A novel real-time control system for next generation gravitational waves detectors

P. Prosperi[1], A. Gennai[1], D. Passuello[1], F. Frasconi[1] , F. Spada[1], F. Pilo[1] , V. Boschi[1], M. Piendibene[1,2] , M. Bitossi[1,3]
[1]INFN sezione di Pisa, [2]Università di Pisa, [3]EGO

Abstract

This poster reports on studies and developments, conducted by the INFN Pisa group, to define and develop a new Real-time Control System (RCS) for next-generation large-scale interferometers, with a focus on the initial phase of the Einstein Telescope (ET) detector for gravitational waves. Two different hardware approaches, "**Katane**" and "**Zancle**," are explored.

"**Katane**" employs DSP processor boards, inspired by the Advanced Virgo (AdV) Super-Attenuator (SA) control system, and is built upon the MicroTCA.4 standard for flexibility and fast data exchange. Its modules include Front-End data converters, FPGA-based pre-processing boards, and DSP main processing cores. Development efforts encompass firmware and software for control and monitoring.

"**Zancle**" investigates GPU-based processing. The current focus is on testing the feasibility of using direct memory access (DMA) techniques to reduce latency between the GPU and data converters. This effort aims to leverage significant computational power and incorporate machine learning algorithms.

"Katane"

Fig.1 – MTCA (custom) based crate used in AdV+ control electronics

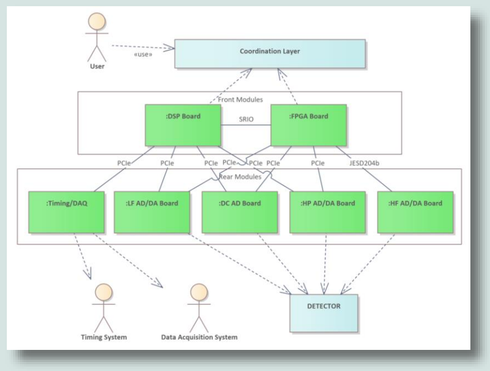

Fig.2 – MTCA4.0 HW architecture for new Katane system

A Real-time Control System (RCS) coordinates complex Multi-Input-Multi-Output feedback loops, managing data collection from sensors, signals processing and actuators control within well defined time constraints.

"***Katane***" is based on MicroTCA.4 standard, ensuring flexibility, easy maintenance, and fast data exchange among its various custom modules. The main processing boards are based on powerful DSP and FPGA modules ensuring high computing power and efficiency. Standard protocols, such as Serial-RapidIO and PCIe, allow fast and reliable data exchange in the system.

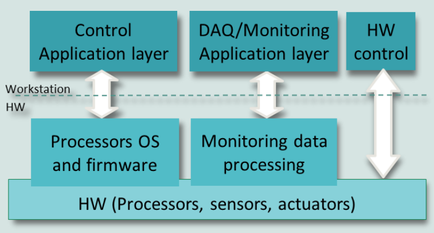

Fig.3 – Overview of SW architecture for Katane RCS

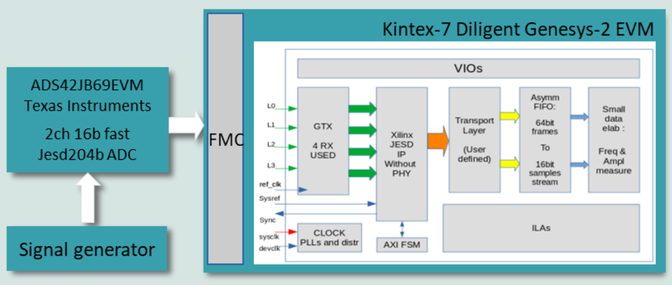

Fig.4 – Jesd204b protocol links test-bench with commercial EVMs

Hardware design is ongoing, with several MicroTCA.4 boards now at an advanced stage, and the first prototypes are about to be produced.

Alongside significant hardware module development, substantial design and testing efforts have been dedicated to creating firmware and software for system control and monitoring.

For instance, JESD204B protocol communication between HF rear modules and FPGA-based front modules has been implemented and tested on commercial EVMs. Additionally, a beta software application for data monitoring has been developed and is being used as an auxiliary tool for R&D purposes in the Pisa lab.

"Zancle"

"***Zancle***" *project* aims to employ cost-effective PCIe based systems designed for the consumer market, although a comprehensive architectural vision for this project is still under development.

However, a modular and flexible architecture, similar to the Katane solution, is foreseen for data converter boards. In this setup, GPU modules should be used in place of the DSP boards and connected through PCIe slots with standard CPU cores and FPGA modules.

Fig.5 – Testbench

Fig.6 – Used GPU and FPGA modules

In our Pisa laboratory, we are using a testbench that includes an NVIDIA Quadro RTX 4000 GPU and a Xilinx ALVEO U50 FPGA module, both connected via PCIe slots to a motherboard with an Intel i9 processor, to investigate GPU based RCSs.

Currently, our focus for this project is on testing the feasibility of utilizing direct memory access (DMA) techniques to achieve low and deterministic latency for data exchange between GPUs and data converters. This will help us determine if and how GPU processors can be used in real-time control systems. Indeed, While the computational power of GPUs is remarkable, the interactions and data exchange between GPUs and CPUs might pose challenges for implementing RCS applications.

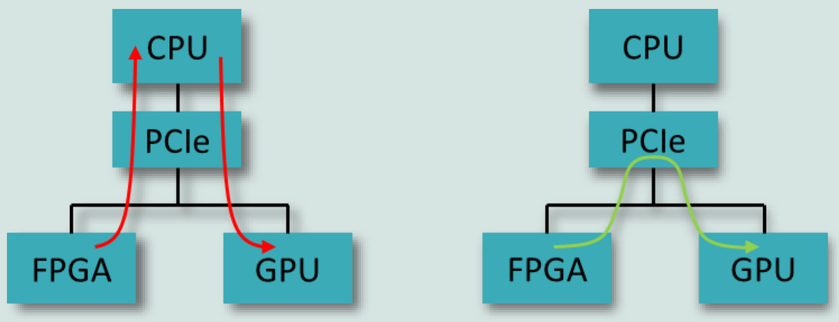

Fig.7 – Typical (red) and wanted (green) data flows FPGA-GPU

paolo.prosperi@pi.infn.it