Quarch Technology

Product Guide

Power Analysis Modules

Advanced DC power analysis

SETUP IN SECONDS AND CAPTURE FOR HOURS

POWER ANALYSIS HAS NEVER BEEN EASIER

CONTENTS

Product Range Overview

BENEFITS

The Quarch power analysis range uses plug-and-play fixturing for fast setup. There is little need for custom cabling, current clamps or probes.

High resolution data can be captured for extended periods of time, giving you an order of magnitude more data than a traditional scope. Capture multiple power rails and digital sidebands to aid debugging.

Capture can be done in a couple of clicks, or via our fully automated Python API.

Analysis can be done in Quarch Power Studio or as an automated post-processing step on exported CSV data.

Our powerful viewing software provides a clear overview of large traces and zooming to the smallest detail.

Quarch tools are widely trusted across the industry and referred to in several industry standards which require accurate power Vs performance metrics.

POWER ANALYSIS MODULE (PAM)

Find out more

USB 2.0 and 100Mb Ethernet Connectivity, 12V External PSU, 3.3v MCX Trigger In and Trigger Out, Fully Isolated from Fixture

DC power analysis

DC APPLICATIONS REQUIRE A POWER ANALYSIS MODULE AND A POWER ANALYSIS FIXTURE

Capture analog and digital signals over long time periods

GEN5 SFF PAM FIXTURE (SAS/SATA/U.2/U.3)

Find out more

SFF PAM: ANALOG CHANNELS: 12V, 5V, 3.3VAUX, SAMPLE RATE 250KS/S, VOLTAGE: 0 – 15V ±(2mV+1%), 12V CURRENT: 100uA-1mA ± (25uA+1%) 1mA-13A ± (2mA+1%), 5V CURRENT: 100uA-1mA ± (25uA+1%) 1mA-13A ± (2mA+1%), 3.3VAUX CURRENT: 0-85mA ± (25uA+1%) DIGITAL CHANNELS: PERST#, CLKREQ#_PERSTB#, WAKE#, SMBCLK, SMBDAT, ACTIVITY#, PWRDIS, PRSNT#, IFDET, IFDET2, HPT0, HPT1, DUAPLORTEN#, P2, SAMPLE RATE 1MS/S

Storage power analysis

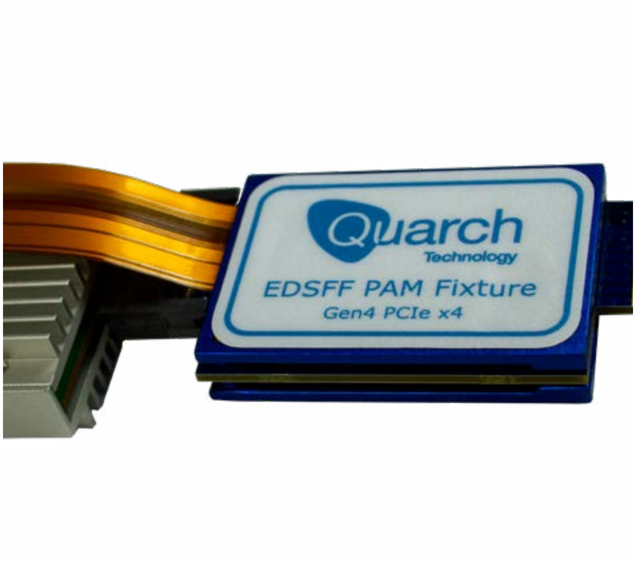
GEN5 M.2 M-KEY PAM FIXTURE

Find out more

M.2 PAM: ANALOG CHANNELS: 3.3V, VIO_1V8, SAMPLE RATE 250KS/S, VOLTAGE: 0 – 15V ±(2mV+1%), CURRENT: 0-1mA ±(15uA+1%) 1mA-13A ±(2mA+1%) DIGITAL CHANNELS: CLKREQ#, PERST#, PEWAKE#, SUSCLK, PEDET, ALERT#, SMB_DATA, SMB_CLK, LED_1#, DEVSLP, MFG_DATA, MFG_CLK, VIO_CFG, PWRDIS, PLA_S3#, PLN# SAMPLE RATE 1MS/S

GEN5 EDSFF PAM FIXTURE (E1.S/E1.L/E3/E3 2T)

Find out more



EDSFF PAM: ANALOG CHANNELS: 12V, 3.3VAUX, SAMPLE RATE 250KS/S, VOLTAGE: 0 – 15V ±(2mV+1%), 12V CURRENT: 100uA-1mA ± (25uA+1%) 1mA-13A ± (2mA+1%), 3.3VAUX CURRENT: 100uA-85mA ± (25uA+1%) DIGITAL CHANNELS:PRSNT0#, PERST1#_CLKREQ#, LED, SMBRST#, SMBDAT, SMBCLK, PWRDIS, PERST0#, DUALPORTEN#, RFU, MFG, SAMPLE RATE 1MS/S

Storage and beyond

GEN5 AIC/SLOT X16 PAM

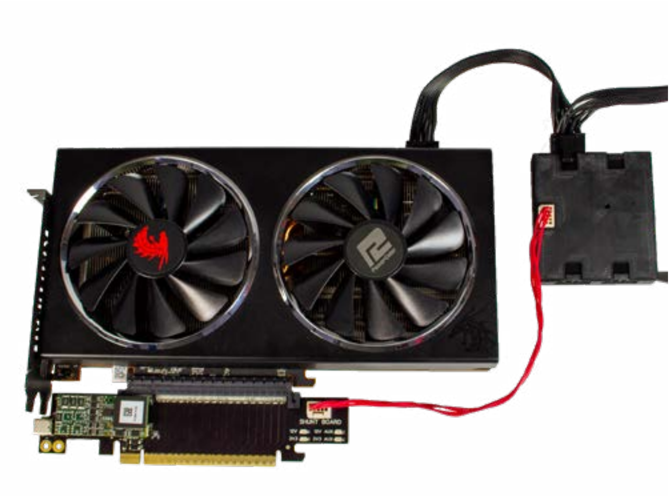
Supports all slot powered AIC devices: SSDs, NICs, HBAs and more

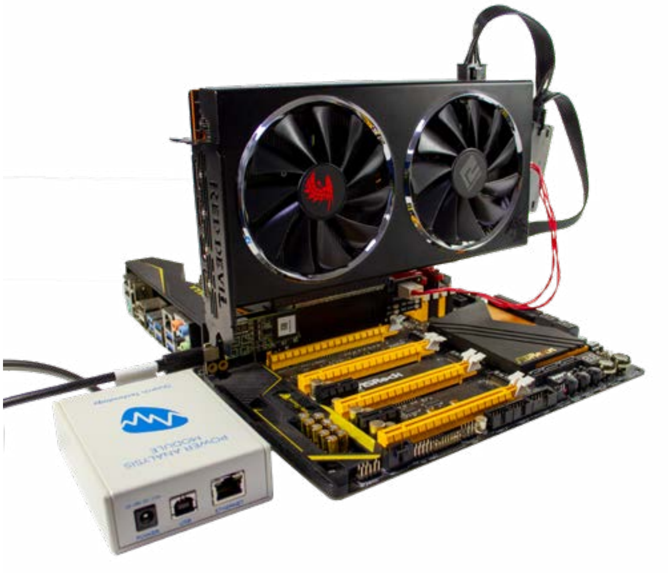
Find out more

AIC PAM: Analog Channels: 12V, 3.3V, 3.3Vaux, Sample Rate 250KS/s, Voltage: 0 – 15V ±(2mV+1%), 12V Current: 100uA-1mA ± (10uA+1%) 1mA-13A ± (2mA+1%), 3V3 Current: 100uA-1mA ± (10uA+1%) 1mA-13A ± (2mA+1%), 3.3Vaux Current: 0-400mA ± (10uA+1%) Digital Channels: PERST#, CLKREQ#, WAKE#, SMBCLK, SMBDAT, Sample Rate 1MS/s

GEN5 AIC/SLOT X16 PAM FIXTURE +AUX POWER

Find out more





AIC +AUX PAM: ANALOG CHANNELS: 12V, 3.3V, 3.3VAUX, 12VAUX, SAMPLE RATE 250KS/S, VOLTAGE: 0 – 15V ±(2mV+1%), 12V CURRENT: 0-32.5A ± (5mA+1%), 3.3V CURRENT: 0-13A ± (2mA+1%), 3.3VAUX CURRENT: 0-3.25A ± (0.5mA+1%), 12VAUX CURRENT: 0-162.5A ± (25mA+1%) DIGITAL CHANNELS: PERST#, CLKREQ#, WAKE#, SMBCLK, SMBDAT, REFCLK_LOS#, SAMPLE RATE 1MS/S

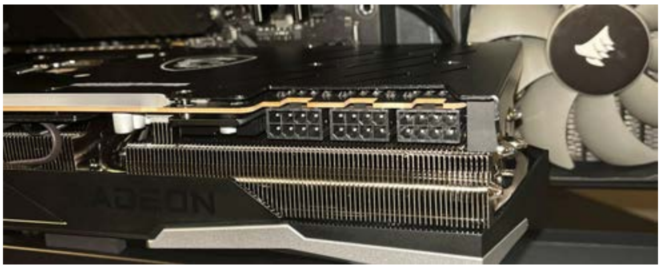
GPU and AI Analysis

AUX FIXTURES (DUAL PCIE, TRIPLE PCIE, 12VHP)

Calibrated fixtures support all major GPU / Accelerator power supplies

Includes power sequencing

Dual PCIe, Triple PCIe and 12VHP versions



2 CHANNEL PAM

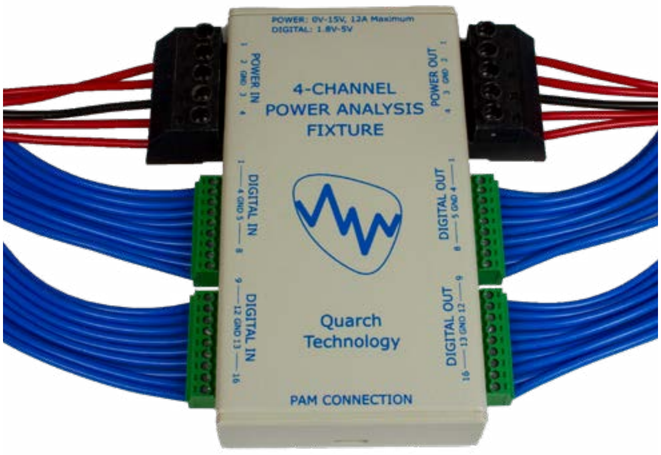
Find out more

2 CHANNEL: ANALOG CHANNELS: 2 CHANNELS, SAMPLE RATE 250KS/S, VOLTAGE: 0 – 15V ±(2mV+1%), CURRENT: 0-1mA ±(15uA+1%) 1mA-13A ±(2mA+1%) DIGITAL CHANNELS: 16 CHANNELS SAMPLE RATE 1MS/S, VOLTAGE RANGE 1.8-5V

Multi-channel fixtures

4 CHANNEL PAM

Find out more

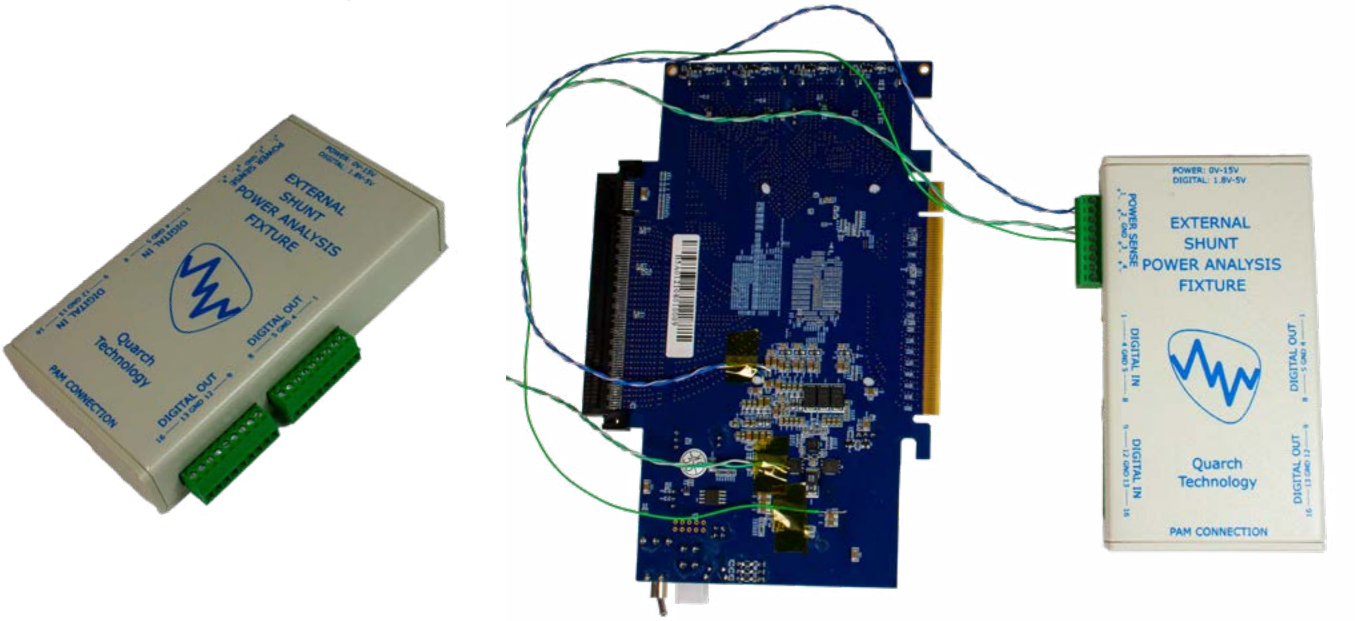


4 CHANNEL: ANALOG CHANNELS: 4 CHANNELS, SAMPLE RATE 250KS/S, VOLTAGE: 0 – 15V ±(2mV+1%), CURRENT: 10mA-13A ±(10mA+1%) DIGITAL CHANNELS: 16 CHANNELS SAMPLE RATE 1MS/S, VOLTAGE RANGE 1.8-5V

EXTERNAL SHUNT PAM

Find out more

Connects into a wide range of embedded shunts



EXTERNAL SHUNT: 4 CHANNELS, SAMPLE RATE 250KS/S, VOLTAGE: 0 – 15V ±(2mV+1%), CURRENT SENSE: 10uV-65mV ±(10uV+1%) DIGITAL CHANNELS: 16 CHANNELS SAMPLE RATE 1MS/S, VOLTAGE RANGE 1.8-5V

Multi-channel fixtures

EXTERNAL SHUNT PAM WITH DIMM FIXTURE

Connected to DIMM adaptor board

Allowing analyis of RAM power consumption

Supports 50mV current sense resistors

SINGLE PHASE AC PAM (IEC C14 CONNECTIONS)

Find out more

AC analysis devices have inbuilt USB/LAN ports and do not require a seperate controller.

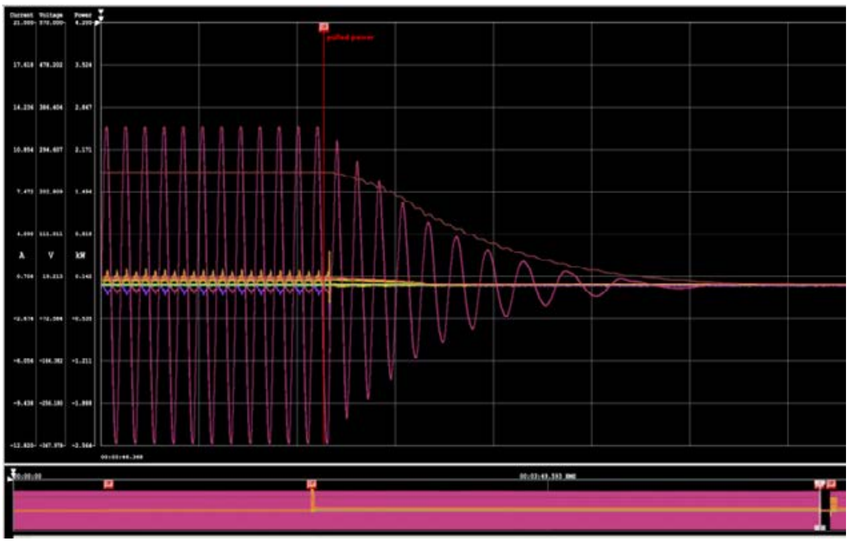
Plug-and-play connection avoids high voltage safety risks

IEC PAM: Single IEC 60320 C14 10A Fused Input, 3 Individually Measured IEC 60320 C13 Outputs, Sample Rate 8KS/s, Voltage: ±495.5V Peak 50Vac-270Vac ±0.5%, Current: 100mA-44A ±(10mA+0.5%)

AC Power Analysis

SUPPORTS UP TO 3 DEVICES AT THE SAME TIME

Suitable for testing everything from consumer electronics to high end server systems



AC products share the same comprehensive analysis and automation tools.

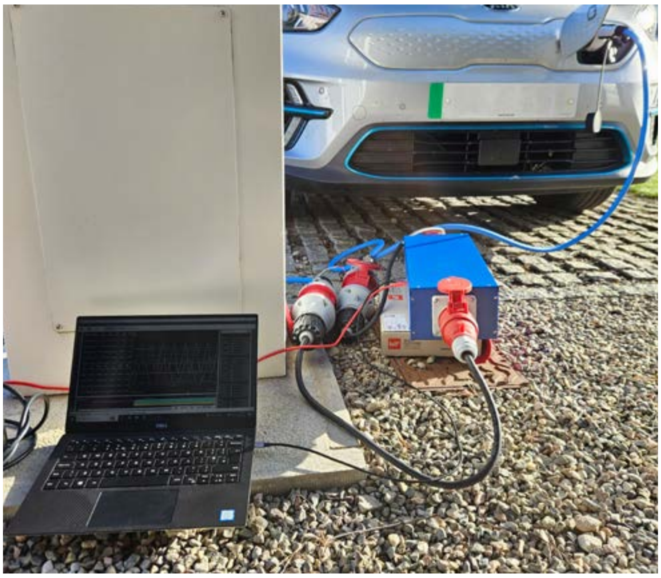
Analyze frequency, phase, THD, power factor and more.

AC Power Analysis

3-PHASE AC PAM (16A, 32A AND 63A VERSIONS)

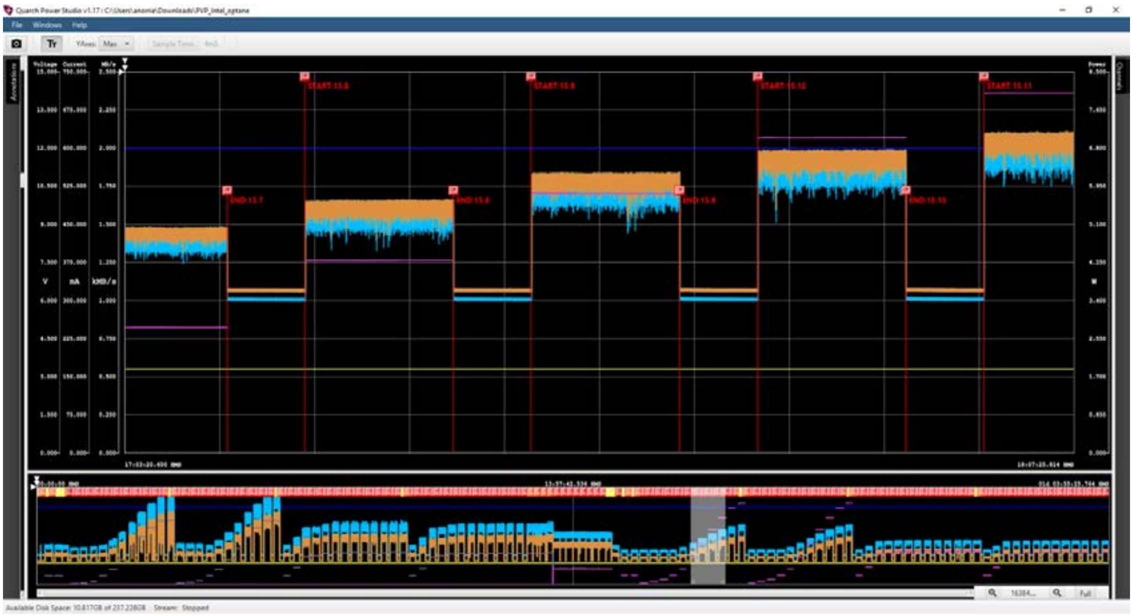
Find out more

Supports EV charging, AC Motor analysis, large IT racks and more,



LONG TERM, HIGH RESOLUTION CAPTURE

Record for hours or days and still zoom in to the smallest details.Add annotations and notes



Quarch Power Studio v1.17 : C:\Users\anomie\Downloads\PVP_Intel_optane

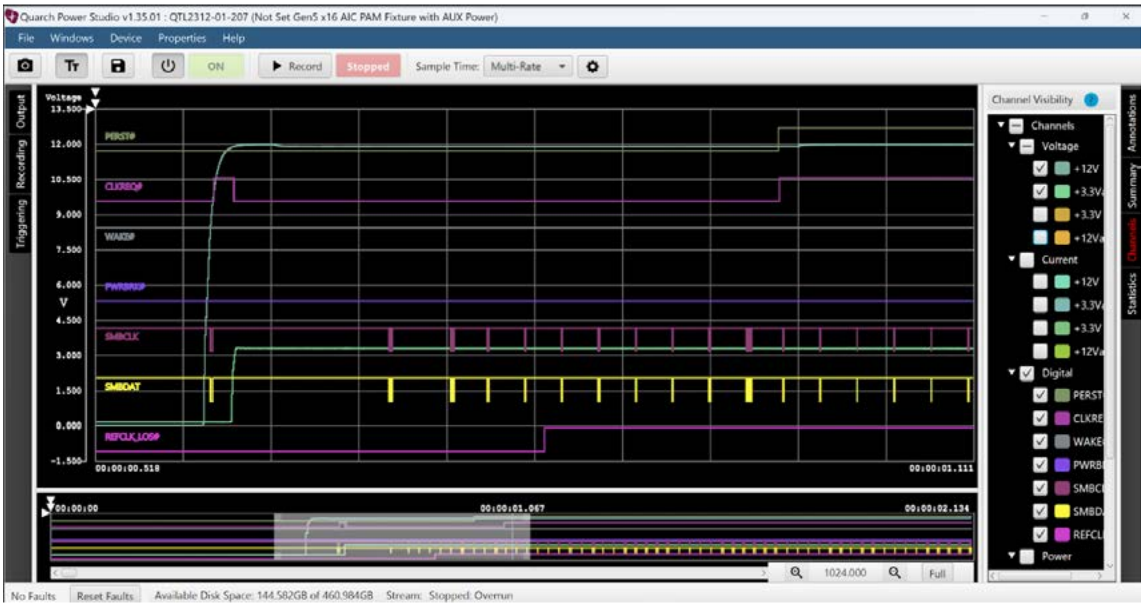
Available Disk Space: 10.817GB of 237.228GB   Stream: Stopped

Quarch Power Studio (QPS)

ANALOG, DIGITAL AND CUSTOMER USER CHANNELS

Find out more

See the entire picture, including custom user channels for your own data (ie: temperature, performance, speed)



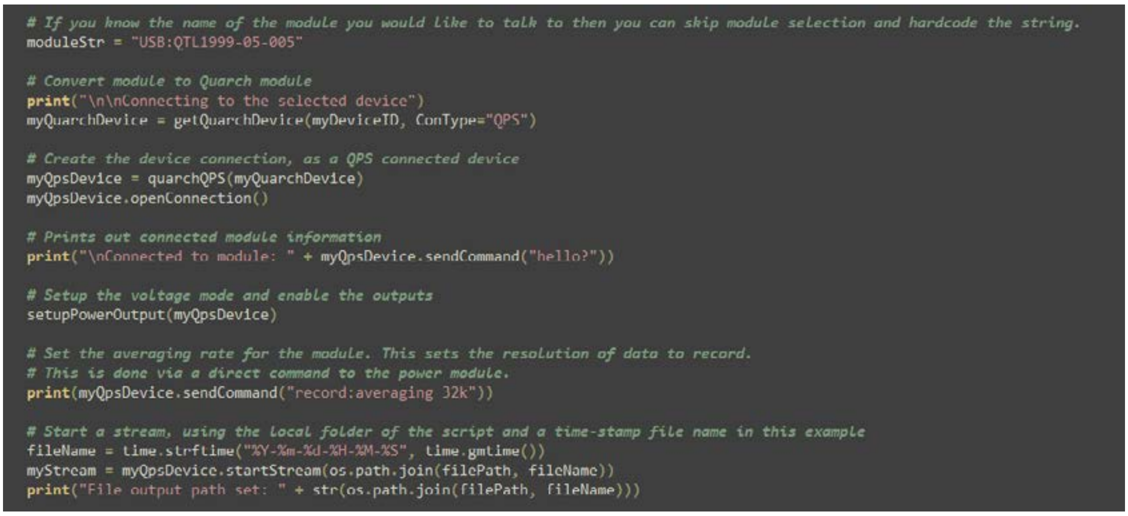
Quarch Power Studio v1.35.01 : QTL2312-01-207 (Not Set Gen5 x16 AIC PAM Fixture with AUX Power)

No Faults   Reset Faults   Available Disk Space: 144.582GB of 460.984GB   Stream: Stopped: Overrun

SIMPLE AUTOMATION API

Find out more

Automate capture, annotations, custom channels, statistic calculations and more. Code examples: www.quarch.com/support/application-note



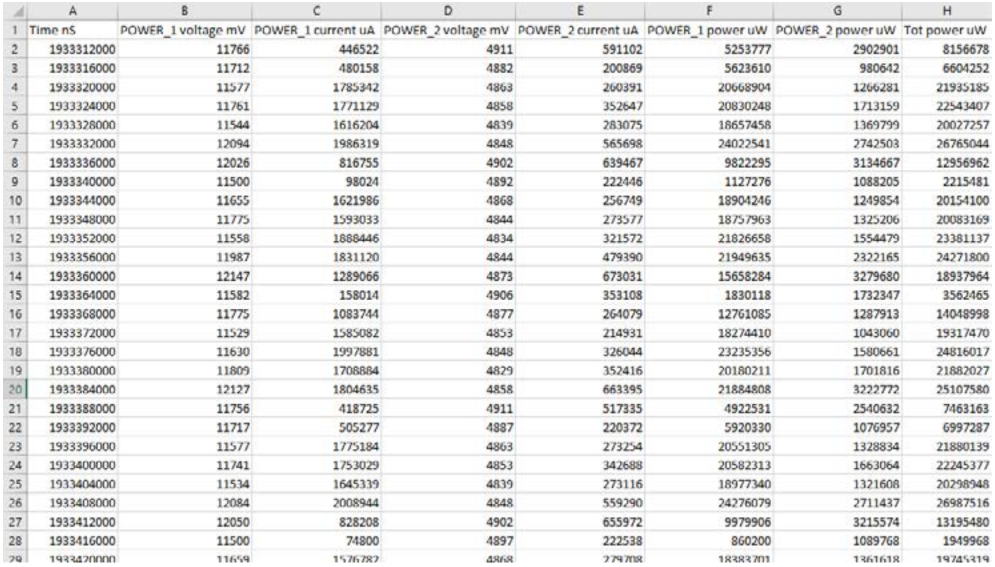
```
# If you know the name of the module you would like to talk to then you can skip module selection and hardcode the string.
moduleStr = "USB:QTL1999-05-005"

# Convert module to Quarch module
print("\n\nConnecting to the selected device")
myQuarchDevice = getQuarchDevice(myDeviceID, ConType="QPS")

# Create the device connection, as a QPS connected device
myQpsDevice = quarchQPS(myQuarchDevice)
myQpsDevice.openConnection()

# Prints out connected module information
print("\nConnected to module: " + myQpsDevice.sendCommand("hello?"))

# Setup the voltage mode and enable the outputs
setupPowerOutput(myQpsDevice)

# Set the averaging rate for the module. This sets the resolution of data to record.
# This is done via a direct command to the power module.
print(myQpsDevice.sendCommand("record:averaging 32k"))

# Start a stream, using the local folder of the script and a time-stamp file name in this example
fileName = time.strftime("%Y-%m-%d-%H-%M-%S", time.gmtime())
myStream = myQpsDevice.startStream(os.path.join(filePath, fileName))
print("File output path set: " + str(os.path.join(filePath, fileName)))
```

Quarch Power Studio (QPS)

EASY SHARING AND POST PROCESSING

Find out more

Share full traces or smaller sections. Export to CSV format for custom post-processing

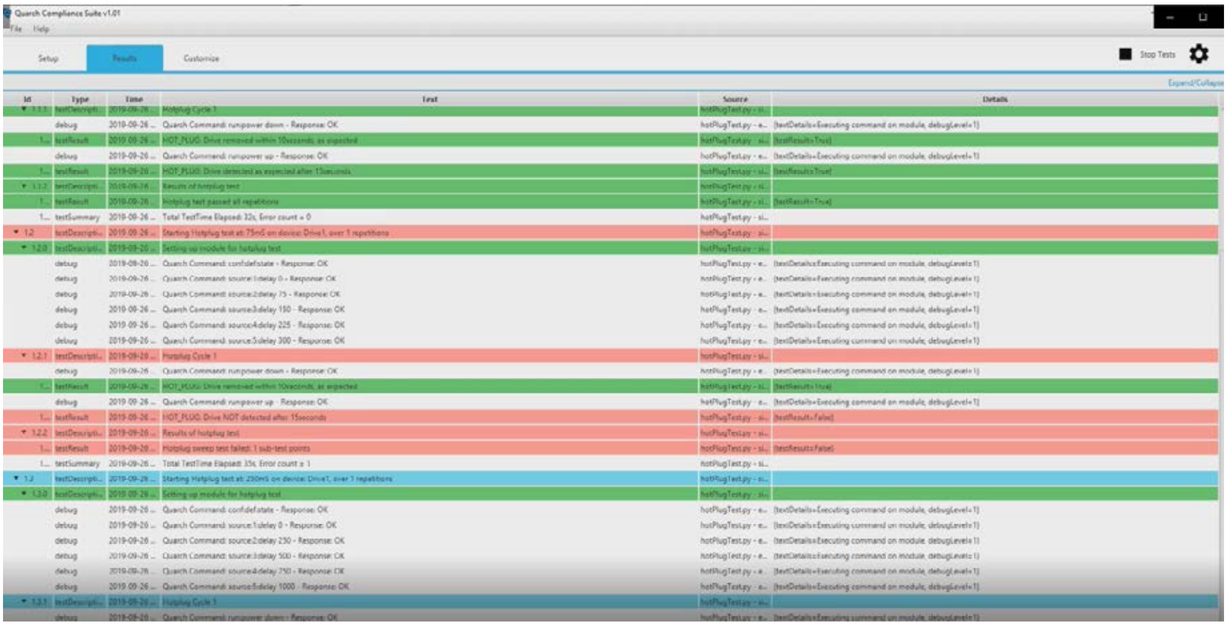


	A	B	C	D	E	F	G	H
1	Time nS	POWER_1 voltage mV	POWER_1 current uA	POWER_2 voltage mV	POWER_2 current uA	POWER_1 power uW	POWER_2 power uW	Tot power uW
2	1933312000	11766	446522	4911	591102	5253777	2902901	8156678
3	1933316000	11712	480158	4882	200869	5623610	980642	6604252
4	1933320000	11577	1785342	4863	260391	20668904	1266281	21935185
5	1933324000	11761	1771129	4858	352647	20830248	1713159	22543407
6	1933328000	11544	1616204	4839	283075	18657458	1369799	20027257
7	1933332000	12094	1986319	4848	565698	24022541	2742503	26765044
8	1933336000	12026	816755	4902	639467	9822295	3134667	12956962
9	1933340000	11500	98024	4892	222446	1127276	1088205	2215481
10	1933344000	11655	1621986	4868	256749	18904246	1249854	20154100
11	1933348000	11775	1593033	4844	273577	18757963	1325206	20083169
12	1933352000	11558	1888446	4834	321572	21826658	1554479	23381137
13	1933356000	11987	1831120	4844	479390	21949635	2322165	24271800
14	1933360000	12147	1289066	4873	673031	15658284	3279680	18937964
15	1933364000	11582	158014	4906	353108	1830118	1732347	3562465
16	1933368000	11775	1083744	4877	264079	12761085	1287913	14048998
17	1933372000	11529	1585082	4853	214931	18274410	1043060	19317470
18	1933376000	11630	1997881	4848	326044	23235356	1580661	24816017
19	1933380000	11809	1708884	4829	352416	20180211	1701816	21882027
20	1933384000	12127	1804635	4858	663395	21884808	3222772	25107580
21	1933388000	11756	418725	4911	517335	4922531	2540632	7463163
22	1933392000	11717	505277	4887	220372	5920330	1076957	6997287
23	1933396000	11577	1775184	4863	273254	20551305	1328834	21880139
24	1933400000	11741	1753029	4853	342688	20582313	1663064	22245377
25	1933404000	11534	1645339	4839	273116	18977340	1321608	20298948
26	1933408000	12084	2008944	4848	559290	24276079	2711437	26987516
27	1933412000	12050	828208	4902	655972	9979906	3215574	13195480
28	1933416000	11500	74800	4897	222538	860200	1089768	1949968
29	1933420000	11659	1576782	4868	279708	18383701	1361618	19745319

QUARCH COMPLIANCE SUITE

Find out more

Run standard automated workload, voltage margining tests and more



Id	Type	Time	Text	Source	Details
▼ 1.1.1	testDescripti...	2019-09-26 ...	Hotplug Cycle 1	hotPlugTest.py - si...	
	debug	2019-09-26 ...	Quarch Command: runpower down - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
1...	testResult	2019-09-26 ...	HOT_PLUG: Drive removed within 10seconds, as expected	hotPlugTest.py - si...	{testResult=True}
	debug	2019-09-26 ...	Quarch Command: runpower up - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
1...	testResult	2019-09-26 ...	HOT_PLUG: Drive detected as expected after 15seconds	hotPlugTest.py - si...	{testResult=True}
▼ 1.1.2	testDescripti...	2019-09-26 ...	Results of hotplug test	hotPlugTest.py - si...	
1...	testResult	2019-09-26 ...	Hotplug test passed all repetitions	hotPlugTest.py - si...	{testResult=True}
1...	testSummary	2019-09-26 ...	Total TestTime Elapsed: 32s, Error count = 0	hotPlugTest.py - si...	
▼ 1.2	testDescripti...	2019-09-26 ...	Starting Hotplug test at: 75mS on device: Drive1, over 1 repetitions	hotPlugTest.py - si...	
▼ 1.2.0	testDescripti...	2019-09-26 ...	Setting up module for hotplug test	hotPlugTest.py - si...	
	debug	2019-09-26 ...	Quarch Command: conf:def:state - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
	debug	2019-09-26 ...	Quarch Command: source:1:delay 0 - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
	debug	2019-09-26 ...	Quarch Command: source:2:delay 75 - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
	debug	2019-09-26 ...	Quarch Command: source:3:delay 150 - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
	debug	2019-09-26 ...	Quarch Command: source:4:delay 225 - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
	debug	2019-09-26 ...	Quarch Command: source:5:delay 300 - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
▼ 1.2.1	testDescripti...	2019-09-26 ...	Hotplug Cycle 1	hotPlugTest.py - si...	
	debug	2019-09-26 ...	Quarch Command: runpower down - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
1...	testResult	2019-09-26 ...	HOT_PLUG: Drive removed within 10seconds, as expected	hotPlugTest.py - si...	{testResult=True}
	debug	2019-09-26 ...	Quarch Command: runpower up - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
1...	testResult	2019-09-26 ...	HOT_PLUG: Drive NOT detected after 15seconds	hotPlugTest.py - si...	{testResult=False}
▼ 1.2.2	testDescripti...	2019-09-26 ...	Results of hotplug test	hotPlugTest.py - si...	
1...	testResult	2019-09-26 ...	Hotplug sweep test failed: 1 sub-test points	hotPlugTest.py - si...	{testResult=False}
1...	testSummary	2019-09-26 ...	Total TestTime Elapsed: 35s, Error count = 1	hotPlugTest.py - si...	
▼ 1.3	testDescripti...	2019-09-26 ...	Starting Hotplug test at: 250mS on device: Drive1, over 1 repetitions	hotPlugTest.py - si...	
▼ 1.3.0	testDescripti...	2019-09-26 ...	Setting up module for hotplug test	hotPlugTest.py - si...	
	debug	2019-09-26 ...	Quarch Command: conf:def:state - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
	debug	2019-09-26 ...	Quarch Command: source:1:delay 0 - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
	debug	2019-09-26 ...	Quarch Command: source:2:delay 250 - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
	debug	2019-09-26 ...	Quarch Command: source:3:delay 500 - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
	debug	2019-09-26 ...	Quarch Command: source:4:delay 750 - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
	debug	2019-09-26 ...	Quarch Command: source:5:delay 1000 - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}
▼ 1.3.1	testDescripti...	2019-09-26 ...	Hotplug Cycle 1	hotPlugTest.py - si...	
	debug	2019-09-26 ...	Quarch Command: runpower down - Response: OK	hotPlugTest.py - e...	{textDetails=Executing command on module, debugLevel=1}

Automation options

QIS & QUARCHPY

Find out more

Java Instrumentation Server allowing simple TCP based control of any Quarch Power Device. Full Python API available for fast integration

```python
from quarchpy.device import *

# Specify the module to control
myDeviceID = "USB:QTL1999-05-005"

# Connect to the module
myQuarchDevice = getQuarchDevice(myDeviceID, ConType = "QIS")

# Convert the base device class to a power device, which provides additional controls, such as data streaming
myPowerDevice = quarchPPM(myQuarchDevice)

module.startStream('Stream1.csv')

```

Download QIS from: **quarch.com/downloads**
Download quarchpy from: **pypi.org/project/quarchpy/**