Quarch Technology Ltd

Hardware Triggering Application Note

# 1. Change History

|  |  |  |
| --- | --- | --- |
| 1.0 |  | Initial Release |
|  |  |  |

2. Contents

[Quarch Technology Ltd 1](#_Toc184889752)

[Hardware Triggering Application Note 1](#_Toc184889753)

[1. Change History 2](#_Toc184889754)

[3. Table of Figures 4](#_Toc184889755)

[4. Introduction 5](#_Toc184889756)

[5. Breaker Applications 6](#_Toc184889757)

[5.1 SMBus into External Analyser 6](#_Toc184889758)

[6. Power Applications 11](#_Toc184889759)

[6.1 12V Power Rail Pulled to Ground Upon PERST# Assert 11](#_Toc184889760)

[6.2 Power Delay Upon Power Up 15](#_Toc184889761)

[6.3 Triggering on Host Power Up 16](#_Toc184889762)

[7. General Trigger Commands 19](#_Toc184889763)

# 3. Table of Figures

[**Figure 1. Triggering Flow Chart** 5](#_Toc184889806)

[**Figure 2. SMBus Decoding Application Hardware Setup** 6](#_Toc184889807)

[**Figure 3. HOST PC Setup, Quarch Module with Triggering cable and M.2 drive with adapter** 7](#_Toc184889808)

[**Figure 4. PicoScope Connected to the Quarch module’s Triggering ports** 7](#_Toc184889809)

[**Figure 5. Commands used in this example** 9](#_Toc184889810)

[**Figure 6. A whole session of SMBus communication decoded by the PicoScope software** 9](#_Toc184889811)

[**Figure 7. The Signature Signal Form of the Start of the Session** 10](#_Toc184889812)

[**Figure 8. Setup with U.2 Breaker + Power Injection Fixture** 11](#_Toc184889813)

[**Figure 9. Flow Chart of a Setup** 12](#_Toc184889814)

[**Figure 10. Setup with MCIO to U.2 Breaker + MCIO Host Card** 12](#_Toc184889815)

[**Figure 11. Flow Chart of Setup** 13](#_Toc184889816)

[**Figure 12. Power Delay Upon Power Up Flow Chart** 15](#_Toc184889817)

[**Figure 13. Requirements** 17](#_Toc184889818)

[**Figure 14. On Host Power Up Flow Chart** 17](#_Toc184889819)

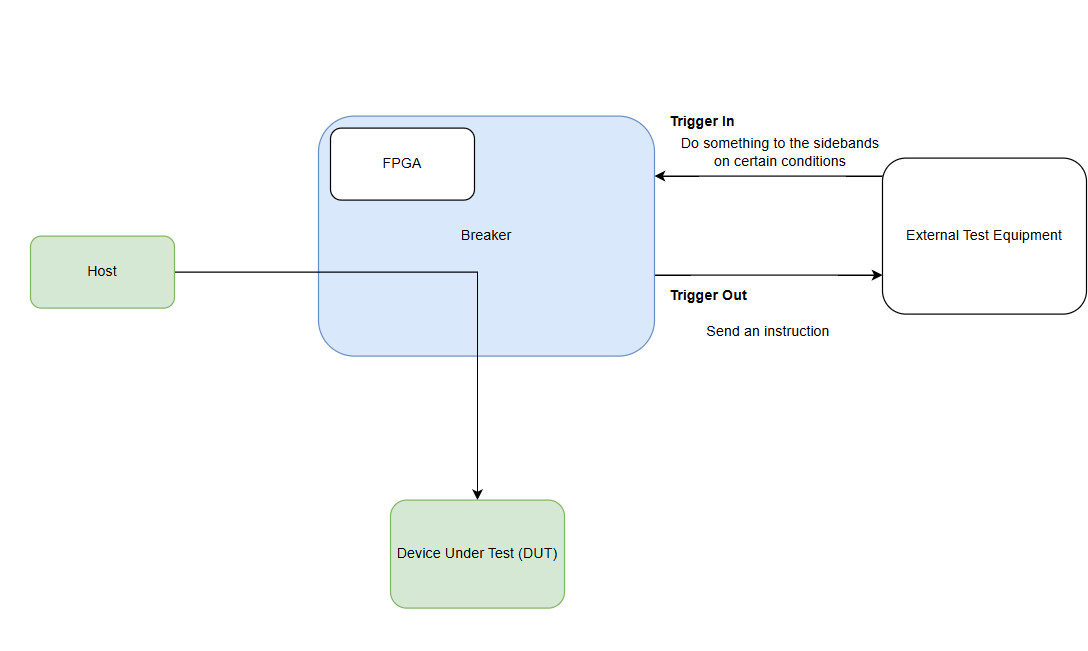
[**Figure 15. On Host Power up Setup** 18](#_Toc184889820)

# 4. Introduction

This application note is the combination of various triggering examples to demonstrate how the triggering functionality can be used in supported Quarch products.

Trigger is the capability to synchronise the module operations with external test equipment through dedicated trigger input (IN) and output (OUT). Facilitate coordinated testing scenarios by allowing external devices to initiate or respond to specific events managed by the module.

Trigger IN: enables for external equipment to control the module's actions.  
Trigger OUT: enables the breaker module to send signals to external equipment, indicating the occurrence of specific events.



**Figure 1. Triggering Flow Chart**

## 5. Breaker Applications

### 5.1 SMBus into External Analyser

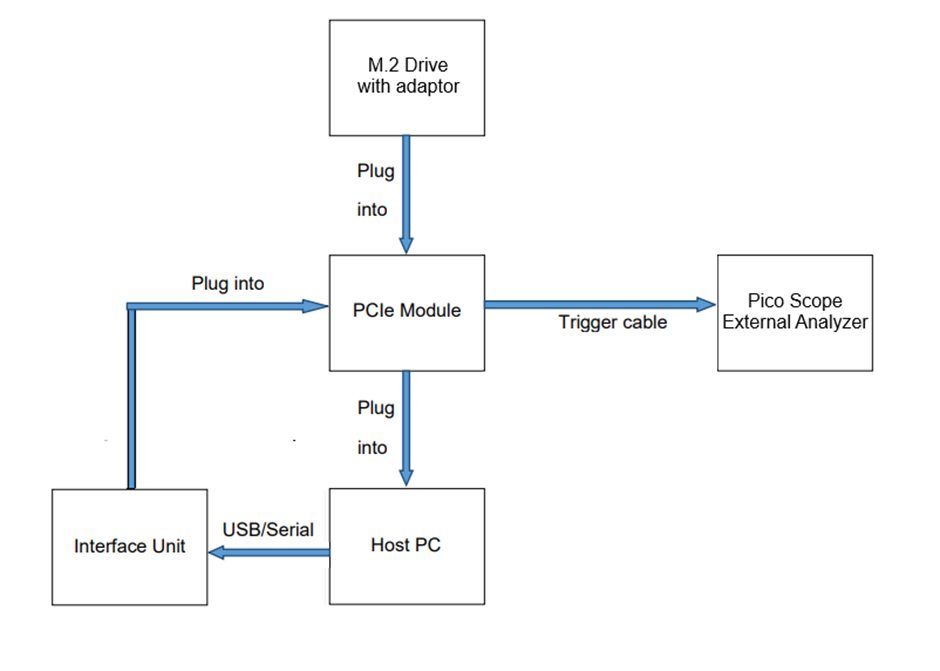
Introduction

Firmware version 4.005 has added the functionality to provide an internal and external trigger on different conditions.  
This application note will focus on showing how to cleverly use the triggering features to route and decode the SMclk and SMdat side bus information with an external analyzer.   
Since the trigger is initiated by the side signals, the same PC can be used to control the modules.

Requirements

1. Quarch PCIe Module which supports Triggering:
2. Quarch Interface Unit
3. Host PC with support of sideband communication (with drivers installed if using USB control)
4. An appropriate script or terminal able to connect and issue commands
5. External Analyzer

Set-up



**Figure 2. SMBus Decoding Application Hardware Setup**

In this example the modules below are used:

* QTL2798 – Gen5 PCIe x16 Breaker Module
* QTL1260 – Torridon Interface Kit
* M.2 driver with an adapter to PCIe
* PC able to communicate on SMBus



**Figure 3. HOST PC Setup, Quarch Module with Triggering cable and M.2 drive with adapter**

* Picoscope 3204A as an external analyser



**Figure 4. PicoScope Connected to the Quarch module’s Triggering ports**

Control Options

* Torridon Terminal can send commands to a module [Torridon Terminal - Quarch Technology](http://quarch.com/file/torridon-terminal)
* TestMonkey provides GUI control over a module [TestMonkey GUI - Quarch Technology](http://quarch.com/file/testmonkey)
* Python can be used to automate control of Quarch products

Basic Concept of Live monitoring

This feature is supported on ‘Triggering’ modules only.  Both the trigger IN and OUT ports can be used to monitor a signal.

*WARNING: As the trigger IN port can be ordered to OUTPUT a status, there is a risk of two devices driving against each other and causing damage.  Before using the live monitoring feature, you must ensure that you do not have any equipment attached that may try to drive the trigger IN port.*

 To begin live monitoring, first enable the trigger ports you want to use.  This is done via additional options to the existing trigger setup commands:

Trigger OUT port setup

# Set the trigger mode to sideband monitor  
**TRIGger:OUT:MODE:SIDEband**

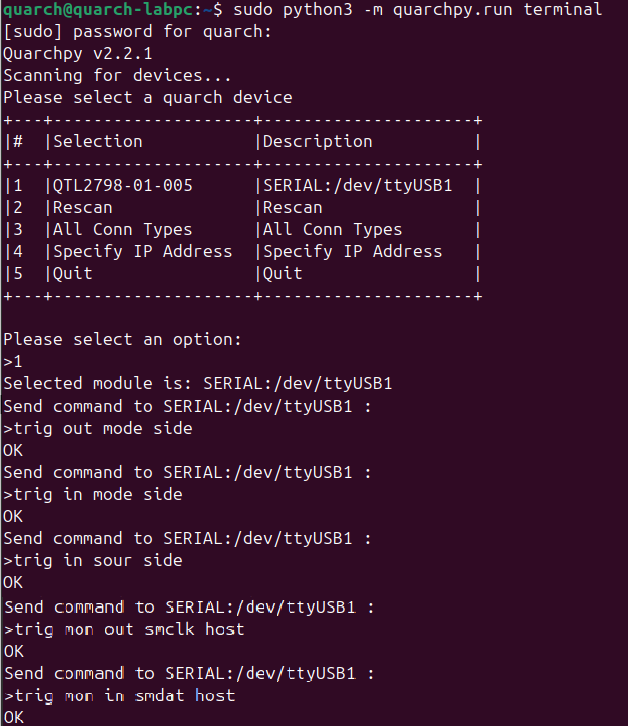
Trigger IN port setup(requires double verification)

# Set the trigger mode to sideband monitor  
**TRIGger:IN:MODE:SIDEband**  
# Also set the trigger IN source to sideband out  
**TRIGger:IN:SOURCE:SIDEband**

Monitoring setup

# Select a signal for live monitoring  
**TRIGger:MONitor[IN|OUT]:[SIGNAL-NAME]:[HOST|DEVice]**

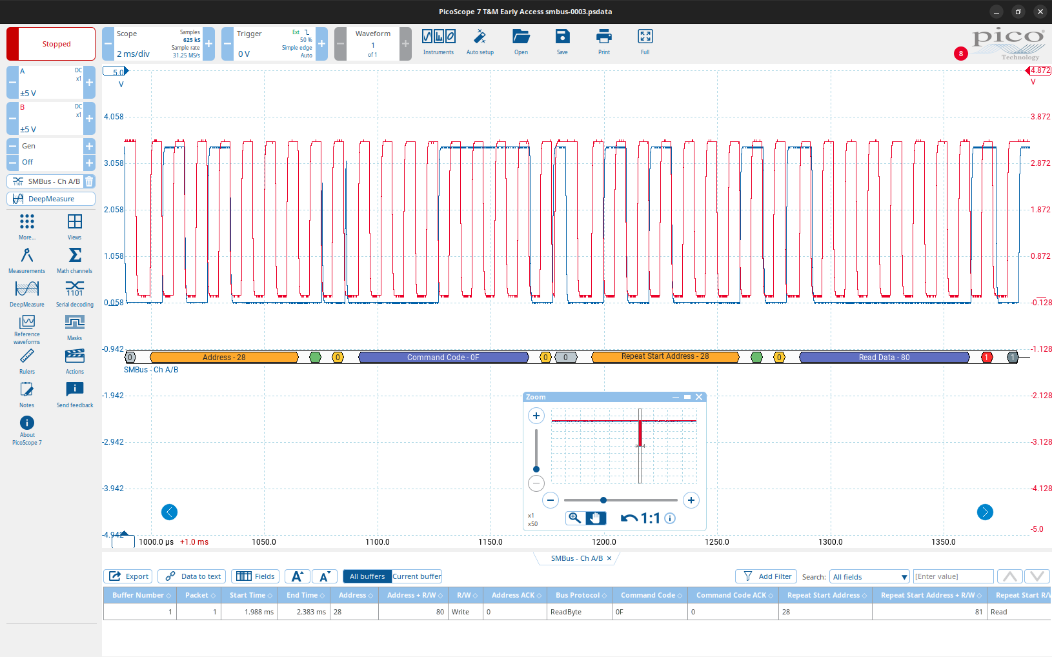
Sets a trigger port to activate live monitoring for a given signal.  The host/device parameter selects the side of the module to monitor on.



**Figure 5. Commands used in this example**

Results

Different types of information can be extracted from the captured signals. SMClk is a 100 kHz clock signal, and SMDat is decoded according to this clock signal. The protocol structure reveals the section start, section end, acknowledgment, write/read operations, as well as the addresses and data.



**Figure 6. A whole session of SMBus communication decoded by the PicoScope software**



**Figure 7. The Signature Signal Form of the Start of the Session**

## 6. Power Applications

### 6.1 12V Power Rail Pulled to Ground Upon PERST# Assert

Introduction

This setup will allow when reset is initiated (PERST#) then 12V will be pulled to GND. All the other signals will remain active except of the PERST#.

PERST# - PE-Reset signal is a fundamental reset to the device.

*Firmware version needed: 6.010*

*FPGA version needed: 1.7*

Warning: Using older firmware + FPGA version, need to use pattern + record command for triggering to work, otherwise it will not.

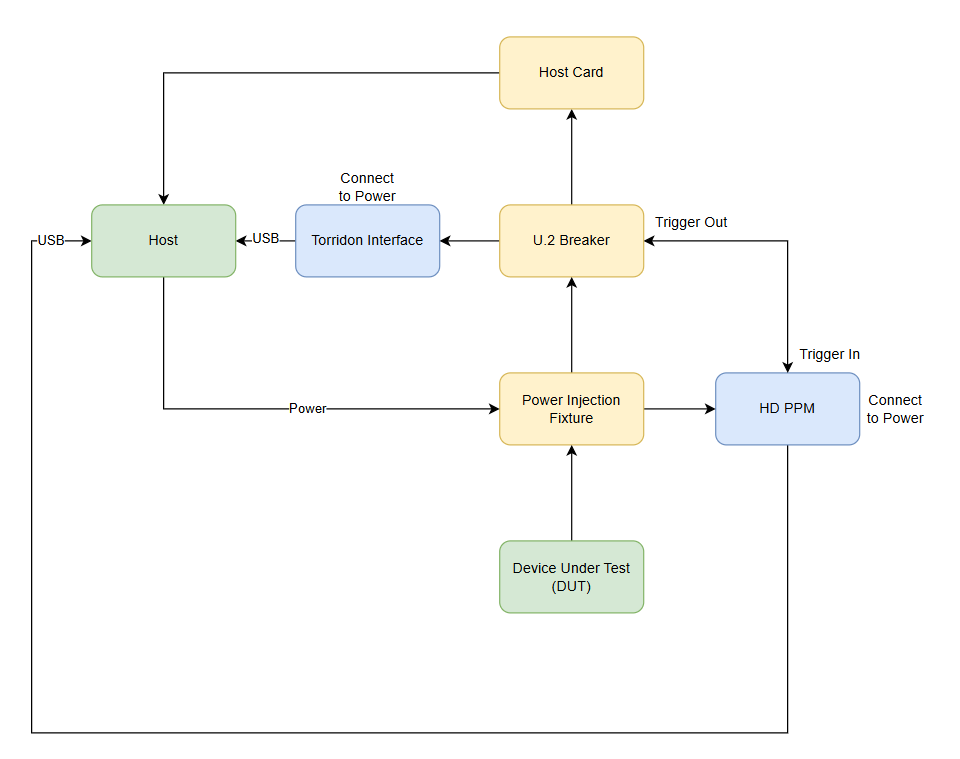
Setup

Variation 1 Requirements:

* HD PPM (QTL1999)
* QTL2266 (U.2) or QTL2270 (U.3) Breaker
* QTL2289 SFF Drive Power Injection Fixture
* Quarch Controller (QTL1079, QTL1260 or QTL1461)
* Male to Male MCX cable (Triggering Cable)
* Host Controller



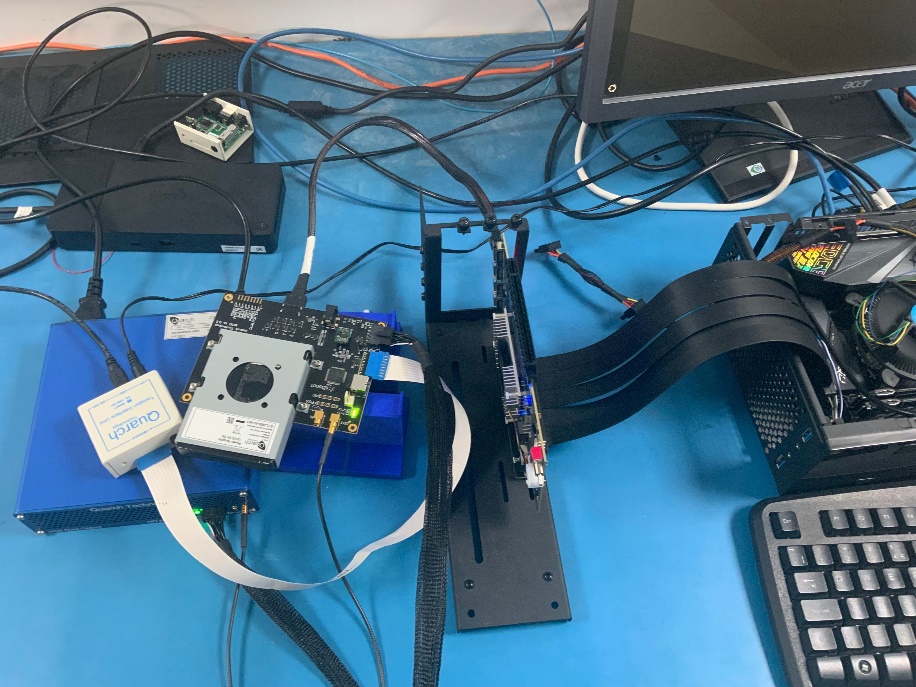
**Figure 8. Setup with U.2 Breaker + Power Injection Fixture**



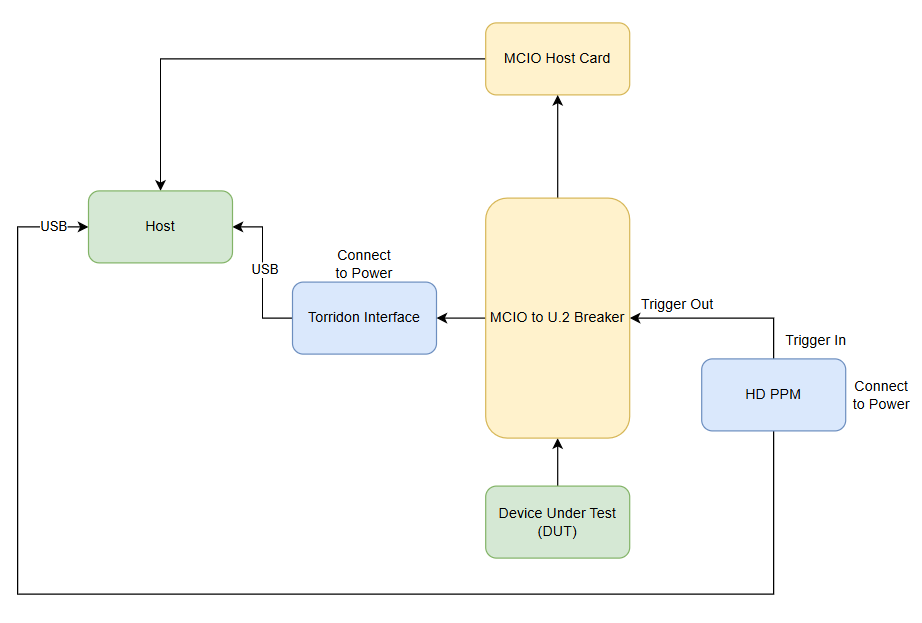
**Figure 9. Flow Chart of a Setup**

Variation 2 Requirements

* HD PPM (QTL1999)
* U.2 to MCIO Breaker (QTL2805)
* Quarch Controller (QTL1079, QTL1260 or QTL1461)
* Male to Male MCX cable (Triggering Cable)
* MCIO Host Controller



**Figure 10. Setup with MCIO to U.2 Breaker + MCIO Host Card**



**Figure 11. Flow Chart of Setup**

**Commands for this application**

The attached script executes the provided code below. After running the script, perform the following:

RUN: POWER DOWN on the breaker. PPM’s 12V line will be pulled to GND after triggering.

On PPM

RUN:POWER UP -- Turn on the power

CONFIG:OUTPUT:12V:PULLDOWN ON -- Enables pull down

SIGNAL:12V:PATTERN:ADD 0S -12000 -- Adds the pattern, when the voltage goes from 12V to 0V the delay of the drop is 0 seconds.

PATTERN:TRIGGER:EXTERNAL ON -- Pattern above will be only triggered externally

PATTERN:TRIGGER:EXTERNAL:TYPE:EDGE -- Sets trigger in to trigger on an edge

On Breaker Module

SIGNAL:ALL:SOURCE 8 -- Sets all signals to source 8 (always on)

SIGNAL:PERST:SOURCE 4 -- Sets PERST# to source 4

SIGNAL:PERST:DRIVE:OPEN LOW -- When PERST is discounted module will drive device side LOW

TRIGGER:OUT:INVERT ON -- Need to invert so it triggers on power down

TRIGGER:OUT:MODE POWER -- Trigger out on power event

To run the test command to the Breaker Module:

RUN:POWER DOWN

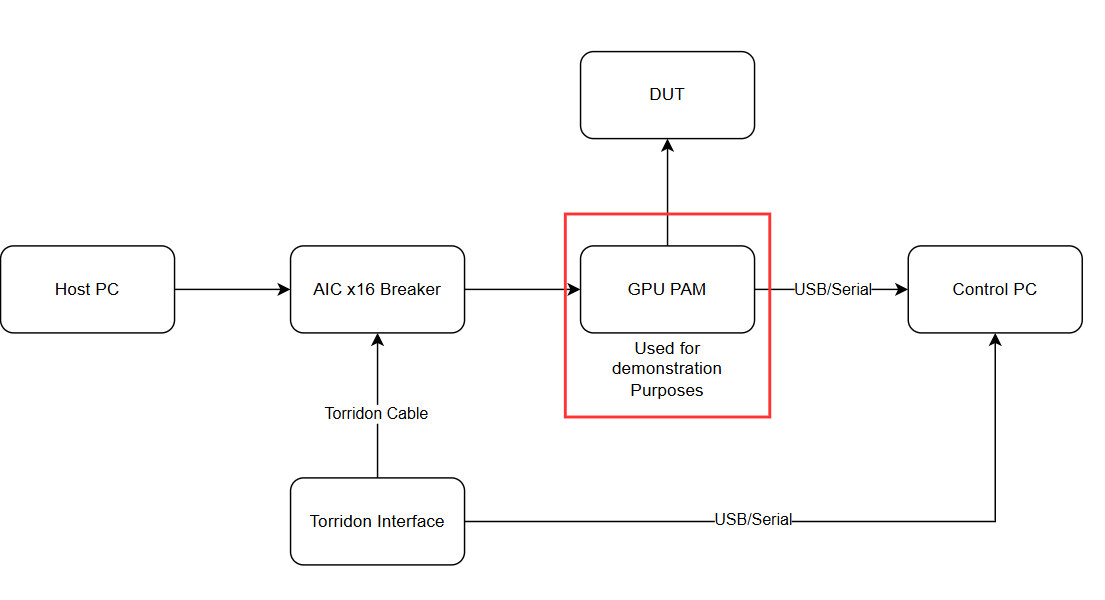
### 6.2 Power Delay Upon Power Up

This setup is a specific triggering scenario, where 12V is delayed from 3V3. Delay can be configured in the script.

**Setup**

Requirements:

* AIC x16 Breaker
* GPU PAM (optional, only if tracking is needed)
* Torridon Interface
* M.2 SSD
* Host and Control PC



**Figure 12. Power Delay Upon Power Up Flow Chart**

### 6.3 Triggering on Host Power Up

Introduction

Firmware version 4.005 has added the functionality to provide an internal and external trigger on host power up.

The external events are triggered by producing an output trigger to another device, either a Quarch device that supports triggering (for example a Power Margining Module) or a third-party piece of equipment that allows triggering input.

This example will focus on using a Quarch Power Margining Module to run a power pattern when the host powers up.

Since the trigger is initiated by the Host powering, a separate PC is used to control the modules.

Requirements

1. Quarch PCIe Module which supports triggering on host power up:

QTL1688(-04 and higher)

1. Quarch Power Margining Module which supports triggering

QTL1999(-01 and higher)

QTL1995(-01 and higher)

QTL1847(-01 and higher)

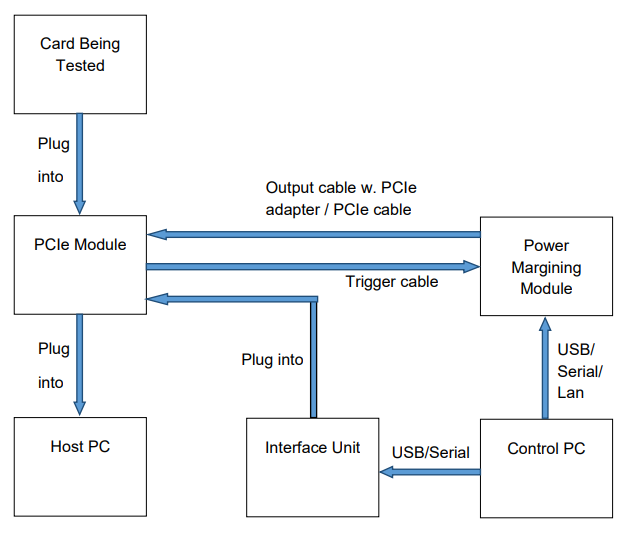
1. Quarch Interface Unit
2. Host PC
3. Control PC (with drivers installed if using USB control)
4. An appropriate script or terminal able to connect and issue commands

A group of electronic components

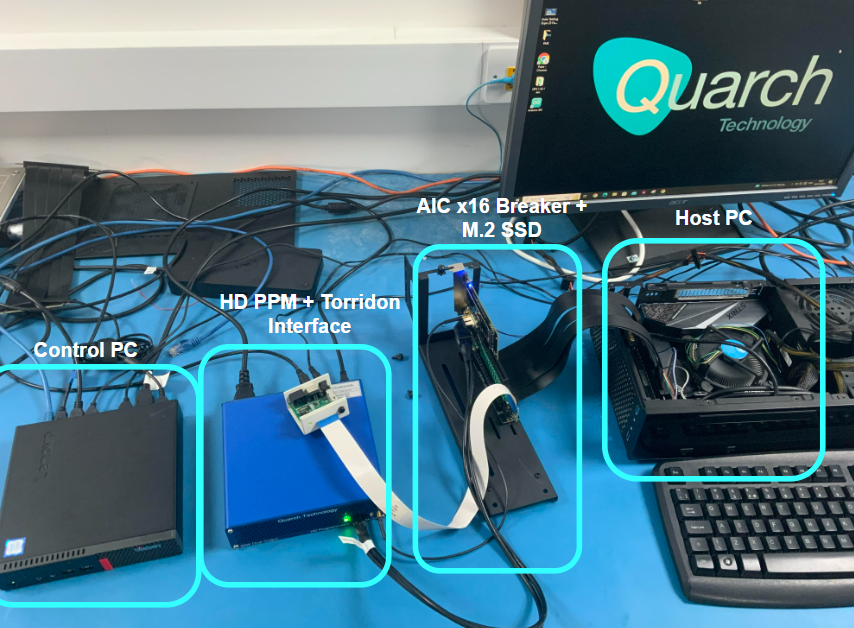
Description automatically generated

**Figure 13. Requirements**

Set-up



**Figure 14. On Host Power Up Flow Chart**



**Figure 15. On Host Power up Setup**

The Basics

The module can be configured to perform a number of actions when the Host is powered up:

External: The module sends out a pulse along the trigger out port, if a cable is connected to this port the signal will travel to the device it is connected to and trigger an event on it according to that device’s configuration

Internal: The following can be triggered on the module:

* Hot Swap
* Glitch

The module can listen to either of the hosts power lines:

* 12V
* 3V3

For the example given an external trigger will occur on 12V host power up.

# 7. General Trigger Commands

TRIGGer:IN:TYPE [EDGE | LEVEL]

Sets the trigger type

EDGE = Actions will start on the asserted edge and complete in full

LEVEL = Actions will run for as long as the trigger signal is asserted

TRIGger:IN:INVERT [ON|OFF]

Sets the trigger invert mode for input triggers

OFF = Trigger acts as normal ON = Trigger responds to an inverted input

TRIGger:OUT:INVERT [ON|OFF] Sets the trigger invert mode for output trigger

OFF = Trigger acts as normal

ON = Trigger outputs in an inverted form

TRIGger:IN:SOURce [EXTernal|???\_host]

Sets the source of the trigger in event EXTernal = Uses the trigger in connector

???\_host = Uses the output of the host power detect system.

??? is the voltage channel, generally 3v3\_host or 12v\_host, though the channel selection options will vary between modules and is not supported on all devices.

TRIGger:IN:MODE [OFF|POWER|GLITCH|SIDEband]

Sets the action to perform on a trigger in event

OFF = No action (default mode)

POWER = Power cycle will be performed

GLITCH = Glitch action will be performed

SIDEband= Sideband monitor mode (if ‘monitoring’ is supported)

TRIGger:OUT:MODE [OFF | POWER | GLITCH | ???\_host | SIDEband]

Sets the action to perform on a trigger out event

TRIGger:MONitor:IN [#SignalName]

TRIGger:MONitor:IN?

TRIGger:MONitor:OUT [#SignalName] [HOST|DEVice]

TRIGger:MONitor:OUT?