

Quarch Instrument Server Technical Manual

Version 1.56

Table of Contents

1. Introduction . . . 1
2. Launching QIS . . . 2
   2.1. Command line parameters . . . 2
   2.2. Terminal . . . 2
   2.3. Basic Setup - Remote terminal . . . 4
3. Command Listing . . . 7
   3.1. QIS Setup commands . . . 7
   3.2. QIS Data commands . . . 8
   3.3. Module Commands . . . 11
4. Command Reference . . . 12
   4.1. $about . . . 13
   4.2. $default [deviceListEntryNumber|deviceIdentifier|connectionPath] . . . 13
   4.3. $default? . . . 14
   4.4. $get filters . . . 14
   4.5. $help . . . 14
   4.6. $history . . . 15
   4.7. $list [details] [extended] . . . 15
   4.8. $load filter [directory path] . . . 15
   4.9. $loglevel [level] . . . 16
   4.10. $loglevel? . . . 16
   4.11. $scan [address] . . . 16
   4.12. $scan? . . . 17
   4.13. $shutdown . . . 17
   4.14. $version . . . 18
   4.15. close . . . 18
   4.16. hello? . . . 18
   4.17. rec stop . . . 18
   4.18. rec stream . . . 19
   4.19. stream bin [all|number of stripes] . . . 19
   4.20. stream buffer text [all|number of stripes] . . . 20
   4.21. stream create channel chan(chName,chGroup) function(arg1, arg2, ...) . . . 20
   4.22. stream created channel chan(chName,chGroup)? . . . 20
   4.23. stream created channel delete chan(chName,chGroup) . . . 20
   4.24. stream created channel enable chan(chName, chGroup) [ON|OFF] . . . 20
   4.25. stream created channels [enabled|disabled]? . . . 21
   4.26. stream created channels clear . . . 21
   4.27. stream created channels? . . . 21
   4.28. stream created function definitions? . . . 21

4.29. stream mode header [v1|v2|v3] . . . 21
4.30. stream mode power [enable|disable|?] . . . 22
4.31. stream mode power total [enable|disable|?] . . . 22
4.32. stream mode resample 1..n[ms|us] . . . 23
4.33. stream mode resample group g 1..n[ms|us] . . . 24
4.34. stream mode resample off . . . 24
4.35. stream mode resample? . . . 24
4.36. stream text [all|number of stripes] . . . 25
4.37. stream text header . . . 25
4.38. stream? . . . 26
4.39. tcp::DeviceIdentifier forceIPAddress aaa.bbb.ccc.ddd . . . 26
4.40. tcp::DeviceIdentifier forcereset . . . 26
5. Example Usage . . . 27
5.1. Simple setup and streaming . . . 27

1. Introduction

The Quarch Instrument Server, or QIS for short, is a Java based server app which allows for simple automation of our range of Power Modules. QIS can connect to USB or remote LAN power modules. It handles module location and connection.

QIS exposes a TCP port, allowing local or remote scripts to access the full abilities of the Quarch power modules.

QIS allows you to send commands, get responses and to stream large blocks of power data from a module. Optimization and buffering are handled within QIS, allowing you to keep your automation scripts simple.

© Quarch Technology 2026

2. Launching QIS

- If you have QuarchPy installed, you can run QIS with the command:

```
python -m quarchpy.run qis
```

- Launching QPS will automatically start an instance of QIS.

2.1. Command line parameters

- Add the terminal parameter to launch QIS with a terminal window.

```
-terminal
```

- Add the headless parameter to launch QIS without a terminal window or system tray icon. This needs to come before any other parameters.

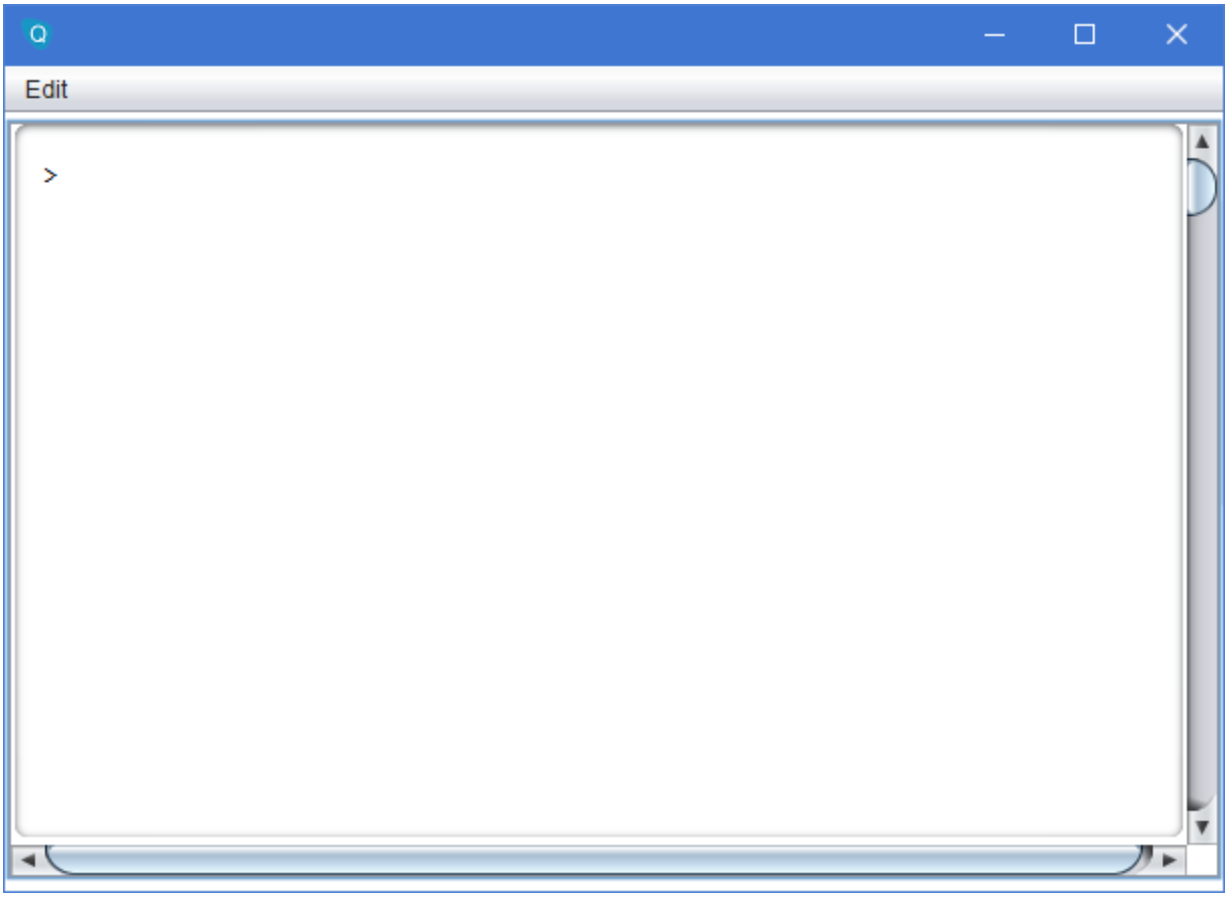
```
-Djava.awt.headless=true
```

2.2. Terminal

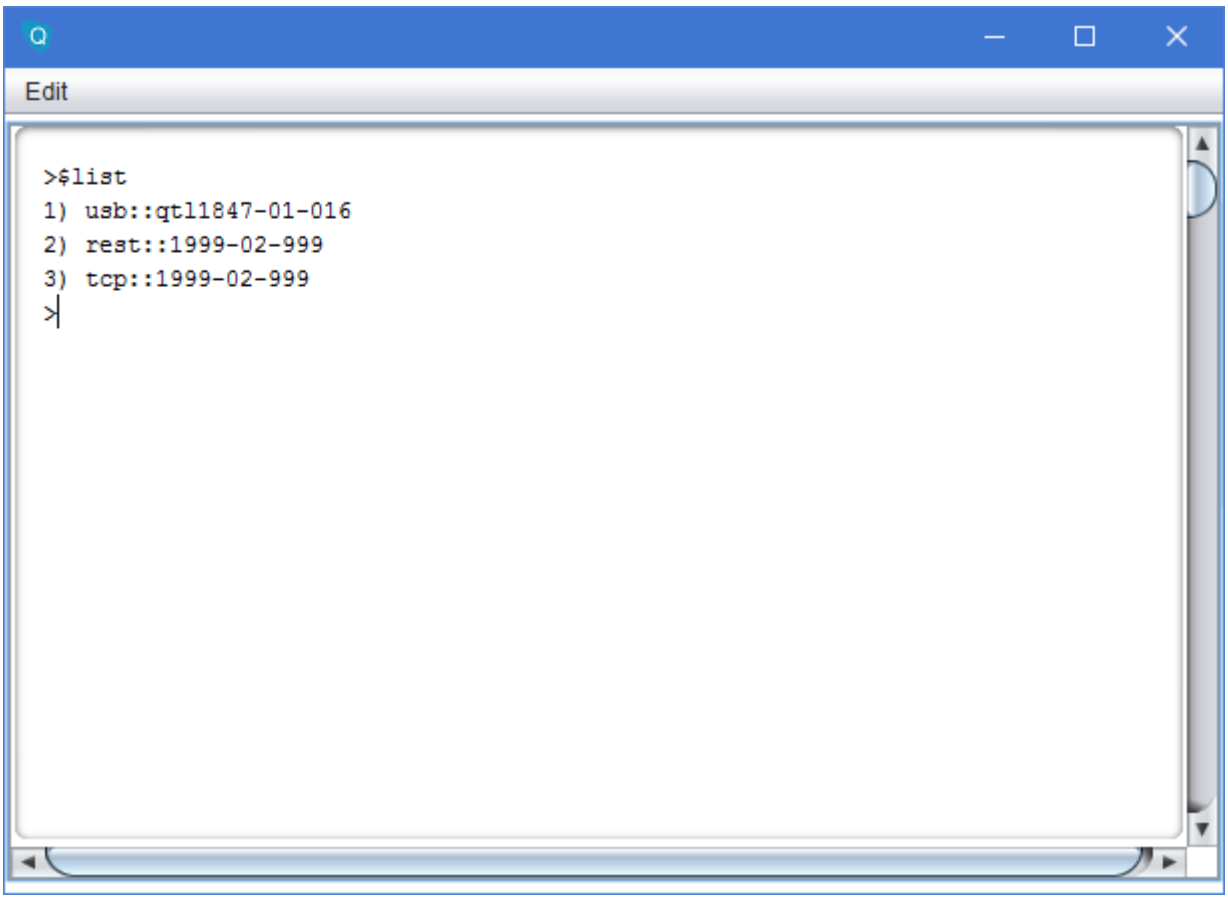
QIS can launch a terminal window for you to enter commands directly. This is useful for testing and debugging.

If QIS is already running, there will be an icon in the task bar. Right click this and select **Show Terminal**. If the icon does not work on your linux distro, run QIS with the `-terminal` option.

© Quarch Technology 2026



This terminal uses the native terminal application for your operating system.



You can use auto-complete by pressing the tab key. For example, if you type `$li` and press tab, it will complete to `$list`. If there are multiple options, press tab again to select between the options, then

© Quarch Technology 2026

enter to select the option.

You can use the arrow keys to scroll through the command history. You can also use the `$history` command to view a list of commands that were entered during this session. The history can be searched by using **CTRL-R** and typing part of the command you are looking for - press **CTRL-R** again to scroll through multiple matches.

You can see a list of all available commands by entering the `$help` command.

QIS will automatically scan for available devices, but you can force it to rescan using the `$scan` command. If the command is followed by an IP address, it will scan for a Quarch device at that IP address.

You can list the available devices using the `$list` command.

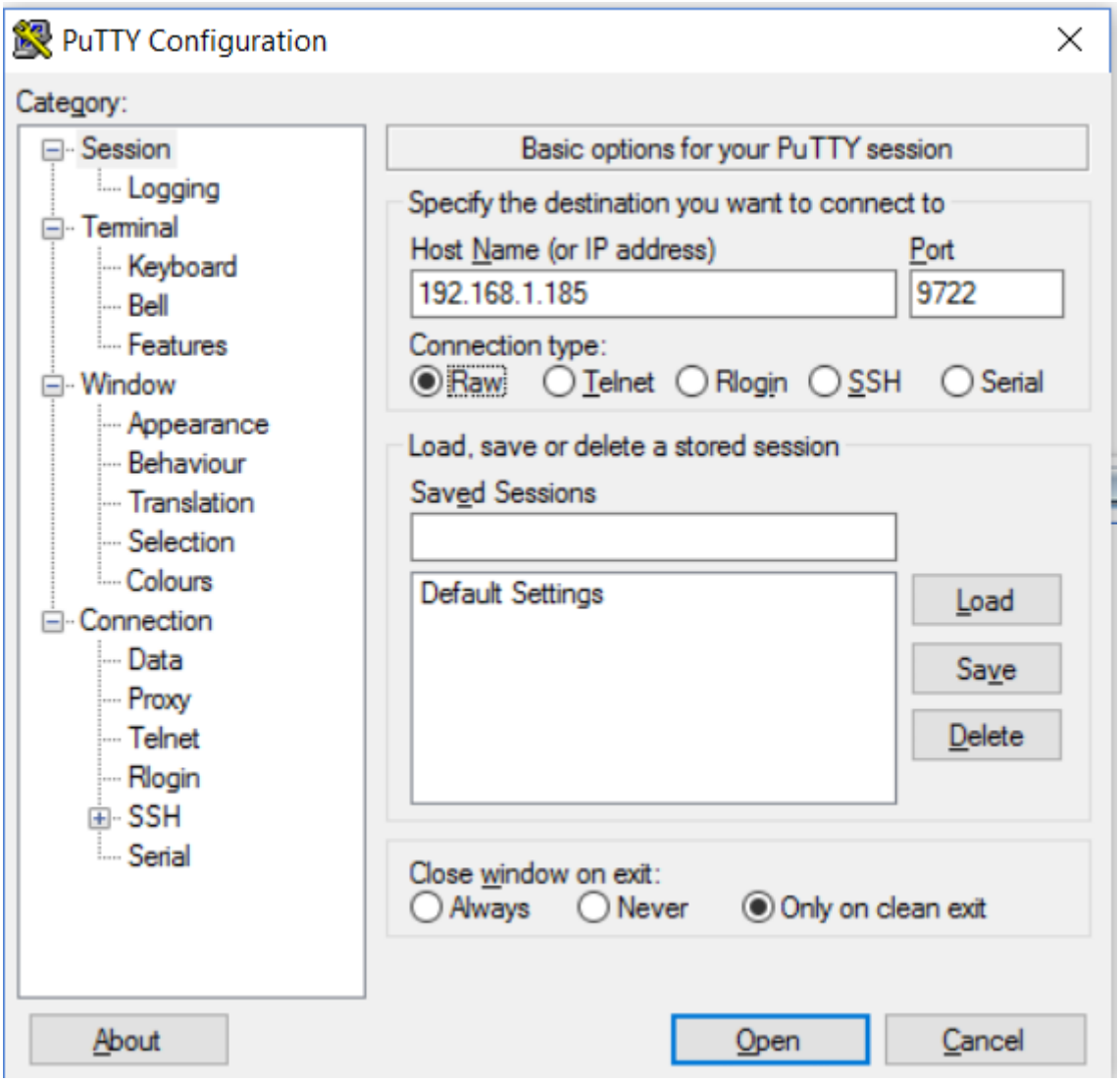
You can set a default device to send commands to using the `$default` command. For example, if you have two devices connected, you can set the default device to the first one by entering `$default 1`. You can then enter commands for that device without having to specify the device each time.

2.3. Basic Setup - Remote terminal

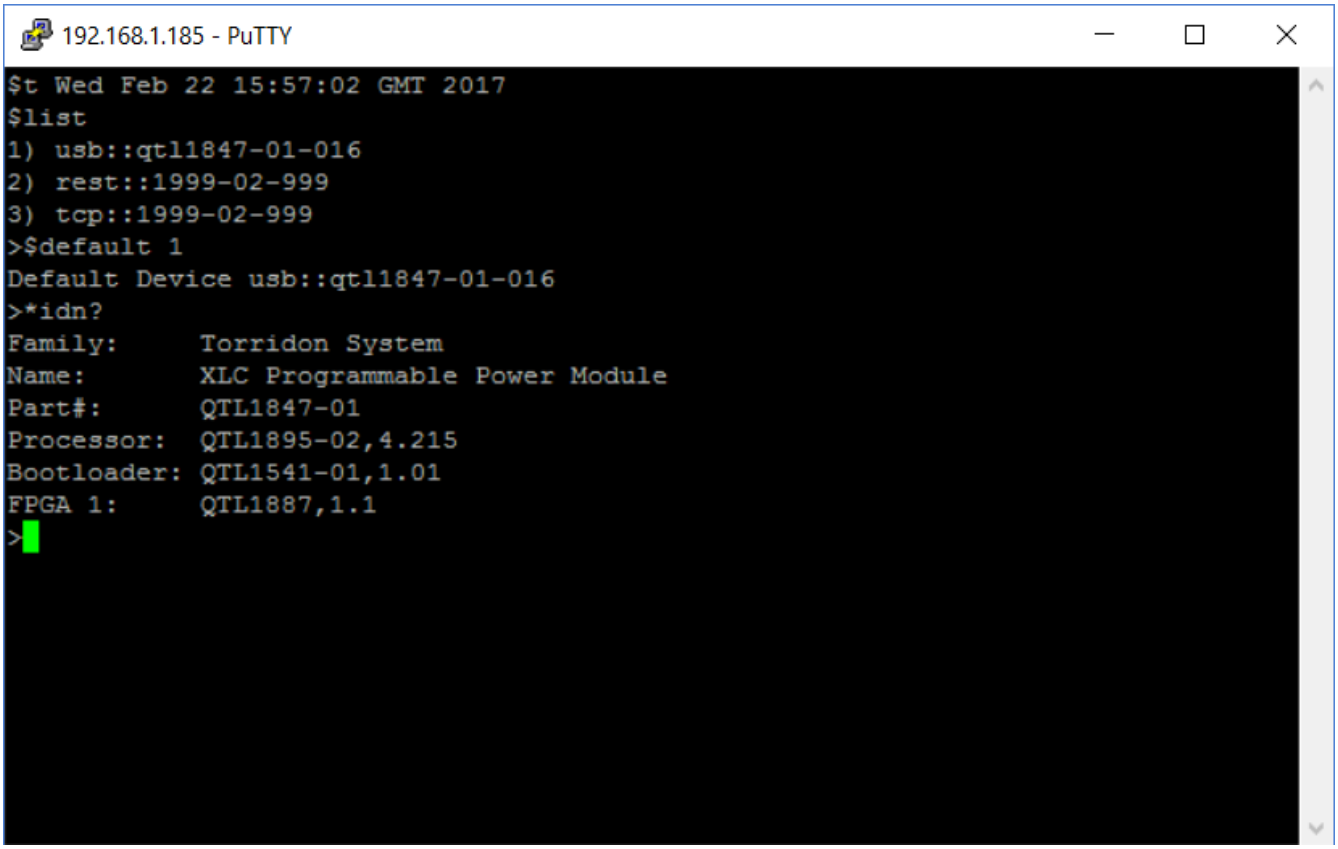
QIS works by exposing a TCP port on the PC it runs on. You can connect to this via a standard terminal program such as PuTTY. Below you can see the settings used.

- The IP address is the IP of the PC running QIS
- The port is 9722
- The connection type must be **Raw**

© Quarch Technology 2026



History, search and auto-complete functions are not available in this mode, but you can still enter commands and see the output. You can also use the `$help` command to see a list of available commands.



© Quarch Technology 2026

3. Command Listing

3.1. QIS Setup commands

These commands are interpreted and responded to by QIS and perform basic setup.

```
$help
```

Displays a basic help screen.

```
$scan
```

Begins a scan of devices, both local USB modules and those visible over the network will be located. QIS can use a broadcast UDP packet or mDNS to locate LAN devices. You can also specify an IP address to scan for a specific device.

```
$list
```

Lists all local and LAN devices. `$scan` command must have been run previously.

```
$list details
```

As `$list` but with additional device information displayed.

```
$default [DeviceConnection]
```

Sets the connection to send all following commands to the given default device:

```
$default tcp::qtl1944-01-020
```

```
$default [DeviceNumber]
```

Sets the connection to send all following commands to the given default device. Uses the list number from the `$list` command.

```
$default 2
```

```
$sysinfo
```


© Quarch Technology 2026

Displays memory usage and stream task information.

```
$shutdown
```

Closes QIS immediately. You can also use the system tray icon to close QIS.

3.2. QIS Data commands

Although these are executed internally by QIS, they differ from the `$` commands in that they require a module identifier in the command, or a currently selected default module (via the `$default` command).



For example, the following command will execute on the default module:
`stream text header`



If a default is not set, you must specify the DeviceConnection string before the command, for example:
`usb::qtl1824-03-161 stream text header`

```
stream text header
```

Returns the stream header information from the most recent stream to be run on the device. There are several versions of the header available which have been extended over time (set with the `stream mode header` command).

```
V1                              // Initial "basic" header
Version: 5      // Underlying stream version, safe to ignore
Format: 15      // Binary coding for the 4 channel enable states
Average: 14     // 2^n value for averaging (32k in this case)
```

```
V2                              // Extended information header
Version: 5      // V1 header information
Format: 15
Average: 10
V2              // Start of v2 header
@Channels       // Start of channel data
Status status NA        // Status channel (triggering)
5V voltage mV   // [Channel] [Measurement] [Unit]
5V current uA
12V voltage mV
12V current uA
5V power uW
12V power uW
@Channels_End   // End of channel data
```


© Quarch Technology 2026

```
V3                              // XML based information header
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<header>
    <version>V3</version>
    <mainPeriod>4096uS</mainPeriod>
    <legacyVersion>5</legacyVersion>
    <legacyFormat>15</legacyFormat>
    <legacyAverage>10</legacyAverage>
    <channels>
        <channel>
            <name>Status</name>
            <group>status</group>
            <units>NA</units>
            <dataPosition>0</dataPosition>
        </channel>
        <channel>
            <name>5V</name>
            <group>voltage</group>
            <units>mV</units>
            <dataPosition>1</dataPosition>
        </channel>
        <channel>
            <name>5V</name>
            <group>current</group>
            <units>uA</units>
            <dataPosition>2</dataPosition>
        </channel>
        <channel>
            ... additional channels ...
        </channel>
    </channels>
</header>
```

```
stream mode header [v1|v2|v3]
```

Sets the header mode to use. This must be set before the start of a stream.

```
stream mode power [enable|disable]
```

Enable/disable the internal power calculation. Must be set before the start of the stream (defaults to disabled). When enabled AND both current and voltage is enabled for a channel, the power values will be calculated and streamed back along with current/voltage.

```
stream?
```

Returns running status and buffer information for the current stream operation, examples:

```
Running                                 // Stream is running
Stripes Buffered: 1167 of 8388608       // 1167 stripes in buffer

Stopped: User                           // Stream stopped at user command
Stripes Buffered: 94 of 8388608         // 94 stripes still to read in buffer
```

```
stream text [all|number of stripes]
```

The option `all` is subject to a limit of 4096 stripes. To ease timing considerations, QIS implements an internal buffer of approximately 8 million stripes.

Returns N stripes in formatted ASCII text from the stream buffer. A "stripe" is defined as all available measurements for a single point in time.

The number of data points will depend on the current record settings of the module you are connected to:

```
[RecordNumber] [StatusFlags] [n data points, as stream header]
```

```
stream text 3
0 0 7 2 8 49
1 0 7 2 8 50
2 0 7 2 8 50
```

In this case we see 3 stripes. These are numbered 1, 2, 3 since the start of the stream. From this number and the stripe averaging mode we can know the exact time of each stripe.

- `[StatusFlags]` value of 0 = No Trigger, NonZero = Trigger Present
- 4 data points are present (power is disabled). Default channel ordering is:
  5V Voltage, 5V Current, 12V Voltage, 12V Current, 5V Power, 12V Power
- When there is no data to return, a line containing `eof` is returned

```
stream bin [all|number of stripes]
```

The option `all` is subject to a limit of 4096 stripes. To ease timing considerations, QIS implements an internal buffer of approximately 8 million stripes.

Returns raw format data from the instrument stream, with each sample encoded as a Java native 32 bit integer.

This is more complex than processing the text data, but the processing overhead is potentially lower if you want to save data directly without converting it to strings.

© Quarch Technology 2026

3.3. Module Commands

Any commands that are not listed above are assumed to be aimed at a module.

If a default module is selected, the command format is:

```
>hello?
HD Programmable Power Module +Triggering
```

If no default is selected, the DeviceConnection string must be added to each command:

```
>usb::qtl1824-03-161 hello?
XLC Programmable Power Module +Triggering
```

This combination allows easy control of the module, and QIS features at the same time.

© Quarch Technology 2026

4. Command Reference

This section lists the commands that can be sent to QIS over the TCP/Telnet interface (port 9722) or the built-in terminal. Commands may be sent as UTF-8 text with the standard line terminator \r\n denoting the end of a command.

Commands prefixed with a `$` character are processed by QIS directly, to control aspects of the QIS server. All other commands are processed by a device and must be addressed to a specific device by including a device identifier as a prefix, unless a default device has first been selected with `$default`.

- $about
- $default [deviceListEntryNumber|deviceIdentifier|connectionPath]
- $default?
- $get filters
- $help
- $history
- $list [details] [extended]
- $load filter [directory path]
- $loglevel [level]
- $loglevel?
- $scan [address]
- $scan?
- $shutdown
- $version
- close
- hello?
- rec stop
- rec stream
- stream bin [all|number of stripes]
- stream buffer text [all|number of stripes]
- stream create channel chan(chName,chGroup) function(arg1, arg2, ...)
- stream created channel chan(chName,chGroup)?
- stream created channel delete chan(chName,chGroup)
- stream created channel enable chan(chName, chGroup) [ON|OFF]
- stream created channels [enabled|disabled]?
- stream created channels clear
- stream created channels?

© Quarch Technology 2026

- stream created function definitions?
- stream mode header [v1|v2|v3]
- stream mode power [enable|disable|?]
- stream mode power total [enable|disable|?]
- stream mode resample 1..n[ms|us]
- stream mode resample group g 1..n[ms|us]
- stream mode resample off
- stream mode resample?
- stream text [all|number of stripes]
- stream text header
- stream?
- tcp::DeviceIdentifier forceIPAddress aaa.bbb.ccc.ddd
- tcp::DeviceIdentifier forcereset

4.1. $about

Returns information about this application (only available from a terminal)

Back to index

4.2. $default [deviceListEntryNumber|deviceIdentifier|connectionPath]

Specifies a default device to receive addressed commands. Once set, addressed commands can omit the device identifier prefix, e.g. "USB::QTL2312-01-190 hello?" can instead be sent as "hello?".

Arguments

#	Argument	Type	Required	Description
1	`deviceListEntryNumber`	`INT`	Yes	The numbered entry of the device to set as default, taken from the $list output, e.g. 2
1	`deviceIdentifier`	`INT`	Yes	The device identifier of the device to set as default, e.g. USB::QTL2312-01-190
1	`connectionPath`	`STRING`	Yes	The connection path of the device to set as default, e.g. tcp::192.168.1.90

Returns: The currently set default device, or an error message if the specified device was not found.

© Quarch Technology 2026

Notes

Only one of the three parameters should be specified, and it is an error to specify more than one.

The short form $def can be used instead of $default.

Examples

```
$default 2
```

```
$default USB::QTL2312-01-190
```

```
$default tcp::192.168.1.90
```

See also: $list [details] [extended]

Back to index

4.3. $default?

Displays the current default device. This will be used for all addressed commands that do not specify a device identifier.

Returns: The currently set default device, or none if no default device has been set.

Notes

The short form $def? can be used instead of $default?.

Examples

```
$default?
```

Back to index

4.4. $get filters

Returns the name and sample rate of the loaded filter sets

Back to index

4.5. $help

Displays copyright, license and help information

Returns: The help information

Back to index

4.6. $history

Returns a list of previously entered commands in this session (only available from a terminal)

Back to index

4.7. $list [details] [extended]

Lists all the devices that are known to QIS as a result of the $scan command, optionally including details of the device capabilities.

Arguments

#	Argument	Type	Required	Description
1	details	STRING	No	Sets whether the command output will have additional details about the capabilities of the device. Include the string "details" in the command.
2	extended	STRING	No	Sets whether the command output will have additional details about the firmware of the device. Include the string "extended" in the command. This must be used with the "details" parameter

Returns: A numbered list of devices showing the available connection type (e.g. USB) and the ID of the device. If the details flag is included then the output will also show additional information, e.g. whether the device is currently locked by another user.

Examples

```
$list
```

```
$list details
```

```
$list details extended
```

Back to index

4.8. $load filter [directory path]

Loads all filter XML files found in the provided directory path

© Quarch Technology 2026

Back to index

4.9. $loglevel [level]

Sets the current log-level.

Arguments

#	Argument	Type	Required	Description
1	`level`	`STRING`	Yes	The log-level to set *Acceptable Values:* `OFF, FATAL, ERROR, WARN, INFO, DEBUG, TRACE, ALL`

Returns: OK if the log-level was set successfully, or an error message if the specified level was invalid.

Examples

```
$loglevel INFO
```

Back to index

4.10. $loglevel?

Displays the current log-level

Returns: The current log-level, one of: [OFF, FATAL, ERROR, WARN, INFO, DEBUG, TRACE, ALL]

Back to index

4.11. $scan [address]

Refreshes the list of known network devices. This command returns immediately while the scan continues in the background - it is recommended to wait for around 3 seconds before using the $list command to see the results, or before selecting a device. If a network address is specified then QIS will query for a device at that address instead of performing a general scan.

Arguments

© Quarch Technology 2026

#	Argument	Type	Required	Description
1	address	STRING	No	An address on the network to check for a device. This should be an IP address, e.g. 192.168.1.90, and can optionally be prefixed with the protocol, e.g. tcp::192.168.1.90

Returns: For a general scan, a status message indicating that the scan has started. For a scan of a specific address, a status message indicating whether a Quarch device was found at the specified location.

Notes

Omit the address parameter to perform a general scan of the network. This command will not return any information about the devices found - use the $list command to see the results of the scan.

Examples

```
$scan
```

```
$scan tcp::192.168.1.90
```

```
$scan 192.168.1.90
```

See also: $list [details] [extended]

Back to index

4.12. $scan?

Returns the state of USB and Ethernet scanners. Ethernet information includes scan time in mS

See also: $scan

Back to index

4.13. $shutdown

Shuts down the QIS application

Returns: None

Notes

Using this command will terminate the QIS application and any open connections e.g. scripts or QPS will lose their connection to QIS.

Examples

```
$shutdown
```

Back to index

4.14. $version

Returns the QIS version number

Back to index

4.15. close

Closes the default or specified device connection

Back to index

4.16. hello?

Queries a device to check that it can be communicated with.

Returns: A response from the device, or an error

Notes

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 hello?
```

```
hello?
```

Back to index

4.17. rec stop

Stops streaming data from a device.

Returns: None

Notes

This is an addressed command - if there has not been a previous call to $default then this command

© Quarch Technology 2026

must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 rec stop
```

```
rec stop
```

Back to index

4.18. rec stream

Starts streaming data from a device.

Returns: None

Notes

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 rec stream
```

```
rec stream
```

Back to index

4.19. stream bin [all|number of stripes]

Returns a number of data stripes in compressed binary format from the stream buffer. This is used by QPS and is not usually required by the user.

Arguments

#	Argument	Type	Required	Description
1	number of stripes	STRING	Yes	One of "all" or an integer number of stripes.

Returns: A binary formatted representation of the data. The maximum number of stripes returned will never exceed 4096 per call - if fewer stripes are available than requested, the smaller number is returned immediately.

Notes

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 stream bin 3
```

```
stream bin all
```

Back to index

4.20. stream buffer text [all|number of stripes]

PAM only - text is derived directly from the buffer and returned in a simple record by record format

Back to index

4.21. stream create channel chan(chName,chGroup) function(arg1, arg2, ...)

Creates a new synthetic channel

Back to index

4.22. stream created channel chan(chName,chGroup)?

Returns "yes" if the specified channel/group is a synthetic channel, else returns "no"

Back to index

4.23. stream created channel delete chan(chName,chGroup)

Deletes the specified synthetic channel

Back to index

4.24. stream created channel enable chan(chName, chGroup) [ON|OFF]

Enables/disables the specified synthetic channel

Back to index

4.25. stream created channels [enabled|disabled]?

Returns a list of all enabled/disabled synthetic channels

Back to index

4.26. stream created channels clear

Deletes all synthetic channels

Back to index

4.27. stream created channels?

Returns a list of all synthetic channels

Back to index

4.28. stream created function definitions?

Returns the XML schema used to define the supported synthetic channel functions

Back to index

4.29. stream mode header [v1|v2|v3]

Selects the version to be returned when the stream text header command is sent. Using the latest version is recommended for new development.

Arguments

#	Argument	Type	Required	Description
1	`version`	`STRING`	Yes	The version of the header to use. *Acceptable Values:* `v1, v2, v3`

Returns: None

Notes

© Quarch Technology 2026

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 stream mode header v3
```

```
stream mode header v3
```

Back to index

4.30. stream mode power [enable|disable|?]

Enables or disables power calculation. If voltage and current values are available for a given channel then a corresponding power channel will be added.

Arguments

#	Argument	Type	Required	Description
1	`enable`	`STRING`	Yes	Whether to enable or disable power calculation. *Acceptable Values:* `enable, disable`

Returns: None

Notes

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 stream mode power enable
```

```
stream mode power disable
```

Back to index

4.31. stream mode power total [enable|disable|?]

Enables or disables power calculation and total power calculation. In addition to enabling the channels as with stream mode power, if there are two or more then a total channel will be added as well.

Arguments

#	Argument	Type	Required	Description
1	`enable`	`STRING`	Yes	Whether to enable or disable total power calculation. *Acceptable Values:* `enable, disable`

Returns: None

Notes

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 stream mode power total enable
```

```
stream mode power total disable
```

Back to index

4.32. stream mode resample 1..n[ms|us]

Forces stream data to a specified resample period, allowing arbitrary resampling to any required time-base. This involves averaging every sample into the new scale, so no information is lost (average power will always remain the same, for example). Resampling can only reduce the number of samples - if a resample time is selected which is smaller than the current instrument rate, the instrument setting will be used. This command can also be used to query the current setting.

Arguments

#	Argument	Type	Required	Description
1	`period`	`STRING`	Yes	An integer value followed by a unit in this set: us, ms, s. Or "off" to turn off the resample.

Returns: None

Notes

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 stream mode resample?
```

```
stream mode resample 100mS
```

Back to index

 © Quarch Technology 2026

4.33. stream mode resample group g 1..n[ms|us]

Forces stream data for a specific group of channels to a specified resample period. See "stream mode resample" for details of how resampling works.

Arguments

#	Argument	Type	Required	Description
1	`group number`	`INT`	Yes	Applies the resample only to the specified group of channels.
2	`period`	`STRING`	Yes	An integer value followed by a unit in this set: us, ms, s. Or "off" to turn off the resample.

Returns: None

Notes

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
stream mode resample 1 10uS
```

See also: stream mode resample 1..n[ms|us]

Back to index

4.34. stream mode resample off

Disables stream resampling

Back to index

4.35. stream mode resample?

Returns the resampling state

Back to index

© Quarch Technology 2026

4.36. stream text [all|number of stripes]

Returns a number of data stripes as text from the stream buffer.

Arguments

#	Argument	Type	Required	Description
1	`number of stripes`	`STRING`	Yes	One of "all" or an integer number of stripes.

Returns: UTF-8 text of the data stripes. Values are separated by spaces, stripes are separated by new lines, in the form: [Stripe number] [Status] [Data channel 0 value] [Data channel 1 value] [...]. The maximum number of stripes returned will never exceed 4096 per call - if fewer stripes are available than requested, the smaller number is returned immediately.

Notes

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 stream text 3
```

```
stream text all
```

Back to index

4.37. stream text header

Gets the stream header information from the most recent stream to be run on the selected module. This will only return a value if a stream has previously been started on the device.

Returns: A header encoded in XML format. A value will only be available if a stream has been started on the device.

Notes

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 stream text header
```

```
stream text header
```

Back to index

© Quarch Technology 2026

4.38. stream?

Queries whether a device is currently streaming.

Returns: Whether the stream is running or stopped, and a count of the number of stripes currently buffered

Notes

This is an addressed command - if there has not been a previous call to $default then this command must be prefixed by a device identifier.

Examples

```
USB::QTL2312-01-190 stream?
```

```
stream?
```

Back to index

4.39. tcp::DeviceIdentifier forceIPAddress aaa.bbb.ccc.ddd

Forces a reset of the specified IP address

Back to index

4.40. tcp::DeviceIdentifier forcereset

Forces a reset of the specified device

Back to index

© Quarch Technology 2026

5. Example Usage

5.1. Simple setup and streaming

Below is a simple example, setting up the system and recording the power up of a device.

```
# Refresh the scan table
$scan
# Connect to the specified module
$default tcp::QTL1999-02-010
# Set the record averaging to 1024 = 4.096mS/sample
record:average 1024
# Set the voltages levels for the output
signal:12v:voltage 12000
signal:5v:voltage 5000
# Enable power calculation
stream mode power enable
```

```
# Begin streaming, QIS will now start buffering stripes
record stream
# Enable the outputs to power up the device
run:power up
```

```
# Sleep here, until the device has fully powered up...
```

```
# Stop streaming
record stop
```

```
# Start downloading the buffered data in 500 stripe sections (this can also be done
while still streaming)
# Loop until response contains "eof"
stream text 500
```


© Quarch Technology 2026