : disabling of master HAND OUT exit.  
(See OUTPUT [: STATe]).*

## FREQ Frequencymeter

---

### Reading the frequency value

MEASure[:FREQuency] (query)

To the question **MEAS?** the apparatus returns the frequency which is read. The value is expressed in format **<NR3>** expressed in Hz.



*The **FREQ** function must be selected before running this command (see DEV:MOD command).*

## GATE mode management (GX 320)

---

### Activating GATE mode

OUTP:GATE (command/query)

The command **OUTP:GATE {0|1|ON|OFF}** activates or not the **GATE** mode (piloting the AC output by GATE IN) when this function is allowed.

- **ON | 1:** **GATE** mode is activated
- **OFF | 0:** **GATE** mode is not enabled:

To the question **OUTP:GATE?** the device returns the state of activation of the GATE mode.



***GATE** mode is allowed only in **CONT**, **SWEEP** and **MODUL** functions.*

## Help and Error

---

### Help

HELP[?] (Query)

To the question **HELP? [< directory entry >]** the instrument answers helping in the SCPI commands available.

**< directory entry >** is a key word (short or long form) of first level in the tree of the command. No distinction is made between small and capital letters.

In absence of parameter, the list of the key words accepted by the function is given. When a key word is introduced, the list and the syntax of all the commands starting with this word is returned by the function.

---

### Error

SYSTem:ERRor[:NEXT]? (Query)

To the question **SYST:ERR?**, the instrument returns the number of error positioned at the top of the queue. The queue has a stack of 20 numbers and is managed as follows : first in, first out.

As the **SYST:ERR?** question arrives, the instrument returns the number of errors in order of arrival, until the queue is empty. Every more **SYST:ERR?** questions involve a negative answer: character "0" (ASCII 48 code). If the queue is full, the case at the top of the queue takes the value -350 (saturated queue).

The queue is empty:

- completely : when the instrument is getting started, at the receipt of a \*CLS command.
- partially : at the reading of the last error.

Response format: <error><NL>

with error = negative or 0, no error.

\* **Command error:**  
(-199 to -100)

They indicate that a syntax error has been detected by the syntax analyzer and causes event register bit 5, called CME, CoMmand Error to be set to 1.

|        |                             |
|--------|-----------------------------|
| -101 : | Invalid character           |
| -103 : | Invalid separator           |
| -104 : | Data type error             |
| -108 : | Parameter not allowed       |
| -109 : | Missing parameter           |
| -111 : | Header separator error      |
| -112 : | Program mnemonic too long   |
| -113 : | Undefined header            |
| -114 : | Header suffix out of range  |
| -121 : | Invalid character in number |
| -128 : | Numeric data not allowed    |
| -131 : | Invalid suffix              |
| -138 : | Suffix not allowed          |
| -141 : | Invalid character data      |
| -148 : | Character data not allowed  |
| -151 : | Invalid string data         |
| -154 : | String data too long        |
| -171 : | Invalid expression          |

**\* Execution errors:  
(-299 to -200)**

They indicate that an error has been detected at the moment of command execution and causes event register bit 4, called EXE, Execution Error, to be set to 1.

|      |   |                     |
|------|---|---------------------|
| -200 | : | Execution error     |
| -213 | : | Init ignored        |
| -221 | : | Settings conflict   |
| -222 | : | Data out of range   |
| -232 | : | Invalid format      |
| -256 | : | File name not found |
| -257 | : | File name error     |

**\* Specific instrument  
errors:  
(-399 to -300)**

They indicate that an abnormal error has been detected during execution of a task, and causes event register bit 3, called DDE, Device Dependent Error to be set to 1.

|      |   |                       |
|------|---|-----------------------|
| -300 | : | Device-specific error |
| -321 | : | Out of memory         |
| -350 | : | Queue overflow        |
| -360 | : | Communication error   |

**\* Query errors:  
(-499 to -400)**

They indicate that an abnormal error has been detected during execution of a task, and cause event register bit 2, called QYE, QuerY Error, to be set to 1.

|      |   |             |
|------|---|-------------|
| -400 | : | Query error |
|------|---|-------------|

## IEEE 488.2 common commands

### Introduction

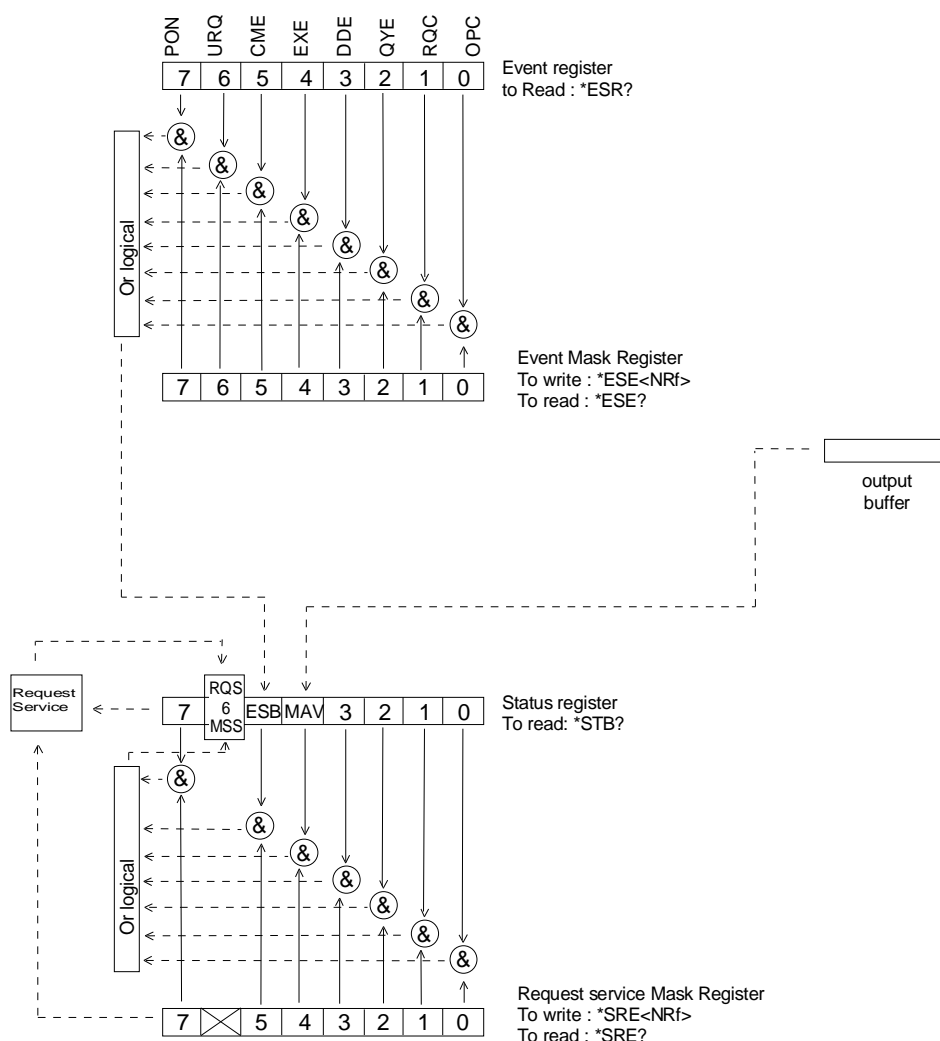
The common commands are defined by the IEEE 488.2 standard. They are operational on all instruments which are specified IEEE 488.2. They command basic functions such as:

- identification,
- reset,
- configuration reading,
- reading of event and status register,
- reset of event and status register.

If a command containing one or several directories has been received, and if a common command has been stacked up, then the instrument stays in this directory and execute normally the commands.

### Events and status management

#### The Registers



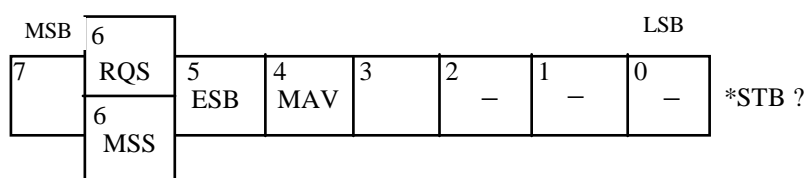


**Status registers**

Reading only → \*STB? common command.

In this case, the (MSS) 6 Bit is returned and remain in the status it was before reading [see §. \*STB (Status Byte)]

The \*CLS common command is reset to zero.

Detailed description**RQS Request Service (6 bit)**

Indicates if the instrument requests a service. The type of COMM used on the instrument does not generate a request, but the byte is accessible in reading. It is reset to 0 after reading and can switch to zero only if the event register is reset to zero (by reading or \*CLS).

**MSS Master Summary Status (6 bit)**

Indicates if the instrument has a reason to request a service. This information is accessible only in reading the status register. (\*STB? command) and stays as it is after the reading.

**ESB Event Status Bit (5 bit)**

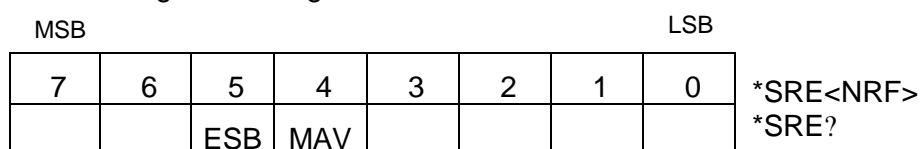
Indicates if at least one of the conditions of the event register is satisfied and not masked.

**MAV Message Available (4 bit)**

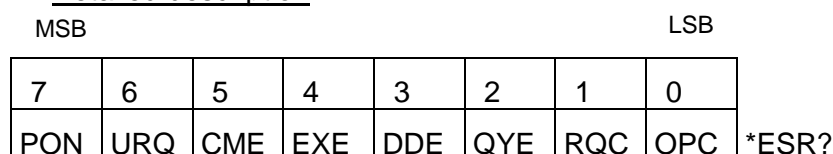
Indicates if at least one response is in the output spooler.

**Service request mask register**

Reading and writing → \*SRE command.

**Event register**

Reading → \*ESR command. Its reading resets to zero.

Detailed description

|            |   |
|------------|---|
| <b>PON</b> | <b>Power On (7 bit)</b><br>Not used   |
| <b>URQ</b> | <b>User request (6 bit)</b><br>Not used   |
| <b>CME</b> | <b>Command Error (5 bit)</b><br>A command error has been detected.                              |
| <b>EXE</b> | <b>Execution Error (4 bit)</b><br>An error execution has been detected.                         |
| <b>DDE</b> | <b>Device Dependant Error 3 (bit)</b><br>An error specific to the instrument has been detected. |
| <b>QYE</b> | <b>Query Error (2 bit)</b><br>A query error has been detected.                                  |
| <b>RQC</b> | <b>Request Control (1 bit)</b><br>Always at zero.   |
| <b>OPC</b> | <b>Operation Complete (0 bit)</b><br>All operations running are ended.                          |

**Event mask register**

Reading and writing → \*ESE command.

| MSB |     |     |     | LSB |     |     |     |           |
|-----|-----|-----|-----|-----|-----|-----|-----|-----------|
| 7   | 6   | 5   | 4   | 3   | 2   | 1   | 0   | *ESE<NRF> |
| PON | URQ | CME | EXE | DDE | QYE | RQC | OPC | *ESE?     |

## The Commands

**\*CLS** (Command)  
(Clear Status)

The common command **\*CLS** reset the status and event register.

**\*ESE** (Command/Query)  
(Event Status Enable)

The **\*ESE <mask>** common command positions the status of the event mask.

**<mask>** is a value in format **<NR1>**, from 0 to 255.

A **1** authorises the corresponding bit of the event register to generate an event, while a **0** masks it.

To the question **\*ESE?**, the instrument returns the current content of the event mask register.

Response format: <value><NL>

value in format **<NR1>** from 0 to 255.

### Event mask register

| MSB |     |     |     | LSB |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|
| 7   | 6   | 5   | 4   | 3   | 2   | 1   | 0   |
| PON | URQ | CME | EXE | DDE | QYE | RQC | OPC |

**\*ESR?** (Query)  
(Event Status Register)

To the question **\*ESR?**, the instrument returns the content of the event register.

Once the register has been read, the content value is reset to zero.

Response format: <value><NL>

value in format **<NR1>** from 0 to 255.

### Event register

| MSB | 6   | 5   | 4   | 3   | 2   | 1   | LSB |
|-----|-----|-----|-----|-----|-----|-----|-----|
| PON | URQ | CME | EXE | DDE | QYE | RQC | OPC |

**\*IDN?** (Query)  
(Identification Number)

To the question **\*IDN?**, the instrument returns the type of instrument and the software version.

Response format:

METRIX <instrument><programmable>,<firmware version>,<version date>,<serial number><NL>

<instrument> Instrument model (**GX 310 / GX 320**)

<programmable> 'P' or 'E' if the instrument is programmable

<firmware version> Software version

<version date> Date of Software version

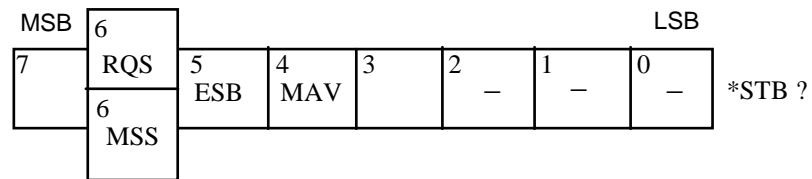
<serial number> Serial number of the instrument

- 
- \*OPC** (Operation Complete) *(Command/Query)*  
 The command **\*OPC** authorises the setting to 1 of the OPC bit in the event register as soon as the current operation is completed. To the question **\*OPC?**, the instrument returns the character ASCII "1" as soon as the current operation is terminated.
- \*RST** (Reset) *(Command)*  
 The command **\*RST** reconfigures the instrument with the factory settings.
- \*SRE** (Service Request Enable) *(Command/Query)*  
 The command **\*SRE <mask>** positions the service request mask register.  
**<mask>** is a value in format **<NR1>**, from 0 to 255.  
 A value of bit at 1 enables the same-rank bit of the status register to request a service (bit of the status register contains 1). A bit value at 0 neutralizes it.  
 To the question **\*SRE?**, the instrument returns the value of the service demand mask register.  
Response format:      <value><NL>  
 value in format **<NR1>** from 0 to 255.
- 

**Service demand mask register**

| MSB |   |     |     | LSB |   |   |   |
|-----|---|-----|-----|-----|---|---|---|
| 7   | 6 | 5   | 4   | 3   | 2 | 1 | 0 |
| 0   | 0 | ESB | MAV | 0   | 0 | 0 | 0 |

- \*STB?** (Status Byte) *(Query)*  
 To the question **\*STB?** the instrument returns the content of its status register (Status Byte Register).  
 The bit 6 returned indicates the MSS value (Master Summary Status) (at 1 if the instrument has a reason for requesting a service). Contrary to RQS, it is not reset to zero after reading the status register (RQS is accessible only by series recognition, and falls to 0 at its end).

**Status register****\*TRG** (Command)

The command **\*TRG** starts an acquisition in the current mode "single" or "continuous".

**\*TST?** (Query)

(Test) To the question **\*TST?**, the instrument returns the status of the autotest procedure.

Response format: <0|1><NL>

- responds 0 when the auto set is successful.
- responds 1 when a problem has been detected.

**\*WAI** (Command)

(Wait) The command **\*WAI** prevents the instrument from performing further commands as long as the current command has not been terminated. This synchronizes the instrument with the application program in progress on the controller.

## Tree structure

### Common commands

| Commands     | Functions                              |
|--------------|--|
| <b>*CLS</b>  | resets the event and status registries |
| <b>*ESE</b>  | writes event mask                      |
| <b>*ESE?</b> | reads event mask                       |
| <b>*ESR?</b> | reads event registry                   |
| <b>*IDN?</b> | reads identifier                       |
| <b>*OPC</b>  | validates OPC bit                      |
| <b>*OPC?</b> | waits till end of execution            |
| <b>*RST</b>  | resets                                 |
| <b>*SRE</b>  | writes service request mask            |
| <b>*SRE?</b> | reads service request mask             |
| <b>*STB?</b> | reads status registry                  |
| <b>*TRG</b>  | runs acquisition in the current mode   |
| <b>*TST?</b> | switches '0'                           |
| <b>*WAI</b>  | synchronises command                   |

### Specific commands

See the table following pages.

## SCPI commands

| Directory | Commands + parameters  | Page |
|-----------|--|------|
| DEvice :  | MODE<br><CONTInuous FSK PSK SWEep AM FM FREQuencymeter SYNCMaster SYNC<br>Slave BURST> | 17   |
|           | MODE?  | 17   |
| DISPlay:  | CONTRast <0 to 1>  | 18   |
|           | CONTRast?  | 18   |
| HELP      | [?]  | 30   |
|           | [?] <directory-entry>  | 30   |
| MEASure   | [:FREQuency]?  | 28   |
| MMEMory:  | CATalog?   | 18   |
|           | DELeTe <1 to15>  | 18   |
|           | LOAD:STATe <0 to15>  | 18   |
|           | STORe:STATe <1 to15>   | 18   |
| OUTPut:   | GATE <1 0 ON OFF>  | 29   |
|           | GATE?  | 29   |
|           | STATe <1 0 ON OFF>   | 17   |
|           | STATe?   | 17   |
| [SOURce:] | AM[:DEPTh] <20 80>   | 24   |
|           | AM[:DEPTh]?  | 24   |
|           | AM:SOURce <INTernal EXTernal>  | 24   |
|           | AM:SOURce?   | 24   |
|           | FM:SOURce <INTernal EXTernal>  | 24   |
|           | FM:SOURce?   | 24   |
|           | FREQuency[:START] <frequency>  | 19   |
|           | FREQuency[:START]?   | 19   |
|           | FREQuency:STOP <frequency>   | 22   |
|           | FREQuency:STOP?  | 22   |
|           | FUNCTION:SHAPE <SINusoid SQUare LOGICal TRIangle DC>                                   | 19   |
|           | FUNCTION:SHAPE?  | 19   |
|           | PHASe[:ADJust] <phase>   | 27   |
|           | PHASe[:ADJust]?  | 27   |
|           | PULSe:COUNt <1 to 65535>   | 26   |
|           | PULSe:COUNt?   | 26   |
|           | PULSe:DCYCLE <duty>  | 19   |
|           | PULSe:DCYCLE?  | 19   |
|           | PULSe:DELay <time>   | 26   |
|           | PULSe:DELay?   | 26   |
|           | PULSe:SOURce <INTernal EXTernal>   | 26   |
|           | PULSe:SOURce?  | 26   |
|           | PULSe:START  | 26   |
|           | SHIFT:SOURce <INTernal EXTernal>   | 25   |
|           | SHIFT:SOURce?  | 25   |
|           | SHIFT:PHASe[:START] <phase>  | 25   |
|           | SHIFT:PHASe[:START]?   | 25   |
|           | SHIFT:PHASe:STOP <phase>   | 25   |
|           | SHIFT:PHASe:STOP?  | 25   |
|           | SWEep:SOURce <INTernal EXTernal>   | 22   |
|           | SWEep:SOURce?  | 22   |
|           | SWEep:SPACing <LINear LOGarithmic>   | 22   |
|           | SWEep:SPACing?   | 22   |
|           | SWEep:TIME <time>  | 23   |
|           | SWEep:TIME?  | 23   |
|           | SWEep:TYPe <TRIangular SAW>  | 22   |
|           | SWEep:TYPe?  | 22   |

| Directory | Commands + parameters                        | Page |
|-----------|--|------|
|           | VOLTage[:LEVel][:IMMediate]:AMPL <amplitude> | 20   |
|           | VOLTage[:LEVel][:IMMediate]:AMPL?            | 20   |
|           | VOLTage[:LEVel][:IMMediate]:HIGH <level>     | 21   |
|           | VOLTage[:LEVel][:IMMediate]:HIGH?            | 21   |
|           | VOLTage[:LEVel][:IMMediate]:LOW <level>      | 21   |
|           | VOLTage[:LEVel][:IMMediate]:LOW?             | 21   |
|           | VOLTage[:LEVel][:IMMediate]:OFFSet <offset>  | 20   |
|           | VOLTage[:LEVel][:IMMediate]:OFFSet?          | 20   |
| SYSTem:   | ERRor[:NEXT]?                                | 20   |
|           | POWer <1 0 ON OFF>                           | 17   |
|           | POWer?                                       | 17   |
| UNIT:     | VOLT:AMPL <PTPeak RMSquare>                  | 20   |
|           | VOLT:AMPL?                                   | 20   |