

DOC-235260-01

2 x ACQ2106+ACQ482



Requirement
32 channels, 20MSPS.
Transient Capture to limit of memory
Example capture, offload, store, plot.

First Steps

- Each ACQ2106 is a NETWORK device, connect ETH0 to a gigabit switch.
- Device defaults to DHCP, else falls back to a static ip address on shipping doc.
- If you're unable to contact the device on Ethernet, please refer to
- **4GUG** Getting Started #5
 - you may need to use the serial console to access the unit
- Example Software support, use git to get LATEST versions
 - API: **HAPI**
 - GUI: **ACQ400CSS**

Configuration: Embedded Web pages

←

→

↻

⚠ Not secure | acq2106_372/d-tacq/#id

Home

System

Firmware

FPGA

Temperature

Power

Status

Top

Interrupts

CARRIER

SITE	MANUFACTURER	MODEL	PART	SERIAL
0	D-TACQ Solutions	acq2106	acq2106	CE4160372

build detail: root@rpi-009 R1010 Fri Jan 27 13:03:59 UTC 2023
eth0 macaddr: 00:21:54:13:01:74 eth0 ipaddr: 10.12.197.159
eth1 macaddr: 00:21:54:23:01:74 eth1 ipaddr:

MODULES

SITE	MANUFACTURER	MODEL	PART	SERIAL
1	D-TACQ Solutions	ACQ482ELF	ACQ482ELF-16-4V-H N=8 M=08	E48210180
2	D-TACQ Solutions	ACQ482ELF	ACQ482ELF-16-4V-H N=8 M=08	E48211180

←

→

↻

⚠ Not secure | acq2106_373/d-tacq/#id

Home

System

Firmware

FPGA

Temperature

Power

Status

Top

Interrupts

CARRIER

SITE	MANUFACTURER	MODEL	PART	SERIAL
0	D-TACQ Solutions	acq2106	acq2106	CE4160373

build detail: root@rpi-009 R1010 Fri Jan 27 14:26:12 UTC 2023
eth0 macaddr: 00:21:54:13:01:75 eth0 ipaddr: 10.12.197.1
eth1 macaddr: 00:21:54:23:01:75 eth1 ipaddr:

MODULES

SITE	MANUFACTURER	MODEL	PART	SERIAL
1	D-TACQ Solutions	ACQ482ELF	ACQ482ELF-16-4V-H N=8 M=08	E48210181
2	D-TACQ Solutions	ACQ482ELF	ACQ482ELF-16-4V-H N=8 M=08	E48211181

acq2106_372 - Google Chrome

acq2106_372

← → ↻ ⚠ Not secure | acq2106_372/d-tacq/#fw

★ Bookmarks IT Google Bookmark TinyURL! Imported From Fir... New Tab Gmail » Other bookmarks

Home System **Firmware** FPGA Temperature Power Status Top Interrupts sfp

acq400.0 acq400.1 acq400.2 mgt400.B mgt400.A adma0 acq480.1 acq480.2

RELEASE acq400-566-20230213175508
RELEASE : /tmp/release.md5
CURRENT : /tmp/current.md5
Base file system /etc/acq400 version:
acq400 buildroot acq400 v2022.03 pgm@staffa3 Wed 8 Feb 09:40:56 GMT 2023 00eb99411f6acb6761b67e800e618722
RELEASE acq400-566-20230213175508
Clean Release Installed

acq2106_372 Sat Mar 11 17:53:45 UTC 2023 ☒ Refresh? Done

acq2106_373 - Google Chrome

acq2106_373

← → ↻ ⚠ Not secure | acq2106_373/d-tacq/#fpga

★ Bookmarks IT Google Bookmark TinyURL! Imported From Fir... New Tab Gmail » Other bookmarks

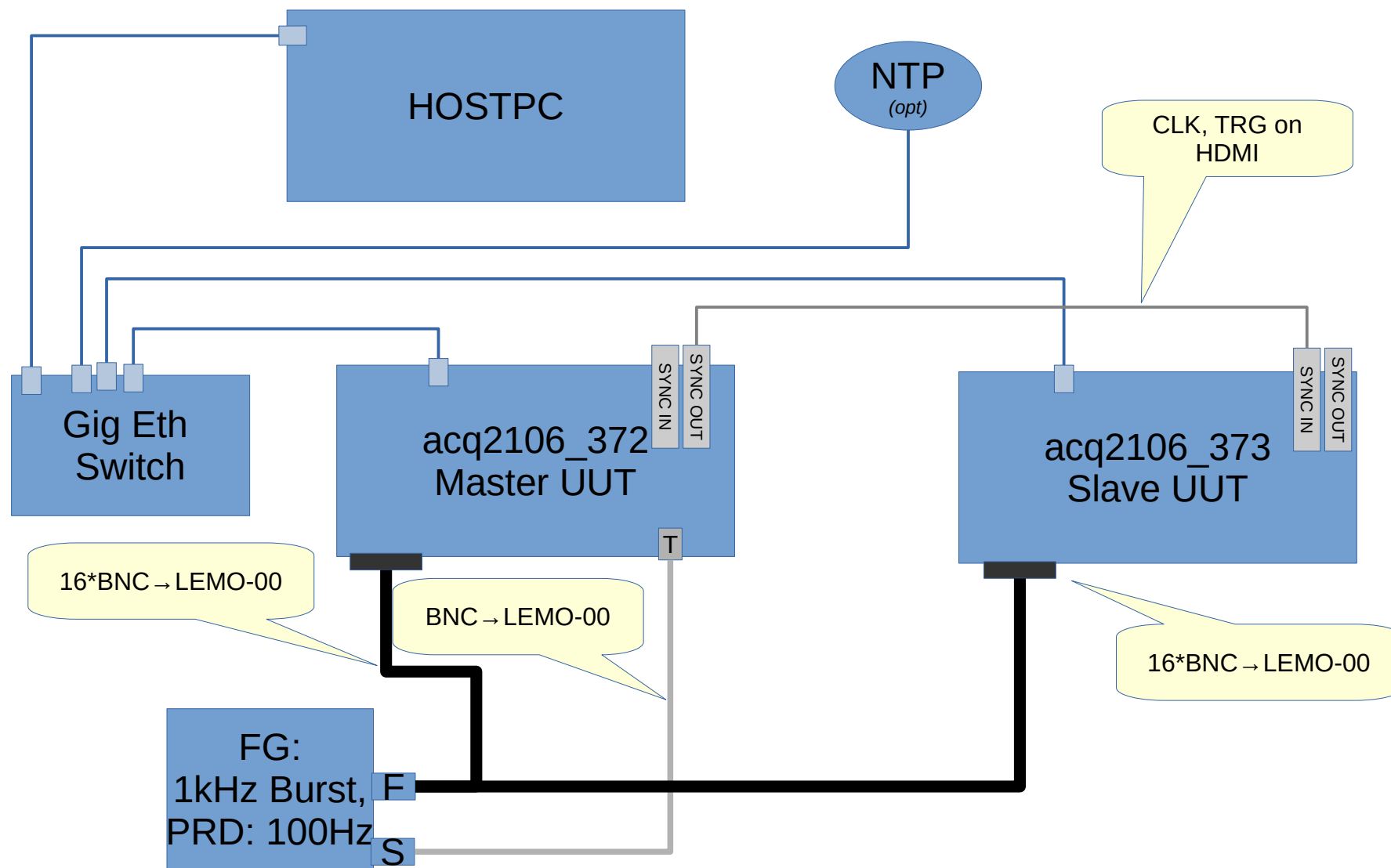
Home System Firmware **FPGA** Temperature Power Status Top Interrupts sfp

acq400.0 acq400.1 acq400.2 mgt400.B mgt400.A adma0 acq480.1 acq480.2

load.fpga loaded /mnt/fpga.d/ACQ2106_TOP_08_08_ff_ff_ff_9011_64B.bit.gz
xloader r1.01 (c) D-TACQ Solutions
eh location set 0
Xilinx Bitstream header.
built with tool version : 48
generated from filename : ACQ2106_TOP_08_08_08_08_08_9011_64B
part : 7z030fbg676
date : 2022/12/07
time : 15:28:40
bitstream data starts at : 134
bitstream data size : 5979916

acq2106_373 Sat Mar 11 17:53:44 UTC 2023 ☒ Refresh? Done

Test Configuration



UUT Firmware Customization

```
acq2106_372> cat /mnt/local/rc.user  
  
/usr/local/CARE/acq2106\+acq480.init  
# Local clock and front panel trigger  
set.site 0 sync_role master 20M 1M TRG:DX=d0
```

Common config,
suitable for using one UUT at a
time (simplest to get started)

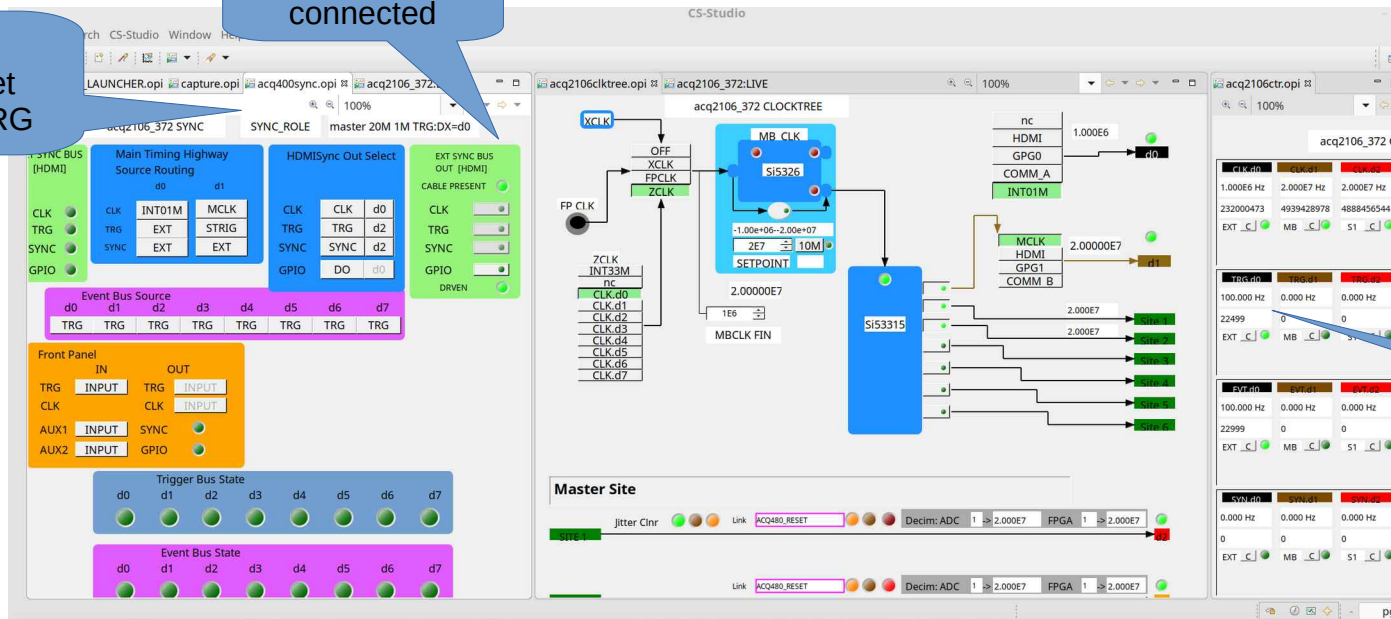
```
acq2106_372> cat /mnt/local/sysconfig/acq400.sh  
REBOOT_KNOB=y  
ACQ480_CUSTOM_BUF=y  
BLEN=4194304  
NBUF=166
```

Memory Customization:
Allocate 640MB to capture
 $16\text{ch} \times 2\text{b} \times 20\text{M} = 640\text{MB/s}$
maximum shot time: 1s
common to both boxes

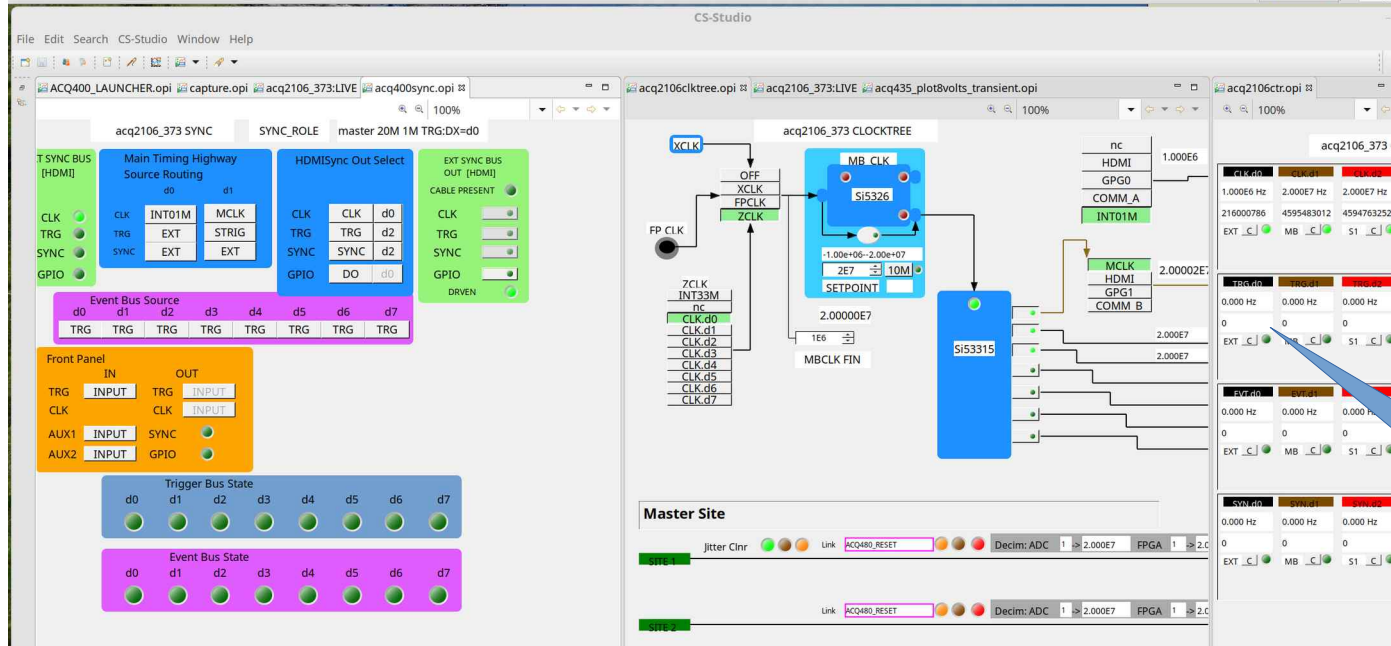
GUI: Not essential, but very useful

Default Boot
Both UUTs are set
master 20M ext TRG

SYNC_OUT
connected



EXT TRG
connected



EXT TRG
NOT connected
(normal)

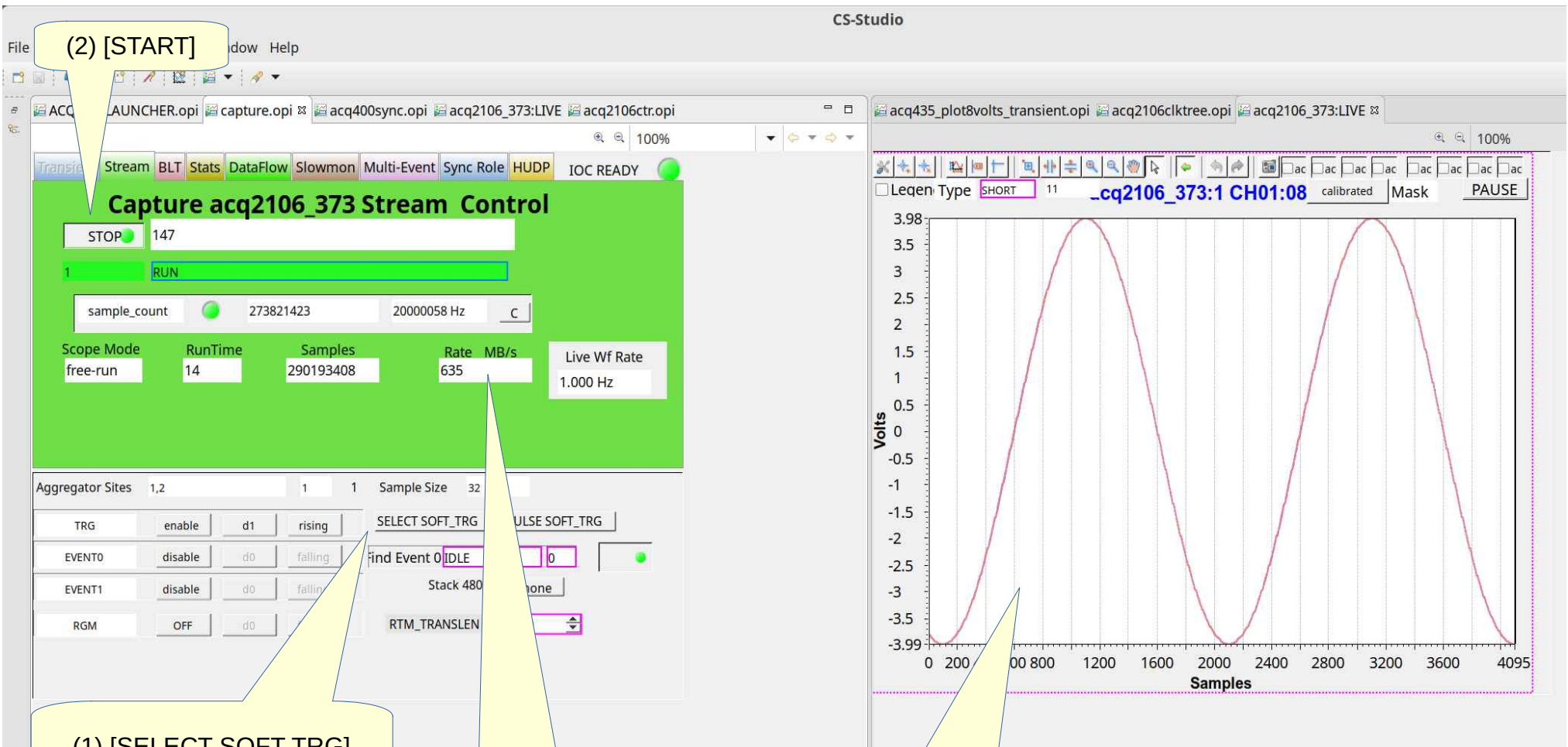
Push Button Livestream From GUI

The screenshot displays the CS-Studio interface. The left panel, titled 'Capture acq2106_373 Stream Control', features a green background and includes a 'START' button, a red 'IDLE' status bar, and various monitoring fields for sample count, scope mode, runtime, samples, rate, and live waveform rate. Below this, there are settings for aggregator sites, triggers, and events. The right panel shows a block diagram for 'acq2106_373 CLOCKTREE', which includes a clock tree for Si5326 and Si53315, with various clock signals and output sites (d0, d1, d2, d3, d4, d5, d6, d7) connected to different modules like nc, HDMI, GPG0, COMM_A, and INT01M.

```
acq2106_373> /mnt/local/enable_streaming
```

UUT is optimized for
Transient, run this for best
Stream experience

Push Button Livestream From GUI (2)



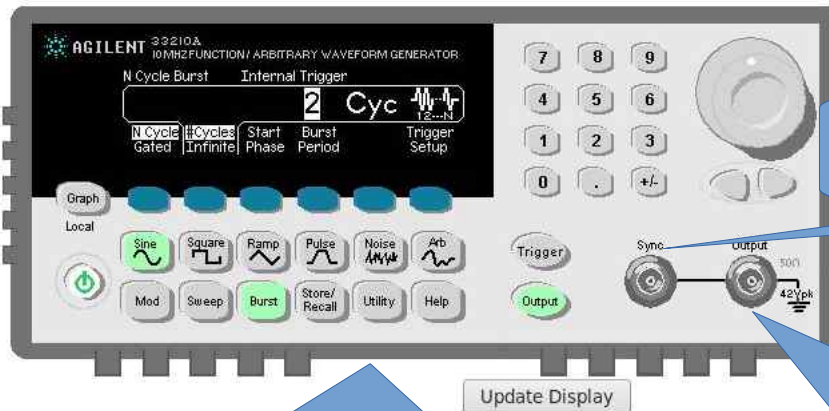
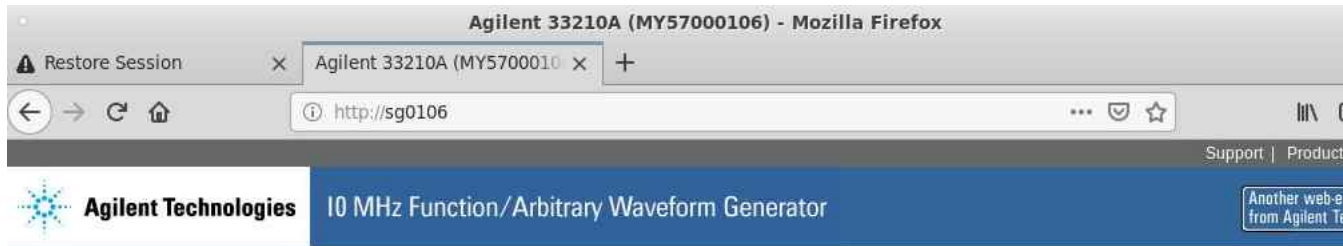
(3) Data Flow starts at about 20s after link training (see CLOCKTREE)

FG set to continuous:
Live WF snapshots, 4096 points

Push Button Transient From GUI



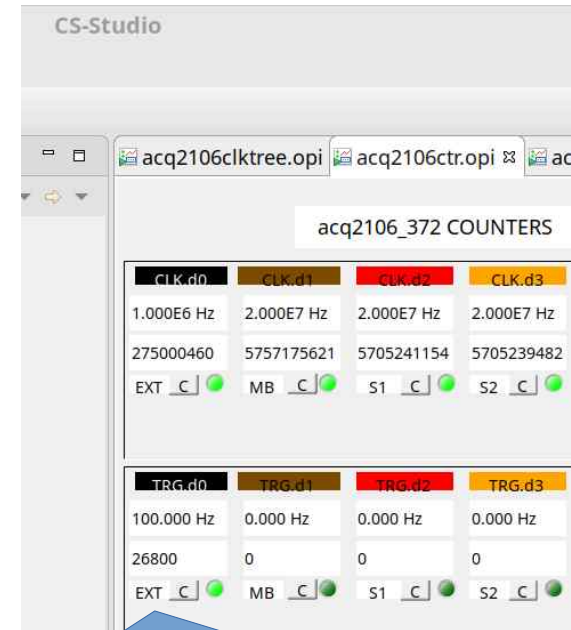
Long Duration Capture



FG fires a short
10kHz burst every 10msec.
(100Hz)

Sync → Master FP TRG
Check: TRG.d0=100Hz

Output (Func) to Analog
Inputs, both UUTs.



Automated Capture With HAPI

```
[dt100@naboo acq400_hapi]$ python3 \
./user_apps/acq400/acq400_upload.py \
--demux=0 --capture=1 --trg=ext,rising \
--post=20M --save_data /mnt/ --shots=1 acq2106_373
host shot 0 uut shot 652
acq2106_373 soft_trigger
acq2106_373 SHOT COMPLETE shot:652
INFO: Shotcontroller.handle_data() acq2106_373 data valid: DATA_VALID
TIMING:func:'handle_data' took: 22.48 sec
Finally, going down
TIMING:func:'run_shot' took: 35.79 sec
TIMING:func:'upload' took: 37.11 sec
TIMING:func:'run_main' took: 37.11 sec
[dt100@naboo acq400_hapi]$ ls -l /mnt/
total 625008
-rw-r--r-- 1 dt100 d-tacq 640000000 Mar 11 18:58 acq2106_373_CH00
```

SINGLE UUT
keep it simple..

ATTENTION!

SINGLE UUT
*Works with either box,
but please ensure that the
external trigger is
applied to the UUT
.. also run single UUT
fresh from reboot without
changing clocking (p16)*



Output (Func) to Analog
Inputs, both UUTs.

Plot Data from HAPI

```
python3 ./user_apps/analysis/host_demux.py \  
--src /mnt/acq2106_373_CH00 --pchan 1 acq2106_373
```

Figure 1

acq2106_373 src /mnt/acq2106_373_CH00

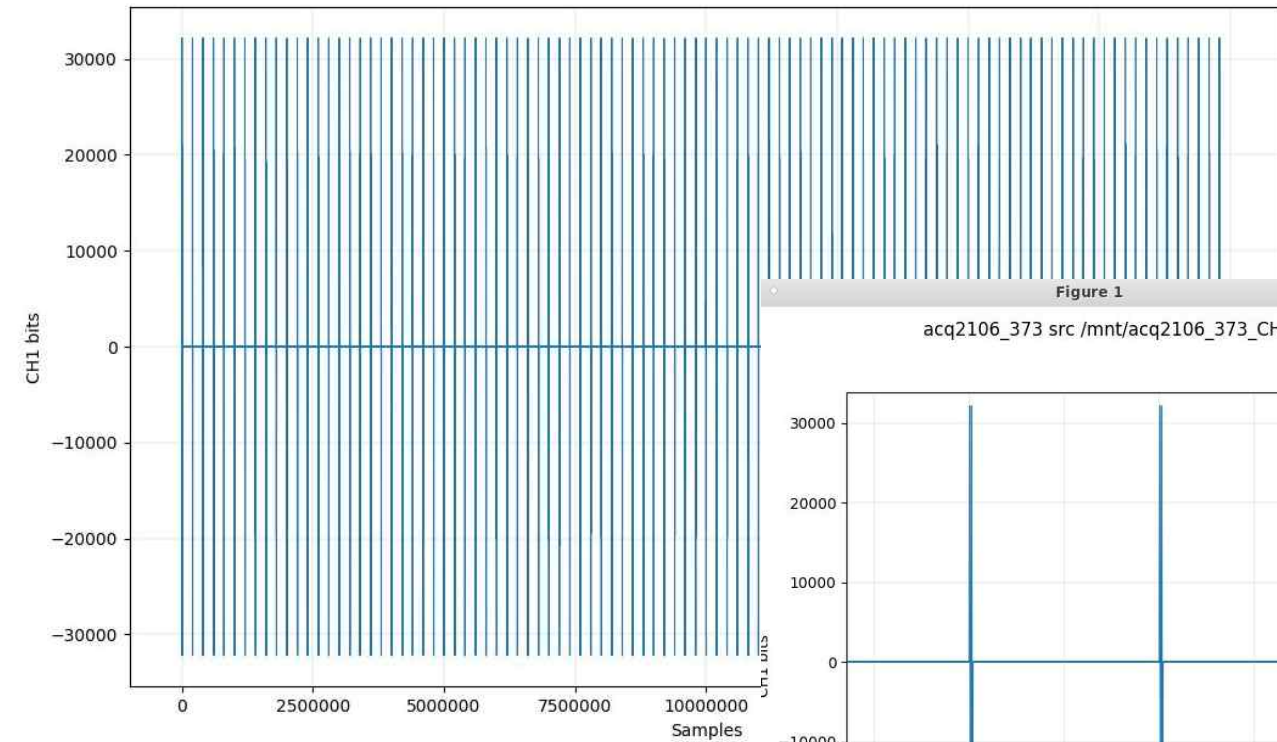


Figure 1

acq2106_373 src /mnt/acq2106_373_CH00

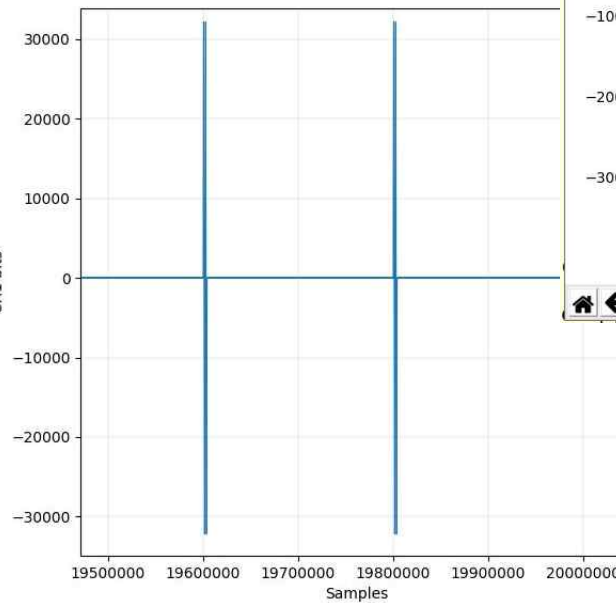
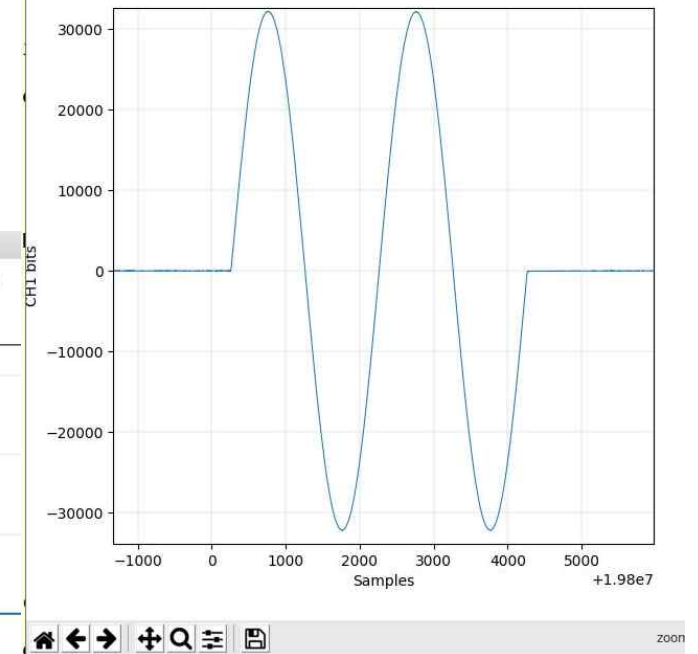


Figure 1

acq2106_373 src /mnt/acq2106_373_CH00



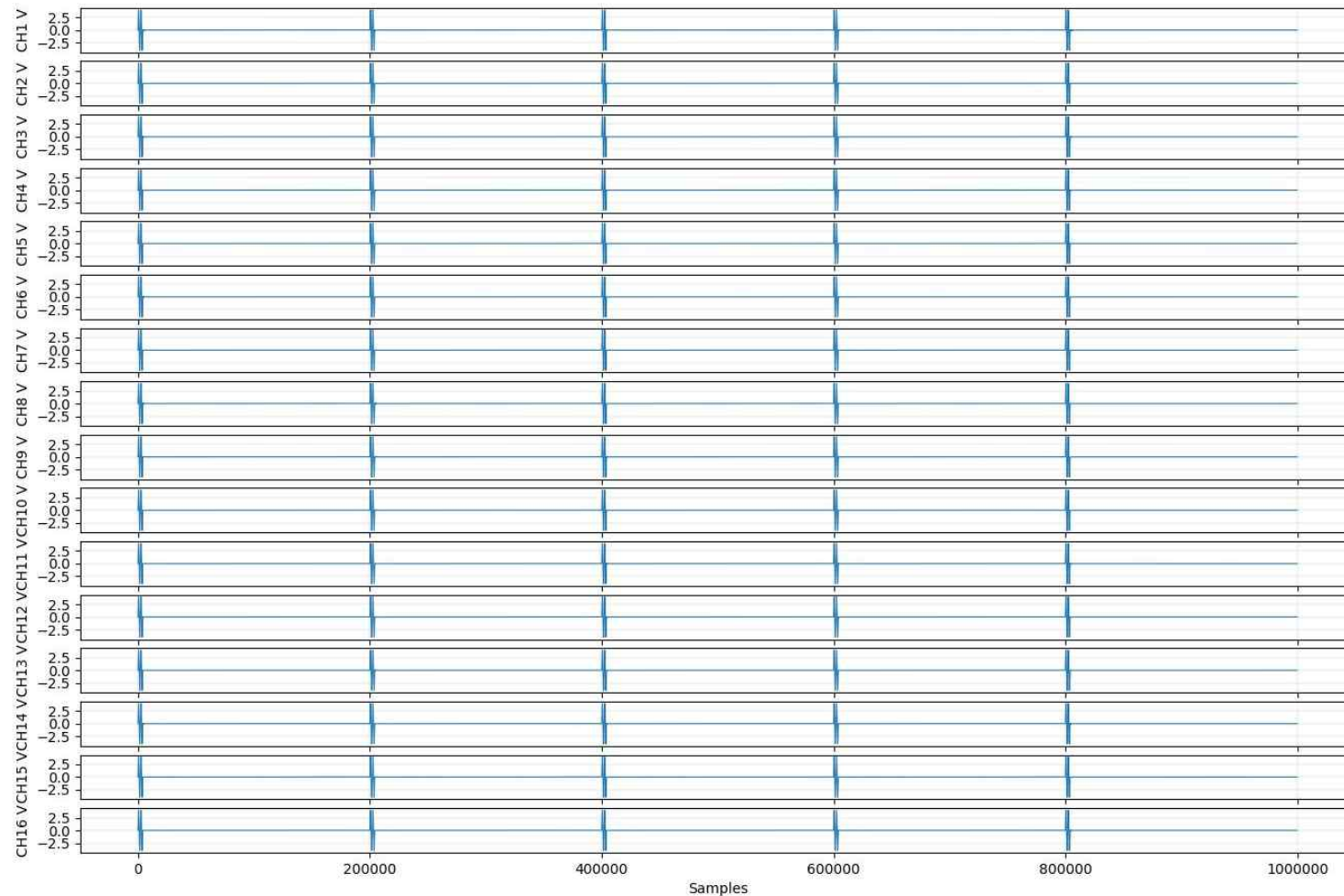
Plot Multiple channels restrict period

```
python3 ./user_apps/analysis/host_demux.py \  
--src /mnt/acq2106_373_CH00 --pses=0:1000000:1 acq2106_373
```

Figure 1

acq2106_373 src /mnt/acq2106_373_CH00

start:end:stride
limit the data to plot..



Command Trace (capture, upload)

record the commands and use in your own program..

```
[dt100@naboo acq400_hapi]$ SITECLIENT_TRACE=1 NETCLIENT_TRACE=1 python3 \
./user_apps/acq400/acq400_upload.py --demux=0 --capture=1 \
--trg=ext,rising --post=20M --save_data /mnt/ --shots=1 acq2106_373
Netclient.init acq2106_373 4220
Siteclient(acq2106_373, 4220) >SITELIST
...
Siteclient(acq2106_373, 4220) >state
Siteclient(acq2106_373, 4220) <0 0 20000000 20578304 0
Netclient.init acq2106_373 2235
Siteclient(acq2106_373, 4220) >sync_role
Siteclient(acq2106_373, 4220) <master 20000000
Siteclient(acq2106_373, 4221) >event0=0,0,0
Siteclient(acq2106_373, 4221) <
Siteclient(acq2106_373, 4221) >trg=1,0,1
Siteclient(acq2106_373, 4221) <
Siteclient(acq2106_373, 4220) >transient=PRE=0 POST=20000000 SOFT_TRIGGER=0 DEMUX=0
Siteclient(acq2106_373, 4220) <
Siteclient(acq2106_373, 4221) >trg=1,0,1
Siteclient(acq2106_373, 4221) <
Siteclient(acq2106_373, 4221) >event0=0,0,0
Siteclient(acq2106_373, 4221) <
Siteclient(acq2106_373, 4220) >TRANSIENT:SET_ABORT=1
Siteclient(acq2106_373, 4220) <
Siteclient(acq2106_373, 4220) >TRANS_ACT:STATE
Siteclient(acq2106_373, 4220) <TRANS_ACT:STATE IDLE
Siteclient(acq2106_373, 4220) >shot
Siteclient(acq2106_373, 4220) <1
Siteclient(acq2106_373, 4221) >shot=1
Siteclient(acq2106_373, 4221) <
Siteclient(acq2106_373, 4220) >TRANSIENT:SET_ARM=1
Siteclient(acq2106_373, 4220) <
Siteclient(acq2106_373, 4220) >soft_trigger=1
Siteclient(acq2106_373, 4220) <
Siteclient(acq2106_373, 4220) >NCHAN
Siteclient(acq2106_373, 4220) <16
Siteclient(acq2106_373, 4220) >data32
Siteclient(acq2106_373, 4220) <0
Siteclient(acq2106_373, 4220) >transient
Siteclient(acq2106_373, 4220) <PRE=0 POST=20000000 OSAM=1 DEMUX=0 SOFT_TRIGGER=0
Siteclient(acq2106_373, 4220) >raw_data_size
Siteclient(acq2106_373, 4220) <640000000
Netclient.init acq2106_373 53000
TIMING:func:'handle_data' took: 18.10 sec
```

*Trace shows commands
a user program would need to
execute*

*or..
run python from Labview?*

Command Trace (plot egu)

record the commands and use in your own program..

```
Siteclient(acq2106_373, 4220) <6400000000
Netclient.init acq2106_373 53000
TIMING:func:'handle_data' took: 18.10 sec
```

(1) after the shot, connect to port 53000/TCP and read RAW data: 640e6 bytes on each UUT.

```
[dt100@naboo ~]$ hexdump -e '16/2 "%04x," "\n"' /mnt/acq2106_373_CH00 | head
fff7,ff8f,ffa7,ff6f,005f,0003,002f,00ab,002b ffd3 0077,fff3,0023,005b,0003,ffbb,
ffef,ff8f,ff9f,ff67,005b,000f,001f,00a3,0023 ffd7 0077,fff3,0023,005f,0003,ffc7,
ffef,ff93,ffa3,ff63,005b,0007,001f,009f,0027 ffc3 007b,fff7,0023,005f,ffff,ffc3,
ffeb,ff97,ff9f,ff6b,005b,0007,002b,00a3,001f ffd7 007b,fffb,001f,0063,ffff,fffb,
ffeb,ff97,ffa3,ff6f,0063,000b,002b,00a7,002b ffd7 006f,fff7,0023,0063,ffff,fffb,
ffef,ff93,ffa3,ff6f,005b,0007,0023,009b,0027 ffd3 007b,fff3,0027,005f,ffff,fffb,
ffeb,ff93,ffa7,ff6f,0063,000f,002b,00af,0027 ffd3 0077,fffb,0027,0063,000b,ffff,
fff3,ffa3,ff9f,ff6b,005f,000b,001b,009f,002f ffd7 007f,fff7,0023,006b,0003,ffff,
ffef,ff97,ffab,ff6f,0063,000b,0037,00a3,002f ffc3 0077,fffb,0023,0063,0007,ffff,
fff7,ff97,ffab,ff73,005b,0013,0033,00af,002f ffd7 0077,fffb,002b,0063,000b,ffff,
```

(2) The raw data is very simply: 16 columns of int16, one row per sample, 20e6 rows

NB: this data is “Little Endian”, INTEL-style format. It’s our understanding that Labview was “Big Endian”, so a byte swap would be required.

We can see the reshape is correct, because the (floating) inputs track, on the left, a channel with a small negative offset, on the right, a channel with a small positive offset.

(3) Read calibrated ESLO (Slope) and EOFF (Offset) value to convert to volts
 $V = \text{Raw} * \text{ESLO} - \text{EOFF}$

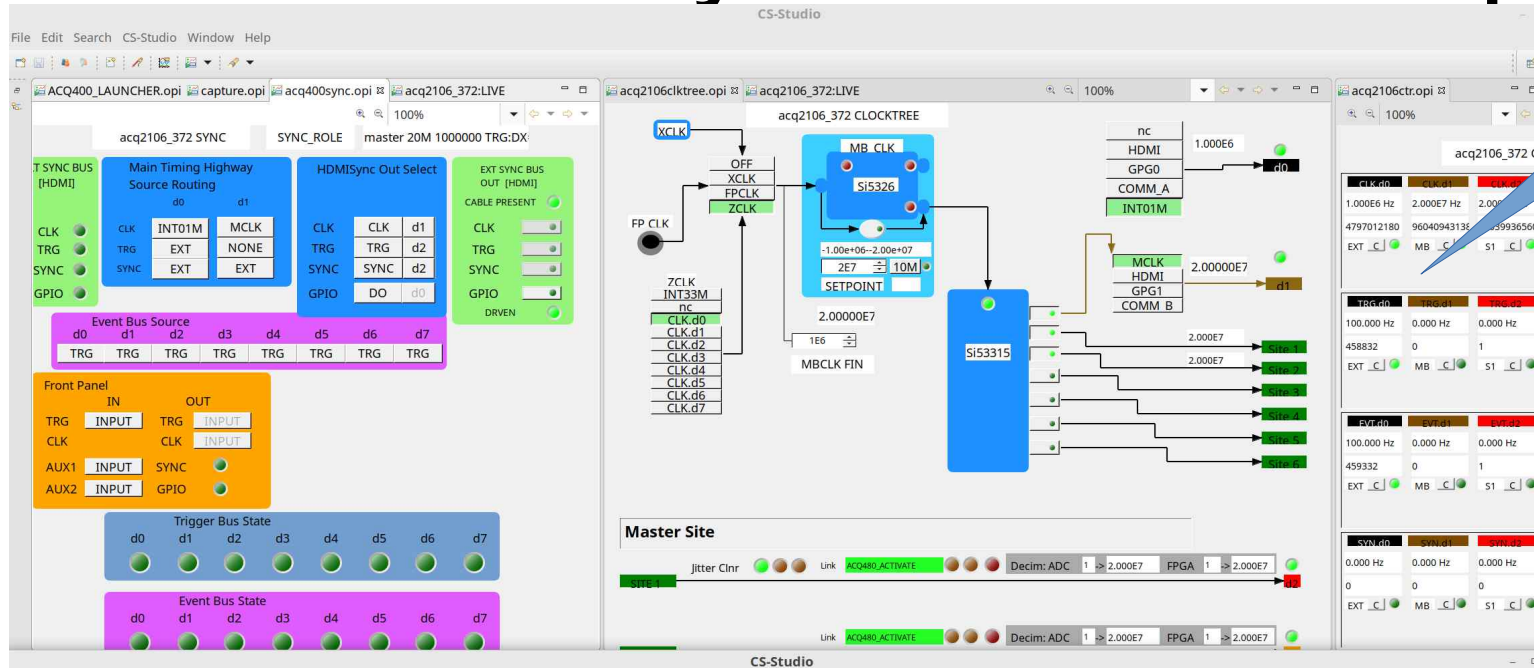
```
[dt100@naboo acq400_hapi]$ SITECLIENT_TRACE=1 python3 \
./user_apps/analysis/host_demux.py --src /mnt/acq2106_373_CH00 --pses=0:1000000:1 --egu=1 acq2106_373
```

```
Siteclient(acq2106_373, 4221) >AI:CAL:ESLO
Siteclient(acq2106_373, 4221) <AI:CAL:ESLO 9 0 0.00012389 0.00012354 0.000123467 0.000123442 0.000123729 0.000123455 0.00012376 0.000123876
Siteclient(acq2106_373, 4221) >AI:CAL:EOFF
Siteclient(acq2106_373, 4221) <AI:CAL:EOFF 9 0 -0.000218326 0.0112563 0.00950572 0.0159721 -0.0138621 -0.00309993 -0.00654523 -0.0223364
Siteclient(acq2106_373, 4222) >AI:CAL:ESLO
Siteclient(acq2106_373, 4222) <AI:CAL:ESLO 9 0 0.000123621 0.000123313 0.000123194 0.00012316 0.000123405 0.000123189 0.000123557 0.00012353
Siteclient(acq2106_373, 4222) >AI:CAL:EOFF
Siteclient(acq2106_373, 4222) <AI:CAL:EOFF 9 0 -0.00745583 0.00306922 -0.0175009 -0.000814031 -0.00650478 -0.0139123 -0.00247568 0.00503548
```

Control Two UUTS M,S: Clocking

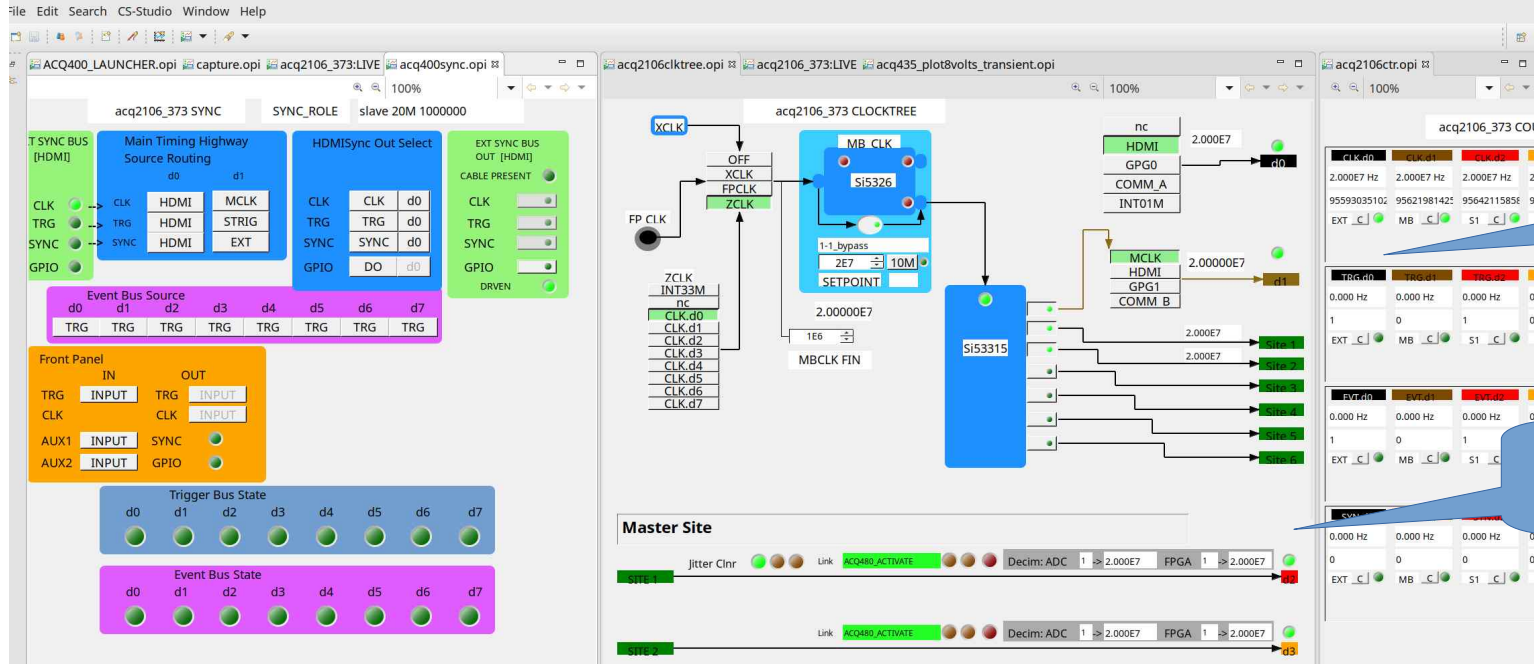
```
[peter@andros acq400_hapi]$ SITECLIENT_TRACE=1 python3 \  
./user_apps/acq400/sync_role.py --toprole master,fptrg --fclk=20M \  
--downstream_bypass=1 acq2106_372 acq2106_373  
Siteclient(acq2106_372, 4220) >SITELIST  
Siteclient(acq2106_372, 4220) <216,1=482,2=482  
Siteclient(acq2106_372, 4221) >module_name  
Siteclient(acq2106_372, 4222) >module_name  
Siteclient(acq2106_372, 4221) <acq480fmc  
Siteclient(acq2106_372, 4222) <acq480fmc  
Siteclient(acq2106_372, 4220) >state  
Siteclient(acq2106_372, 4220) <0 0 0 0 0  
Siteclient(acq2106_372, 4220) >sync_role=master 20M 1000000 TRG:DX=d0 TRG:SENSE=rising  
Siteclient(acq2106_372, 4220) <  
Siteclient(acq2106_372, 4220) >SIG:SYNC_OUT:CLK:DX=d1  
Siteclient(acq2106_372, 4220) <  
Siteclient(acq2106_372, 4220) >SIG:SRC:TRG:0=NONE  
Siteclient(acq2106_372, 4220) <  
Siteclient(acq2106_372, 4220) >SIG:SRC:TRG:1=NONE  
Siteclient(acq2106_372, 4220) <  
Siteclient(acq2106_373, 4220) >SITELIST  
Siteclient(acq2106_373, 4220) <216,1=482,2=482  
Siteclient(acq2106_373, 4221) >module_name  
Siteclient(acq2106_373, 4222) >module_name  
Siteclient(acq2106_373, 4221) <acq480fmc  
Siteclient(acq2106_373, 4222) <acq480fmc  
Siteclient(acq2106_373, 4220) >state  
Siteclient(acq2106_373, 4220) <0 0 0 0 0  
Siteclient(acq2106_373, 4220) >sync_role=slave 20M 1000000  
Siteclient(acq2106_373, 4220) <  
Siteclient(acq2106_373, 4220) >si5326bypass  
Siteclient(acq2106_373, 4220) <0  
Siteclient(acq2106_373, 4220) >set_si5326_bypass=1
```

Summary of Clock Setup



M: EXT TRG
Free run

M: after ONE Shot,
ONE output trigger



S: after ONE Shot,
ONE input trigger

Each site sees the
same 20MHz clock

Run Multi-UUT shot

```
SITECLIENT_TRACE=1 python3 ./user_apps/acq400/acq400_upload.py \  
    --demux=0 --capture=1 --post=20M --save_data /mnt/ --shots=1 --remote_trigger=EXT \  
    acq2106_372 acq2106_373  
ost shot 0 uut shot 2  
acq2106_372 SHOT COMPLETE shot:3  
acq2106_373 SHOT COMPLETE shot:3  
INFO: Shotcontroller.handle_data() acq2106_372 data valid: DATA_VALID  
TIMING:func:'handle_data' took: 41.56 sec  
Finally, going down  
TIMING:func:'run_shot' took: 55.14 sec  
TIMING:func:'upload' took: 59.11 sec  
TIMING:func:'run_main' took: 59.11 sec
```

42s to offload two UUTS.
could be faster if multi-threaded.

```
[peter@andros acq400_hapi]$ ls -l /mnt/  
total 1250008  
-rw-r--r--. 1 peter d-tacq 640000000 Mar 13 13:56 acq2106_372_CH00  
-rw-r--r--. 1 peter d-tacq 640000000 Mar 13 13:56 acq2106_373_CH00
```

604MB data per
UUT.

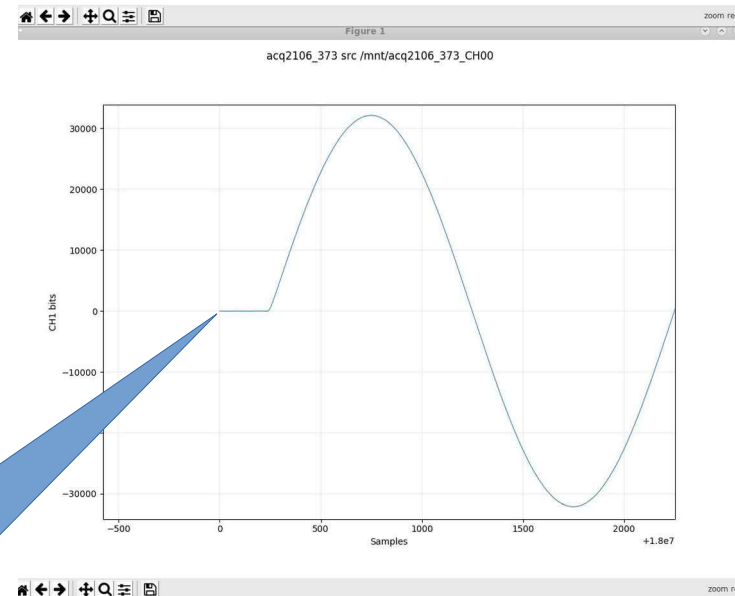
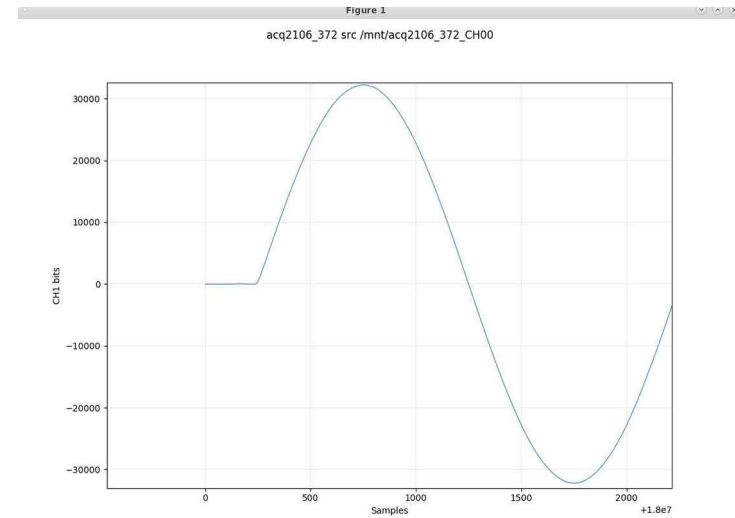
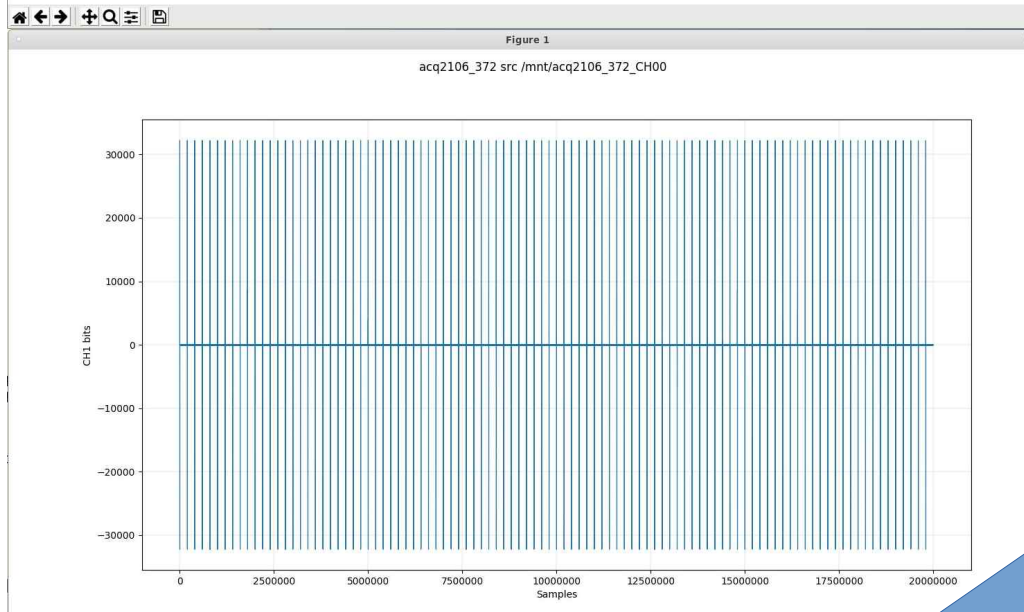
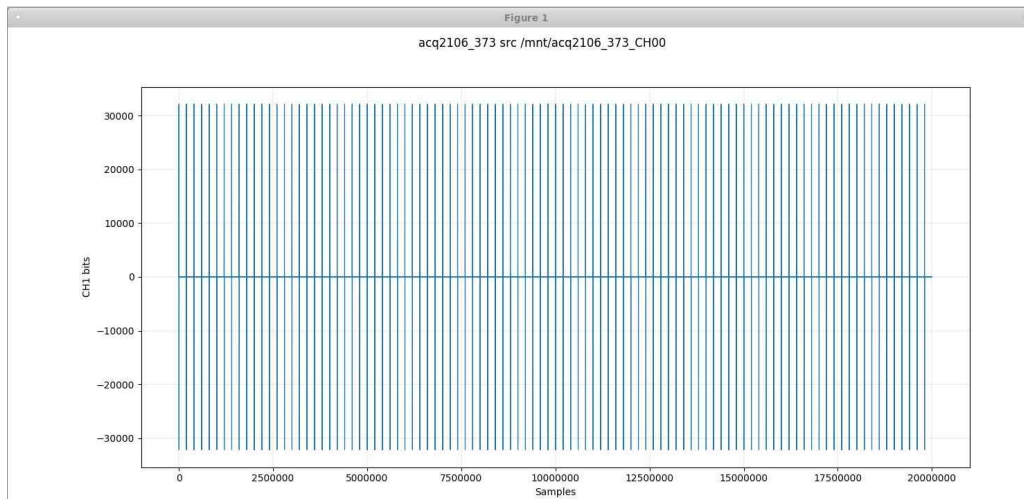
```
Plot Data:  
python3 ./user_apps/analysis/host_demux.py \  
    --src /mnt/acq2106_372_CH00 --pchan 1 acq2106_372 &  
python3 ./user_apps/analysis/host_demux.py \  
    --src /mnt/acq2106_373_CH00 --pchan 1 acq2106_373
```

Plot full duration

```
python3 ./user_apps/analysis/host_demux.py \  
    --src /mnt/acq2106_372_CH00 --pchan 1 --pses=18000000:18250000:1 acq2106_372 &  
python3 ./user_apps/analysis/host_demux.py \  
    --src /mnt/acq2106_373_CH00 --pchan 1 --pses=18000000:18250000:1 acq2106_373
```

Plot end of data for
easier plot exam.

Plotting Multi-UUT data..



Zoom to last pulse.
Pulse start is identical on
both UUTs

Copyright © D-TACQ Solutions Ltd 2023