Available on CMS information server **CMS CR -2016/257**

The Compact Muon Solenoid Experiment

Conference Report

Mailing address: CMS CERN, CH-1211 GENEVA 23, Switzerland

13 October 2016 (v3, 27 October 2016)

Implementation of the data acquisition system for the Overlap Modular Track Finder in the CMS experiment

Wojciech M. Zabolotny, Michal Bluj, Karol Bunkowski, Adrian P. Byszuk, Jakub Dobosz, Krzysztof Doroba, Pawel Drabik, Maciej Gorski, Artur Kalinowski, Krzysztof Z. Kierzkowski, Marcin Konecki, Wojciech Oklinski, Michal Olszewski, Krzysztof Pozniak, Krystian Zawistowski on behalf of the CMS collaboration

Abstract

The CMS experiment is currently undergoing the upgrade of its trigger, including the Level-1 muon trigger. In the barrel-endcap transition region the Overlap Muon Track Finder (OMTF) combines data from three types of detectors (RPC, DT, and CSC) to find the muon candidates.To monitor the operation of the OMTF, it is important to receive the data which were the basis for the trigger decision. This task must be performed by the Data Acquisition (OMTF DAQ) system.The new MTCA technology applied in the updated trigger allows implementation of the OMTF DAQ together with the OMTF trigger in the MTF7 board. Further concentration of data is performed by standard AMC13 boards.The proposed data concentration methodology assumes parallel filtering and queuing of data arriving from all input links (24 RPC, 30 CSC, and 6 DT). The data are waiting for the trigger decision in the input buffers. The triggered data are then converted into the intermediate 72-bit format and put into the sorter queues. The block responsible for the building of events receives data originating from the particular Bunch Crossing (BX) from the consecutive sorter queues, converts them to the 64-bit AMC payload words, and puts them into the output queue. That block also generates the AMC header at the beginning and the AMC trailer at the end of the event data.The system is implemented in a flexible way, and handling of a new data source requires implementation of two specialized blocks the input data formatter to translate the link data into the sorter queue data and the output data formatter to translate the sorter queue data into the AMC payload. The AMC payload format used by the OMTF DAQ provides bit field allowing the context-free detection of the data source.The system may send data not only from the bunch crossing (BX) in which the L1 trigger was generated but also from a configurable number of BXs before the trigger (up to 3) and after the triger (up to 4). Therefore, according to the current trigger rules, it is possible that the data from a certain BX may belong to two different events. To handle such cases the OMTF DAQ system uses two output queues alternately for assembling the consecutive events. It is easily possible to increase the number of output queues if a single BX may belong to a higher number of events due to the change of the trigger rules or number of BX-es transmitted before or after the trigger.The system in current state handles the RPC data. The data handlers for CSC and DT detectors are being developed. The presented methodology may be reused for other triggered DAQ systems concentrating data from various sources with different formats.

Presented at *TWEPP-16 Topical Workshop on Electronics for Particle Physics*

Prepared for submission to JINST

Implementation of the data acquisition system for the Overlap Modular Track Finder in the CMS experiment

---

Wojciech M. Zabołotny,[a,b,1] **Michał Bluj,**[c] **Karol Buńkowski,**[b] **Adrian P. Byszuk,**[a,b] **Jakub Dobosz,**[b] **Krzysztof Doroba,**[b] **Paweł Drabik,**[a] **Maciej Górski,**[c] **Artur Kalinowski,**[b] **Krzysztof Z. Kierzkowski,**[b] **Marcin Konecki,**[b] **Wojciech Okliński,**[b] **Michał Olszewski,**[b] **Krzysztof Poźniak**[a,b] **and Krystian Zawistowski**[b] **on behalf of the CMS collaboration**

[a]*Institute of Electronic Systems, Warsaw University of Technology*
ul. Nowowiejska 15/19, 00-665 Warszawa, Poland
[b]*Institute of Experimental Physics, Faculty of Physics, University of Warsaw*
ul. Pasteura 5, 02-093 Warszawa,Poland
[c]*National Centre for Nuclear Research*
ul. Andrzeja Sołtana 7, 05-400 Otwock, Świerk, Poland

E-mail: wzab@ise.pw.edu.pl

Abstract: The Overlap Muon Track Finder (OMTF) is the new system developed during the upgrade of the CMS experiment which includes the upgrade of its Level-1 trigger. It uses the novelty approach to finding muon candidates basing on data received from three types of detectors: RPC, DT and CSC. The upgrade of the trigger system requires also upgrade of the associated Data Acquisition (DAQ) system. The OMTF DAQ transmits the data from the connected detectors that were the basis for the Level-1 trigger decision. To increase its diagnostic potential, it may also transmit the data from a few bunch crossings (BXs) preceding or following the BX, in which the L1 trigger was generated. The paper describes the technical concepts and solutions used in the OMTF DAQ system. The system is still under development. However, it successfully passed the first tests.

Keywords: Data acquisition concepts, Data acquisition circuits, Digital electronic circuits, Trigger concepts and systems

---

[1] Corresponding author

Contents

1 Introduction

Compact Muon Solenoid (CMS) is one of the experiments at the Large Hadron Collider (LHC) at CERN. After the successful operation in years 2010-2013, which resulted in the discovery of Higgs boson, it has been undergone the upgrade of its trigger, including the Level-1 muon trigger [1]. In the barrel-endcap transition region it is possible to combine signals from 3 types of muon detectors - Resistive Plate Chambers (RPC), Drift Tubes (DT) and Cathode Strip Chambers (CSC). The Overlap Muon Track Finder [2] (OMTF) is a dedicated electronic system analyzing the data received from those detectors and finding trigger muon candidates. The result of the OMTF processing is transferred to the CMS Level-1 Global Muon Trigger. To monitor the operation of the OMTF, it is important to record the data which were the basis for the trigger decision. This task must be performed by the Data Acquisition (OMTF DAQ) system.

The OMTF trigger uses six OMTF processors in each overlap area[1]. The OMTF DAQ system in a single board must be able to accept the data from 24 RPC links, 35 CSC links, and 6 DT links.

---

[1] There are two overlap areas, one on each end of the detector.

2 OMTF DAQ requirements

2.1 Data triggering

The detectors of the CMS experiment deliver the hit data after every crossing of the LHC proton or heavy ion bunches (BX). The CMS Level-1 Trigger analyses the information received from different triggers (including the OMTF) and generates the trigger decision (Level-1 Accept - L1A) for each potentially interesting BX ("triggered BX"). In principle, the OMTF DAQ must record only the data originating from the triggered BXes. However, to better monitor the operation of the trigger, it is desirable to record the data from a few BXes before and after the triggered BX. Those neighboring BXes together with triggered BX are denoted as "recorded BXes" (RBX). The data from RBXes associated with the particular occurrence of L1A create the “event”. Currently, it is assumed that it should be possible to record up to 8 RBXes - up to 3 before the L1A and up to 4 after it (the number of RBXes within these limits may be selected at the configuration time). Each RBX in the event is given its “short relative BX number” (SBXN) describing its position with respect to the L1A. The SBXN is a 3-bit number from 0 to 7. The triggered BX has always SBXN=3 (see figure 1). The allowed frequency of L1A is defined by the “trigger rules”. The currently used trigger rules state that there should be “no more than 1 trigger in 3 BXes, 2 in 25, 3 in 100 and 4 in 240” [3]. That means, however, that a single BX may belong to two events simultaneously, if two L1As are sufficiently near and if the selected number of recorded BXes in the event is sufficiently high. One example of such situation is shown in figure 2. That feature affects the implementation of the OMTF DAQ system, described in sections 3.1 and 3.5.

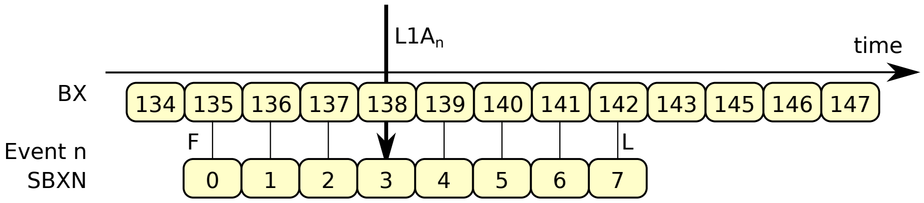

Figure 1. Definition of “recorded BXes” in a simple case. The first and last BXes in the event are marked with special atributes (’F’ and ’L’ respectively) used later on in the event building process. The short relative BX number (SBXN) describes the position of the BX in respect to the triggered BX (which has always SBXN=3).

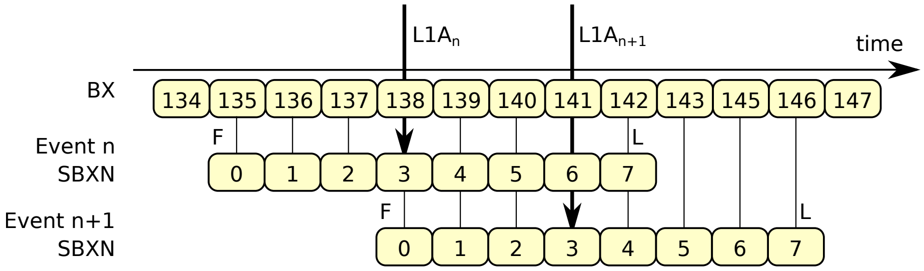

Figure 2. Definition of “recorded BXes” in a complex case with some BXes belonging to two events sinmultaneously. It should be possible to assign each BX up to two sets of attributes (F, L and SBXN) associated with different L1As

2.2 DAQ hardware platform

The recommended hardware platform for upgraded CMS trigger is μTCA. The OMTF trigger uses the MTF7 board [4, 5] containing two Xilinx FPGA chips: XC7VX690T and XC7K70. The OMTF trigger algorithm occupies a significant part of the XC7VX690T chip [6] but leaves sufficient amount of resources to allow implementing the OMTF DAQ system. That design allows reusing of certain blocks (e.g., link inputs, clock, and trigger inputs) both for the OMTF trigger and OMTF DAQ (see figure 3). The single μTCA crate may contain up to 6 MTF7 boards, that transmit the DAQ data to the standard AMC13 board [7] which is the DAQ board dedicated for physics experiments. Transmission of the DAQ data from the MTF7 board to the AMC13 board is handled by a dedicated IP core[2] [8].

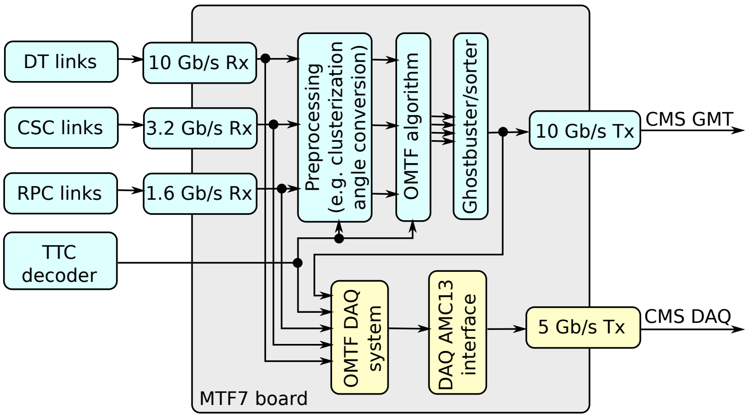

Figure 3. Block diagram of the OMTF firmware implemented in the MTF7 board, containing both the OMTF trigger and OMTF DAQ.

2.3 OMTF DAQ data format

The DAQ data format required by the AMC13 backplane link core (the "AMC to AMC13 Data Format") is described in [9]. The data are sent as so-called "event fragments" consisting of 64-bit words. Each event fragment contains two header words at the beginning and one trailer word at the end. The payload words are placed between the header and the trailer. The maximum length of the event fragment is 2^{20} words.

The "AMC to AMC13 Data Format" requires that a single event fragment is transmitted for each L1A. It is not possible to send the data from different detectors in separate event fragments. Therefore, the OMTF data format must allow assigning each data word to the particular detector. The RPC DAQ system used in the first run of LHC [10] used the sophisticated contextual data format, with the payload consisting of variable-length chunks, where the meaning of the word depended on the preceding subheader or separator words. That format appeared to be difficult to decode in the software. To avoid that, the OMTF DAQ format has been designed as context-free. Each payload word should contain all information necessary for its interpretation (of course together

---

[2] The original IP core had to be slightly modified due to non-standard connection of μTCA backplane multigigabit links in the MTF7 board. The modified version with the link receiver moved to the XC7K70 chip was provided by the developers of the MTF7 board.

with the event fragment header). Except for the hit data from three detectors, the event fragment should contain additional information like the output of the OMTF trigger algorithm and the version of the firmware. Therefore, the 4 most significant bits of the payload word are used to describe the type of the word[3]. Additionally, all payload words except the firmware version word contain the 3-bit short recorded BX number (SBXN). Therefore, only the 57 bits remain available for the type-specific data. The CSC and DT chambers may deliver up to 2 hits in a single BX. Those hits are transmitted as separate payload words. The RPC chamber may transmit up to 8 data frames a single BX. Each RPC data frame is 16-bits long. Therefore, it is possible to place up to 3 such frames together with the link number in the RPC payload word. The finally chosen OMTF DAQ data format is shown in figure 4.

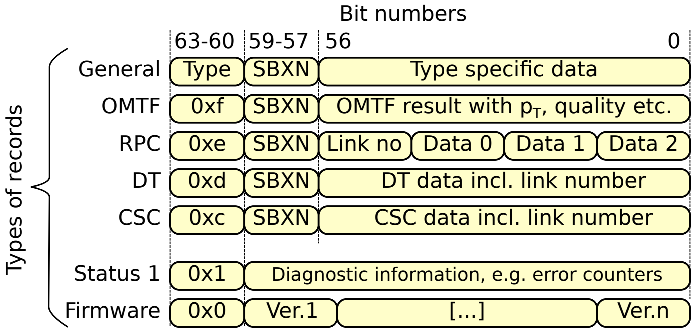

Figure 4. Different types of the payload words used in the OMTF DAQ system. The type of the word is defined by the 4 most significant bits.

3 OMTF DAQ Implementation

The block diagram of the OMTF DAQ system is shown in figure 5. On the input of the OMTF DAQ, the data from different input links are time aligned with the L1A signal using the dedicated shift registers.

3.1 Generation of RBX descriptors

The RBX descriptor is a data structure describing the recorded BX, that may belong to one or two events. It stores the 12-bit absolute BX number, and two sets of attributes describing the possible association of the RBX with one event. Each set contains:

- three 1-bit flags: 'T' (Triggered - set if the particular set contains valid event data), 'F' (First - set when this is the first RBX in the event), and 'L' (Last - set when this is the last RBX in the event).
- The 3-bit SBXN describing the position of the RBX in the event (see section 2.1 and figures 1, and 2).
- The 12-bit number of the BX in which the corresponding L1A occurred.

---

[3] Currently only 5 different types are used, so 3 bits should be sufficient for encoding of type. However, one additional bits has beed reserved for possible future extensions (e.g. additional diagnostic information)

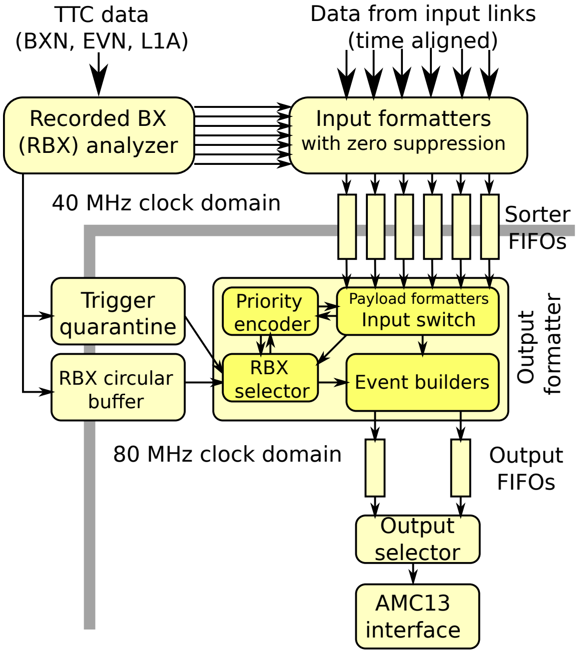

Figure 5. The block diagram of the OMTF DAQ system.

- The 24-bit absolute number of the event.

The RBX descriptors are created in the "RBX analyzer" block. Its main part is a shift register storing the RBX descriptors. It is shifted after every BX, and the first stage is initialized with the empty descriptor. When the L1A is asserted in the particular BX, the associated RBX descriptors are filled in the "RBX analyzer" block. The first set of attributes is filled for every even occurrence of L1A (event) and the second set is filled for every odd event. This solution ensures that in the case of "overlapping events" (as shown in figure 2) associations with both L1As are correctly stored in the RBX descriptor. When the filled RBX descriptor leaves the "RBX analyzer" it is stored in the RBX circular buffer. Its address is used as its identifier. That identifier is delivered to the block responsible for the selection of the input data.

3.2 Handling of input data

The part of the system responsible for the handling of input data works at the clock frequency of 40 MHz[4]. The input data are delayed so, that they are synchronized with the RBX descriptors leaving the "RBX analyzer". Whenever the nonempty RBX descriptor appears on the output of the "RBX analyzer" the input data originating from the corresponding BX should be written into the "sorter queues". This is the task of the so-called "input formatter". It checks if the data on the particular input is correct and non-empty (it performs the *zero suppression*), and transfer the data supplemented with the RBX descriptor identifier to the sorter queue.

However, in some special cases the "input formatter" must perform additional actions. The RPC link may deliver a few data frames originating from the same BX. In such a case some of those

---

[4] In fact this is the LHC clock frequency slightly above 40 MHz

data are delivered with a non-zero delay [11]. Therefore, the RPC input formatter must reassemble those data, packing up to three 16-bit hit data in a sorter queue word.

Other special cases are the CSC inputs. Each CSC chamber may deliver data of 2 hits per BX. Some parts of those data are common for those hits. To provide uniform data handling, each CSC input formatter splits those input data into two separate words (one for each hit) transferred to two independent sorter queues.

3.3 Reception of data from sorter queues

The part of the system located after the sorter queues works at the clock frequency of 80 MHz[5]. The sorter queues allow grouping the data originating from the same RBX delivered by different links. The "RBX selector" extracts the available RBX descriptor from the RBX circular buffer. It is important that the amount of time after generation of that descriptor be long enough that the associated input data could reach the output of the sorter queues. Simple waiting for the required time would impair the performance of the system in the case of high occupancy. Therefore a special "trigger quarantine" mechanism has been implemented. It is a plain shift register with multiple comparators. When the RBX descriptor is written into the circular buffer, its ID is written into that shift register and gets shifted after every BX. When the RBX descriptor is retrieved from the circular buffer, the comparators check that its ID is still in the quarantine shift register. If yes, the system waits until it is shifted out. After the quarantine check the RBX is fed to the priority encoder, which finds the first sorter queue providing the data from that RBX. The data from such queue are transmitted via the appropriate source-specific "payload formatter" and "input switch" to the general "Event builder". All those blocks are located in the "Output formatter". When the sorter queue has no more data for that RBX, the priority encoder outputs the number of the next queue providing such data. If there is no queue delivering the requested data, the "RBX selector" extracts the next RBX descriptor from the circular buffer and repeats the above procedure.

Due to the high number of the sorter queues, the priority encoder was designed as a two-stage block. The first stage is working with the inverted clock signal and analyzes groups of sorter queues consisting of 8-queues. In each group, the number of the first queue delivering the requested RBX data ("active queue") is found, or the information about the unavailability of those data is generated. The second stage uses the normal clock polarization. It finds the first group reporting the availability of the RBX data and outputs the number of that group concatenated with the 3-bit number of its first active queue. If there is no such group, the information about the unavailability of requested RBX data is generated[6].

3.4 Building of events

The main purpose of the "Event builder" is to encapsulate the RBX data into the event fragments described in section 2.3. The "Event builder" receives the descriptor of the currently processed RBX and the payload data. Whenever handling of the new RBX is started, the "Event builder" checks its "First" and "Last" attributes. If the RBX is the "First" one, the event fragment header described in [9] is inserted into the output data stream before the payload data. If the RBX is the

---

[5] In fact this is the doubled LHC clock frequency slightly above 80 MHz

[6] The versatile, configurable version of the described priority encoder is available at [12]

"Last" one, the event fragment trailer is inserted into the output data stream after the payload data. The "Event builder" also calculates the length of the generated event fragment, that must be written into the event fragment trailer.

3.5 Handling of overlapping events

As it was described in section 2.1 (see figure 2), it is possible that the same RBX may belong to two events associated with different L1As. That may create certain problems in the generation of event fragments. After transferring the data from the last RBX of the first event fragment (RBX 142 in figure 2), we should get back to the first RBX of the second event fragment (RBX 138 in figure 2). However, the OMTF DAQ processes the DAQ data as a stream, and it is not possible to return to already processed RBXes. The problem has been solved by using two independent sections consisting of the "Event builder" and the "output queue". The first section builds the even events, and the second section builds the odd events. The belonging of the particular RBX to one or two consecutive events is described by its attributes set in the "RBX analyzer".

Consecutive events are combined into a single data stream by the output collector, which is a plain multiplexer switching the active input after the event fragment trailer is received from the currently selected input. That design allows the simple handling of overlapping events. However, the designer must be aware of one significant limitation. Both output queues must always fit the whole overlapping part of event fragments. [7]

Finally the data from the "Output selector" are delivered to the "AMC13 backplane link core", that transmits them to the AMC13 board.

3.6 Implementation of the backpressure

The OMTF DAQ core monitors the occupancy of all queues. If the AMC13 link is not able to receive data, the output queues are filled and the operation of "Output formatters" is suspended. The data are then accumulating in the sorter queues. The backpressure signals are generated if the sorter queues are filled above the predefined allowable thresholds. Those signals are transmitted by the AMC13 to the Trigger Control and Distribution System (TCDS) and used to block the generation of next L1As. The OMTF DAQ core monitors also the occupancy of the RBX circular buffer. If the L1As are generated at the too high rate, that buffer may be filled with the non-serviced yet RBX descriptors even if the triggered data do not fill the sorter queues. Therefore the backpressure signals are also generated for the too high occupation of the RBX circular buffer.

4 Flexibility of the design

The presented DAQ system may be easily extended. Currently implemented features provide support both for data sources producing a single word per link in each BX and for sources producing multiple

---

[7] If the overlapping part is too big, the queue which is currently used for building the latter event fragment will fill up and block writing of next words by the EB. Therefore the queue handling the former event fragment will never receive the trailer. As the result the output switch will not start to empty the filled queue and the system will lock-up. The problem may be worked-around by dropping the superfluous payload data at the input of the latter queue, and marking the event as corrupted. However the preferred solution is to use the output queues able to fit the maximum possible length of the event fragment.

	Slice LUTs available 433200		Slice Registers available 866400		Block RAM tiles available 1470	
Firmware block	Used units	Percentage use	Used units	Percentage use	Used units	Percentage use
The whole OMTF firmware	294134	67.90%	279089	32.21%	1053.5	71.67%
The OMTF DAQ block	18381	4.24%	26833	3.10%	125	8.50%
The RPC input alignment block	482	0.11%	473	0.05%	6.5	0.44%
The CSC input alignment block	1182	0.27%	2257	0.26%	31.5	2.14%

Table 1. Occupancy of the XC7VX690T FPGA in a single MTF7 board

data words in each BX, and delivering them with a certain delay (like RPC). Adding a new data source requires:

- Implementing the filter deciding whether the particular data word originating from a recorded BX should be transmitted to DAQ (zero-suppression).
- Implementing the formatter, that packs the received data into the sorter queue, and converts the data on the output of the sorter queue into the 64-bit words in OMTF DAQ data format, that may be used as payload in the event.

The number of handled RPC, CSC, and DT inputs is parametrized and may be easily changed.

The design may also be adjusted to possible changes in operating conditions of the experiment. For example, if the number of recorded BXes or the trigger rules are modified so that a single RBX may belong to more than two events simultaneously, it is possible to modify the RBX descriptor so that it contains the appropriate number of sets of attributes. The number of “Event builders” and output queues should also be increased accordingly. Thanks to the parametrized, high-level VHDL implementation of the OMTF DAQ the necessary modifications may be introduced easily.

Of course the scalability of the design is limited by the resources available in the FPGA.

5 Results & conclusions

All features of the OMTF DAQ have been tested in simulations using the GHDL simulator. The tests in the hardware are still being performed. At the time of writing transmission of the RPC data has been tested both with the data injected from the diagnostic pulser embedded in the OMTF firmware and with the real data from RPC chambers. Transmission of the CSC data is tested with the pulser data. The tests have proven correct operation of the system (however some minor bugs were discovered and corrected). The system was able to work correctly at L1A trigger rates up to 120 kHz. Transmission of the DT data in the real hardware is still under development.

The full OMTF firmware, containing both OMTF trigger and OMTF DAQ (without support for DT links yet) was successfully synthesized for the XC7VX690T chip available in the MTF7 board. The chip occupancy is given in table 1.

The reported FPGA occupancy should allow implementation of the DT links, leaving sufficient amount of resources to avoid timing problems.

The concepts and solutions used in the presented design may be useful for developers of other triggered synchronous data acquisition systems with back pressure and zero suppression, particularly for the High Energy Physics experiments.

Acknowledgments

Work supported by National Science Centre, Poland, grant 2013/11/B/ST2/04202 and 2015/19/B/ST2/02861.

References

[1] CMS Collaboration, *CMS Technical Design Report for the Level-1 Trigger Upgrade*. No. CERN-LHCC-2013-011. CMS-TDR-12 in Technical Design Report CMS. Geneva, Jun, 2013.

[2] M. Bluj et al., *From the Physical Model to the Electronic System — OMTF Trigger for CMS*, *Acta Physica Polonica B Proceedings Supplement* **9** (2016), no. 2 181.

[3] CMS Collaboration, *Performance of the CMS Level-1 trigger during commissioning with cosmic ray muons and LHC beams*, *Journal of Instrumentation* **5** (Mar., 2010) T03002–T03002.

[4] D. Acosta et al., *The cms modular track finder boards, mtf6 and mtf7*, *Journal of Instrumentation* **8** (2013), no. 12 C12034.

[5] A. Byszuk et al., *OMTF firmware overview*, *Proc. SPIE* **9662** (2015) 966241–966241–6.

[6] W. Zabolotny and A. Byszuk, *Algorithm and implementation of muon trigger and data transmission system for barrel-endcap overlap region of the CMS detector*, *Journal of Instrumentation* **11** (Mar., 2016) C03004–C03004.

[7] E. Hazen, A. Heister, C. Hill, J. Rohlf, S. X. Wu, and D. Zou, *The amc13xg: a new generation clock/timing/daq module for cms microtca*, *Journal of Instrumentation* **8** (2013), no. 12 C12036.

[8] E.Hazen, *AMC13 to AMC backplane link*, February, 2016. `http://bucms.bu.edu/twiki/bin/view/BUCMSPublic/AMC13BackplaneLink` .

[9] E.Hazen, *AMC13 DAQ facilities*, September, 2015. `http://ohm.bu.edu/~hazen/CMS/AMC13/UpdatedDAQPath.pdf` .

[10] W. M. Zabolotny et al., *Implementation of the data acquisition system for the Resistive Plate Chamber pattern comparator muon trigger in the CMS experiment*, *Measurement Science and Technology* **18** (Aug., 2007) 2456–2464.

[11] M. Gorski, I. Kudla, and K. Pozniak, *Resistive plate chamber (rpc) based muon trigger system for the {CMS} experiment – data compression/decompression system*, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **419** (1998), no. 2–3 701 – 706.

[12] W.Zabolotny, *Two-phase configurable priority encoder in VHDL*, October, 2016. `https://github.com/wzab/wzab-hdl-library/tree/master/2stage_prior_enc` .