Quarch Technology Ltd

MiniSAS HD Physical Layer Switch

Technical Manual

For use with:

QTL1564 – 12 Gbps MiniSAS HD Physical Layer Switch

Thursday, 13 October 2016

Change History

1.0	3rd April 2012	Initial Release
1.1	24th October 2012	Updated for 1U enclosure
		Added Ethernet setup commands
1.2	10th April 2015	Added updates for ReST API
1.3	13th October, 2016	Updated to new style manual

Contents

About This Manual

Thank you for your purchase of this Quarch Module. This manual is intended as a technical reference, describing the features, operations and control API of the unit.

Additional documents available include Quick start guides, Datasheets and Application notes.

Scope and the Purpose

This manual provides the following:

- An overview of the QTL1564 12 Port MiniSAS HD Physical Layer Switch.
- General description of the module and its features
- Details of its control interfaces
- Full command spec
- Where to find additional help

MiniSAS HD Physical Layer Switch at a glance

This chapter briefly describes the module in a quick glance showing the various ports on the front and the back panel with its ports and connections.

Front Panel

Ports: There are 12 MiniSAS HD device ports on the switch numbered 1-12. These support passive, active and optical cables.

Back Panel

12v:	2.1mm Centre Positive Jack. For use with the supplied 12V PSU
LAN:	For control via a 10BASE-T network
Torridon:	For control via a Separate ‘Torridon’ controller
USB:	For control via a USB connection

Rack Mount

Dual 1U rack mount kit. This allows up to two switch units to be mounted in 1U.

Technical Specifications

Usage and Safety:

This device is part of the 'Torridon System'. The separate 'Torridon System Manual' contains important information regarding usage and safety that apply to all parts of the Torridon System. This Product is intended for use by experienced technical users in a test lab environment. You should ensure you are familiar with this information before using this device.

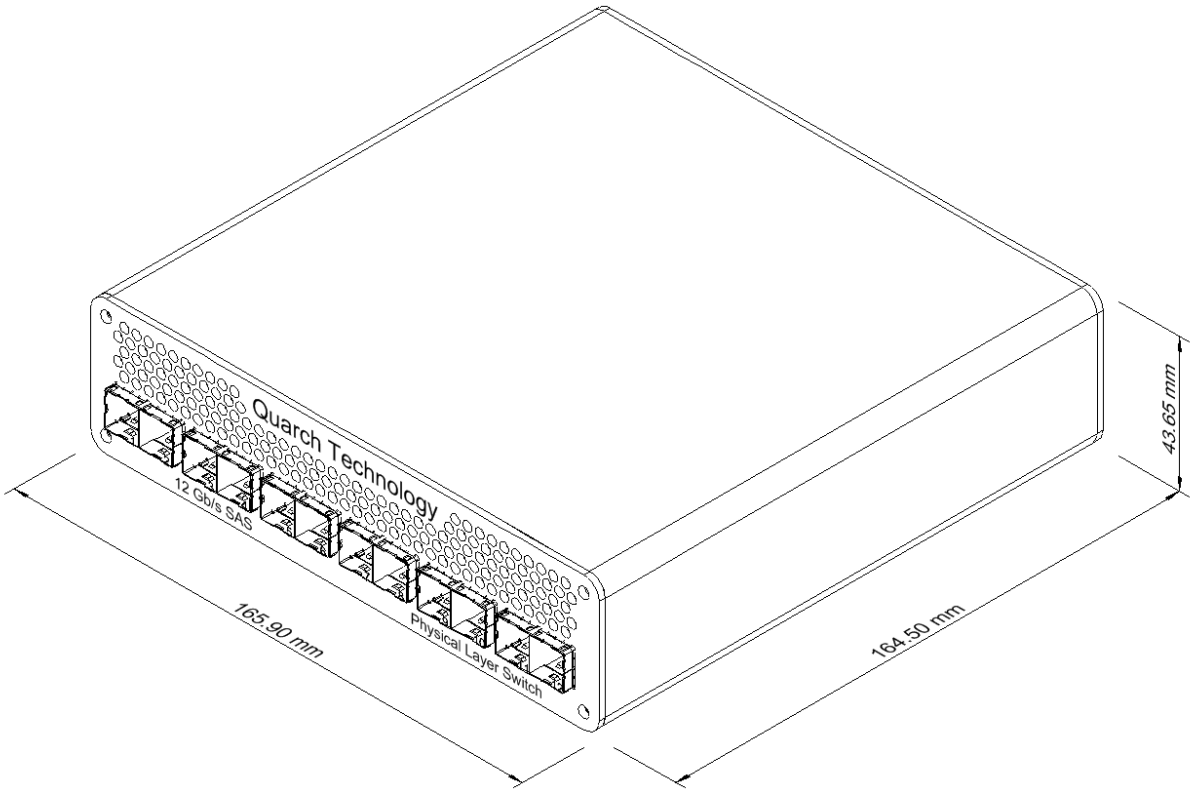
Power Requirements:

The module is supplied with a 12V 60W DC power supply. Only original Quarch manufactured or approved parts should be used with the system. The table below shows the electrical characteristics of the module

Parameter	Min	Typical	Max
Maximum Data Rate	12 GBps		
Input differential voltage range	250mV	-	1500 mV
Output differential voltage swing (Programmable)	600mV	-	1200mV
Output rise/fall time (20%-80%)	-	35 pS	-
Deterministic output jitter	-	0.16UI	-
Random output jitter	-	60 mUI pp	-
Propagation delay	0.25 nS	-	1.0 nS
Channel to channel skew	-	35 pS	-

Mechanical Characteristics:

The Figure shows the mechanical dimensions of the QTL1564 MiniSAS HD Physical Layer Switch

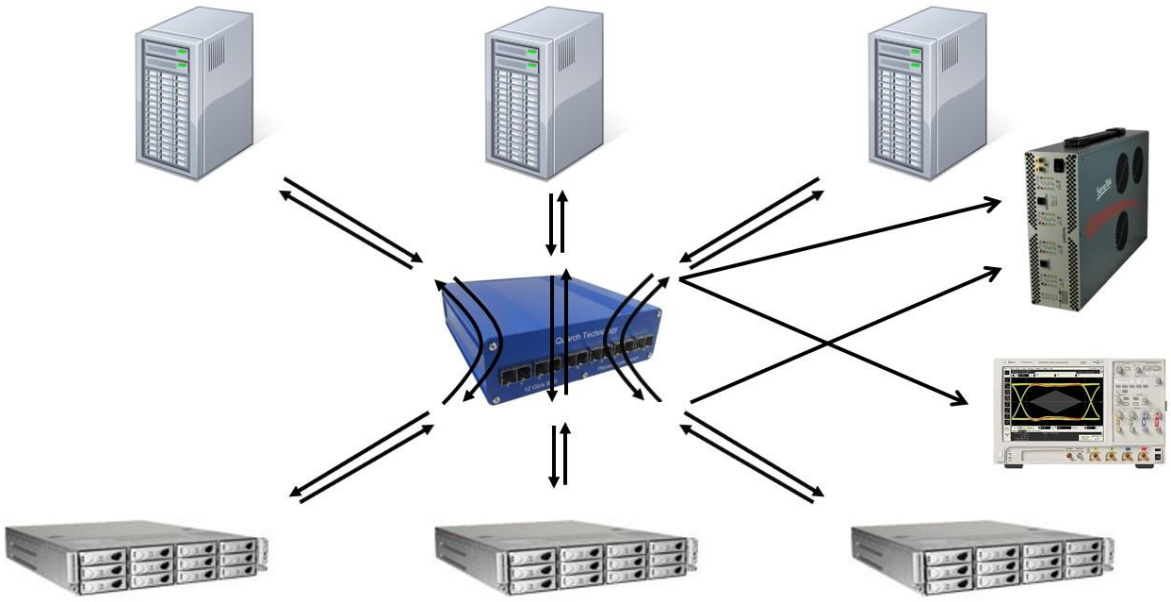


Introduction

The MiniSAS HD Physical Layer Switch has 12 MiniSAS HD device ports. Connections can be automatically and remotely routed between any one of the 12 ports and their individual lanes. This provides a fully automated solution for remote re-cabling of systems.

Data rates of up to 12Gbps per lane are supported. The product is based around a fully non-blocking asynchronous cross point switch.

Connections can be made on a port-by-port or lane-by-lane basis for a high degree of control. It is also possible to forward data to multiple destinations, allowing copies of the data to be sent to additional pieces of test equipment.



Received data can be sent to multiple ports to connect test equipment without inserting it into the signal path.

Features

- Fully automate any test that requires reconfiguration of SAS data paths
- Programmable Amplitude, Equalisation and Pre-Emphasis available on each port
- Each signal is re-driven by the module resulting in high signal quality whilst preserving the timing characteristics of the original.
- Supports Active and Optical cables
- Can be used with single or dual MiniSAS HD cables, including active copper and optical cables.
- Split one wide port cable into 4 single lanes
- Switch one target between multiple initiators or multiple initiators between one target.
- Dual 1U rack mount kit available.

Basic Concepts

The module's primary function is to switch HD MiniSAS data paths between its 12 output ports. Each output driver on the cross point switch is connected to a single receiver, to create a bidirectional link between two ports, each ports transmitter is connected to the other ports receiver.

Each MiniSAS HD port is comprised of 4 lanes which may be configured as a group or separately. The incoming signal is re-driven but not re-timed. This gives high signal quality whilst preserving the timing characteristics of the original signal. The signal received on one lane may be transmitted on any number of other lanes.

Examples:

To connect to two Ports, use the command below

MUX:CONnect 1 6

The data can be forwarded from one port to another with the command

MUX:FORWard 1 7

Port can be turned OFF with the command below.

MUX:OFF 1 - Will Turn off Port 1

Each port support Active Cables compliant to SFF-8449 in any operating mode. Cable management Interface data may be read from and written to each port individually through the command line interface.

Programmable Amplitude, Equalisation and Pre-emphasis is available on each port.

Control Interfaces

The Switch Module has three physical ports:

- USB
- LAN
- Torridon (8-pin Connector)

Which supports following communications:

Native USB

The USB 2 port requires the Torridon USB driver on Windows, or the LibUSB driver if you are developing your own code.

Native LAN

This 10BASE-T port can link into your network and supports NetBIOS and DHCP. Both Telnet (single connection only) and REST API can be used for control

Torridon Port

Using an external controller allows multiple modules to be controlled as part of a combined test system. The controller can also add additional connection options. All controller options support USART Serial, which would allow direct Serial control of the switch from a PC or terminal server.

Controlling the Module

All control interfaces use the same simple, text command set. Each command is human readable ASCII text of up to 64 characters in length.

All responses to commands are 1 or more lines, of up to 64 characters each. This makes issuing commands and parsing responses very simple.

Commands can be sent from a standard terminal program, such as Putty, TerraTerm or command line Telnet clients. They can also be generated from scripts or user code (Perl, Python, C, .NET and more).

Serial Control

Any script, code or terminal program capable of opening a serial port can use this communication method.

Serial Settings

The table below shows the serial settings

Settings	Required value
Baud Rate	19,200
Data Bits	8
Stop Bits	1
Handshaking	None or Hardware

Ethernet/LAN Control

Telnet Terminal

This mode uses exactly the same commands as the serial ASCII terminal, but sent over a standard LAN Telnet connection. This can be done from any standard Telnet client, or most scripting languages such as Perl and Python. Only one Telnet connection can be open any time.

ReST

ReST is a standard method of accessing a remote device over a network. It used standard http web technology, and commands can be sent from a web browser or script language. ReST command calls are formatted as:

http://192.168.1.99/*IDN

USB Control

Quarch's TestMonkey application can control the Switch module via USB, this allows simple graphical control of the module. Alternatively you can use Torridon API to connect to the Switch. For more advanced users, we can also supply USB examples for Python, .NET and Java.

USB uses the same commands as the other modes, but the correct driver must be present. There are also some additional commands to be sent to ensure the module is in the correct mode. Please contact us for the latest examples.

Command Set

These commands are based on the SCPI style control system that is used by many manufacturers of test instruments. The entire SCPI specification has NOT been implemented but the command structure will be very familiar to anyone who has used it before.

- SCPI commands are NOT case sensitive
- SCPI commands are in a hierarchy separated by ':' (LEVel1:LEVel2:LEVel3)
- Most words have a short form (e.g. 'register' shortens to 'reg'). This will be documented as REGister, where the short form is shown in capitals.
- Some commands take parameters. These are separated by spaces after the main part of the command (e.g. "meas:volt:self 3v3?" Obtains the 3v3 self test measurement)
- Query commands that return a value all have a '?' on the end
- Commands with a preceding '*' are basic control commands, found on all devices
- Commands that do not return a particular value will return "OK" or "FAIL". Unless disabled, the fail response will also append a text description for the failure if it can be determined.

Basic Commands

[comments]

Any line beginning with a # character is ignored as a comment. This allows commenting of scripts for use with the module.

*RST

Triggers a reset, the module will behave as if it had just been powered on

*CLR

Clear the terminal window and displays the normal start screen. Also runs the internal self test. The same action can be performed by pressing return on a blank line.

*IDN?

Displays a standard set of information, identifying the device. An example return is shown below

```
Family: Torridon System                  [The parent family of the device]
Name: Ethernet Cable Pull Module  [The name of the device]
Part#: QTL1271-01                        [The part number of the hardware]
Processor:  QTL1159-01,3.50          [Part# and version of firmware]
Bootloader: QTL1170-01,1.00        [Part# and version of bootloader]
FPGA 1: 1.0                              [Version of FPGA core]
```

*TST?

Runs a set of standard tests to confirm the device is operating correctly, these tests are also performed at start up. Returns ‘OK’ or ‘FAIL’ followed by a list of errors that occurred, each on a new line.

General Setup Commands

CONFig:MODE BOOT

Configures the card for boot loader mode (to update the firmware), requires an update utility on the PC.

CONFig:MESSages [SHORt|USER]
CONFig:MESSages?

Gets or sets the mode for messages that are returned to the user's terminal

Short: Only a "FAIL" or "OK" will be returned

User: Full error messages are returned to the user on failure

CONFig:TERMinal [USER|SCRIPT]
CONFig:TERMinal?

Sets the ASCII terminal configuration mode:

Returns the current terminal mode

USER: Standard mode for manual control via HyperTerminal etc.

SCRIPT: Does not echo characters, sends <CR> after the cursor. This is to assist scripting, especially from Linux based systems.

CONFig:DEFault STATE

Resets the state of the module. This will set all connections and signal conditioning settings to its default power-on values. Terminal setting will not be affected. This command allows the module to be brought back to a known state without resetting it.

CONFig:DEFault FACTory

Resets the state of the module back to factory defaults. This will set ALL settings to the factory default state. This also resets all persistent EEPROM stored settings, such as Ethernet/IP options.

Switch Signal Names

The switch allows full connections between ports and also partial connections.

Port Name	Description
[PORT]	Every lane of port [PORT] (where PORT is a number from 1-12).
[PORT].[LANE]	Specific Lane on the specified port (where LANE is a number from 0 to 3)

Switch Commands

MUX:CONnect [PORT] [PORT]

Creates a bidirectional connection of all signals between the two ports, any other connections or forwards to/from either port are turned off. [PORT] must be a number from 1 to 12.

The CONFig:MUX:DELay command specifies the time between disconnect and re-connect. It is not possible to run a second connection command until the first is complete

MUX:CONnect [PORT].[LANE] [PORT].[LANE]

Connects one lane of one port to one lane on an another. [Port] must be a number from 1 to 12 and [Lane] must be a number from 0 to 3. Any previous connection to either lane is removed.

The CONFig:MUX:DELay command specifies the time between disconnect and re-connect. It is not possible to run a second connection command until the first is complete

MUX:FORward [PORT 1] [PORT 2]

Forwards all traffic received at [PORT 1] to [PORT 2] , This differs from MUX:CONnect [PORT] [PORT] because it is unidirectional and does not remove existing connections or forwards to other ports.

MUX:FORward [PORT 1].[LANE] [PORT 2].[LANE]

Forwards [Port 1]. [Lane] to [Port 2]. [Lane]. This differs from MUX:CONnect [PORT].[LANE] [PORT].[LANE] because it is unidirectional and does not remove existing connections or forwards to other ports.

MUX:[PORT]:SOURce?

Returns the ports or lanes that [PORT] is receiving data from

MUX:[PORT].[LANE]:SOURce?

Returns the lane that [PORT].[LANE] is receiving data from

MUX:OFF [PORT|ALL]

Turns off the transmitter for [Port], or if ALL is used, it turns off all ports

MUX:OFF [Port].[Lane]

Turns off the transmitter for the specified [Lane] on the specified [Port].

Switch Config Commands

CONFig:MUX:DELay #

Set the duration in seconds between disconnecting and reconnecting when making a connection. CONFig:MUX:Delay 0 gives approx. 1ms switching time. Ranges below 0.1 seconds will become less accurate.

CONFig:MUX:DELay?

Return the current connection delay.

CONFig:MUX:[PORT]:PREemphasis #

CONFig:MUX:[PORT]:PREemphasis?

Set the level of transmit de-emphasis for the [PORT] according to the following table:

0	No De-Emphasis
1	~1.5 dB output de-emphasis
2	~3 dB output de-emphasis
3	~4.5 dB output de-emphasis
4	~6 dB output de-emphasis
5	~7.5 dB output de-emphasis
6	~9dB output de-emphasis
7	~10.5dB output de-emphasis

CONFig:MUX:[PORT]:EQUalisation #
CONFig:MUX:[PORT]:EQUalisation?

Set the level of receiver equalization on [PORT] from 0 (minimum) to 31 (maximum). All values can be used, the values not displayed in the table will provide an intermediate value.

0	Minimum Equalisation (~3dB)
5	~8dB Equalisation
9	~12dB Equalisation
18	~17dB Equalisation
20	~22dB Equalisation
27	~24dB Equalisation
31	~27dB Equalisation

CONFig:MUX:[PORT]:AMPlitude #
CONFig:MUX:[PORT]:AMPlitude?

Set the transmitter differential amplitude for [PORT] from 0 (minimum) to 2 (maximum)

0	600mVp-p
1	800mVp-p
2	1200mVp-p

CONFig:MUX:LOAD:MAPping [ON|OFF]
CONFig:MUX:LOAD:MAPping?

Sets/Gets the enable state for switch mapping persistence. When turned ON, the switch will store the current port mapping in EEPROM. On power cycle, the mapping will be restored. Defaults to OFF.

Changes to the saved EEPROM settings will only occur when the flag is ON.

Ethernet Commands

CONFig:ETHernet:NAME[xxxx]
CONFig:ETHernet:NAME?

Sets the NetBios name of the unit. This command will automatically store the settings in EEPROM. You will have to reset/power cycle the unit for the changes to take effect.

CONFig:ETHernet IP xxx.xxx.xxx.xxx
CONFig:ETHernet IP?

Set and read the current IP address. Setting a new IP address will disable the DHCP function (if enabled). To return to automatic DHCP addressing, the DHCP setting must be manually re-enabled.

CONFig:ETHernet MASK xxx.xxx.xxx.xxx
CONFig:ETHernet MASK?

Set and read the current IP mask. Setting a new IP mask will disable the DHCP function (if enabled). To return to automatic DHCP addressing, the DHCP setting must be manually re-enabled.

CONFig:ETHernet:DNS[1|2] [x.x.x.x]
CONFig:ETHernet DNS[1|2]?

Sets the DNS of the unit. This command will automatically store the setting in EEPROM. You will have to reset/power cycle the unit for the change to take effect.

CONFig:ETHernet:GATE[x.x.x.x]
CONFig:ETHernet GATE?

Sets the Gateway of the unit. This command will automatically store the setting in EEPROM. You will have to reset/power cycle the unit for the change to take effect.

CONFig:ETHernet DHCP [ON|OFF]
CONFig:ETHernet DHCP?

Enables and Disables the DHCP client. When enabled, module will try to obtain an IP address from the local DHCP server.

CONFig:ETHernet:REST [ON|OFF]
CONFig:ETHernet:REST?

Enables and Disables the ReST Mode. This command will automatically store the settings in EEPROM. You will have to reset/power cycle the unit for the changes to take effect.

Voltage Measurements

The module is capable of measuring various voltages both for self-test and to assist in the testing of a customer's system. The following measurement points are available:

Measurement Command	Description	Resolution / Accuracy
MEASure:VOLTage:SELF 1v2?	Returns the voltage of the modules internal 1.2v power rail	64mV / 5%
MEASure:VOLTage:SELF 1v8?	Returns the voltage of the modules internal 1.8v power rail	64mV / 5%
MEASure:VOLTage:SELF 3v3?	Returns the voltage of the modules internal 3.3v power rail	64mV / 5%
MEASure:VOLTage:SELF 12v?	Returns the voltage of the modules internal 12v power rail	64mV / 5%

Default Start up State

On initial power up or reset, the control modules enter a default state. Each port is connected to its neighbour as shown below. The switch does not differentiate between Initiators and Targets, an Initiator may be plugged into any port as may a target, and this may result in illegal SAS configurations if the device is miss-cabled or set up incorrectly but it will not cause damage to any equipment.

Switch Settings:

Port	Connected to	Persists
1	2	N[*1]
3	4	N[*1]
5	6	N[*1]
7	8	N[*1]
9	10	N[*1]
11	12	N[*1]

[*1] Persistence can be turned through a config command

Default Signal Conditioning Settings

Signal Conditioning	Default	Persists
Pre-emphasis	0, No De-Emphasis	N[*1]
Equalisation	0,Minimum Equalisation (~3dB)	N[*1]
Amplitude	2,Maximum (1200mVp-p)	N[*1]

[*1] Persistence can be turned through a config command

Control Settings

Setting	State	Persists
Terminal Mode	USER (Echo ON, no extra line feeds)	Y
Message Returns	USER (Detailed error responses)	Y
IP Address	192.168.1.99	Y
Mask	255.255.255.0	Y
DNS 1/2	0.0.0.0	Y
Gateway	0.0.0.0	Y
DHCP	Enabled	Y
NetBIOS Name	Serial number of device	Y
ReST enable	Enabled	Y
Mapping Persistence	Disabled	Y

Appendix

Customer Support

For further help and support, email us on

support@quarch.com

Software Updates and Drivers

For up to date software and drivers, visit our website

www.quarch.com

Documentation

You can also download the technical manuals from our website. To help you start quickly we can provide you the additional documents, such as examples in Perl, Python and C#, Telnet and Serial instructions and more, please contact us if you require access to these examples.