DAQ architecture proposal for RD50 HVCMOS developments

Ricardo Marco Hernández
Salvador Martí i García

Introduction

- From our previous experience at IFIC in radiation sensors and developing data acquisition systems (DAQ) for radiation sensors like ALIBAVA, we learned:
    - A considerable effort is required to develop a custom DAQ for a specific purpose.
    - Once the DAQ has been developed is very difficult to use it for other purposes.
    - Commercial modular DAQ systems are expensive and sometimes not so flexible as required.
- Recently, we started to think about a DAQ system which could:
    - Be used for different purposes (radiation detectors, medical physics, accelerators, etc.) without redesigning all the hardware again and again.
    - Be scaled up easily and have new functionalities using the same basic hardware.
    - Be as open as possible (in terms of hardware, firmware and software) to facilitate future developments.
- We found a preliminary DAQ architecture which could fulfill these goals.
- We think that this architecture would fit the requirements of a DAQ for the HVCMOS developments.

DAQ requirements

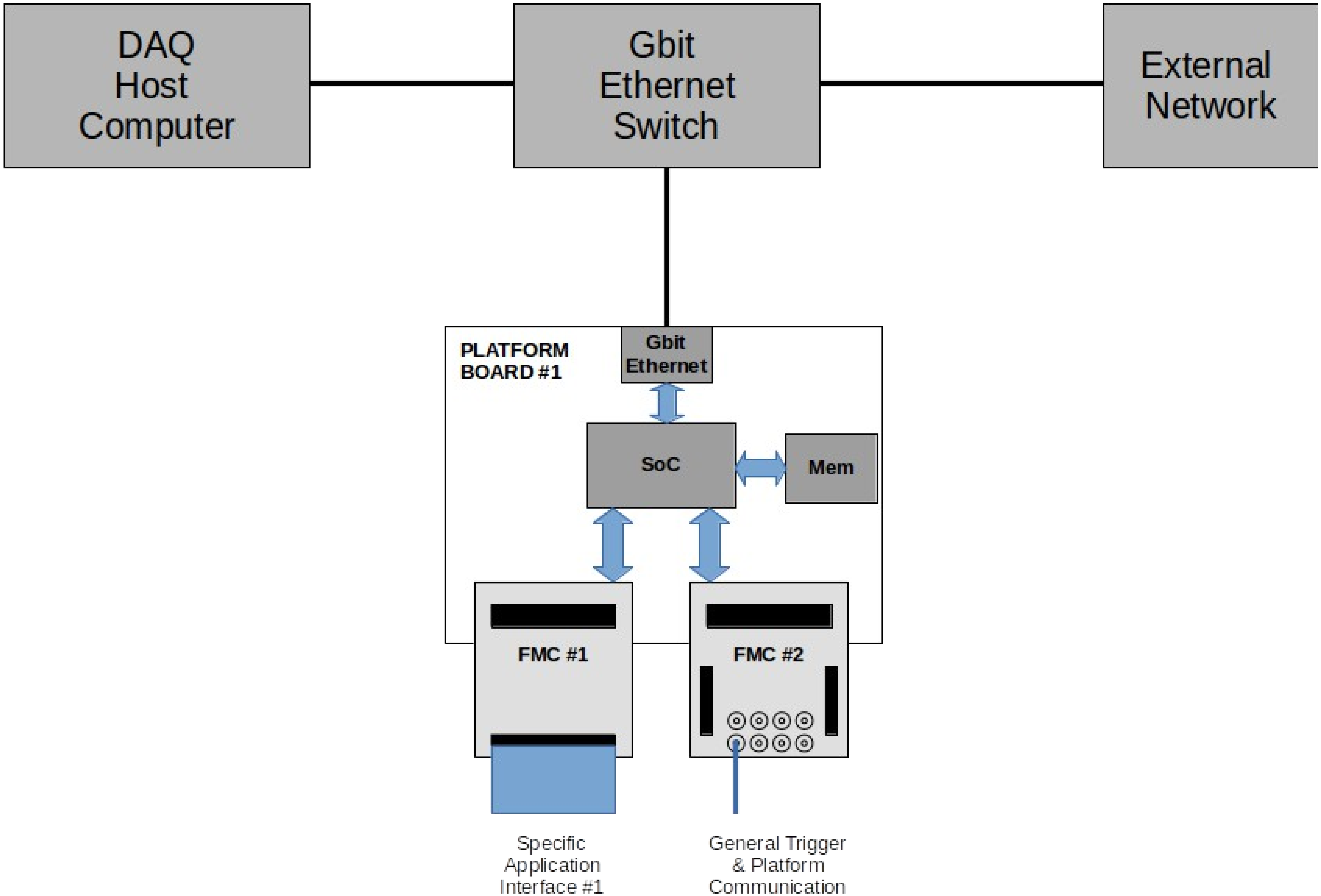
- To read out and to control different HVCMOS structures.
- To be able to process external trigger input signals (e.g. signal derived from a radioactive source).
- To be able to generate trigger output signal (e.g. for driving a laser system).
- It must work both in standalone mode (e.g. a basic laboratory test bench) and in a test beam environment (e.g. several read out planes are required with synchronization).
- Easy to operate with user friendly DAQ software.
- In order to have flexibility for current and future applications, it must be easily scalable, configurable and modular.
- …

First approach

- Programmable logic is needed for this application: FPGA.
- Combining FPGA resources with processor(s) to have more flexibility: System-on-chip (SoC).
- Modularity and standardization: FMC connectors to connect function specific FMC modules to a common platform board.
- Data storage capability: on-board volatile and non-volatile memory required.
- Fast system control and bulk data transfer to host computer: Gigabit Ethernet communication.
- Use a development board to shorten development time.

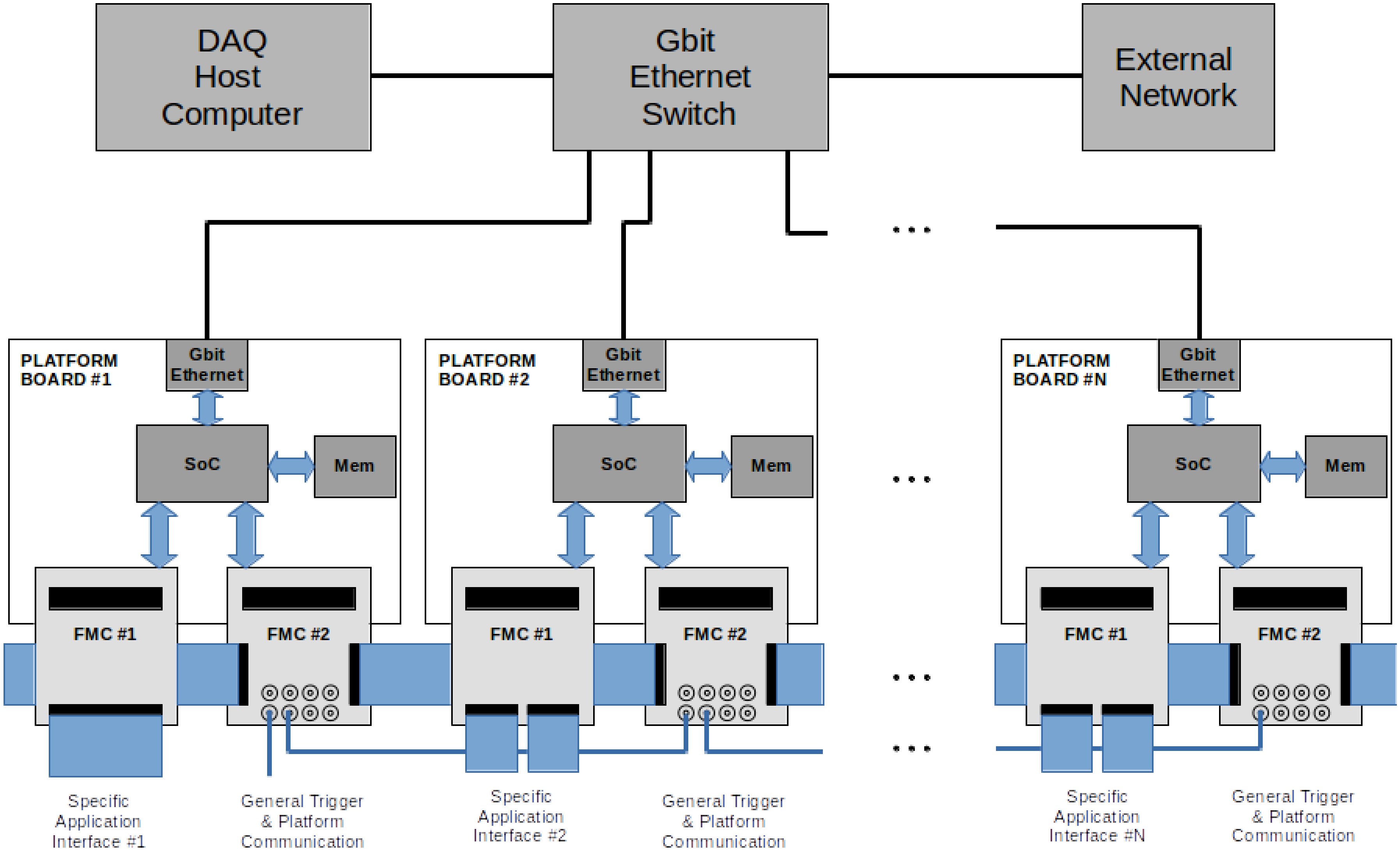
Standalone DAQ

- FMC board 1 is a custom board for a specific application (e.g. read-out/control HVCMOS structure(s)).
- FMC board 2 is a custom board used for trigger input/output processing.



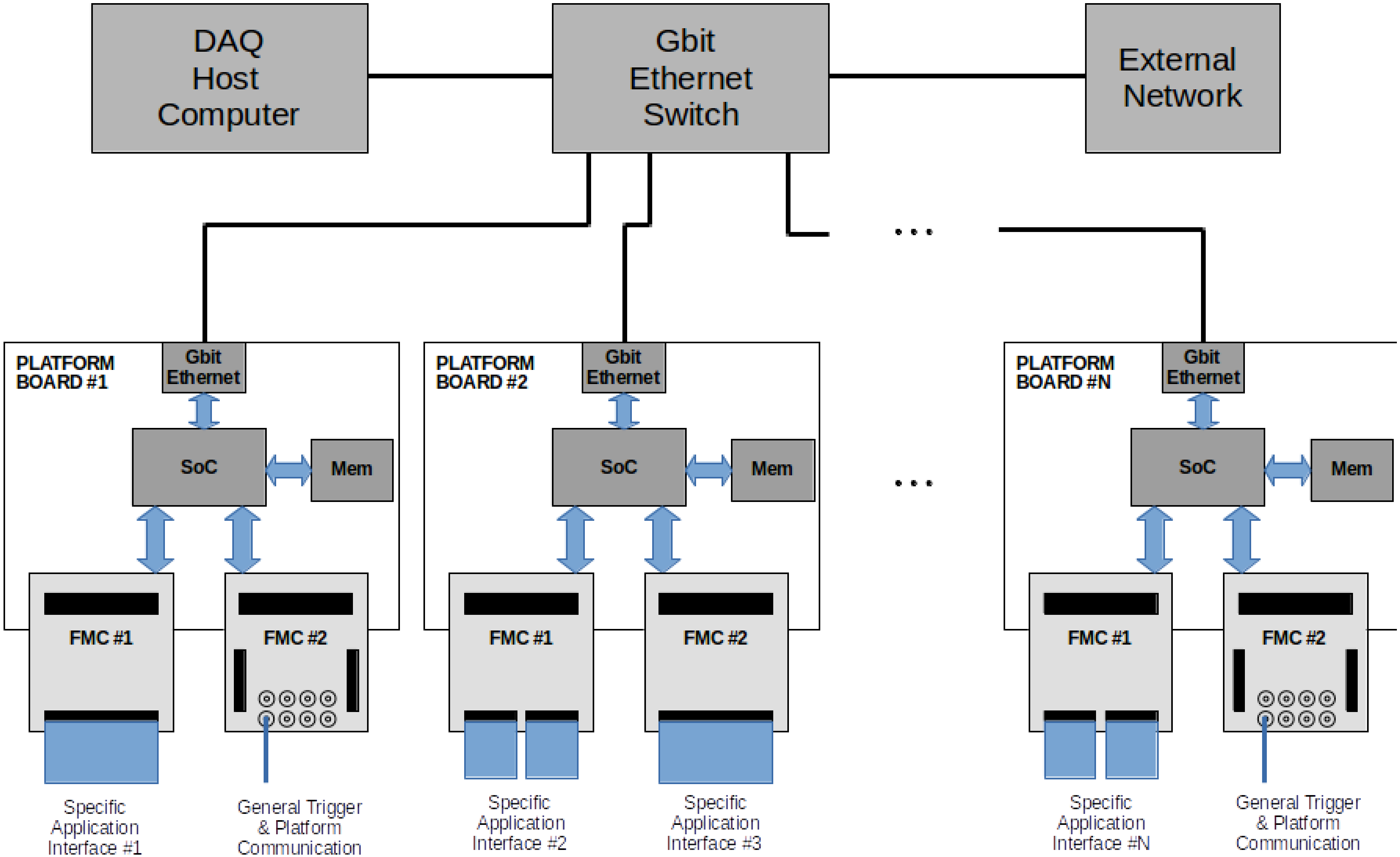
DAQ with several platform boards synchronized

- FMC board 1 is a custom board for a specific application (e.g. read-out/control HVCMOS structure(s)).
- FMC board 2 is a custom board used for trigger input/output processing and platform communication & synchronization (I2C and LVDS signals).



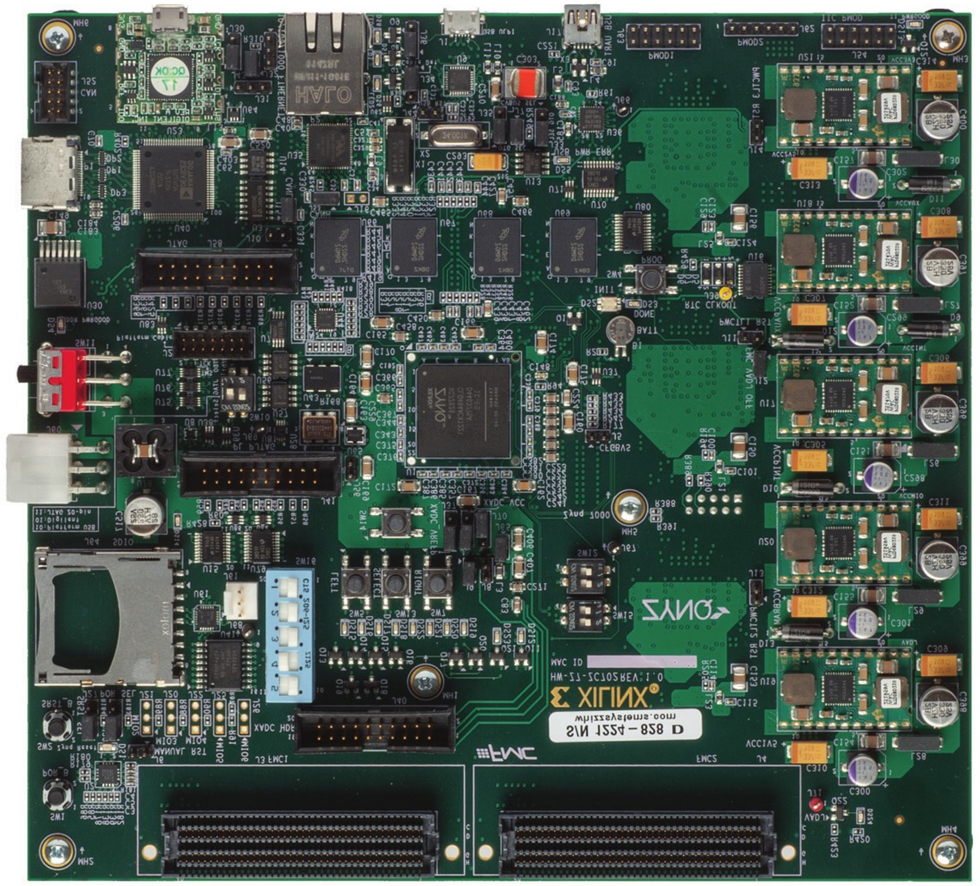
DAQ with several platform boards (not synchronized)

- FMC board 1 is a custom board for a specific application (e.g. read-out/control HVCMOS structure(s)).
- FMC board 2 is a custom board for trigger input/output processing or a custom board for a specific application (e.g. read-out/control HVCMOS structure(s)).



Platform board candidate

- ZC702 development board from Xilinx.
- SoC is Zynq-7000 XC7Z2020 with a dual-core ARM Cortex-A9 and a mid-range Artix 7 FPGA (85k programmable logic cells).
- Two FMC connectors (160-pin low pin count) with up two 68 single-ended (34 differential) signals, 2 differential clocks, I2C signals and power/gnd signals (12V, 3.3V and 2.5V).
- Tri-mode Ethernet PHY (10/100/1000 Mbps).
- 1 GB DDR3 SDRAM memory.
- 128 Mb SPI flash memory.
- SD card interface (8GB card included).
- 12 V/5 A power input.
- Other features (USB, HDMI, XADC, etc).
- All the board documentation (schematics, PCB design files, etc.) available from Xilinx.
- Board distributed by Xilinx. Easily available at Farnell (896€).



ZC702 development board

Tasks to develop

- Hardware:
  - One or several custom FMC cards for HVCMOS read out and control.
  - Custom FMC card for trigger processing and platform communication.
  - Optionally, a custom platform board with the trigger processing and platform communication implemented in the board two have both FMC connector free.
- SoC firmware:
  - Specific firmware blocks implemented in the SoC FPGA for interfacing the FMC cards designed.
- SoC software:
  - SoC dual-core ARM A9 processor can run a light Linux distribution (PetaLinux) where custom pieces of software (C/C++ programs as well as python or bash scripts) can be run to interface the firmware blocks or peripherals (Ethernet MAC, memory interfaces, I2C, SPI, GPIO, etc.).
  - Alternatively, a standalone C/C++ can be run by the processor.
- Host Computer software:
  - Low level tasks (communication with different platforms, data processing, etc.) with python scripts or C/C++ programs.
  - Graphical user interface for monitoring the data and controlling the DAQ.
  - Post processing scripts for the data acquired (Python or ROOT).

DAQ architecture proposal for RD50 HVCMOS developments

Ricardo Marco Hernández
Salvador Martí i García