The ATLAS IBL BOC Demonstrator

L. Ancu, D. Falchieri, J. Dopke, T. Flick, T. Heim, A. Kugel, M. Neumann, A. Gabrielli, J. Grosse-Knetter, J. Joseph, N. Krieger, A. Polini, P. Morettini, B. Schneider, N. Schroer

Abstract—The Insertable-B-Layer (IBL) is a new pixel detector layer to be installed at the ATLAS experiment at the LHC, CERN in 2013. It will be integrated into the general pixel readout and software framework, hence the off-detector readout electronics has to support the new front-end electronics whilst maintaining a high degree of interoperability to the components of the existing system. The off-detector readout is realised using a number of VME card pairs ROD and BOC plus a VME crate controller and a custom timing distribution system. The main elements of the new BOC design comprise optical interfaces towards the detector, signal conditioning and data recovery logic. We present the demonstrator used to verify the design approach. The demonstrator is based on a XILINX SP605 FPGA evaluation board and uses a Microblaze processor inside the FPGA to provide easy and flexible access to all essential BOC functions and the corresponding emulator modules, which enable full test of the entire BOC functionality even without any external components. However, optical interfaces may be connected via a mezzanine card. We present the details of the emulation circuitries together with measurement results showing the operation of the BOC logic.

Index Terms—Detectors, Field Programmable Gate Array, Data acquisition.

I. INTRODUCTION

THE IBL is a new pixel detector layer to be installed at the ATLAS experiment at the LHC, CERN in 2013. It will be integrated into the general pixel readout and software framework, hence the off-detector readout electronics has to support the new front-end electronics whilst maintaining a high degree of interoperability to the components of the existing system. The off-detector readout is realised using a number of VME card pairs ROD and BOC plus a VME crate controller and a custom timing distribution system. The BOC-firmware is being developed and evaluated using a demonstrator, based upon a commercial FPGA development board, in order to be ready when the final BOC hardware will be available. This paper presents the details of the emulation circuitries together with measurement results showing the operation of the BOC logic.

The important aspects of the IBL readout architecture are explained in chapter II. Demonstrator project organisation and

Fig. 1. FE-I4 Schematics

functionality are given in chapter III. Test software features are listed in chapter IV. The firmware building blocks are presented in chapter V. Chapter VI summarizes the results.

II. READOUT ARCHITECTURE

The IBL will become a part of the existing ATLAS pixel detector after installation and must be compatible with the existing infrastructure as much as possible. The on-detector electronics will be exposed to a significantly increased signal rate (or occupancy) compared to the present system. This rate is handled by a new readout ASIC called FE-I4, which provides a 4-times higher transfer speed of 160Mbit/s, simplified and faster command decoding and an active area of 19*20mm serving 26k pixels. Fig. 1 ([1]) shows the schematics of the FE-I4 ASIC.

The FE-I4 is connected to a nearby optical interface arrangement which in turn connects to the off-detector readout electronics via 70m long fibers. The off-detector electronics of the present pixel detector is based on card-pairs - Read-Out-Driver (ROD) and Back-of-Crate-Card (BOC) - housed in VME crates. The BOC interfaces to the ATLAS timing system, to the detector control system (DCS), to the main ATLAS DAQ system and to the optical interfaces towards the detector. All low level timing functions, signal coding/decoding and opto-electrical conversion are done on this card. The ROD is dealing

Manuscript received November 18, 2011. This work was supported in part by the German Federal Ministry of Education and Research under grants 05H09VHA, 05H09PX1 and 05H09MGA.

L. Ancu and B. Schneider are with Bern University, Switzerland.

A. Gabrielli is with Bologna University, Italy.

D. Falchieri and A. Polini are with INFN Bologna, Italy.

J. Dopke is with CERN, Switzerland.

T. Flick, T. Heim and M. Neumann are with Wuppertal University, Germany.

J. Grosse-Knetter and N. Krieger are with Göttingen University, Germany.

J. Joseph is with LBNL, USA.

P. Morettini is with Genova University, Italy.

A. Kugel and N. Schroer are with Heidelberg University, Germany (e-mail: akugel@ieee.org).

Fig. 2. ROB-BOC Card Pairs

with all kinds of data formatting and processing for both data taking and calibration use cases. Besides it's interconnect to the BOC it connects to a system controller via VME for configuration, control and monitoring. This approach (see fig. 2) is also the baseline for IBL, in order to minimize the impact on the system infrastructure and to provide interoperability with the corresponding cards of the present system.

The new implementations take advantage from the latest FPGA technology and integrate all logic, including timing adjustment, into a small number of FPGAs. Apart from the savings in PCB area this is a prerequisite for the mentioned interoperability. Input and output bandwidth of the IBL ROD-BOC system is 4-times higher compared to the pixel ROD-BOC pair. A total of 32 FE-I4 devices can be attached ($32 * 160\text{Mbit/s}$) and 4 standard "HOLA S-Link" [3] readout-links (ROLs) in dual-output configuration drive the data to the main ATLAS readout system (ROS) and to the new fast track trigger (FTK). The task sharing between ROD and BOC is sketched in fig. 3 and works as follows:

- The VME interface allows the crate controller processor to steer the operation of the ROD and - via the setup-bus - of the BOC.
- Upon a trigger or a VME command the ROD prepares the corresponding serial data stream and sends it to the BOC which in turn encodes the bit stream into one (or multiple) biphasic-mark (BPM) encoded signal(s). Signal timing conditioning via mark-space-ratio (MSR) adjustment are performed here as well, followed by optical output.
- Data received by the optical interface are run through a clock/data recovery logic and are subsequently decoded according to the 8bit-10bit (8B10B) protocol. Idle characters are suppressed and all remaining characters are forwarded to the ROD.
- The ROD formats the data, builds the event fragments and returns them to the BOC, which in turn drives the S-Link outputs.

III. IBL BOC

The main elements of the new BOC design comprise optical interfaces towards the detector, signal conditioning and data recovery logic. While output signal conditioning for the commands towards the FE-I4 will most likely not be required, fine-tuning of delays and duty cycle is needed when the BOC has to interface to the FE-I3 based pixel detector.

Fig. 3. ROB-BOC Task Distribution

Fig. 4. BOC Diagram and PCB

A. Target BOC

There is no final decision yet on the type of optical interfaces on the BOC side. The first prototypes of the target BOC use "SNAP12" ([2]) parallel optics transmitters and receivers with 12 fibers as a baseline. These components are specified for datarates above 2Gbit/s but have been measured to work reliably down to 300kHz. Measurements done so far indicate they will be a viable solution for use on the receiving side. However, the transmitters might need to deliver more optical power than the SNAP12 can provide. The prototype BOC therefore provides space for an "old-style" optical transmitter module as well as for test connectors to attach different optical mezzanies for evaluation in view of the final choice. If SNAP12 proves to be compatible with the optical power requirements, no redesign will be necessary.

For the S-Link output transceivers two alternative implementations are realized, one using 4 individual SFP-style devices housed in 4-ganged case, the other one using a single quad-SFP (QSFP) device, consuming only about one-third of the area of the former variant.

Fig. 4 shows the block diagram and the PCB of the new BOC. The I/O is distributed across two Xilinx Spartan-6LX150 FPGAs which are connected via a high-speed serial interface in order to provide some local routing capabilities. These two "datapath" FPGAs are controlled by a third FPGA of the same family (Xilinx Spartan-6LX75). This FPGA connects to the so-called "setup-bus" interface of the ROD and provides an extension to the ROD register area. The setup-bus is a simple 8 bit asynchronous bus and is used to configure and monitor the operation of the BOC as the BOC does not have an interface to VME by itself.

Fig. 5. Functional Blocks

B. Demonstrator

In order to develop and evaluate the various firmware blocks comprising the BOC functionality before the BOC prototype hardware is available the BOC demonstrator project was set-up with the goal to provide a portable hardware and software environment for use by the different groups involved. To facilitate this, a demonstrator manual was prepared, an SVN repository for firmware and test software has been set up as well as a Wiki to document the activities and the results.

The demonstrator is based on a XILINX SP605 FPGA evaluation board and uses a Microblaze processor (MB) inside the FPGA to provide easy and flexible access to all essential BOC functions and the corresponding emulator modules, which enable full tests of the entire BOC functionality even without any external components. However, optical interfaces may be connected via a mezzanine card.

Without VME and setup-bus the MB emulates the crate controller and the ROD-BOC interconnect. All logic related to the setup-bus is accessed via a setup-bus emulation, similar to the target design. The logic related to generic demonstrator functionality is accessed through separate memory interfaces in order to have a clean separation of the functionalities in the software layer.

Figure 5 provides an overview of the functional blocks of the demonstrator, which are divided into four groups:

- Generic BOC blocks (red): Timebase, setup-bus, dual S-Link, BPM, MSR and FE-I4 input
- Cpu main blocks (grey): MB processor with peripheral areas, GPIO status and control, clock checking
- Cpu auxiliary blocks (blue): user interface via RS232, LEDs and switches
- Cpu FIFOs (yellow): FIFOs to capture data sent towards the (fake) ROD and data to be sent towards the FE-I4 command output (running at 40MHz) or to be looped back to the FE-I4 input (running at 160MHz). In the latter case, frequency variations in the order of 1% can be applied to test the robustness of the clock-data-recovery logic.

The latter three groups are on the first view specific only to the demonstrator and not relevant for the target application but will most likely become part of the self-test suite in the final system.

The full set of functions is implemented on a Xilinx Spartan-6 development card (SP605, see fig 6). Electrical and optical interfaces are available via a custom I/O expansion (fig. 7), for

Fig. 6. Xilinx SP605 card

Fig. 7. I/O expansion card for SP605

example to attach an FE-I4 evaluation module, an oscilloscope, a ROD emulator card or optical fibers via SNAP12 and SFP modules. A loopback option is implemented allowing to feed the outbound signals (after the MSR stage) back to the FE-I4 input block, whilst maintaining the signal drivers. This enables to emulate FE-I4 data patterns from within the demonstrator, apply arbitrary signal conditioning and observe the resulting behaviour of the CDR logic.

The firmware is parameterized for channel number and S-Link availability and can therefore be ported to different platforms even with lower pin count and without high-speed serializers.

IV. TEST SOFTWARE

The demonstrator software is standard "C"-code based upon the Xilinx embedded development kit (EDK) software framework. The EDK allows convenient simulations of VHDL firmware blocks together with software executables. The firmware parameters (e.g. number of channels, simulation mode) are passed to the software layer via a "C"-header file, generated by a VHDL module. This way, the program can adapt to the actual features of the hardware and a single set of source files code can be used for all options.

V. FIRMWARE BUILDING BLOCKS

A. Receive Path

The receive path consists of three stages:

- Bit synchronization
- 8B10B decoder and de-serializer
- ROD multiplexer

Input data are generated by the FE-I4 devices which operate at a nominal speed of 160Mbit/s, derived from the common

Fig. 8. Xilinx I/O serializer/de-serializer

Fig. 9. 4 phase sampling with ISERDES

ATLAS timebase of 40MHz. Therefore, re-synchronization to a local 160MHz clock, derived from the same timebase, should be straightforward using a simple phase adjustment circuitry. Nevertheless a more robust approach is desirable for the BOC, which is implemented by a CDR logic taking advantage from the Xilinx input de-serializer feature (ISERDES). The ISERDES block (fig. 8) is clocked at 4-times the nominal bit rate, hence 640MHz, and provides 4 bits of data with every cycle of the nominal clock. Therefore, the 4 bits represent 4 samples of the input bit with phase offsets of 0, 90, 180 and 270 degrees. There are two main advantages of using the ISERDES for this sampling mechanism. First, the registers inside the IO blocks are more robust against metastability than the registers of the regular FPGA fabric. Second, there is only one additional clock needed, compared to 3 additional clocks for an internal 4-phase sampling scheme.

The 4 phases are processed by a state machine which detects rising and/or falling edges of the signal, selects the appropriate phase as output value (typically offset by 2 steps) and adjusts the sampling position when the edge position changes due to a difference in frequencies. At certain state transitions either 0 or 2 data bits are generated, compensating lower or higher speed of the data source respectively.

Fig. 9 shows the sampling at the 4 positions and the selection of a particular output position.

The 8B10B-decoder assembles the recovered data into 10-bit words and searches for the word synchronization pattern (5 '1'-bits) present only in special control characters in order to properly align the decoder input. The 10bit words are subsequently decoded into 8bit characters plus a control flag, indicating idle, start-of-packet and end-of-packet conditions.

The decoded data, apart from idle characters, are fed into a small FIFO operating at 160MHz. For transmission to the

Figure 18: SNAP-12 TX (front), RX (rear)

Fig. 10. SNAP12 transmitter and receiver

Fig. 11. SNAP12 optical loopback at 300kHz

ROD, 4 FIFOs are combined into a ROD data channel, by multiplexing the FIFO output in a round-robin fashion. The multiplexed readout operates at 80MHz. Due to the 8B10B operation the maximum input rate is limited to $4 * 16\text{MHz}$, which allows for a sufficient margin.

B. SNAP12 Optics

The SNAP12-technology is an attractive solution for parallel optics application and matches quite well the granularity of the IBL readout, where the S-Link output bandwidth demands a grouping factor of 8 FE-I4 devices. However, the input data rate of 160Mbit/s is low compared to state-of-the-art optical interfaces, which operate in the multi-gigabit regime.

An important task of the BOC demonstrator project was to assess the suitability of the SNAP12 devices for the IBL readout.

SNAP12 devices (fig. 10) from two different vendors were tested in an optical loopback configuration and all devices are capable to operate down to a frequency of approximately 300kHz, see fig. 11.

C. S-Link

The HOLA S-Links are the standard interconnect links in ATLAS between the RODs of the detector readout and the ROS of the data acquisition system. HOLA uses a 2Gbit/s bi-directional optical transmission. An FPGA-based protocol engine provides the interface between user logic and the media interface and comes in two flavours, a link source implementation (LSC) and a link destination implementation (LDC). At the ROS end an embedded LDC implementation is

Figure 10: Dual-HOLA

Fig. 12. Dual S-Link mechanism

Figure 16: Mark-space-ratio adjustment

Fig. 13. Mark-space-ratio adjustment circuitry

Figure 18: MSR delay characteristics: AND, OR, Fixed

Fig. 14. MSR resolution

used, while all RODs to date use mezzanine LSCs. To achieve a high density implementation the IBL BOC will use an embedded LSC implementation with support for the new dual-output (see fig. 12) required to attach the FTK. The SP605-based demonstrator provides one optical interface directly, the second one is available via the expansion mezzanine. The firmware for the dual implementation has been implemented and simulated successfully but has not been tested yet in a real setup with an LDC.

D. MSR Adjustment

The circuitry used to adjust the mark-space-ratio for the existing pixel system is shown in fig. 13. It consists of an AND-gate with one delayed input followed by an OR-gate with one delayed input. The AND logic decreases the duration of the high level of the input signal, while the OR logic increases the duration. Tuning of the delay in either AND or OR path adjusts the mark-space-ratio. A subsequent delay is used to bring the conditioned signal (after the OR gate) to a well-defined position with respect to the master clock. The corresponding signal timing is shown in fig. 15.

The required precision of the tuning is in the order of 1ns, the ratio adjustment range in the order of a few ns and the delay must be able to span a clock period of 25 ns. The delays are now realized using special multiplexer elements (MUX) within the FPGA with a known delay, depending on the FPGA family. For Spartan-6, the delay is 300ps, hence less than 100 elements are needed for a full period. While the MUXes provide a quite constant delay, the internal routing introduces some additional, variable delays which generates a slightly non-linear characteristics, as shown in fig. 14. The linearity can be improved somewhat using directed placement of the

Fig. 15. MSR function

MUXes, however the precision of the unguided placement is still acceptable.

VI. CONCLUSION

Within the BOC demonstrator project all building blocks of the final BOC, including the legacy timing adjustment features, have been prototyped, simulated and in part already tested in hardware. Due to the complexity of the project, particular blocks have been tested in selective implementations at some institutes, but not with the full demonstrator code. However, valid results have been obtained here as well.

The primary functions - command output, signal conditioning and data input have been tested successfully.

Initial tests of the optical transmission using SNAP12 parallel optics provided promising results, although final tests in particular of the optical output in conjunction with the corresponding on-detector optics are still to be done.

Likewise, the S-Link transmission has to be tested.

The Microblaze based test software allows convenient co-simulation of firmware and software and can be re-used partially in the future self-test code of the BOC.

REFERENCES

- [1] The ATLAS collaboration, *Submission of the First Full Scale Prototype Chip for Upgraded ATLAS Pixel Detector at LHC, FE-I4A*, ATL-UPGRADE-PROC-2010-006, <https://cdsweb.cern.ch/record/1301523/files/ATL-UPGRADE-PROC-2010-006.pdf>, CERN, Oct 2010
- [2] www.reflexphotonics.com
- [3] hsi.web.cern.ch/hsi/s-link/devices/hola