PIXEL 2014 INTERNATIONAL WORKSHOP
SEPTEMBER 1–5, 2014
NIAGARA FALLS, CANADA

Firmware development and testing of the ATLAS IBL Back Of Crate card

M.E. Stramaglia on behalf of the ATLAS collaboration

AEC-LHEP University of Bern,
Sidlerstrasse 5, 3102 Bern, Switzerland

E-mail: maria.elena.stramaglia@cern.ch

ABSTRACT: ATLAS is one of the four big LHC experiments and recently its Pixel Detector was upgraded with a new innermost 4th layer: the Insertable B-Layer (IBL). The upgrade will result in better tracking efficiency, improved precision of measurements and, in the future, compensation for radiation damage of the Pixel-Detector. Newly developed front-end electronics and the higher than originally planned LHC luminosity required a complete re-design of the Off Detector Electronics consisting of the Back Of Crate card (BOC) and the Read Out Driver (ROD). The main purposes of the BOC card are the distribution of the LHC clock to all Pixel Detector components as well as interfacing the detector and the higher level readout optically. The data-path to the detector runs a 40 MHz bi phase mark (BPM) encoded stream. The 160 MHz 8b10b encoded data path from the detector is phase and word aligned in the firmware and then forwarded to the ROD after decoding. The ROD will send out the processed data that is then forwarded to the higher level readout by the BOC card. An overview of the newly developed firmware will be presented together with the results from production tests and the system test at CERN. Focus is put on the partial reconfiguration and results of the fine delay measurements.

KEYWORDS: Electronic detector readout concepts (solid-state); Solid state detectors; Particle tracking detectors (Solid-state detectors)

2015 JINST 10 C02035

Contents

1 The Insertable B-Layer

ATLAS is one of the four big LHC experiments [1, 2] and recently its Pixel Detector [3] was upgraded with a new innermost 4th layer: the Insertable B-Layer (IBL).

The Insertable B-Layer (IBL) is the new and innermost layer of the ATLAS Pixel detector. Shown in figure 1 is the insertion in the detector in May 2014 and it is currently being commissioned. The IBL is a silicon pixel detector made from two types of sensors: planar and 3D sensors. It consists of 14 staves, each of them holding 32 Front-End read-out chips (FE-I4) [4] bump-bonded to the sensor tiles: eight 3D sensors and 12 planar sensors on each stave.

Every FE serves 26880 pixels. For the read-out convention, every FE pair is commonly referred to as a "module" (or more precisely DAQ module). Each planar sensor serves two FE chips, while 3D chips serve one FE each; this means that one planar sensor (plus its FE chips) corresponds to one module while two 3D sensors are needed to form one module. More details regarding the detector and the sensor technologies can be found in [5, 6].

2 Read out chain

The read out scheme is shown in figure 2(a) [7]. The modules are electrically connected to the "optoboards" (2 boards per stave), which handle the electrical-optical conversion. The data transmission runs through optical fibers to serve several aspects: mass reduction, loss reduction, crosstalk elimination, and decoupling on from off-detector systems. The signals from each optoboard run to a card pair consisting of the Back Of Crate (BOC) [8] card and the Read Out Driver (ROD) card [9], both hosted in a VME crate. Card pairs are dedicated to IBL (14) and one is dedicated to the Diamond Beam Monitor (DBM) [7].

The ROD handles the detector calibration and data taking. The BOC serves as the interface with the detector and the higher level Read-Out Subsystem (ROS) and distributes the LHC clock from the Timing-Trigger Control Interface Module (TIM) to the ROD and to the detector.

An Ethernet connection allows remote access to both the cards from external computers for many purposes: e.g. tests, stand-alone operation and firmware upgrades.
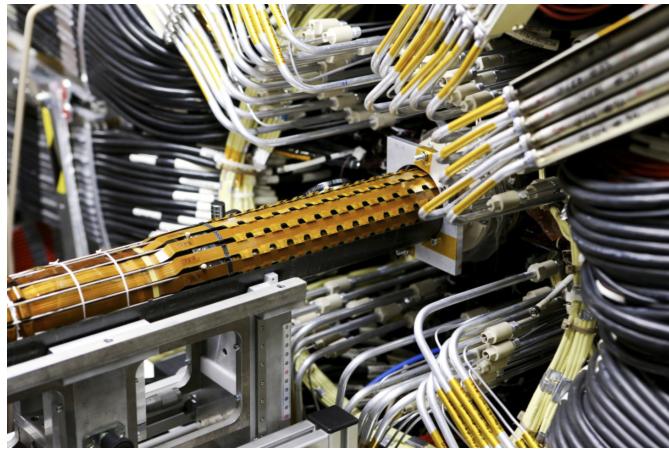
Figure 1. IBL during the insertion into the ATLAS Pixel Detector. Half of the staves are visible, as well as the services in the outermost part. A mechanical support is under the detector.
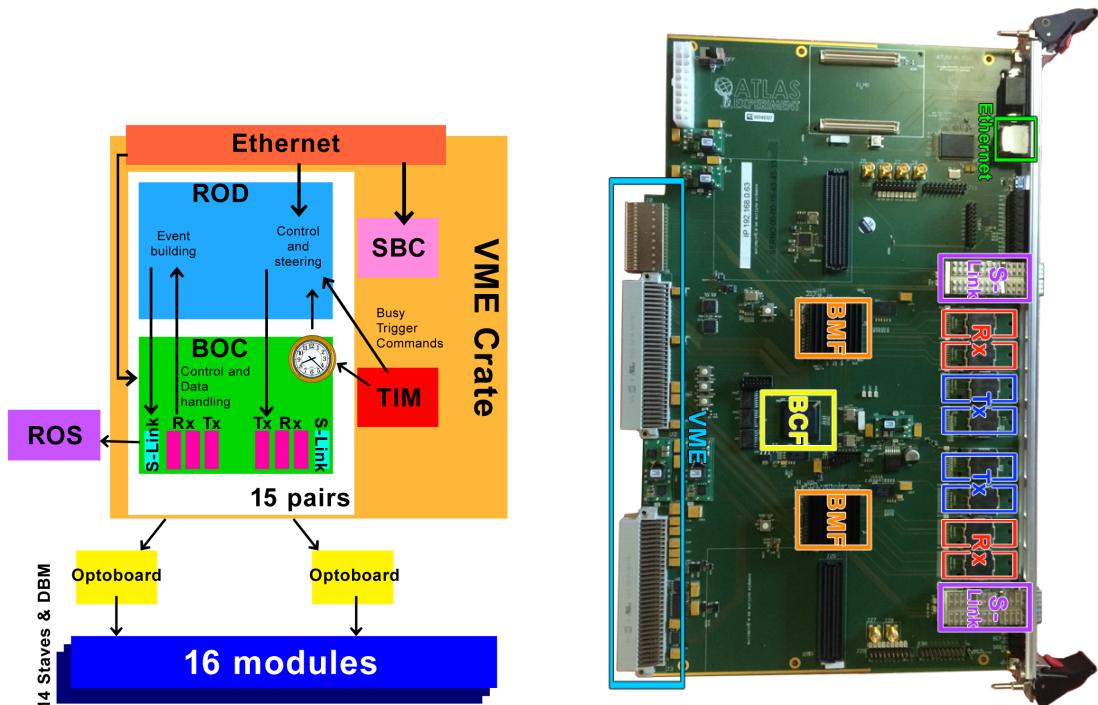
(a) Read-out chain scheme: one VME crate containing 15 card pairs (one per stave plus one dedicated to the Diamond Beam Monitor); external ethernet connections. The VME crate contains as well a TIM card (see text) and a Single Board Computer (SBC). The Read-Out Subsystem (ROS) is instead outside the crate.

(b) Back Of Crate card. 3 FPGAs: BCF (yellow) and BMFs (orange); 4 AVAGO SNAP12 receivers (red) and 4 transmitters (blue); VME connection (light blue); S-Link (violet); Ethernet port (green).

Figure 2. Read-out.
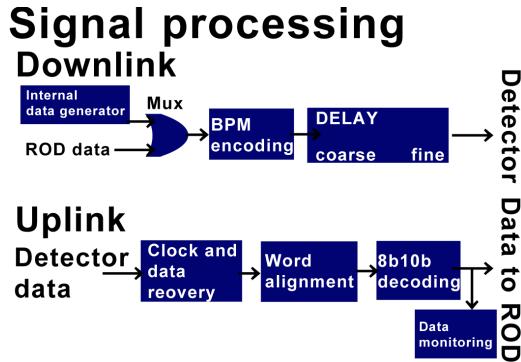
Figure 3. Signal processing from the ROD/ethernet port to the optoboards (Downlink) and from the opto-boards to the ROD (Uplink).

The new IBL BOC card (shown in figure 2(b)) is equipped with three Xilinx Spartan 6 FP-GAs [10] for signal processing:

- One BOC Control FPGA (BCF) for slow control interfaces from the ROD and the Gigabit Ethernet connection.

- Two BOC Main FPGAs (BMF) dedicated to the signal processing and monitoring of incoming data.

The BOC-ROD communication is made of high-speed signals and a relatively slow setup-bus. The detector interface is handled by commercial SNAP12 plugins [11] that work as optical/electrical converters and vice-versa. FPGA settings control the transmission rates (Tx at 40–80–160 Mbps) for testing purposes while the reception is kept constant (Rx – 160 Mbps).

Each module on IBL staves needs one Tx channel and provides two Rx lines, in a future application, two transmitters per module will be needed. The plugins have 8 channels enabled each, therefore the Tx channels needed for IBL are half the Rx channels needed.

The connection to the higher level read-out uses the S-Link Protocol, QSFP transceiver modules provide four links dedicated to the ATLAS Fast Tracker (FTK) [12] and four to the ATLAS ROS.

3 Signal processing and tests

The signal processing can be divided into two different paths: downlink from the read-out chain to the detector, and uplink from the detector to the ROD. Commands and triggers to the modules are sent from the ROD through the downlink path as shown in figure 3 (the BOC features have blue background). In case the ROD is not available, commands can also be generated by an external source and sent directly to the Tx FIFO bus through the ethernet port.

The commands are BPM encoded sending data and clock information on the same line. The optoboard then splits the clock and the data signal into two parallel lines before forwarding them to the modules.

(a) Direct Loopback.   **(b)** Optoboard Loopback.
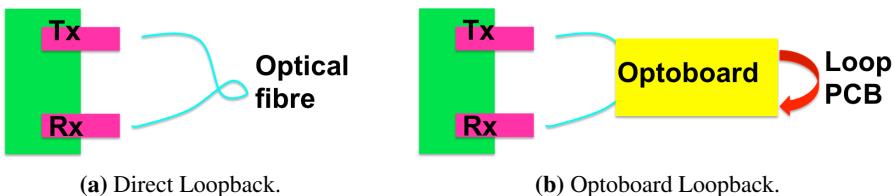
Figure 4. Loopback tests scheme. (a) Loopback connection for testing the BOC transmission, reception and decodification of the data. (b) Loopback connection to test the BOC-optoboard communication.

In order to synchronize the phases of the BOC and ROD clocks with the ATLAS Timing, Trigger and Control System (TTC), the detector timing is adjusted using coarse and fine delay blocks.

The incoming data is phase aligned for a correct decoding (lower part in figure 3). The search for word boundaries in the 8b10b encoding looks for comma words [13]. The aligned data is decoded and forwarded to the ROD. Data monitoring functions, including monitoring of decoding and disparity errors, as well as frame errors, are available to measure the signal quality. Also the module health status is monitored and forwarded to the detector control system (DCS).

To assure functionality of the final system, the transmission was checked at each connection step checking BPM encoding and decoding. Two particular tests were dedicated to the data trasmission on the BOC side. One test verified that the data sent from the Tx could reach the Rx side without any corruption after passing through a fiber connection, the other test included the optoboard in the loop. The direct loopback was performed between Tx and Rx on the same bridge (half of the BOC under the control of one BMF) as represented in figure 4(a), the loopback in ATLAS including the optoboard is instead in figure 4(b).

In the first case, figure 4(a), 160 Mbps and only 8b10b encoded data (Rx request) without BPM encoding (no optoboard included) were sent. Data were randomly generated by a software code and sent through the ethernet to the Tx bus. Then they were forwarded through optical fiber to the Rx and read back on the Rx bus for comparison. The second loopback test, figure 4(b), was done following the previous scheme but including the optoboard into the loop and therefore still 8b10b encoded data but at a 40 Mbps. A dedicated firmware allowed the Rx line on the BOC to receive data at a lower frequency. A loopback PCB connected the output to the input line on the optoboard. For this second test the information was BPM encoded and only the data line was then forwarded from the optoboard to the Rx channels.

In both cases all data sent were correctly received and decoded. For the first loop, it is possible to estimate a limit for the bit error rate of the order of 10^{-13}; for the second one, the forseen bit error rate limit is of the order of 10^{-11} due to a faster test. The required limit was 10^{-13} for the BOC per link.

4 Fine delay implementation and FE emulator

To allow for phase adjustment of each detector module with respect to the LHC, the BOC allows delaying clock and commands. The coarse delay gives a step of 6.25 ns per setting. A fine delay is then needed for adjustment of the timing of the detector, especially for low Time Over Threshold
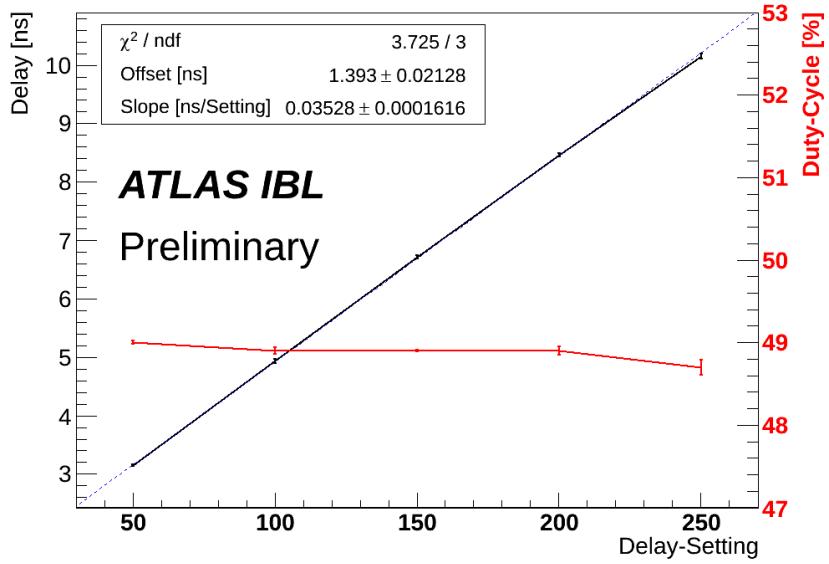
(a) Single channel measurement of delay (left Y-axis) and duty cycle (right-red Y-axis) depending on the delay-setting [15]. The result shown is the raw one from the fitter.
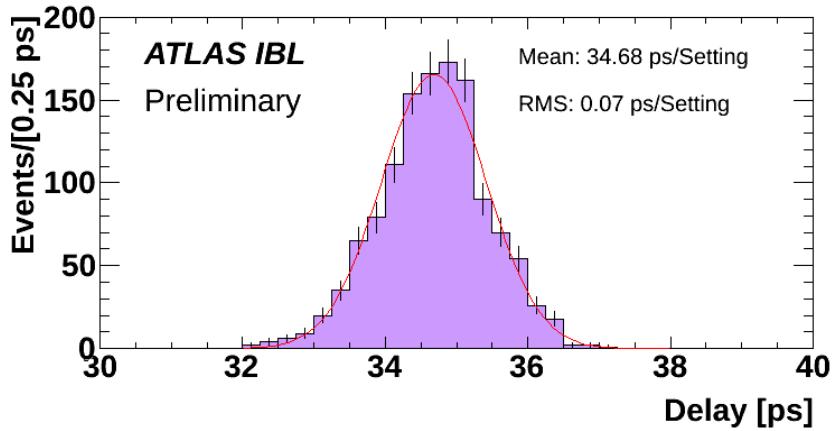
(b) Distribution of the delay for a large subset of tested channels [15].

Figure 5. Delay measurements results.

(TOT) hits. The required maximum step was fixed to 100 ps, while the optoboard requires an ideal duty-cycle of the signal that should not exceed $(50 \pm 2)\%$. A variable output delay is obtained changing the 8 bit wide fine delay-setting (IODELAY2) by using partial reconfiguration as shown in figure 5(a) [8, 14, 15].

An Internal Configuration Access Port (ICAP) allows access to the internal configuration memory and to write the configuration bits which affect the delay setting in the IODELAY2-primitive. The mapping of the configuration bits is part of the Intellectual Property so a netlist containing the FPGA configuration had to be prepared from which the bitfile is generated.

The configuration inside the netlist can be changed through the "FPGA-Editor" tool provided by Xilinx [16]. An appropriate differential bitfile is generated by changing the delay value of the output delay. This bitfile is loaded through the ICAP interface into the FPGA configuration memory

– 5 –
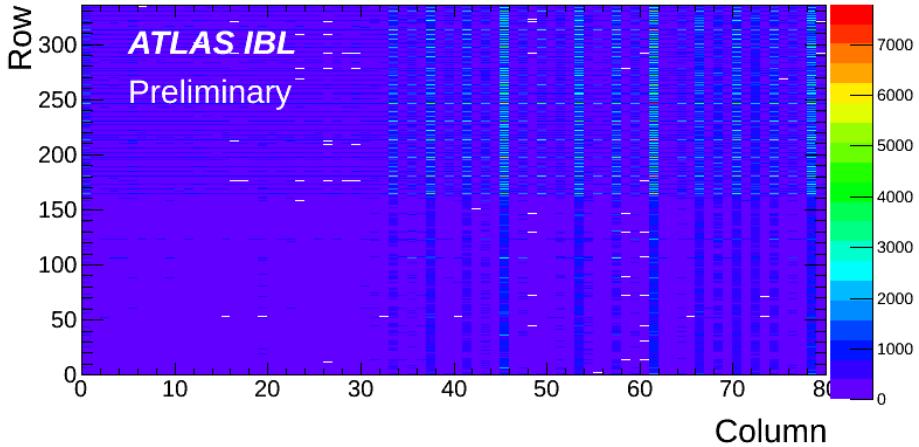
Figure 6. Occupancy histogram generated by the FE emulator after 100000 triggers and 188 hits. Automatic hit generation [15].

and the output delay immediately changes. In total 1249 channels have been tested from the whole card production, including all the channels of the cards finally chosen. The results are presented in figure 5(b) [15]. The average slope of the delay is (34.7 ± 0.7) ps per setting, more than one half lower than the maximum allowed. A single channel measurement in figure 5(a) shows that the delay increases linearly with the setting of the output delay while the duty cycle is only minimally affected [8].

An FE emulator was implemented on the BOC. It is intended for testing the full readout chain without the connection to the optoboard and to the actual detector. The emulator is able to simulate the detector response and sends out an editable number of hits. The emulator works in both run and configuration modes respectively simulating the configuration of the detector and its operational behaviours; it's possible to simulate the reading and writing of the "chip" global registers; and it has a manual as well as an automatic mode. In manual mode one can define the hit pattern, which should be sent when a trigger is received (completed with ID and header/trailer), this only works per trigger. Here a FIFO will be filled and read out. In the automatic mode, instead, one can define how many hits will be generated per received trigger. These hits are then randomly distributed over the "chip".

In figure 6, the occupancy histogram obtained shows sufficient uniformity of the automatic hit distribution generated after 100000 triggers with 188 hits each [15]. This matches the expectations for a correctly working front end chip, when it is properly configured and receives the same number of hits and triggers everywhere. The fact that some pixels and columns are "preferred" is due to imperfections of the random number generator inside the emulator.

5   Conclusion

The firmware for the BOC card is ready and performs very well. Loopback tests were intensively used to qualify the setup and demonstrate a reliable data transmission. The implementation of the fine delay using IODELAY2 has successfully been used and will be adjusted before data taking using the partial reconfiguration method.

References

[1] ATLAS collaboration, *The ATLAS experiment at the CERN Large Hadron Collider*, 2008 *JINST* **3** S08003.

[2] L. Evans and P. Bryant, *LHC machine*, 2008 *JINST* **3** S08001.

[3] G. Aad et al., *ATLAS pixel detector electronics and sensors*, 2008 *JINST* **3** P07007.

[4] M. Garcia-Sciveres et al., *The FE-I4 pixel readout integrated circuit*, *Nucl. Instrum. Meth.* **A 636** (2011) S155.

[5] ATLAS IBL collaboration, *Prototype ATLAS IBL modules using the FE-I4A front-end readout chip*, 2012 *JINST* **7** P11010.

[6] ATLAS collaboration, *ATLAS Insertable B-Layer technical design report*, CERN-LHCC-2010-013, ATLAS-TDR-19 (2010).

[7] M. Červ, *The ATLAS Diamond Beam Monitor*, 2014 *JINST* **9** C02026.

[8] M. Wensing et al., *Testing and firmware development for the ATLAS IBL BOC prototype*, 2012 *JINST* **7** C12027.

[9] G. Balbi et al., *"The Read-Out Driver" ROD card for the Insertable B-Layer (IBL) detector of the ATLAS experiment: commissioning and upgrade studies for the Pixel Layers 1 and 2*, 2014 *JINST* **9** C01044.

[10] Xilinx Inc., *Spartan-6 FPGA Family*, http://www.xilinx.com/products/silicon-devices/fpga/spartan-6/index.htm.

[11] *SNAP12 Multi-Source Agreement*.

[12] J. Anderson et al., *FTK: a Fast Track Trigger for ATLAS*, 2012 *JINST* **7** C10002.

[13] A.X. Widmer and P.A. Franaszek, *A DC-balanced, partitioned-block, 8B/10B transmission code*, *IBM J. Res. Dev.* **27** (1983) 440.

[14] Xilinx Inc., *AR #34276*, http://www.xilinx.com/support/answers/34276.html.

[15] *ATLAS experiment — public results*, https://twiki.cern.ch/twiki/bin/view/AtlasPublic/ApprovedPlotsPixel#IBL_Tests_in_the_Pit.

[16] Xilinx Inc., *FPGA Editor guide*, http://www.xilinx.com/support/sw_manuals/2_1i/download/fpedit.pdf.