

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*

PROLOGIX

GPIB-ETHERNET CONTROLLER



USER MANUAL

VERSION 1.6.6.0

May 14, 2013

PROLOGIX.BIZ

Table of Contents

1. Introduction 4
2. Installation 4
3. Firmware Upgrade 4
4. Host Software 4
5. Network Configuration 5
6. GPIB Configuration 5
7. Operating Modes 5
   7.1. Controller Mode 5
   7.2. Device Mode 6
8. Data Transmission 6
   8.1. Binary Data Transmission 7
9. Commands 8
   9.1. addr 8
   9.2. auto 8
   9.3. clr 9
   9.4. eoi 9
   9.5. eos 10
   9.6. eot_enable 10
   9.7. eot_char 11
   9.8. ifc 11
   9.9. llo 11
   9.10. loc 11
   9.11. lon 11
   9.12. mode 12
   9.13. read 12
   9.14. read_tmo_ms 13
   9.15. rst 13
   9.16. savecfg 13
   9.17. spoll 14
   9.18. srq 14
   9.19. status 14
   9.20. trg 15
   9.21. ver 15
   9.22. help 15
10. Specifications 15

Change Log

May 14, 2013	Added ++llo command description.
May 7, 2012	Corrected error in ++status documentation.
April 18, 2011	Added ++lon command description.
Sept 14, 2009	Added ++savecfg command descripion. Corrected maximum value of read_tmo_ms to 3000 ms.
May 22, 2009	Added ++status command description. Updated ++addr, ++spoll and ++trg command descriptions to include secondary address. Added section on binary data transmission.
Aug 5, 2008	Initial version.

1. Introduction

Prologix GPIB-ETHERNET controller converts any computer with a network port into a GPIB Controller or Device.

In Controller mode, Prologix GPIB-ETHERNET controller can remotely control GPIB enabled instruments such as Oscilloscopes, Logic Analyzers, and Spectrum Analyzers.

In Device mode, Prologix GPIB-ETHERNET controller converts the computer into a GPIB peripheral for downloading data and screen plots from the instrument front panel.

In both modes, Prologix GPIB-ETHERNET controller interprets high level commands received from the host computer and performs the appropriate low-level GPIB protocol handshaking.

2. Installation

Connect Prologix GPIB-ETHERNET controller to any network enabled computer using an Ethernet cable. No special drivers are required. The type of cable to use depends on the computer. If the computer supports auto-MDIX, which almost all newer ones do, use a straight Ethernet cable. If the computer does not support auto-MDIX, use a cross-over Ethernet cable. If a cross-over Ethernet cable is not available, or does not work, connect both Prologix GPIB-ETHERNET controller and computer to a network hub (or switch).

3. Firmware Upgrade

Prologix GPIB-ETHERNET controller firmware is field upgradeable. Latest firmware and upgrade installations are available at prologix.biz

4. Host Software

A wide variety of host software may be used to communicate with Prologix GPIB-ETHERNET controller:

Terminal programs – any terminal emulation program such as HyperTerminal, Tera Term Pro, or Minicom can be used to communicate with the controller and instruments connected to it.

Custom applications – any programming language or environment that provides network access may be used to develop custom applications. Graphical programming environments like National Instruments LabView and Agilent VEE may be used as well.

EZGPIB – an easy to use, programming environment developed by Ulrich Bangert for developing data acquisition applications. Web link to this tool can be found at prologix.biz.

Plotter emulators – plotter emulation applications such as 7470.exe can be used to obtain screen plots from GPIB enabled instruments using Prologix GPIB-ETHERNET controller. Details on how to configure these tools are available at prologix.biz.

5. Network Configuration

Prologix GPIB-ETHERNET controller supports static IP address and dynamic (DHCP) IP address. The controller's network parameters can be configured using the **NETFINDER** tool available at prologix.biz.

6. GPIB Configuration

GPIB parameters of Prologix GPIB-ETHERNET controller can be configured using any of the following methods:

Prologix.exe – Prologix.exe is an open source tool developed by John Miles for configuring the controller. Web link to the tool can be found at prologix.biz.

Terminal program – any terminal emulation program such as HyperTerminal, Tera Term Pro, or Minicom can be used to configure the controller by manually entering appropriate commands (See Commands). Using the terminal program open a TCP connection to port 1234 at Prologix GPIB-ETHERNET controller's IP address. You may want to enable the "Local Echo" feature in the terminal program to view the commands being entered. Please consult the FAQ at prologix.biz, or the program's user manual, for detailed configuration steps for various terminal programs.

Prologix GPIB-ETHERNET Controller stores the latest configuration settings in non-volatile memory. These settings are not address specific. If you have multiple instruments on the GPIB bus that require different configuration settings, you must change the settings before communicating with each instrument.

7. Operating Modes

Prologix GPIB-ETHERNET controller can operate in two modes – CONTROLLER and DEVICE. You can switch between the two modes using ++mode command (see Commands).

7.1. Controller Mode

In Controller mode, the GPIB-ETHERNET Controller acts as the Controller-In-Charge (CIC) on the GPIB bus. When the controller receives a command over the network port terminated by the network terminator – CR (ASCII 13) or LF (ASCII 10) – it addresses

the GPIB instrument at the currently specified address (See ++addr command) to listen, and passes along the received data.

When Read-After-Write feature is enabled (See ++auto command) Prologix GPIB-ETHERNET Controller will addresses the instrument to talk after sending a command, in order to read its response. All data received from instruments over GPIB is sent to host over the network. Read-After-Write feature simplifies communication with instruments. You send commands and read responses without consideration for low level GPIB protocol details.

When Read-After-Write feature is not enabled Prologix GPIB-ETHERNET controller does not automatically address the instrument to talk. You must use the ++read command to read data.

Controller mode is used to remotely control instruments and to download screen plots by sending plot commands from a host computer.

7.2. Device Mode

In Device mode, Prologix GPIB-ETHERNET Controller acts as another peripheral on the GPIB bus. In this mode, the controller can act as a GPIB TALKER or GPIB LISTENER only. Since Prologix GPIB-ETHERNET Controller is not the Controller-In-Charge while in this mode, it expects to receive commands from a GPIB controller. When Device mode is enabled Prologix GPIB-ETHERNET controller configures itself as a GPIB Listener. All data received by the controller over the GPIB port is passed along to the network port without buffering.

All data received from the host over the network is buffered until the GPIB controller addresses Prologix GPIB-ETHERNET controller to talk, at which time the buffered data is passed along to the GPIB port. The controller can buffer only one command. A subsequent command received over the network will overwrite the previously buffered one, if the previous one has not yet been transmitted over GPIB.

Device mode is used to download screen plots from the instrument front panel for rendering using plotter emulation software.

8. Data Transmission

In Controller and Device modes, characters received over ETHERNET port are aggregated in an internal buffer and interpreted when an ETHERNET termination character – CR (ASCII 13) or LF (ASCII 10) – is received. If CR, LF, ESC (ASCII 27), or '+' (ASCII 43) characters are part of ETHERNET data they must be escaped by preceding them with an ESC character. All un-escaped LF, CR and ESC and '+' characters in ETHERNET data are discarded.

As mentioned earlier, an un-escaped CR or LF acts as the ETHERNET terminator. The terminating CR or LF is removed and GPIB termination characters (specified by ++eos command) are appended before transmitting data to instruments.

Any ETHERNET input that starts with the unescaped "++" character sequence is interpreted as a controller command and not transmitted over GPIB.

When configured to do so – using the ++auto command or the ++read command – characters received from instruments are transmitted to host. Unlike while sending data to instruments, no character substitution is performed.

The ++eot_char command may be used to detect GPIB EOI signal assertion.

8.1. Binary Data Transmission

Prologix GPIB-ETHERNET controller can send and receive binary data to and from GPIB enabled instruments.

No special action is necessary to receive binary data from instruments. Any binary data received from the instrument is transmitted over ETHERNET to PC unmodified, just as with ASCII data. Since binary data from instruments is not usually terminated by CR or LF characters (as is usually the case with ASCII data), you may want to use the ++eot_enable command to detect EOI indicating end of data. See ++eot_enable command help for more details.

Special care must be taken when sending binary data to instruments. If any of the following characters occur in the binary data -- CR (ASCII 13), LF (ASCII 10), ESC (ASCII 27), '+' (ASCII 43) – they must be escaped by preceding them with an ESC character.

For example, to send the following (decimal) binary data:

00 01 02 13 03 10 04 27 05 43 06

it must be escaped as follows:

00 01 02 **27** 13 03 **27** 10 04 **27** 27 05 **27** 43 06

Further more, most instruments will get confused if GPIB termination characters, such as CR or LF, are appended to binary data. Use ++eos 3 command to disable such behavior. See ++eos command help for more details.

9. Commands

Prologix GPIB-ETHERNET controller provides several commands to configure its behavior. They are explained in detail in the following sections. All commands start with the "++" character sequence.

9.1. addr

The addr command is used to configure, or query the GPIB address. Meaning of the GPIB address depends on the operating mode of the controller. In CONTROLLER mode, it refers to the GPIB address of the instrument being controlled. In DEVICE mode, it is the address of the GPIB peripheral that Prologix GPIB-ETHERNET controller is emulating.

An optional secondary address may also be specified. Secondary address must be separated from the primary address by a space character. Valid secondary address values are 96 to 126 (decimal). Secondary address value of 96 corresponds to secondary GPIB address of 0, 97 corresponds to 1, and so on. Specifying secondary address has no effect in DEVICE mode.

If the command is issued with no parameters, the currently configured address (primary, and secondary, if specified) is returned.

SYNTAX: ++addr [<PAD> [<SAD>]]
PAD (Primary Address) is a decimal value between 0 and 30.
SAD (Secondary Address) is a decimal value between 96 and 126. SAD is optional.

MODES AVAILABLE: CONTROLLER, DEVICE

EXAMPLES:

++addr 5	– Set primary address to 5
++addr	– Query current address
++addr 9 96	– Set primary address to 9 and secondary address to 0

NOTE:
Default GPIB address of many HP-GL/2 plotters is 5.

9.2. auto

Prologix GPIB-ETHERNET controller can be configured to automatically address instruments to talk after sending them a command in order to read their response. The feature called, Read-After-Write, saves the user from having to issue read commands repeatedly. This command enabled or disabled the Read-After-Write feature.

In addition, auto command also addresses the instrument at the currently specified address to TALK or LISTEN. ++auto 0 addresses the instrument to LISTEN and ++auto 1 addresses the instrument to TALK.

If the command is issued without any arguments it returns the current state of the read-after-write feature.

SYNTAX: ++auto [0|1]

MODES AVAILABLE: CONTROLLER

NOTE:
Some instruments generate "Query Unterminated" or "-420" error if they are addressed to talk after sending a command that does not generate a response (often called non-query commands). In effect the instrument is saying, I have been asked to talk but I have nothing to say. The error is often benign and may be ignored. Otherwise, use the ++read command to read the instrument response. For example:

++auto 0	—	Turn off read-after-write and address instrument to listen
SET VOLT 1.0	—	Non-query command
*idn?	—	Query command
++read eoi	—	Read until EOI asserted by instrument
"HP54201A"	—	Response from instrument

9.3. clr

This command sends the Selected Device Clear (SDC) message to the currently specified GPIB address. Please consult the programming manual for details on how a particular instrument responds to this message.

SYNTAX: ++clr

MODES AVAILABLE: CONTROLLER

9.4. eoi

This command enables or disables the assertion of the EOI signal with the last character of any command sent over GPIB port. Some instruments require EOI signal to be asserted in order to properly detect the end of a command.

SYNTAX: ++eoi [0|1]

MODES AVAILABLE: CONTROLLER, DEVICE

EXAMPLES:

++eoi 1	Enable EOI assertion with last character
++eoi 0	Disable EOI assertion
++eoi	Query if EOI assertion is enabled or disabled

9.5. eos

This command specifies GPIB termination characters. When data from host is received over the network, all non-escaped LF, CR and ESC characters are removed and GPIB terminators, as specified by this command, are appended before sending the data to instruments. This command does not affect data from instruments received over GPIB port.

If the command is issued with no arguments then the current configuration is returned.

SYNTAX: ++eos [0|1|2|3] where: 0 – CR+LF, 1 – CR, 2 – LF, 3 – None

MODES AVAILABLE: CONTROLLER, DEVICE

EXAMPLES:

++eos 0	Append CR+LF to instrument commands
++eos 1	Append CR to instrument commands
++eos 2	Append LF to instrument commands
++eos 3	Do not append anything to instrument commands
++eos	Query current EOS state

9.6. eot_enable

This command enables or disables the appending of a user specified character (see eot_char) to network output whenever EOI is detected while reading a character from the GPIB port.

If the command is issued without any argument, the current state of eot_enable is returned.

SYNTAX: eot_enable [0|1]

MODES AVAILABLE: CONTROLLER, DEVICE

EXAMPLES:

++eot_enable 1	Append user defined character when EOI detected
++eot_enable 0	Do not append character when EOI detected
++eot_enable	Query current eot_enable state

9.7. eot_char

This command specifies the character to be appended to network output when eot_enable is set to 1 and EOI is detected.

If the command is issued without any argument, the currently specified character is returned.

SYNTAX: eot_char [<char>] where <char> is a decimal value less than 256

MODES AVAILABLE: CONTROLLER, DEVICE

EXAMPLES:

++eot_char 42	Append * (ASCII 42) when EOI is detected
++eot_char	Query currently configured eot_char

9.8. ifc

This command asserts GPIB IFC signal for 150 microseconds making Prologix GPIB-ETHERNET controller the Controller-In-Charge on the GPIB bus.

SYNTAX: ++ifc

MODES AVAILABLE: CONTROLER

9.9. llo

This command disables front panel operation of the currently addressed instrument.

SYNTAX: ++llo

MODES AVAILABLE: CONTROLLER

9.10. loc

This command enables front panel operation of the currently addressed instrument.

SYNTAX: ++loc

MODES AVAILABLE: CONTROLLER

9.11. lon

This command configures the GPIB-ETHERNET controller to listen to all traffic on the GPIB bus, irrespective of the currently specified address. This configuration is also known as “listen-only” mode. In this mode, the controller can only receive, but cannot send any data.

SYNTAX: lon [0|1]

MODES AVAILABLE: DEVICE

EXAMPLES:

++lon 1	Enable "listen-only" mode
++lon 0	Disable "listen-only" mode
++lon	Query "listen-only" mode

9.12. mode

This command configures the Prologix GPIB-ETHERNET controller to be a CONTROLLER or DEVICE.

If the command is issued without any arguments, the current mode is returned.

SYNTAX: ++mode [0|1] where 1 – CONTROLLER, 0 – DEVICE

MODES AVAILABLE: CONTROLLER, DEVICE

EXAMPLES:

++mode 1	Switch to CONTROLLER mode
++mode 0	Switch to DEVICE mode
++mode	Query current mode

9.13. read

This command can be used to read data from an instrument until:

- EOI is detected or timeout expires, or
- A specified character is read or timeout expires, or
- Timeout expires

Timeout is set using the read_tmo_ms command and applies to inter-character delay, i.e., the delay since the last character was read. Timeout is not be confused with the total time for which data is read.

SYNTAX: ++read [eoi|<char>] where <char> is a decimal value less than 256

MODES AVAILABLE: CONTROLLER

EXAMPLES:

++read	Read until timeout
++read eoi	Read until EOI detected or timeout

++read 10 Read until LF (ASCII 10) is received or timeout

9.14. read_tmo_ms

This command specifies the timeout value, in milliseconds, to be used in the read command and spoll command. Timeout may be set to any value between 1 and 3000 milliseconds.

SYNTAX: ++read_tmo_ms <time> where <time> is decimal value between 1 and 3000

MODES AVAILABLE: CONTROLLER

9.15. rst

This command performs a power-on reset of the controller. The process takes about 5 seconds. All input received over the network during this time are ignored.

SYNTAX: ++rst

MODES AVAILABLE: CONTROLLER, DEVICE

9.16. savecfg

This command enables, or disables, automatic saving of configuration parameters in EPROM. If enabled, the following configuration parameters are saved whenever they are updated – mode, addr, auto, eoi, eos, eot_enable, eot_char and read_tmo_ms.

However, frequent updates may eventually wear out the EPROM. This command may be used to temporarily disable automatic saving of configuration parameters to reduce EPROM wear.

The savecfg setting itself is not saved in EPROM. It is always enabled on startup (after power up, or reset).

SYNTAX: ++savecfg [0|1]

MODES AVAILABLE: CONTROLLER, DEVICE

EXAMPLE:

++savecfg 1 Enable saving of configuration parameters in EPROM
++savecfg 0 Disable saving of configuration parameters in EPROM
++savecfg Query current setting

NOTE:

"++savecfg 1" command will immediately save the current values of all configuration parameters, in addition to enabling the automatic saving of parameters.

9.17. spoll

This command performs a serial poll of the instrument at the specified address. If no address is specified then this command serial polls the currently addressed instrument (as set by a previous ++addr command). This command uses the time-out value specified by the read_tmo_ms command.

SYNTAX: ++spoll [<PAD> [<SAD>]]
PAD (Primary Address) is a decimal value between 0 and 30.
SAD (Secondary Address) is a decimal value between 96 and 126. SAD is optional.

MODES AVAILABLE: CONTROLLER

EXAMPLE:

++spoll 5	Serial poll instrument at primary address 5
++spoll 9 96	Serial poll instrument at primary address 9, seconday address 0
++spoll	Serial poll currently addressed instrument

9.18. srq

This command returns the current state of the GPIB SRQ signal. The command returns '1' is SRQ signal is asserted (low) and '0' if the signal is not asserted (high).

SYNTAX: ++srq

MODES AVAILABLE: CONTROLLER

9.19. status

The status command is used to specify the device status byte to be returned when serial polled by a GPIB controller. If the RQS bit (bit #6) of the status byte is set then the SRQ signal is asserted (low). After a serial poll, SRQ line is de-asserted and status byte is set to 0. Status byte is initialized to 0 on power up.

SRQ is also de-asserted and status byte is cleared if DEVICE CLEAR (DCL) message, or SELECTED DEVICE CLEAR (SDC) message, is received from the GPIB controller.

If the command is issued without any arguments it returns the currently specified status byte.

SYNTAX: ++status [0-255]

MODES AVAILABLE: DEVICE

EXAMPLE:

++status 72	Specify serial poll status byte as 72. Since bit #6 is set, this command will assert SRQ.
++status	Query current serial poll status byte.

9.20. trg

This command issues Group Execute Trigger GPIB command to devices at the specified addresses. Up to 15 addresses maybe specified. Addresses must be separated by spaces. If no address is specified then Group Execute Trigger command is issued to the currently addressed instrument (as set by a previous ++addr command). Refer to the programming manual for a specific instrument's response to Group Execute Trigger command.

SYNTAX: ++trg [<PAD1> [<SAD1>] <PAD2> [SAD2] … <PAD15> [<SAD15>]]

MODES AVAILABLE: CONTROLLER

9.21. ver

This command returns the version string of the Prologix GPIB-ETHERNET controller.

SYNTAX: ++ver

MODES AVAILABLE: CONTROLLER, DEVICE

9.22. help

This command prints a brief summary of all available commands.

SYNTAX: ++help

MODES AVAILABLE: CONTROLLER, DEVICE

10.Specifications

Supported OS: Windows 98/2000/XP/Vista/7, Mac OS 8/9/X, Linux, FreeBSD
Supported Standards: IEEE 488.1, IEEE 488.2
GPIB commands not supported: PARALLEL POLL, PASS CONTROL
Power: 8-15V DC, 200 mA
Indicators: Power
IP configuration: Static and Dynamic (DHCP)
TCP port: 1234
Dimensions: 2.5 in. (L) x 2.5 in. (W) x 1.0 in. (H)

Weight: 3 oz.

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP