Published by IOP Publishing for Sissa Medialab

Received: *November 3, 2024*
Accepted: *December 19, 2024*
Published: *January 16, 2025*

Topical Workshop on Electronics for Particle Physics
University of Glasgow, Scotland, U.K.
30 September–4 October 2024

EMCI-EMP: developments and experience with the novel detector control solution

D. Ecker[a,b,*] **P. Dubru**[b] **S. Foged**[c] **N. Kanellos**[b,d] **P. Moschovakos**[b] **V. Ryjov**[b] **and S. Schlenker**[b]

[a] *Bergische Universität Wuppertal,*
Gaußstraße 20, 42119 Wuppertal, Germany
[b] *CERN,*
1211 Geneva 23, Switzerland
[c] *Aarhus University,*
Nordre Ringgade 1, 8000 Aarhus Centrum, Denmark
[d] *National Technical University of Athens,*
Leof. Alimou Katechaki, Zografou 157 72, Athens, Greece

E-mail: dominic.ecker@cern.ch

Abstract: The Embedded Monitoring and Control Interface (EMCI) and Embedded Monitoring Processor (EMP) system is a state-of-the-art solution developed for the Detector Control System (DCS) upgrade in the ATLAS experiment. The EMCI serves as a radiation-tolerant interface for control and monitoring data of the detector Front-End electronics. The EMP, a multi-processing System on Chip (MPSoC) platform, serves as the interface between the EMCIs and the Back-End of the DCS, providing centralized data handling and monitoring capabilities. The firmware and software ecosystem includes the EMP operating system (*epos*), dedicated firmware IP blocks, associated software libraries, and OPC UA based middleware applications. This paper focuses on the ongoing hardware verification and development of the firmware and software infrastructure for the EMCI-EMP system.

Keywords: Detector control systems (detector and experiment monitoring and slow-control systems, architecture, hardware, algorithms, databases); Software Engineering

*Corresponding author.

https://doi.org/10.1088/1748-0221/20/01/C01020

2025 JINST 20 C01020

Contents

2025 JINST 20 C01020

1 Introduction

The primary task of the ATLAS experiment's Detector Control System (DCS) [1] is to ensure safe and coherent operation of the detector by serving as a homogeneous interface to all subsystems and technical infrastructure, allowing their operational status to be controlled, operational parameters to be monitored and abnormal behavior to be reported. The DCS must operate independently of the experiment's power and communication systems and always maintain uninterrupted operation. Additionally, the equipment located in the detector cavern must be resistant to magnetic fields and possess the necessary radiation tolerance during the experiment's lifespan. In the framework of the HL-LHC project, new challenges must be addressed to maintain the operational integrity of the DCS. These challenges, including the need for higher I/O density and enhanced radiation tolerance of hardware, have been addressed through the development of the EMCI-EMP system [2].

2 The novel DCS interface solution for on-detector Front-End components

The original ATLAS DCS hardware components have demonstrated consistent performance over the years. Given the outlined challenges the current hardware and its infrastructure cannot meet the demands of the HL-LHC environment. The developed EMCI-EMP solution is designed to complement the existing DCS. It offers a scalable architecture and supports communication rates with Front-End (FE) electronics of up to 320 Mbps, which is orders of magnitude higher than the current standard (max. 500 kbps). A diagram of the system architecture is presented in figure 1.

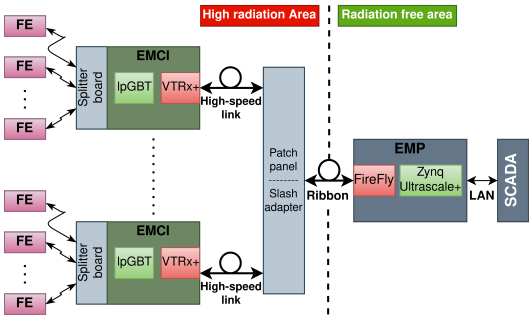

Figure 1. EMCI-EMP system overview.

The EMCI [3], built with radiation-tolerant CERN-developed components (lpGBT [4], VTRx+ [5], and FEASTMP [6]), can be positioned based on detector needs. Conversely, the EMP, which is based on a highly flexible System-on-Chip (SoC) device and other commercial off-the-shelf (COTS) components, cannot function in a radiation environment.

The purpose of the EMCI is to serve as an interface for control and monitoring data exchanged between various detector FEs and the DCS system. It is based on the lpGBT ASIC, which aggregates all FE signals into a single bidirectional channel and interfaces with the VTRx+, an optical transceiver that transmits data through a high-speed optical link.

The EMP [7] is a compact processing platform that serves as the backbone of the DCS control and monitoring system. It connects up to 12 EMCIs in a star topology through optical links and interfaces with the Supervisory Control and Data Acquisition (SCADA) system via Ethernet. To accommodate and power the EMP modules, a custom crate was developed. It complies with the 19-inch, 1U rack-mount chassis standard and can house up to three EMP modules along with a redundant power supply.

2.1 EMP hardware design

The "heart" of the EMP is a TE0807 MPSoC plug-on mezzanine module (SoM) from Trenz Electronic [8], equipped with an AMD Xilinx Zynq Ultrascale+ device [9]. The Zynq chip integrates a 64-bit quad-core ARM Cortex-A53 Application Processing Unit (APU) and Programmable Logic (PL). The module is mounted on the custom-developed EMP baseboard which provides access to most of the available communication interfaces and GPIOs of the SoM. It features 16 GTH Multi-Gigabit Transceivers (MGTs), 12 of which are connected to FireFly transceivers [10] and one to an SFP+ transceiver. All these 13 links comply with Versatile Link+ (VL+) specifications [11], a bi-directional optical data link to operate in tandem with the lpGBT. Additionally, the EMP provides access to the other digital and analog peripherals of the MPSoC for control and monitoring (GPIOs, USB3, UART, ADC, etc.). Figure 2 presents an overview of the MPSoC module and baseboard components.

2.2 Operational spectrum of the EMCI-EMP interface

The EMCI-EMP interface is a highly customizable and versatile system, allowing the implementation of multiple functional entities operating concurrently:

- Monitoring and control of EMP hardware: configures and monitors on-board peripherals, provides clocking for the PL, and gathers sensor data to assess EMP hardware status.
- Control and configuration of the lpGBT ASIC on the EMCI via the Internal Control (IC) channel: manages lpGBT register configuration, monitoring, and ASIC management, with IC communication supporting slow control I/O pins of the lpGBT.
- FE-based target applications: supports custom protocols over lpGBT eLinks, enabling DC-balanced data transfer with FEC and integration of FE modules into the DCS via EMP firmware.

Additional potential applications include the use of single ended or differential signaling via the SoM's I/Os, the deployment of standard interfaces like CAN, UART, SPI, and I2C using PL IP cores, as well as the adoption of processing system (PS) interfaces such as Multiplexed I/Os (MIOs) or UART.

2025 JINST 20 C01020

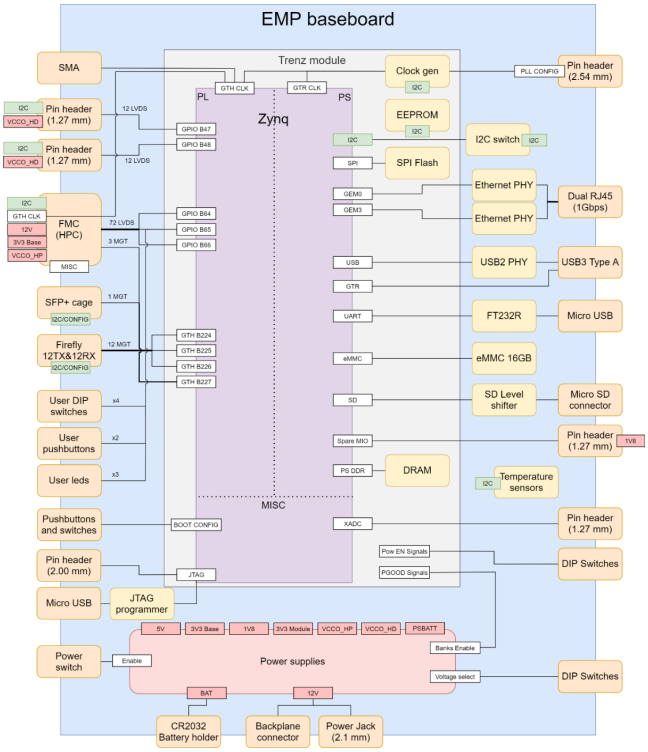

Figure 2. MPSoC and EMP baseboard design.

3 The EMP software and firmware environment

The configuration of each EMP is determined by its specific operational context and requirements, with the final application integrating multiple target-specific applications. These target applications comprise several standardized and custom software, firmware, and hardware components. To streamline development, a modular architectural structure is provided as reference, utilizing standardized tools and reusable building blocks as illustrated in figure 3. This includes support for shared hardware functions (e.g., Ethernet), standardized software elements (e.g., operating system), and standardized firmware IP cores. While this framework provides a basic structure that Front-End (FE) developers can leverage, they are still responsible for creating device-specific components tailored to their respective Front-End hardware.

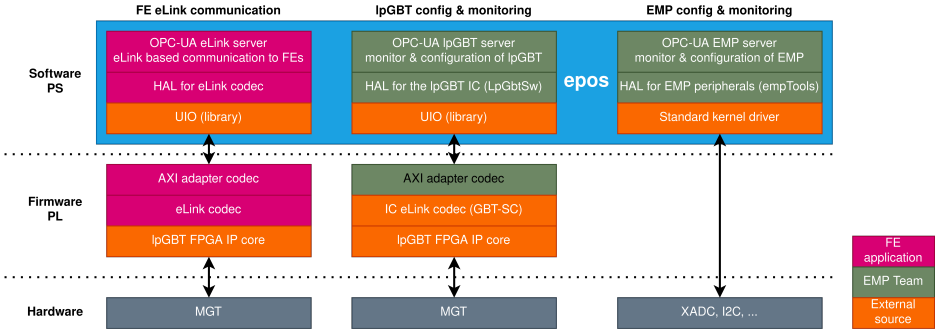

Figure 3. Modular architecture of the EMCI-EMP interface, showing building blocks for specific target applications. Pink denotes components from external sources; green indicates modules provided as part of the reference software; orange represents custom building blocks to be developed for the respective FE hardware.

2025 JINST 20 C01020

epos — the EmP Operating System. epos [12] provides a unified framework to minimize custom development of shared EMP features and ensures a high level of standardization within the DCS. It consists of two main components: epos-bsp, which includes a PetaLinux board support package (BSP) for building and packaging the EMP boot firmware, U-Boot configuration, and Linux kernel image; and epos-rootfs, which offers tools to create a production-ready Linux root filesystem with essential packages. Additionally, the epos repository includes tools for generating device tree blob overlays (DTBOs), enabling dynamic firmware loading from the PS during runtime.

quasar OPC UA Servers for EMP. quasar [13] is a generic framework for the generation of model-driven OPC UA servers [14], corresponding OPC UA clients and SCADA tools for the establishment of OPC UA connectivity. Given that OPC UA is the mandatory middleware standard for all DCS-related functions, such as FE communication and the monitoring of the EMP itself, the framework is the recommended tool for creating OPC UA servers operating within the EMP.

Hardware access layer (HAL). High-level application programming interfaces (APIs) have been developed to provide software access to EMP and EMCI hardware. These APIs are utilized by the quasar framework as a HAL to implement the device-specific logic for the respective servers.

The LpGbtSw library [15] is a versatile C++-based solution that provides register-based interactions with the lpGBT through a polymorphic API. It is designed to support a range of lpGBT functionalities, including configuration and slow-control interfaces (such as ADC, GPIO, and I2C).

The empTools [16] provides tools for configuring, controlling, and monitoring EMP peripherals. It interfaces with the EMP firmware via an AXI register map to access monitoring information and reset signals, and connects EMP peripherals through standard kernel drivers.

The EMP reference firmware and AXI Adapter Core. The EMP reference firmware integrates a range of custom and standard IP blocks, establishing the interface between the PS of the EMP and the lpGBT. Central to the firmware is the custom EMP-lpGBT IP Core, which is built upon the lpGBT-FPGA IP Core [18, 19] and the GBT-SC IP Core [20]. The IP core provides the necessary lpGBT frame encoding and decoding schemas (lpGBT-FPGA) and implements EC and IC channel communication protocols (GBT-SC), enabling read and write operations on lpGBT registers.

The EMP team also developed a custom AXI-Adapter Core, generated using AirHdl [21], to streamline access to the GBT-SC IC Core from the PS. This core interfaces via AXI4-Lite, enabling register map based peripheral connectivity supported by the UserIO (uio) software model [22]. Additionally, the Puzzled Lizard Wizard tool [23] generates AXI4-Lite register maps in software, providing full access to the low-level hardware abstraction layer (HAL) in C++ with integrated libuio support, optimizing both AXI4-Lite integration and software accessibility.

The down-/uplink signals are processed by an MGT IP Core, interfacing the FireFly transceiver modules at the lpGBT transmission rate of 10.24 Gbps.

4 EMP and EMCI test benches

The complexity of the EMCI and EMP modules has led to the development of two dedicated testing and validation platforms, particularly with the upcoming production volumes. Each test bench, equipped with custom hardware, firmware, and software components, validates all EMCI and EMP

2025 JINST 20 C01020

functionalities through a series of comprehensive tests. Both platforms are semi-automated, featuring a graphical user interface (GUI) and test result logging.

The EMCI test bench is based on an AMD Virtex 7 FPGA VC707 Evaluation Kit, with a custom-designed FMC mezzanine for interfacing with the EMCI. This setup covers all board-related functionalities, including optical communication, I2C, eLinks, GPIOs, ADC, and DAC.

The EMP test bench operates directly on the EMP and interfaces with the tester GUI via a TCP-based client-server connection allowing the EMP to self-test all interfaces, communicates results, and generates a test report. Additionaly, an Integrated Bit Error Ratio Tester (IBERT) IP core [24] evaluates all high-speed optical links with transmission rates of 10.24 Gbps and in particular every GPIO pin on the baseboard undergoes a functional test.

5 Thermal study

During the validation of the EMP crate, which houses up to three EMP baseboards and a redundant power supply, effective heat dissipation was identified as a critical factor. Key EMP components, like the FireFly transceiver modules, must meet strict temperature requirements for continuous operation. To prevent accelerated aging it is required to keep the "Case Temperature" — the internal, monitorable temperature of the FireFly heatsink — at 50 °C or lower. To assess this, a thermal study was conducted, deploying the fully populated crate in a standard production rack equipped with horizontal backdoor ventilation [25]. The study identified close correlations between crate internal component temperatures and surrounding ambient conditions. Under normal operational conditions where there is no cooling system failure or maintenance, the most sensitive components remained with temperature margins >10 °C with respect to their specification limits.

6 Conclusion

The development of the EMCI-EMP system marks a significant improvement in the ATLAS DCS, addressing key challenges posed by the HL-LHC environment, including the need for higher radiation tolerance, enhanced data throughput and I/O density. By integrating radiation-tolerant FE devices and a flexible BE SoC platform, the system offers reliable control and monitoring of on/off-detector components. The EMP's firmware and software, including its reference FPGA IPs and standardized "*epos*" operating system, ensure consistency and ease of deployment. Extensive testing has confirmed the system's performance and thermal stability, making the EMCI-EMP interface a robust, scalable solution for maintaining the detector's operational integrity throughout the HL-LHC's lifespan.

References

[1] A. Barriuso Poy et al., *The detector control system of the ATLAS experiment*, 2008 *JINST* **3** P05006.

[2] P. Moschovakos et al., *The Embedded Monitoring Processor for High Luminosity LHC*, *JACoW I* **CALEPCS2023** (2023) 1470.

[3] EMCI-EMP team, *Embedded Monitoring and Control Interface*, https://edms.cern.ch/ui/file/2332346/2/EMCI_electronic_specifications_v1.3_docx_cpdf.pdf.

[4] GBT team, *lpGBT Manual*, https://lpgbt.web.cern.ch/lpgbt/v1/.

2025 JINST 20 C01020

[5] J. Troska, *Versatile Link+ Transceiver (VTRx+). Technical Specification, part 2.1*, https://edms.cern.ch/ui/file/1719329/1/VTRxPlus_spec.pdf.

[6] DCDC converters team, *FEASTMP Datasheet*, https://power-distribution.web.cern.ch/assets/datasheets/FEAST2Mod_Datasheet_gb2016.pdf.

[7] EMCI-EMP team, *Embedded Monitoring Processor*, https://edms.cern.ch/ui/file/2631648/1/EMP_hardware_specifications_v1.4_docx_cpdf.pdf.

[8] Trenz Electronic, *Trenz Electronic TE0807 MPSoC Modules with AMD Zynq™ UltraScale+™*, https://shop.trenz-electronic.de/en/Products/Trenz-Electronic/TE08XX-Zynq-UltraScale/TE0807-Zynq-UltraScale/.

[9] Xilinx, *Zynq UltraScale+ MPSoC Data Sheet: Overview*, https://docs.amd.com/v/u/en-US/ds891-zynq-ultrascale-plus-overview.

[10] Samtec, *CERN Versatile Link Plus FireFly™ Data Sheet*, Samtec Confidential (2019).

[11] J. Troska, C. Soos and L. Olanterä, *The Versatile Link+ Application Note*, https://edms.cern.ch/ui/file/2149674/1/VTRxPlusApplicationNote.pdf.

[12] P. Moschovakos, *epos gitlab repository*, https://gitlab.cern.ch/emci-emp/epos.

[13] P. Nikiel, P. Moschovakos and S. Schlenker, *quasar: The Full-Stack Solution for Creation of OPC-UA Middleware*, in the proceedings of the *17th International Conference on Accelerator and Large Experimental Physics Control Systems*, New York, NY, U.S.A. (2020) [DOI:10.18429/JACoW-ICALEPCS2019-MOPHA100].

[14] M. Damm, S.H. Leitner and W. Mahnke, *OPC unified architecture*, Springer (2009) [DOI:10.1007/978-3-540-68899-0].

[15] P. Moschovakos, *LpGbtSoftware gitlab repository*, https://gitlab.cern.ch/atlas-dcs-emp/LpGbtSw.

[16] D. Ecker, *emp-tools gitlab repository*, https://gitlab.cern.ch/emci-emp/emp-tools.

[17] EMCI-EMP team, *emp-firmware gitlab repository*, https://gitlab.cern.ch/emci-emp/emp-firmware.

[18] GBT team, *LPGBT-FPGA gitlab repository*, https://lpgbt-fpga.web.cern.ch/doc/html/.

[19] GBT team, *LpGBT-FPGA Documentation*, https://lpgbt-fpga.web.cern.ch/doc/html/.

[20] GBT team, *GBT-SC gitlab repository*, https://gitlab.cern.ch/gbtsc-fpga-support/gbt-sc.

[21] noasic GmbH, *Register Management Done Right* , https://airhdl.com/#/.

[22] Benedikt Spranger, *libuio — UserspaceIO helper library*, https://gitlab.cern.ch/gbtsc-fpga-support/gbt-sc.

[23] P. Nikiel, P. Moschovakos and M. Kristensen, *PuzzledLizard Wizard gitlab repository*, https://github.com/missinglinkelectronics/libuio/.

[24] Xilinx, *IBERT for UltraScale GTH Transceivers v1.4*, https://docs.amd.com/v/u/en-US/pg173-ibert-ultrascale-gth.

[25] F. Wickens, *The LHC PC Rack Project*, in the proceedings of the *10th LECC Workshop*, Boston, U.S.A. (2004).

2025 JINST 20 C01020