

A novel centralized slow control and board management solution for ATCA blades based on the Zynq Ultrascale+ System-on-Chip

Luis Ardila-Perez^{1,*}, André Cascadan², Luigi Calligaris², Denis Tcherniakhovski¹, Matthias Balzer¹, Marc Weber¹, and Oliver Sander^{1,**}

¹Karlsruhe Institute of Technology

²São Paulo Research and Analysis Center

Abstract. Data acquisition systems (DAQ) for high energy physics experiments utilize complex FPGAs to handle unprecedented high data rates, especially in the first stages of the data processing chain. The complexity of developing and commissioning these systems increases as additional local processing intelligence is placed closer to the detector directly on the ATCA blades. On the other hand, sophisticated slow control is as well desired. In this contribution, we introduce a novel solution for ATCA based systems, which combines the IPMI, a Linux based slow-control software, and an FPGA for custom slow-control tasks in one single Zynq Ultrascale+ (ZUS+) System-on-Chip (SoC) module.

1 Introduction - The Phase-2 tracker electronics system

In the context of the high-luminosity upgrade of the LHC (HL-LHC), the level-1 CMS trigger system will be upgraded by adding tracking and high-granularity calorimeter information [1]. The CMS silicon tracker will be replaced with a novel design, based on double-sided detector modules arranged in a tilted geometry [2]. The new modules can select pairs of hits compatible with high transverse momentum tracks ($p_T > 2\text{-}3\text{ GeV}$), distinguishing them among a large number of background hits of little interest for use in the level-1 trigger. The selected pairs of hits, known as "stubs," are then sent to the back-end system at the full collision rate, where they are reconstructed into candidate tracks that will contribute to the global level-1 trigger decision. Furthermore, the modules are also capable of streaming the full set of hits of those triggered events to the data-acquisition system.

In the back-end electronics system, track reconstruction is done based on a time-multiplexed architecture (shown in Figure 1) requiring the use of two layers of data processing [3]. The first layer is represented by the Data, Trigger, and Control (DTC) boards, which extract and pre-process the stub data – for example, by converting the position from module-local coordinates to experiment-wide 3-dimensional coordinates – and route it to the correct card in the Track Finding layer. Routing of stubs follows a set of rules aimed at splitting the data as a function of time and geometrical origin in the detector (time multiplexing is based on event number and azimuthal sector position). The DTC cards are also responsible

*e-mail: luis.ardila@kit.edu

**e-mail: oliver.sander@kit.edu

for interpreting and repackaging the raw hit data from triggered events, sending the latter to the DAQ & TTC Hub (DTH) cards. They also provide the control and timing signals to the tracker front-end modules. The second layer, the Track Finding Processor (TFP), receives all data for a given event and reconstructs the trajectories of charged particles in the tracker.

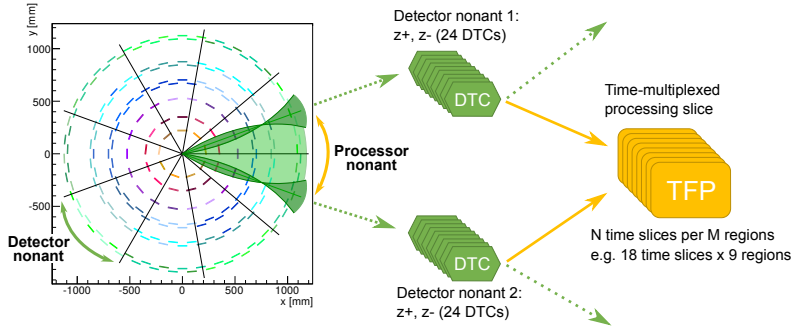


Figure 1: CMS tracker back-end system architecture using two neighboring detector nonants DTCs to time-multiplex and duplicate stub data across processing nonant boundaries for transmission to the Track Finding Processors (TFPs) [4]

1.1 Hardware Developing Platforms

CMS has adopted the Advanced Telecom Communications Architecture (ATCA) as a platform for the Phase-2 back-end electronics. Three types of ATCA node blades (the OT-DTC, IT-DTC, and the OT-TFP) will be involved in back-end processing for the Phase-2 Tracker. Most of the design and operating requirements for these blades are common. Currently, two hardware approaches are being developed as part of the R&D program within the level-1 trigger community [5].

The Serenity [6] board uses a commercial COM-Express Type-10 module with an Intel Atom CPU running CentOS Linux, assisted by a small Artix7 FPGA for low-level control (e.g., I^2C , JTAG). The COM-Express communicates to the main FPGAs via PCIe Gen2 protocol and uses the CERN-IPMC [7] for shelf management tasks running the Pigeon Point commercial software [8]. It has two processing sites where different FPGAs can be mounted using a Samtec Z-Ray interposer interconnect. Both processing sites support up to 96 optical links at 28 Gb/s, using Samtec Firefly optical modules.

The Apollo [9] board separates the blade physically into generic infrastructure and application-specific parts. The Apollo "Service Module" (SM) carries a seven series Xilinx Zynq System-on-Module (SoM) [10] providing a dual ARM-A9 processor and FPGA logic in the same package. It communicates with the main FPGAs via two lanes of AXI Chip to Chip and can either mount the CERN-IPMC or the UW-IPMC [11] by changing resistor configurations in the board. The Apollo "Command Module" (CM) could be tailored to each application. The current configuration contains two FPGAs with several hundred optical links with speeds up to 16 or 25 Gb/s, where Samtec Firefly modules are used for the optical connections as well. The CM contains a micro-controller performing temperature monitoring and low-level configuration for different devices.

2 Unified slow control architecture

Several efforts are being made to develop common hardware infrastructure based on the ATCA standard, which could support the technical requirements of the CMS level-1 track trigger. Some of the challenges include the appropriate selection of slow-control and shelf manager devices to enable full control of the blade and seamless integration with other sub-components of the CMS experiment.

Current prototypes contain small FPGA parts, micro-controllers, and SoCs to perform specific tasks that could be all integrated into a single heterogeneous device like the Zynq Ultrascale+ (ZUS+) from Xilinx as shown in Figure 2.

The ZUS+ SoC provides FPGA logic, high-performance ARM-A53 multi-core processors, and two ARM-R5 real-time capable processors. The ZUS+ is divided into four power domains, which can be enabled or disabled independently.

The ARM-R5 cores are used to implement time-critical and deterministic tasks using freeRTOS or bare-metal applications. They provide the Intelligent Platform Management Controller (IPMC) functionality and communicate via the backplane of the ATCA crate with the shelf manager to negotiate the card power-up and subsequent stable operation. The ARM-R5 are also connected to the power supplies (via PMBus), to voltage and current monitors, to clock generators and jitter cleaners (via I²C, SPI) providing the required configuration at start-up.

Once full power is enabled from the crate, a CentOS Linux based operating system starts on the ARM-A53 cores. The FPGA is used to implement some of the low-level interfaces, including IPBus or glue-logic. The SoC is the central entry point to the main FPGAs on the motherboard via IPMB and TCP/IP based network interfaces. The communication between the ZUS+ SoC and the main FPGAs uses the AXI chip-to-chip protocol via MGT pairs keeping infrastructure requirements in the main FPGAs to a minimum.

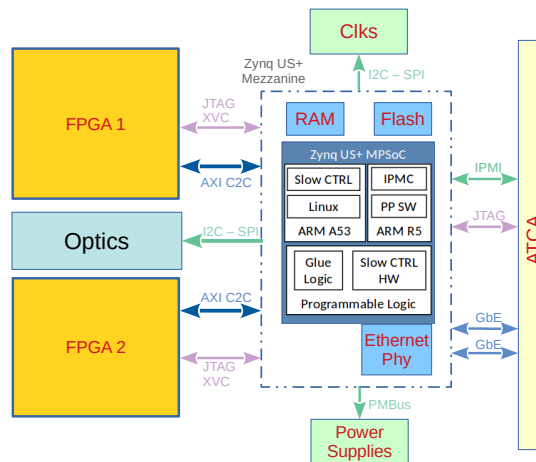


Figure 2: Unified slow control architecture

3 Setup Description - Trenz to Serenity Adapter

An adapter board (in Figure 3) was designed to map the COM-Express and CERN-IPMC footprint found on the Serenity Board to a commercially available ZUS+ module manu-

factured by Trenez Electronic [12] featuring the XCZU4EG device. The adapter uses two voltages, 3.3 V, and 1.8 V, which are used to power all components. The main 3.3 V is either generated by regulating down the "Payload" 12 V voltage, or by connecting the "3.3 Standby" power present in the CERN-IPMC SODIMM. The TPS2121 priority power multiplexer allows the adapter to receive power from either one source or switch between them seamlessly.

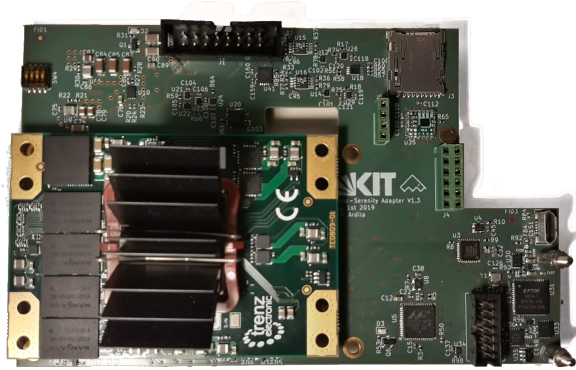


Figure 3: Trenez-Serenity adapter v1.3

Several peripherals were integrated inside the adapter providing different functionalities:

- **USB:** The USB3340 physical layer (PHY) is used to provide USB 2.0 host capabilities, it uses the low pin interface (ULPI) to connect to the MIO Pins at the ZUS+ device and a USB type A female connector present in the motherboard.
- **Ethernet:** The 88E1512 Ethernet PHY serves ethernet connectivity to the module, the RGMII interface connects to the MIO pins while the Media Dependent Interface (MDI) is connected to the COM-Express pinout which in turn has wired the Ethernet switch present in Serenity. This switch grants connectivity to the two back-plane base interfaces and ETH links to other devices on the board.
- **microSD:** A self-designed $7 \times 7 \text{ mm}$ 24 pin QFN adapter was produced containing the NVT4857UK chip. It allows the SD 3.0-SDR104 protocol with up to 104 MB/s data rate. It significantly increases the speed of reads and writes to the memory. The microSD adapter in Figure 4 was done so that the small 0.4 pitch WLCSP package could be contained in its own project where appropriate production quality measures were implemented (*e.i* x-ray imaging for a whole batch production).
- **Sata:** One processor-high-speed-transceiver is connected to an M.2 SATA SSD present on the motherboard. The Linux operating system can recognize the drive, format it, and mount it as a persistent storage unit.
- **On-board Programmer:** A micro USB port provides a JTAG and two UART interfaces powered by the USB connector and using the FTDI FT4232 chip. 5 V are regulated down to 3.3 V reducing the load from the 3.3 V standby power when the circuit is not needed.
- **AXI-C2C:** The programmable logic MGTs of the ZUS+ are connected to each of the main processing FPGAs and the auxiliary FPGA present on the motherboard, it is possible to communicate via AXI-C2C or PCIe.
- **IPMC:** Two independent I^2C buses are available at the low power domain peripherals for communication with the shelf manager using the hot-swappable controller LTC4300-1.

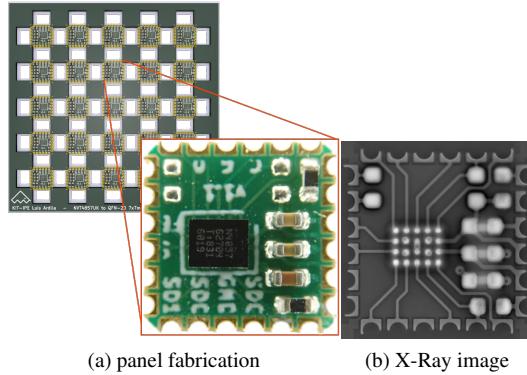


Figure 4: microSD controller module

4 IPMC Software Implementations

IPMC is a fundamental component for electronic boards using the Advanced Telecommunications Computing Architecture (ATCA) standard. It is responsible for monitoring the health parameters of the board, managing its power states, and providing board control, debug, and recovery functionalities to remote clients. The IPMC works out of the 3.3 V standby rail with a total maximum allowed power of 11 W. The Trenz-Serenity Adapter has been successfully used with two different software implementations of the IPMC, both targeting the ARM-R5 cores in lock-step mode mapping the First Stage BootLoader (FSBL) to the Low-Power Domain (LPD) of the ZUS+ as shown in Figure 5.

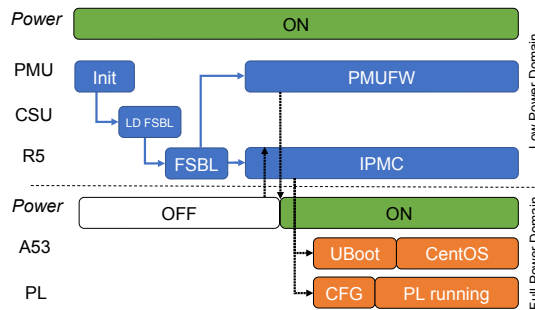


Figure 5: Boot sequence of the Zynq Ultrascale+ device

A custom hardware project was created using the Vivado system block design tool, where an isolation configuration was put in place to map peripherals exclusively to the LPD. The Power Management Unit (PMU) was as well customized to handle the reset release of the Full-Power Domain (FPD) once the payload power was enabled. Upon insertion of a Field Replaceable Unit (FRU), the 3.3 V standby power available in the PIM400 module turns ON the LPD of the ZUS+, where the boot process loads the FSLB in the R5 cores which then instructs the PMU to reset and load the IPMC software. At this point, communication with the Shelf Manager (ShM) begins.

The IPMC starts by communicating the transition from the virtual state M0 to state M1, where the FRU is inactive. The ShM requests the Device ID and information about the installed FRU like the name, firmware version, and supported features. When the front panel handle is in the closed position, the IPMC notifies the transition M1-M2 state, and the ShM responds, requesting the SDRs of the blade. The ShM instructs the transition M2-M3. In M3, the power negotiation takes place to allocate available power in the crate according to the board requirements. The FRU information is also requested by the ShM to know which E-Key interfaces to enable. At this point, the board enables the payload power, configures the programmable logic, and launches the uboot loader followed by CentOS7 on the A53 cores. Then the IPMC notifies the M4 FRU active state to the ShM. The deactivation process starts by opening the handle or by request of the ShM bringing the FRU state back to M1.

4.1 Pigeon Point IPMC Software

The Pigeon Point IPMC software is a commercially available software stack that has been tailored and customized for utilization on the ZUS+ architecture. It is based on an implementation targeting other processor architectures. Therefore the implementation on ZUS+ is not yet an off-the-shelf choice but rather an on-going development here presented. Inside the software, there is the functionality to select where peripherals are connected utilizing a hardware abstraction layer, minimizing the number of files that need user modification. However, this does not apply to the more underlying features requiring specific architecture functions to instruct the PMU of the ZUS+ device, for example.

The Pigeon Point implementation is extensive and complete. However, not all the features in the code apply to the setup and have not been validated using the ZUS+ architecture. Some require specific hardware components that are not present in the current setup. Further enabling and validation of features will take place in the future.

4.2 OpenIPMC Software

OpenIPMC [13] is a free and open-source firmware designed to operate as an Intelligent Platform Management Controller (IPMC). It is based on the FreeRTOS real-time operating system, and it is designed to be architecture-independent, allowing it to be run on a variety of different processor architectures. Having open-source code is novel for this kind of firmware, allowing full customization by the user.

The OpenIPMC makes use of signal call-backs and a hardware abstraction layer to hide from the main functionality the specifics of the hardware platform it is running on. The code has been tested on two different hardware platforms with identical results on the implemented functions. The code has been written to prioritize those core functionalities in the ATCA standard, which are essential and continued towards those required but not critical. So far, the dual multi-master IPMB_A/B interface has been implemented, bringing the board to the state where the payload is active. Power-up and -down sequences were verified, signaling the transition states using the blue ATCA LED in the front panel. FRU information is stored and read from an EEPROM memory.

5 Results

The IPMC functionality was verified via its redundant dual connection to the backplane with two independent software implementations achieving the mandatory power-up sequence, negotiating the different ATCA states with the shelf manager. The software "ATCA Tester"

from Polaris Networks [14] was used to test the level of conformity against the standard with the Pigeon Point implementation. Of the 106 tests performed, 72 were successful, and 34 failed. The faulty features were not enabled in the software at the time of testing and therefore expected to fail. The OpenIPMC implementation was tested in an ATCA crate but not formally using the conformance test yet.

6 Conclusion

Future DAQ systems will require local processing intelligence for slow-control, monitoring, and calibration. Heterogeneous SoC architectures allow integration of these tasks into a single device. It was demonstrated that a unified slow control architecture based on Zynq Ultrascale+ SoC is possible. Implementing time-critical applications like the IPMC on the R5 cores and a CentOS Linux on the application A53 processors was achieved utilizing isolation of memory and unique assignment of peripherals to the two software stacks. Two IPMC implementations were explored; a full-featured commercial solution and a basic-feature open-source version. Though this is still work in progress, the results are very promising. Our future work will focus on implementing the missing features and testing in a realistic environment.

Acknowledgments

The authors wish to thank Andrew Rose and Gregory Iles for their valuable contribution in designing the Serenity board and for providing us with one of the first prototypes for validation of our ideas. This research acknowledges the support by the DFG-funded Doctoral School "Karlsruhe School of Elementary and Astroparticle Physics: Science and Technology"

References

- [1] D. Contardo, M. Klute, J. Mans, L. Silvestris, J. Butler, Tech. Rep. CERN-LHCC-2015-010. LHCC-P-008. CMS-TDR-15-02, Geneva (2015), <https://cds.cern.ch/record/2020886>
- [2] CMS Collaboration, Tech. Rep. CERN-LHCC-2017-009. CMS-TDR-014, CERN, Geneva (2017), <http://cds.cern.ch/record/2272264>
- [3] G. Hall, Nucl. Instrum. Methods Phys. Res. **824**, 292 (2016)
- [4] L. Ardila-Perez, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **936**, 329 (2019), frontier Detectors for Frontier Physics: 14th Pisa Meeting on Advanced Detectors
- [5] C. Collaboration, Tech. Rep. CERN-LHCC-2017-013. CMS-TDR-017, CERN, Geneva (2017), this is the CMS Interim TDR devoted to the upgrade of the CMS L1 trigger in view of the HL-LHC running, as approved by the LHCC., <https://cds.cern.ch/record/2283192>
- [6] A. Rose et al., *Serenity: An ATCA prototyping platform for CMS Phase-2*, in *Topical Workshop on Electronics for Particle Physics (TWEPP18): Antwerpen, Belgium, September 17-21, 2018* (2019), Vol. TWEPP2018, p. 115
- [7] J.M. Mendez, V. Bobillier, S.L. Haas, M. Joos, S. Mico, F. Vasey, *CERN-IPMC solution for AdvancedTCA blades*, in *Topical Workshop on Electronics for Particle Physics (TWEPP17): Santa Cruz, CA, USA, September 11-15, 2017* (2018), Vol. TWEPP-17, p. 053

- [8] Pigeon Point, *Pigeon Point's PICMG products* (2020), http://www.pigeonpoint.com/products_picmg.html
- [9] A. Albert et al., *The Apollo ATCA Platform*, in *Topical Workshop on Electronics for Particle Physics (TWEPP19) Santiago de Compostela, Spain, September 2-6, 2019* (2019), 1911.06452
- [10] E. GmbH, *Mercury ZX1 Xilinx Zynq 7030/7035/7045 SoC Module* (2020), <https://www.enclustra.com/en/products/system-on-chip-modules/mercury-zx1/>
- [11] M. Vicente, T. Gorski, A. Svetek, J. Tikalsky, R. Fobes, S. Dasu, W. Smith, PoS **TWEPP-17**, 102 (2017)
- [12] Trenz Electronic GmbH, *TE0803 - Zynq UltraScale+* (2018), <https://shop.trenz-electronic.de/en/Products/Trenz-Electronic/TE08XX-Zynq-UltraScale/TE0803-Zynq-UltraScale/>
- [13] A. Cascadan, L. Calligaris, A. França, A. Shinoda, B. Casu, L. Ramalho, S. de Souza, L. Ardila, O. Sander, *OpenIPMC: a free open source Intelligent Platform Management Controller for AdvancedTCA* (2020), <https://gitlab.com/openipmc/openipmc>
- [14] Polaris Networks Inc , *AtcaTest3* (2020), https://members.picmg.org/kshowcase/view/view_item/f66bd9bbd767677d1e7ece11ca07c31a5ffca72d