

# The Final LHCb Readout Supervisor “ODIN”

R. Jacobsson, P. König

CERN, 1211 Geneva 23, Switzerland  
Richard.Jacobsson@cern.ch, Pascal.Koenig@cern.ch

A. Chlopik, Z. Guzik

Soltan Institute for Nuclear Studies, Swierk-Otwock, Poland  
arek@ipj.gov.pl, zbig@ipj.gov.pl

## *Abstract*

The LHCb Timing and Fast Control (TFC) system is entering the final phase of prototyping. This paper proposes improvements of the main TFC modules.

Since the Readout Supervisor “ODIN” is a very complex module, the prototyping was staged by starting out with a minimal version incorporating the most critical logic and then adding the remaining logic in subsequent prototypes. The paper also covers the additions in order to implement the final full version.

## I. INTRODUCTION

A more detailed description of the overall architecture and the functionality of the LHCb Timing and Fast Control system can be found in Reference [1] and [2].

The TFC system is responsible for maintaining the timing of the LHCb synchronous readout. It distributes the LHC clock, the two high-rate triggers (Level 0 and Level 1) of LHCb, and synchronous reset and control commands to the Front-End Electronics. It also controls the trigger rates in order to ensure that overflows do not occur in the Front-End buffers.

The distribution network is based on the CERN TTC system [3] and some of the standard TTC modules. However, the entire mastership of the TFC system is located in an LHCb specific module: the Readout Supervisor “ODIN”[4]. The TFC Switch “THOR” [5] is a programmable patch panel that allows partitioning, that is, completely autonomous running of parts of the Front-End electronics by associating one of the extra ODINs. The two Throttle Switches “Munin” feed back throttle signals from the slower buffers in the synchronous readout as a means of signalling imminent buffer overflows.

A first prototype of the TFC Switch “THOR” has been built. The aim with the prototype was to test and measure the fast switch logic and implement a first version of the control interface. The control interface is based on a commercial Credit Card PC from *Digital Logic AG* with Ethernet. This interface is one of the LHCb standards and is used on all the LHCb specific TFC modules.

A first prototype of ODIN has also been built. The first version was a minimal version[6] and the aim was to test the most important and critical functionality.

The second prototype of the TFC Switch “THOR” and the Readout Supervisor “ODIN” are underway. The second TFC Switch is intended to be a final version. The second ODIN will be a first prototype of the complete final ODIN.

In addition to describing the additions and the major improvements this paper also outlines the implementation and the use of the control interface. ODIN is a complex board and requires an efficient user interface for the control to make the debugging possible. It is mentioned at the end.

## II. MODIFICATIONS AND IMPROVEMENTS

With respect to the first prototype, the second version of ODIN will house more performance and statistics counters and more state machines for sending various types of auto-triggers and commands to the Front-End electronics etc. It will also have history buffers for the internal and external throttle signals, but most importantly the second version contains several important modifications and additions. They are described below.

### *A. Programming, configuring and monitoring*

The board bus to configure parameters and read status and monitoring information on the first prototype was based on a simple non-multiplexed 32-bit bus with five address lines and chip selects. The bus was formed in an intermediate FPGA from a PLX local bus derived from the PCI bus of the CC-PC. A JTAG bus for programming the FPGAs was routed across an intermediate interface mezzanine, which also carries the PLX chip, from the parallel port of the CC-PC and split up as a star in the intermediate FPGA. This allows each of the ten FPGAs to be programmed individually.

In the new version of ODIN the board bus will entirely be based on a multiplexed 32-bit PLX bus derived directly from the PCI bus of the CC-PC. Instead of driving a JTAG from the parallel port, it will be formed by mapping a predefined PCI memory addresses in the user space to a JTAG bus and do the JTAG transactions via PCI. This allows implementing easily several JTAG busses. The new ODIN will also have an  $\overset{?}{\text{F}}\text{C}$  bus implemented in the same way using PCI.

The new approach is based on a self-designed PCI interface embedded into an APEX chip. The PCI core implements most of the functions of a PLX9030. The FPGA is implemented on a “Tiny Glue” mezzanine shown in Figure 1 and it is programmed in-situ via a Byteblaster using the parallel port of the CC-PC. In addition to the PCI core, the FPGA contains the  $\mu$ C state machine and the necessary logic to drive the JTAG busses. The General Purpose I/O lines of the PLX has also been implemented and are used to drive the system reset of the entire ODIN logic and various “hard” parameters such as selecting between using an internal clock or the external clock to drive ODIN.

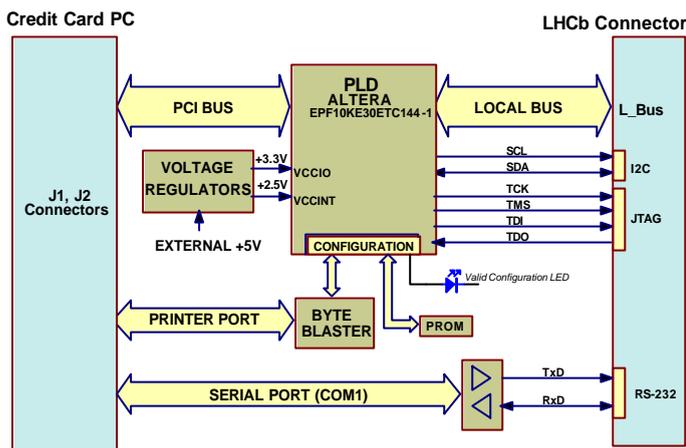


Figure 1: Block diagram of the intermediate mezzanine “Tiny Glue” between the control interface, the CC-PC, and the board logic.

## B. The FPGA chip set

In the current versions of the TFC prototype boards, the digital processing is almost entirely based on the Altera FLEX 10KE family of FPGAs and MAX 7000B for logic functions demanding speed. Due to limited capacity of these chips it was necessary to implement many such FPGAs, which rendered difficult the interfacing and satisfying the speed and latency criteria.

The Altera APEX family is characterized by more than two times better speed performance and has typically 20 times more logic gates than the FLEX family. Therefore the APEX20K200EQC240-1 has been chosen. Deploying these devices allows reducing ODIN’s entire chip count from ten to only four.

The first version of ODIN was written in AHDL. In order to make the FPGA code more manageable and facilitate the reorganisation of the function, a coding convention and a new modular structure were defined and the entire functionality was rewritten into VHDL.

The organization of the functionality in the four chip has been made by minimizing board connections between related functions. The aim was also to reduce latency between functions that in the current prototype originated from the need of pipelining at the inputs and the outputs of the chips. The organization of the new ODIN is shown in Figure 2. The four chips have the following most important tasks:

- **Q\_L0** : Handling of the L0 triggers, including pipelining with a programmable depth,

synchronisation check, L0 auto-triggers, rate control, L0 accept FIFO control and L0 statistics. It also incorporates a FIFO which is 160 deep to sample eight bits of external information that, if possible, will be commig from the special LHC beam monitors located near the experiments. The data is subsequently handled by the ODIN Front-End.

- **Q\_L1** : Handling of the L1 triggers, including synchronisation check, L1 auto-triggers, rate control, L1 trigger derandomizer control, L1 trigger broadcast requesting and L1 statistics.
- **Q\_GEN** : A mix of tasks of which the most important are handling of all TTC channel B broadcasts, all local and Front-End resets, Front-End commands and related statistics. It also manages the local TTCrx and formats data from the TTCrx (see below).
- **Q\_FE** : Handles the ODIN Front-End data and controls the L1FE buffer (see below). It also contains the SLINK functionality.

## C. L0 accept FIFO and L1 trigger derandomizer

ODIN contains a FIFO to store the Level 0 accept triggers (AFIFO) during the latency of the Level 1 triggers. It was considered to use the internal RAM bits of the FPGAs to implement the FIFO on the second version. However, as the latency of the Level 1 trigger is subject to discussion and will probably increase by several times, it will still be implemented in a static FIFO.

ODIN also contains a FIFO to derandomize the L1 trigger broadcasts (TFIFO) in order to take into account the readout times needed by the Front-End electronics to take out an event from the L1 buffer. For the same reason as above, it will also still be implemented in a static FIFO.

## D. ODIN Front-End

In terms of functionality the most important addition to the second version is the ODIN Front-End. The ODIN Front-End samples run related information, statistics and performance data and transmits it to the Data Acquisition System for storing with the event data. In particular this will consist of:

- Sampled at 40 MHz:

Bunch current	8 bits
<b>Total</b>	<b>8 bits</b>
- Sampled at positive Level 0 decision (~1 MHz)

“Level 0 bunch current”	8 bits
Bunch ID (internal)	12 bits
GPS time	40 bits
Detector status	24 bits
L0 Event ID	24 bits
Trigger type	3 bits
L0 force keep bit	1 bit
Bunch ID (received)	12 bits

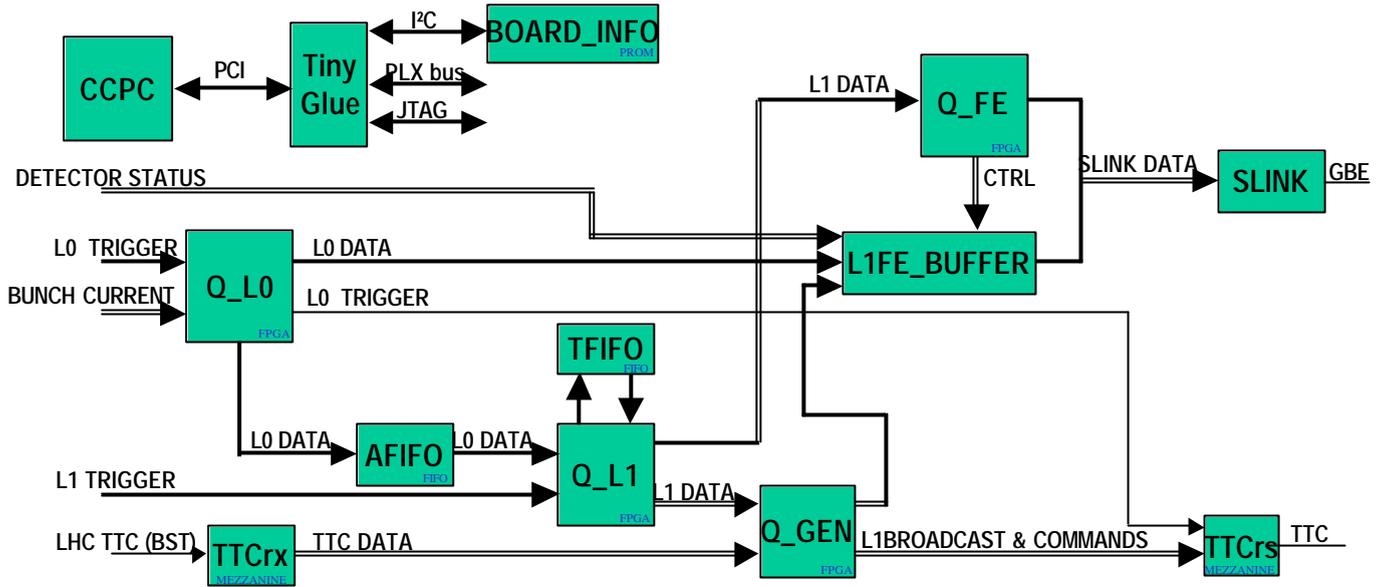


Figure 2 : Block diagram of the data flow in the second version of ODIN

Bunch crossing type	2 bits
L0 synchronisation error	1 bit
L0 synch. error, kept	1 bit
<b>Total</b>	<b>128 bits</b>

- Sampled at arrival of positive L1 decision (~40 kHz)

L0 data above	128 bits
L1 event ID	32 bits
L0 Event ID (received)	12 bits
Bunch ID (received)	2 bits
L1 decision	1 bit
L1 synch. error(Bunch ID)	1 bit
L1 synch. error (Event ID)	1 bit
L1 synch. error, kept	1 bit
<b>Total</b>	<b>178 bits</b>

The eight bits of data sampled at 40 MHz will be stored during the latency of the L0 trigger (4  $\mu$ s) in a pipeline implemented using the internal RAM bits of the APEX chip. The 128 bits of L0 data will be stored in an external RAM (L1FE\_BUFFER) during the latency of the L1 decision. Currently this buffer is ~2000 events deep but an increase of the L1 latency by several times times is being discussed. At the arrival of the L1 trigger the 178 bits of data will be temporarily stored in the “Q\_FE” while being transmitted to the DAQ.

### E. Interface to the DAQ

The interface to the DAQ system consists of an SLINK-to-GbEthernet link. The idea is to incorporate the S-LINK functionality[7] and the LHCb protocol in the “Q\_FE” FPGA and implement the Ethernet MAC directly on the board.

### F. TTCrx mezzanine

The TTCrx on the TTCrx mezzanine will be connected to the LHC Beam Synchronous Timing (BST) information network. The BST network will distribute a number of long broadcasts every turn of the accelerator with the status of the machine, information about the beams, and the GPS time. The “Q\_GEN” FPGA will control the TTCrx and will extract in particular the GPS time to store it in the ODIN Front-End data. This is useful in order to be able to cross-check later with information stored in databases about status of other systems, such as the detector controls.

### G. Board information

Since the control interface, the CC-PC, can be replaced its ethernet address can not be used for board identification from the point of view of the control system. Instead ODIN will carry a 128-bit I<sup>2</sup>C EPROM (BOARD\_INFO) that can be read directly from the control system. It will contain the board identifier, the version and the revision number and a serial number.

## III. LOCAL CONTROL SYSTEM

To facilitate the testing and the debugging of the TFC modules a “local control system” with graphics user interfaces has been devised. It will also serve as a “TFC run control” when testing and commissioning Front-End electronics systems. It is also a first step towards integrating the TFC into the overall Experiment Control System of LHCb. The local control is based on the PVSS system and the Distributed Information Management (DIM) package. DIM allows to set up a server which is running in the CC-PC and with which PVSS can communicate over the network by subscribing to services and commands that the server publishes. In order to access the hardware, the server is linked with a library for accessing PCI. The server is also linked with a library

containing a STAPL player in order to program the FPGAs using STAPL byte-code data.

When a change is applied to a parameter in the PVSS system the information is passed to the server via the command. The server writes the register and passes back the result to the PVSS system. A similar mechanism is used to send programming data for the FPGAs. As mentioned above the server executes the STAPL player and writes the JTAG data to the FPGA via the PCI bus.

#### IV. CONCLUSION

The first prototypes of the TFC modules specific to LHCb have been built and are being tested. The next version of the TFC Switch "THOR" is aimed to be the final prototype. It has already been designed and will soon be produced.

The first prototype of ODIN was a minimal version to test critical functions. The next version of ODIN will be a first prototype of the complete ODIN and will include all remaining functionality. Detailed specifications have been written and the actual implementation have been outlined.

#### V. REFERENCES

- [1] R. Jacobsson and B. Jost, "The LHCb Timing and Fast Control system", LHCb 2001-016 DAQ.
- [2] R. Jacobsson et al., "The LHCb Timing and Fast Control System", proceedings to the LEB2001 conference.
- [3] RD-12 Documentation on WWW (<http://www.cern.ch/TTC/intro.html>) and references therein.
- [4] R. Jacobsson, B. Jost, Z. Guzik, "Readout Supervisor Design Specifications", LHCb 2001-012 DAQ.
- [5] Z. Guzik, Richard Jacobsson, and B. Jost, "The TFC Switch specifications", LHCb 2001-018 DAQ.
- [6] Z. Guzik, R. Jacobsson, A. Chlopik, "Implementation issues of the LHCb Readout Supervisor", proceedings to the LEB2001 conference.
- [7] O. Boyle et al., "The S-LINK Interface Specification", <http://www.cern.ch/HSI/s-link/>.