Early prototyping and testing of CERN LHC CMS high-granularity calorimeter slow-control system

Martim Rosado*†, Stavros Mallios*, Pedro Tomás†, Nuno Roma† and André David*

*on behalf of the CMS Collaboration, CERN, Geneva, Switzerland
†INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal

Abstract—The Compact Muon Solenoid (CMS) high-granularity calorimeter (HGCAL) upgrade for CERN's Large Hadron Collider (LHC) high-luminosity phase is a detector with more than 6 million channels that will provide precise sensing and measurement of position, timing, and energy of the particles produced in the collisions of the beams. The HGCAL electronics are a large and complex set of processing systems split into front-end and back-end. The front-end, located in the experimental cavern, consists of ≈150 thousand radiation tolerant ASICs. The high-density FPGA-based back-end is housed away from the radiation area in a set of Advanced Telecommunications Computing Architecture (ATCA) boards and crates hosting ≈100 FPGAs. Each ATCA back-end board will comprise one (or two) FPGAs, managing up to ≈120 optical links, each providing a transmission rate of 10.24 Gb/s between the back-end and the front-end electronics. Each back-end FPGA is responsible for configuring and monitoring up to ≈3500 front-end ASICs and will be controlled by software running on a back-end MPSoC that provides the entry point for the whole control procedure. This paper presents the design and implementation of the prototyping infrastructure deployed to test and validate the slow-control block of the HGCAL back-end electronics, together with the related interfaces with the controller MPSoC and the front-end transceiver ASICs. The required functionalities have been validated with a ZCU102 Xilinx Ultrascale+ development board, which emulated the back-end elements that are still under development and not yet available for this comprehensive test. This development board was connected to other custom ASIC development boards via optical links, emulating the front-end side of the system, also still under development. Besides providing reliable testing and validation of the operation of the whole infrastructure, the prototyping platform also allowed to attain the required software/hardware portability that ensures easy integration/replacement of all the (still) emulated components with their final implementations.

Index Terms—Fast Prototyping, System Emulation, Testing and Validation, Field-Programmable Gate-Array

I. INTRODUCTION

The High Luminosity phase of the Large Hadron Collider (HL-LHC) at CERN (European Organization for Nuclear Research) motivates the replacement and upgrade of multiple sub-detectors of the Compact Muon Solenoid (CMS) detector. The upgrade is needed because the current detector will not have an acceptable performance under HL-LHC operation conditions with the higher levels of radiation. The High Granularity Calorimeter (HGCAL) [1] is a new detector that will replace the current CMS endcap calorimeter systems, which comprise both the electromagnetic and the hadronic calorimeters. With more than 6 million channels, the HGCAL will measure position, timing and energy of particles produced in collisions of the LHC beams with unprecedented resolution.

The intense radiation environment necessitates the use of a resilient electronics chain which is based on the use of radiation-tolerant ASICs (Application Specific Integrated Circuit) in the front-end electronics of the HGCAL detector. These ASICs read the more than 6 million active channels, then digitise and process the signals acquired from silicon sensors and from plastic scintillator tiles read out by silicon photomultipliers. That information is conveyed using 10.24 Gb/s optical links to ≈100 large (radiation sensitive) FPGAs (Field Programmable Gate Array) mounted on Advanced Telecommunications Computing Architecture (ATCA) boards in the back-end (off detector). Each back-end FPGA is responsible not only for processing data from up to ≈120 of these links but also for controlling and monitoring up to 3.5 thousand front-end ASICs producing those data [2]. Two types of commands are needed to control the front-end ASICs:

1) Synchronous commands (fast-control) responsible for triggering and managing data acquisition, e.g. preventing buffer overflows in the FPGAs. These commands are sent to the front-end in a 320 Mb/s serial bit stream.

2) Asynchronous commands (slow-control) responsible for configuring and monitoring the front-end ASIC parameters. These commands are issued asynchronously to the data-taking and are transmitted via an 80 Mb/s serial bit stream; the ASIC configuration does not need to be changed often, thus, these commands do not need to be sent at a high frequency.

Considering the cost and magnitude of the overall detection and processing system, the feasibility of the concept must be thoroughly tested and prototyped. Consequently, performing rapid prototyping of a vertical slice of the HGCAL global system was required, i.e. a system in which the front-end sensors are connected to the back-end electronics with the minimum number of components. The system will then grow horizontally, integrating more components of the same type, up to the size of the final prototype. As part of this effort, it was decided to prototype the FPGA hardware block that will handle sending slow-control commands to the front-end ASICs, ahead of the availability of the final FPGAs and ATCA modules.

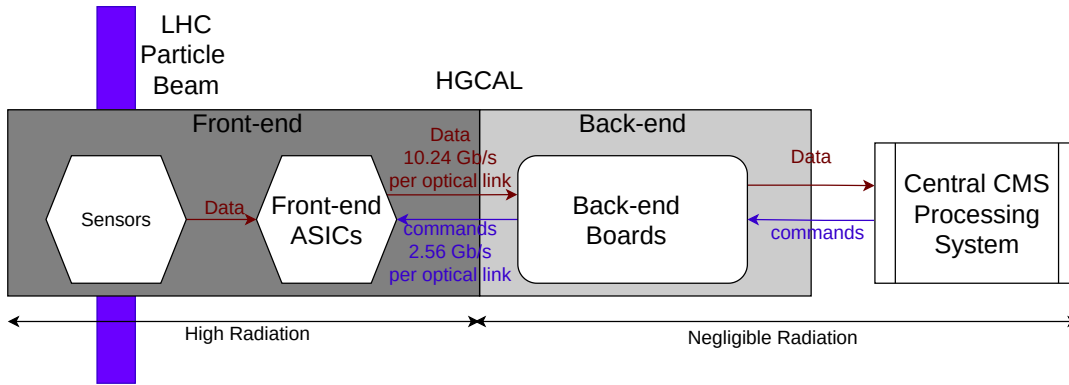The main contribution of this work is the conception of

Fig. 1. Schematic overview of the HGCAL electronics systems, including those on-detector in the high-radiation environment and those off-detector in a shielded underground cavern.

a prototyping system used to test and validate the interface between the slow-control block and the radiation-tolerant ASICs of the final detector. The validation of the slow-control block allows for other HGCAL test systems and prototypes to use the final detector framework of front-end configuration.

Since both the final back-end infrastructure and ATCA communication boards are still under development and not available for testing, a ZCU102 Xilinx FPGA development board [3], equipped with a Zynq MPSoC, was used to emulate the back-end platform. The front-end infrastructure was also emulated with custom development boards containing the target ASICs. This way, even though the final detector boards might be different from those used in this prototyped emulation, it is possible to test the communication infrastructure with the detector ASICs that will be used in the final system, validating the whole configuration chain.

Moreover, this prototyping infrastructure allowed a comprehensive portability evaluation of both the software and the FPGA-deployed hardware elements that will be present in the final system. Furthermore, this setup allowed for the parallel development of the slow-control FPGA hardware, its interface software, and the corresponding feasibility studies for its integration into the final back-end system (to be implemented in FPGA).

The remainder of this paper is organized as follows: Section II briefly describes the HGCAL detector system; Section III describes the slow-control block and its interactions and interfaces within the broader system; The developed test system is the subject of section IV along with a discussion of the test procedures; Results are presented in section V; The conclusion in section VI; Finally, section VII discusses future work.

II. CMS HIGH GRANULARITY CALORIMETER

The High Granularity Calorimeter of CMS is a particle detector being developed in the scope of the HL-LHC upgrade, expected to be completed by the end of 2027. This detector will sample data (timing, energy, and position) of particle collisions with unprecedented high resolution. The HGCAL electronics system is divided into two distinct parts: the front-end and the back-end, as shown in Fig. 1.

The front-end electronics are located in the CMS detector and hence operate in a high radiation area. With the use of silicon sensors and plastic scintillator tiles as sensitive materials, data gathered from the particle collisions are acquired through more than 6 million channels and undergo initial processing before being conveyed to the back-end electronics. The front-end electronics are based on radiation-tolerant ASICs to deal with the high radiation environment.

The back-end is connected to the front-end via optical links and is located off the detector, safe from radiation. It receives the front-end data at 10.24 Gb/s per link. The back-end incorporates several ATCA communication boards, each with one (or two) VU13P (Virtex Ultrascale+) Xilinx FPGAs. The back-end not only receives data from the front-end but also pre-processes it before sending the data to the central CMS processing system. Each back-end FPGA manages data from up to ≈120 of these links while also configuring and monitoring up to ≈3500 front-end ASICs.

III. SYSTEM BEING PROTOTYPED

The slow-control block is located in the back-end FPGA on the ATCA boards (see Fig. 2) and is responsible for configuring and monitoring the front-end ASICs. This is achieved with multiple channels through which transactions are performed, consisting of several read/write operations to the front-end ASICs' internal registers. The information is communicated to the slow-control via a software layer running on a Zynq MPSoC, also hosted on the back-end ATCA board. This MPSoC provides the software with an entry point that opens a communication line with the back-end FPGA via Advanced eXtensible Interface (AXI) Chip2Chip [4], [5].

Each slow-control block interfaces with up to 756 front-end ASICs of two different types:

1) The Low Power Giga Bit Transceiver (lpGBT) [6] ASIC that provides high-speed bidirectional communication, and

2) The Giga Bit Transceiver - Slow-Control Adapter (GBT-SCA) [7] ASIC, which was specifically developed for sending configuration and monitoring commands throughout the front-end of particle detectors.
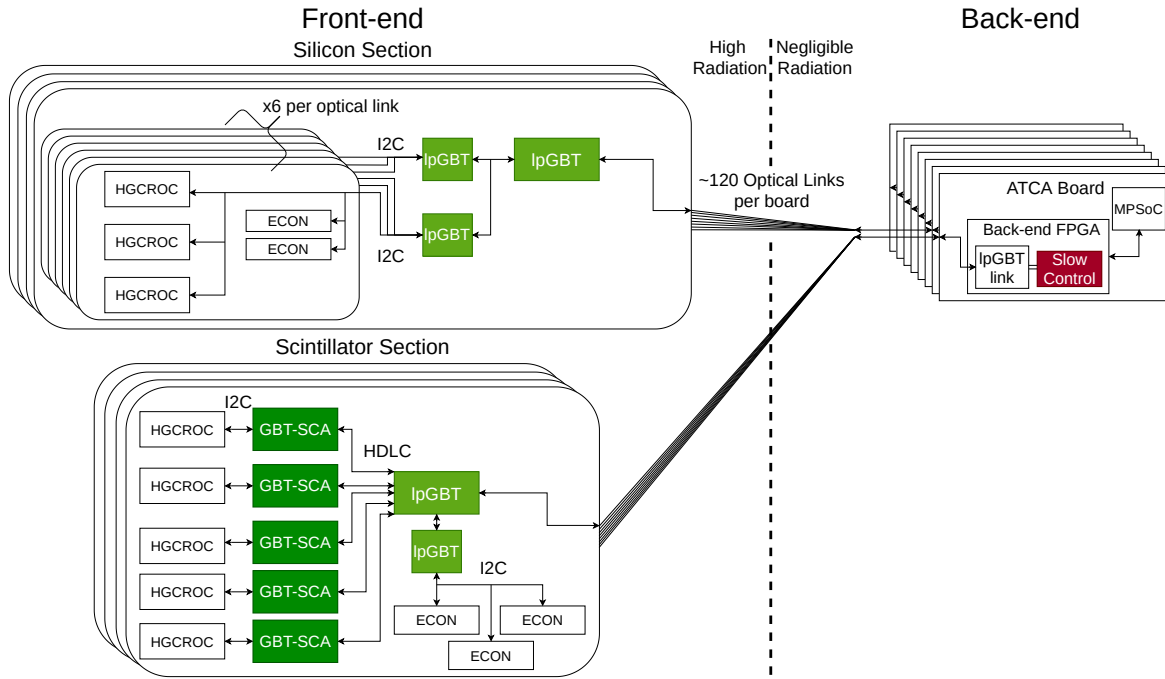
Fig. 2. Main interfaces and front-end endpoints of the back-end slow-control block being prototyped in this work.

A. Slow-control Connection to the Front-end

The HGCAL front-end is divided into two sections by the type of sensitive material used; silicon sensor (silicon section) or plastic scintillator (scintillator section), leading to two architectures, shown in Fig. 2.

After particle collisions, the sensors' signals are first digitized by the HGCAL readout chip (HGCROC). Next, the data are transmitted to the Elink concentrator ASICs (ECON) and pre-processed before being sent to the back-end. Both the HGCROC ASICs and ECON ASICs are configured via I2C using either an lpGBT or GBT-SCA that have I2C masters that are controlled by the slow-control block in the back-end.

The silicon section does not use GBT-SCAs and has a higher ASIC density per back-end FPGA than the scintillator section. For communication, each back-end FPGA will service ≈120 optical links, each connected to a single lpGBT. Each lpGBT is subsequently connected to:

1) 2 other lpGBTs, further connected to ≈18 HGCROCs and ≈12 ECONs (I2C targets), totalling up to ≈3500 ASICs to configure per back-end FPGA in the silicon section, and
2) 1 other lpGBT, up to 5 GBT-SCAs and up to ≈8 I2C targets, totalling up to ≈1600 ASICs to configure per back-end FPGA in the scintillator section.

Although the GBT-SCA data is transmitted through the lpGBTs, the GBT-SCA transactions are transparent to the lpGBT; therefore, the GBT-SCAs can be considered to be directly interfaced to the slow-control. The same applies to the lpGBTs not connected directly to the optical links.

To establish communication between the back-end and front-end through the slow-control block, the following procedure is undertaken:

1) The software running in the ATCA MPSoC transfers the data to the slow-control block in the back-end FPGA;
2) The slow-control block processes these data and issues the required read/write transactions to the target lpGBTs and/or GBT-SCAs;
3) The replies from the target lpGBTs and/or GBT-SCAs are read by the slow-control block and stored;
4) The MPSoC software reads the data in the replies from the slow-control block.

To fully understand the requirements, it is important to observe that, although the front-end configuration is not changed very often, it must be done with low latency so as to minimize the detector dead time when configuring the front-end electronics, as important data may be missed while the beams are colliding.

B. Communication Infrastructure and Protocols

To reach the front-end, the slow-control block implements the communication protocols of both the lpGBT and GBT-SCA transceivers [2]. The lpGBTs use a custom protocol [6], and the GBT-SCAs use the HDLC (High-level Data Link Control) protocol. These protocols allow the reading from and writing to registers in the lpGBTs and GBT-SCAs, such as configuration and status registers of their I2C masters. It should be noted that each transaction using these protocols triggers a reply from the corresponding lpGBT or GBT-SCA.

For each lpGBT and GBT-SCA interface, the slow-control block produces an 80 Mb/s output signal and receives another

80 Mb/s input signal. These bitstreams are implemented internally in the back-end FPGA as 2-bit encoded signals at 40 MHz and correspond to a subset of the full lpGBT data frame presented in Fig. 3, namely the Internal Control (IC) [6] and External Control (EC) [6] fields. Therefore these signals cannot be connected directly to the transceiver that sends data to the lpGBT and need to be provided to an intermediate block (lpGBT link) that is in charge of encoding the complete lpGBT data frames. The GBT-SCA stream is also connected to this lpGBT interface and uses part of the eLinks field [6].

Fig. 4. Simplified block diagram of the slow-control architecture, showing how a single FPGA holds a number of slow-control cores, each multiplexing signals to multiple lpGBT or GBT-SCA interfaces.
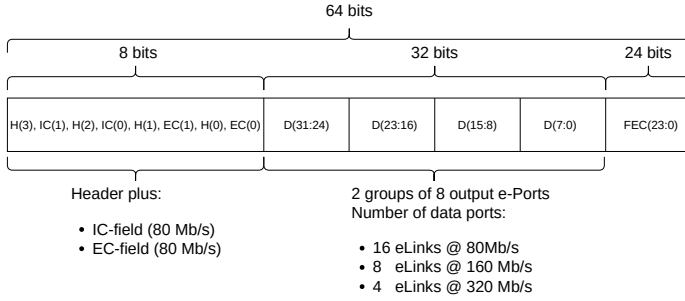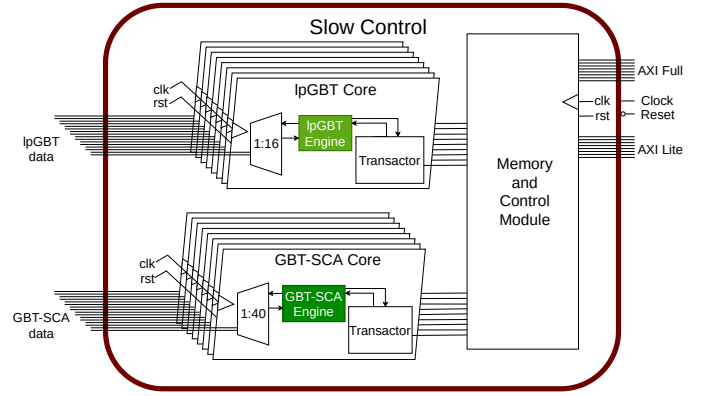

Fig. 3. The lpGBT data frame, from [6]. The slow-control block makes use of both the IC and EC fields that are dedicated to this type of communication and, in the scintillator section, also uses some of the eLinks to reach up to 5 GBT-SCAs in total.

In each optical link, the front-end lpGBT that is directly connected to the back-end is reached via the IC bitstream while the other lpGBTs are reached via the EC bitstream. As an example, in the silicon section, this EC channel is shared by two lpGBTs. Hence, per optical link, two 80 Mb/s output streams are required to interface up to 3 lpGBTs, and one lpGBT link must be instantiated in the back-end FPGA.

C. Slow-control Architecture

Because of FPGA resource limitations [2], it is not possible to implement a fully parallel structure that allows the transmission of simultaneous transactions to all lpGBT and GBT-SCA interfaces, so it is necessary to multiplex the transactions between several front-end lpGBTs or GBT-SCAs. To achieve a compromise between FPGA resource usage and detector configuration time, several small blocks with multiplexing capabilities, denoted as "lpGBT cores" or "GBT-SCA cores" [2], are used to achieve some level of parallelism. Fig. 4 shows the slow-control block architecture.

1) Memory and Control Module: A memory module is needed because each of these cores has to receive transaction data from the back-end MPSoC ARM core and store the front-end replies. This functionality is implemented through the use of BRAMs in the FPGA programmable logic. Each lpGBT or GBT-SCA core has a send buffer and a receive buffer, and its own set of status and control registers. The data word size for each transaction is 128 bits and it was decided to store data from 1024 transactions in each buffer. To accommodate this, both buffers are built with 4 BRAMs and have "AXI Full" connections to communicate with the MPSoC ARM cores.

The control and status registers are accessible via an "AXI Lite" interface.

2) The lpGBT and GBT-SCA Cores: Each core is associated with the front-end lpGBT or GBT-SCA communication protocol. The core is responsible for reading the transaction data from the respective buffer, processing it, and multiplexing the data to the respective 80 Mb/s stream, which corresponds to the correct lpGBT or GBT-SCA ASIC in the front-end. Afterwards, the core receives and stores the reply in the receive buffer. The current design [2] has a multiplexing factor of 16 in the lpGBT core and of 40 in the GBT-SCA core. With these configurations, it is possible to achieve a full detector coverage with 16 units of each core in each back-end FPGA, as depicted in Fig. 4.

Both core types are similar in structure: a small state machine (the transactor) interfaces with the buffers and controls the sending and receiving of data. It was chosen to not support concurrent transactions in each lpGBT and GBT-SCA core, meaning that one transaction only starts after the previous one is finished. This way, except in a timeout scenario, each transactor waits for the response of the current transaction before starting the next.

Each transactor output is fed to the respective engine (lpGBT or GBT-SCA). The engine is responsible for the decoding of software data into lpGBT and GBT-SCA transactions and encoding back the corresponding response. The engine output is routed to the channel connected to the target lpGBT or GBT-SCA.

The described chain corresponds to a complete sending procedure. A similar, but reverse, operation happens when receiving front-end replies. The response from the target lpGBT or GBT-SCA is sent to the engine that encodes the data into the data format expected by the MPSoC ARM core. The engine output is fed into the transactor, which stores the data into the respective receive buffer and then starts processing the next transaction.

IV. System Emulation, Verification, and Validation

Preliminary prototyping and subsequent verification of the slow-control block requires careful validation of both its internal operation and its interfaces, namely the AXI interface with the MPSoC ARM core, and the connection with the lpGBTs and GBT-SCAs. This section describes what was done to achieve such a comprehensive validation.

A. Emulation and Testing Infrastructure

The dimension and complexity of the sensing and data acquisition system that supports this CMS detector upgrade, allied with the fact that a significant amount of the intervening parts and components of the HGCAL are still under development and not already available for a complete deployment and system integration mean that certain modules to which the slow-control block will be interfaced are implemented in surrogate hardware, shown in Fig. 5. Such surrogates ensure accurate stimulus of the several elements, together with the reception and validation of the received replies.

Due to the current unavailability of a Zynq-controlled ATCA board prototype, a Xilinx ZCU102 development board was selected as the surrogate for the back-end platform. This board allows to connect the incoming optical links from the front-end at the required data rates to its SFP+ connectors and MGTs (Multi-Gigabit Transceiver). The ZCU102 is equipped with a Zynq MPSoC based on the same technology as the target back-end FPGA, the VU13P. In particular, both have Ultrascale+ FPGA fabric. Furthermore, the ARM processing system on the MPSoC also allows the use of AXI connections as in the final system, alongside a Linux installation, also used in the final system. The test system uses the CentOS 7 Linux distribution.

In the back-end board's programmable logic (PL), the slow-control block is connected to the MPSoC processing system (PS) through its two AXI interfaces and a Xilinx AXI interconnect IP. Its 80 Mb/s data streams to/from the front-end are connected to the lpGBT link module, which is responsible for parsing the complete lpGBT data frames and ensuring a reliable link between the back-end and front-end. The bitstreams corresponding to the lpGBT data were connected to the IC field of the lpGBT frame and the bitstreams corresponding to the GBT-SCA were packed in the EC field. Although in the final system the GBT-SCA connection will not use this EC field of the lpGBT frame, this setup is acceptable for slow-control prototyping purposes since it does not affect the validation of the communication protocol as GBT-SCA data remains transparent to the lpGBT.

The output of the lpGBT link is then connected to the ZCU102 MGTs, allowing an optical fibre to be connected to the board SPF+ connectors. The GTH MGT in the ZCU102 board was configured using a Xilinx IP to operate at a transmission rate of 10.24 Gb/s. Data decoding and encoding when interfacing the GTH are done by the lpGBT link as it is implementing its own custom protocol designed to withstand radiation conditions [6].

To fulfil these prototyping conditions, the ZCU102 settings were changed to clock the MGT at 320 MHz as required by the lpGBT transmission rates. This clock was also used to generate the clock signal that drives the slow-control block at 40 MHz since the phase of the clocks of the MGT and the slow-control block need to be related. Both AXI interfaces are also clocked at 40 MHz but are driven from a different clock source from the PS. The design of the FPGA logic was done with the Xilinx Vivado 2019.2 software and using the VHDL hardware description language.

The surrogate for the front-end side of the prototyping system was a VLDB+ development board [8] that contains an lpGBT device like the HGCAL front-end detector boards. The VLDB+ was connected to the optical fibre coming out of the back-end board.

A similar procedure was followed to add a GBT-SCA to the system. A VLDB development board [9], which includes a GBT-SCA, was connected to the VLDB+ board and assigned the EC channel of the lpGBT, completing the testing infrastructure.

B. Validation Criteria and Tests

The AXI interface between the slow-control block and the MPSoC ARM core can be validated by writing a predefined sequence of data words to system memory followed by a subsequent reading of the same addresses. The validation procedure for this specific interface consisted of an exhaustive test that checked every available address on the specified address spaces of both the AXI Full and AXI Lite interfaces of the slow-control block.

Similarly, communication with the front-end was validated by issuing a sequence of write and read operations to the lpGBT and GBT-SCA internal registers. These operations can be used to change the configuration of the ASICs such as changing the operation mode, enabling I2C masters, controlling general purpose input/output pins (GPIOs), etc. The outcome of such operations was validated by the reply data stored in the receiving buffers.

V. Results

To comprehensively validate and test the operation of the system in Fig. 4, a sequence of testing procedures targeting each individual part of the infrastructure was devised and is described in this section. To ensure the proper operation of the AXI interfaces, a Python script was run on the prototyping MPSoC ARM core that writes data words to all address positions on the memory and control module of the slow-control block and then reads back the same positions.

To test the communication with the front-end lpGBT, the values of several lpGBT internal registers were changed and read back to verify that the changes were successfully applied. Some changes concerned specific configuration registers of the lpGBT, such as changing the operation state of the lpGTB or configuring the EC channel. By monitoring an LED on the VLDB+ board that indicates whether the operation of the lpGBT is in its ready state, it was possible to confirm the
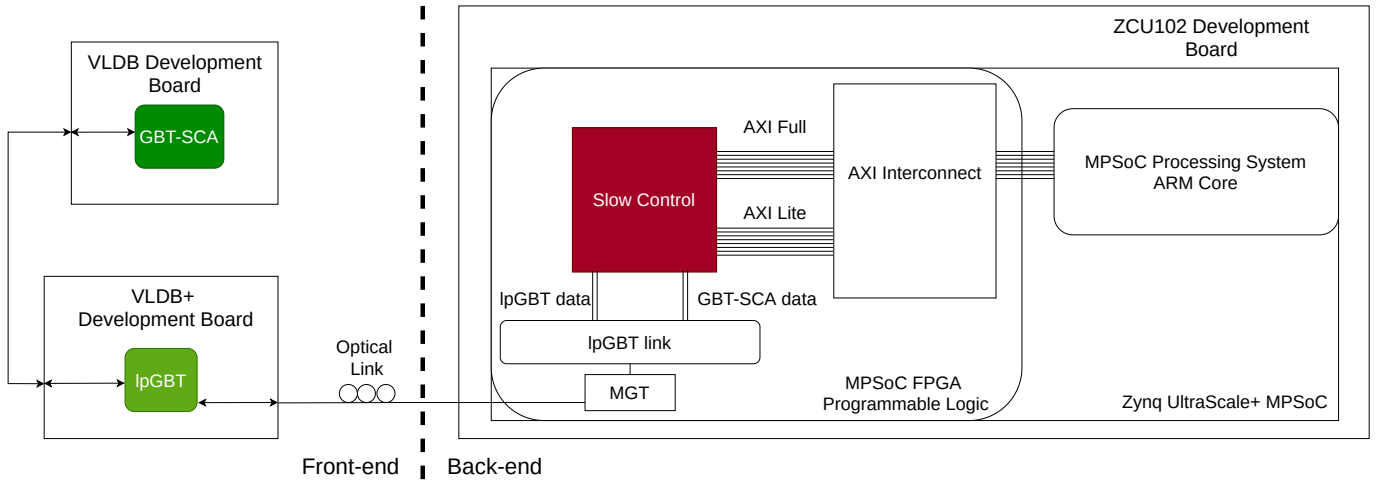
Fig. 5. Slow-control block testing infrastructure using surrogate hardware. The ZCU102, VLDB+, and VLDB boards present the same functionality and interface as the final hardware even if the latter is not yet available.

requested change in the lpGBT operation state. The correct behaviour of the configured EC channel was tested by conveying correct information to the GBT-SCA.

Similarly to the lpGBT, the GBT-SCA HDLC protocol was validated by writing and reading configuration registers of the GBT-SCA. Moreover, control of the GBT-SCA's GPIOs was validated using the connected LEDs on the VLDB board.

Although not all of the internal registers of the lpGBT and GBT-SCA were accessed in these tests, the functionality of the slow-control block was stress-tested with a specific procedure. First, the send buffers were loaded to full capacity. Next, the slow-control issued the corresponding transactions to the front-end and the MPSoC ARM core read all the front-end replies. This stress test was repeated 10000 times in a row and all ($\approx$ 10 million) transactions were successful.

Both lpGBT and GBT-SCA cores achieved a transaction rate greater than 230000 transactions per second. The observed rates can be used to roughly extrapolate the total HGCAL configuration time. In order to have a reasonably accurate estimate, other contributions have to be considered as well, including the I2C transaction time, the optical fibre latency, and software overheads. With rough estimates of these other contributions, it was possible to estimate that the full HGCAL can be configured in $\approx$1 minute.

Although this result yields a reasonable configuration time for the detector, there are still uncertainties in several contributing factors such as front-end experimental results and software bottlenecks. In particular, there are both unknowns that can increase the configuration time (e.g. when taking into account the configuration verification read-back) and opportunities for optimisation that remain unexplored (e.g. in terms of concurrent software access). Overall, these likely cancel each other out, and the order of magnitude for the time to configure the HGCAL is expected to remain on the order of minutes. In this sense, the performance of the conceived slow-control architecture is expected to allow for an acceptable

configuration time of the detector.

Once the front-end communication protocols were validated, the slow-control block is ready to be integrated into the HGCAL global prototype and other HGCAL test systems, fulfilling the purpose of the developed test system.

Table I lists the hardware resources required by the slow-control block testing system developed for the ZCU102 board, which is shown on the right side of Fig. 5. The small size of this system allows for further expansion, with a possible change in scope to test other detector functionality, like fast commands, data acquisition, etc.

TABLE I
HARDWARE RESOURCES REQUIRED BY THE SLOW-CONTROL TESTING SYSTEM WHEN IMPLEMENTED FOR THE ZCU102 DEVELOPMENT BOARD IN VIVADO 2019.2.

Resource	Used	Available	Fraction of ZCU102 (%)
LUT	7718	274080	2.82
LUTRAM	541	144000	0.38
FF	10986	548160	2.00
BRAM	16	912	1.75
DSP	2	2520	0.08
IO	2	328	0.61
GT	1	24	4.17
BUFG	8	404	1.98
MMCM	1	4	25.00

Although the final detector back-end boards are not yet available, the testing infrastructure allowed the prototyping of an accurate model of the final detector system, which validates the configuration chain between the MPSoC ARM core and the front-end.

Moreover, the employed technologies ensure complete portability of the slow-control block from the testing to the final systems. In particular, by sharing a standardized Linux framework and AXI interfaces, the software drivers that were developed to access the slow-control buffers and registers are completely portable and can be used in the final detector. Similarly, the slow-control block is compatible with the back-

end FPGA of the final system as they are both deployed on Xilinx Ultrascale+ technology. The abstraction provided by the AXI interface not only facilitates the development of the MPSoC software but also eases the implementation of further improvements and adjustments to the FPGA hardware modules.

VI. CONCLUSION

Each ATCA board on the HGCAL back-end system is responsible for configuring and monitoring up to ≈ 3500 radiation-tolerant ASICs on the front-end. This is performed through the slow-control block that interfaces up to 756 transceiver ASICs, lpGBTs and GBT-SCAs, which house I2C masters that then communicate with the remaining front-end ASICs.

Although the final ATCA board and corresponding back-end FPGA are still under development and not available for testing, the infrastructure described in this work allowed for the validation of the functionality of the whole system by modelling an accurate interface between the slow-control block, the software part running at the MPSoC, and the front-end.

Despite its reduced size and simplicity when compared to the final infrastructure, this test system was able to provide a reliable validation of the communication protocol with the front-end ASICs. This allowed the simultaneous development of the final system software and FPGA hardware, and their use in other test systems that are presently in use. The shared technology between the MPSoC of the back-end board and the final system back-end FPGA also ensures full portability between the platforms used by the test and final detector system.

VII. FUTURE WORK

The presented prototyping and validation of the communication infrastructure between the slow-control block at the back-end and both the lpGBT and GBT-SCA at the front-end allows not only the integration of the slow-control block in other HGCAL test systems but also the expansion of the current test system by expanding its scope vertically (by adding additional functionality and features of either the front-end or the back-end domain) and horizontally (by adding already existing components to increase the size of the prototype). In both cases, the hardware can be upgraded to use the final detector components, either the back-end FPGA hardware or the front-end ASICs. Both can be done in parallel, using different test systems.

The front-end expansion will involve the establishment of more communication channels with the back-end by adding more ASICs. It will also involve the replacement of the VLDB+ and VLDB boards with custom HGCAL hardware parts, targeted for the final detector. This process should be straightforward for the slow-control, since the communication is already validated with both lpGBTs and GBT-SCAs, only requiring that more channels be instantiated. Moreover, any further vertical integration of I2C targets on the test systems

does not require the slow-control block to change because the interface with such ASICs is done via the lpGBTs or GBT-SCAs.

In the back-end domain, the FPGA hardware has to be updated with more back-end functionalities and, upon availability, the ZCU102 board will be replaced by the final ATCA detector board, based on a VU13P FPGA. These back-end migrations are also straightforward from the perspective of the slow-control block due to its modular development and shared technology with the VU13P FPGA. The involved software also does not need any significant changes since the Linux platform is the same, as well as the AXI connection from the FPGA fabric to the ARM core in the MPSoC.

REFERENCES

[1] CMS Collaboration, "The Phase-2 Upgrade of the CMS Endcap Calorimeter," CERN, Geneva, Tech. Rep., Nov 2017. [Online]. Available: https://cds.cern.ch/record/2293646

[2] S. Mallios, P. Dauncey, A. David, and P. Vichoudis, "Firmware architecture of the back end DAQ system for the CMS high granularity endcap calorimeter detector," *Journal of Instrumentation*, vol. 17, no. 04, p. C04007, apr 2022. [Online]. Available: https://doi.org/10.1088/1748-0221/17/04/c04007

[3] *ZCU102 Evaluation Board User Guide*, Xilinx. [Online]. Available: https://docs.xilinx.com/v/u/en-US/ug1182-zcu102-eval-bd

[4] ARM, "AMBA® AXI™ and ACE™ Protocol Specification," AXI4 protocol specifications from ARM. [Online]. Available: https://developer.arm.com/documentation/ihi0022/e/AMBA-AXI3-and-AXI4-Protocol-Specification

[5] *AXI Chip2Chip v5.0 LogiCORE IP Product Guide*, Xilinx. [Online]. Available: https://docs.xilinx.com/r/en-US/pg067-axi-chip2chip/AXI-Chip2Chip-v5.0-LogiCORE-IP-Product-Guide

[6] P. Moreira, S. Baron, S. Biereigel, J. Carvalho, B. Faes, M. Firlej, T. Fiutowski, J. Fonseca, R. Francisco, D. Gong, N. Guettouche, P. Gui, D. Guo, D. Hernandez, M. Idzik, I. Kremastiotis, T. Kugathasan, S. Kulis, P. Leitao, P. Leroux, E. Mendes, J. Mendez, J. Moron, N. Paulino, D. Porret, J. Prinzie, A. Pulli, Q. Sun, K. Swientek, K. Wyllie, D. Yang, J. Ye, T. Zhang, and W. Zhou, *lpGBT documentation: release*, Mar 2022. [Online]. Available: https://cds.cern.ch/record/2809058

[7] A. Caratelli, S. Bonacini, K. Kloukinas, A. Marchioro, P. Moreira, R. De Oliveira, and C. Paillard, "The GBT-SCA, a radiation tolerant ASIC for detector control and monitoring applications in HEP experiments," *JINST*, vol. 10, p. C03034, 2015. [Online]. Available: https://cds.cern.ch/record/2158969

[8] D. Montesinos, S. Baron, N. Guettouche, and J. Mendez, "The versatile link+ demo board (VLDB+)," *Journal of Instrumentation*, vol. 17, no. 03, p. C03032, mar 2022. [Online]. Available: https://doi.org/10.1088/1748-0221/17/03/c03032

[9] R. Martin Lesma, F. Alessio, J. Barbosa, S. Baron, C. Caplan, P. Leitao, C. Pecoraro, D. Porret, and K. Wyllie, "The Versatile Link Demo Board (VLDB)," *JINST*, vol. 12, p. C02020. 12 p, 2017. [Online]. Available: https://cds.cern.ch/record/2275133