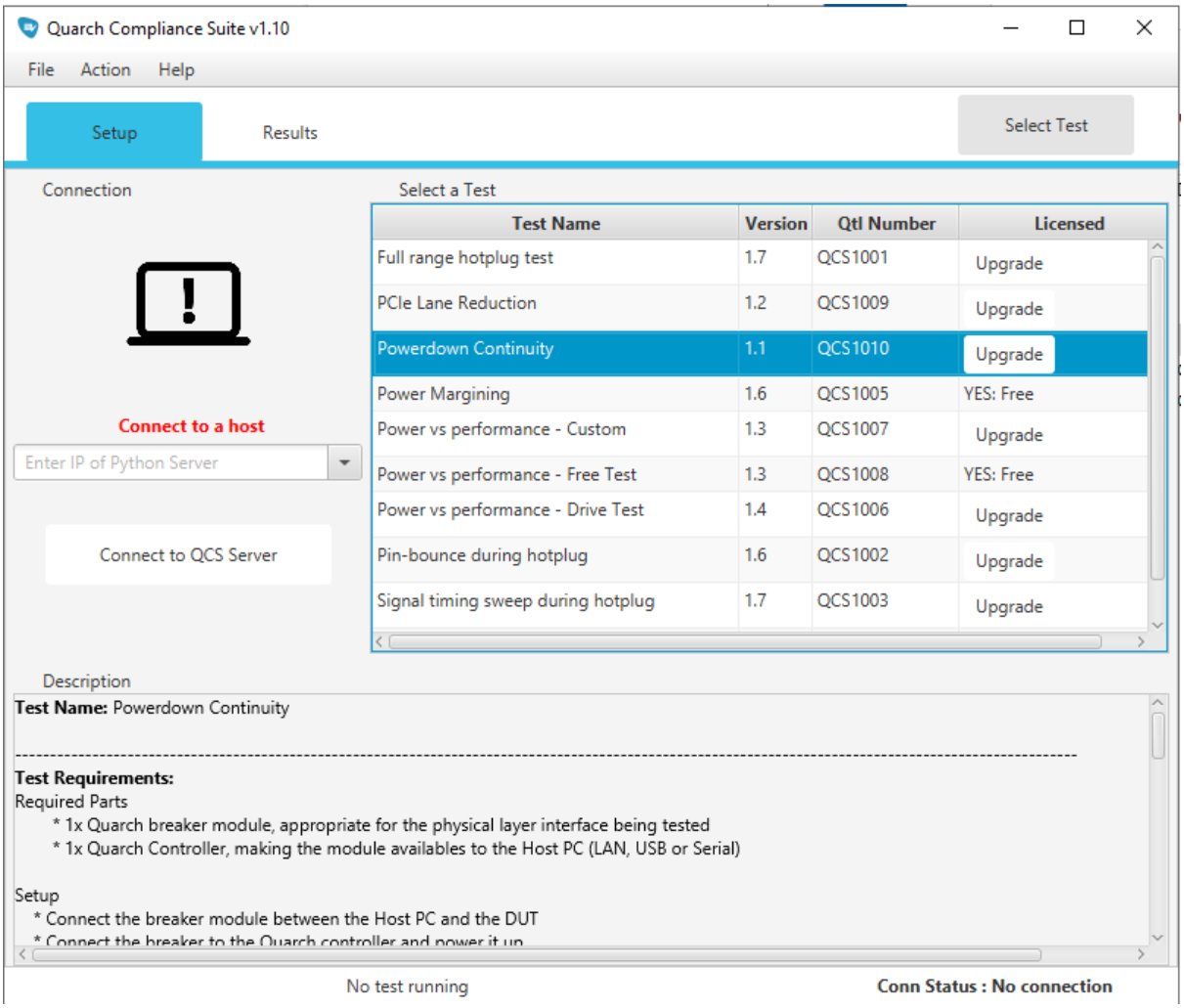
Quarch Technology Ltd

Quarch Compliance Suite

Setup & Test Specification

QCS1010 - Powerdown Continuity

V1.0



Change History

1.0		Initial Release

Contents

Introduction

Quarch's power down continuity test aims to verify FIO write workloads on a DUT after different power down scenarios.

With this test, we attempt to confirm a DUT has fully written all FIO data to storage. and that different types of power downs cannot cause errors in data that was said to be written.

Requirements

Host PC

- This is the PC which will mount the storage device under test (DUT). This system required admin privileges to install and execute the QCS server. Windows and Linux are supported.

Client PC

- This is the PC which will run the QCS client and record the results. This can be the same as the Host PC but it NOT recommended for this test. Windows and Linux are supported.

Quarch breaker module.

- A breaker module with the correct form factor and generation as the drive under test.

Device under Test (DUT)

- Your storage device. SAS, SATA and PCIe NVMe devices are supported.

QCS license

- This test requires a Quarch Compliance Suite license. The process for obtaining a paid license is documented here: https://quarch.com/support/faqs/qcs-licensing/

Installation

Initial installation and setup is described in the QCS ‘Quickstart’ guide. Please see this document if you are setting up for the first time. It will walk you through the QCS install process.

Setup

Host and Client PC setup is common across all QCS tests, so some of the steps below are only required the first time you prepare for testing.

- Setup the Host PC
    - As described in the QCS Quickstart document
    - Ensure the Host PC is connected to the LAN (assuming a separate Client PC is in use) Use of WIFI is highly discouraged for this testing.
- Setup the Client PC
    - As described in the QCS Quickstart document
    - Ensure the Client PC is connected to the LAN (assuming a separate Host PC is in use) Use of WIFI is highly discouraged for this testing.
- Setup the Quarch Breaker module
    - Place the Quarch breaker between the DUT and host slot. Connect the ribbon cable to the Quarch Interface kit or Quarch array module. Connect this to the Host PC.
    - Power on the Quarch Module

      Choose either USB or Serial for the Quarch Breaker Module. The Module must be accessible to the **Host** PC, so be sure you cable to the right one if using USB.

      If you need to configure the breaker module settings, you can do so via TorridonTerminal: https://quarch.com/file/torridon-terminal/

      This is also helpful to ensure you have connected the module correctly and can see it. If you cannot see the module and wish to check it is turned on, use TorridonTerminal to issue the command:

      ```
      > run:power up
      ```

Typical equipment layout

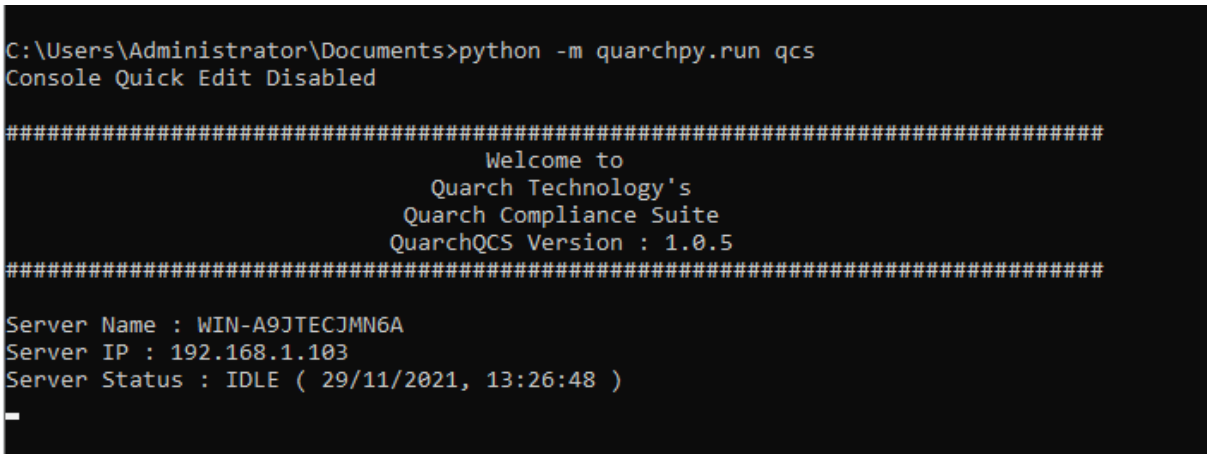
A typical equipment layout is shown below.

Begin the test

- Start QCS server of the Host PC

- > python -m quarchpy.run qcs

The server should start up almost immediately. Note the IP address and mDNS name which you will use to connect to the server later.

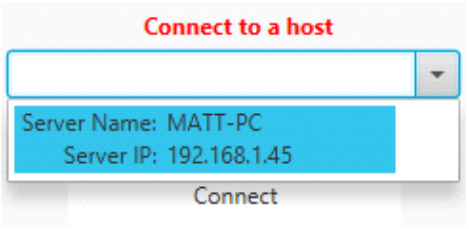


- Start QCS application on the Client PC

- > Double click on the QCS icon to start

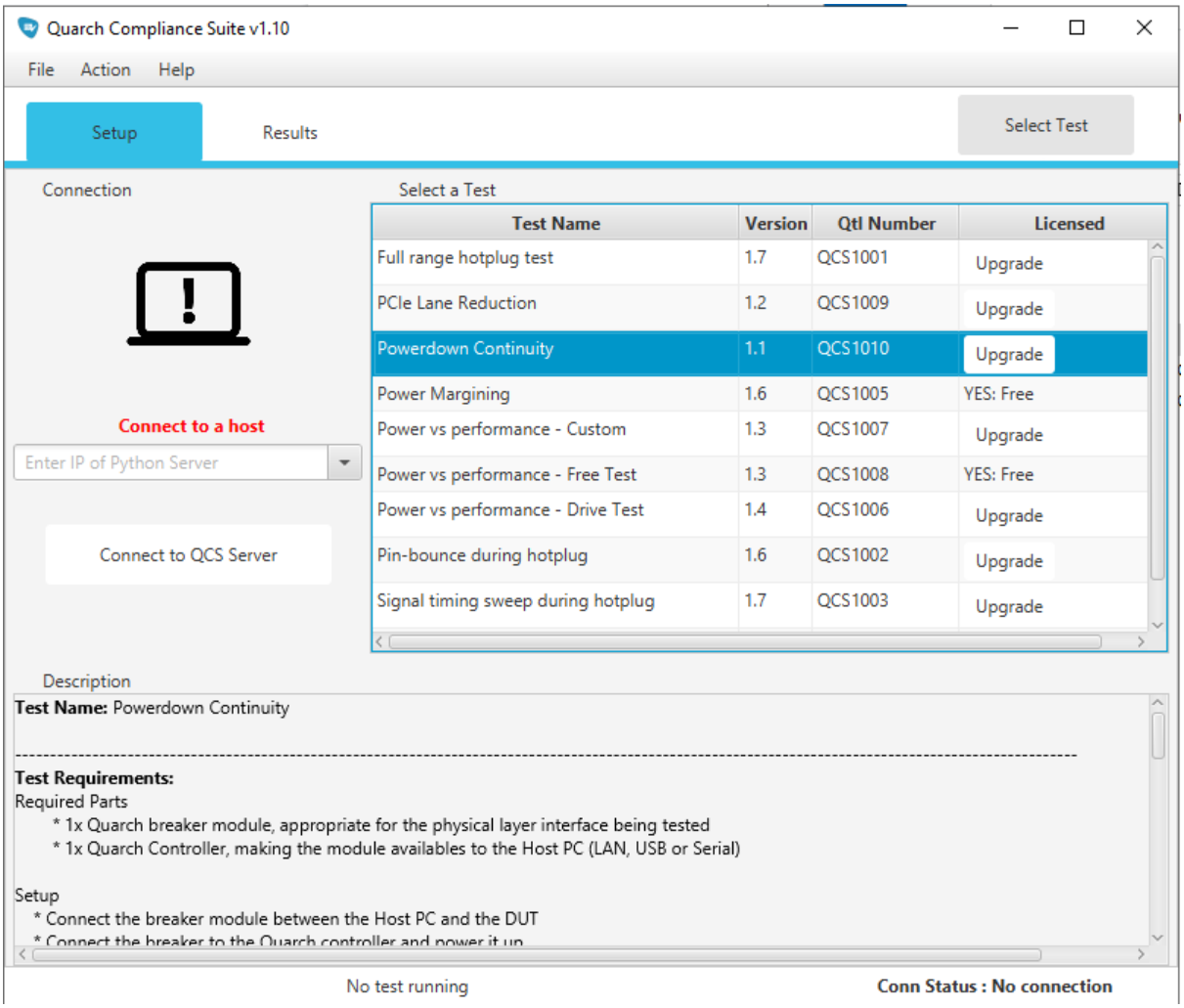
- Connect the Client to the Server

- Enter the IP address of the Server into the Client connection box. If ‘zeroconf’ is installed on the server then it should auto detect and be visible in the connection drop-down menu for fast connection.



- Select the test to run

- In this case, select the QCS1010, ‘Powerdown Continuity’ and either double-click or select ‘Select Test’



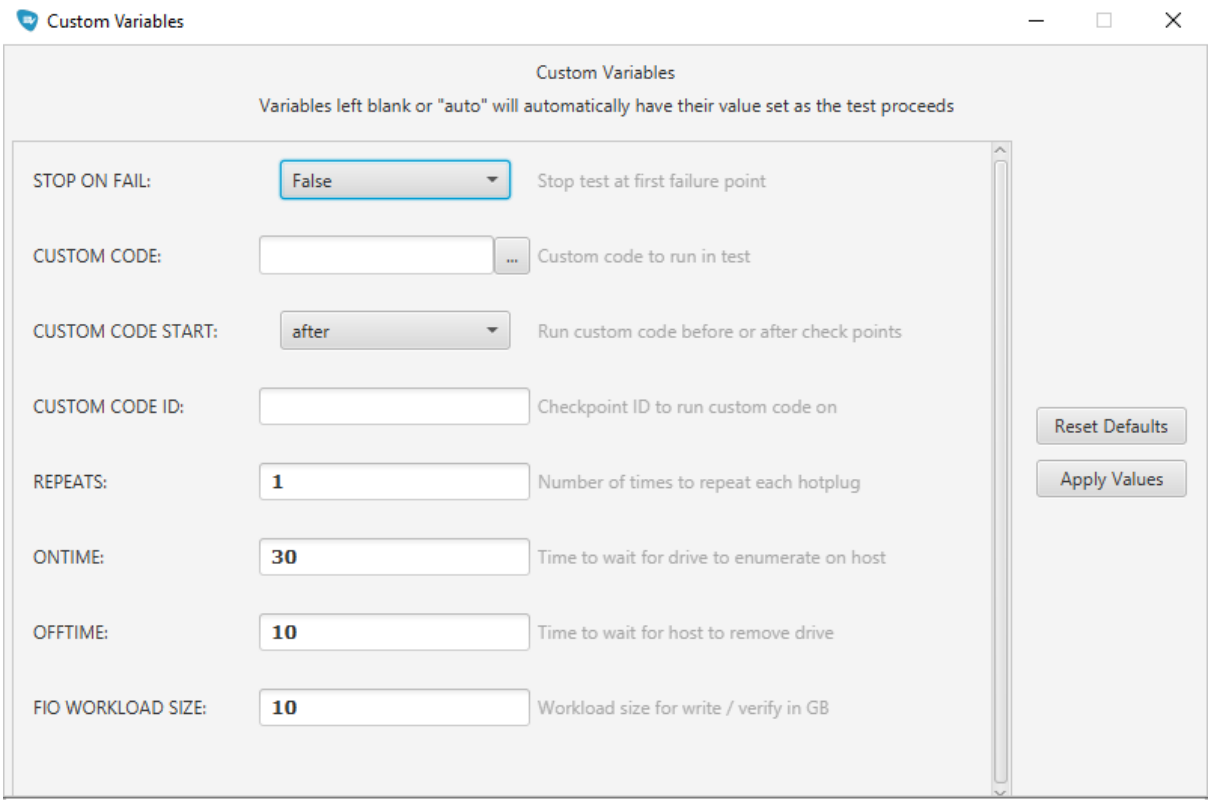
- When the test is ready, the ‘Start/Play’ button will become available to begin running. If you want to change any setting for the test, you can do so now in the ‘Custom Variables’ window.

- When you have made any changes you require, press the ‘start tests’ button to begin.

Custom variables

Most tests have several variables that can be set. These allow for things such as setting the number of times that a test loops or setting the time to wait for a drive to enumerate. These will be different in each test suite.

QCS1010 has several useful settings to consider:



Custom Code

- Custom python code to be executed during test

Custom code start

- Run code before or after the check points

Custom code ID

- ID's of check points to run custom code on
- Acceptable formats:
  - **'1-3'** (*Any checkpoints with unique ID between 1 and 3 - 1.1.1 e.t.c)*
  - **'1.2.1'** (*Specific unique ID*)
  - **'1.x.3'** (*Any check point with a unique ID that follows this format*)
- Formats can be combined if separated by a ',' character.
  - e.g. 1-2**,**1.3**,**1.x.4

Stop on fail

- When set to true, the test will halt if there are any errors or test failures

Repeats

- Amount of times to repeat each section of this test

OnTime

- Time to wait for drive to be discovered on system after a hotplug

OffTime

- Time to wait for host to remove drive

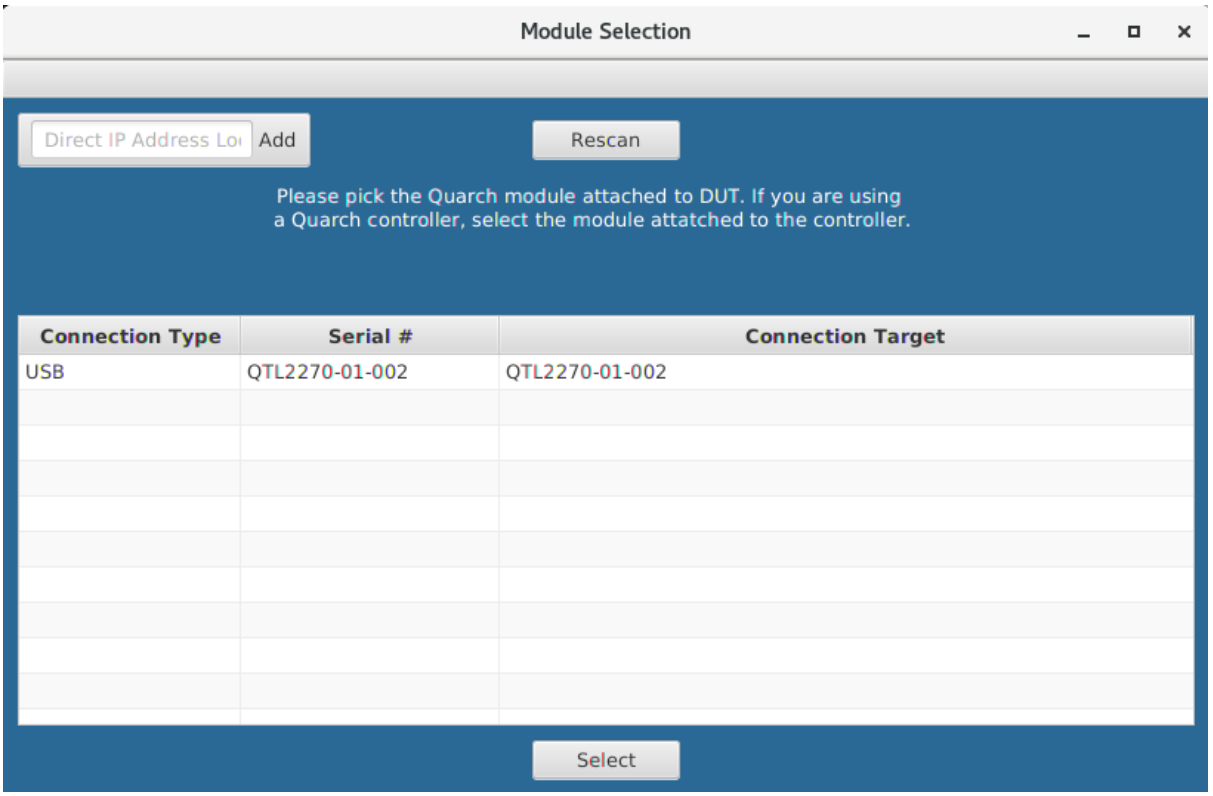
Select the devices

As the test starts running, you will be prompted to select the Quarch Breaker Module and storage device to use. The dialogs have a rescan button: just in case you have forgotten to plug something in!

Select the breaker module

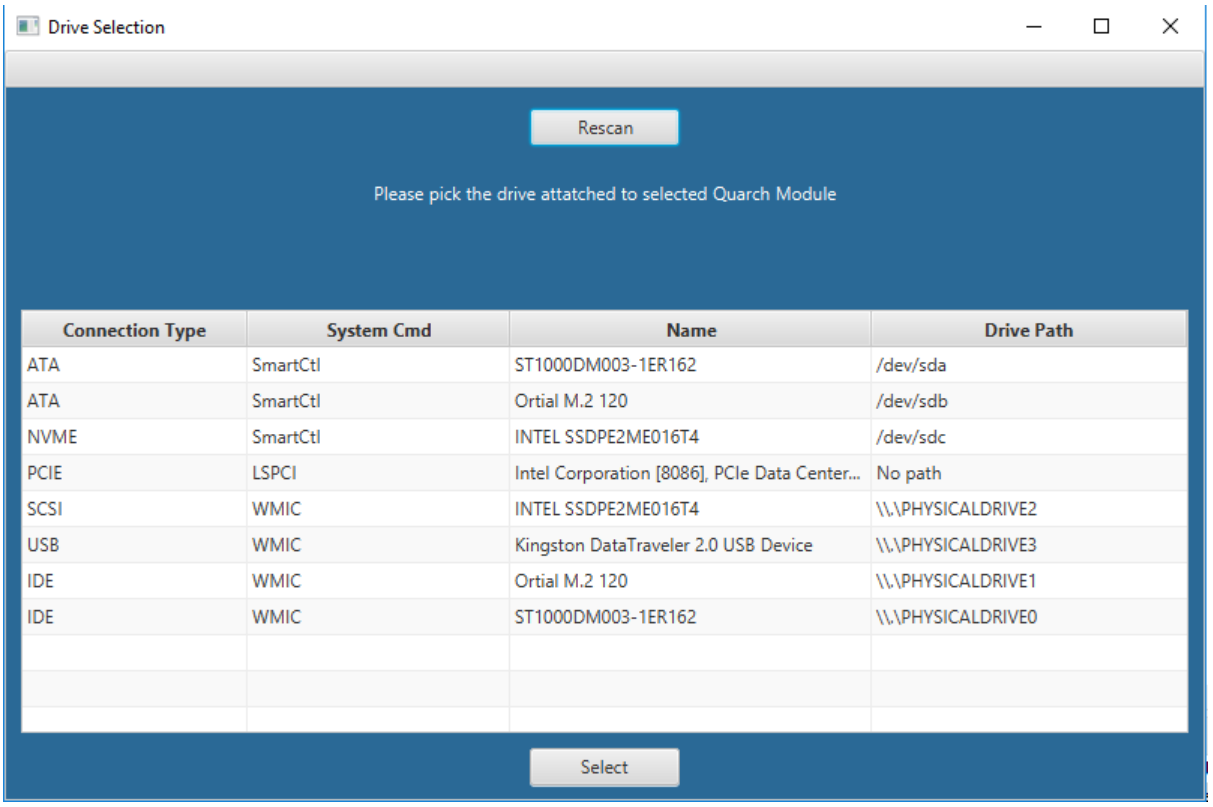
First you will be prompted to select the Breaker Module. Here we have a QTL2270 (Gen4 PCIe U.3) module connected to the Client PC via USB.

The breaker module must be available to the **Host** PC. If this is not the case, you will not see it shown here.



Select the storage device (DUT)

Next you will be prompted to select the storage device to test.



Be sure that you select the correct DUT.
Selecting the wrong drive will show fail test points throughout half of the test as the drive being monitored in QCS is not the drive being hotplugged.

View test progress

This test suite with default variables is fast to complete, currently under 5 minutes depending on how fast the drive under test enumerates. Customizing the test variables will increase this total test time.

Real time results are displayed in the log. Any failures will be clearly marked in red, and noted on the error counters.

If many failures occur early on, it is likely that you have a problem with the setup of the equipment. Expanding the failed rows gives more information on the failure.

Common test failures

- **Command to a Quarch module fails to get a response**.
- Module has become disconnected or powered down. Check your cabling.
- **Command to a Quarch module returns a bad command or invalid parameter error.**
- The Module connected is not the correct one for the test, or it required an upgrade.
- **The DUT is not removed from the system correctly during the power cycle test.**
- The wrong DUT may be selected. Make sure you have chosen the right device. If uncertain, you can use TestMonkey or TorridonTerminal to manually hot swap the DUT and verify that it is powering the correct drive.
- **The DUT is not enumerated on the system after each hotplug.**

- Does the DUT and Host system support hot plug? If you cannot manually hot plug a device in the Host and have it enumerated correctly, then the test will not work.

Test specification

Basis for testing.

Your comments, requests and suggestions are very welcome and can be directed to support@quarch.com.

Future versions of the test suite will aim to include these improvements where practical.

Assumptions

We assume that the drive connected is in ‘optimal’ condition, to begin testing (ie: that it is empty and in ‘out the box’ condition).

We assume the setup is correct with DUT and Host being capable of hotplug.

We assume PC’s are connected across LAN.

Test itinerary

- Power down scenarios tested
    - 1 - Basic power off - Switching off power to all signals.
    - 2 - Removing power to signal : **Perst** (If applicable)
    - 3 - Removing power to signal : **Ref_clk** (If applicable)
    - 4 - Removing power to signal : **Pwr_Dis** (If applicable)
    - 5 - Removing power signals only
- Power down scenarios are started within 1 second of FIO job finishing.
    - This is the time it takes for the FIO process to end and QCS to send a command to the module initiating the power down scenario
- Scenarios 2-5 are run for a time period of 5 seconds.
- Each power down scenario is repeated once by default
- **After each scenario, the DUT is hotplugged and the quarch module reset.**
- Every repeat checks for both drive removal and insertion for the system & link speed, lane width if the drive is of type NVMe PCIe.
- By default, the DUT is expected to enumerate within 15 seconds of the power up command. In rare occasions where this is exceeded, this variable can be changed inside of the custom variables.

Test steps

1. **Check drive powered up**

   Ensure drive under test is powered on and discovered by the system.

2. **Run write workload to the DUT**

   'fio --rw=randwrite --verify=pattern --verify_pattern=0x{current time}

   --do_verify=0 --size={user variable(default 10)}g --filename={drive chosen}

   --name=job1'

3. **Run power down scenario**.

4. **Power cycle DUT**

   Test automatically runs power down and power up commands to the module.

5. **Check drive is enumerated on the system**

   Poll system commands to check for drive enumeration on system

6. **Run verification FIO workload**

   fio --rw=randread --verify=pattern --verify_pattern=0x{taken from write}

   --do_verify=1 --size={user variable(def 10)}g --filename={drive chosen}

   --name=job1'

7. **Check for any errors reported from FIO**

Test completion

1. **Generate Test Report**

   Under the “Action” tab, there is an option to auto generate a test report upon completion of the test.

Passing a pull event.

After sending a “run power down” to the Quarch module, the test will consistently query the system command used to find the drive. A drive passes this check point if it is successfully removed from the list of drives returned from the system command.

Passing a plug event.

After sending a “run power up” to the Quarch module, the test will consistently query the system command used to find the drive. A drive passes this check point if it is successfully added to the list of drives returned from the system command.

Passing a link speed check – NVMe / PCIe devices.

After the DUT is powered up and discovered, the test queries the LSPCI command using “lspci –vvv”, giving a very verbose output of the drive capabilities. Found within these capabilities is its current link speed. A drive passes this check point if the link speed is consistent as to what was expected. By default – This is the link speed a drive has at the beginning of the test.

Passing a lane width check – NVMe / PCIe devices.

After the DUT is powered up and discovered, the test queries the LSPCI command using “lspci –vvv”, giving a very verbose output of the drive capabilities. Found within these capabilities is its current lanewidth. A drive passes this check point if the lane width is consistent as to what was expected. By default – This is the lane width a drive has at the beginning of the test.