



Single AGET Module

P. Hellmuth B. Lachacinski A. Rebi

Plan

- SAM : Goal
- SAM : PCB
- SAM : Firmware
- SAM : Software
- SAM : Perspectives

SAM : GOAL

→ The creation of an autonomous acquisition system with few analogue channels, based on the GET experience.

→ Replacing the RCoBo module

→ Future of ML507

No longer manufactured

Virtex5 old generation

Move to ARM technology (Zynq)

Use FMC Standard on adapter board
(compatible with all evaluation boards)

→ SAM (Single AGET Module)

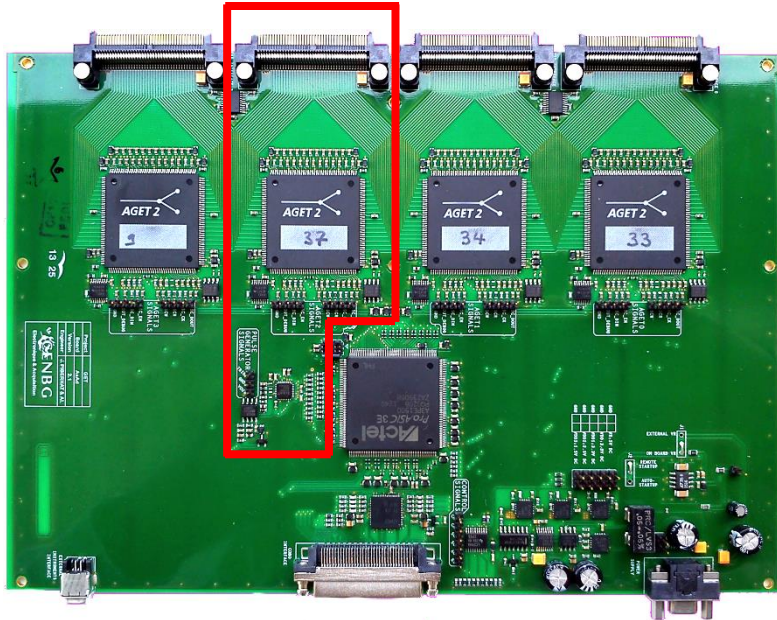
Used at CENBG for the DPR experiment

(Development of a gaseous proton recoil detector)

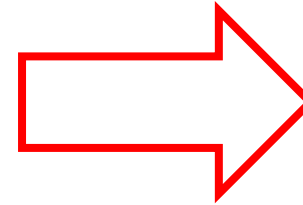
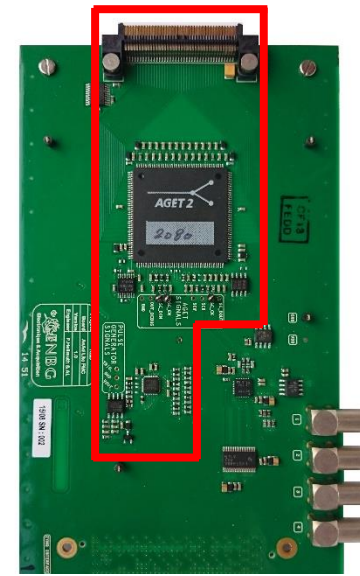


SAM : PCB

AsAd Board

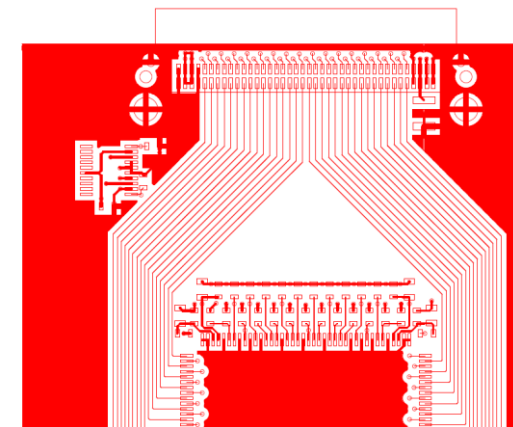


SAM Module

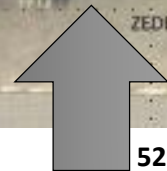
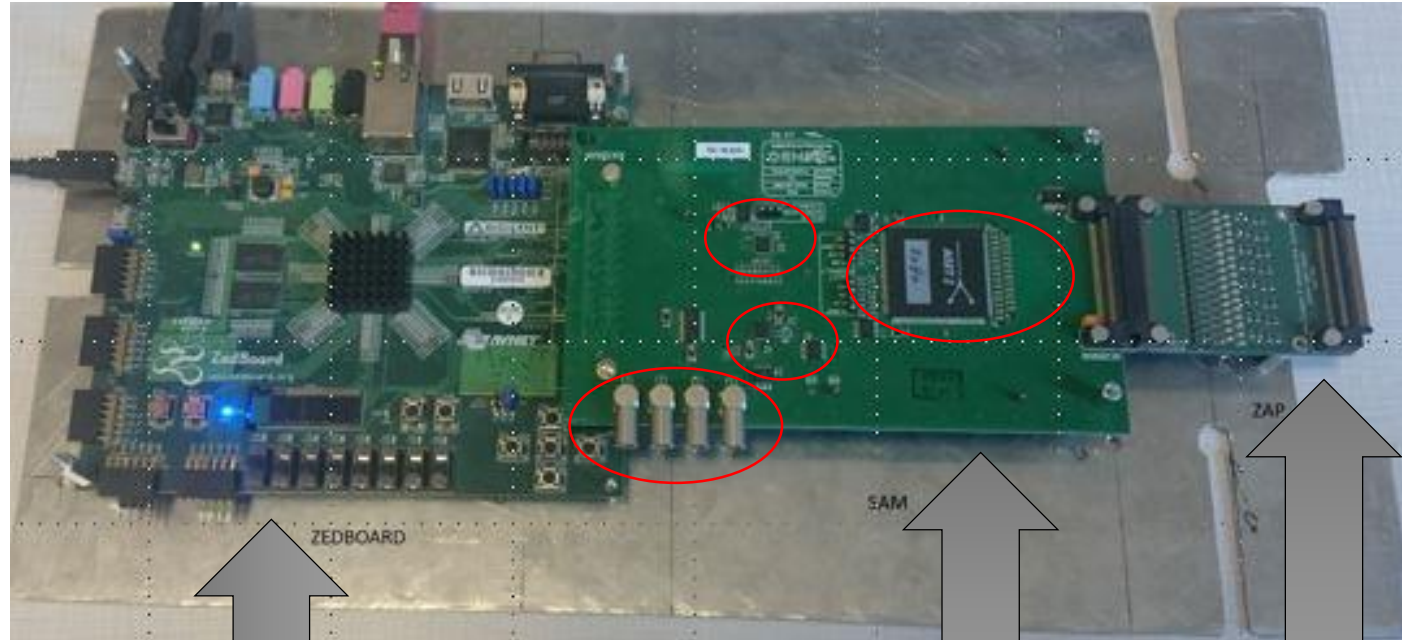


PCB 10 layers
Cross-section
Routing entries
Pulse routing

Identical to AsAd Board



SAM : PCB



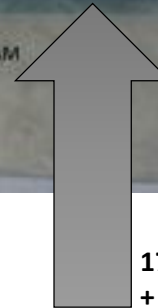
520 €

ZedBoard

Content :

Zynq-7000 AP SoC XC7Z020-CLG484
Dual-core ARM Cortex A9
Memory:
512 MB DDR3
256 Mb Quad-SPI Flash
4 GB SD card

Onboard USB-JTAG Programming
10/100/1000 Ethernet
USB OTG 2.0 and USB-UART
PS & PL I/O expansion (FMC, Pmod™,
XADC)
Multiple displays (1080p HDMI, 8-bit
VGA, 128 x 32 OLED)
I²S Audio CODEC



1700 €
+ 1200 if modif.

SAM

Content :

1 AGET
12 bit ADC ADS6124
Pulser with 14 bit DAC AD9707
4 Inspection lines



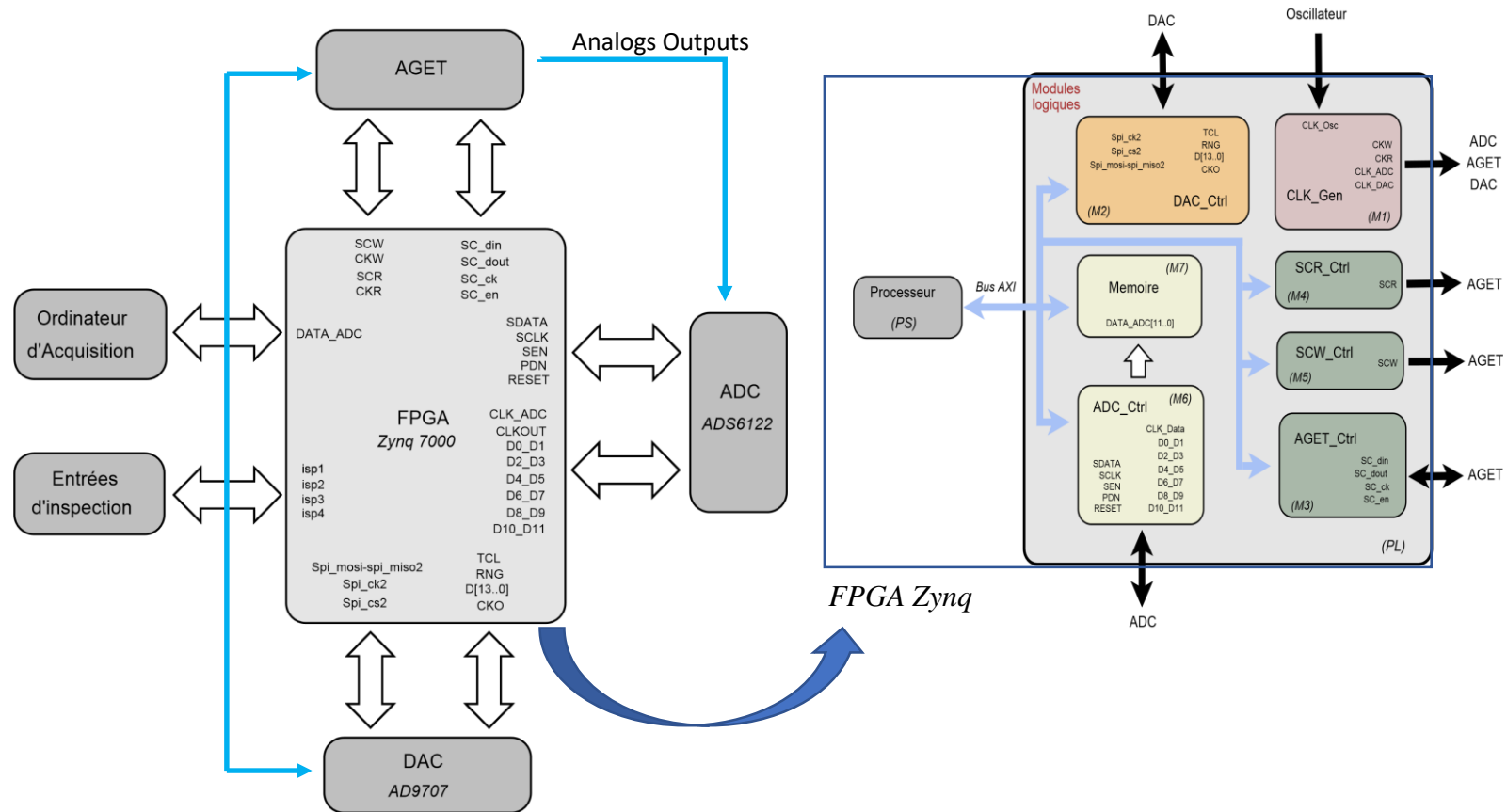
150 €

ZAP

Content :

64 inputs Analog Channels
64 protection circuits

SAM : Firmware



SAM : Firmware

➤ Clocks



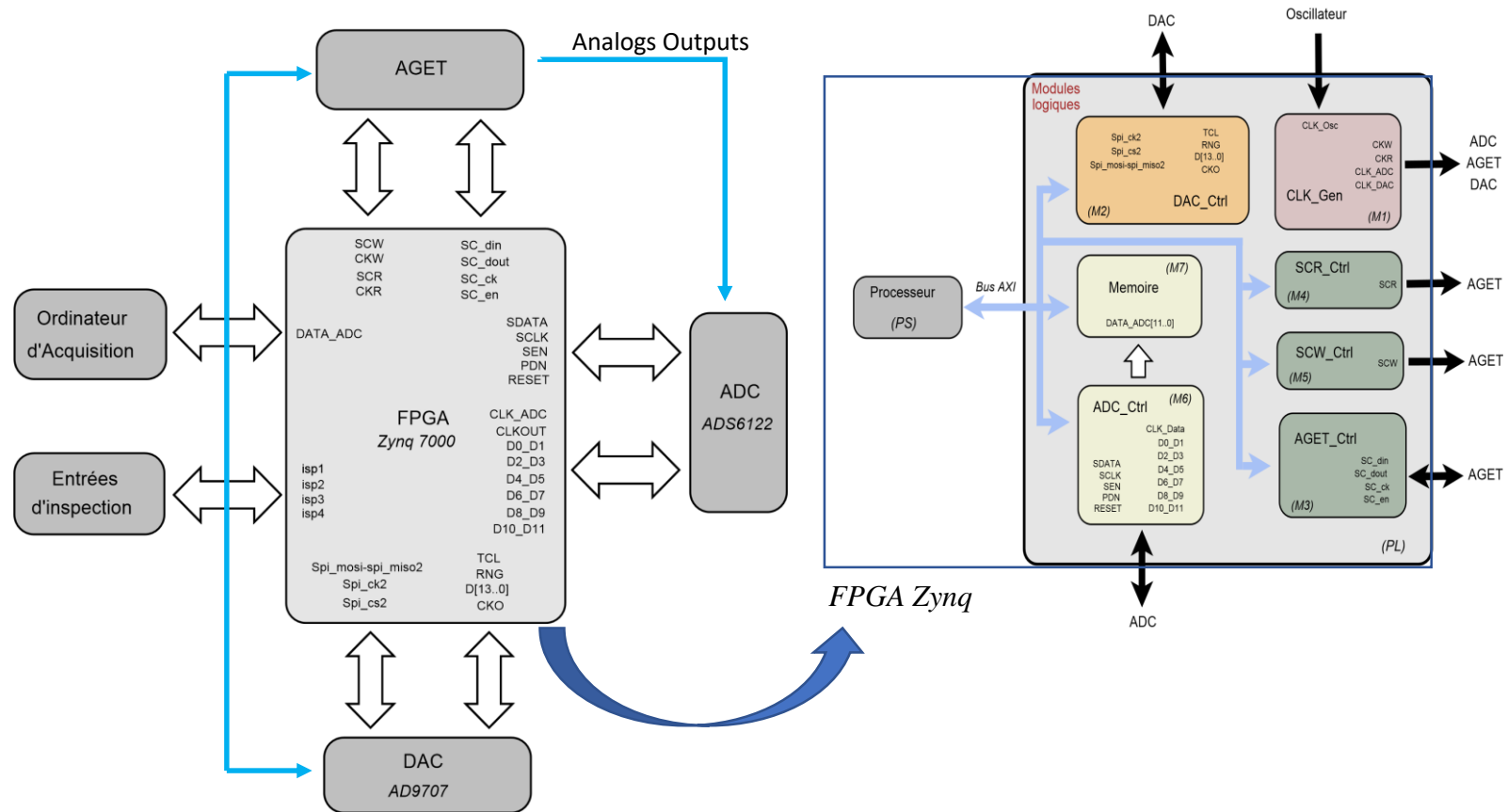
- Creates necessary clocks
 - Deserializer
 - AGET fast Control
 - AGET, ADC, DAC slow Control
 - AGET Sampling
 - AGET readout
- LogiCORE™ IP Clocking Wizard
- Frequency equation :

$$F_{OUT} = F_{IN} * \frac{Mult_Counter}{Divide_Counter * CLKout_Divider}$$

CLKout_Divider Register

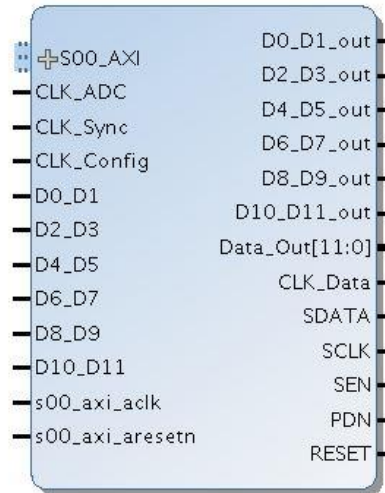
Bit[31 ..19]	Bit[18]	Bit[17:8]	Bit [7..0]
X	Fractional part enable ?	Fractional part	Integer part

SAM : Firmware



SAM : Firmware

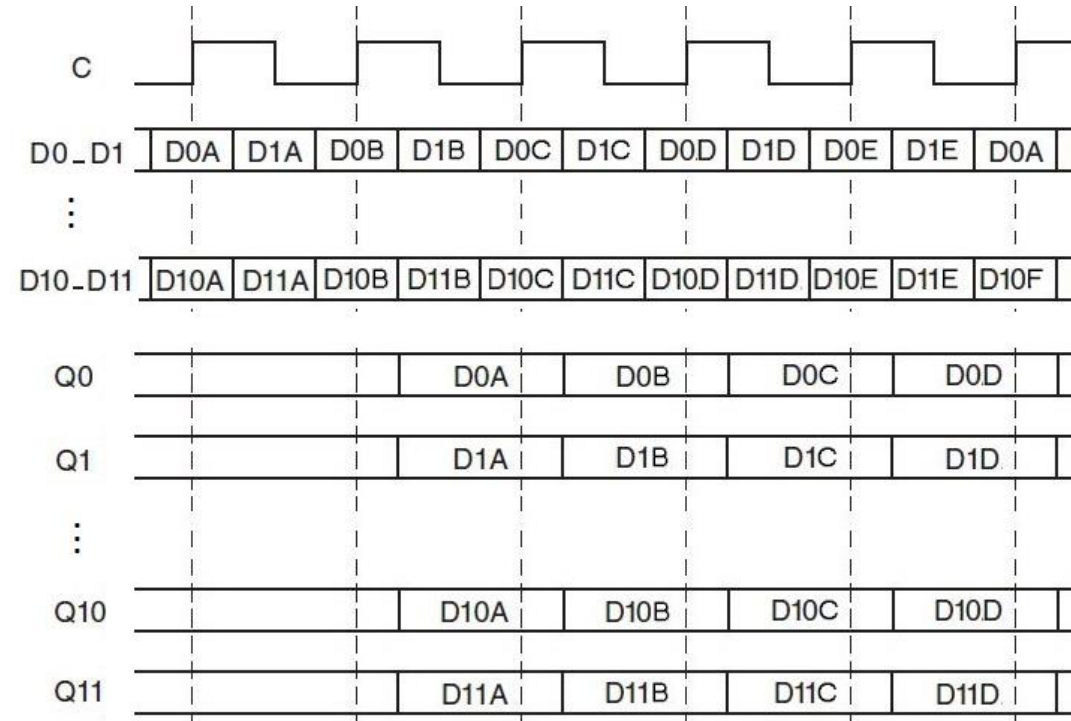
➤ IP Deserializer



Datas from the ADC

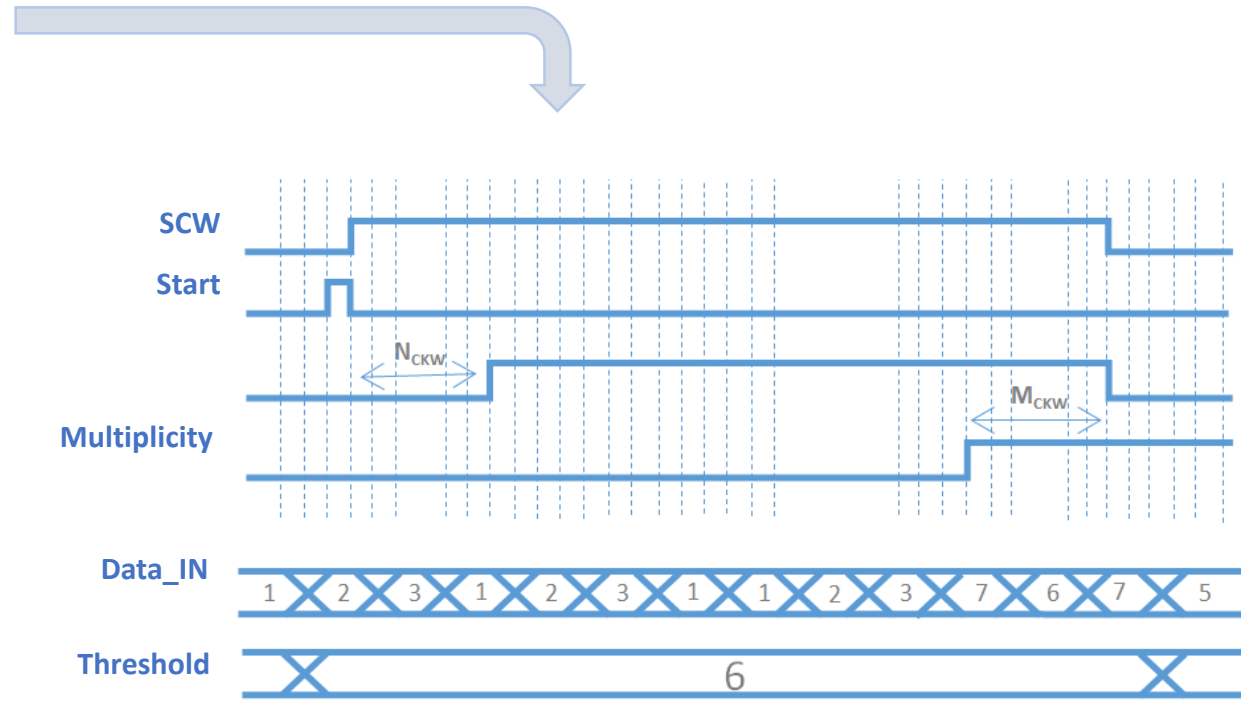
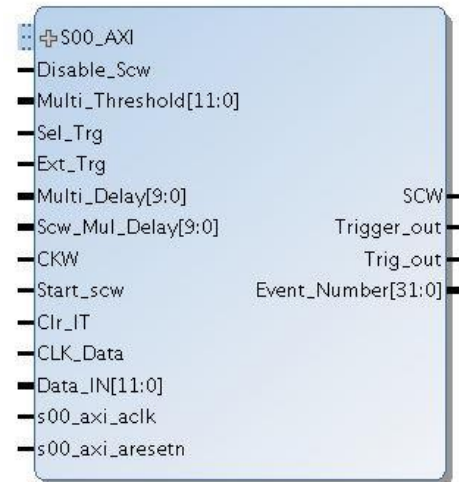
- 6 parallel inputs
 - 2 serial bits
 - Dual Data Rate

IDDR Primitive



SAM : Firmware

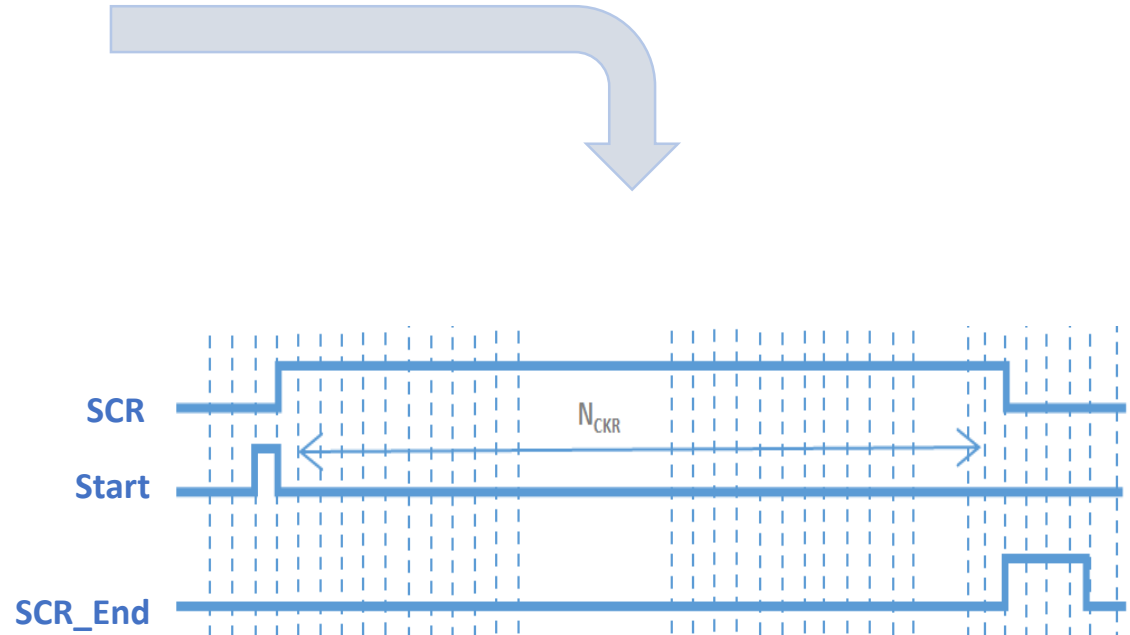
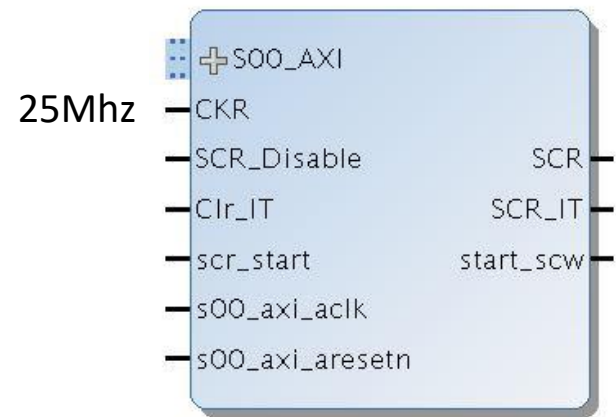
- SCA manager
 - SCW_Module
 - Creates enable write signal
 - Multiplicity detection



SAM : Firmware

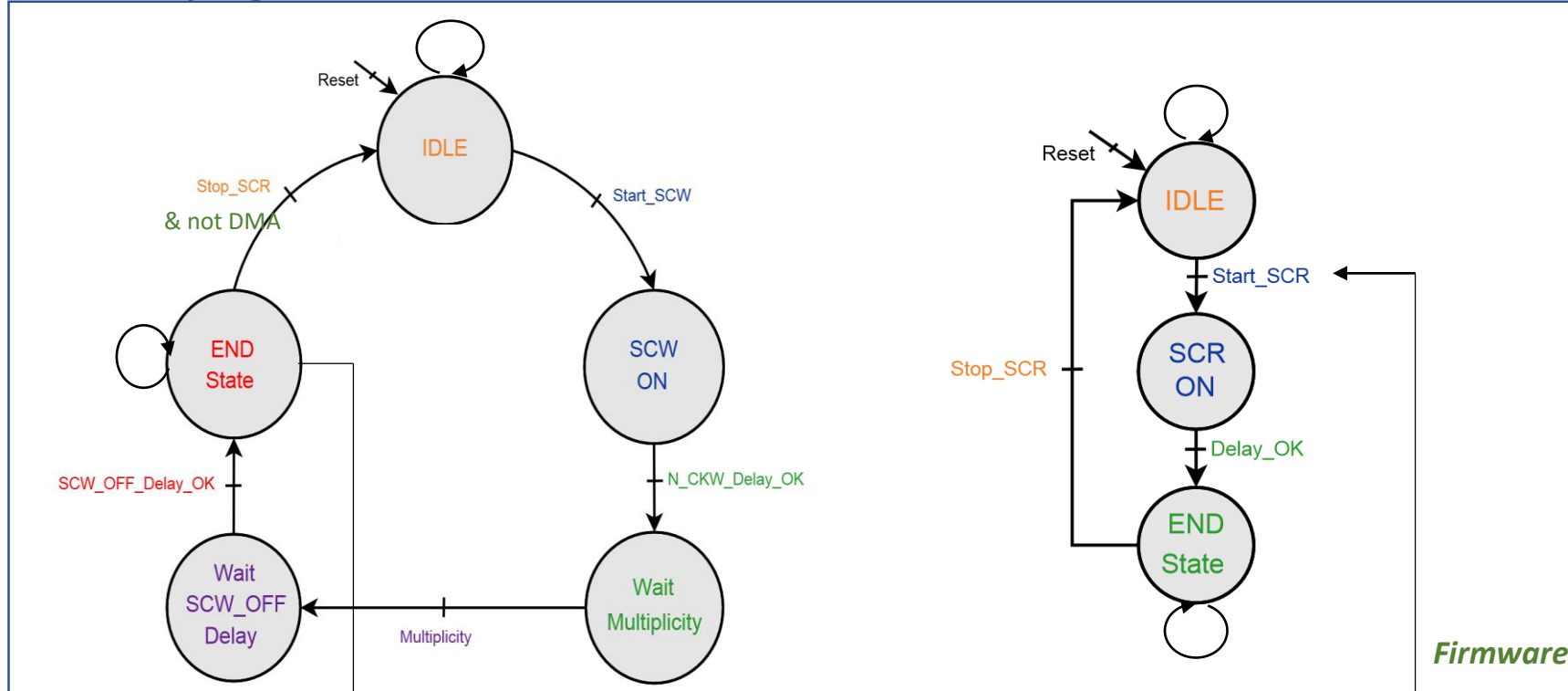
➤ SCR_Module

- Creates enable read signal

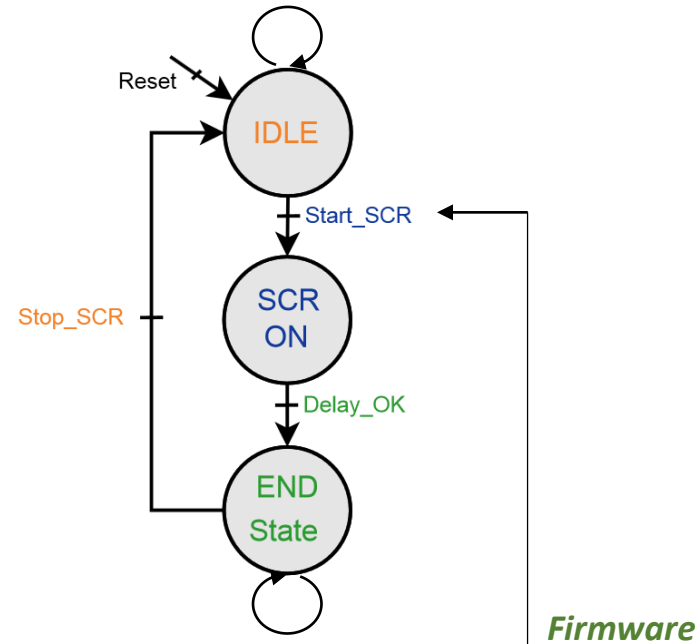


SAM : Firmware

AGET Sampling : SCW Controller



AGET Readout : SCR Controller



Firmware

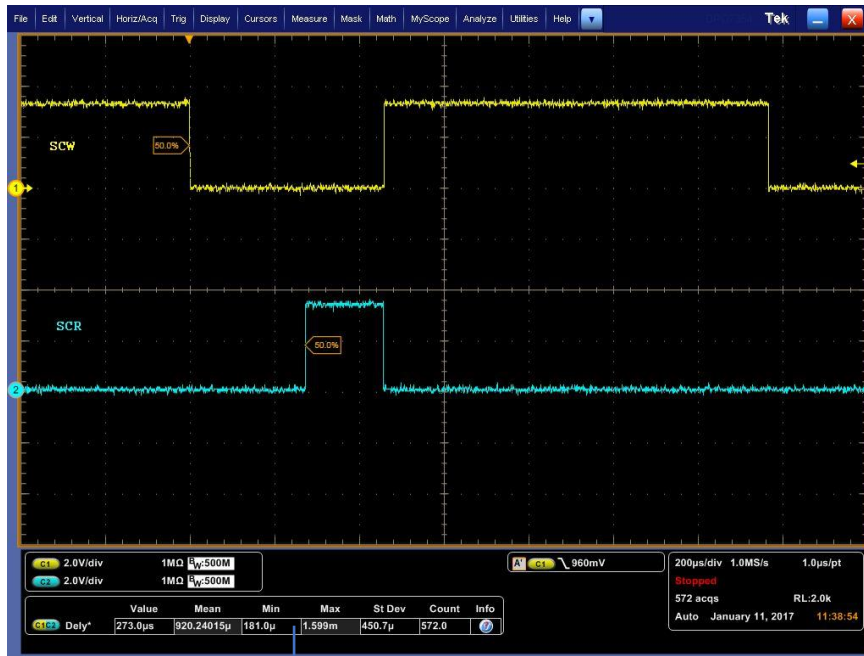
Hit Manager



Software

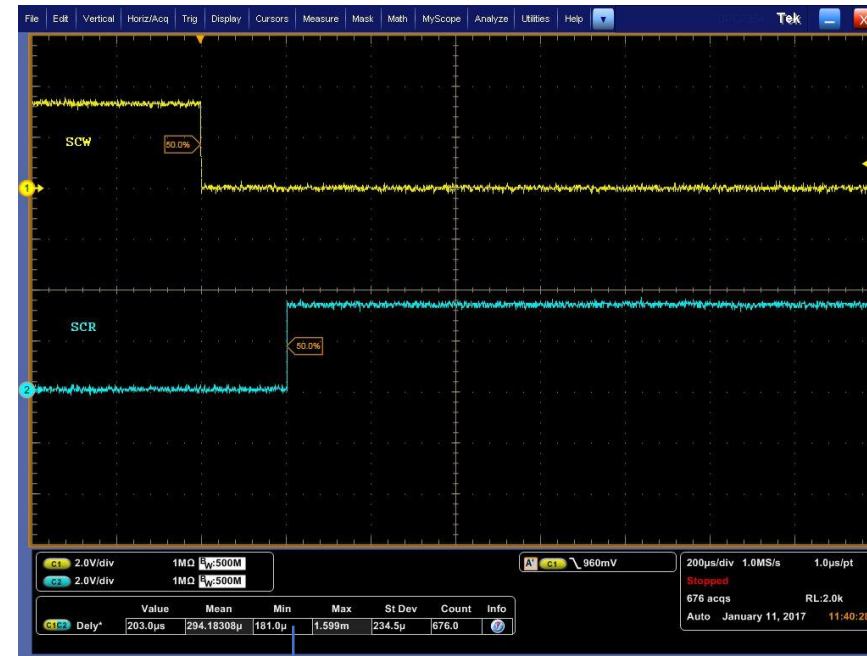
SAM : Firmware

Partial readout



	Value	Mean	Min	Max
C1C2 Dely*	273.0µs	920.24015µ	181.0µ	1.599m

Full readout

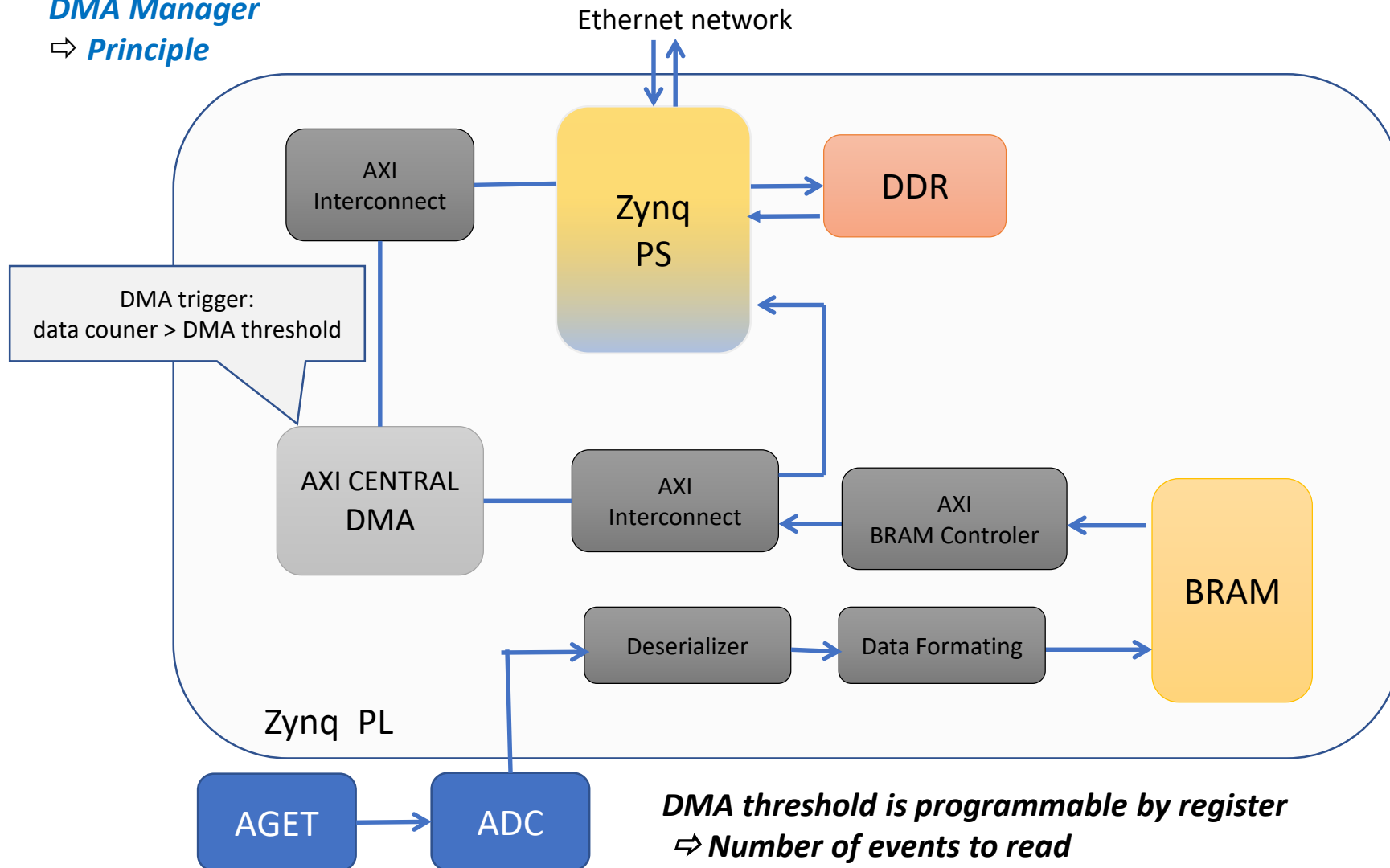


	Value	Mean	Min	Max
C1C2 Dely*	203.0µs	294.18308µ	181.0µ	1.599m

SAM : Firmware

DMA Manager

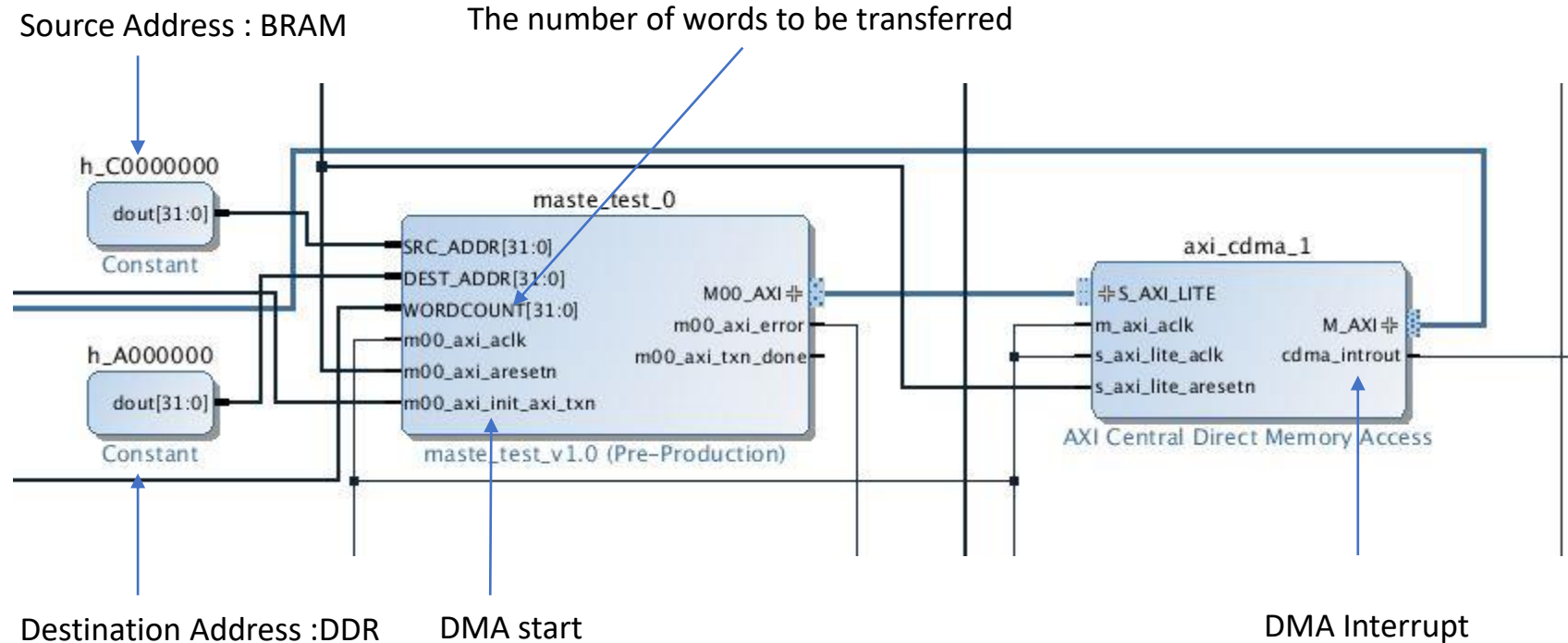
⇒ *Principle*



SAM : Firmware

DMA Manager

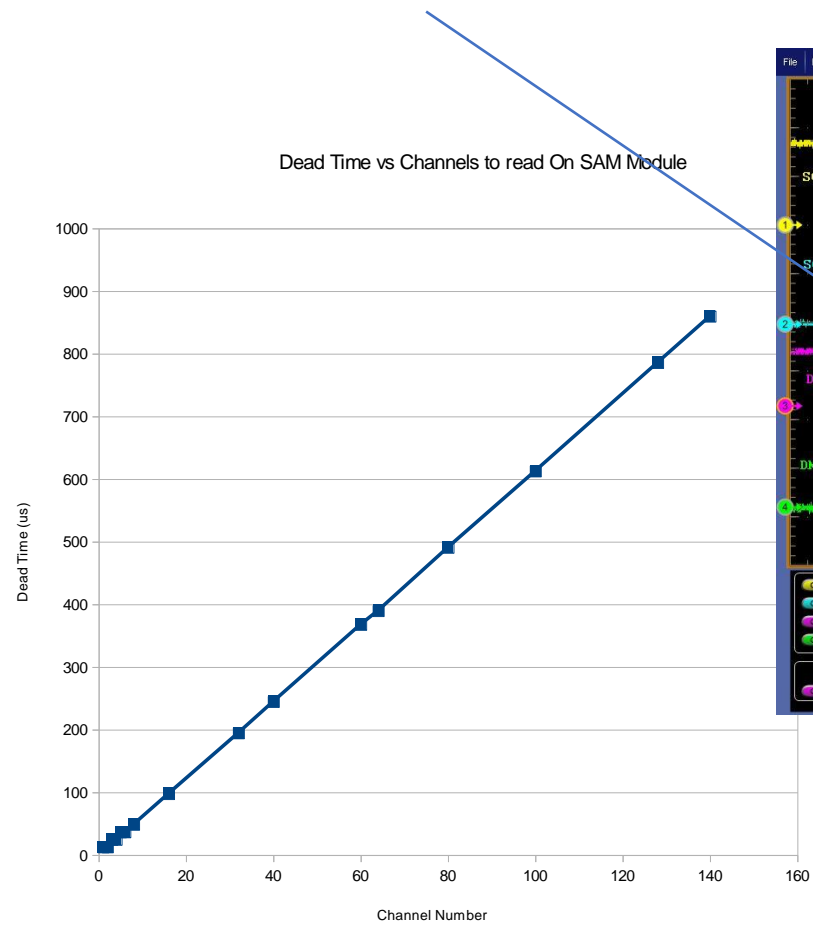
⇒ *In Vivado*



- ⇒ **The addresses (Source & destination) are fixed in the firmware**
- ⇒ **The word counter, the threshold comparator and the start of the DMA are implemented in firmware**

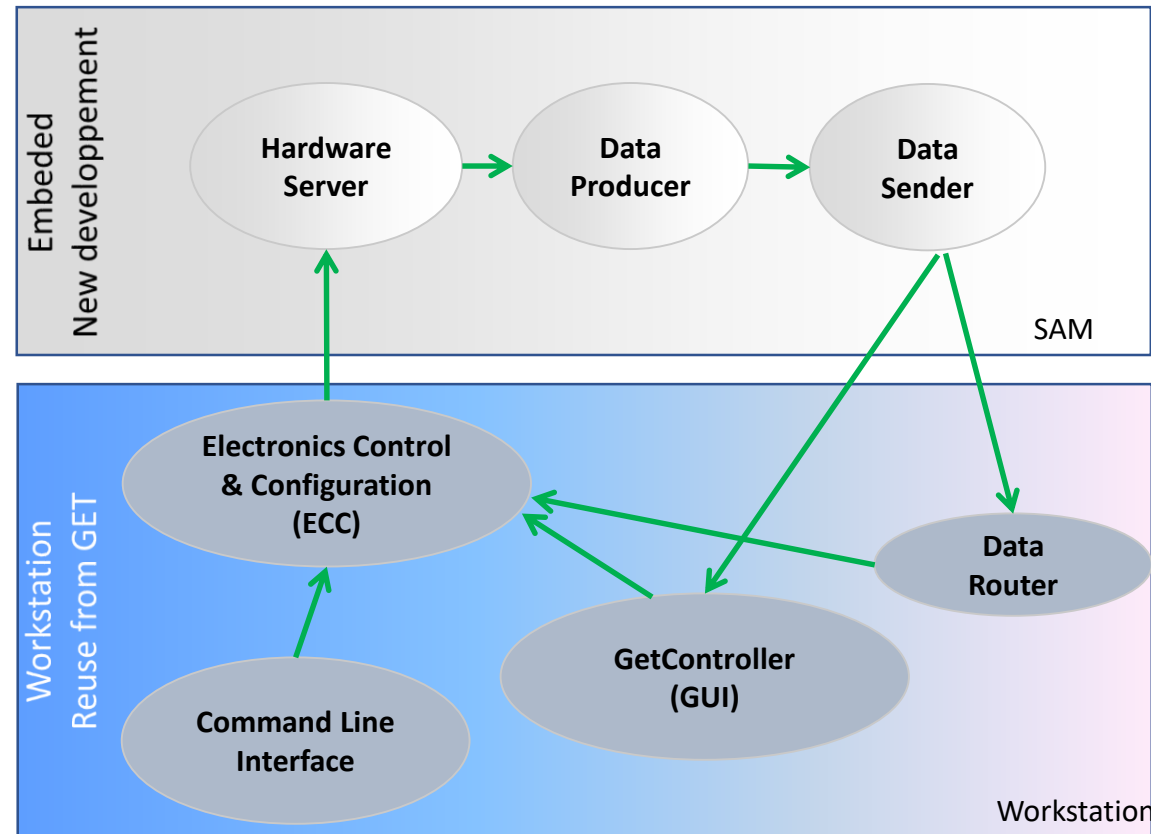
SAM : Dead Time

Dead Time = DMA width



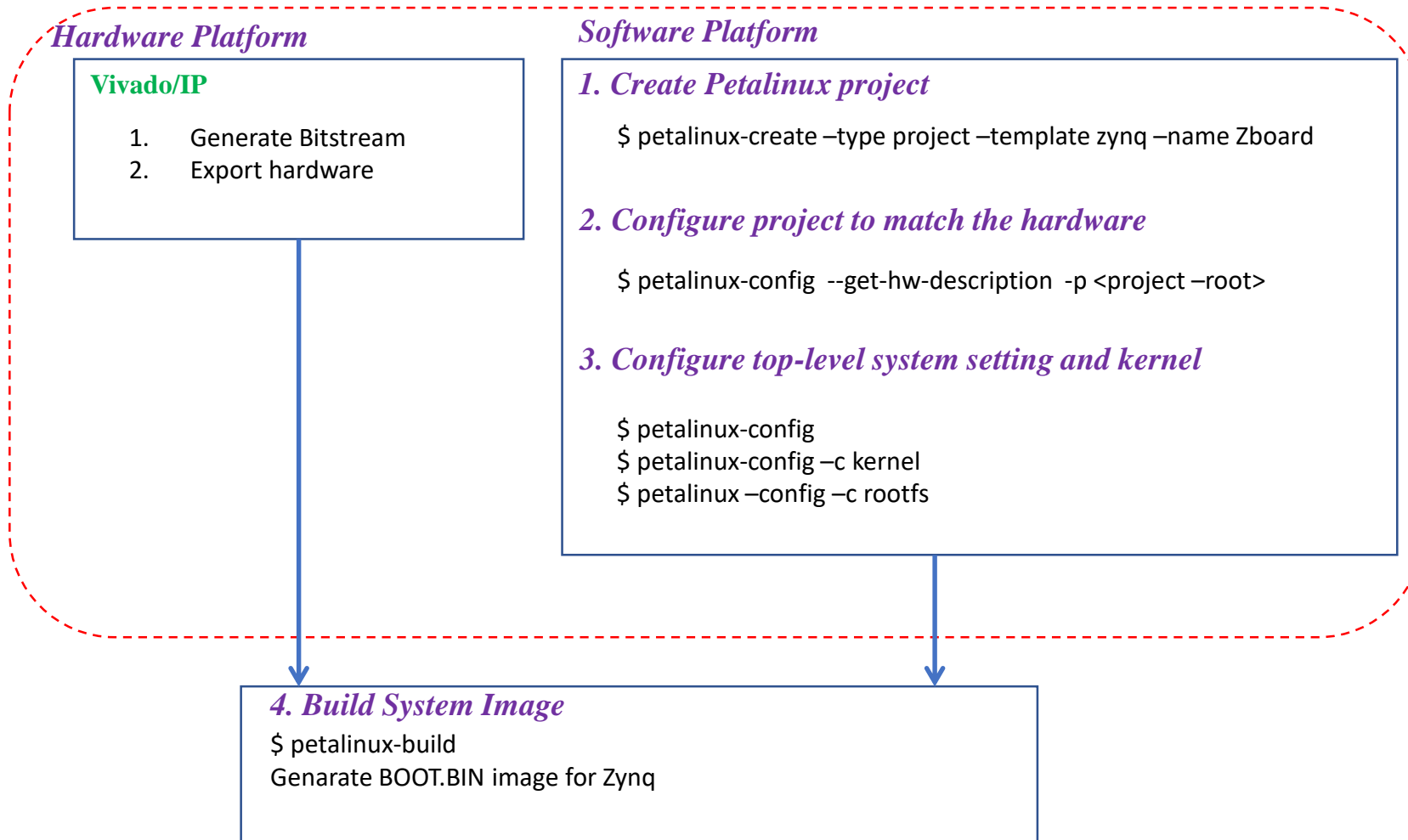
SAM : Software

▷ Overall Architecture



SAM : Software

▷ Embedded Petalinux kernel installation



SAM : Software

▶ Hardware Server : *getHwServer*

GetBench Package :

1. ***ControllerDeviceZboard.hpp*** : to define a read and write process for registers in DAC, TCM, ADC and ISP devices
2. ***ControllerDeviceAget.hpp*** : to define a read and write process for registers on AGET device
3. ***ControllerDeviceStd.hpp*** : to define a read and write process for registers in CTRL device

Programs Sources Compilation for the ARM Architecture



```
Terminal
Fichier Édition Affichage Rechercher Terminal Aide
root@Zboard:/mnt# ./getHwServer
Start Server ...
Creating LED manager
STARTED server on 0.0.0.0:46001
state machine built, state = IDLE
Device /READOUT : debug OFF
STARTED server on 0.0.0.0:46004
█
```

SAM : Software

► Software Client: GetController

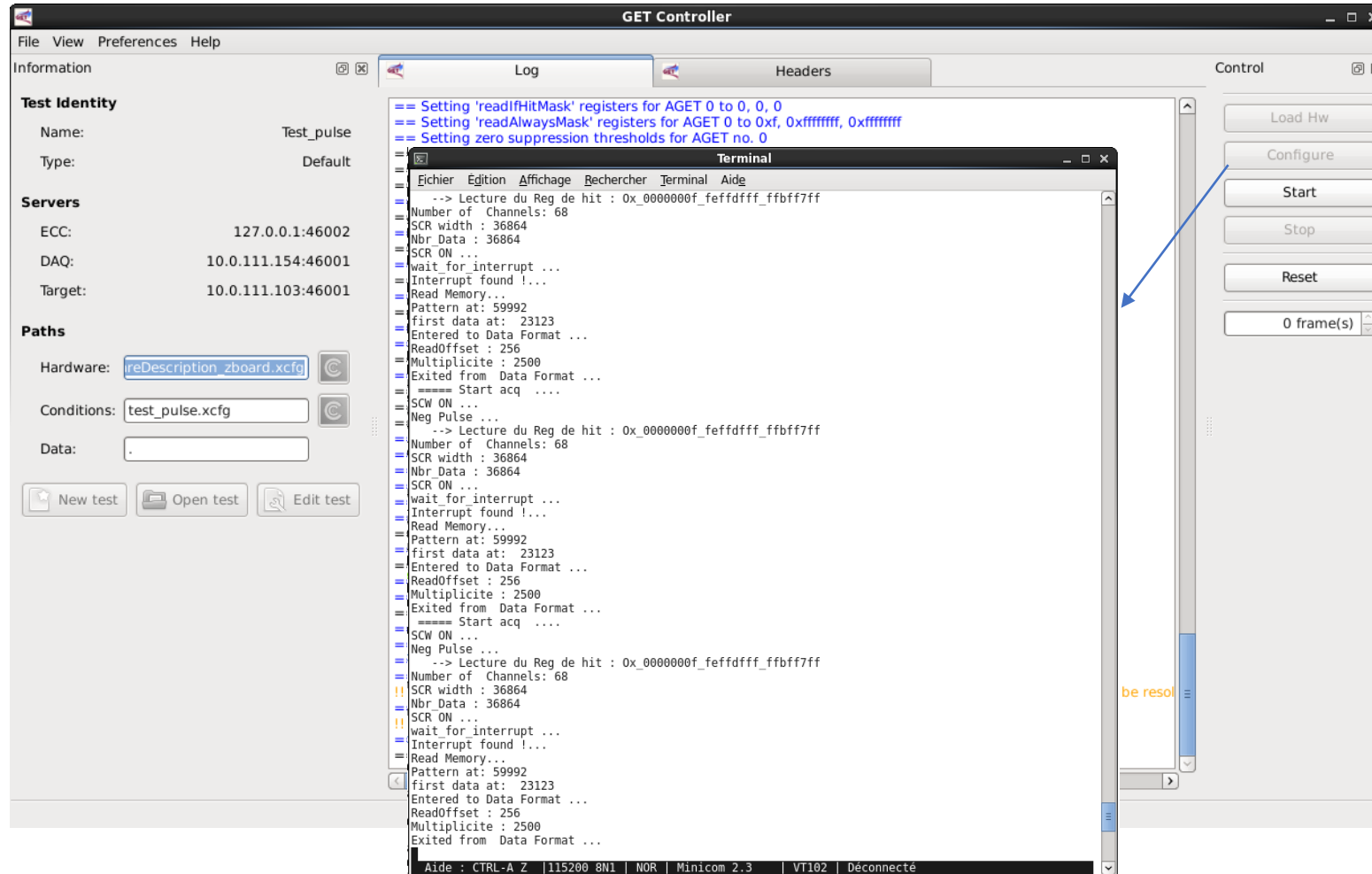
The screenshot displays the SAM software interface, which is used for configuring hardware and test conditions. It consists of three main windows:

- Main Window:** Shows test identity information (Name: Test, Type: Default), server details (ECC: 127.0.0.1:46002, DAQ: 10.0.111.154:46001, Target: 10.0.111.103:46001), and paths. The hardware path is set to `reDescription_zboard.xcfg` and the conditions path is `test_mode.xcfg`. Buttons for "New test", "Open test", and "Edit test" are visible at the bottom.
- Edit hardware configuration window:** Displays the configuration for `hardwareDescription_zboard.xcfg`. It shows a tree structure under `Setup[Hardware]` with nodes for `Device[*]`, `Device[ctrl]`, `Device[aget]`, and `Device[asad]`.
- Edit test conditions window:** Displays the configuration for `test_mode.xcfg`. It shows a tree structure under `Setup[Conditions]` with nodes for `ECC`, `Node[*]`, `Node[CoBo]`, `Instance[*]`, `Instance[0]`, `AsAd[0]`, `AsAd[1]`, `AsAd[2]`, `AsAd[3]`, `Instance[1]`, `Instance[2]`, `Instance[3]`, `Instance[4]`, `Instance[5]`, `Instance[6]`, and `Instance[7]`. The `isActive` property is set to `true`.

SAM : Software

▶ Software Client: GetController

⇒ Data Producer



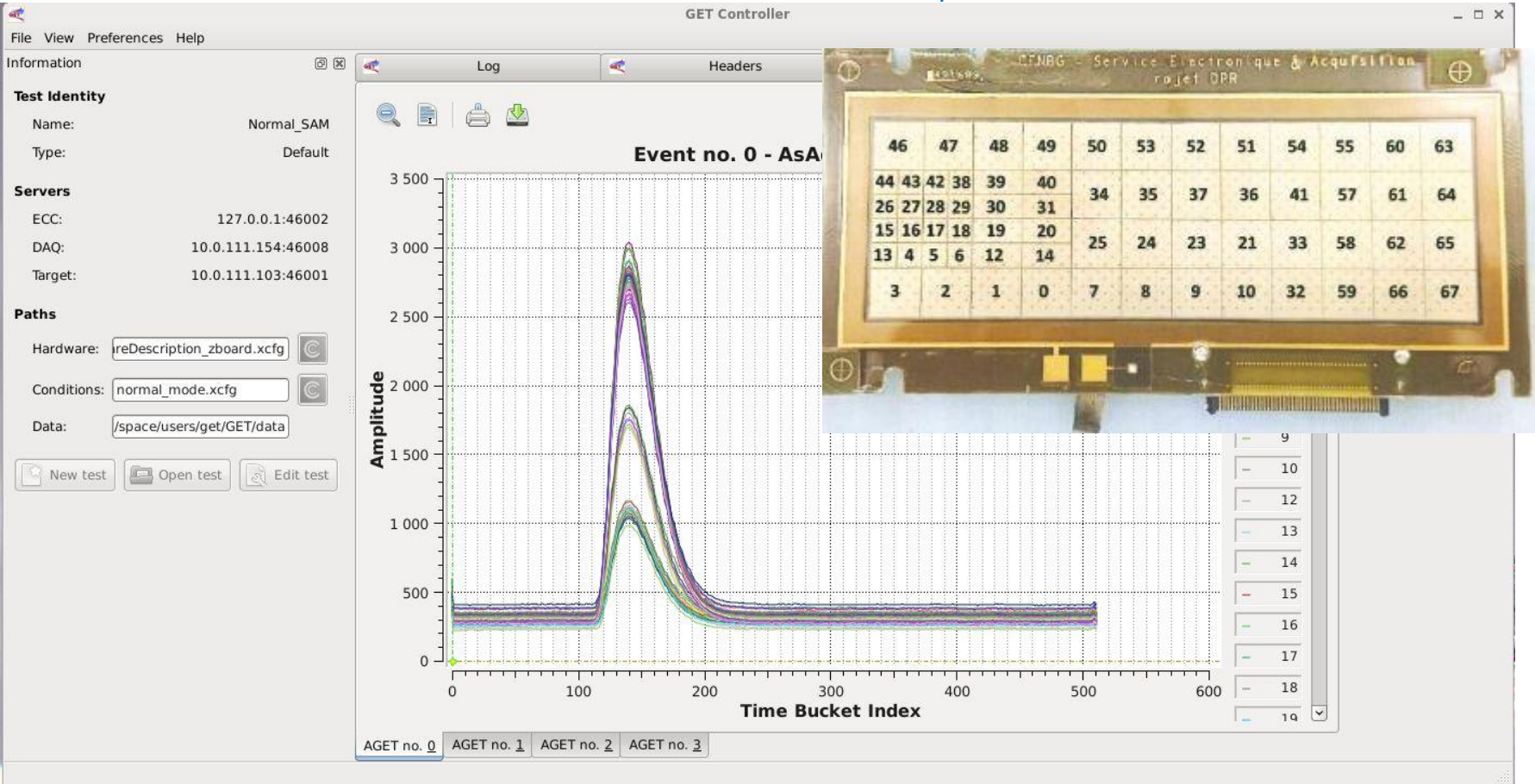
SAM : Software

▶ Client: GetController

⇒ Frames Viewer

Tests with DPR detector

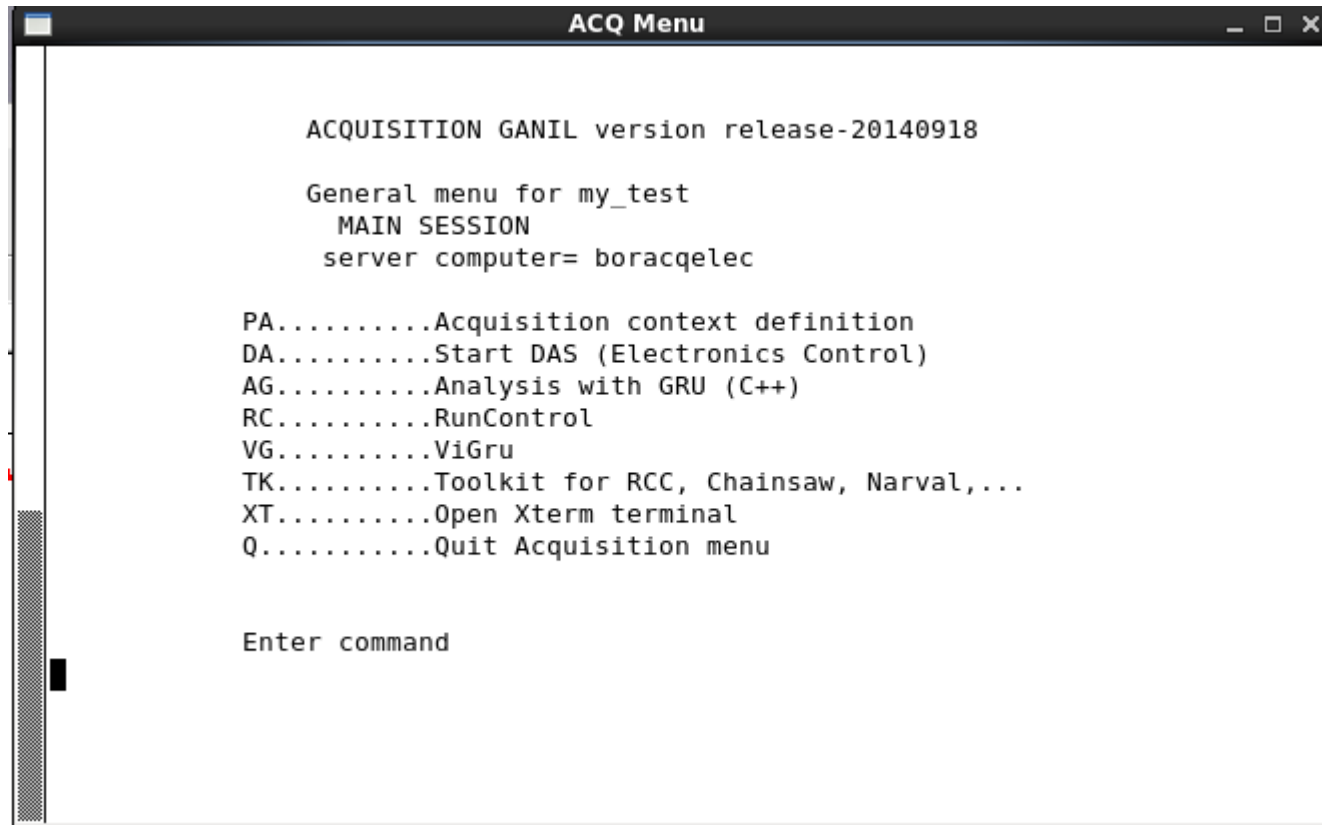
⇒ Tree pads dimensions



SAM : Software

▶ *SAGE (Systeme d'Acquisition GANIL Externe) with NARVAL*

⇒ *Inspired by what was done with the RCoBo module
and the first ACTAR / TPC experience tests*



```
ACQ Menu
-----
ACQUISITION GANIL version release-20140918

General menu for my_test
  MAIN SESSION
  server computer= boracqelec

PA.....Acquisition context definition
DA.....Start DAS (Electronics Control)
AG.....Analysis with GRU (C++)
RC.....RunControl
VG.....ViGru
TK.....Toolkit for RCC, Chainsaw, Narval,...
XT.....Open Xterm terminal
Q.....Quit Acquisition menu

Enter command
```

SAM : Software

► SAGE Acquisition: RunControl

The screenshot displays the Run Control GUI interface. The main window title is "Run Control GUI" and the file path is "/space/users/get/ganacq_manip/my_test/my_test.rcc.xml".

Monitoring mode: The interface shows a "Monitoring mode" section with a green indicator and the text "RUNNING". Below it, a "my_test" section shows a tree view with "NARVAL" selected and "ECC" as a sub-component.

System Status: A central diagram shows the system architecture. It includes components like "ECC", "Actor", "watcher", "run #8", and "Storage". Data flow is indicated by arrows and labels: "000.0 KB/s" between ECC and Actor, "005.2 MB/s" between Actor and watcher, and "005.2 MB/s" between watcher and run #8. All components are marked as "RUNNING".

Configuration Windows: Several configuration windows are open:

- Modify ECC:** Name: ECC, HostName: 10.0.111.154, Port: 8083.
- Modify CoBo[Crate00_Slot00]:** Name: CoBo[Crate00_Slot00], HostName: 10.0.111.154, Log level: info, File: gnarval_mfm_catcher, Watcher: , Watcher port number: 10201.
- Modify watcher:** Name: watcher, HostName: 10.0.111.154, Log level: warn, File: gnarval_mfm_watcher, Watcher: , Watcher port number: 10205.

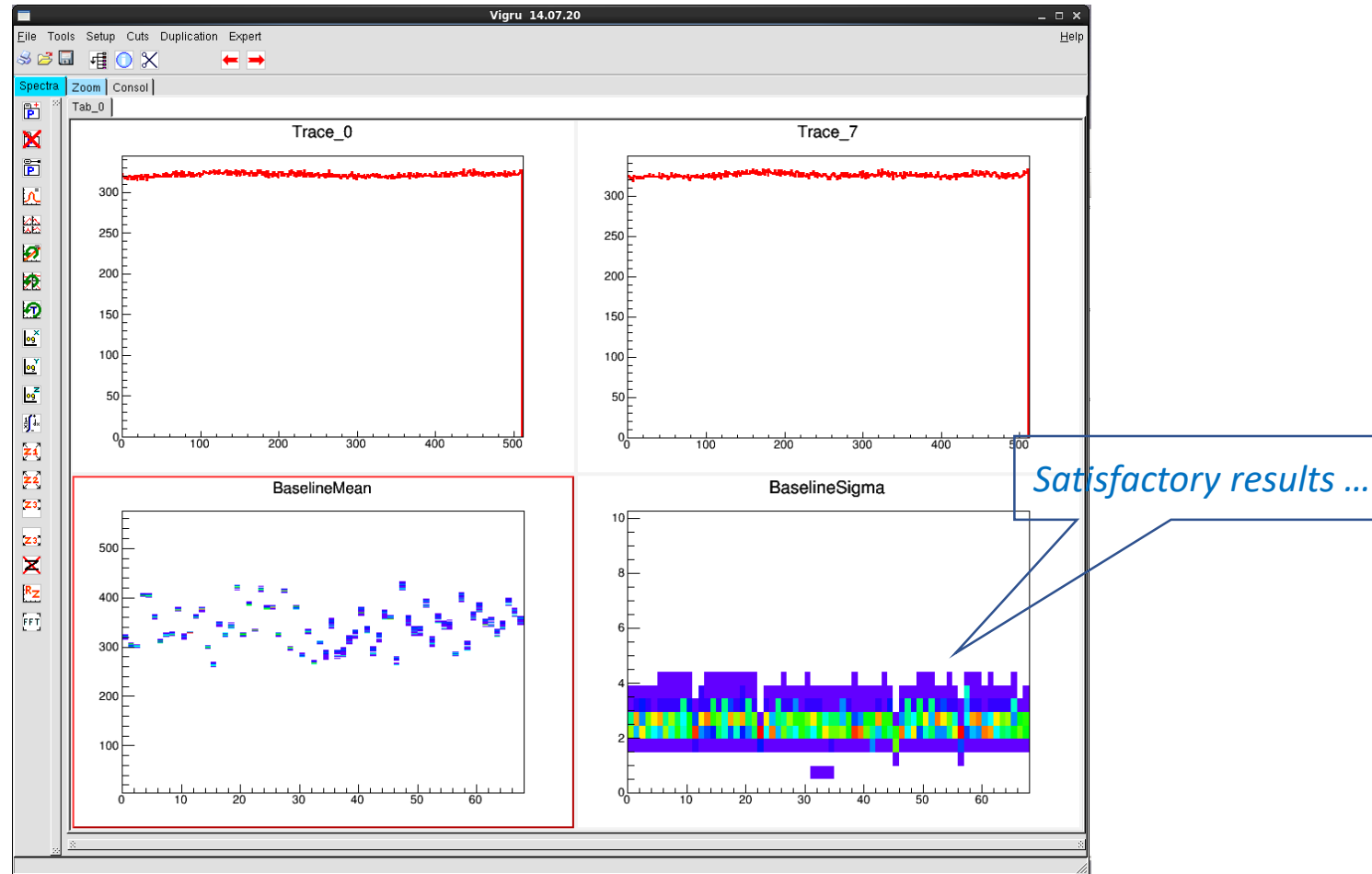
Log Window: The bottom of the interface shows a log window with the following entries:

- 01/03/2017 15:10:54 INFO log from chef_orchestre : NARVAL CoBo[Crate00_Slot00] inscrit
- 01/03/2017 15:10:54 INFO log from NARVAL CoBo[Crate00_Slot00] remote logging on

SAM : Software

▶ *Ganil Acquisition: GRU & ViGRU*

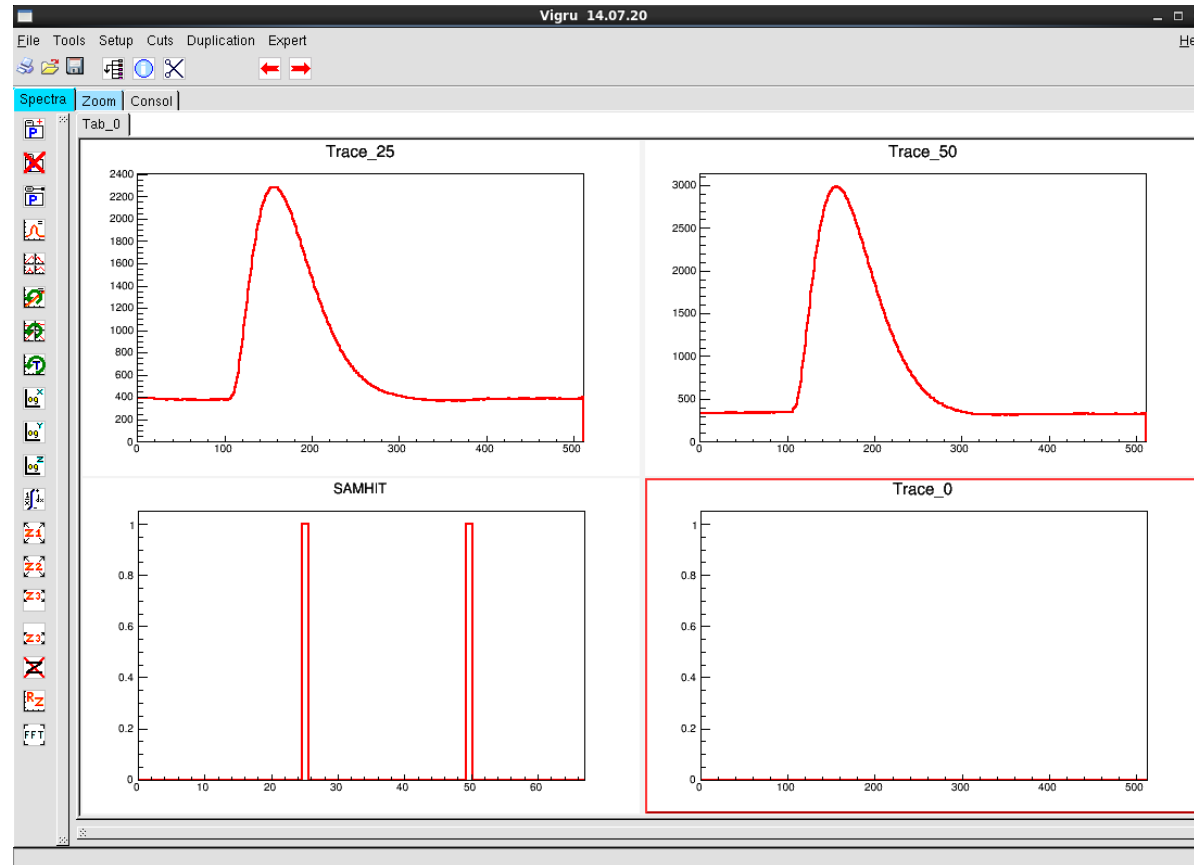
⇒ *Full Readout Mode : AGET Baseline*



SAM : Software

▶ *Ganil Acquisition: GRU & ViGRU*

⇒ *Partial Readout Mode : Hit register*



⇒ *Possibility to see only each channel .*

SAM : Perspectives

■ Conclusions:

- SAM was successfully deployed during the DPR experiment in LP2IB
- SAM is an acquisition system for the RenoCATS experiment at GANIL
- SAM system is fully functional
- Can be shared with other collaborations ...

■ Perspectives :

- Implementation of time-stamp
 - with CENTRUM module for RenoCATS experiment