DOC-218480-01

Prepared By: Peter Milne

Date: February 22 2022.

Rev	Date	Description
1	20220202	Initial.
2	2023-06-12	Update to show AO+DO, DO only. Appendix 9,10,11

Table of Contents

Copyright and Attribution.

Document created using Libre Office https://www.libreoffice.org

This document and D-TACQ Software comprising platform Linux port, Linux kernel modules and most applications are released under GNU GPL/FDL:

Document:

Software:

1 Introduction

1.1 Summary

- Customer order comprises 5xACQ2106 with mix of payload modules.
- We know that one or more boxes will be used in a PCS/LOW LATENCY.
- Customer requires AFHBA404 operation with Dell R740

1.2 Action

- We create an “Example PCS System” with all 5xACQ2106 boxes. This is likely NOT the end-user’s system but it’s the maximal case.
- We test the system using 2xAFHBA404 cards plugged into the R740.
- We use ACQPROC to automatically configure the system and to generate test data.
- We try the AFHBA404 cards in all possible slots in the R740 and report

1.3 Intended Audience

System integrators.

1.4 Scope

ACQ2106, Low Latency Control, AHFBA404, x864, Linux Host.

1.5 References

1. ACQ2106 6 site DAQ appliance.
2. AFHBA404 Host Bus Adapter
3. R740 Dell server (Host PC)
4. ACQPROC Software framework for LLC.
5. 4GUG D-TACQ 4G Product Reference.

1.6 Glossary

- ACQ2106 : D-TACQ 6 site DAQ Appliance, with MGT links
- MGT : Multi-Gigabit Transceiver
- AFHBA404 : Quad SFP Host Bus Adapter.
- UUT : instance of ACQ2106
- HOSTPC : computer with HBA present
- GUIPC: computer running D-TACQ GUI client display.
- LLC : Low Latency Control
- PCS: Plasma Control System. High rate MIMO control.
- VI : Vector IN, comprising
  - AI16: 16 bit analog input
  - DI32: 32 bit Digital Input
  - SP32: 32 bit Scratch Pad, max 16, to pad to 64 byte boundary.
- VO: Vector OUT, comprising
  - AO16: 16 biut analog outputs
  - DO32: 32 bit Digital Output
  - PWM32 : PWM control codes (special firmware).
  - PAD32 : padding to make length up to a 64 byte boundary

2 Hardware in the Order

2.1 Hardware Listing

ITEM	PRODUCT DESCRIPTION	QTY	UNIT#1
1	**ACQ2106+4xACQ424ELF-32+AO424ELF-32+DIO432ELF** 1Ux19" DAQ Appliance, PCS 128AI, 32AO, 32DIO 1GB DRAM, gigabit Ethernet embedded Linux, EPICS inside. MGT482 Quad Port SFP adapter, with 2xGbps fiber links. 4*ACQ424ELF-32: 32 ch simultaneous ADC, 1MSPS/16b 1*AO424ELF-32 : 32 ch simultaneous DAC, 500kSPS/16b 1*DIO432ELF : 32 bit DIO	1	SYS_CE4160349
2	**ACQ2106+4xACQ424ELF-32** 1Ux19" DAQ Appliance, PCS 128AI	1	SYS_CE4160337
3	**ACQ2106+DIO432ELF** 1Ux19" DAQ Appliance, PCS 32DIO	1	SYS_CE4160348
4	**ACQ2106+ACQ424ELF32** 1Ux19" DAQ Appliance, PCS 32AI	1	SYS_CE4160336
5	**ACQ2106+ACQ424ELF32+DIO432ELF** 1Ux19" DAQ Appliance, PCS 32AI	1	SYS_CE4160344
6	**AFHBA404** 4 port PCIe HOST BUS ADAPTER, provided with 4x5m SFP AOC	3	H3404090 H3404091 H3404092
7	**BNCPANEL-32:** 2U x 19" steel panel, 32 floating BNC Connectors, transil overvoltage suppression, 1m back cable.	13	

2.2 Mapping System serial with UUT Name

ITEM	*SYS SERIAL*	*UUTNAME*	*MAC ADDRESS*
1	SYS_CE4160349	acq2106_349	00:21:54:13:01:5D
2	SYS_CE4160337	acq2106_337	00:21:54:13:01:51
3	SYS_CE4160348	acq2106_348	00:21:54:13:01:5C
4	SYS_CE4160336	acq2106_336	00:21:54:13:01:50
5	SYS_CE4160344	acq2106_344	00:21:54:13:01:58

- Recommend: use DDNS.
- Failing that, setup name - address mapping in //etc/hosts

2.3 Installation

- Connect AC Mains, network
- Recommended: connect serial console.
- Power up and check networking is good. Inventory the units by looking at web pages

2.4 Configuration For test

- We stacked the units in serial number order and
- Connected an HDMI SYNC daisy chain in numerical order

#	UUT	SYNC IN	SYNC_OUT	CABLE
1	acq2106_336	X		--
			X	\
2	acq2106_337	X		/
			X	\
3	acq2106_344	X		/
			X	\
4	acq2106_348	X		/
			X	\
5	acq2106_349	X		/
			X	--

- Connected SFP cables as follows

#	UUT	PORT	HOST SLOT	PORT
1	acq2106_336	A	5	A
2	acq2106_337	A	4	B
3	acq2106_344	A	4	A
4	acq2106_348	A	4	C
5	acq2106_349	A	5	B

3 Software Installation

3.1 OS : CENTOS7

Latest CENTOS7, with kernel-devel, g++, git, x2go server

full python3 installation: python3, numpy, matplotlib

3.1.1 User dt100

- By convention we set up user “dt100”

```
mkdir ~/bin  # check this is in the PATH
```

3.2 D-TACQ HOST Software

- mkdir PROJECTS

3.2.1 HAPI

Host Application Python Interface: for Ethernet automation.

```
cd PROJECTS
git clone https://github.com/D-TACQ/acq400_hapi.git
```

3.2.2 AFHBA404

HOST side device driver and ACQPROC LLC Framework

- cd PROJECTS
- git clone https://github.com/D-TACQ/AFHBA404.git
- follow instructions:
  - https://github.com/D-TACQ/AFHBA404/blob/master/README.md
  - https://github.com/D-TACQ/AFHBA404/blob/master/INSTALL
- Install:

```
cd AFHBA404
make
sudo ./scripts/loadNIRQ
get-ident-all
```

3.3 D-TACQ GUI Software

- D-TACQ recommends cs-studio. Typically this is installed on GUIPC; typically this is NOT the HOST PC (server).
- Please follow instructions at https://github.com/D-TACQ/ACQ400CSS

4 Run The Test

4.1 Initialise

```
[dt100@roger740 ~]$ cd PROJECTS/acq400_hapi
[dt100@roger740 acq400_hapi]$ source ./setpath
[dt100@roger740 acq400_hapi]$ cd ../AFHBA404
[dt100@roger740 AFHBA404]$ sudo ./scripts/loadNIRQ
[dt100@roger740 AFHBA404]$ get-ident-all
roger740 0 acq2106_336 A
roger740 1 acq2106_349 A
roger740 4 acq2106_344 A
roger740 5 acq2106_337 A
roger740 6 acq2106_348 A
```

4.2 Get Configuration

```
[dt100@roger740 AFHBA404]$ python3 \
./HAPI/lsafhba.py --verbose=1 --save_config ACQPROC/configs/swip5_45.json
```

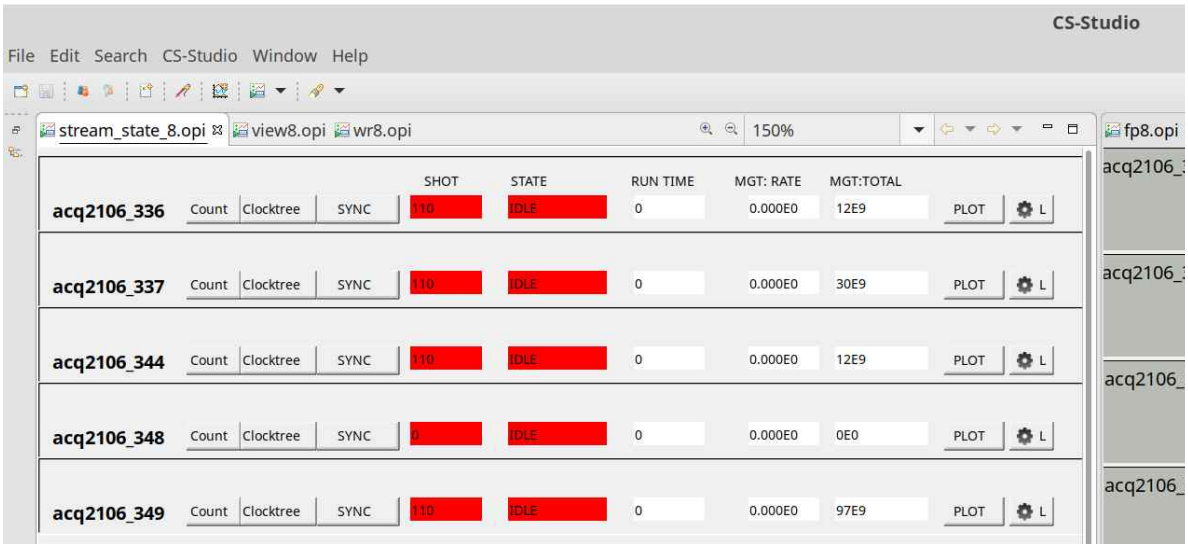
We chose config “swip5_45” because:

- 5 ACQ2106 units
- 2 x AFHBA404 from the listing below, we see
  - Slot 5 has 2 fibers (shows at dev 0, 1)
  - Slot 4 has 3 fibers (shows as dev 4,5,6).

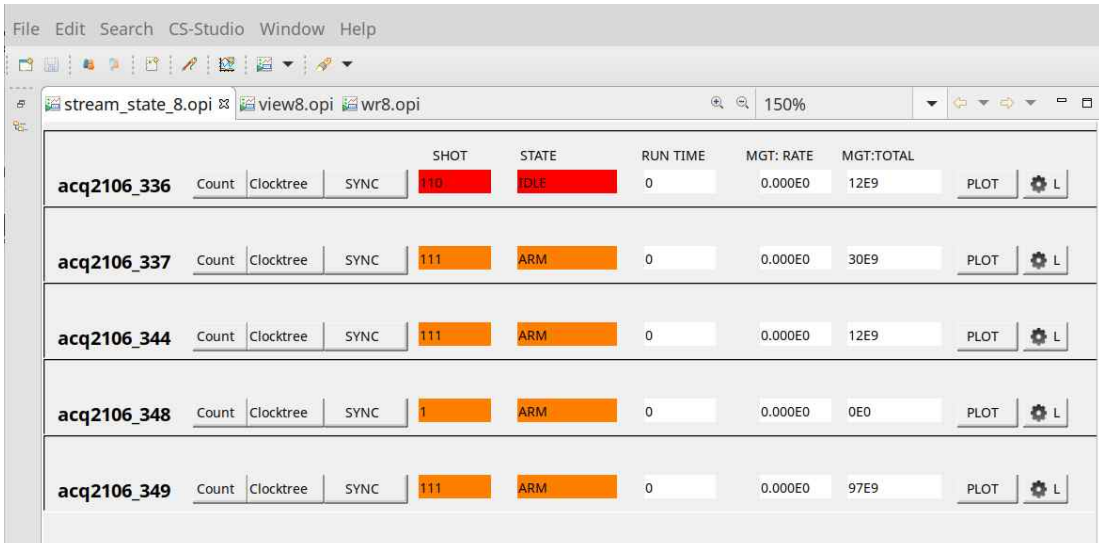
```
[dt100@roger740 AFHBA404]$ python3 ./HAPI/lsafhba.py --verbose=1 --save_config
ACQPROC/configs/swip5_45.json
0 HostComms(host='roger740', dev='0', uut='acq2106_336', cx='A')
                payload:[1]
                                site:1 MODULE ACQ424ELF AI16 32
1 HostComms(host='roger740', dev='1', uut='acq2106_349', cx='A')
                payload:[1, 2, 3, 4, 5, 6]
                                site:1 MODULE ACQ424ELF AI16 32
                                site:2 MODULE ACQ424ELF AI16 32
                                site:3 MODULE ACQ424ELF AI16 32
                                site:4 MODULE ACQ424ELF AI16 32
                                site:5 MODULE AO424ELF AO16 32
                                site:6 MODULE DIO432ELF DI32 32
2 HostComms(host='roger740', dev='4', uut='acq2106_344', cx='A')
                payload:[1, 6]
                                site:1 MODULE ACQ424ELF AI16 32
                                site:6 MODULE DIO432ELF DI32 32
3 HostComms(host='roger740', dev='5', uut='acq2106_337', cx='A')
                payload:[1, 2, 3, 4]
                                site:1 MODULE ACQ424ELF AI16 32
                                site:2 MODULE ACQ424ELF AI16 32
                                site:3 MODULE ACQ424ELF AI16 32
                                site:4 MODULE ACQ424ELF AI16 32
4 HostComms(host='roger740', dev='6', uut='acq2106_348', cx='A')
                payload:[1, 6]
                                site:1 MODULE ACQ424ELF AI16 32
                                site:6 MODULE DIO432ELF DI32 32
```

4.3 Run

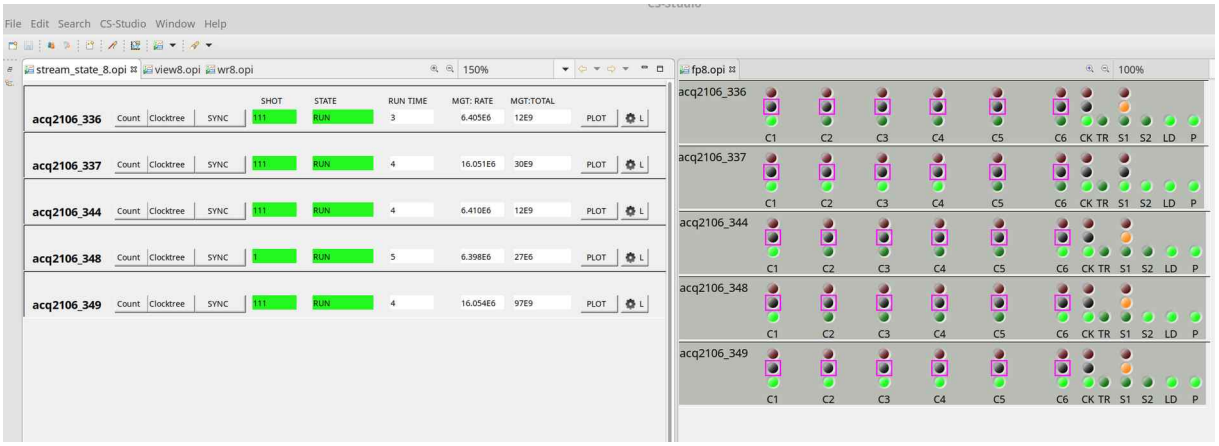
```
[dt100@roger740 AFHBA404]$ \
./scripts/acqproc_multi.sh ACQPROC/configs/swip5_45.json
```



- All IDLE at start

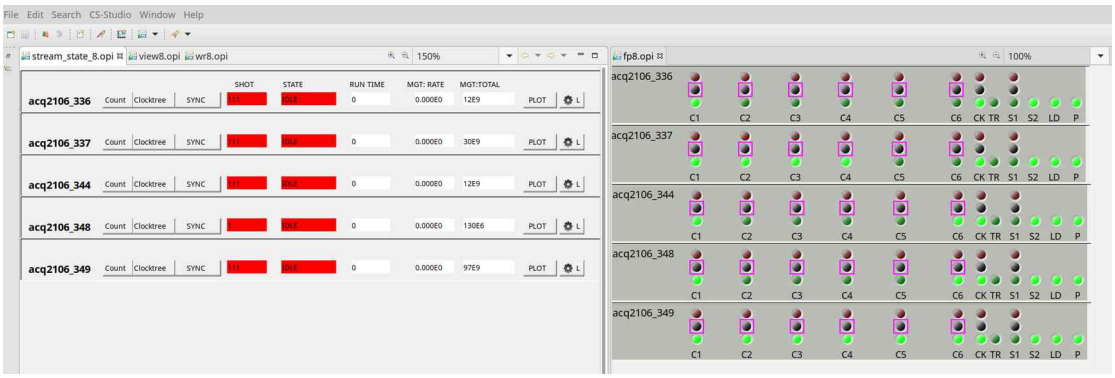


- UUT's are ARMED from bottom to top.



- During the shot, all units show state RUN

- Live display of Run Time, MGT (Comms A Rate and Total).



- End of shot. IDLE again.

4.3.1 Run Transcript

4.3.1.1 Configure: configure_uut

```
[dt100@roger740 AFHBA404]$ ./scripts/acqproc_multi.sh
ACQPROC/configs/swip5_45.json
UUT1 acq2106_336
UUT2 acq2106_349
UUTS acq2106_336 acq2106_349 acq2106_344 acq2106_337 acq2106_348
DEVMAX 6
python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/sync_role.py --
toprole=master --fclk=50000 acq2106_336 acq2106_349 acq2106_344
acq2106_337 acq2106_348

python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/sync_role.py
done 0
python3 scripts/llc-config-utility.py ACQPROC/configs/swip5_45.json
acq2106_336 Aggregator settings: sites=1 (spad=16) (comms=A)
acq2106_349 Aggregator settings: sites=1,2,3,4,6 (spad=15) (comms=A)
acq2106_349 Distributor settings: sites=5,6 pad=15 comms=A on
acq2106_344 Aggregator settings: sites=1,6 (spad=15) (comms=A)
acq2106_344 Distributor settings: sites=6 pad=15 comms=A on
acq2106_337 Aggregator settings: sites=1,2,3,4 (spad=16) (comms=A)
acq2106_348 Aggregator settings: sites=1,6 (spad=15) (comms=A)
acq2106_348 Distributor settings: sites=6 pad=15 comms=A on
python3 scripts/llc-config-utility.py done 0
```

4.3.1.2 Control Script 1: control_script

```
python3
/home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/acq400_streamtonowher
e.py --samples=1000000 acq2106_336 acq2106_349 acq2106_344
acq2106_337 acq2106_348
nsamples set 1000000
```

4.3.1.3 Control Program 1: control_program

```
HBA6 VI:1024 VO:76 devs=0,1,4,5,6
            [0] dev:0 acq2106_336 VI:128 VO:0 Offset of SPAD IN VI :64
 System Interface Indices 0,0
            [1] dev:1 acq2106_349 VI:320 VO:68 Offset of SPAD IN VI :260
 System Interface Indices 32,16
            [2] dev:4 acq2106_344 VI:128 VO:4 Offset of SPAD IN VI :68
 System Interface Indices 160,31
            [3] dev:5 acq2106_337 VI:320 VO:0 Offset of SPAD IN VI :256
 System Interface Indices 192,46
            [4] dev:6 acq2106_348 VI:128 VO:4 Offset of SPAD IN VI :68
 System Interface Indices 320,62
```

4.3.1.4 During the shot ..

```
Streamed 0 of 1000000 samples
Streamed 89584 of 1000000 samples
Streamed 139608 of 1000000 samples
Streamed 189632 of 1000000 samples
Streamed 239656 of 1000000 samples
Streamed 289680 of 1000000 samples
Streamed 339704 of 1000000 samples
Streamed 389728 of 1000000 samples
Streamed 439752 of 1000000 samples
Streamed 489776 of 1000000 samples
Streamed 539800 of 1000000 samples
Streamed 589824 of 1000000 samples
Streamed 639848 of 1000000 samples
Streamed 689873 of 1000000 samples
Streamed 739896 of 1000000 samples
Streamed 789920 of 1000000 samples
Streamed 839944 of 1000000 samples
Streamed 889969 of 1000000 samples

stored acq2106_336_VI.dat, len=1000000
stored acq2106_336_VO.dat, len=0
stored acq2106_349_VI.dat, len=1000000
stored acq2106_349_VO.dat, len=1000000
stored acq2106_344_VI.dat, len=1000000
stored acq2106_344_VO.dat, len=1000000
stored acq2106_337_VI.dat, len=1000000
stored acq2106_337_VO.dat, len=0
stored acq2106_348_VI.dat, len=1000000
stored acq2106_348_VO.dat, len=1000000
Control Program Finished
Streamed 939993 of 1000000 samples
Streamed 990017 of 1000000 samples

Stream finished.
```

4.3.1.5 Analysis: analysis

```
Running analysis now.
--------------------
python3 scripts/acqproc_analysis.py --ones=1 --json=1 --json_src=./runtime.json
--src=/home/dt100/PROJECTS/AFHBA404/
Running analysis for UUT: acq2106_336
show SPAD: hexdump -ve '32/2 "%04x," 16/4 "%08x," "\n"' acq2106_336_VI.dat  | cut -d, -f33-40 | head
-n 4
00000001,0000000d,00000000,00000000,00000000,00000000,a6666336,a7777336
00000002,00000021,00000000,00000000,00000000,00000000,a6666336,a7777336
00000003,00000035,00000000,00000000,00000000,00000000,a6666336,a7777336
00000004,00000049,00000000,00000000,00000000,00000000,a6666336,a7777336
show whole VI like this:
hexdump -ve '32/2 "%04x," 16/4 "%08x," "\n"' acq2106_336_VI.dat
Finished collecting data
T_LATCH differences:  1 , happened:  625003  times
T_LATCH differences:  2 , happened:  0  times
T_LATCH differences:  3 , happened:  0  times
Cannot run latency analysis for this UUT.
Running analysis for UUT: acq2106_349
show SPAD: hexdump -ve '128/2 "%04x," 1/4 "%08x," 15/4 "%08x," "\n"' acq2106_349_VI.dat  | cut -d, -
f130-137 | head -n 4
00000001,00000001,00000000,00000000,01240124,01330122,0a1baff6,a7777349
00000002,00000015,00000000,00000000,01240124,01330122,0a1baff6,a7777349
00000003,00000029,00000000,00000000,01240124,01330122,0a1baff6,a7777349
00000004,0000003d,00000000,00000000,01240124,01330122,0a1baff6,a7777349
show whole VI like this:
hexdump -ve '128/2 "%04x," 1/4 "%08x," 15/4 "%08x," "\n"' acq2106_349_VI.dat
Finished collecting data
T_LATCH differences:  1 , happened:  625003  times
T_LATCH differences:  2 , happened:  0  times
T_LATCH differences:  3 , happened:  0  times
(160000000,)
data end: 129
spad_len:  15
160
len diffs = 19990
max data = [4.38  4.305 4.275 ... 4.56  4.275 4.305]
scripts/acqproc_analysis.py:145: MatplotlibDeprecationWarning: The 'nonposy' parameter of __init__()
has been renamed 'nonpositive' since Matplotlib 3.3; support for the old name will be dropped two
minor releases later.
  axs.set_yscale('log', nonposy='clip')
latency plot saved as acq2106_349_latency.png
Running analysis for UUT: acq2106_344
show SPAD: hexdump -ve '32/2 "%04x," 1/4 "%08x," 15/4 "%08x," "\n"' acq2106_344_VI.dat  | cut -d, -
f34-41 | head -n 4
00000001,00000002,00000000,00000000,00000000,00000000,a6666344,a7777344
00000002,00000016,00000000,00000000,00000000,00000000,a6666344,a7777344
00000003,0000002a,00000000,00000000,00000000,00000000,a6666344,a7777344
00000004,0000003e,00000000,00000000,00000000,00000000,a6666344,a7777344
show whole VI like this:
hexdump -ve '32/2 "%04x," 1/4 "%08x," 15/4 "%08x," "\n"' acq2106_344_VI.dat
Finished collecting data
T_LATCH differences:  1 , happened:  625004  times
T_LATCH differences:  2 , happened:  0  times
T_LATCH differences:  3 , happened:  0  times
Cannot run latency analysis for this UUT.
Running analysis for UUT: acq2106_337
show SPAD: hexdump -ve '128/2 "%04x," 16/4 "%08x," "\n"' acq2106_337_VI.dat  | cut -d, -f129-136 |
head -n 4
00000001,00000002,00000000,00000000,00000000,00000000,a6666337,a7777337
00000002,00000016,00000000,00000000,00000000,00000000,a6666337,a7777337
00000003,0000002a,00000000,00000000,00000000,00000000,a6666337,a7777337
00000004,0000003e,00000000,00000000,00000000,00000000,a6666337,a7777337
show whole VI like this:
hexdump -ve '128/2 "%04x," 16/4 "%08x," "\n"' acq2106_337_VI.dat
Finished collecting data
T_LATCH differences:  1 , happened:  625004  times
```

```
T_LATCH differences:  2 , happened:  0  times
T_LATCH differences:  3 , happened:  0  times
Cannot run latency analysis for this UUT.
Running analysis for UUT: acq2106_348
show SPAD: hexdump -ve '32/2 "%04x," 1/4 "%08x," 15/4 "%08x," "\n"' acq2106_348_VI.dat  | cut -d, -
f34-41 | head -n 4
00000001,00000001,00000000,00000000,00000000,00000000,a6666348,a7777348
00000002,00000015,00000000,00000000,00000000,00000000,a6666348,a7777348
00000003,00000029,00000000,00000000,00000000,00000000,a6666348,a7777348
00000004,0000003d,00000000,00000000,00000000,00000000,a6666348,a7777348
show whole VI like this:
hexdump -ve '32/2 "%04x," 1/4 "%08x," 15/4 "%08x," "\n"' acq2106_348_VI.dat
Finished collecting data
T_LATCH differences:  1 , happened:  625004  times
T_LATCH differences:  2 , happened:  0  times
T_LATCH differences:  3 , happened:  0  times
Cannot run latency analysis for this UUT.
python3 scripts/acqproc_analysis.py done 0
[dt100@roger740 AFHBA404]$
```

4.3.1.6 Summary Of Analysis

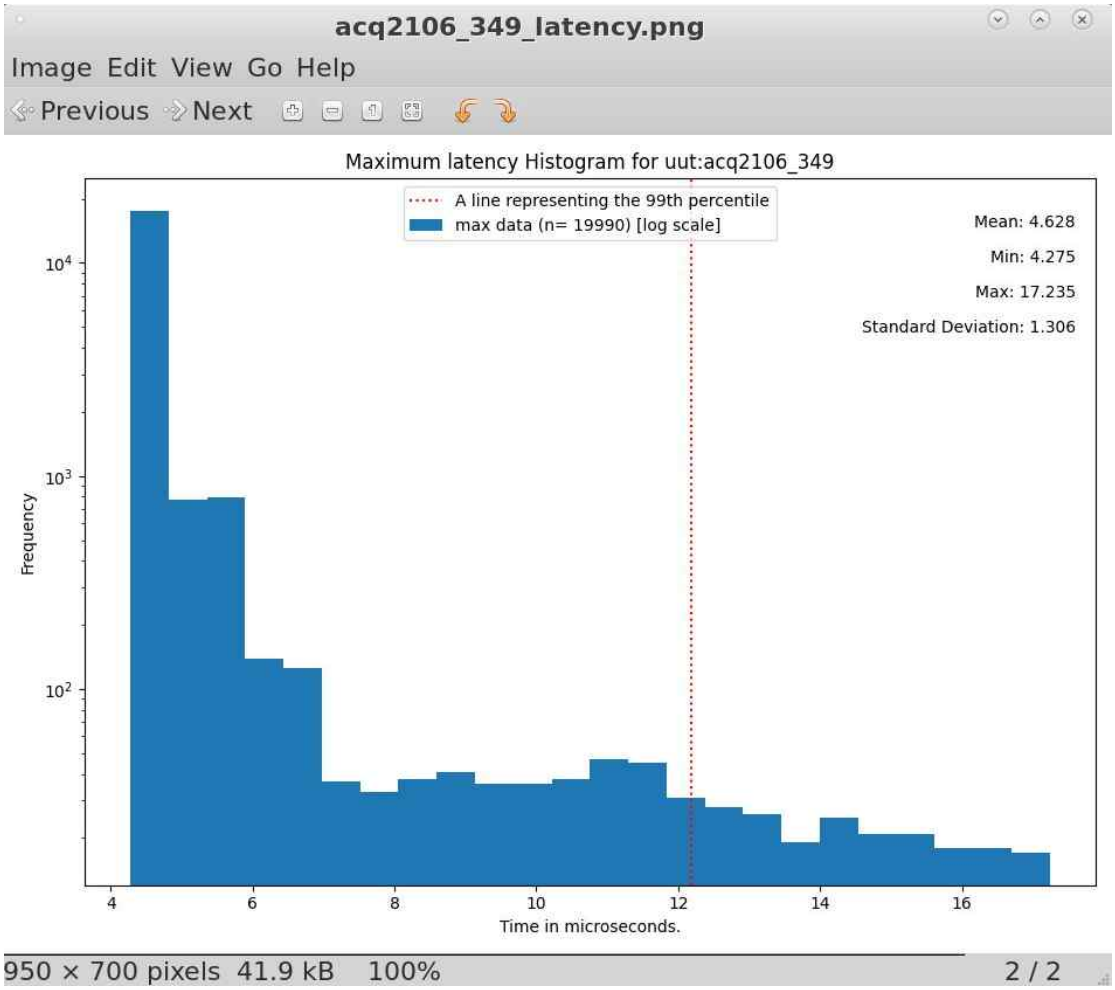
- Scratchpad hexdump

```
show SPAD: hexdump -ve '32/2 "%04x," 16/4 "%08x," "\n"' acq2106_336_VI.dat  | cut -d, -f33-40 | head
-n 4
00000001,0000000d,00000000,00000000,00000000,00000000,a6666336,a7777336
00000002,00000021,00000000,00000000,00000000,00000000,a6666336,a7777336
00000003,00000035,00000000,00000000,00000000,00000000,a6666336,a7777336
00000004,00000049,00000000,00000000,00000000,00000000,a6666336,a7777336
```

- Shows first 4 lines of SPAD region (8 columns)
  - SPAD[0] : TLATCH (Sample Count: 1,2,3,4)
  - SPAD[1] : USECS
  - SPAD[6] : IDENT
  - SPAD[7] : IDENT : COMMS | 77777 | SERIAL
    - Comms A,
    - UUT #336
- TLATCH differences: In a PERFECT system, TLATCH increases by one every sample
  - In this case, it's PERFECT. We're not running in RT mode, but also we're not attempting any heavy processing and 50kHz is NOT a challenging repetition rate.

4.3.1.7 Latency Graph

For system with AO, the hardware makes a histogram of end to end latency, this is published as a graph, stored on disk as a .png file by default:



5 APPENDIX: AFHBA404 in R740

5.1 Needs latest firmware

Current shipping version: AFHBA404 Revision “0e” or later is required:

```
[dt100@roger740 AFHBA404]$ ./scripts/*ser*
af-ba-40-40-11-00-41-0e
af-ba-40-40-11-00-90-0e
```

- af-ba-40-40 : AFHBA404
- 11 : Link capability: 6Gbps with retry
- nn-nn : Serial number
  - 00-41 : SN 41
  - 00-90 : SN 90
- 0e : Firmware revision e.

5.2 Slots where it works

CONFIG	Slot	Proc	Works?
C1	1 / 4SFP	P1	Yes
	2	P2	No Slot
	3	P1	SAS Card
C2	1,4,5	P1, P2	Working
C3	4,5 / 5SFP	P2	Yes, it works
	6	P1?	Half Height Slot
C4	7,8	P2	Empty lspci

- C1: Slot1 works perfectly with 1 x AFHBA404
- C3: Slot 4,5 work perfectly with 2 x AFHBA404
- C2: Best configuration, 3 units working.
- C4: Slots 7, 8 : AFHBA404 is NOT visible in lspci. A 3rd party card was not visible either. We assume we either have a faulty riser or faulty BIOS.
- Slot 6 is too small for AFHBA404 and so is not considered.

5.2.1 Summary and Further work

- R740 works for up to 3 xAFHBA404 and up to 12 ACQ2106 so far.


1
Works
2
Fake slot
3
Raid Controller
4
Works
5
Works
6
Half slot
7
Doesn't work
8
Doesn't work
EPP
1100W
EPP
1100W

6 APPENDIX: Hardware configuration.

6.1 Udev rules for Serial TTY

D-TACQ uses an automated serial console mapping based on udev:

- https://github.com/D-TACQ/USB_TTY_IDS
- git clone https://github.com/D-TACQ/USB_TTY_IDS.git
- update rules.d/90-acq2106.rules:

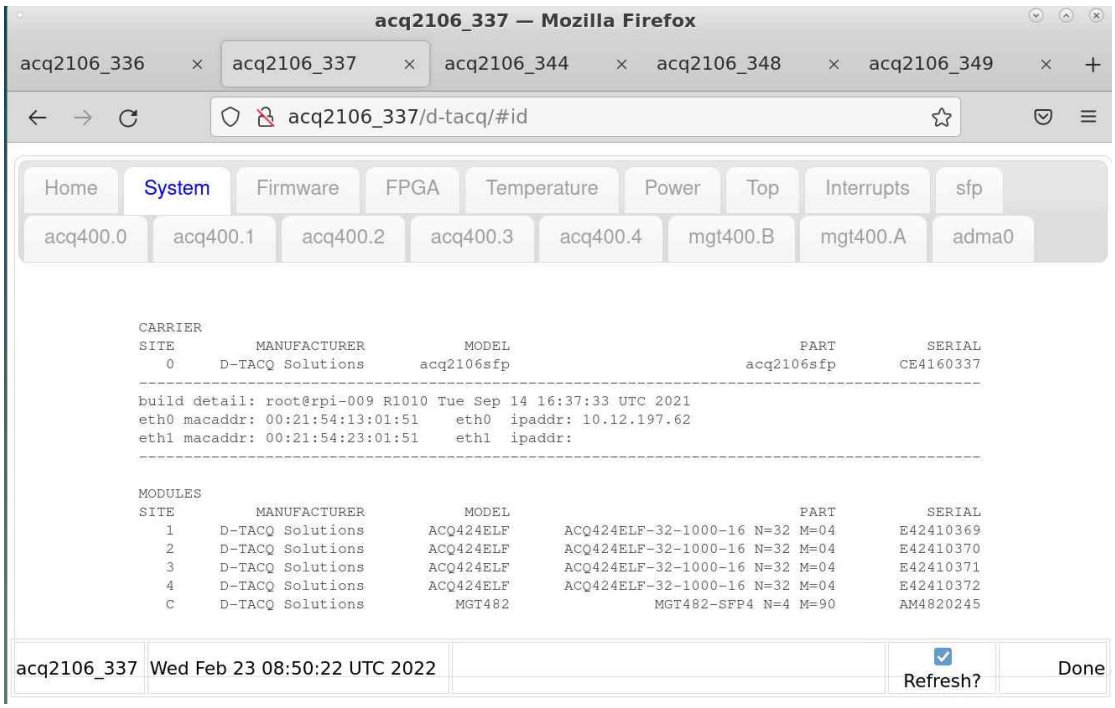
```
SUBSYSTEMS=="usb", ATTRS{serial}=="AU01BSMP",
SYMLINK+="tty_acq2106_349" RUN+="/usr/local/bin/make-tty-symlink"
SUBSYSTEMS=="usb", ATTRS{serial}=="AU01CGEH",
SYMLINK+="tty_acq2106_337" RUN+="/usr/local/bin/make-tty-symlink"
SUBSYSTEMS=="usb", ATTRS{serial}=="AU01BTR6",
SYMLINK+="tty_acq2106_348" RUN+="/usr/local/bin/make-tty-symlink"
SUBSYSTEMS=="usb", ATTRS{serial}=="AU01BU0E",
SYMLINK+="tty_acq2106_336" RUN+="/usr/local/bin/make-tty-symlink"
SUBSYSTEMS=="usb", ATTRS{serial}=="AU01BT2I",
SYMLINK+="tty_acq2106_344" RUN+="/usr/local/bin/make-tty-symlink"
```

- install on a terminal server (we recommend Raspberry PI)
- connect to consoles by name:
  - kk tty_acq2106_348

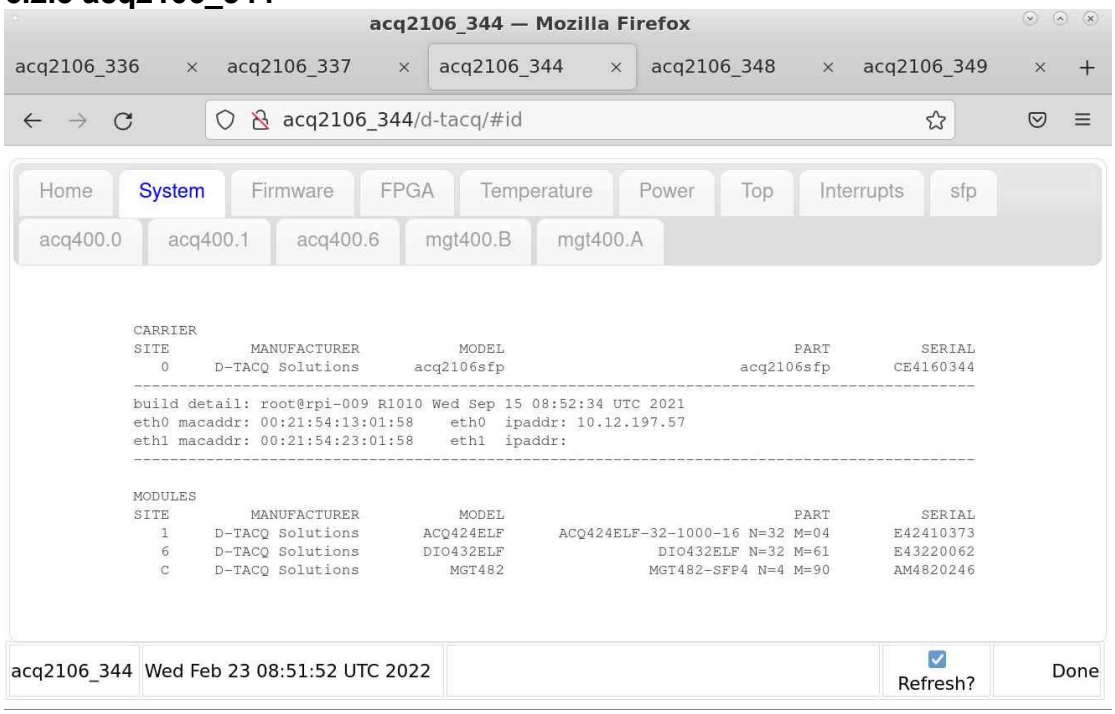
6.2 Configurations

6.2.1 acq2106_336

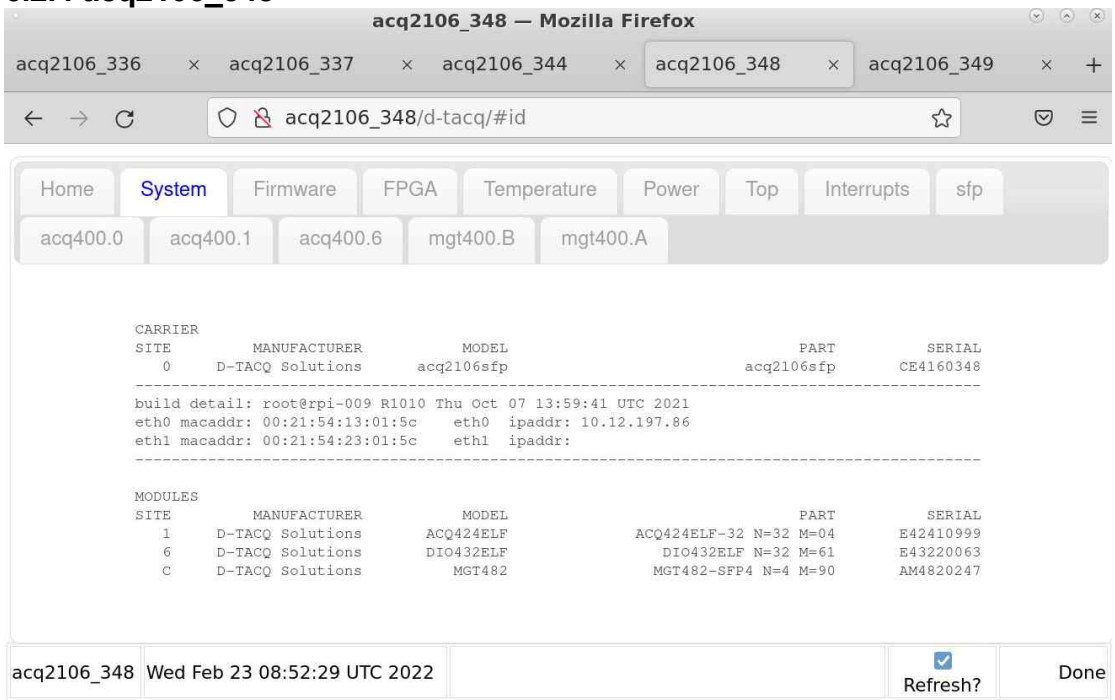
6.2.2 acq2106_337



6.2.3 acq2106_344

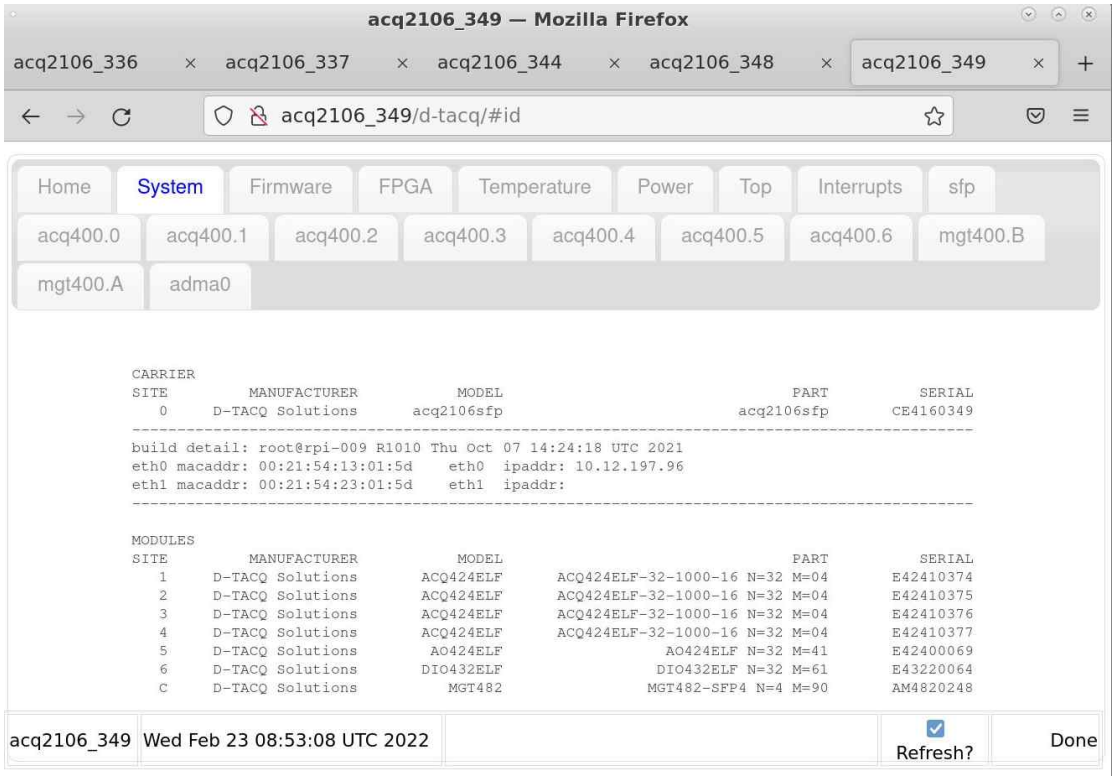


6.2.4 acq2106_348



NB: “Site 1 E4240999” is “FAKE” module, needed to run the unit in PCS mode. Customer may decide to retain or to remove.

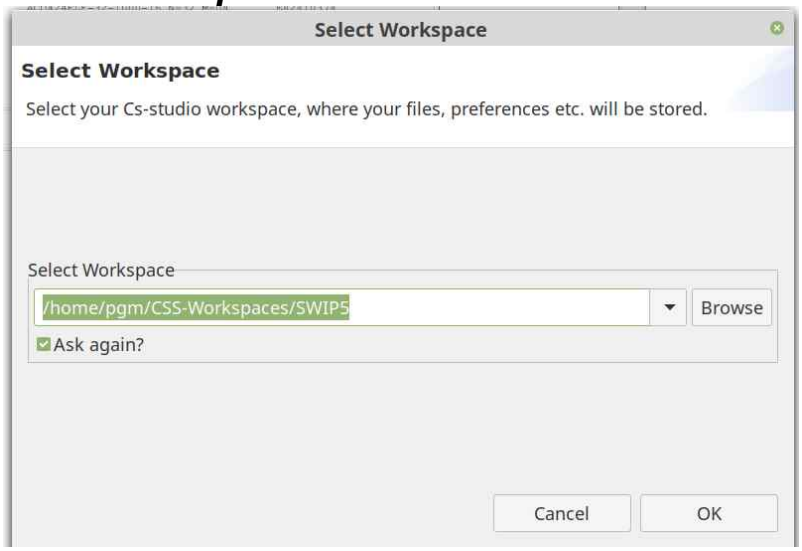
6.2.5 acq2106_349



7 APPENDIX: GUI Configuration.

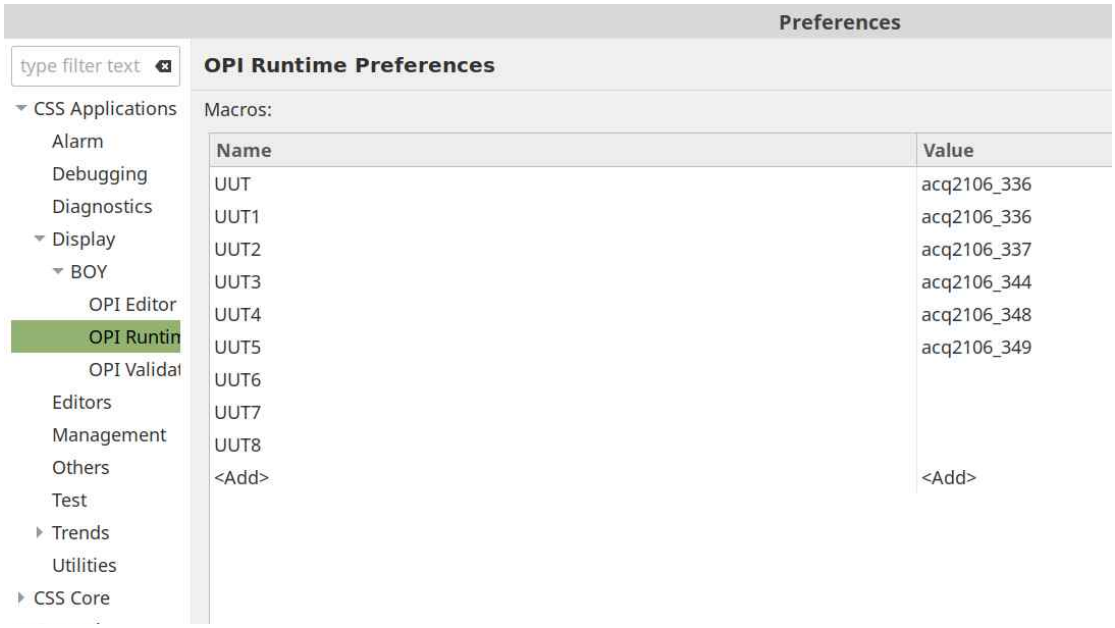
Although the GUI is not required for operation, it does provide an invaluable window in to the state of the system, and we do recommend setting it up, this is time well spent!

7.1 Dedicated Workspace

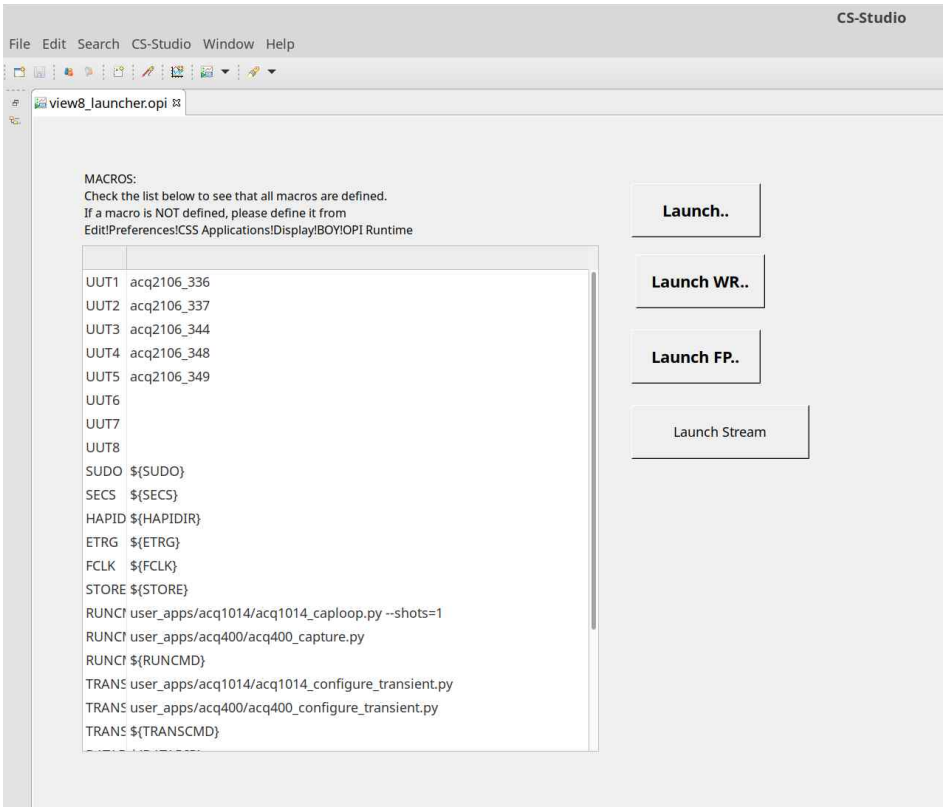


7.2 Launch OPI and Macros

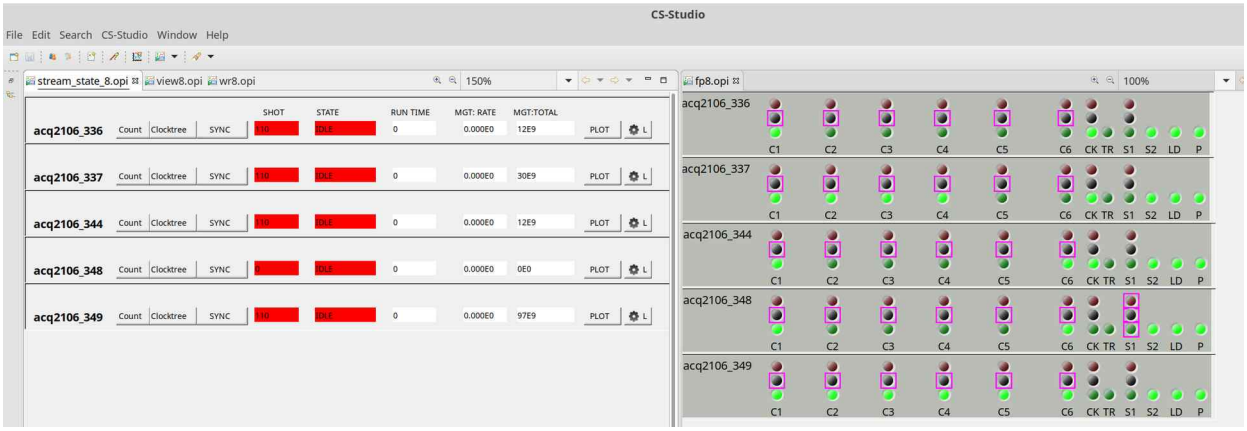
7.3 Macros



7.4 Launch OPI



7.5 Top Level Screen.



8 APPENDIX: Theory of LLC with ACQ2106

- LLC : Low Latency Control: data transfer mechanism optimised for Low Latency, used on PCS (Plasma Control System).
- ACQ2106 features two MGT comms links CA and CB, that provide a low latency fiber optic link back to our PCI-Express card AFHBA404.
- ACQ2106 runs two types of control, concurrently:
  - A9/Z, configured over Ethernet, this is the master control for the system.
  - CA, CB links: can run concurrently transferring some or all of the data setup up A9.
- ACQ2106 has 4 concurrent Data Paths, 3 IN, 1 OUT.
  - A9/Z Aggregator: collects data from a set of INPUT modules and delivers the data to ZYNQ DMA, and hence to local DRAM. The A9 Aggregator controls the system.
  - CA Aggregator: Comms A link can collect a subset of the A9 Aggregator
  - CB Aggregator: Comms B link can collect a subset of the A9 Aggregator
    - Typically in LLC, CA, CB collect ALL A9 aggregator data.
    - CA is the “normal path”, while CB is an alternate path for either data mirroring, or for hotswap between PC controllers.
  - Distributor: sends data to a set of OUTPUT modules.
    - The Distributor allows for ONE of 3 sources to be selected:
      - A9, CA, CB
      - Only CA, CB are relevant for LLC.
- Viewed from the HOST, for each ACQ2106, the HOST will
  - Receive VI Vector In, comprising all the INPUTS + SPAD
    - SPAD (scratchpad) is essentially another input, from the motherboard).
  - Transmit VO Vector OUT, comprising all the OUTPUTS
- In LLC mode, the link transfer is optimised for minimum latency, typically transferring ONE sample at a time in either direction.
- Data flow is entirely hardware mediated. In LLC, typically:
  - On the clock, one vector VI comprising all INPUTS is sent to HOST DRAM (PUSH)
  - On the clock, the ACQ2106 fetches VO, comprising all OUTPUT values, and updates the values. (PULL)
  - Typically, PULL is initiated exactly on the sample clock, while PUSH first has to wait for the ADC conversion related to the clock to complete.
  - In a “normal system”, operation is “N1 mode” (PULL picks up results from the previous cycle). It’s also possible to get faster results by retarding

the PULL, so that the system switches to “N Mode” where VO comprises results calculated from VI in the SAME clock cycle. In general, N1 mode works best for fast clock cycles (>50kHz), while N Mode can greatly reduce latency, especially for relatively low clock rates (<20 kHz). Please contact D-TACQ for a white paper with full details.

- SPAD, or metadata, up to 32 x u32 values at the tail of VI
  - SPAD[0] “Tlatch” : the sample number
  - SPAD[1] “Usecs” : elapsed time in USECS
  - SPAD[2..7] may be used for ad-hoc signaling
  - SPAD is required to VI to the nearest 64 byte boundary to maintain sample alignment in memory (and in time).

8.1 ACQPROC

With choice of 6 sites per ACQ2106, choice from 10 AI, AO, DIO modules, and possibility for 1..12 ACQ2106 units per HOST PC, LLC systems can quickly grow VERY COMPLEX!

The ACQPROC software framework aims to automate this complexity.
ACQPROC comprises:

- A configuration file CF to represent the system configuration
  - (json format)
  - NEW : this file can be created automatically by introspection on the HBA.
- A Control Program CP "ACQPROC" that will configure memory management based on the CF and run the realtime process.
  - CP will detect new VI data, and gather all VI's in the system
  - CP will scatter all the VI's into functional blocks and present this data to a Control Algorithm
    - ie separate vectors AI16, DI32, SP32
  - CP will gather a new composite VO from the Control Algorithm, and scatter the VO as a series of VO vectors, one per ACQ2106.
  - CP will publish a model of the composite memory for client programs (including the live PCS Control Algorithm to use)
- A Control Script CS that will configure all the ACQ2106 A9 side over ethernet. A series of HAPI python scripts autoconfigure from the CF to set up the following on each box:
  - sync_role (CLK and TRG)
  - Aggregator Settings (VI size and destination)
  - Distributor Settings (VO size and source)
  - Global Shot Control.
- After the shot, HAPI analysis scripts check in-shot data for various figures of merit, in particular data integrity and RT performance.

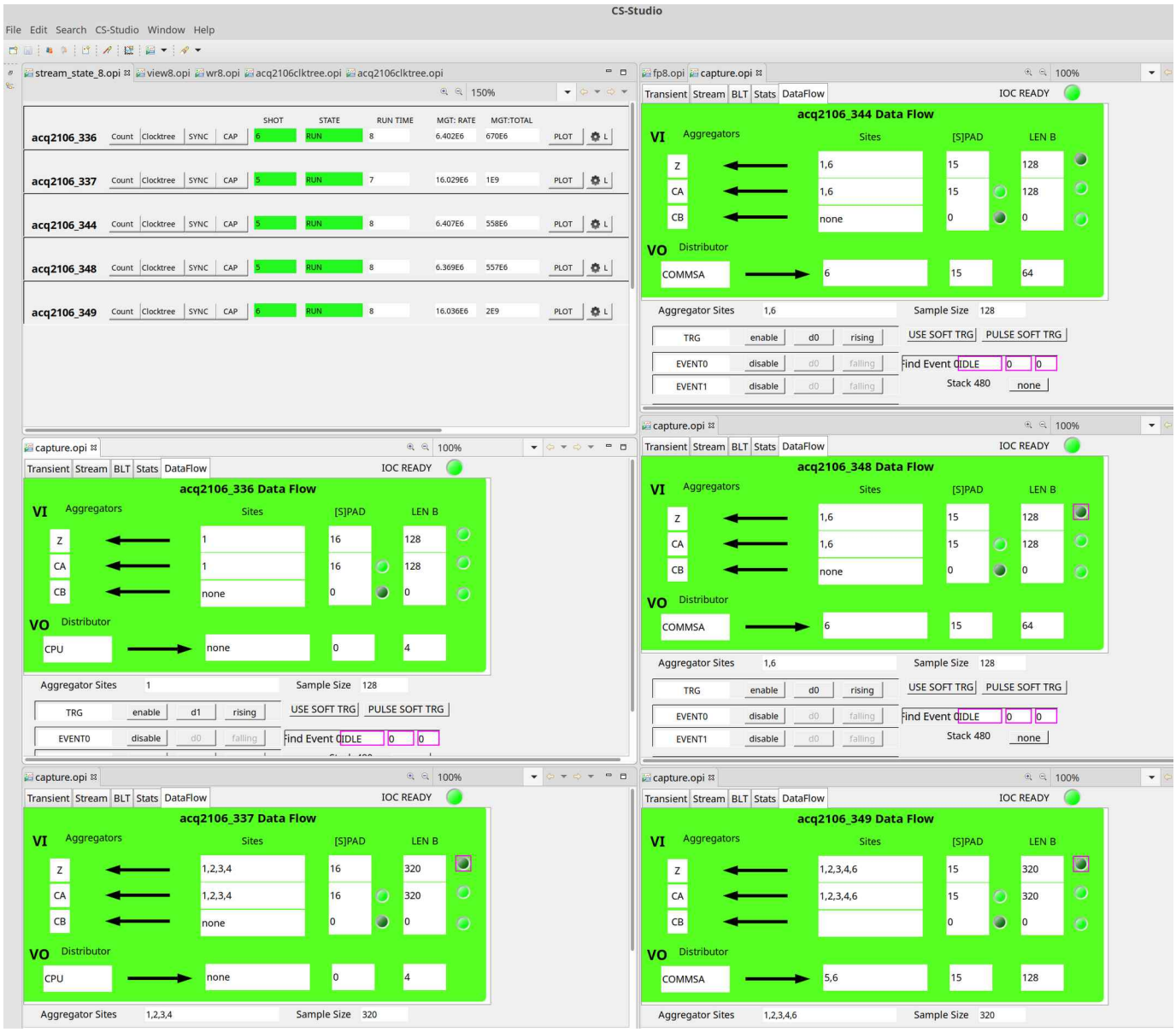
We appreciate that existing ACQ2106 PCS systems may already implement their own CP, but with the automated configuration, we suggest that the CS framework would still be useful to configure the system and to control the shot.

8.1.1 Sample config file “swip5_45.json”

```
{
  "AFHBA": {
    "UUT": [
      {
        "DEVNUM": 0,
        "name": "acq2106_336",
        "type": "pcs",
        "sync_role": "master",
        "COMMS": "A",
        "VI": {  "AI16": 32,   "SP32": 16  },
        "VO": {   "PAD32": 0 }
      },
      {
        "DEVNUM": 1,
        "name": "acq2106_349",
        "type": "pcs",
        "sync_role": "slave",
        "COMMS": "A",
        "VI": { "AI16": 128, "DI32": 1,  "SP32": 15 },
        "VO": {  "AO16": 32,  "DO32": 1, "AOSITES": [ 5 ],  "PAD32": 0
        }
      },
      {
        "DEVNUM": 4,
        "name": "acq2106_344",
        "type": "pcs",
        "sync_role": "slave",
        "COMMS": "A",
        "VI": {  "AI16": 32,  "DI32": 1,  "SP32": 15 },
        "VO": {  "DO32": 1,  "PAD32": 0 }
      },
      {
        "DEVNUM": 5,
        "name": "acq2106_337",
        "type": "pcs",
        "sync_role": "slave",
        "COMMS": "A",
        "VI": {"AI16": 128, "SP32": 16 },
        "VO": {  "PAD32": 0  }
      },
      {
        "DEVNUM": 6,
        "name": "acq2106_348",
        "type": "pcs",
        "sync_role": "slave",
        "COMMS": "A",
        "VI": { "AI16": 32,  "DI32": 1, "SP32": 15 },
        "VO": { "DO32": 1,  "PAD32": 0 }
      }
    ]
  }
}
```

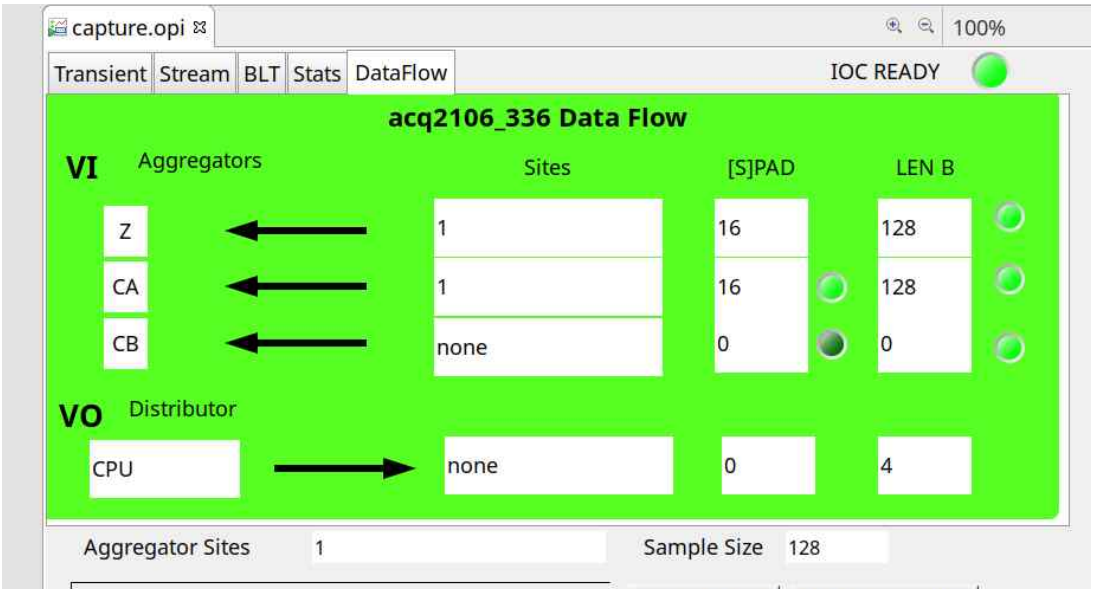
8.2 3 Aggregators, 1 Distributor

Best reviewed with reference to cs-studio mimics:



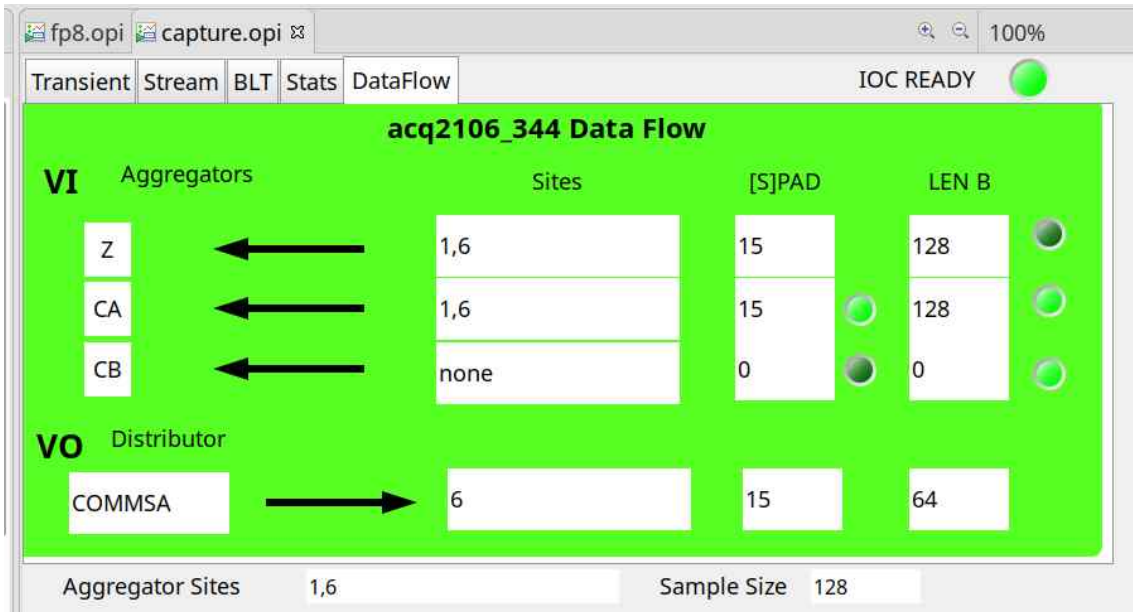
Here we have a shot in progress, with all the Data Flow mimics showing.

8.2.1 acq2106_336



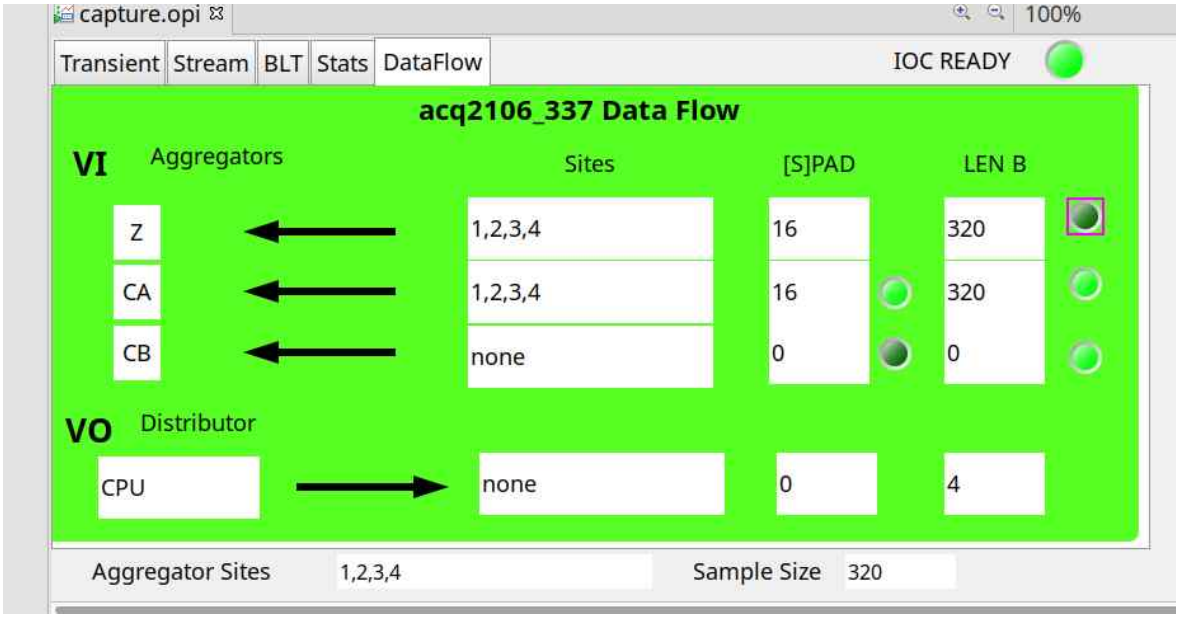
- AI16 = 32, SP32 = 16 : A9/Z aggregator, CA aggregator: 128 bytes
- No outputs, distributor source CPU.

8.2.2 acq2106_344



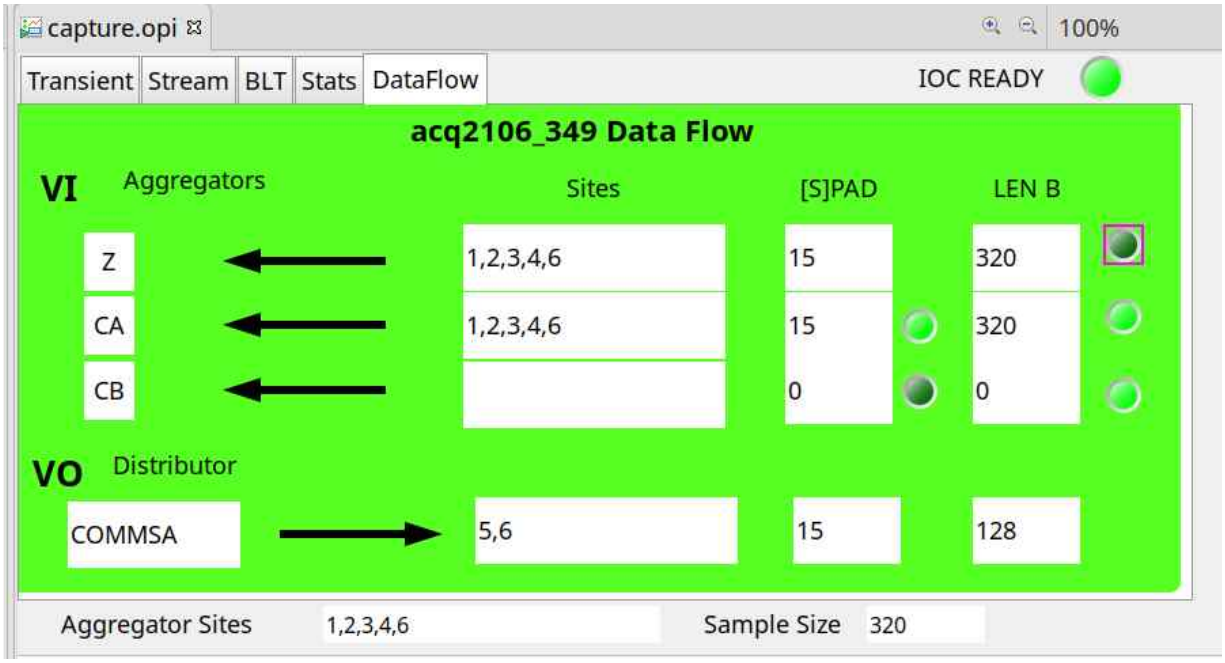
- AI16=32, DI32=1, SP32=15, Z and CA aggregator: 128 bytes
- DO32=1, PAD32=15, distributor source CA, len = 64 bytes

8.2.3 acq2106_337



- AI16=128, SP32=16 Z, CA aggregator len 320 Bytes
- No VO, src = CPU.

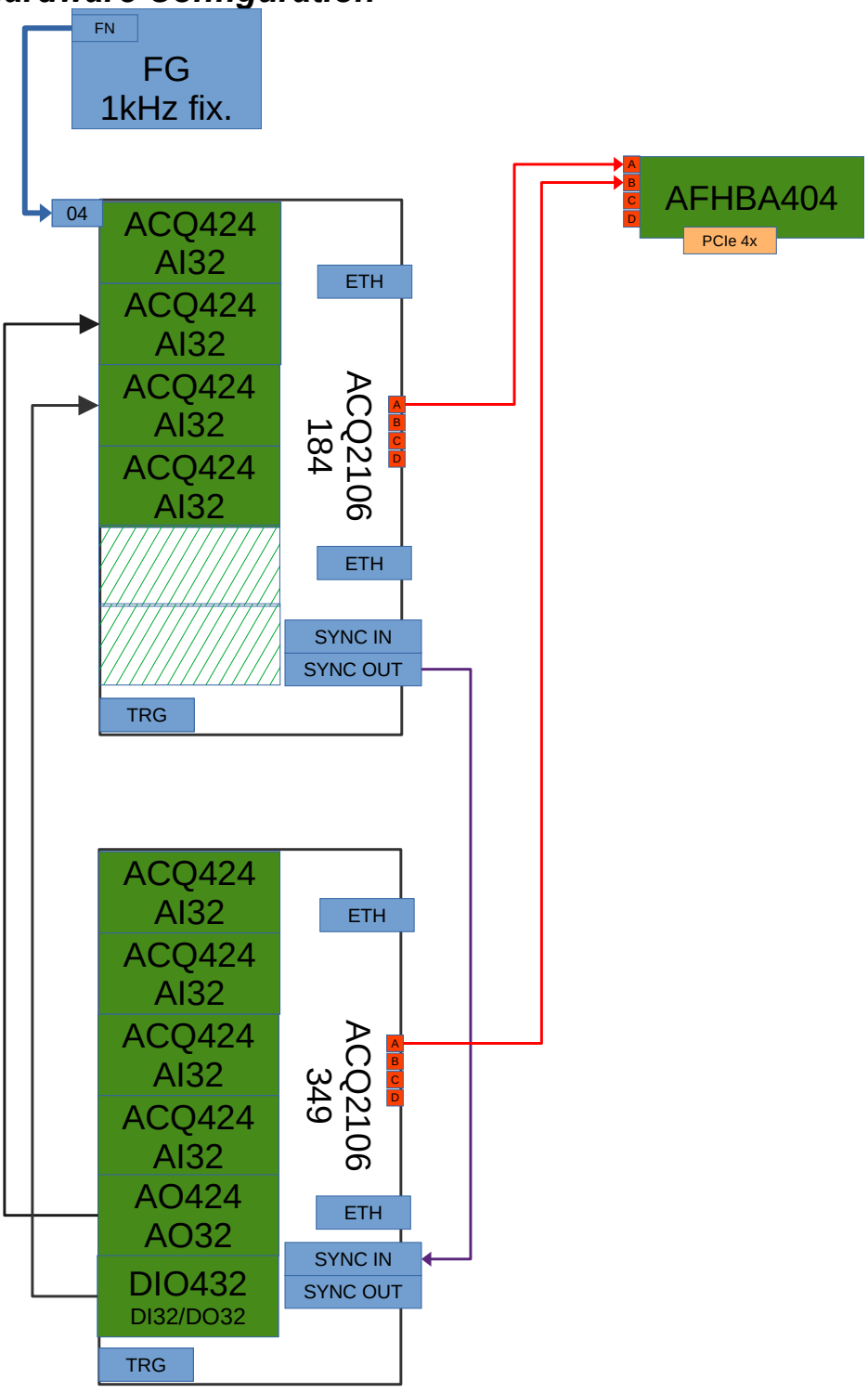
8.2.4 acq2106_349



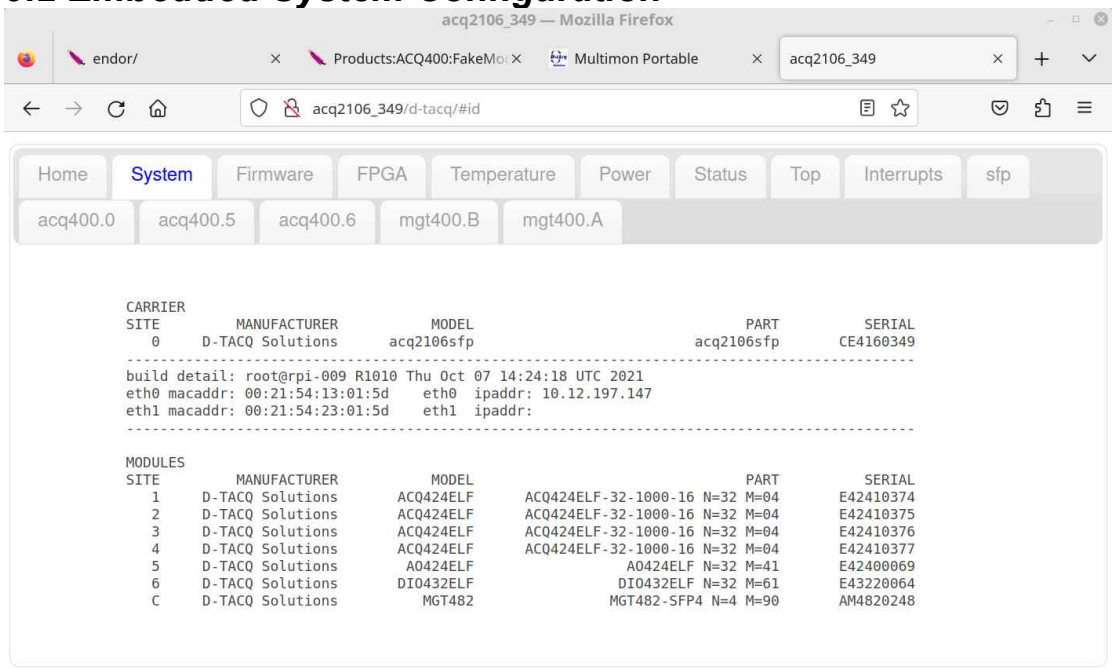
- AI16=128, DI32=1, SP32=15, Z, CA aggregator len 320 Bytes.
- AO32=1 DO32=1, PAD32=15, Distributor src=CA, len = 128 Bytes

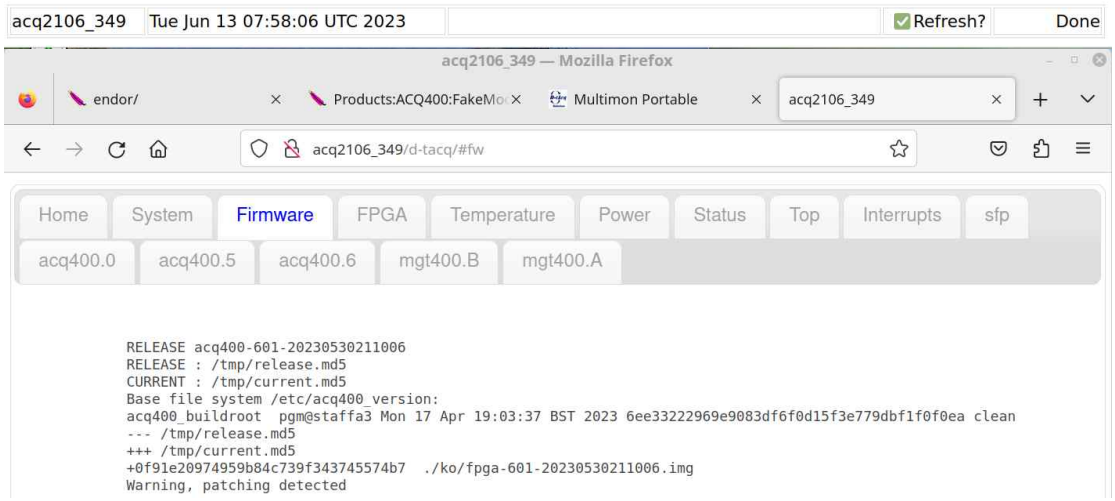
9 ACQPROC ex: AI128+AI128,AO32,DO32

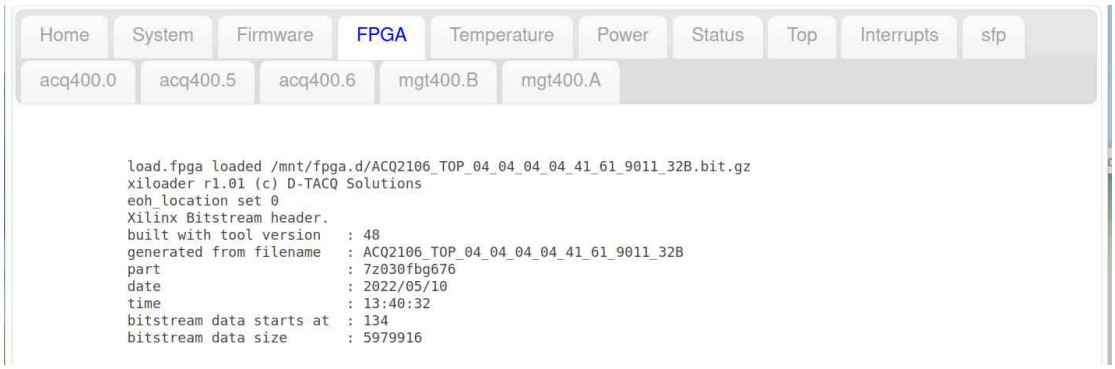
9.1 Hardware Configuration



9.2 Embedded System Configuration







9.3 Host Software Configuration.

9.3.1 AFHBA404 Version:

https://github.com/D-TACQ/AFHBA404/releases/tag/v2.7 or later.

9.3.2 Create System Model

```
[dt100@kamino AFHBA404]$ ./HAPI/lsafhba.py
0 HostComms(host='kamino.d-tacq', dev='0', uut='acq2106_184', cx='A')
1 HostComms(host='kamino.d-tacq', dev='1', uut='acq2106_349', cx='A')
2 HostComms(host='kamino.d-tacq', dev='2', uut='acq2106_176', cx='A')
```

Our UUTS are acq2106_184 and acq2106_349, on LPORTS 0,1:

```
[dt100@kamino AFHBA404]$ ./HAPI/lsafhba.py \
-l 0,1 --byte_is_output=1,1,1,1 \
-m SWIP_acq2106_184_128_acq2106_349_128_32_32.json
0 HostComms(host='kamino.d-tacq', dev='0', uut='acq2106_184', cx='A')
1 HostComms(host='kamino.d-tacq', dev='1', uut='acq2106_349', cx='A')
```

Creates the system model file

SWIP_acq2106_184_128_acq2106_349_128_32_32.json:

```
{
    "AFHBA": {
        "UUT": [
            {
                "DEVNUM": 0,
                "name": "acq2106_184",
                "type": "pcs",
                "sync_role": "master",
                "COMMS": "A",
                "VI": {
                    "AI16": 128,
                    "SP32": 16,
                    "AISITES": [
                        1,
                        2,
                        3,
                        4
                    ],
                    "NXI": 4
                },
                "VO": {
                    "NXO": 0
                }
            },
            {
                "DEVNUM": 1,
                "name": "acq2106_349",
                "type": "pcs",
                "sync_role": "slave",
                "COMMS": "A",
                "VI": {
                    "AI16": 128,
                    "DI32": 1,
                    "SP32": 15,
                    "AISITES": [
                        1,
                        2,
                        3,
                        4
                    ],
                    "DIOSITES": [
                        6
                    ],
                    "NXI": 5
                },
                "VO": {
                    "AO16": 32,
                    "DO32": 1,
                    "AOSITES": [
                        5
                    ],
                    "DO_BYTE_IS_OUTPUT": [
                        "1,1,1,1"
                    ],
                    "DIOSITES": [
                        6
                    ],
                    "NXO": 2
                }
            }
        ]
    }
}
```

9.3.3 Run the Shot

```
[dt100@kamino AFHBA404]$ SINGLE_THREAD_CONTROL=control_dup1=3 \
./scripts/acqproc_multi.sh \
SWIP_acq2106_184_128_acq2106_349_128_32_32.json
```

- Transcript

```
[dt100@kamino AFHBA404]$ SINGLE_THREAD_CONTROL=control_dup1=3  ./scripts/acqproc_multi.sh
SWIP_acq2106_184_128_acq2106_349_128_32_32.json
UUT1 acq2106_184
UUT2 acq2106_349
UUTS acq2106_184 acq2106_349
DEVMAX 3
python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/sync_role.py --toprole=master --
fclk=50000 acq2106_184 acq2106_349

python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/sync_role.py done 0
python3 scripts/llc-config-utility.py SWIP_acq2106_184_128_acq2106_349_128_32_32.json
Starting llc-config-utility
CONFIGURING acq2106_184
Link Good
config_VI
acq2106_184 Aggregator settings: sites=1,2,3,4 (spad=16) (comms=A)
CONFIGURING acq2106_349
Link Good
DO_BYTE_IS_OUTPUT ['1,1,1,1']
DO_BYTE_IS_OUTPUT ['1,1,1,1']
config_VI
acq2106_349 Aggregator settings: sites=1,2,3,4,6 (spad=15) (comms=A)
config_VO
acq2106_349 Distributor settings: sites=5,6 pad=15 comms=A on
python3 scripts/llc-config-utility.py done 0
control_program vanilla
python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/acq400_streamtonowhere.py --
samples=1000000 acq2106_184 acq2106_349
nsamples set 1000000
HBA1 VI:640 VO:68 devs=0,1
                [0] dev:0 acq2106_184 VI:320 VO:0 Offset of SPAD IN VI :256
 System Interface Indices 0,0
                [1] dev:1 acq2106_349 VI:320 VO:68 Offset of SPAD IN VI :260
 System Interface Indices 128,16
Arming systems now - please wait. Do not trigger yet.
All UUTs are armed and ready for trigger.
Streamed 0 of 1000000 samples
Streamed 106462 of 1000000 samples
Streamed 156486 of 1000000 samples
...
Streamed 856820 of 1000000 samples
Streamed 906844 of 1000000 samples
stored acq2106_184_VI.dat, len=319999680
stored acq2106_349_VI.dat, len=319999680
stored acq2106_349_VO.dat, len=67999932
Control Program Finished
Streamed 956868 of 1000000 samples

Stream finished.
python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/acq400_streamtonowhere.py done 0

Running analysis now.
---------------------
python3 scripts/acqproc_analysis.py --ones=1 --json=1 --json_src=./runtime.json
--src=/home/dt100/PROJECTS/AFHBA404/
TKAgg not available for matplot
Running analysis for UUT: acq2106_184
show SPAD: hexdump -ve '128/2 "%04x," 16/4 "%08x," "\n"' acq2106_184_VI.dat  | cut -d, -f129-136 |
head -n 4
00000001,00000005,00000000,00000000,00000000,00000000,a6666184,a7777184
```

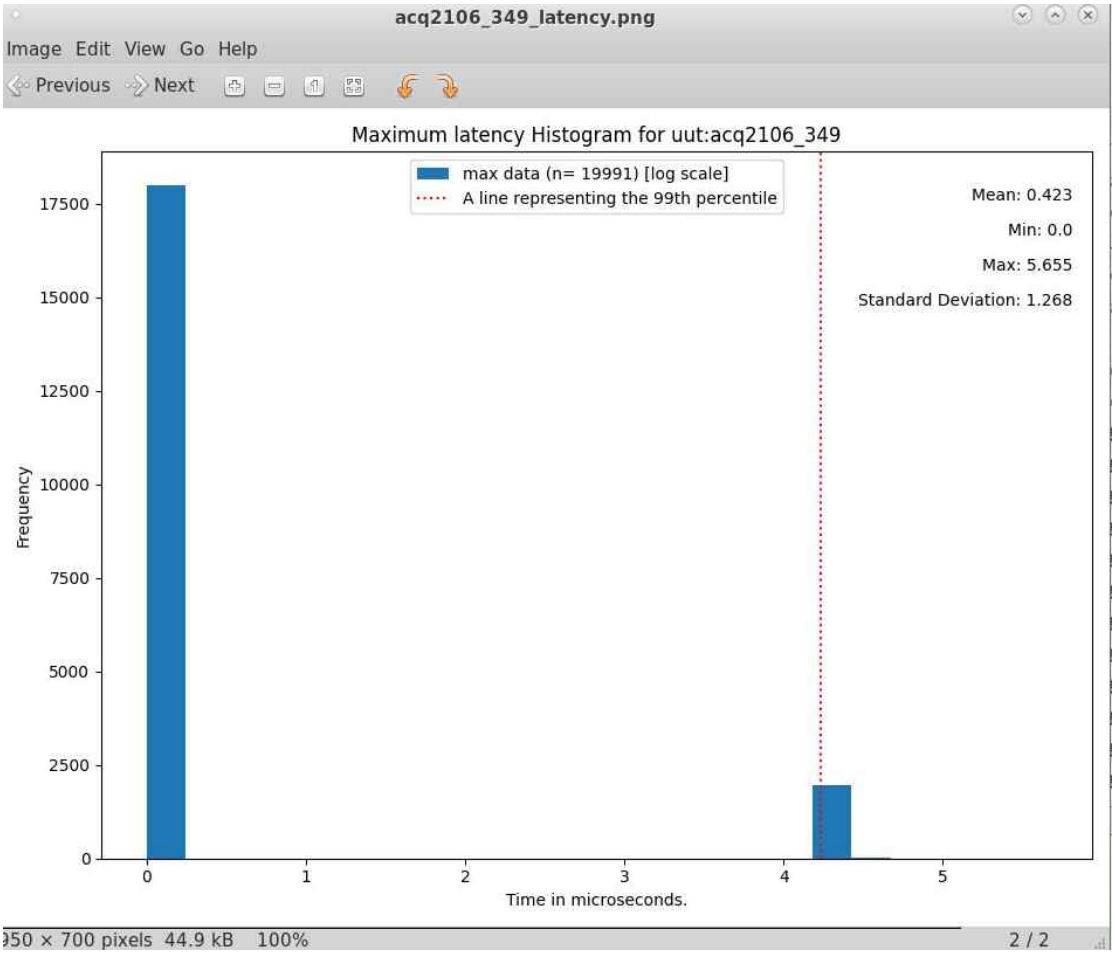
```
00000002,00000019,00000000,00000000,00000000,00000000,a6666184,a7777184
00000003,0000002d,00000000,00000000,00000000,00000000,a6666184,a7777184
00000004,00000041,00000000,00000000,00000000,00000000,a6666184,a7777184
show whole VI like this:
hexdump -ve '128/2 "%04x," 16/4 "%08x," "\n"' acq2106_184_VI.dat
/home/dt100/PROJECTS/AFHBA404//acq2106_184_VI.dat
tlatch = data[64::80]
Finished collecting data
T_LATCH differences:  1 , happened:  375010  times
T_LATCH differences:  2 , happened:  0  times
T_LATCH differences:  3 , happened:  0  times
qt.qpa.xcb: X server does not support XInput 2
Cannot run latency analysis for this UUT.
Running analysis for UUT: acq2106_349
show SPAD: hexdump -ve '128/2 "%04x," 1/4 "%08x," 15/4 "%08x," "\n"' acq2106_349_VI.dat  | cut -d, -
f130-137 | head -n 4
00000001,00000001,00000000,00000000,00000118,0000ffff,0000591c,a7777349
00000002,00000015,00000000,00000000,00000118,0000ffff,0000591c,a7777349
00000003,00000029,00000000,00000000,00000118,0000ffff,0000591c,a7777349
00000004,0000003d,00000000,00000000,00000118,0000ffff,0000591c,a7777349
show whole VI like this:
hexdump -ve '128/2 "%04x," 1/4 "%08x," 15/4 "%08x," "\n"' acq2106_349_VI.dat
/home/dt100/PROJECTS/AFHBA404//acq2106_349_VI.dat
tlatch = data[65::80]
Finished collecting data
T_LATCH differences:  1 , happened:  375010  times
T_LATCH differences:  2 , happened:  0  times
T_LATCH differences:  3 , happened:  0  times
(159999840,)
data end: 129
spad_len:  15
160
len diffs = 19991
max data = [0. 0. 0. ... 0. 0. 0.]
latency plot saved as acq2106_349_latency.png
```

- Outputs

```
[dt100@kamino AFHBA404]$ ls -lt | head
total 697548
-rw-rw-r--. 1 dt100 dt100     44853 Jun 13 09:10 acq2106_349_latency.png
-rw-r--r--. 1 root  root   67999932 Jun 13 09:10 acq2106_349_VO.dat
-rw-r--r--. 1 root  root  319999680 Jun 13 09:10 acq2106_349_VI.dat
-rw-r--r--. 1 root  root  319999680 Jun 13 09:10 acq2106_184_VI.dat
-rw-rw-r--. 1 dt100 dt100      4056 Jun 13 09:09 runtime.json
```

- Graph Showing Latency
- RAW VO data for 349, RAW VI data for 349
- RAW VI data for 184
- runtime.json: detail view of ACQPROC memory interface.

9.3.3.1 Latency Graph



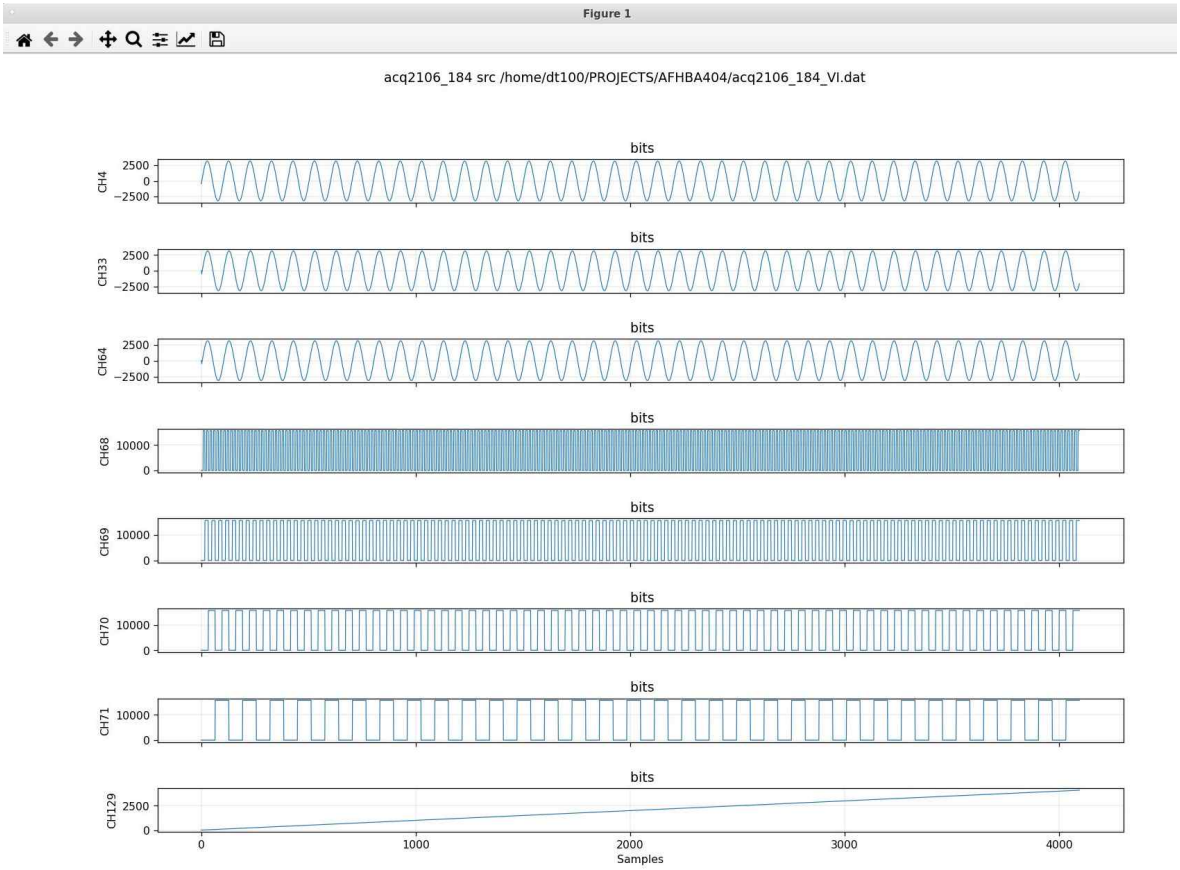
We have one latency failure. ACQPROC was NOT running with any realtime settings.

9.3.3.2 Data Flows

9.3.4 View Raw Data

```
dt100@kamino acq400_hapi]$ ./user_apps/analysis/host_demux.py \
--src=/home/dt100/PROJECTS/AFHBA404/acq2106_184_VI.dat \
--pchan=4,33,64,68,69,70,71,129  --plot_mpl=1 --pses=0:4096:1 acq2106_184
```

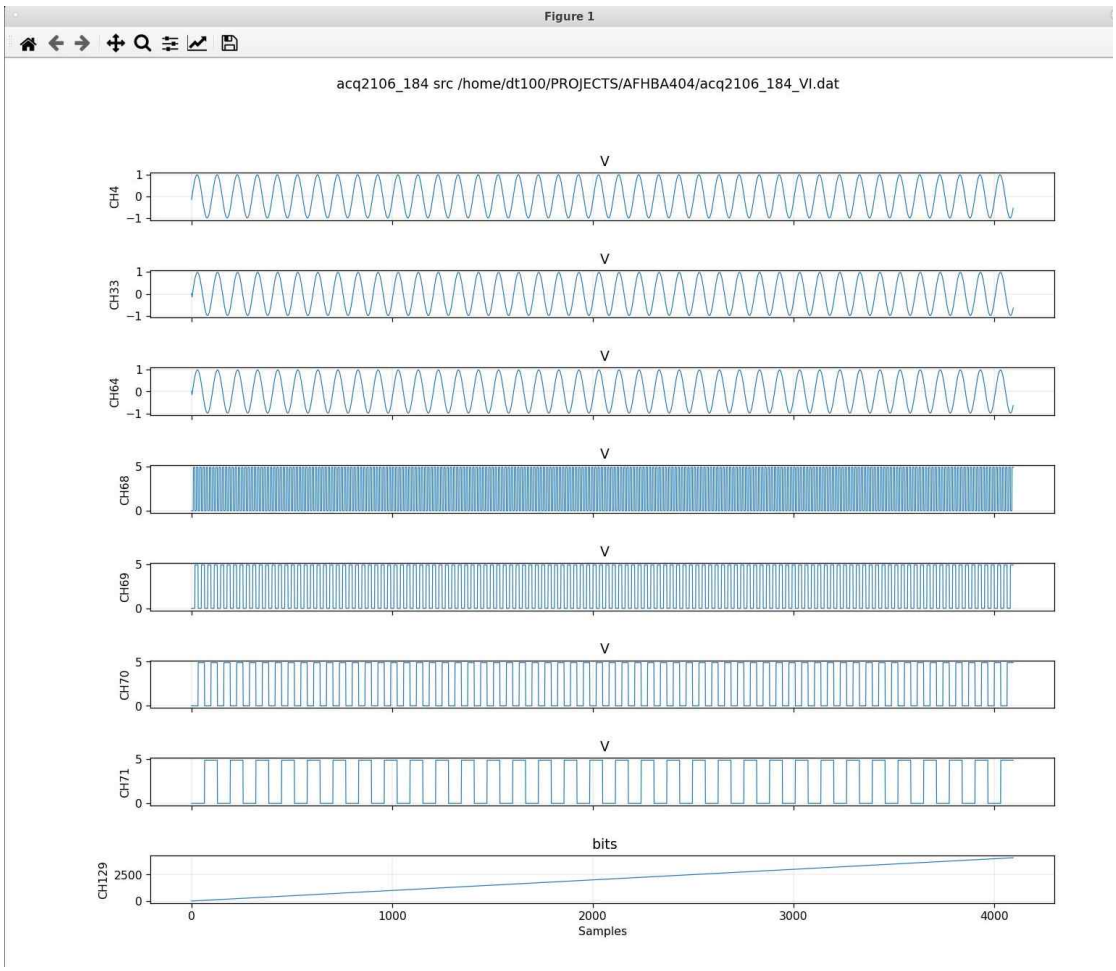
- CH04: Input from FG to AI32.1
- CH33,64: Input from AO32 to AI32.2
- CH65..96: Input from DO32 to AI32.3 # show slower signals
- CH129 : SPAD[0] Sample Count or “TLATCH”



9.3.5 View EGU (Volts) Data

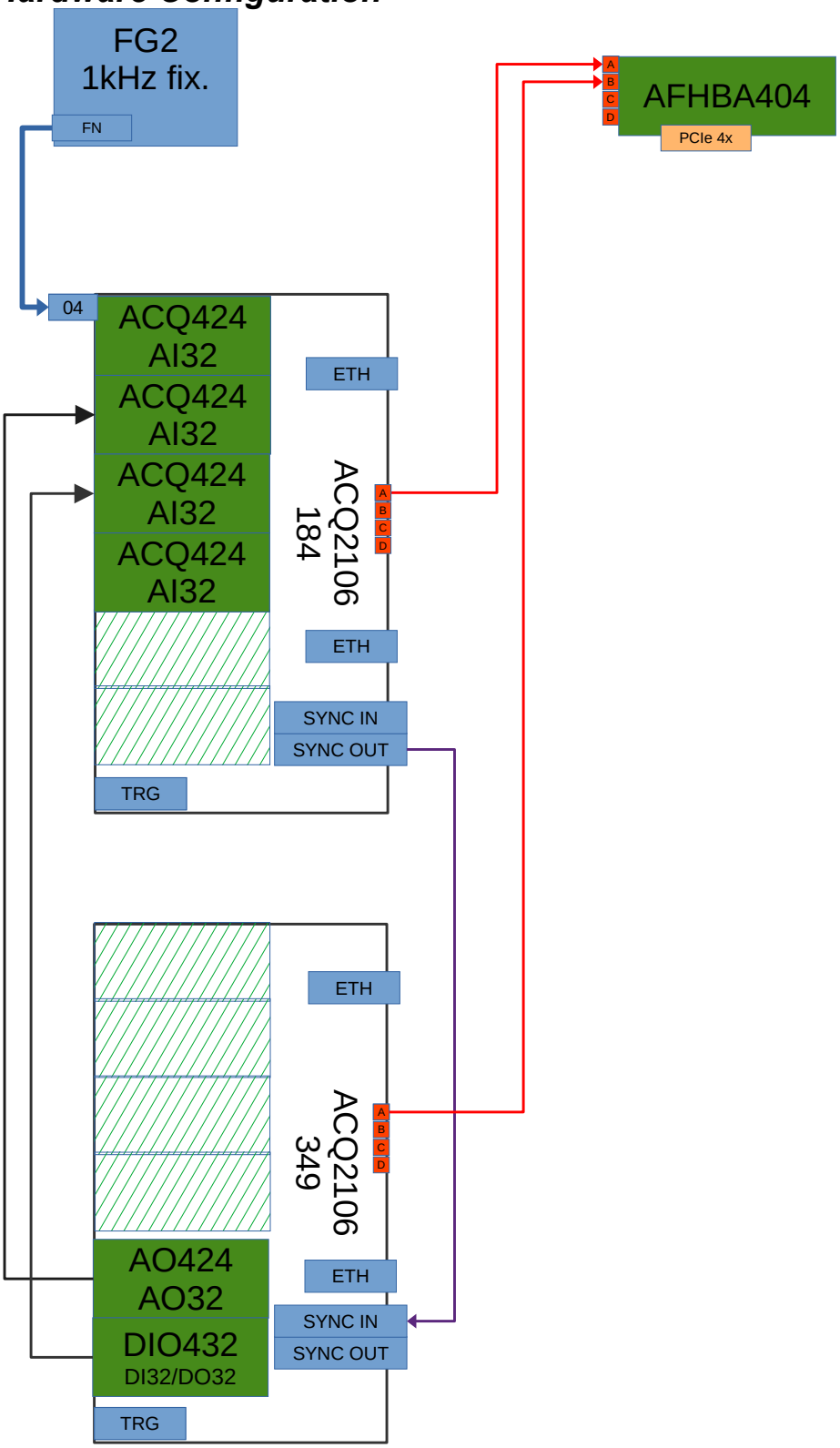
```
dt100@kamino acq400_hapi]$ ./user_apps/analysis/host_demux.py \
--src=/home/dt100/PROJECTS/AFHBA404/acq2106_184_VI.dat --egu=1 \
--pchan=4,33,64,68,69,70,71,129  --plot_mpl=1 --pses=0:4096:1 acq2106_184
```

- CH04: Input from FG to AI32.1
- CH33,64: Input from AO32 to AI32.2
- CH65..96: Input from DO32 to AI32.3 # show slower signals
- CH129 : SPAD[0] Sample Count or “TLATCH”

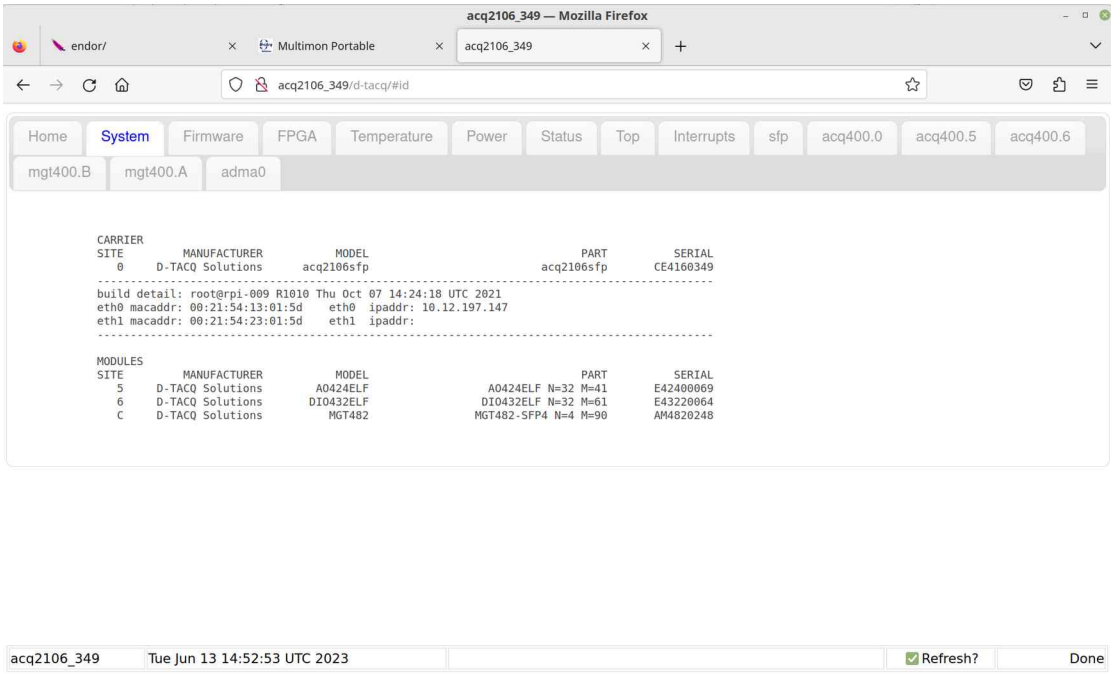


10 ACQPROC ex: AI128+AO32,DO32

10.1 Hardware Configuration



10.2 Embedded System Configuration



10.3 Host Software Configuration.

10.3.1 AFHBA404 Version:

https://github.com/D-TACQ/AFHBA404/releases/tag/v2.7 or later.

10.3.2 Create System Model

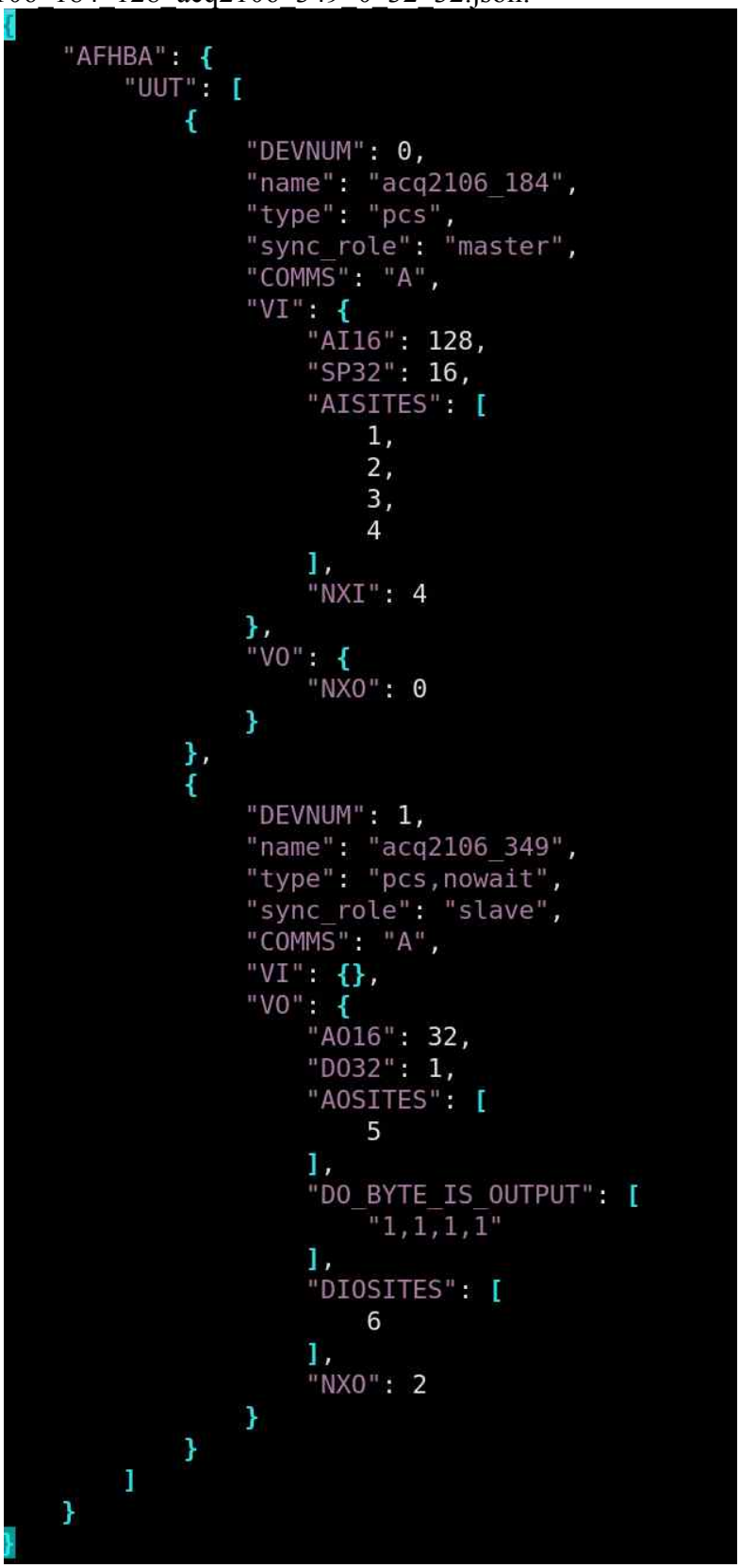
```
[dt100@kamino AFHBA404]$ ./HAPI/lsafhba.py
0 HostComms(host='kamino.d-tacq', dev='0', uut='acq2106_184', cx='A')
1 HostComms(host='kamino.d-tacq', dev='1', uut='acq2106_349', cx='A')
2 HostComms(host='kamino.d-tacq', dev='2', uut='acq2106_176', cx='A')
```

Our UUTS are acq2106_184 and acq2106_349, on LPORTS 0,1:

```
[dt100@kamino AFHBA404]$ ./HAPI/lsafhba.py -l 0,1 \
--byte_is_output=1,1,1,1 \
-s SWIP_acq2106_184_128_acq2106_349_0_32_32.json
0 HostComms(host='kamino.d-tacq', dev='0', uut='acq2106_184', cx='A')
1 HostComms(host='kamino.d-tacq', dev='1', uut='acq2106_349', cx='A')
```

Creates the system model file
SWIP_acq2106_184_128_acq2106_349_0_32_32.json:

SWIP_acq2106_184_128_acq2106_349_0_32_32.json:



```
{
    "AFHBA": {
        "UUT": [
            {
                "DEVNUM": 0,
                "name": "acq2106_184",
                "type": "pcs",
                "sync_role": "master",
                "COMMS": "A",
                "VI": {
                    "AI16": 128,
                    "SP32": 16,
                    "AISITES": [
                        1,
                        2,
                        3,
                        4
                    ],
                    "NXI": 4
                },
                "VO": {
                    "NXO": 0
                }
            },
            {
                "DEVNUM": 1,
                "name": "acq2106_349",
                "type": "pcs,nowait",
                "sync_role": "slave",
                "COMMS": "A",
                "VI": {},
                "VO": {
                    "AO16": 32,
                    "DO32": 1,
                    "AOSITES": [
                        5
                    ],
                    "DO_BYTE_IS_OUTPUT": [
                        "1,1,1,1"
                    ],
                    "DIOSITES": [
                        6
                    ],
                    "NXO": 2
                }
            }
        ]
    }
}
```

10.3.3 Run The Shot

```
[dt100@kamino AFHBA404]$ SINGLE_THREAD_CONTROL=control_dup1=3
./scripts/acqproc_multi.sh SWIP_acq2106_184_128_acq2106_349_0_32_32.json
```

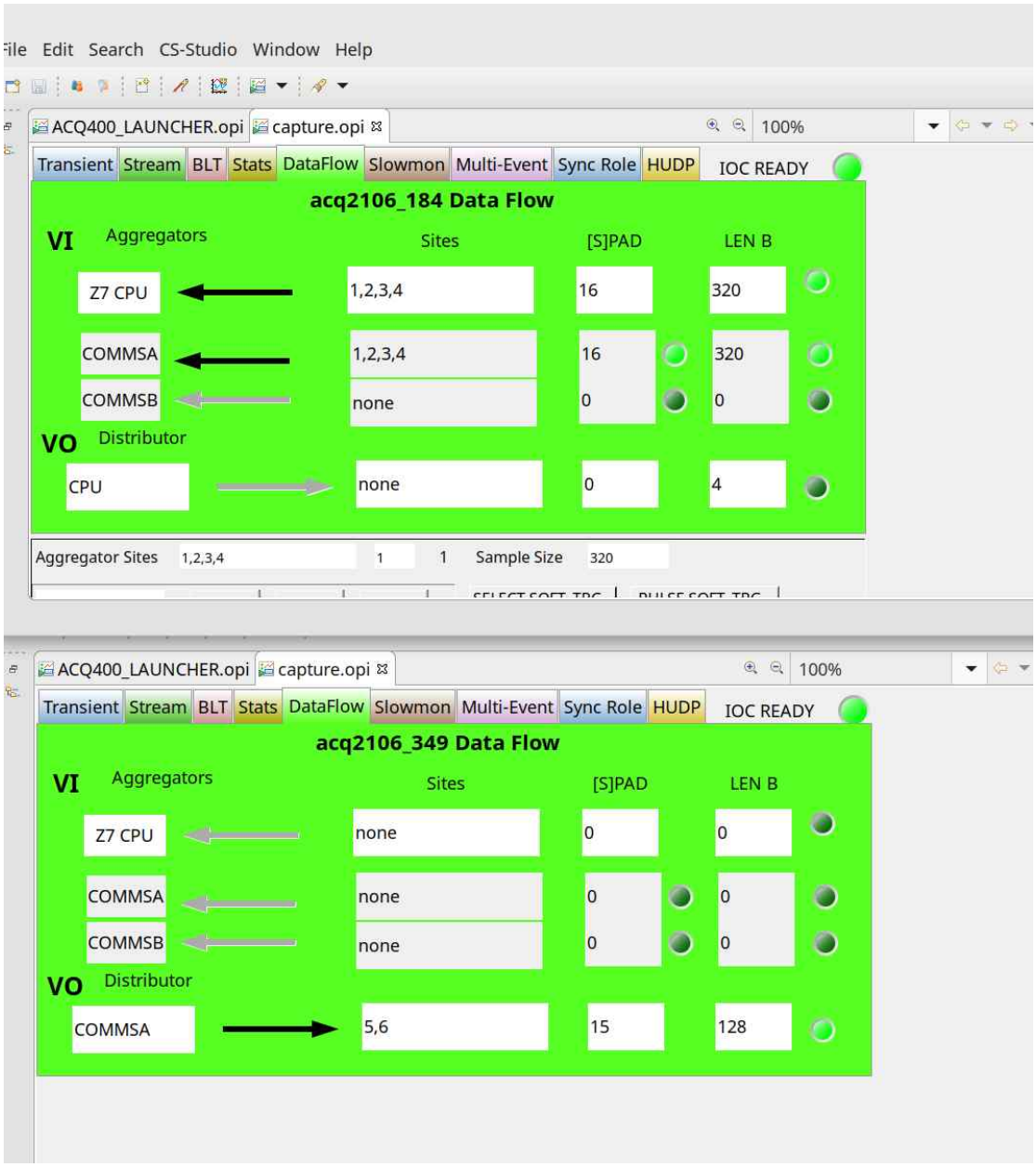
- Transcript

```
[dt100@kamino AFHBA404]$ SINGLE_THREAD_CONTROL=control_dup1=3  ./scripts/acqproc_multi.sh
SWIP_acq2106_184_128_acq2106_349_0_32_32.json
UUT1 acq2106_184
UUT2 acq2106_349
UUTS acq2106_184 acq2106_349
DEVMAX 3
python3 scripts/llc-config-utility.py SWIP_acq2106_184_128_acq2106_349_0_32_32.json
Starting llc-config-utility
CONFIGURING acq2106_184
Link Good
config_VI
acq2106_184 Aggregator settings: sites=1,2,3,4 (spad=16) (comms=A)
CONFIGURING acq2106_349
Link Good
DO_BYTE_IS_OUTPUT ['1,1,1,1']
json_override_actual( acq2106_349 [] VI AISITES)
json_override_actual( acq2106_349 [] VI DIOSITES)
DO_BYTE_IS_OUTPUT ['1,1,1,1']
config_VO
acq2106_349 Distributor settings: sites=5,6 pad=15 comms=A on
python3 scripts/llc-config-utility.py done 0
python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/sync_role.py --toprole=master --
fclk=50000 acq2106_184 acq2106_349

python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/sync_role.py done 0
control_program vanilla
python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/acq400_streamtonowhere.py --
samples=1000000 acq2106_184
nsamples set 1000000
NOTICE: port 0 set nowait
HBA1 VI:320 VO:68 devs=0,1
                [0] dev:0 acq2106_184 VI:320 VO:0 Offset of SPAD IN VI :256
 System Interface Indices 0,0
                [1] dev:1 acq2106_349 VI:0 VO:68 Offset of SPAD IN VI :0
 System Interface Indices 128,16
Arming systems now - please wait. Do not trigger yet.
All UUTs are armed and ready for trigger.
Streamed 0 of 1000000 samples
Streamed 100694 of 1000000 samples
...
Streamed 901076 of 1000000 samples
stored acq2106_184_VI.dat, len=319999680
stored acq2106_349_VO.dat, len=67999932
Control Program Finished
Streamed 951100 of 1000000 samples

Stream finished.
python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/acq400_streamtonowhere.py done 0

Running analysis now.
--------------------
python3 scripts/acqproc_analysis.py --ones=1 --json=1 --json_src=./runtime.json
--src=/home/dt100/PROJECTS/AFHBA404/
TKAgg not available for matplot
Running analysis for UUT: acq2106_184
show SPAD: hexdump -ve '128/2 "%04x," 16/4 "%08x," "\n"' acq2106_184_VI.dat  | cut -d, -f129-136 |
head -n 4
00000001,00000010,00000000,00000000,00000000,00000000,a6666184,a7777184
00000002,00000024,00000000,00000000,00000000,00000000,a6666184,a7777184
00000003,00000038,00000000,00000000,00000000,00000000,a6666184,a7777184
00000004,0000004c,00000000,00000000,00000000,00000000,a6666184,a7777184
show whole VI like this:
hexdump -ve '128/2 "%04x," 16/4 "%08x," "\n"' acq2106_184_VI.dat
/home/dt100/PROJECTS/AFHBA404//acq2106_184_VI.dat
```

```
tlatch = data[64::80]
Finished collecting data
T_LATCH differences:  1 , happened:  375009  times
T_LATCH differences:  2 , happened:  0  times
T_LATCH differences:  3 , happened:  0  times
```

Latency Graph is not available for box without VI and VO.

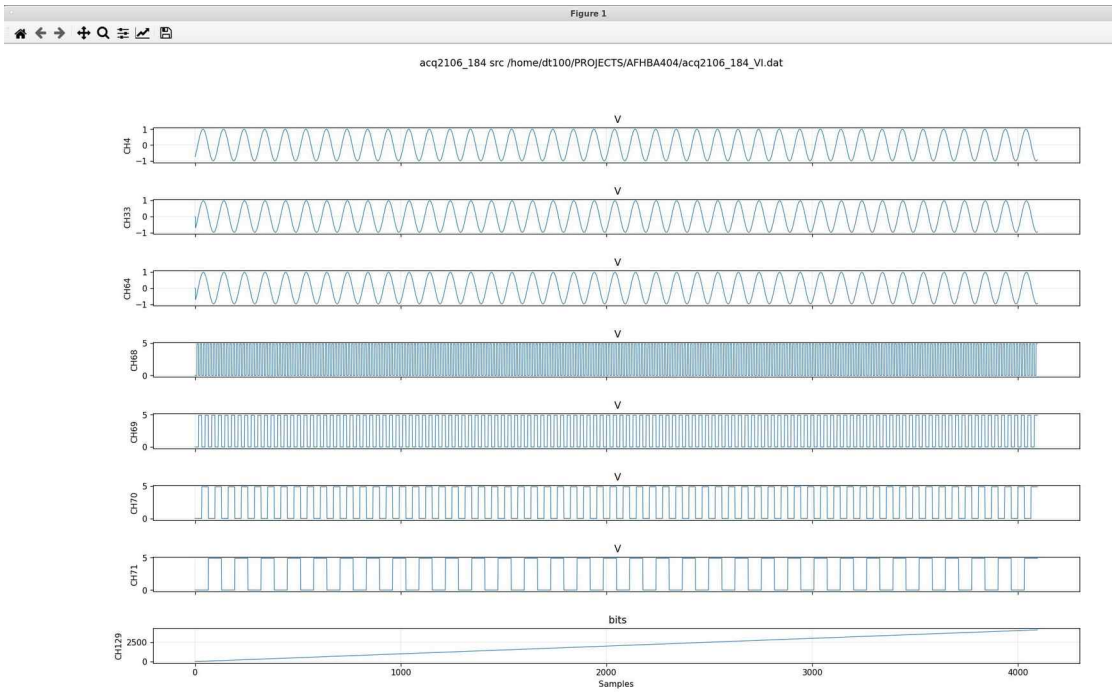
10.3.3.1 Data Flows



10.3.4 View EGU (Volts) Data:

```
dt100@kamino acq400_hapi]$ ./user_apps/analysis/host_demux.py \
--src=/home/dt100/PROJECTS/AFHBA404/acq2106_184_VI.dat --egu=1 \
--pchan=4,33,64,68,69,70,71,129  --plot_mpl=1 --pses=0:4096:1 acq2106_184
```

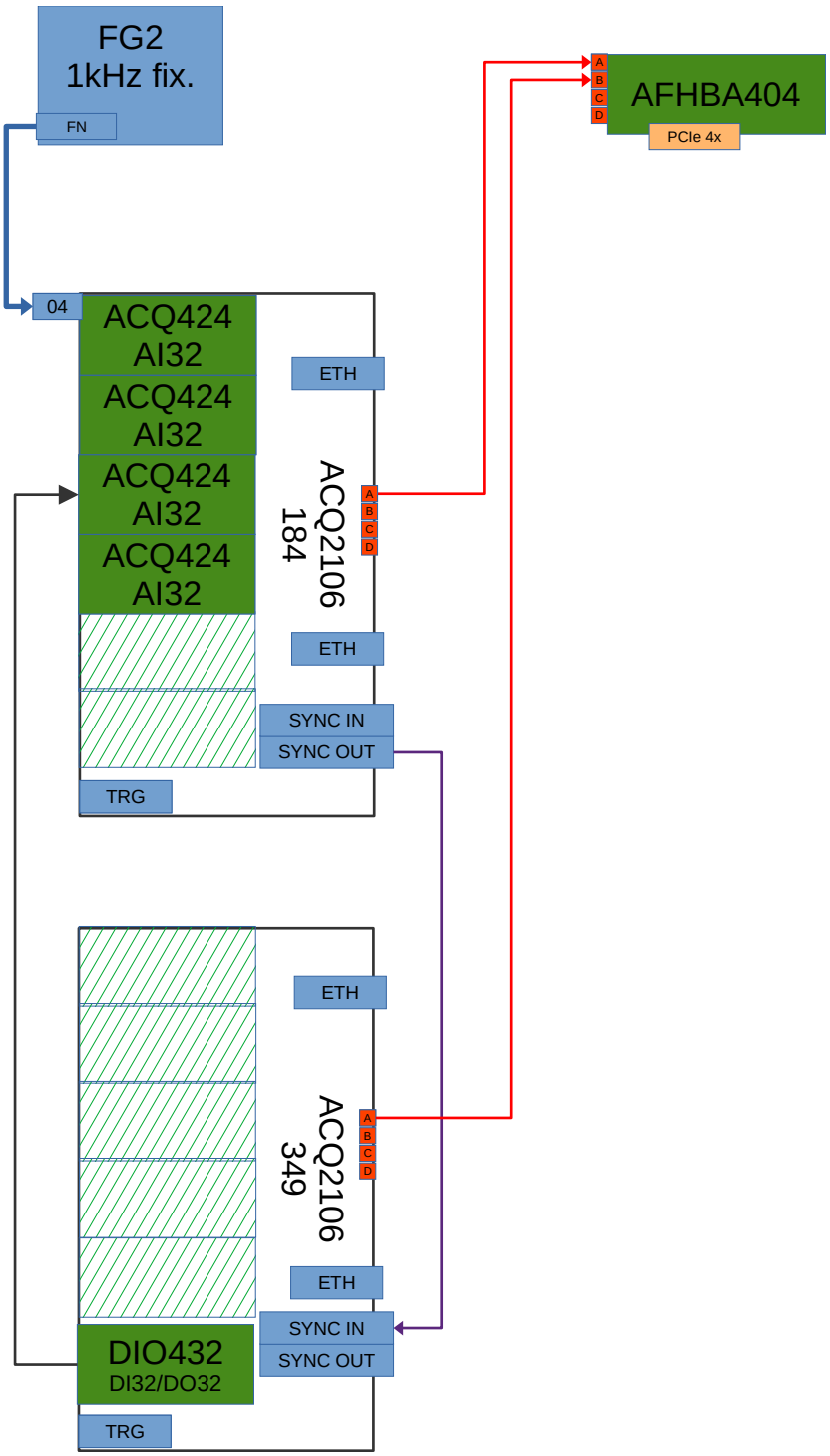
- CH04: Input from FG to AI32.1
- CH33,64: Input from AO32 to AI32.2
- CH65..96: Input from DO32 to AI32.3 # show slower signals
- CH129 : SPAD[0] Sample Count or “TLATCH”



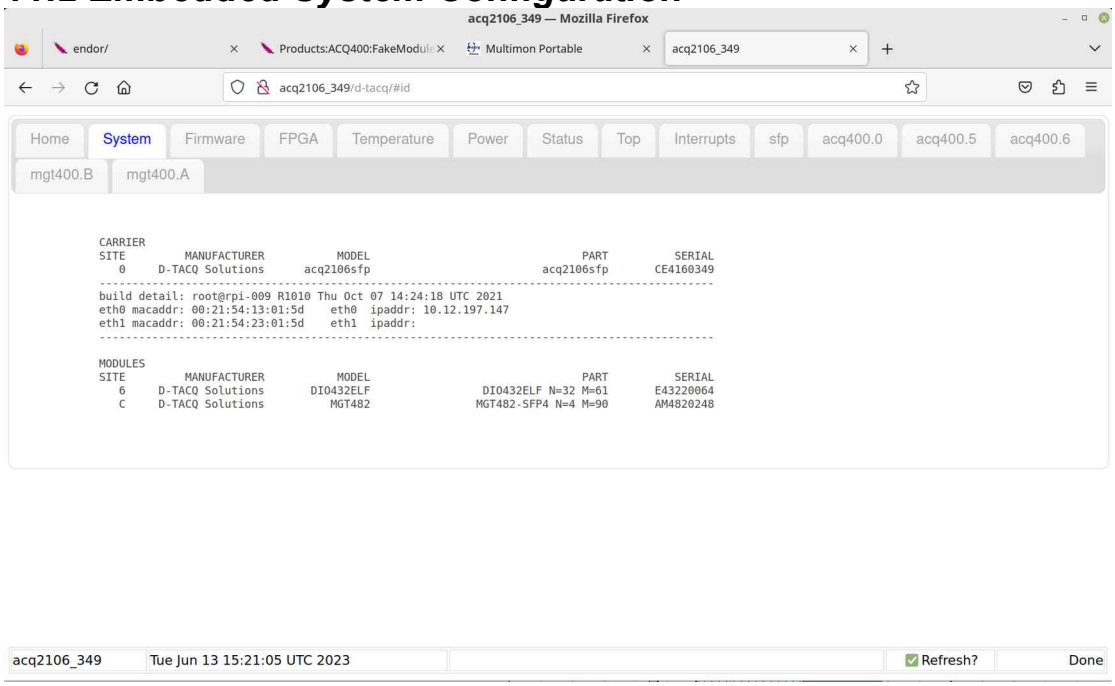
- Same as before: This is Good
- (OK, slight difference in AI phase, this is random from capture to capture)

11 ACQPROC ex: AI128+DO32

11.1 Hardware Configuration



11.2 Embedded System Configuration



11.3 Host Software Configuration.

11.3.1 AFHBA404 Version:

https://github.com/D-TACQ/AFHBA404/releases/tag/v2.7 or later.

11.3.2 Create System Model

```
[dt100@kamino AFHBA404]$ ./HAPI/lsafhba.py
0 HostComms(host='kamino.d-tacq', dev='0', uut='acq2106_184', cx='A')
1 HostComms(host='kamino.d-tacq', dev='1', uut='acq2106_349', cx='A')
2 HostComms(host='kamino.d-tacq', dev='2', uut='acq2106_176', cx='A')
```

Our UUTS are acq2106_184 and acq2106_349, on LPORTS 0,1:

```
[dt100@kamino AFHBA404]$ ./HAPI/lsafhba.py -l 0,1 \
--byte_is_output=1,1,1,1 \
-s SWIP_acq2106_184_128_acq2106_349_0_0_32.json
0 HostComms(host='kamino.d-tacq', dev='0', uut='acq2106_184', cx='A')
1 HostComms(host='kamino.d-tacq', dev='1', uut='acq2106_349', cx='A')
created SWIP_acq2106_184_128_acq2106_349_0_0_32.json
```

Creates the system model file SWIP_acq2106_184_128_acq2106_349_0_0_32.json:

SWIP_acq2106_184_128_acq2106_349_0_0_32.json:

```
{
    "AFHBA": {
        "UUT": [
            {
                "DEVNUM": 0,
                "name": "acq2106_184",
                "type": "pcs",
                "sync_role": "master",
                "COMMS": "A",
                "VI": {
                    "AI16": 128,
                    "SP32": 16,
                    "AISITES": [
                        1,
                        2,
                        3,
                        4
                    ],
                    "NXI": 4
                },
                "VO": {
                    "NXO": 0
                }
            },
            {
                "DEVNUM": 1,
                "name": "acq2106_349",
                "type": "pcs,nowait",
                "sync_role": "slave",
                "COMMS": "A",
                "VI": {},
                "VO": {
                    "DO32": 1,
                    "DO_BYTE_IS_OUTPUT": [
                        "1,1,1,1"
                    ],
                    "DIOSITES": [
                        6
                    ],
                    "NXO": 1
                }
            }
        ]
    }
}
```

11.3.3 Run The Shot

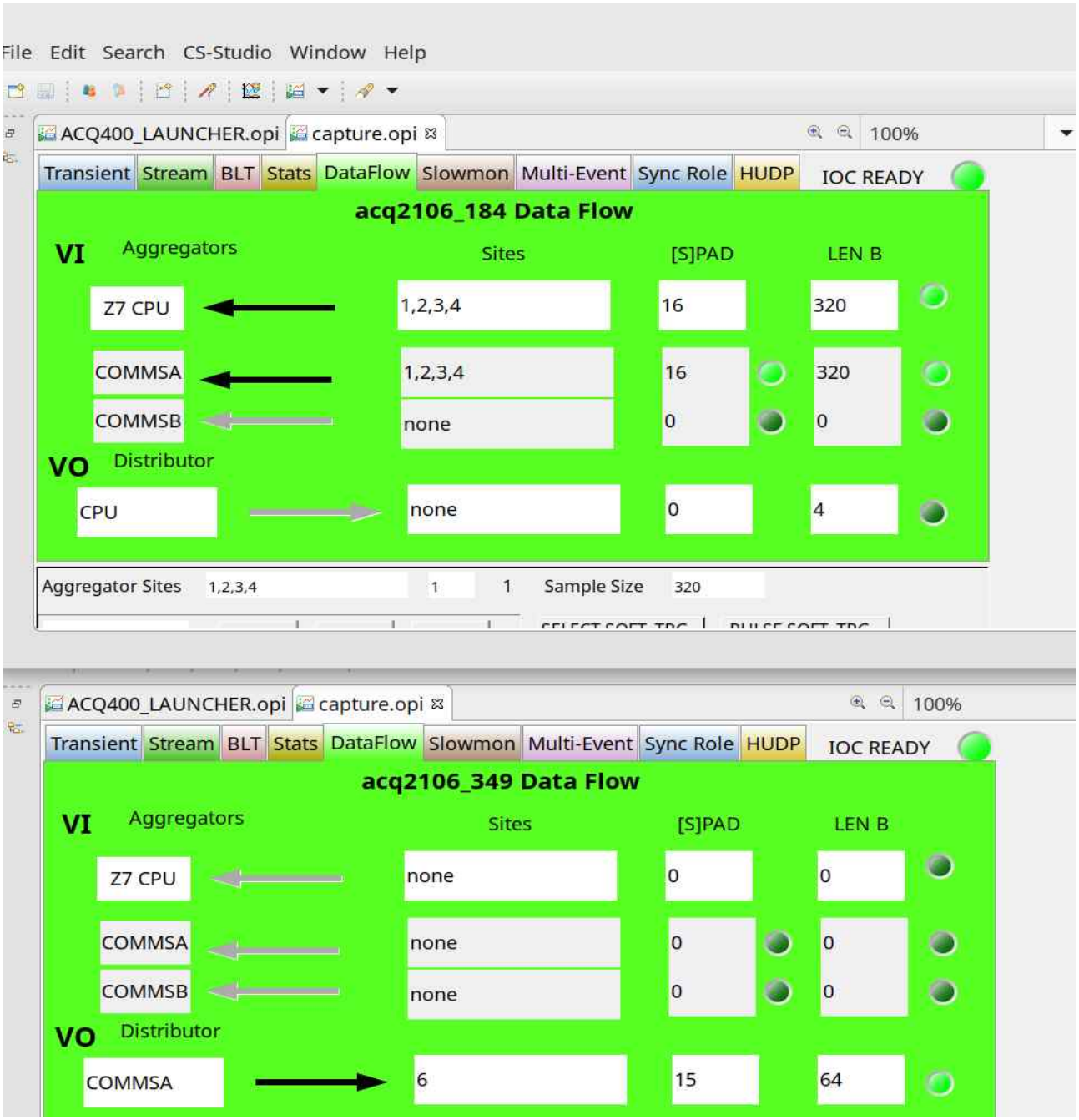
```
SINGLE_THREAD_CONTROL=control_dup1=3  ./scripts/acqproc_multi.sh
SWIP_acq2106_184_128_acq2106_349_0_0_32.json
```

- Transcript

```
[dt100@kamino AFHBA404]$ SINGLE_THREAD_CONTROL=control_dup1=3  ./scripts/acqproc_multi.sh
SWIP_acq2106_184_128_acq2106_349_0_0_32.json
UUT1 acq2106_184
UUT2 acq2106_349
UUTS acq2106_184 acq2106_349
DEVMAX 3
python3 scripts/llc-config-utility.py SWIP_acq2106_184_128_acq2106_349_0_0_32.json
Starting llc-config-utility
CONFIGURING acq2106_184
Link Good
config_VI
acq2106_184 Aggregator settings: sites=1,2,3,4 (spad=16) (comms=A)
CONFIGURING acq2106_349
Error: LANE_UP=1 RPCIE_INIT=0
Link down: attempting to correct 0/3
Link Fixed 0/3
DO_BYTE_IS_OUTPUT ['1,1,1,1']
json_override_actual( acq2106_349 [] VI AISITES)
json_override_actual( acq2106_349 [] VI DIOSITES)
DO_BYTE_IS_OUTPUT ['1,1,1,1']
config_VO
acq2106_349 Distributor settings: sites=6 pad=15 comms=A on
python3 scripts/llc-config-utility.py done 0
python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/sync_role.py --toprole=master --
fclk=50000 acq2106_184 acq2106_349

python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/sync_role.py done 0
control_program vanilla
python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/acq400_streamtonowhere.py --
samples=1000000 acq2106_184
nsamples set 1000000
NOTICE: port 0 set nowait
HBA1 VI:320 VO:4 devs=0,1
                [0] dev:0 acq2106_184 VI:320 VO:0 Offset of SPAD IN VI :256
 System Interface Indices 0,0
                [1] dev:1 acq2106_349 VI:0 VO:4 Offset of SPAD IN VI :0
 System Interface Indices 128,16
Arming systems now - please wait. Do not trigger yet.
All UUTs are armed and ready for trigger.
Streamed 0 of 1000000 samples
Streamed 75731 of 1000000 samples
..
Streamed 876113 of 1000000 samples
stored acq2106_184_VI.dat, len=319999680
stored acq2106_349_VO.dat, len=3999996
Control Program Finished
Streamed 926137 of 1000000 samples
Streamed 976160 of 1000000 samples

Stream finished.
python3 /home/dt100/PROJECTS/acq400_hapi/user_apps/acq400/acq400_streamtonowhere.py done 0

Running analysis now.
---------------------
python3 scripts/acqproc_analysis.py --ones=1 --json=1 --json_src=./runtime.json
--src=/home/dt100/PROJECTS/AFHBA404/
TKAgg not available for matplot
Running analysis for UUT: acq2106_184
show SPAD: hexdump -ve '128/2 "%04x," 16/4 "%08x," "\n"' acq2106_184_VI.dat  | cut -d, -f129-136 |
head -n 4
00000001,00000008,00000000,00000000,00000000,00000000,a6666184,a7777184
00000002,0000001c,00000000,00000000,00000000,00000000,a6666184,a7777184
00000003,00000030,00000000,00000000,00000000,00000000,a6666184,a7777184
00000004,00000044,00000000,00000000,00000000,00000000,a6666184,a7777184
```

```
show whole VI like this:
hexdump -ve '128/2 "%04x," 16/4 "%08x," "\n"' acq2106_184_VI.dat
/home/dt100/PROJECTS/AFHBA404//acq2106_184_VI.dat
tlatch = data[64::80]
Finished collecting data
T_LATCH differences:  1 , happened:  375011  times
T_LATCH differences:  2 , happened:  0  times
T_LATCH differences:  3 , happened:  0  times
```

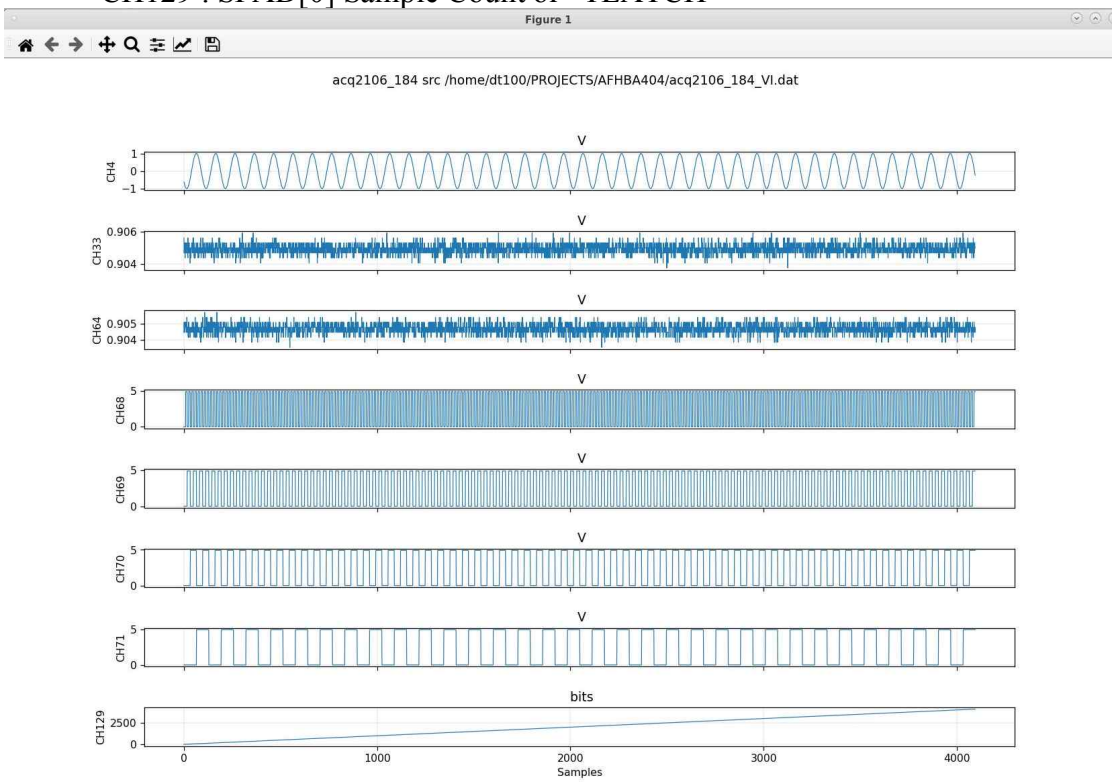
- Data Flows:



11.3.4 View EGU (Volts) Data:

```
dt100@kamino acq400_hapi]$ ./user_apps/analysis/host_demux.py \
--src=/home/dt100/PROJECTS/AFHBA404/acq2106_184_VI.dat --egu=1 \
--pchan=4,33,64,68,69,70,71,129  --plot_mpl=1 --pses=0:4096:1 acq2106_184
```

- CH04: Input from FG to AI32.1
- CH33,64: Input from AO32 to AI32.2
- CH65..96: Input from DO32 to AI32.3 # show slower signals
- CH129 : SPAD[0] Sample Count or “TLATCH”



- This time, the DO32 is the SAME, but the AO is missing, as expected: PASS!