Quarch Technology Ltd

Torridon Lite Cable Pull Module

Technical Manual

For use with:

QTL1215 - Torridon Lite Drive Control Module

Using Quarch firmware version 3.5 and above

Thursday, 25 November 2010

Change History

1.1	06 August 2010	Initial Release
1.2	02 September 2010	Added Source Enable Commands (firmware 3.52 and higher)
1.3	17 November 2010	Corrected Switching Characteristics Section
1.4	25 November 2010	Updated common sections to new format

Contents

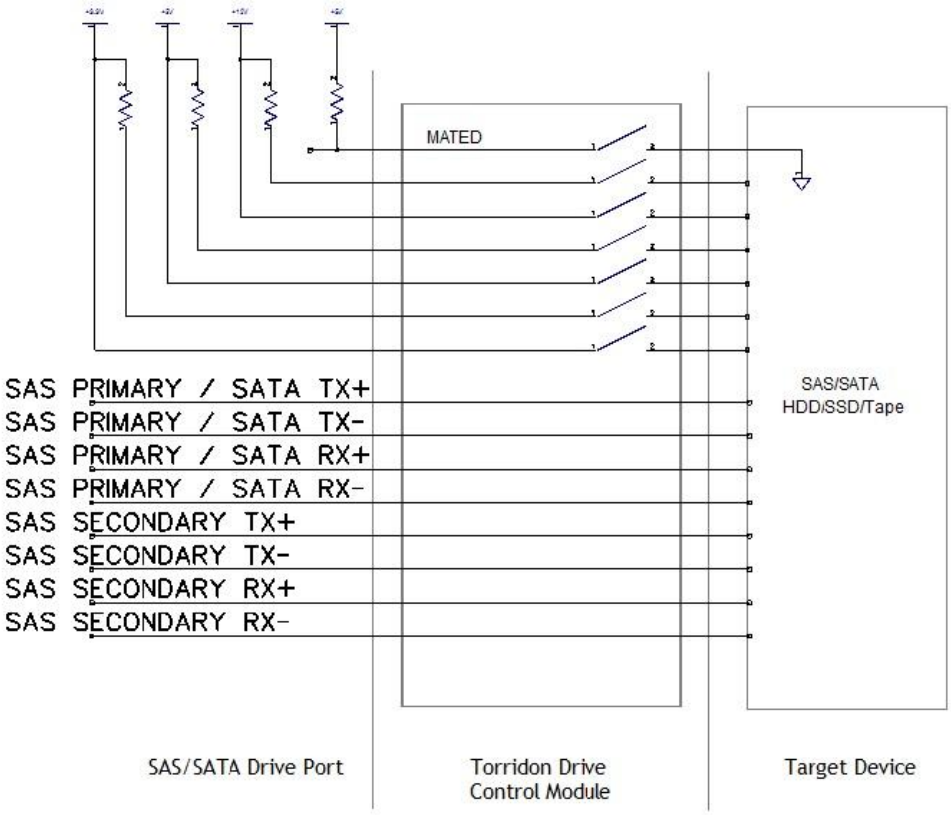
Introduction

The **Torridon Lite Drive Control Module** allows remote switching of the power supply rails and drive presence signal to a SAS or SATA Disk Drive for test automation or fault injection purposes.

These features allow the user to simulate a number of typical events in a storage system, such as hot insertion and removal of a disk drive and drive failure.

Each set of Power and Pre-Charge Pins are individually switched, allowing complete control over the power up sequence of a drive.

The Lite card supports two vendor specific drive presence circuits, please see the Drive Presence section later in this manual for a description. Quarch modules may be customized to support other proprietary signals on request.

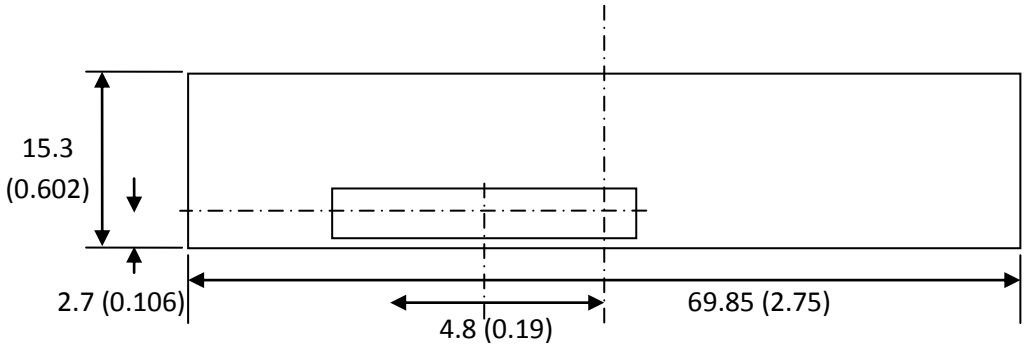


Technical Specifications

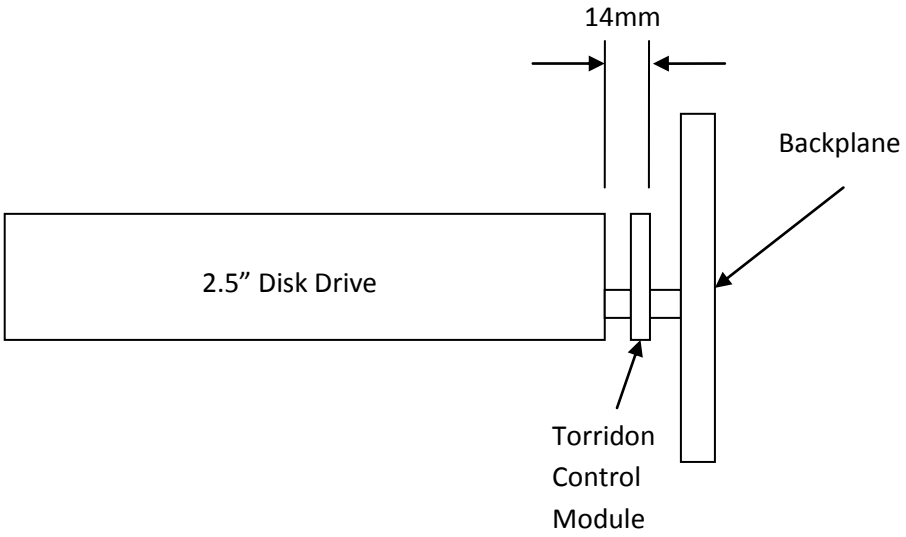
Switching Characteristics:

SAS Connector Pin	Description	Switching Action
S1,S4,S7,S8,S11,S14	SAS Data Ground Pins	Each signal individually connected between plug and receptacle and isolated from Module Ground
P5,P6,P10,P12	Power Ground Pins	All connected to digital ground on the module
S2,S3,S5,S6,S9,S10,S12,S13	SAS Data Signal pins	Each signal is individually connected between plug and receptacle
P1,P2	3.3V Power Pins	P1 on backplane is switched by 11A power FET and connected to P1 & P2 on drive side. P1 and P2 on backplane side can be connected together by 0.75A FET for drive detection (see section Drive Presence Support)
P3, P7, P13	3.3V, 5V and 12V Pre-Charge Power Pins	Each pin is individually switched by 11A power FET
P4	Ground / SPECIAL_1 (vendor specific mated)	Switched by 0.75A FET (see section Drive Presence Support)
P8,P9, P14, P15	5V and 12V Power Pins	The pins on each rail are connected together and switched by 11A power FET.
P11	Ready LED	Individually connected from plug to receptacle

Mechanical Characteristics:



- The Modules have the same cross section as a 2.5” disk drive, allowing them to fit into any 2.5” or 3.5” drive enclosure.



Control Interfaces

All Torridon Control Modules are designed to be used with a Torridon Array Controller (QTL1079) or a single Torridon Interface Card (QTL1144).

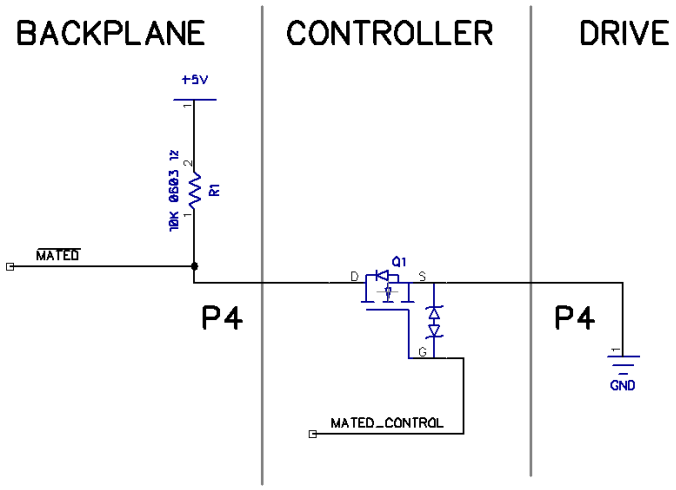
The control cable is an ultra-thin Flex cable.

Control Interface	**Form Factor**	**Torridon Module Ports**	**Control Methods Available**	**Interfaces**
Torridon Array Controller	1U 19” Rack Mounted unit	24 at the front, 4 at the rear	Serial Scripting Script Generation through **TestMonkey GUI**	Serial via DB9 or RJ45 Ethernet
Torridon Interface Card	102mm x 26mm PCB	1 port	Serial Scripting Script Generation through **TestMonkey GUI**	Serial via DB9 or RJ45

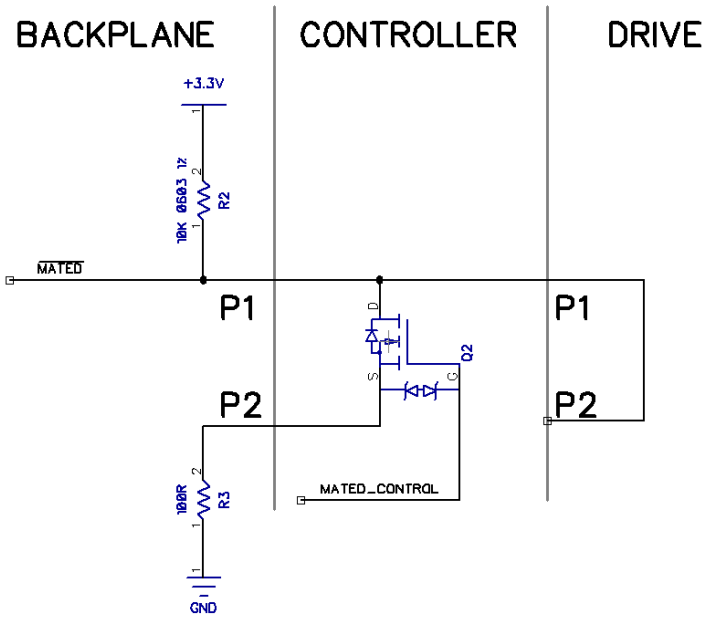
Drive Presence Support

The module supports two different proprietary drive presence circuits, all are switched by the same control signal SPECIAL_1. If the detection circuits are not implemented on the backplane this circuitry should have no adverse effect on drive operation.

GND pin P4 detection



P1/P2 continuity detection

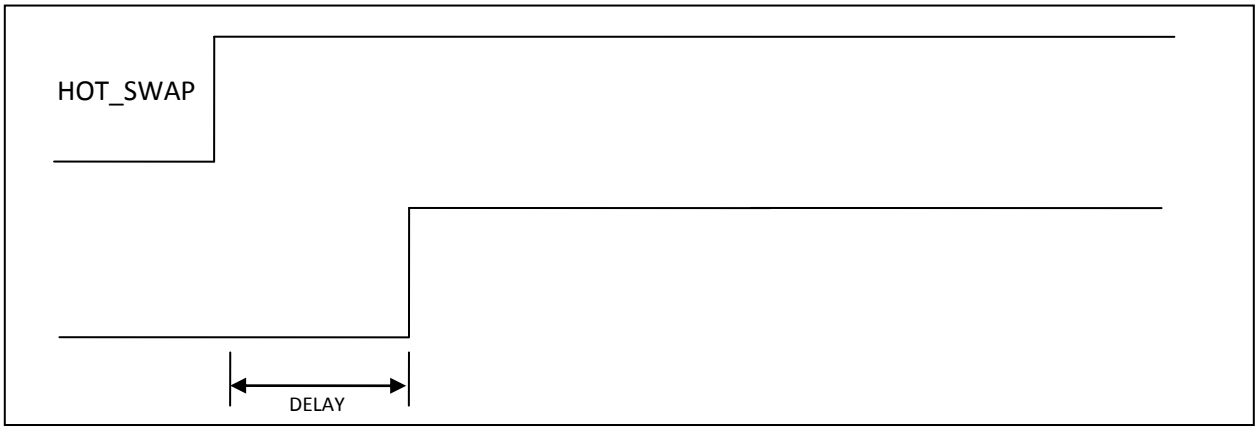


Basic Concepts

Each switch on the Torridon control cards is called a 'Signal' and can be programmed to follow one of 6 programmable delay profiles (called 'Sources'). This allows the user to sequence power and signal connections to the drive in up to six programmable steps.

Each of the programmable delay profiles is called a control source, S1 to S6. For each control source the user can set up a delay parameter. Three special sources (S0, S7 and S8) are also provided as described in the table below.

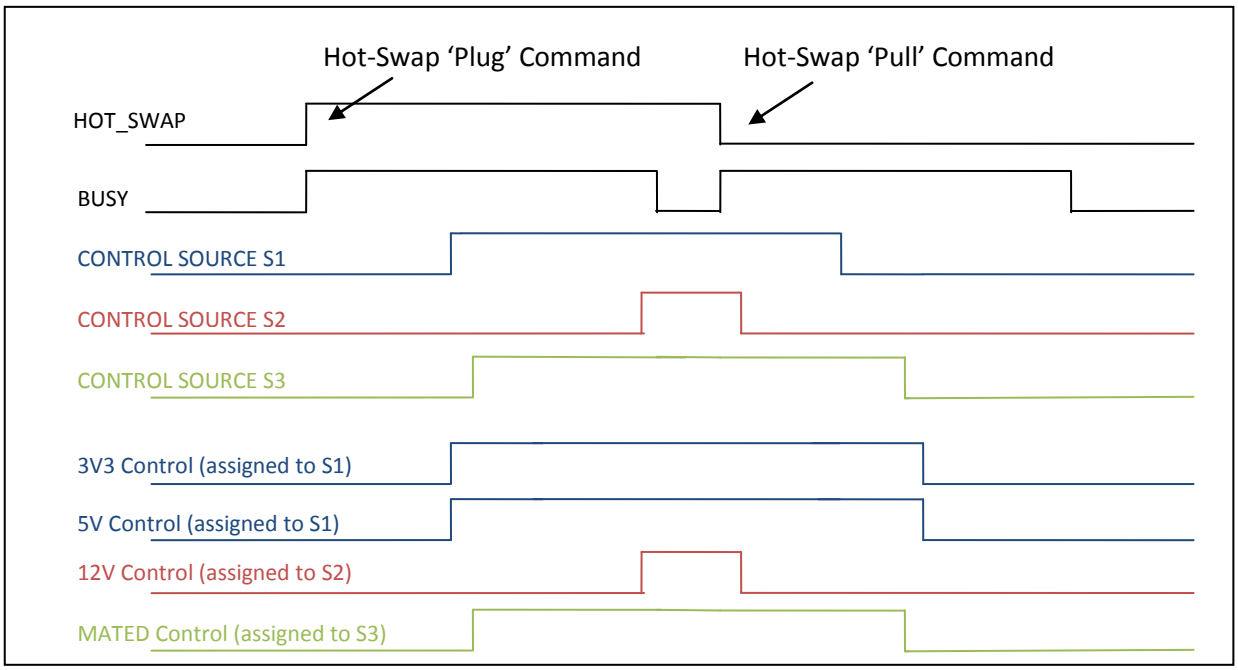
Control Source Parameters for a power up event:



Once each delay period is set up, the user assigns each signal to follow the relevant control source, then uses the "run:power up" and "run:power down" commands to initiate the hot-swap.

The BUSY bit 1 in the control register is set during a power up and power down operation. This may be used to monitor for the completion of timed events.

Power up and Power down example:

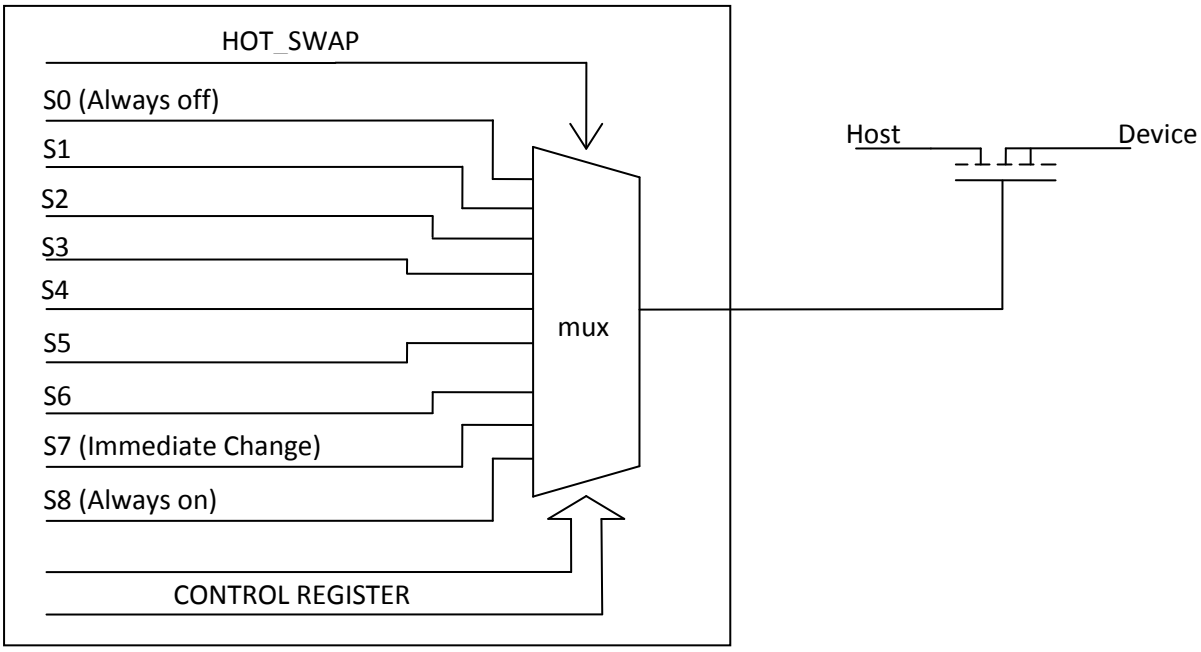


Signal Configuration

Each signal that is switched by the module is usually assigned to one of the 6 timed sources, S1 – S6. Each signal can also be assigned directly to 'always off' (source 0), 'immediate change' (source 7) or 'Always on' (source 8).

To assign a signal to a control source, write to its CONTROL_REGISTER:

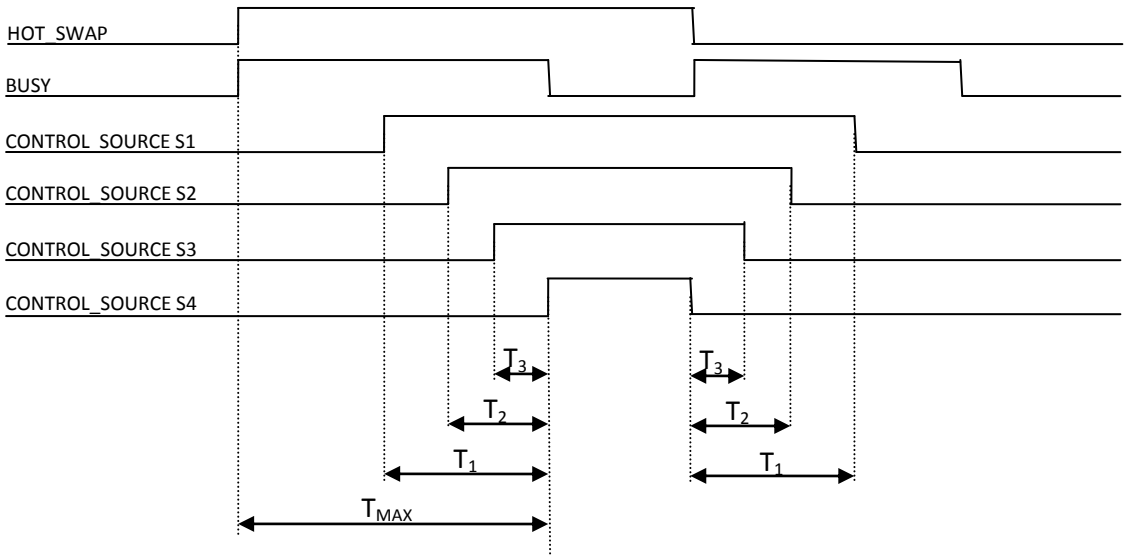
CONTROL_REGISTER Value	Description
0	Signal is always OFF
1	Signal assigned to control source 1
2	Signal assigned to control source 2
3	Signal assigned to control source 3
4	Signal assigned to control source 4
5	Signal assigned to control source 5
6	Signal assigned to control source 6
7	Signal changes with HOT_SWAP
8	Signal is always ON



This diagram shows the 9 possible source settings entering the control MUX for a switched signal. The value of the control register will determine which of the sources are used to control the signal. When enabled, the hot-swap line will cause the MUX to pass the control signal from that source through to the switch.

Power Up vs. Power Down Timing

Each control source is always configured with power-up parameters; the power-down profile is automatically generated by the module, and is the mirror image of the power up:



Power Down Times are automatically generated by the module but can be calculated as follows:

T_{MAX} = Largest Time Period
= the largest value from: (S1_DELAY),
(S2_ DELAY),

up to

(S6_ DELAY)

Power Down Delay T_X = T_{MAX} – (S_X_DELAY)

If you require a different power down sequence then you can alter any of the source timing values, pin bounce or signal assignments while the drive is in the plugged state. When you initiate the 'pull' action, the new settings will be used.

Voltage Measurements

The modules are 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 5v?	Returns the voltage of the modules internal 5v power rail	64mV / 3%
MEASure:VOLTage 3v3in?	Returns the voltage of the 3.3V power pins on the backplane (unswitched) side of the Module	64mV / 3%
MEASure:VOLTage 3v3out?	Returns the voltage of the 3.3V power pins on the drive (switched) side of the module	64mV / 3%
MEASure:VOLTage 5vin?	Returns the voltage of the 5V power pins on the backplane (unswitched) side of the Module	64mV / 3%
MEASure:VOLTage 5vout?	Returns the voltage of the 5v power pins on the drive (switched) side of the module	64mV / 3%
MEASure:VOLTage 12vin?	Returns the voltage of the 12V power pins on the backplane (unswitched) side of the Module	64mV / 3%
MEASure:VOLTage 12vout?	Returns the voltage of the 12v power pins on the drive (switched) side of the module	64mV / 3%

Default Startup State

On power up or reset, the control modules enter a default state. To make the module as easy to use as possible, the default state is a ‘standard’ hot-swap scenario with preset source and signal settings such that the “run:power up” command will immediately power up the drive without needing any initial setup.

The default hot-swap scenario will connect pre-charge then power then pins, each step with a 25mS delay. All sources are enabled.

Source Number	Source Enabled	Initial Delay
1	YES	0mS
2	YES	25mS
3	YES	50mS
4	YES	0mS
5	YES	0mS
6	YES	0mS

Signal	Assigned Source
SPECIAL1	Source 1
3V3_CHARGE, 5V_CHARGE, 12V_CHARGE	Source 2
3V3_POWER, 5V_POWER, 12V_POWER	Source 3

Hot-Swap State:
Drive is in the ‘pulled’ state, waiting for a “**RUN:POWer UP**” command to attach it.

Controlling the Module

The module can be controlled either by:

- Serial ASCII terminal (such as HyperTerminal)
  This is normally used with scripted commands to automate a series of tests. The commands are normally generated by a script or user code (PERL, TCL, C, C# or similar).
- Telnet Terminal (Only when connected to an Array Controller). This mode uses exactly the same commands as the serial ASCII terminal

Serial Command Set

When connected via a serial terminal, the module has a simple command line interface

SCPI Style Commands

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.

 # [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.

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

Sets the terminal response mode to the default 'User' setting. This is intended for use with HyperTerminal or similar and manually typed commands

CONFig:TERMinal SCRIPT

Sets the terminal response mode for easier parsing. Especially useful from a UNIX/LINUX based system. Characters sent from the PC are not echoed by the device and a <CR><LF> is sent after the cursor to force a flush of the USART buffer.

CONFig:TERMinal ?

Returns the current terminal mode

DEPRECATED COMMANDS – Provided for backwards compatibility, we strongly suggest you use the 'Signal' and 'Source' commands instead.

REGister:READ [0xAA]

Returns the value of the register with address [0xAA]. [0xAA] should be in hex format and preceded by the suffix "0x". e.g. "0x6D". The value is returned in the same form as the address.

REGister:DUMP [0xA1] [0xA2]

Returns the value of each register in a range, starting at the first register address, up to the second. [0xA1] and [0xA2] should be in hex format and preceded by the suffix "0x". Each data value will be returned on a new line.

REGister:WRITe [0xAA] [0xDD]

Writes the byte [0xDD] to register [0xAA], both [0xDD] and [0xAA] should be in hex format and preceded by the suffix "0x". The command returns "OK" or "FAIL".

MEASure:VOLTage [3v3in?|3v3out?|5vin?|5vout?|12vin?|12vout?]

Returns the voltage on the specified rail in mV. Vin refers to the upstream or host side of the card, and Vout refers to the switched, drive side. Values are returned in the form "3300mV".

M**EAS:VOLTage:SELF [5v?]**

Returns the self test voltages. These are measurements of voltage rails required for correct operation of the module. The values are returned in the form "5000mV"

SOURce:[1-6|ALL]:SETup [#1]

Sets up the source in a single command. All parameters are positive decimal numbers:

#1 = Initial delay (mS)
[Limits: 0 to 9999ms in steps of 1ms]

SOURce:[1-6|ALL]:DELAY [#ms]
SOURce:[1-6|ALL]:DELAY?

Sets the initial delay of a source in mS. The delay is entered as a decimal number with no units. E.g. “Source:1:delay 300”.

#1 = Initial delay (mS)
[Limits: 0 to 9999ms in steps of 1ms]

SOURce:[1-6|ALL]:STATE [ON|OFF]
SOURce:[1-6|ALL]:STATE?

Sets or returns the enable state of the source. Any signals assigned to a disabled (off) source will immediately be disconnected and vice versa. If a source state is changed, all signals assigned to it will change at exactly the same time (if a change is required). NOTE: If the ‘ALL’ selector is used, all 6 sources will be affected BUT there will be a very small delay between the change of each source. If timing is critical, assign all signals you want to change to the SAME source.

SIGnal:[SIG_NAME|ALL]:SETup [#num]
SIGnal:[SIG_NAME|ALL]:SOURce [#num]

Sets a given signal to a numbered timing source (0-8). SIGNAL_NAME is one of the signals/groups found in the ‘Signal Names’ appendix at the end of this manual.

SIGnal:[SIG_NAME]:SOURce?

Returns the source number that the signal is assigned to.

RUN:POWer [UP|DOWN]

Initiates a plug or pull operation (legacy name used to preserve compatibility between Torridon modules). This is done by changing the HOT_SWAP bit, register 0x00 bit 0. This is the master control for all switches on the card. The same action can be performed by writing this bit directly.

The command will fail if you order a power up when the module is already in the connected state and vice-versa as the action cannot be performed.

The “OK” response will be returned as soon as the hot-swap event has begun. If your timing sequence is very long you may have to poll the BUSY bit in register 0 to check when it has completed.

Control Register Map

This map is provided for backwards compatibility with old modules only. While you can use the 'Read' and 'Write' commands, we STRONGLY recommend you use the SCPI based 'Signal' and 'Source' commands instead of writing to the registers.

Address	Name	Description
0x00	Global Control	Initiate a hot swap event
0x01	Reserved	Reserved
0x02	Reserved	Reserved
0x03	Reserved	Reserved
0x04	S1 & S2 Control	Enable Sources 1 and 2
0x05	S3 & S4 Control	Enable Sources 3 and 4
0x06	S5 & S6 Control	Enable Sources 5 and 6
0x07	S1 Initial Delay	Set the source 1 delay
0x18	S2 Initial Delay	Set the source 2 delay
0x29	S3 Initial Delay	Set the source 3 delay
0x3A	S4 Initial Delay	Set the source 4 delay
0x4B	S5 Initial Delay	Set the source 5 delay
0x5C	S6 Initial Delay	Set the source 6 delay
0x6D	3V3_CHARGE,3V3_POWER	Signal control register
0x6E	5V_CHARGE,5V_POWER	Signal control register
0x6F	12V_CHARGE, 12V_POWER	Signal control register
0x70	SPECIAL1 (Mated)	Signal control register

Register Definitions

Control Register

0x00 - Global Control

7	6	5	4
RESERVED	RESERVED	RESERVED	RESERVED

3	2	1	0
RESERVED	RESERVED	RESERVED	HOT_SWAP

Name	Description
HOT_SWAP	Setting this bit initiates a hot plug sequence, and clearing it initiates a hot pull

Bit 0 controls the hot-swap state and can be read and written to. All other bits will read as '0'. When writing to the register the 7 upper bits should always be set to 0 (i.e. the only valid write values are 0x00 and 0x01)

Source Enable Registers

Source Enable [0x04, 0x05, 0x06]

7	6	5	4
0	0	0	Source Enable

3	2	1	0
0	0	0	Source Enable

Name	Description
Source Enable	Changing this alters the enable state of the source. All signals assigned to a disabled source will be disconnected and will not change with the HOT_SWAP state.

Bits 4 and 0 of these registers controls the enable state of the sources and can be read and written to. All other bits will read as '0'. When writing to the register the Remaining bits should always be set to 0. Bit 4 relates to the higher of the two sources in the description.

Source Registers

Each source is setup by a block of 9 16-bit wide registers. Below is the register map for a generic source. The list of registers in the title indicates the actual address of the byte in each of the 6 timed sources.

Source Delay [0x07, 0x18, 0x29, 0x3A, 0x4B, 0x5C]

7	6..0
DELAY_MULTIPLIER	DELAY

Name	Description
Sx_DELAY_MULTIPLIER	When 0, Delay Multiplier is 1mS When 1, Delay Multiplier is 10mS
Sx_DELAY	The Total delay after HOT_SWAP is set (by issuing RUN:POWer UP) until the signals attached to this source beginning to mate T_{DELAY} = xV_DELAY x xV_DELAY_MULTIPLIER i.e. 00000010 = 2mS, 10001001 = 90mS

Initial delay values range from 0 to 1270mS when set by writing the register value directly. However, if you use the "source:n:delay" command then values from 0 to 9999 are available.

Signal Registers

Two switched signals are controlled by each signal byte, one in each nibble. The 4 bits of the nibble stores a single number that describes which source the signal should be following.

Each nibble assigns the named signal to one of the six control sources, always off, always on, or on with HOT_SWAP:

Nibble Value	Assigned Control Source
0	OFF
1	Follow Source S1
2	Follow Source S2
3	Follow Source S3
4	Follow Source S4
5	Follow Source S5
6	Follow Source S6
7	On when HOT_SWAP is set
8	Always ON

0x6D – 3V3 & 3V3-Charge Control Source

7..4	3..0
3v3-Charge Control Source	3v3 Control Source

0x6E – 5V & 5V-Charge Control Source

7..4	3..0
5v-Charge Control Source	5v Control Source

0x6F – 12V & 12V-Charge Control Source

7..4	3..0
12V-Charge Control Source	12V Control Source

0x70 – SPECIAL1 (MATED) Control Source

7..4	3..0
Reserved	MATED Control Source

Appendix 1 - Signal Names

The following signal names are used to specify a single signal or a group of signals. These may be used in commands that take a parameter “SIGNAL_NAME”. Note that some commands, such as those returning a value, only accept a parameter that resolves to a single signal. In this case you cannot use the group names

Signals

3V3_POWER
3V3_CHARGE
5V_POWER
5V_CHARGE
12V_POWER
12V_CHARGE
SPECIAL1 (MATED on GND)

Signal Groups

ALL (Allows change of all signals at the same time)