Fourth short status update of FPGA based DAQ for the read out of SiPM

Cristian Alejandro Fuentes Rojas

cafuente@cern.ch

ADC

10 x LVDS / ADC

8 data LVDS

out_clk LVDS

start_out LVDS

command

slow_clk

out_clk LVDS

SERDES + word boundary detection

80 bits / ADC

7 bits address

clk sample time

counter: Sample Nº

7 bits

counter: Time

64 bits

SPI

Reference clock generator

0

BRAM

N_T-16

N_T-14

N_T

127

7 bits address

Trigger Detector

counter: Trigger/Event

32 bits

Event Packet Builder

32 bits

7 bits (windows size)

IPBUS Firmware

0

FIFO

1023

Computer Linux

uHAL C++

Trigger

Signal

Samples

N_T-11

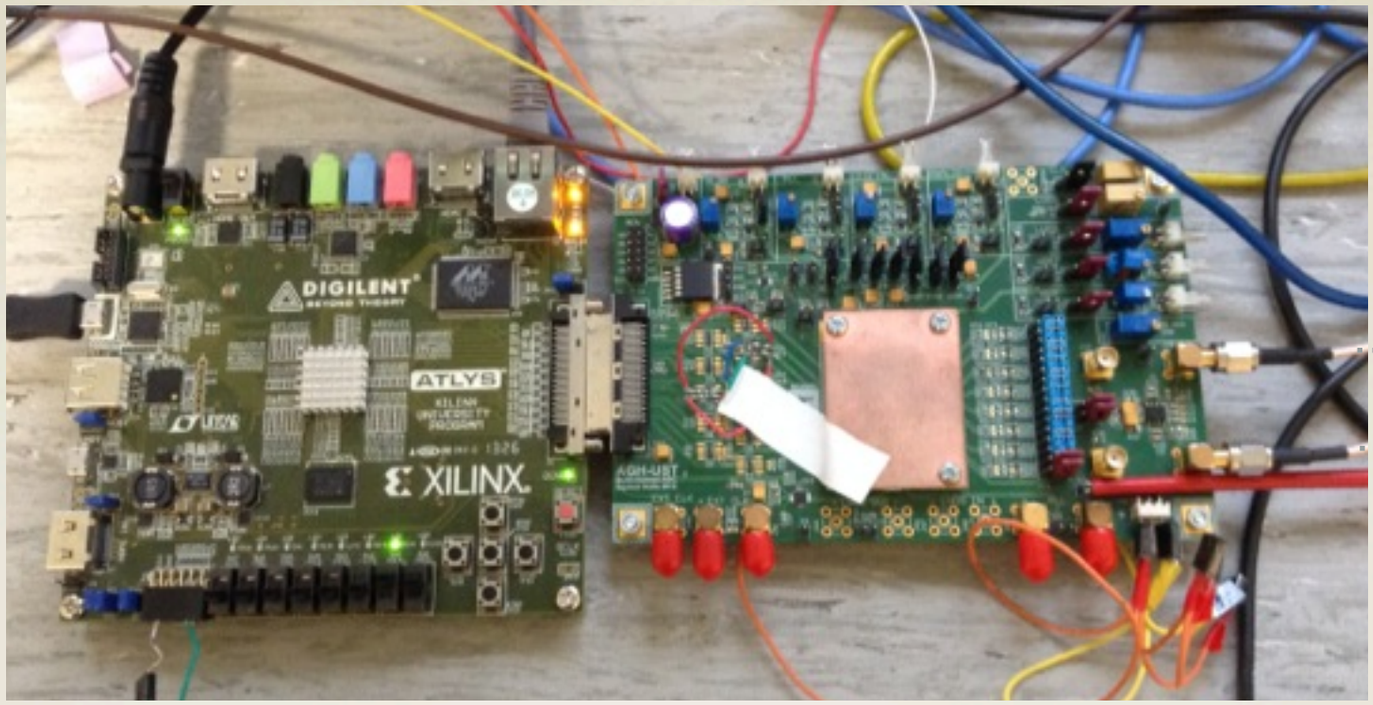
N_T-10

N_T-16 N_T-14

N_T-5

N_T

From the user's point of view..



DUT

Register Slave 0

bit 0	used by IPBUS	
bit 1	enable SPI	(input)
bits 2 to 21	command SPI	(input)
bits 22 to 31	Free	

Register Slave 1

bit 0	empty_FIFO	(output)
bit 1	full_FIFO	(output)
bit 2	valid_data_FIFO	(output)
bit 3	underflow_FIFO	(output)
bits 4 to 13	Free	
bits 14 to 17	Smp. freq.	(input)
bits 18 to 24	N	(input)
bits 25 to 31	Nb	(input)

One C++ code to initialise the register's values

data window

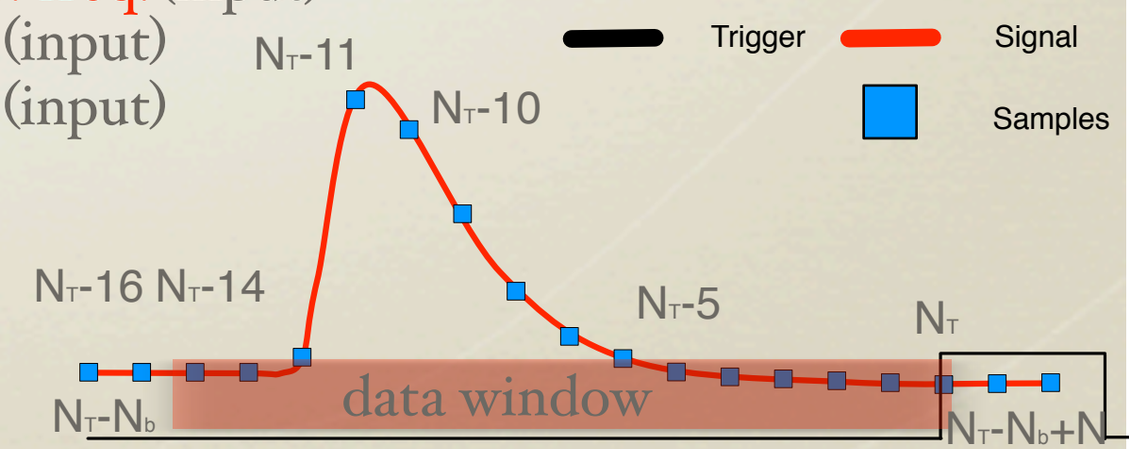
N_T-N_b N_T-N_b+N

where N_b>N

FIFO Slave 4

1023 words of 32 bits saved in a text file (output)

One C++ code to read FIFO and save results in a text file



Event Packet's Structure

HEADER
Event Number (32 bits)
Time Stamp (bits 63:32)
Time Stamp (bits 31:0)
Channel's mask
"00000000000" (11bits), N_b (7 bits) , N (7bits), N_T(7bits)
Samples:

Event Packet's Structure: Samples

INDX (0 .. N-1)

SMP

Ch (0 .. N_C-1)

where each sample has 10 bits

32 bits

SMP_0^0 10 bits	SMP_1^0	"00000"	Sample (7bits) BRAM Address
SMP_2^0 10 bits	SMP_3^0	"00000"	Sample (7bits) BRAM Address
SMP_4^0 10 bits	SMP_5^0	"00000"	Sample (7bits) BRAM Address

N x $\frac{N_C}{2}$ words

N: Number of samples (windows size)

N_C: Number of channels

Example of a generated text file

U... — Edited

Helvetica | Regular

ffff0001	HEADER
652f	Event Number
4	Time Stamp (bits 63:32)
1d3f5e83	Time Stamp (bits 31:0)
ff	Channel's mask
3c7ec	"00000000000" (11bits), Nb (7 bits) , N (7bits), NT(7bits)
fffff05e	Sample 0: channel 0 and channel 1 and their BRAM Address
fffff05e	Sample 0: channel 2 and channel 3 and their BRAM Address
fa3ff05e	Sample 0: channel 4 and channel 5 and their BRAM Address
fffff05e	Sample 0: channel 6 and channel 7 and their BRAM Address
fffff05f	Sample 1: channel 0 and channel 1 and their BRAM Address
fffff05f	Sample 1: channel 2 and channel 3 and their BRAM Address
fffff05f	Sample 1: channel 4 and channel 5 and their BRAM Address
fffff05f	Sample 1: channel 6 and channel 7 and their BRAM Address
fe3fc060	Sample 2: channel 0 and channel 1 and their BRAM Address
ffffe060	Sample 2: channel 2 and channel 3 and their BRAM Address
ffffc060	Sample 2: channel 4 and channel 5 and their BRAM Address
fe3fc060	Sample 2: channel 6 and channel 7 and their BRAM Address
fa3ec061	Sample 3: channel 0 and channel 1 and their BRAM Address
fd3f4061	Sample 3: channel 2 and channel 3 and their BRAM Address
fe7ec061	Sample 3: channel 4 and channel 5 and their BRAM Address
fa3f0061	Sample 3: channel 6 and channel 7 and their BRAM Address

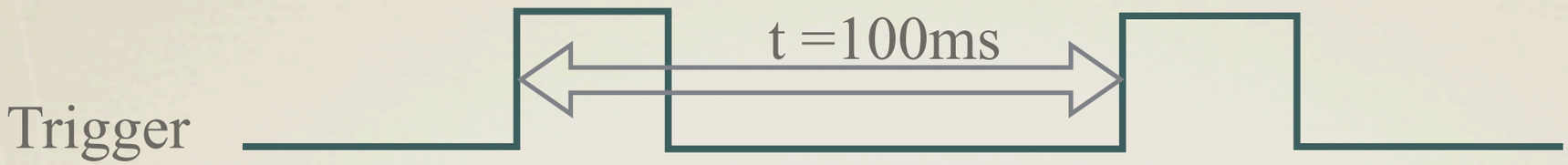
share the same BRAM Address

share the same BRAM Address

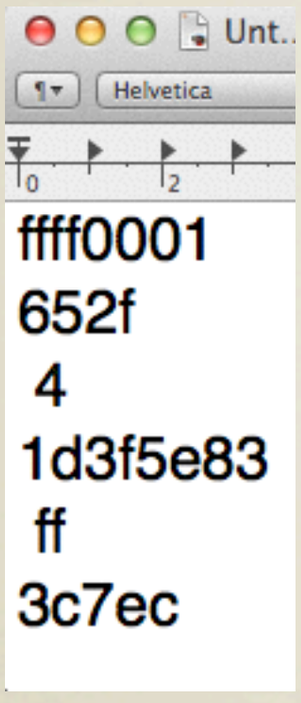
share the same BRAM Address

Trigger Debugging

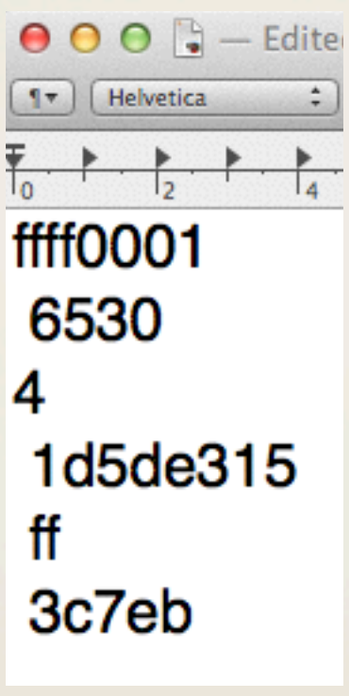
Inject a trigger signal with equally spaced pulses



See the created files with consecutive event packets. Subtract their time stamp.



HEADER
Event Number
Time Stamp (bits 63:32)
Time Stamp (bits 31:0)
Channel's mask



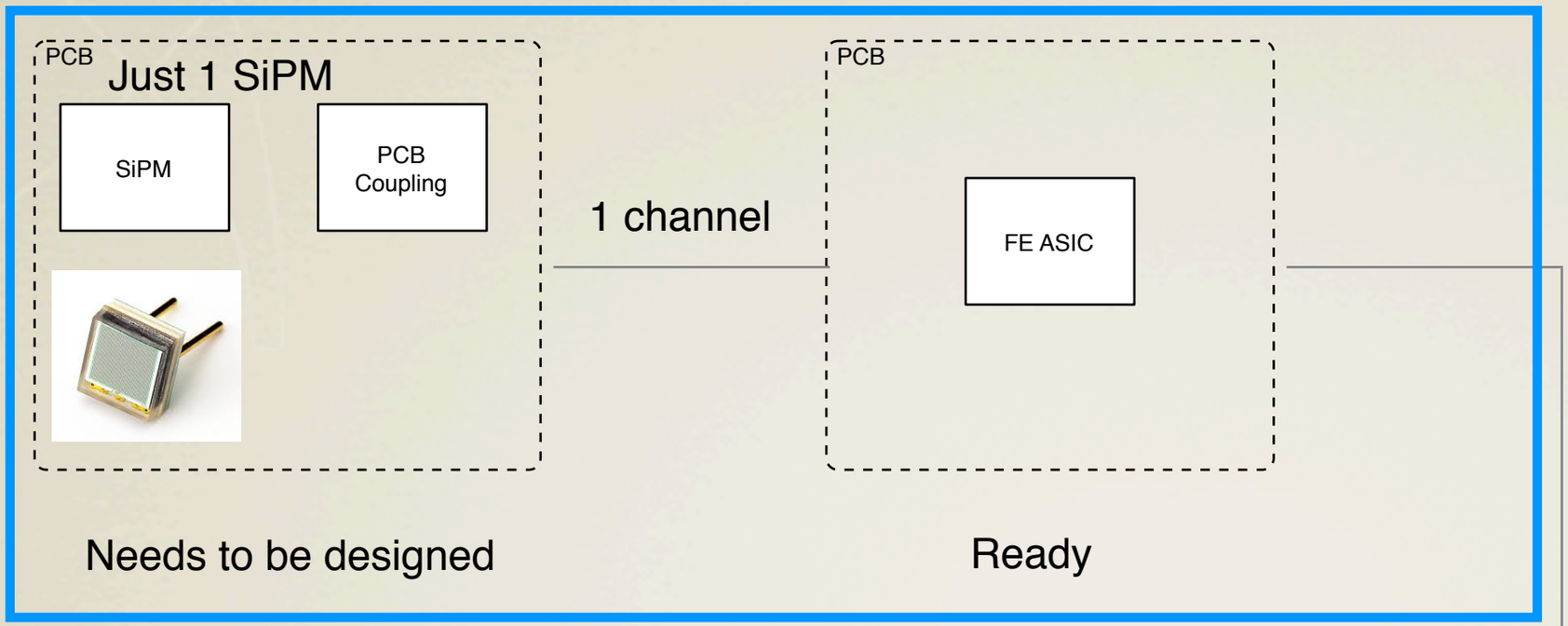
Hexadecimal:
1d5de315 - 1d3f5e83

decimal:
492692245-490692227=2000018

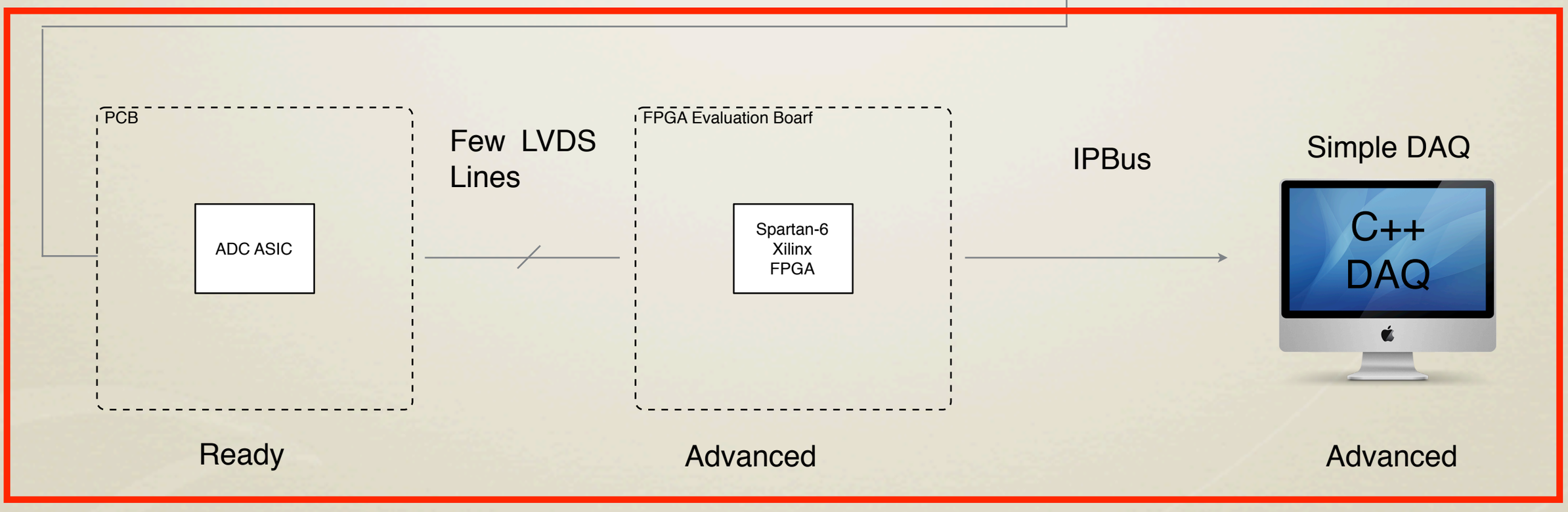
Divide the result by the sampling frequency

$$t = \frac{2000018}{20MHz} = 100ms$$

Integration test

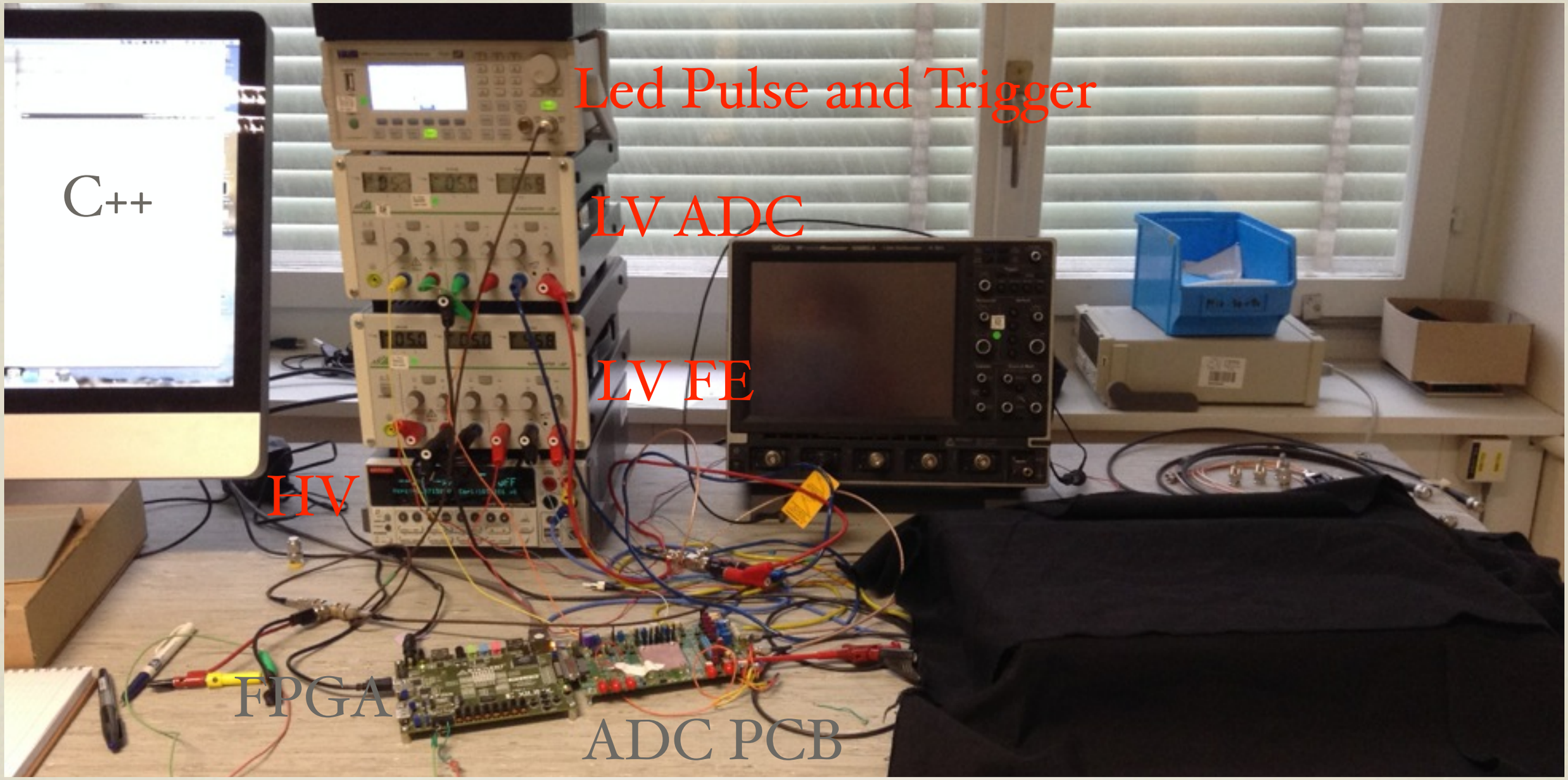


Done with Thibault



From my side

How it looks in the lab



√ I can see the pulse in the recorded text file.

√ I can adjust the windows size and position in order to reduce the amount of data read.

Calibration and Window finding

The user can define Nb and N before the data acquisition

```
Doing read...
Enter the # Nb between 1 and 127
80
Enter the # N between 2 and 120, and smaller than Nb
60
Enter the # to select the command to send to the ADC
1) 1010000000000000000 -- Normal operation
2) 1010000000000000010 -- Adds PLL ON to normal operation
3) 1010000000001100000 -- Internal counter with ordinary order of counting
4) 1010000000001000000 -- Internal counter with pseudo random order of counting
1
cafuente@ubuntu:~/clic_ecal_daq/IPBUS_Software/Trunk$
```

By setting Nb and N to 127 and 120, they can have the biggest window available. Then they can find the pulse and afterwords select a smaller window

Today I use a double output trigger generator to adjust the delay between the trigger and the signal to be read.
An internal delay for the trigger will be soon implemented.

Possible Future work

Make a GUI C++ user interface

Make a simple PCB to have the possibility to see 8 channels.

(Today in a simple configuration I can put the same SiPM to the 8 channels)

Improved the readout of the FIFO.

Modify sampling time from IPBUS C++ . I have assigned the register. I just have to add a few code lines.

Thanks :)