

# A Read-out Driver for Silicon Detectors in ATLAS

T. Vickey<sup>a</sup>

on behalf of the ATLAS Collaboration

<sup>a</sup> Department of Physics, University of Wisconsin-Madison, Madison, WI, USA

Trevor.Vickey@cern.ch

## Abstract

I present an overview of a read-out driver (ROD) for silicon detectors in the ATLAS experiment at the Large Hadron Collider (LHC). Two silicon-based ATLAS tracking systems, referred to as the Pixel Detector and the Semiconductor Tracker (SCT), are controlled and read-out using a common 9U VME board. A hybrid design of Field Programmable Gate Arrays (FPGAs) and Digital Signal Processors (DSPs) has allowed the silicon ROD to meet the challenges of format error-counting and event trapping without interfering in the construction and transmission of event fragments to the next level in the read-out system. Performance of the ROD during detector assembly, calibrations and cosmic-ray data-taking are also discussed.

## I. INTRODUCTION

The ATLAS experiment [1] is a general-purpose detector centered around one of the LHC  $pp$  collision points. Two major ATLAS detector subsystems [2] lie closest to the interaction point—the Pixel Detector [3], comprised of more than  $8.0 \times 10^7$  channels, surrounded by the SCT [4], itself containing more than  $6.2 \times 10^6$  channels; both are essential for providing the tracking information in the pseudo-rapidity region  $0 \leq |\eta| < 2.5$  and are used to tag secondary vertexes from  $B$ -hadron decays. The ROD is a 9U VME board common between both of these detector systems and designed to meet the formidable challenge of module configuration and read-out, trigger distribution and event fragment construction.

### A. The ATLAS Pixel Detector

The ATLAS Pixel Detector is the tracking subsystem that lies closest to the interaction point. It is comprised of 1456 Pixel modules [5] in three Barrels and 144 modules in each of two Endcaps.

Each Pixel module has 16 Front-end ICs, with 2880 channels per IC for a total of 46080 channels. There are 18 columns and 160 rows of  $50 \times 400 \mu\text{m}$  pixels on each silicon sensor and the ICs are bump-bonded to the sensor. A Module Controller Chip (MCC) [6] is employed on the module for the purpose of collecting data from the 16 FE chips as well as translating commands into chip signals. Groups of seven modules are connected to a single optoBoard, enabling the transmission and reception of optical signals. The read-out rate for the inner-most Barrel layer, known as the “B-Layer,” is 160 Mbit/s. Both Pixel Endcaps and Layer-1 (the middle Barrel layer) are read-out at a rate of 80 Mbit/s. The outer-most Barrel layer, Layer-2, is read out at a rate of 40 Mbit/s.

### B. The ATLAS Semiconductor Tracker (SCT)

The SCT lies just outside of the Pixel Detector and contains 2112 modules [7] in four Barrels and 988 modules in each of two Endcaps.

An SCT module has two sides with one Hamamatsu silicon sensor on each; the top and bottom sensors are offset from one another by a 40 mrad angle. Each sensor contains 768 instrumented strips at an  $80 \mu\text{m}$  pitch. Each side of the module contains an array of six binary read-out chips, known as the ABCD3TA ASIC [8], that read-out 128 channels each and contain a discriminator, pipeline, data compression logic as well as a read-out buffer. A harness connected to the module includes and opto package to send data off-detector optically; the read-out rate for all SCT Barrels and Endcaps is 40 Mbit/s.

## II. THE SILICON READ-OUT DRIVER (ROD)

The silicon Read-out Driver (ROD) [9] is central to the data acquisition chain (DAQ). The primary purpose of the silicon ROD is module configuration, trigger propagation and data formatting. The secondary purpose of the ROD is detector calibration and monitoring. Control commands are sent from the ROD to the modules as serial data streams. These commands can be Level-1 triggers, bunch-crossing resets, event counter resets, calibration commands or module register data. Each ROD board is responsible for the configuration and read-out of up to 48 SCT or 32 Pixel modules. After formatting the data collected from the modules the ROD transmits these fragments to the ATLAS read-out subsystem (ROS) via S-Link.

Up to 16 RODs can reside inside of a single VME crate. Each VME crate contains one single-board computer (SBC) that is used for communication between the DAQ host computer and the RODs. A single Timing Module (TIM) [10] in the crate receives timing, trigger and control information from ATLAS and passes this information on to the RODs. There is one Back-of-Crate (BOC) [11] card for each ROD and, as the name suggests, the BOC resides in the back of the VME crate just behind the ROD with the backplane forming a partition between them. The BOC is capable of converting electrical commands from the ROD (e.g., triggers) into optical signals and passing them on to the detector modules via optical fibers. Conversely, the BOC can receive optical signals from the modules (e.g., event data) and convert these into electrical signals before passing them on to the ROD. The BOC also hosts the S-Link, which is exploited by the ROD to send formatted event data to the ROS.

The hardware on the ROD itself (Figure 1) is common between the Pixel and the SCT subsystems, however it should be noted that differences in firmware do exist. A hybrid architec-

ture of FPGAs and DSPs allow the ROD maximum versatility during physics-running and calibrations. The FPGAs are dedicated to performing time-critical operations such as ROD setup, module configuration and the formatting, building or routing of events. A single “Master” and four “Slave” DSPs reside on the board and are utilized for the control and coordination of on-ROD operations, as well as performing high-level tasks such as data monitoring and module calibration. Once configured, the ROD FPGAs handle the event data-path to ATLAS Level-2 without further assistance from the DSPs.

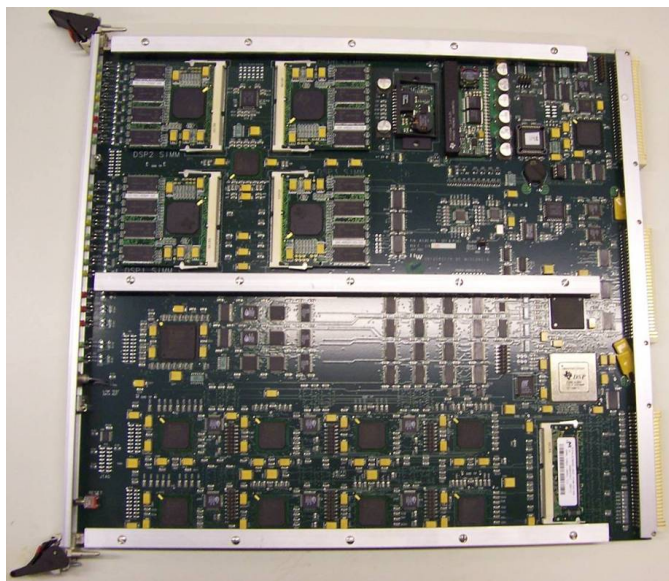


Figure 1: The ATLAS silicon Read-out Driver (ROD).

ROD reset and the VME interface are handled by an FPGA referred to as the Program Reset Manager (PRM). At power-up, the PRM activates the host-to-ROD VME interface and resets the Master DSP (MDSP). Once the MDSP has booted, configuration information is transmitted for the Controller FPGA as well as the data-path FPGAs—known as the Formatter, the Event Fragment Builder and the Router (Figure 2). The farm of Slave DSPs (SDSPs) are loaded with boot code from the host via the MDSP.

The ROD supports two main modes of operation: physics data-taking and calibrations. During physics data-taking triggers issued from ATLAS are relayed to the ROD via the TIM. When running calibrations the MDSP serial ports can be used to issue triggers to the modules. In both modes of operation, the data-path through the Formatter and the Event Fragment Builder remain the same. In calibration mode the Router sends events to the farm of SDSPs for histogramming, whereas in data-taking mode they are routed to the ROS via the S-Link.

In the event that the S-Link is receiving data from the ROD faster than it can be transferred to the ROS during physics data-taking, the S-Link will apply back-pressure to the ROD thereby halting the output of events from the ROD EFB. If back-pressure continues to be applied and the EFB memories become full, back-pressure is propagated to the Formatter FPGAs stopping the transmission of event data from the Formatters. If a critical

limit in the Formatter memories is reached, a ROD Busy signal will be sent to the TIM halting triggers from ATLAS. The likelihood of this occurring even with 2.5 times the projected occupancy at the highest luminosity is exceptionally low.

### A. ROD Controller FPGA and Master DSP

The main function of the ROD Controller FPGA and Master DSP is to setup the ROD data-path, to execute BOC and silicon module configuration and to process as well as propagate triggers.

The MDSP is a Texas Instruments 6201 integer DSP running at 160 MHz with two internal 64 kB blocks of memory and an additional 32 MB of (slower) external memory. The MDSP has ROD and BOC registers connected to one of its External Memory InterFaces (EMIFs) thereby allowing any ROD or BOC registers to be set from the host via the MDSP. The MDSP runs software to perform system functions while the FPGA performs real-time functions. Module configuration is performed by the MDSP using its multi-channel buffered serial ports (SP0 and SP1); configuration data is passed to the MDSP from the host. In calibration mode the MDSP serial ports are also used to send triggers.

During normal ATLAS running the trigger and event description information (Level-1 ID, bunch-count ID and trigger-type) is supplied to the ROD by the TIM. The TIM trigger is detected inside of the Controller and expanded into the trigger codes required by the Pixel and SCT modules. The trigger code is then sent out a 48-wide mask gate and propagated on to the modules.

### B. Formatter FPGAs

The Pixel MCC and the SCT ABCD3TA each return a bit stream. The BOC splits these into individual 40 MHz streams. The ROD input links receive the data streams from the BOC. For the inner-most layers of the silicon tracker, the data comes in over several links at multiples of this 40 MHz.

Eight identical ROD Formatter FPGAs serve the purpose of serial-to-parallel conversion of the event data coming from the modules and to de-randomize event fragments. The Formatters are also used to detect module packet errors and inform the Controller FPGA when link trailers have been received.

The FPGA firmware is subsystem specific. The version for the Pixel Detector contains four links per Formatter and the version supporting the SCT allows for 12 links per Formatter. A single ROD SCT Formatter gathers the data from up to six modules (there are two read-out links per module). For the Pixel Detector the number of modules read out by a single Formatter depends on the rate used. The 40 Mbit/s, 80 Mbit/s and 160 Mbit/s modes support the read-out of four, two and one module(s) per Formatter, respectively.

### C. Event Fragment Builder FPGA

The purpose of the ROD Event Fragment Builder (EFB) is to collect the Formatter output, check the L1 and BC IDs, count errors and generate the event header and trailer.



### C. Tasks

Primitives are typically executed once whereas tasks execute ROD functions over an extended period of time. These operate independent of the Primitive Lists, however the ROD can still process Primitive Lists while various tasks are running. Tasks start and stop by the host (or MDSP) sending primitives and they run until they complete or are halted manually. The Event Trapping and Histogramming Tasks are just a couple of examples. The former runs on the SDSPs to handle event trapping while the latter manages the issuance of triggers along with the processing and binning of event data.

## IV. OPERATING MODES

Both physics data-taking and calibration modes are supported by the silicon ROD.

### A. Physics Data-taking

Once the data-path on the ROD has been setup, it is completely handled by the FPGAs and does not require any intervention from the DSPs. During physics data-taking the Router FPGA possesses the capability of capturing events on a user-defined pre-scale (i.e., every *n*th event) and sending them to the farm of SDSPs on a non-interfering basis. One could conceive of collecting such events, binning them, and comparing with a set of reference histograms thereby providing detector experts with information on channels with unusually high or low occupancies.

### B. Calibration

In calibration mode the transmission of data through the S-Link is inhibited. Instead, frames of data (256 32-bit word chunks) are passed from the Router FPGA to the SDSPs using Direct Memory Access (DMA) transfer. Tasks running on the SDSPs flag these transferred events for processing and subsequent histogramming. A monitoring task can be run on the SDSPs that is capable of parsing the event errors flagged by the FPGAs and reporting these errors back to the host.

Two of the most common Pixel calibration scans focus on the determination of thresholds and noise. During the Threshold Scan, on-chip charge injection is carried out for each individual pixel. The number of hits for each injected charge is scanned to obtain the discriminator threshold (Figure 3). A Noise Scan is simply a Threshold Scan without any charge injection and is used to measure the noise occupancy.

SCT calibration scans include tests of basic communication such as the Rx Threshold Test (optimize the Rx threshold value in the BOC data-receiver chip), as well as a suite of digital and analog tests like the N-Mask Test (demonstrate that the chip mask register functions properly; Figure 4) and the Noise Occupancy Test.

## V. ROD PERFORMANCE

Production RODs have been used extensively during silicon detector assembly, runs with test beam, and most recently dur-

ing a period of cosmic-ray data-taking.

### A. ROD use during Detector Assembly

Calibration scans have been used routinely during module production [12] to verify module functionality before and after being mounted or transported. Such scans also help to rank production modules by the number of defects so that only the very best may be mounted on the detector and installed in ATLAS.

The detector assembly phase has given essential feedback to the ROD hardware, firmware and software developers. Several DSP software and FPGA firmware problems have been uncovered during the production and assembly tests thus giving the programmers critical feedback before physics running with complete detectors in the ATLAS Collision Hall.

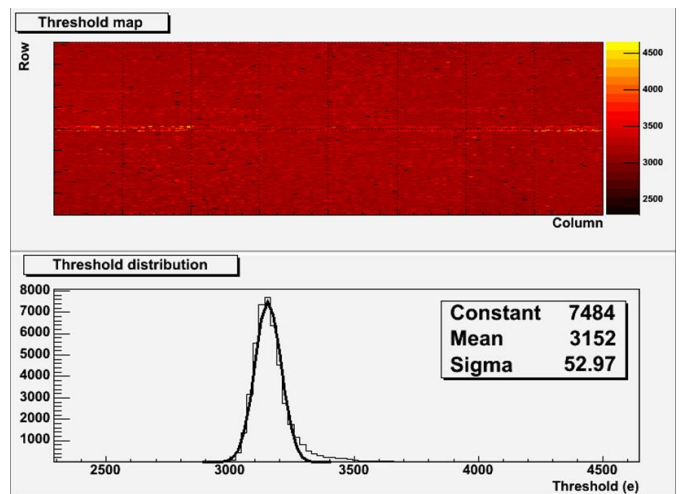


Figure 3: The results of a Pixel Threshold calibration scan.

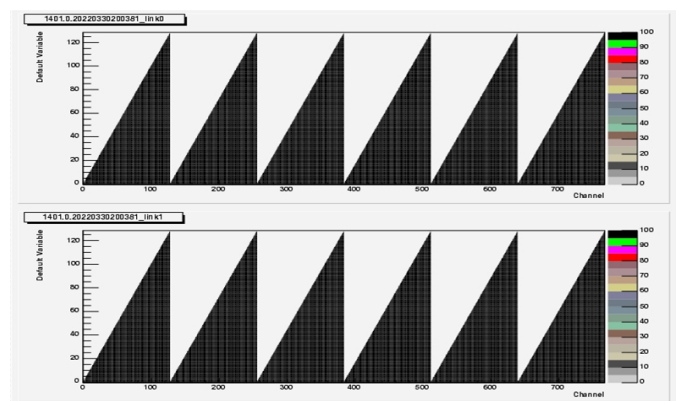


Figure 4: The results of an SCT N-Mask calibration scan.

### B. ROD use during Cosmics Running

The production ROD as well as some of the very latest FPGA firmware and DSP software were tested extensively during a period of combined cosmics running using the ATLAS Transition Radiation Tracker (TRT) and SCT Barrels in May 2006. The ROD performed well during this very successful run

in which more than 400k events were collected. This large-scale test involved  $\sim 500$  SCT modules ( $\sim 10$  RODs) and served as a useful DAQ and detector shake-down in addition to providing data for detector efficiency and alignment studies.

## VI. CONCLUSIONS

Approximately 260 production ROD boards have been assembled and tested. Roughly 160 of these are dedicated to the Pixel Detector while the remaining fraction are assigned for use by the SCT Barrels and Endcaps. Many of these boards are now installed in the ATLAS USA-15 counting-room crates; their location for ATLAS physics data-taking.

The final production version of the ROD has already been used extensively during module production and detector assembly at various sites around the world, as well as during test-beam runs at CERN [13] and the combined cosmic-ray data-taking run.

We are looking forward to larger-scale multi-crate tests and with the SCT Barrels now lowered into the ATLAS Collision Hall that opportunity is rapidly approaching. Additional testing during the upcoming cosmics data-taking run using one of the Pixel Endcaps is another opportunity to gain additional experience reading-out large numbers of modules.

Future work will focus on speeding up the DSP software to shorten the time taken for both module configuration and calibration scans. Any implementation of on-ROD data-quality and detector monitoring during physics data-taking as well as an automated response for module recovery and re-configuration (in conjunction with the DAQ and Detector Control System) would certainly prove to be very beneficial.

## VII. ACKNOWLEDGEMENTS

We acknowledge the support of the funding authorities of the collaborating institutes including the Australian Research Council; the Australian Department of Education, Science and Training; the Bundesministerium für Bildung und Forschung of Germany; the Ministry of Education, Culture, Sports, Science and Technology and the Japan Society for promotion of Science; Norwegian Research Council; the Polish State Committee for Scientific Research; Ministry of Higher Education, Science and Technology of the Republic of Slovenia; the Spanish National Programme for Particle Physics; Swedish Research Council; Knut and Alice Wallenberg Foundation Sweden; the Particle Physics and Astronomy Research Council of the United King-

dom; the United States Department of Energy and the United States National Science Foundation.

## REFERENCES

- [1] ATLAS Detector and Physics Performance: Technical Design Report, CERN-LHCC-99-014 (1999); CERN-LHCC-99-015 (1999).
- [2] ATLAS Inner Detector: Technical Design Report, CERN-LHCC-97-016 (1997); CERN-LHCC-97-017 (1997).
- [3] ATLAS Pixel Detector: Technical Design Report, CERN-LHCC-98-013 (1998).
- [4] J. N. Jackson *et al.*, The ATLAS semiconductor tracker (SCT), *Nucl. Instrum. Meth.* **A541**, 89 (2005).
- [5] M. S. Alam *et al.*, The ATLAS silicon pixel sensors, *Nucl. Instrum. Meth.* **A456**, 217 (2001).
- [6] R. Beccherle *et al.*, MCC: The Module Controller Chip for the ATLAS pixel detector, *Nucl. Instrum. Meth.* **A492**, 117 (2002).
- [7] D. Robinson *et al.*, Silicon microstrip detectors for the ATLAS SCT, *Nucl. Instrum. Meth.* **A485**, 84 (2002).
- [8] F. Campabadal *et al.*, Design and performance of the ABCD3TA ASIC for readout of silicon strip detectors in the ATLAS semiconductor tracker, *Nucl. Instrum. Meth.* **A552**, 292 (2005).
- [9] D. Fasching *et al.*, A Read-out Driver for the ATLAS Silicon Detector Modules, (in preparation).
- [10] J. Butterworth *et al.*, TTC Interface Module for ATLAS Read-Out Electronics: Final production version based on Xilinx FPGA devices, ATL-COM-ELEC-2005-001 (2005).
- [11] M. L. Chu *et al.*, The off-detector opto-electronics for the optical links of the ATLAS Semiconductor Tracker and Pixel detector, *Nucl. Instrum. Meth.* **A530**, 293 (2004).
- [12] S. Moed, EndCap module production of the ATLAS Semiconductor Tracker, *Nucl. Instrum. Meth.* **A560**, 75 (2006).
- [13] F. Campabadal *et al.*, Beam tests of ATLAS SCT silicon strip detector modules, *Nucl. Instrum. Meth.* **A538**, 384 (2005).