Analog Data Acquisition and Processing FPGA-based Solutions Integrated in areaDetector using FlexRIO Technology

Ricardo Herrero, *Student Member IEEE*, Antonio Carpeño, Sergio Esquembri, *Student Member IEEE*,
Mariano Ruiz, *Member, IEEE and* Eduardo Barrera *Member, IEEE*

***Abstract*— Analog data acquisition used in diagnostics and control of large physics experiments require high sampling rates and real-time functionalities. Field Programmable Gate Array (FPGA) devices allow efficient implementation of such solutions. Currently, large scientific facilities are using middleware platforms to simplify systems integration. EPICS (Experimental Physics and Industrial Control System) is one of the most extended middleware for this purpose. Heterogeneous hardware integration in these middleware is a complex task, and different approaches attempt to standardize. One of these approaches is areaDetector. An open source module for EPICS that is mainly used for image acquisition. areaDetector simplifies integration of heterogeneous image systems, has also been used with some analog signals like quadEM. This paper presents the integration of an acquisition and processing solution in a PXIe platform using FlexRIO technology via a hardware model that solves acquisition and processing in a FPGA and a software model implemented in C++ with the IRIO library (open source solution for RIO devices in the Linux environment) to get to EPICS through areaDetector with analog data.**

***Index Terms* — areaDetector, Computerized Instrumentation, EPICS, FPGA, Measurement.**

I. Introduction

Data acquisition systems in large physics experiments require real time processing and high sampling rates [1]. Long-term operation or steady state implies continuous monitoring and real-time response. Event detection and real-time control usually suffer unacceptable delays when using conventional data acquisition and analysis systems. These large facilities have high heterogeneity of measurement devices.

Field Programmable Gate Arrays (FPGAs) have been used for embedded instrumentation systems, promoting intelligent data acquisition architectures (IDAQs), DAQs with a FPGA between the I/O elements and the communication bus. This architecture allows the user to customize control and processing.

II. Large physics experiments control and monitoring

Analog signal acquisition is very common in large physics experiments. For example, ITER will use discharges that lasts more than 1000s and require high sampling rates up to 125 MS/s. Steady operation also implies continuous monitoring of the plasma parameters. IFMIF requires real-time measurements to control neutron flux stability and to provide early warnings. The ESS, European Spallation Source will present 22 large measurement instruments and will use 8 times larger pulses than similar experiments. These experiments demand a framework that homogenizes instrumentation and distributes control and monitoring. These experiments will use EPICS as control and monitoring framework [2].

A. EPICS

EPICS is a control and measurement system that is placed as middleware. EPICS connects heterogeneous devices exchanging data among large facilities. The EPICS IOC, *Input Output Controller*, defines equipment connected to one or more I/O devices and operates as *Channel Access Server (CA Server)*. Each variable loaded in the IOC database is a *Process Variable* or *PV. Data is supplied by CA Servers* to *Channels Access Client,* achieving a distributed processing environment. EPICS also provides a common toolkit for developing new applications for heterogeneous devices.

B. areaDetector

EPICS provides a set of tools that allows distributed control and monitoring. However, EPICS does not provide a close set of PVs to devices. EPICS also does not provide an unique access mechanism to hardware. These functions must be programmed. Some software initiatives provide platforms for integration. Nominal Device Support [3] and

All authors are with the Instrumentation and Applied Acoustic Research Group, Universidad Politécnica de Madrid, Campus Sur UPM, Madrid 28031 (telephone: +34-913364696, e-mail: ricardo.herrero@i2a2.upm.es)

978-1-5090-2014-0/16/$31.00 ©2016 European Union

areaDetector [2] are some of them. The first is already used in projects as ITER. areaDetector is currently used with 2-D detectors. areaDetector provides an interface with EPICS via the asynNDArrayDriver module –based on asyndriver [4]- that eases the development of hardware drivers and reduces coding effort, see Fig 1. (4). We can add the required elements (5) for managing each particular hardware (6) inside areaDetector model.

The areaDetector model offers many advantages, e.g., high performance, as it uses a pool to transfer NDArrays. Pre-processing capabilities for acquired data, previous to EPICS reception; in this manner, we are not required to send raw data to EPICS. areaDetector also provides an architecture based on software reuse and mechanisms to perform analysis and distributed and parallel processing in real time [4].

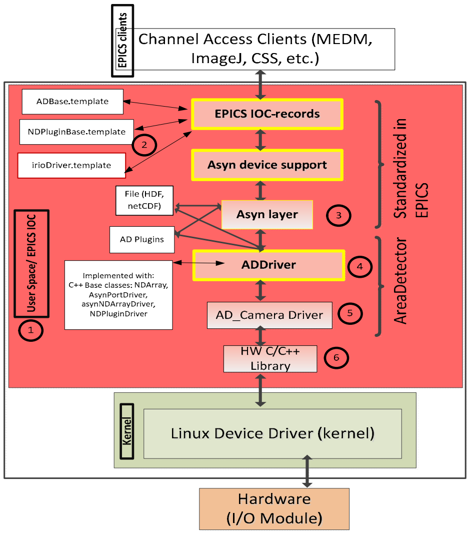

Fig 1. areaDetector software architecture. areaDetector standardizes the PVs and their connection with an asyn layer.

C. *Data acquisition devices based on FPGA*

An FPGA in a data acquisition system allows it to implement a huge variety of hardware implementations [4], pre-processing and other IDAQ solutions. Unlike general purpose processors, FPGAs admit deterministic parallel processing. Each independent processing function is assigned to a section of the chip and can be executed autonomously.

Nonetheless, all this flexibility in the possible implementations can result in a very complex, so the design can be very difficult. The proposal of a function-oriented implementation model of the FPGA simplifies this interface. The chosen solution in this work is exemplified using data acquisition devices from the FlexRIO family.

III. Waveform Data Acquisition with AreaDetector and FPGAs

We will describe the areaDetector driver implementation, IRIOAD_1D. We have integrated the driver with the kernel module that accesses FPGA resources in a Linux Operating System, named NI-RIO Linux Device Driver. These libraries replace (5) and (6) in Fig 1. The software driver for areaDetector replaces (4) in Fig 1. This module is both the areaDetector driver and the FPGA interface.

A. *IDAQ implementation model for the FPGA*

This architecture includes setup and control of the sampling acquisition thereby simplifying new processing hardware integration for common algorithms. The proposed architecture can be used with different adapter modules and tuned to different needs. Some elements must be set up before compiling the design. The remaining elements can be modified when the FPGA is in the execution state.

Hardware components of the FPGA model are divided in blocks. *Acquisition Control* block is in charge of sampling signals from the input analog channels. The hardware responsible for signal processing depends on each particular application in the *Signal Processing Algorithm* block.

Acquired channels can send raw data or undergo some pre-processing before they are sent to the upper layers. Results are sent through DMA channels. *DMA Channel Frame Generation* block builds 64-bit word frames to maximize throughput.

The architecture can generate analog and digital signals. There is a block named *Waveform Generation and Control* which generates waveforms and *Pattern Generation* block generates digital patterns. In addition, there are auxiliary registers with customizable functions by the user.

Once data acquisition is started, all the information that is acquired will be sent to EPICS using DMA. DMA channel signals are controlled by the *DMA Channel Transfer Control* block.

B. *Data acquisition management in IRIOAD_1D.*

IRIO offers an API for the development of C/C++ applications. It has been designed for image and data acquisition devices. IRIOAD_1D retrieves the acquired data from the FPGA through DMA channels, achieving high transfer rates. The IRIOAD_1D driver arranges received data, splitting it up in the acquired channels

When an operator acts on one of the available PVs, through a CA client, some read and write operations occur in the driver and in the FPGA via the interfaces in IRIOAD_1D/asynNDArrayDriver. As previously stated, data are in NDArray format; to view its representation as waveform in EPICS we must use the NDStdArray plugin. The same NDArray can also be consulted by different plugins. Once all of the plugins finish reading the NDArray, it will be released and returned to the pool.

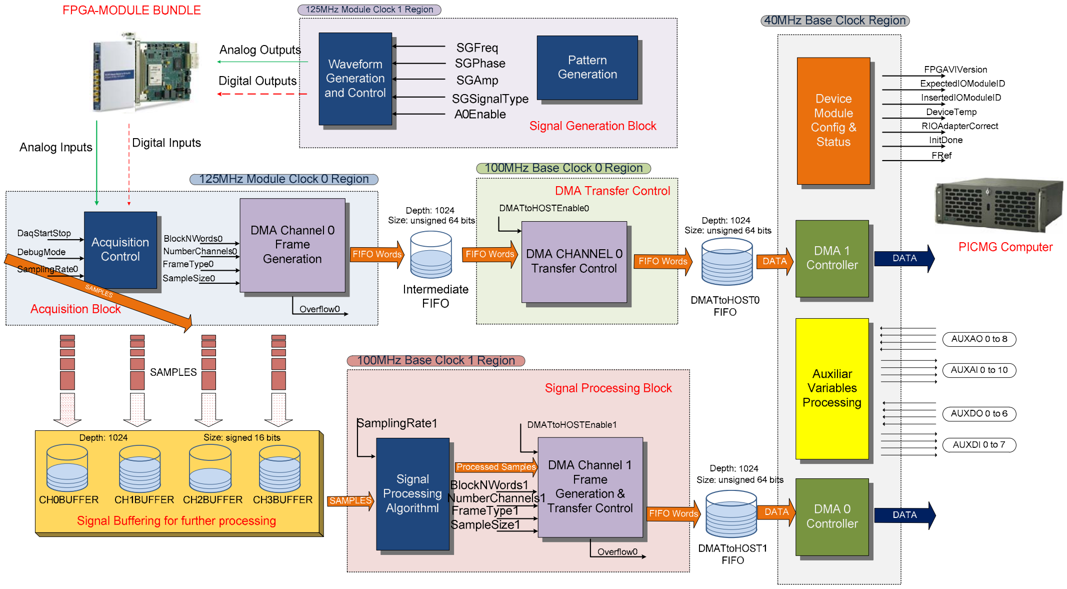

Fig 2. FPGA architecture model solving data acquisition, processing and interface with host computer.

C. *1D areaDetector driver implementation for high sampling rates*

IRIOAD_1D inherits from the main class of AD, ADDriver. This class includes a set of attributes and base methods that are common to a variety of devices. Some attributes are no longer required for analog acquisition. In fact, asynNDArrayDriver implements the actual connection to the asyndriver layer and provides support for PVs and NDArrays. IRIOAD_1D must also contain all the attributes and functions to fully implement III.A and III.B functionalities.

It is possible to implement a class parallel to ADDriver that inherits from asynNDArrayDriver. This approach will reflect the analog acquisition devices functionality. We can then avoid class-specific design for images, which is inefficient. A new class is derived from asynNDArrayDriver that also encapsulates all functions common to IDAQs.

IV. Test and Results

To test the solution, we used an industrial computer connected to a PXIe chassis with a FlexRIO PXIe7966R card, allowing four-channel acquisition up to 125 MS/s. The implementations in the FPGAs were performed in accordance with III.A model. When data are sent to EPICS, we have obtained a continuous rate of 5 MS/s. We must consider that EPICS is not designed to admit the FPGA throughput. Designs use the following FPGA resources: 43.1% slices (LUT and Registers), 55% of DSP elements, and 26.2% block RAM.

V. Conclusions

This work establishes basis to integrate FPGA IDAQ devices in areaDetector and EPICS. Moreover, this work also proposes a generic description model for acquisition and processing using FPGAs. It can be implemented with any FPGA. This solution focuses the effort exclusively in hardware integration. The model in III.A, synthesized with the aid of LabVIEW/FPGA and XILINX tools, is just one way to reduce hardware development. This solution can be applied to other targets using the same approach as presented in [6], where OpenCL is also used.

References

[1] Abbasi, R.; Ackermann, M.; Adams, J. M. Ahlers; J. Ahrens, K Andeen, et al., "The IceCube data acquisition system: Signal capture, digitization, and timestamping," Nuclear Instruments & Methods in Physics Research section a-accelerators spectrometers detectors and associated equipment Volume: 601 Issue: 3 Pages: 294-316 Published: APR 1 2009.

[2] M.Rivers, areaDetector:EPICS software for areadetectors,2010.

[3] S. Simrock. “Standardization of EPICS Device Support for Data Acquisition”. Proceedings of the Sixth Low-Level Radio Frequency Workshop 2013.

[4] T. Madden, P. Jemian, S. Narayanan, A.R. Sandy, M. Sikorski, M. Sprung, et al, “Real-time compression of streaming X-ray photon correlation spectroscopy area-detector data”. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, Volume 649, Issue 1, 1 September 2011, Pages 237-239, ISSN 0168-9002

[5] F. Carena, W. Carena, S. Chapeland, et al, “The ALICE data acquisition system”, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, Volume 741, 21 March 2014, Pages 130-162, ISSN 0168-9002

[6] Srikanth Sridharan. “Evaluation of 'OpenCL for FPGA' for Data Acquisition and Acceleration in High Energy Physics” Journal of Physics: Conference Series, 664(9):092023+, December 2015