Contribution ID: **68** Type: **Poster**

The New Version of the LHCb SOL40_SCA Core to Drive Front-End GBT-SCAs for the LHCb Upgrade

Wednesday, 13 September 2017 17:45 (15 minutes)

The LHCb experiment is currently engaged in an upgrade effort that will implement a trigger-less 40 MHz readout system. The upgraded Front-End Electronics profits from the GBT chipset functionalities and bidirectional optical fibers for readout, control and synchronization. This paper describes the new and final version of the firmware core that transmits slow control information from the Control System to thousands of Front-End chips, discussing the implementation that expedites and makes the operation more versatile. The detailed architecture, original interaction with the software control system and integration within the LHCb upgraded architecture are described. First tests in FPGAs are shown.

Summary

The LHCb experiment is one of four experiments operating at the Large Hadron Collider and it consists of a high precision detector that focuses in the detection of decays of particles containing B mesons.
LHCb is currently involved in an Upgrade effort that will happen between 2019 and 2020. During this upgrade, most of the on-detector and back-end electronics will be replaced. In order to relay the event information from all bunch crossings to the Software Trigger, the readout electronics will have to cope with the full 40MHz bunch crossing rate without performing first-level event triggering. In view of this requirement, the experiment is using the GBT chipset and its related components developed by the EP-ESE group at CERN, for data readout as well as for fast control and slow control tasks. In particular, to aid the slow control tasks, the GBT-SCA ASIC is specifically used to configure and monitor the Front-End electronics (FE) via a set of protocols such as JTAG, I2C and SPI, among others. In LHCb, these are controlled via FPGA-based electronics boards -commonly referred to as SOL40 –interfaced to the software control system via a PCIe bus. It is estimated that the LHCb experiment will contain around 2500 Master GBTs and GBT-SCAs and around 10000 FE chips. To be able to drive the numerous GBT-SCA chips in the FE, a specific firmware core was initially developed. It consists of a technology agnostic firmware VHDL core, developed in a generic way to support all buses and protocols of the GBT-SCA chips. Such core is driven by a bus such as the Altera Avalon bus and drives the GBT-SCA chip through optical links via GBTs. However, the first implementation employed too many FPGA logical resources and it lacked the necessary speed to configure and monitor the FE electronics in a fast way.
This document discusses the improvements and solutions in the firmware core so that it can handle the configuration and constant monitoring of the FE chips during the operation of the LHCb experiment. These improvements aim at increasing the amount of GBT-SCAs that can be reached by each core instantiation in order to save resource utilisation. They also aim at shortening the execution time of the most taxing control operations through the use of protocol accelerators and memories, reducing the interaction between software and hardware. The most immediate use cases are the mass configuration of GBTx chips dedicated to the transmission of data and the remote configuration of FPGAs located at the FE. At LHCb, this design was implemented for both an Arria X FPGA that is mounted on the LHCb PCIe40 board, and a Stratix V FPGA that is mounted on the LHCb MiniDAQ prototype board. The architecture of this core will be discussed in detail, as well as its integration in the global readout control architecture and its interaction with the software control system. Tests using real Front-End hardware were performed for all GBT-SCA protocols, including remote

FPGA configuration of Kintex7 and IGLOO2 FPGAs.

Primary author: VIANA BARBOSA, Joao Vitor (CERN)

Co-authors: ALESSIO, Federico (CERN); GASPAR, Clara (CERN)

Presenter: VIANA BARBOSA, Joao Vitor (CERN)

Session Classification: POSTER Session

Track Classification: Programmable Logic, Design Tools and Methods