Tracker Readout Board – Getting started

1 Preliminaries

1.1 Hardware

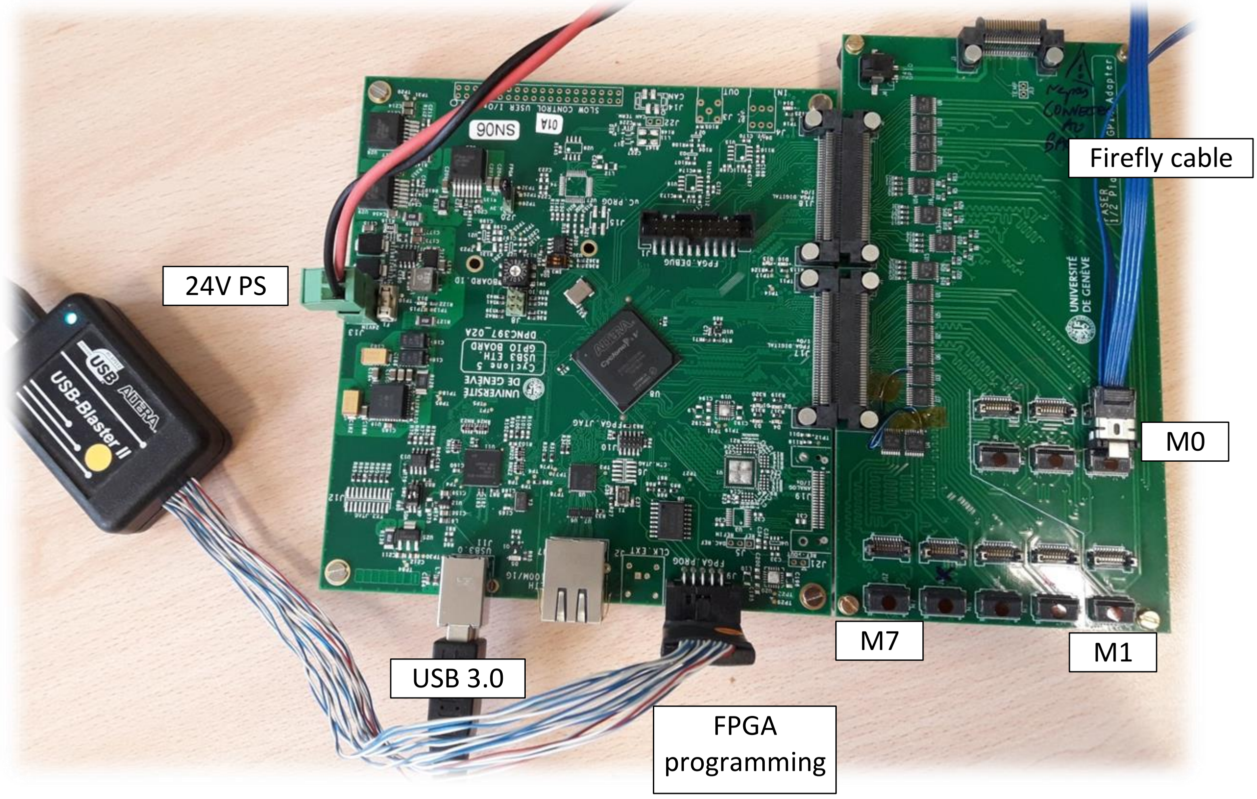
A single 24V (0.5A max, 0.1A nominal) and USB3.0 connection is required for the GPIO/TRB

A USB-Blaster II device connected to 'FPGA PROG.' connector is required for FPGA firmware upgrade. The firmware version is located on http://dpnc.unige.ch/~favrey/FASER/TRB/FW/. Please follow 'FPGA-UpgradeProcedure.txt' procedure for the upgrading the firmware. You also have a summary of the FPGA versions in 'Readme-Versions.txt'.

The module(s) is(are) connected with the SAMTEC FireFly connector(s) on UEC5/UCC8 connector(s) having the label M0, M1, … M7 which correspond to the GUI enabled module(s).

Do all connections with SCT modules and GPIO board switched off

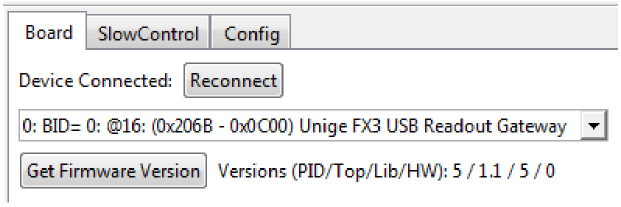
The following pictures show the hardware and its connections.



1.2 Software

The Unige GUI for the GPIO/TRB is located on http://dpnc.unige.ch/~favrey/FASER/TRB/FrontEnd/. Windows and linux version is available. Please follow 'install.txt' procedure for the installation procedure.

Once connected to the board and the GPIO switched on, the board 'Unige FX3 USB Readout gateway' should appear in the Board Tab, in the list below 'Device Connected' label:



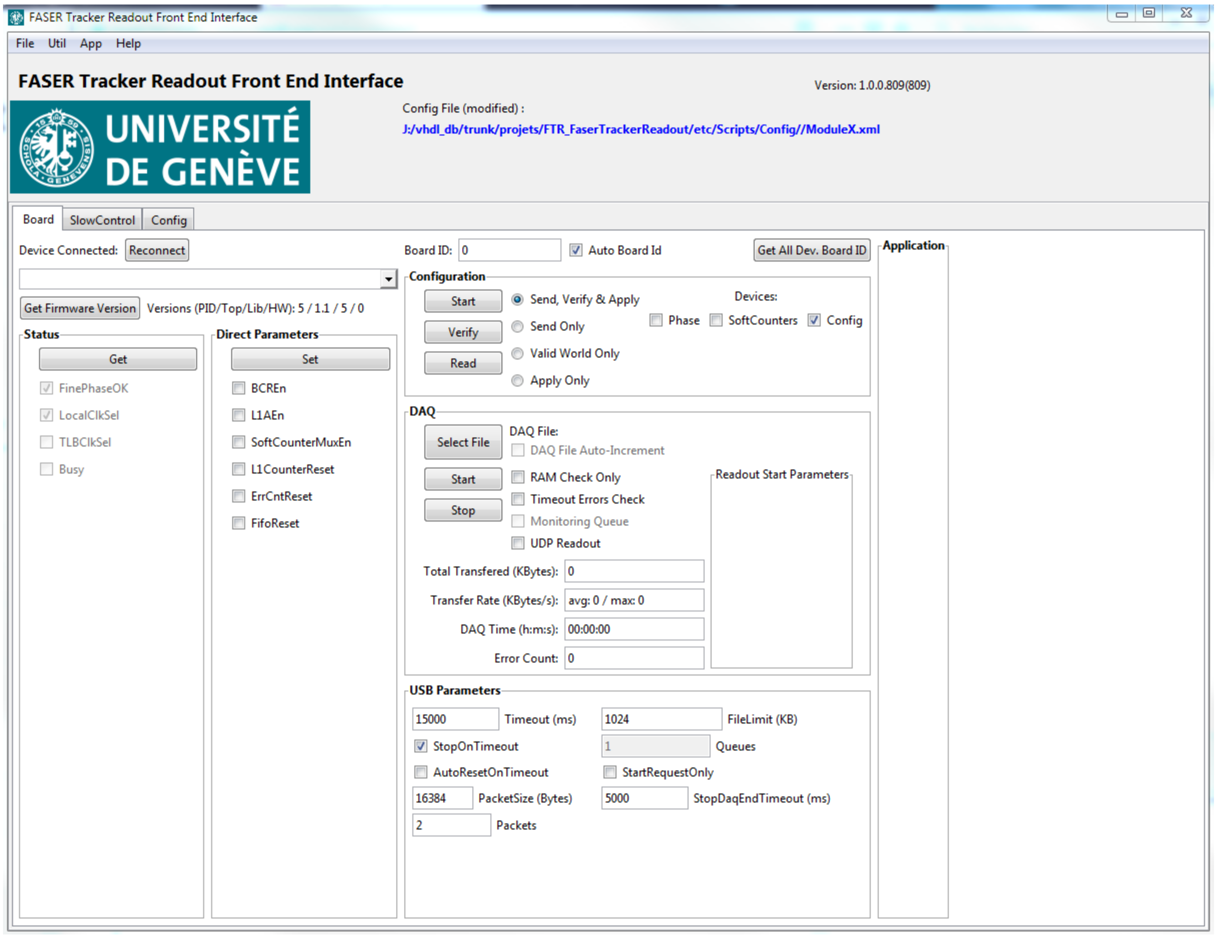
Device Connected: 0: BID= 0: @16: (0x206B - 0x0C00) Unige FX3 USB Readout Gateway

Versions (PID/Top/Lib/HW): 5 / 1.1 / 5 / 0

The versions should also appear (5/1.1/5/0) with 1.1 being the current FW version (and can change). You can have full text versions with the menu 'App/Version'.

The following pictures show the application and its tabs.

All parameters have an online 'tooltip' displayed when you stop the mouse on the parameter.



Version: 1.0.0.809(809)

Config File (modified) :
J:/vhdl_db/trunk/projets/FTR_FaserTrackerReadout/etc/Scripts/Config//ModuleX.xml

Device Connected:

Board ID: 0 ☑ Auto Board Id

Versions (PID/Top/Lib/HW): 5 / 1.1 / 5 / 0

Status

- ☑ FinePhaseOK
- ☑ LocalClkSel
- □ TLBClkSel
- □ Busy

Direct Parameters

- □ BCREn
- □ L1AEn
- □ SoftCounterMuxEn
- □ L1CounterReset
- □ ErrCntReset
- □ FifoReset

Configuration

- ● Send, Verify & Apply
- ○ Send Only
- ○ Valid World Only
- ○ Apply Only

Devices: □ Phase □ SoftCounters ☑ Config

DAQ

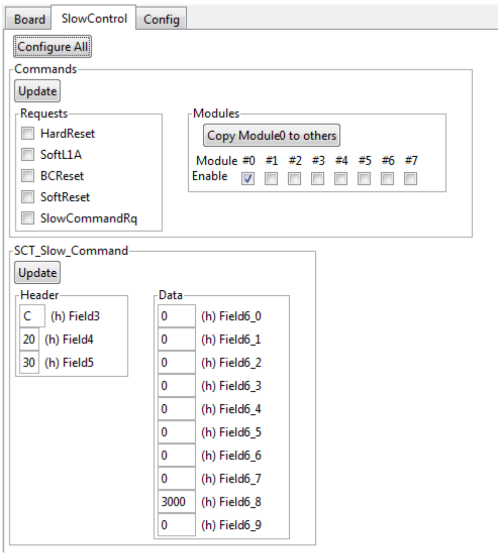
DAQ File:

- □ DAQ File Auto-Increment
- □ RAM Check Only
- □ Timeout Errors Check
- □ Monitoring Queue
- □ UDP Readout

Total Transfered (KBytes): 0
Transfer Rate (KBytes/s): avg: 0 / max: 0
DAQ Time (h:m:s): 00:00:00
Error Count: 0

USB Parameters

15000 Timeout (ms)
1024 FileLimit (KB)
☑ StopOnTimeout
1 Queues
□ AutoResetOnTimeout
□ StartRequestOnly
16384 PacketSize (Bytes)
5000 StopDaqEndTimeout (ms)
2 Packets



Commands

Requests

- □ HardReset
- □ SoftL1A
- □ BCReset
- □ SoftReset
- □ SlowCommandRq

Modules

Module	#0	#1	#2	#3	#4	#5	#6	#7
Enable	☑	□	□	□	□	□	□	□

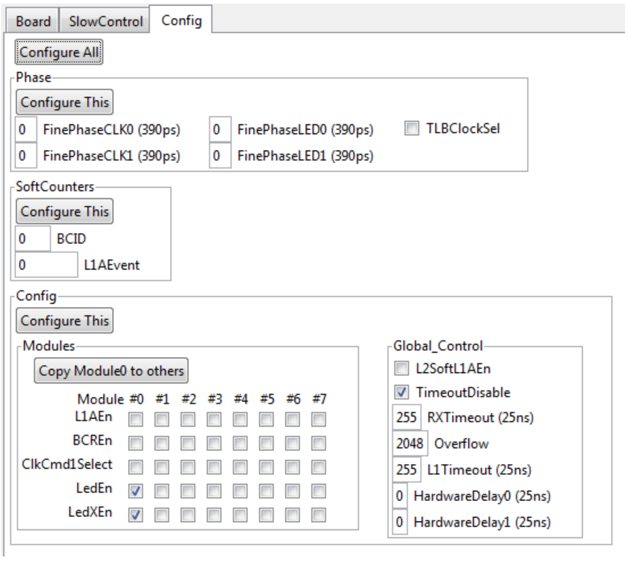
SCT_Slow_Command

Header

C	(h) Field3
20	(h) Field4
30	(h) Field5

Data

0	(h) Field6_0
0	(h) Field6_1
0	(h) Field6_2
0	(h) Field6_3
0	(h) Field6_4
0	(h) Field6_5
0	(h) Field6_6
0	(h) Field6_7
3000	(h) Field6_8
0	(h) Field6_9



Phase

0 FinePhaseCLK0 (390ps) | 0 FinePhaseLED0 (390ps) | □ TLBClockSel

0 FinePhaseCLK1 (390ps) | 0 FinePhaseLED1 (390ps)

SoftCounters

0 BCID

0 L1AEvent

Config

Modules

Module	#0	#1	#2	#3	#4	#5	#6	#7
L1AEn	□	□	□	□	□	□	□	□
BCREn	□	□	□	□	□	□	□	□
ClkCmd1Select	□	□	□	□	□	□	□	□
LedEn	☑	□	□	□	□	□	□	□
LedXEn	☑	□	□	□	□	□	□	□

Global_Control

- □ L2SoftL1AEn
- ☑ TimeoutDisable
- 255 RXTimeout (25ns)
- 2048 Overflow
- 255 L1Timeout (25ns)
- 0 HardwareDelay0 (25ns)
- 0 HardwareDelay1 (25ns)

2 Hardware commissioning

2.1 Basic TRB/PP check – no module

1. Connect the Firefly cable to the TRB/M0 connector and to the patch panel module connector to test
2. Disconnect the Module Flex connectors on the PP.
3. Switch ON GPIO and PP 5V PS
4. Apply VDD = 4V on PP for module 0
5. Check the following signals on the patch panel:
    a. Select line : < 0.4V when GUI 'Config' Tab ClkCmd1Select[0] is unchecked
    b. Select line : >3.6V when GUI 'Config' Tab ClkCmd1Select[0] is checked
6. Unchecked GUI ClkCmd1Select[0] in the 'Config' Tab (Select line = 0)
7. On GUI 'Slow Control' Tab/'Commands' area, check Module #0 Enable
8. Put a scope probe on CLK0 and CMD0 signals **before** the PP transceiver and verify that you see the CLK running at 40MHz and also the L1A, BCReset and SoftReset bit pulses '110', '1010010' and '1010100' when checking the corresponding Requests SoftL1A, BCReset and SoftReset and pushing the Button 'Update' in the GUI 'Commands' area. The levels should be compatible with LVDS levels
9. Do the same for CLK0 and CMD0 signals **after** the PP transceiver. The signals should be less nice since no 100 ohms is present
10. Checked GUI ClkCmd1Select[0] in the 'Config' Tab (Select line = 1)
11. Put a scope probe on CLK1 and CMD1 signals and do the same verification than 8. and 9.

NB: Reset line cannot be tested at this stage since a pullup resistor is required when the module is disconnected.

12. Do the same when you'll use the other modules (M1 to M7)

2.2 Basic TRB/PP check with module

1. From 2.1 done, connect the Module Flex connectors on the PP.
2. Switch ON GPIO and PP 5V PS
3. Apply VDD = 4V on PP for module 0
4. Check with a scope probe the following signals on the patch panel (needed again since now module loads the open drain of the drivers) :
    a. Select line: < 0.4V when GUI ‘Config’ Tab ClkCmd1Select[0] is unchecked
    b. Select line: >3.6V when GUI ‘Config’ Tab ClkCmd1Select[0] is checked
    c. Reset line: you should see a low level pulse (active low) of ~1us (sharp falling edge, long rising edge) when checking the Request ‘HardReset’ and pushing the Button ‘Update’ in the GUI ‘Commands’ area. Verify that high level is >3.6V and low level is <0.4V
5. Unchecked GUI ClkCmd1Select[0] in the ‘Config’ Tab (Select line = 0)
6. On GUI ‘Slow Control’ Tab/’Commands’ area, check Module #0 Enable
7. Put a scope probe on CLK0 and CMD0 signals **after** the PP transceiver and verify that you see the CLK running at 40MHz and also the L1A, BCReset and SoftReset bit pulses ‘110’, ‘1010010’ and ‘1010100’ when checking the corresponding Requests ‘SoftL1A’, ‘BCReset’ and ‘SoftReset’ and pushing the Button ‘Update’ in the GUI ‘Commands’ area. The levels should be compatible with LVDS levels
8. Put a scope probe on LED and LEDX signal, you should see a signal running at 20MHz.

9. Do the same when you’ll use the other modules (M1 to M7)

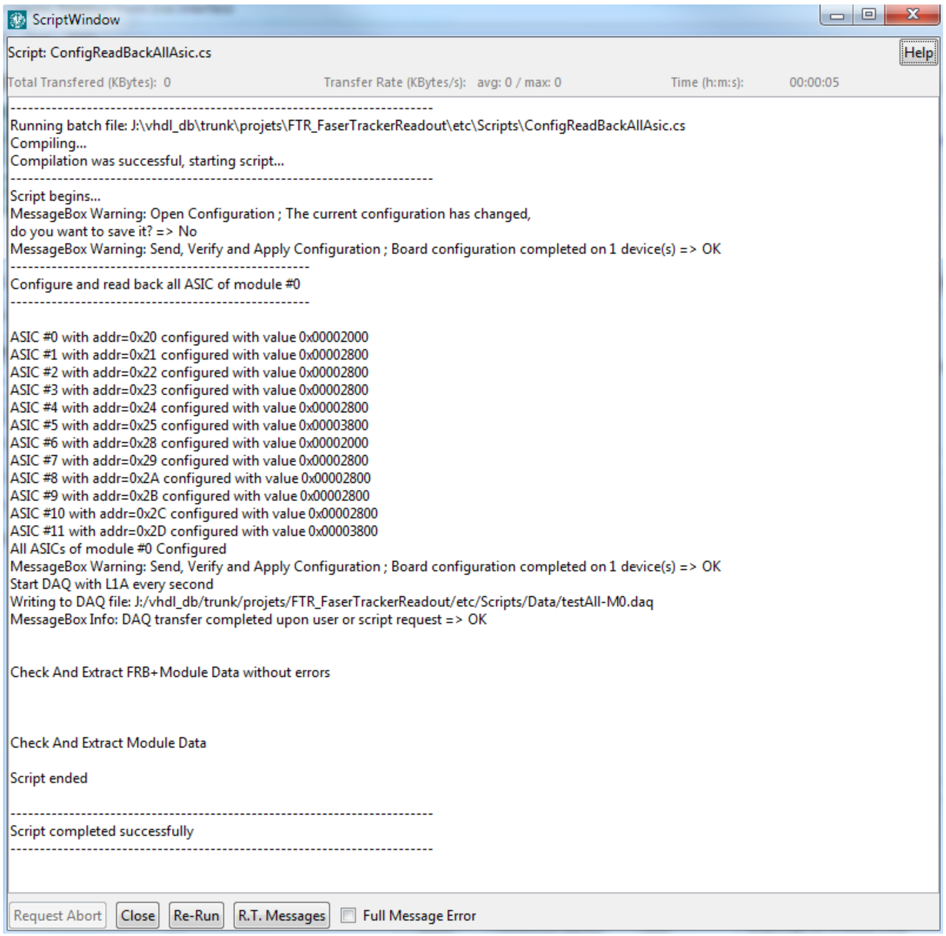
3 Software commissioning

3.1 Preliminaries

You will use now the C# scripts in the Unige GUI software.
The scripts are located on http://dpnc.unige.ch/~favrey/FASER/TRB/FrontEnd/Scripts/

1. Create a local directory on your computer ‘*somewhere*/Script’ and download the scripts .cs of the directory.
2. In your ‘Script’ directory create a local directory on your computer ‘*somewhere*/Script/Config’ and download the xml configs.
3. Create a local directory on your computer ‘*somewhereElse*/Data’, that’s where the script data will be put.

In order to run a script, do Menu ‘File/Run Script‘ and choose the script you want to run. Recent scripts are also memorized in ‘File/Recent Scripts‘. The screen should be like the following figure.



Script: ConfigReadBackAllAsic.cs

Total Transfered (KBytes): 0 Transfer Rate (KBytes/s): avg: 0 / max: 0 Time (h:m:s): 00:00:05

```
----------------------------------------------------------------------
Running batch file: J:\vhdl_db\trunk\projets\FTR_FaserTrackerReadout\etc\Scripts\ConfigReadBackAllAsic.cs
Compiling...
Compilation was successful, starting script...
----------------------------------------------------------------------
Script begins...
MessageBox Warning: Open Configuration ; The current configuration has changed,
do you want to save it? => No
MessageBox Warning: Send, Verify and Apply Configuration ; Board configuration completed on 1 device(s) => OK
--------------------------------------------------
Configure and read back all ASIC of module #0
--------------------------------------------------

ASIC #0 with addr=0x20 configured with value 0x00002000
ASIC #1 with addr=0x21 configured with value 0x00002800
ASIC #2 with addr=0x22 configured with value 0x00002800
ASIC #3 with addr=0x23 configured with value 0x00002800
ASIC #4 with addr=0x24 configured with value 0x00002800
ASIC #5 with addr=0x25 configured with value 0x00003800
ASIC #6 with addr=0x28 configured with value 0x00002000
ASIC #7 with addr=0x29 configured with value 0x00002800
ASIC #8 with addr=0x2A configured with value 0x00002800
ASIC #9 with addr=0x2B configured with value 0x00002800
ASIC #10 with addr=0x2C configured with value 0x00002800
ASIC #11 with addr=0x2D configured with value 0x00003800
All ASICs of module #0 Configured
MessageBox Warning: Send, Verify and Apply Configuration ; Board configuration completed on 1 device(s) => OK
Start DAQ with L1A every second
Writing to DAQ file: J:/vhdl_db/trunk/projets/FTR_FaserTrackerReadout/etc/Scripts/Data/testAll-M0.daq
MessageBox Info: DAQ transfer completed upon user or script request => OK


Check And Extract FRB+Module Data without errors



Check And Extract Module Data

Script ended

----------------------------------------------------------------------
Script completed successfully
----------------------------------------------------------------------
```

All standard C# functions are accessible and very well documented on the web (type C# followed by ‘*what do you wanna do in C#*’ on google)
In order to access to the hardware, you can have help either directly in the script window (‘Help’ button) or from Menu ‘Help/Scripting’. Each function must be preceded by its library in the C# script e.g. Sync.Sleep(10), BoardLib.isTranferingData …

3.2 Configuration read back

1. With a text editor (ideally recognizing C# files, like NotePad++), open 'ConfigReadBackAllAsic.cs' file.
2. The 2 1st lines (*l_cfgFile* and *l_daqfile*) correspond to the locations of the config file ('*somewhere*/Script/Config') and the data directory ('*somewhereElse*/Data'). Change the directory of these variables accordingly but do not change the config file name (ModuleX.xml).
3. You can eventually change the *l_daqfile* file name (*testAll* in the script)
4. Run this script that will:
    a. ask you to provide the module number you're going to test [0..7]
    b. configure all the 12 ASICs of the selected module
    c. launch 5 L1A triggers in order to read back the ASIC configuration and save it into a binary .daq file
    d. decode and check the FRB data from .daq file and write a report into a CSV file
    e. write 2 .mod files, one for the LED line and another for the LEDX line, both having 6 ASICs binary data for the corresponding line
    f. decode and check the ASICs data from .mod files and write a report into a CSV file
5. You should get on the screen what is in http://dpnc.unige.ch/~favrey/FASER/TRB/FrontEnd/Scripts/Data/ConfigReadBackAllAsicScriptReport.txt
6. The data written in you Data directory should be the **same** than in http://dpnc.unige.ch/~favrey/FASER/TRB/FrontEnd/Scripts/Data/ConfigReadBackAllAsic.zip
7. Verify the correspondence of these files and your data written

3.3 Data taking basic mode

1. With a text editor open 'DataTaking.cs' file.
2. The 2 1st lines (*l_cfgFile* and *l_daqfile*) correspond to the locations of the config file ('*somewhere*/Script/Config') and the data directory ('*somewhereElse*/Data'). Change the directory of these variables accordingly but do not change the config file name (ModuleX.xml).
3. You can eventually change the *l_daqfile* file name (*testDT* in the script)
4. You can modify the module number under test ('l_moduleEn' variable)
5. Run this script that will:
    a. configure all the 12 ASICs of the module
    b. set them in data taking mode
    c. launch 5 L1A triggers and save data packets into a binary .daq file
    d. decode and check the FRB data from .daq file and write a report into a CSV file
    e. write 2 .mod files, one for the LED line and another for the LEDX line, both having 6 ASICs binary data for the corresponding line
    f. decode and check the ASICs data from .mod files and write a report into a CSV file
6. You should get on the screen what is in http://dpnc.unige.ch/~favrey/FASER/TRB/FrontEnd/Scripts/Data/DataTakingScriptReport.txt
7. The data written in you Data directory should be the **similar** (depends on module noise) than in http://dpnc.unige.ch/~favrey/FASER/TRB/FrontEnd/Scripts/Data/DataTaking.zip

8. Verify the correspondence of these files and your data written

3.4 Threshold scan preliminary

9. With a text editor open ‘ThresholdScanAsic.cs’ file.
10. **Create a ‘scan’ directory in your ‘*somewhereElse*/Data’ directory**
11. The 2 1st lines (*l_cfgFile* and *l_daqfile*) correspond to the locations of the config file (‘*somewhere*/Script/Config’) and the data directory (‘*somewhereElse*/Data’). Change the directory of these variables accordingly but do not change the config file name (ModuleX.xml).
12. You can eventually change the *l_daqfile* file name (*testAll* in the script)
13. You can modify the module number under test (‘l_moduleEn’ variable)
14. You can modify the threshold scan loop parameters (next lines …)
15. Run this script that will:
    a. configure all the 12 ASICs of the module
    b. do the threshold loop
    c. do the chip address loop
    d. do calibration mode loop
    e. configure threshold and calibration value register
    f. configure config. register for calibration mode
    g. enable data taking mode
    h. prepare calibration pulse and delay command
    i. launch 5 calibration commands and its L1A triggers and save data packets into a binary .daq file
    j. decode and check the FRB data from .daq file and write a report into a CSV file
    k. write 2 .mod files, one for the LED line and another for the LEDX line, both having 6 ASICs binary data for the corresponding line
    l. decode and check the ASICs data from .mod files and write a report into a CSV file
    m. close the 3 loops
16. You should get on the screen what is in http://dpnc.unige.ch/~favrey/FASER/TRB/FrontEnd/Scripts/Data/ThresholdScanAsicScriptReport.txt
17. The data written in you Data directory should be the **similar** (depends on module noise) than in http://dpnc.unige.ch/~favrey/FASER/TRB/FrontEnd/Scripts/Data/ThresholdScanAsic.zip
18. Verify the correspondence of these files and your data written