The Quality-Assurance Test of the ATLAS New Small Wheel Read-Out Controller ASIC

Stefan Popa[*][†]

Transilvania University, Brasov, Romania
E-mail: stefan.popa@unitbv.ro; stefan.popa@cern.ch

Sorin Martoiu

IFIN-HH, Magurele, Romania
E-mail: sorin.martoiu@cern.ch

Mihai Luchian

Transilvania University, Brasov, Romania
E-mail: luchiann.mihai@gmail.com

Radu Coliban

Transilvania University, Brasov, Romania
E-mail: coliban.radu@unitbv.ro

Mihai Ivanovici

Transilvania University, Brasov, Romania
E-mail: mihai.ivanovici@unitbv.ro

The Read-Out Controller (ROC) ASIC will be used to store, de-randomize, aggregate, filter and form complex packets with the digitized data coming from the New Small Wheel (NSW) muon detectors of the ATLAS experiment. The ASIC test setup is based on a Xilinx Kintex Ultrascale FPGA evaluation board, implementing input data streams emulators and output data analyzers for functional verification which are controlled and monitored by a MicroBlaze microprocessor. The jitter and skew of the ASIC's PLL outputs are measured using oscilloscopes and logic analyzers. The design validation, test procedure and quality-assurance mass-testing results are presented.

---

[*]Speaker.

[†]On behalf of the ATLAS Muon Collaboration

1. Introduction

The Read-Out Controller (ROC) [1] is a custom data packet processor implemented as an ASIC (Application Specific Integrated Circuit) in the IBM CMOS (Complementary Metal Oxide Semiconductor) 130 nm technology. The die is squared shape with an area of 22.5 mm^2 and has 232 pads from which 187 are for IO (Input Output) signals. The ROC will be used to store, de-randomize, aggregate, filter (based upon an additional level of trigger) and form complex packets with the digitized data coming from the New Small Wheel (NSW) [2] muon detectors within the ATLAS (A Toroidal LHC ApparatuS, LHC is Large Hadron Collider) [3] experiment. It will also distribute 40 MHz BC (Bunch Crossing) clock signals, 160 MHz RO (Read Out) clock signals and decoded TTC (Time Trigger and Control) commands to the other front-end integrated circuits providing fine skew and latency control. The ROC ASIC is designed to work in both Phase-I and Phase-II environments being compatible with either one or two-level trigger scenarios.

For testing the first ROC dies, validate the implementation and then mass-test the BGA (Ball Grid Array) packaged ROC a custom test setup was developed consisting of a Xilinx Kintex Ultrascale KCU105 FPGA (Field Programmable Gate Array) evaluation board and custom PCBs (Printed Circuit Boards) for the ROC.

2. Test setup

We embraced a generic test setup that consists of stimuli generators (SGs) which are connected to the DUT's (Device Under Test) inputs, output data capture and analysis (OCA) modules for each DUT output and a monitor and control (MC) module as depicted in Figure 1. The SGs are configurable in order to create test scenarios that cover all the functional features of the DUT. The OCA modules check the coherency of all the output data in correlation with the data injected by the SGs into the DUT. The MC module configures the DUT and the SGs, ensures the timing and the sequence of operations, collects statistics and interprets the testing results.
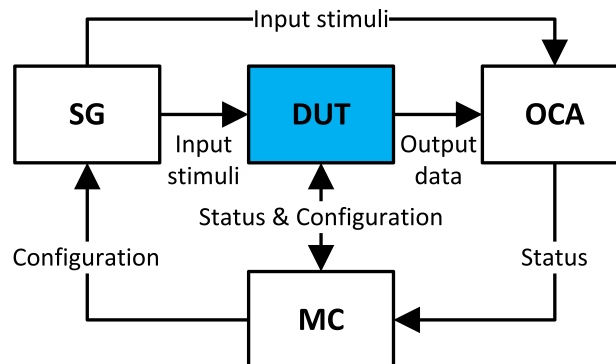

Figure 1: Generic test setup structure.

For the implementation of the proposed architecture from Figure 1 we chose an FPGA evaluation board that meets all the speed and resources requirements for the fully-configurable design.

The FPGA firmware contains packet generators emulating the ROC input data streams, a TTC stream emulator and two I^{2}C (Inter-Integrated Circuits) masters for the ROC's control and status

check. Firmware-based output data analysers check all the ROC output data for encoding, protocol coherency, parity, checksum and content errors. All these modules are controlled and monitored using a MicroBlaze soft-core microprocessor. The DUT is located on custom PCBs which assure the voltage supply and the interconnection between the ROC and the FPGA through FMC (FPGA Mezzanine Card VITA-57) connectors.

2.1 Testing PCBs

On the testing PCB the ROC is powered by two LDO (Low DropOut) voltage regulators. The input voltage of the regulators is selectable between an external source and the 3.3 V supplied by the FPGA evaluation board through the FMC connector.

The first version of the PCB routed the phase-adjustable ROC output clock signals and the decoded TTC commands (which are in phase with these clock signals) to pin headers in order to be measured with an oscilloscope or logic analyzer. All the other ROC signals are routed to the FMC HPC (High Pin Count) connector.

The second board version uses an Yamaichi [4] open-top socket for fast insertion in order to optimize the mass-testing process of the BGA packaged ROC. The regulators are programmable and current and power monitors were added. The ROC output clock and TTC signals are routed directly or through high-speed switches to the second FMC connector, LPC (Low Pin Count). This allows for complete automatic test coverage of the ASIC, including jitter and skew evaluations, without the use of external measurement devices. All these programmable integrated circuits are connected to the dedicated I^{2}C bus of the HPC FMC connector.

2.2 Testing firmware

The testing firmware, implementing the structure from Figure 1 on the FPGA, requires a clock signals supply system (to drive the DUT and the various SGs, MC and OCA modules) and the use of a RISC (Reduced Instruction Set Computer) microprocessor to perform all the control, monitoring and reporting of test results in a fast, easy to upgrade and flexible way.

The FPGA evaluation board is configured to supply a 160 MHz clock signal to the FPGA. Inside the FPGA a MMCM (Mixed-Mode Clock Manager) uses this clock as a reference and outputs three phase aligned clock signals, one with 320 MHz, one with 160 MHz and the last one with 40 MHz frequency. The 320 MHz clock signal is required by the input and output serial delay lines used to synchronize the serial data streams between the FPGA and the ROC. The 160 MHz clock signal is supplied to the rest of the firmware. The 40 MHz clock signal is forwarded to the ROC which uses it as a reference clock for all its PLLs (Phase-Locked Loop) and for driving the configuration logic of these PLLs.

A Xilinx MicroBlaze [5] soft-core 32-bit RISC microprocessor is interfaced with a 128 KB dual-port BRAM (Block Random Access Memory) memory for data and instructions, has a basic JTAG (Joint Test Action Group) debug module, an interrupt controller and an AXI4 (Advanced eXtensible Interface) data interface. The AXI4 data interface is connected to an AXI Interconnect Core which works as a bus scheduler for the 4 peripherals in the design: AXI UART (Universal Asynchronous Receiver-Transmitter), AXI I^{2}C, AXI Timer and AXI Register Bank. These form the MC module of the test setup. The AXI UART core is used as the user interface. The AXI I^{2}C

core controls the integrated circuits on the second version of the ROC testing PCB. The AXI Timer is used to time the ROC tests and other events. The AXI Register Bank is a custom register bank containing 64 32-bit registers. The outputs of the first 36 registers are control signals for the SGs and the DUT. The inputs of the last 28 registers are the status and error flags of the DUT and OCA modules.

The SGs consist of eight input data generators, one for each input channel of the ROC, which inject L0 (Level 0) events into the ROC. The average frequency of the generated packets can be selected from 100 kHz to 1400 kHz in 100 kHz steps. For each frequency the percentage of empty packets (events with no detector data) and the average size of the non-empty packets can be selected from predefined values. Worst case trigger and packet bursts scenarios were hard-coded. There is the option of using constant size events and constant frequency which makes debugging easier. The content of the packets is deterministic, different for each generator. The resulted stream is 8b10b encoded and fed to two DDR (Double Data Rate) serializers, resulting a throughput of 640 Mbps per generator. The outputs of the two serializers are passing through calibrated configurable FPGA delay lines (ODELAY Xilinx Primitive). The identifiers of the generated L0 events are pushed into a TTC latency FIFO (First-In-First-Out).

The SGs also include the TTC Generator which synchronizes the BCID (Bunch Crossing IDentity) counters from the FPGA and the ROC by sending the corresponding TTC commands through the 320 Mbps TTC stream. Using the local BCID counter and the trigger information from the TTC latency FIFO the TTC Generator is able to sent L1 trigger commands that match the sent L0 events. The latency of the L1 triggers in relation to the matching L0 events is configurable in steps of 1 μs from 20 μs up to 300 μs.

Two I^{2}C masters with 10-bit addressing and dedicated I^{2}C buses are used to configure and verify the status of the ROC's PLLs and digital logic.

The OCA modules contain DDR de-serializers conected to the serial data lines from each SROC (Sub ROC) module within the ROC via FPGA calibrated and configurable delay lines (IDELAY Xilinx primitive). The resulted data stream is passed to an alignment module that searches for K.28.5 8b10b comma symbols. After at least two consecutive comma symbols are detected the alignment to the data stream is achieved. The aligned data stream is passed to an 8b10b decoder which supplies one byte at a time to the Assembler FSM (Finite-State Machine). This FSM checks the coherency of the data, the protocol syntax, the parity bits, the checksum, the reported length, the expected content and the L1 (Level 1) trigger information of each output event. Error and status flags are supplied to the MicroBlaze. L1 event counters are used to check the synchronicity of the four SROCs.

2.3 Testing software

A single threaded standalone C program runs on the MicroBlaze and uses the UART interface as standard input (stdin) and standard output (stdout) channels. All the peripherals from the MicroBlaze system have address spaces allocated in the 32-bit virtual memory. The access to the content of the AXI Register bank and the configuration and status registers of the other peripherals is thus made using pointers.

The C program initializes and tests the peripherals, then detects the ROC testing PCB version by interrogating the corresponding integrated circuits through I^{2}C. It calibrates the phases of the

output and input signals coming out of and into the FPGA respectively, so that both the ROC and the FPGA de-serializers will sample the data in the middle of the bit in order to avoid setup and hold violations.

For mass-testing we designed a suite of 10 tests that were programmed to cover all the functional features of the ROC. All the settings within the ROC configuration are varied in order to test the corresponding effects. Different size, content and frequency for the input packets and the L1 triggers are used. Each test is run for several seconds in order for all the ROC FIFO pointers to loop several times and write and read various deterministic patterns. A chip is considered *good* if it passes all the tests with no errors.

3. Results and Conclusions

We implemented the ROC ASIC which is a data packet processor that will be used in the data path of the NSW muon detectors of the ATLAS experiment. We validated the design by extensively testing the ASIC using a custom test setup based on a Xilinx Kintex Ultrascale FPGA evaluation board, custom firmware, software and PCBs. The test setup was optimised for mass-testing the ROC ASICs needed for the experiment. Each chip will be subjected to a suite of 10 tests that cover all the functional features. Additional firmware and software updates for the test setup will allow jitter and skew evaluations, without the use of external measurement devices.

An initial batch of 130 ASICs were tested using the mass-testing setup. The output clock signals were evaluated using osciloscopes and logic analyzers. The majority of chips showed a RMS (Root Mean Square) jitter of 15 ps. Several chips had higher jitter (up to 20 ps). The phase control of the output frequency works as expected. The resulting yield is 84%. The extrapolated number of ROCs that will be produced is 10,000.

References

[1] R.-M. Coliban et al, *The Read Out Controller for the ATLAS New Small Wheel*, *Journal of Instrumentation*, 11(16):C02069, 2016.

[2] ATLAS Collaboration, *New Small Wheel Technical Design Report*, *2013, CERN-LHCC-2013-006, ATLAS-TDR-020*, http://cdsweb.cern.ch/record/1552862

[3] ATLAS Collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, *Journal of Instrumentation*, 3(8):S08003, 2008.

[4] Yamaichi Electronics, *1.00mm Pitch Series NP352 / NP483 / NP486 (Open top - TH) Overview BGA / CSP* https://www.yamaichi.de/uploads/media/bga-100-pitch_01.pdf

[5] Xilinx, *MicroBlaze Processor Reference Guide*, June 21, 2018, http://www.xilinx.com/support/documentation/sw_manuals/xilinx2018_2/ug984-vivado-microblaze-ref.pdf