Preprint typeset in JINST style - HYPER VERSION

A demonstration of a Time Multiplexed Trigger for the CMS experiment

R. Frazier[a], S. Fayer[b], G. Hall[b], C. Hunt[b], G. Iles[b]*, D. Newbold[a] and A. Rose[b]

[a] *University of Bristol,*
H.H. Wills Physics Laboratory, Tyndall Avenue, Bristol BS8 1TL, UK
[b] *Imperial College London,*
Blackett Laboratory, Prince Consort Road, London SW7 2BW, UK

E-mail: g.iles@imperial.ac.uk

ABSTRACT: A novel approach to first-level hardware triggering in the LHC experiments has been studied and a prototype system built. Calorimeter trigger primitive data (~5 Tb/s) are re-organised and time-multiplexed so that a single processing node (FPGA) may access the data corresponding to the entire detector for a given bunch crossing. This provides maximal flexibility in the construction of new trigger algorithms, which will be an important factor in ensuring adequate trigger performance at the very high levels of background expected at the upgraded LHC.

A test system that incorporates all the key technologies for a final system and demonstrates the time-multiplexing and algorithm performance is presented.

KEYWORDS: CMS; trigger; time-multiplexed; AMC; MicroTCA.

*Corresponding author.

Contents

1. Introduction

The CMS detector [1] can trigger readout out the experiment at up to 100 kHz to extract interesting physics events from a collision rate of 40 MHz. Collision data are stored on the detector in pipeline memories in the intervening period. The detector pipelines have a maximum depth of 160 bunch crossings and thus the latency of the Level-1 trigger is restricted to 4.0 μs.

The present trigger system has been built from a series of electronic cards which all view a small portion of the detector at the 40 MHz collision rate. These share data along their boundaries so that they have sufficient overlap for identifying physics objects. Building the trigger was technically very demanding because of the large data rate (e.g. ~5 Tb/s for the Calorimeter Trigger), the technology available at the time and the latency constraint. These requirements drove a design that will provide a good performance up to the LHC design luminosity of 10^{-34}cm^{-2}s^{-1}. The performance will degrade as the luminosity increases to $2\text{x}10^{-34}$cm^{-2}s^{-1} with the planned LHC phase I upgrade in 2018 and degrade further as the luminosity reaches $5\text{x } 10^{-34}$cm^{-2}s^{-1} with the LHC phase II upgrade in 2022.

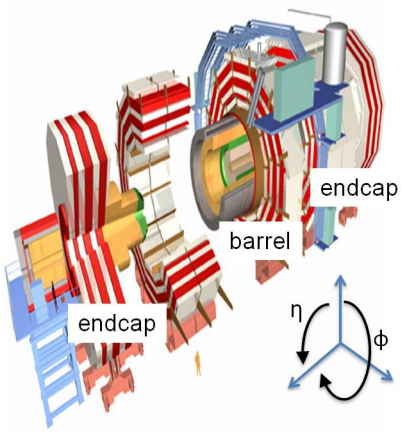

Figure 1. The geometry of the CMS experiment. The ϕ and η dimensions are unrolled to create a grid that spans 56 towers in η and 72 towers in ϕ.

The Calorimeter Trigger Primitives (TPGs) that drive the trigger decision span 56 towers in η and 72 in ϕ (figure 1) excluding the Forward Hadronic Calorimeter (HF). The TPGs from the Electromagnetic Calorimeter (ECAL) and Hadronic Calorimeter (HCAL) have a size of 9 bits per tower and are comprised of 8 bits compressed energy and 1 bit of feature information. The links currently include a Hamming code; however, if that were replaced by a simple CRC at the end of each orbit they could supply up to 12 bits of information per tower per calorimeter (i.e. 24 bits in total for both ECAL & HCAL). There are some subtleties with boundaries within ECAL, but they are not discussed here. Currently, the tower information is supplied on 1.2 Gb/s copper links that cover 2 towers; however, the upgrade will concentrate 8 towers (2 in η by 4 in ϕ) onto a 4.8 Gb/s optical link to better match modern FPGA I/O capability.

The trigger upgrade relies on both the phenomenal increase in FPGA logic and serial link capabilities over the last decade. Serial links are essential for enabling the trigger design to process the data with just 3 stages in a conventional trigger (share data, fine processing, coarse processing) and 2 in a time multiplexed design (time multiplex, process). This is critical given the large latency of a serialiser-deserialiser (SerDes) block (i.e. ~100 ns). It is only by massively reducing the number of SerDes stages (i.e. also cards) in the system that the use of serial links becomes feasible. It is the increase in logic, coupled with the ability to provide a reasonable overlap with neighbouring processing nodes that provide the opportunity to greatly enhance the capability of the trigger by fully utilising the ECAL & HCAL TPGs generated at 40 MHz.

1.1 Conventional Trigger

Designs already exist that would simply upgrade the existing structure with more modern FPGA technology [2]. For example, it is possible to build a Xilinx Virtex 5 design with 24x 5 Gb/s input links and 14x 5 Gb/s sharing links that would span 8 towers in η and 12 in ϕ with a single, full resolution, tower of overlap. It would require 48 cards. Upgrading to a Virtex 7 with 48x 5 Gb/s input links and 18x 10 Gb/s sharing links would allow the design to span 16 towers in η and 12 in ϕ with two, full resolution, towers of overlap. It would require 24 cards and reduce the percentage of links used for sharing data from 37% to 27%.

In both the Virtex 5 and 7 designs the overlap area is limited to one or two towers at full resolution; however, a jet is typically 8 towers wide and thus jet reconstruction would have to occur at a second stage from some form of reduced information. The current Global Calorimeter Trigger jet clustering algorithm operates at a much coarser resolution of 4x4 towers, formed from the sum of ECAL and HCAL energies. Boundary sharing requirements are reduced further by pre-clustering jet fragments.

1.2 Time Multiplexed Trigger

In order to move a away from boundary constraints a novel approach was proposed [3] [4]. For a conventional trigger the area spanned by an FPGA is normally set by the IO available (i.e. number of links and link speed). To overcome this limitation in the new design the TPG data are time-multiplexed so that the data from a particular bunch crossing are re-transmitted over ~10 bx (bunch crossing). In this way the TPG data for the entire calorimeter can be passed through a single FPGA, although the FPGA takes 10 bx to process the data. It is therefore necessary to have 10 FPGAs

running in a round robin scheduling manner to handle every bunch crossing. Although conceived for the Calorimeter Trigger this approach is very well suited to any application that requires a large overlap of data. Note, however, that the overlap will still be limited, but this time by the logic resources within the FPGA and latency. For example, while it is conceivable that a jet finding algorithm would search over 8x8 towers in ϕ and η, it may not be possible for it to look over 16x16 towers. The precise limit would be dependent on the algorithm complexity and latency available.

The baseline design (figure 2) consists of 28(+4) Pre Processors that time multiplex the data from 36x 4.8 Gb/s ECAL & HCAL links that form a ring 2 towers wide in η and all 72 towers in ϕ. The extra 4 Pre Processor cards are required to handle subtleties at boundaries within ECAL and to include HF. The details of this are not described here. Although the time multiplexing period is 10 bx only 9.0 bx + 0.5 bx (optional) are used for data. The rest is used for SerDes alignment commas and a CRC check.

Data are transmitted to 10 Main Processors that each accept 2 links from each Pre Processor (i.e. nominal 64 links). The card will be designed to accept up to 72 links for additional flexibility. There are 2 spare MP cards that can either be used as redundant cards that can be swapped in "on-the-fly", or used for testing new Main Processor firmware in parallel to the existing system. The results are transmitted to the Global Trigger on 2x 10 Gb/s links. Quantities that are formed from the entire event (e.g. total energy) must be transmitted at the end of the event. The current system (i.e. time multiplexing over 10 bx) foresees 192 bits for these quantities and 3456 bits for all electron and jet candidates (i.e. ~200x 16 bit candidates versus 20 in the current system).

In order to minimise latency, e/γ and jet candidates are transmitted as soon as they are found by the processing algorithm, as it steps around the ring of ϕ. At first glance it would seem that the GT might be swamped by all this data; however, if it wished to reduce this to the 4 electron, 4 iso-electrons, 4 tau jets, 4 central jets and 4 forward jets, as in the current system, it could do this easily and with sub-bx latency. It can achieve this because the data arrive over many bx. Imagine that the GT receives 4 of each type of candidate for every bx, for a period of 9 bx. In this scenario the GT would operate 8 to 4 sorts for each new bx of data. The few new candidates per clock cycle ensure that the sort will be very efficient and require an additional latency which is well within the sub-bx range.

2. Demonstrator System

The demonstrator system is built around a Vadatech VT892 MicroTCA crate [5] that is based on the VT891 chassis, but modified for the vertical airflow that is present in a standard CMS rack. It has a dual star backplane with 12 double width, full height AMC slots. Full height slots were chosen so that large heatsinks could be used and to provide flexibility for optical components (i.e. both for component height and fibre optic routing over the PCB). The desire for double width cards was simply because the real estate on a single width card was not sufficient for most applications (i.e. typically a large FPGA, optical components, power supplies, etc).

A NAT Europe MCH (Micro TCA Carrier Hub) [6] provides communication via GbE and IPMI (Intelligent Platform Management Interface). IPMI defines a standardized, abstracted, message based management system. It also defines standardized records for describing platform management devices and their characteristics.

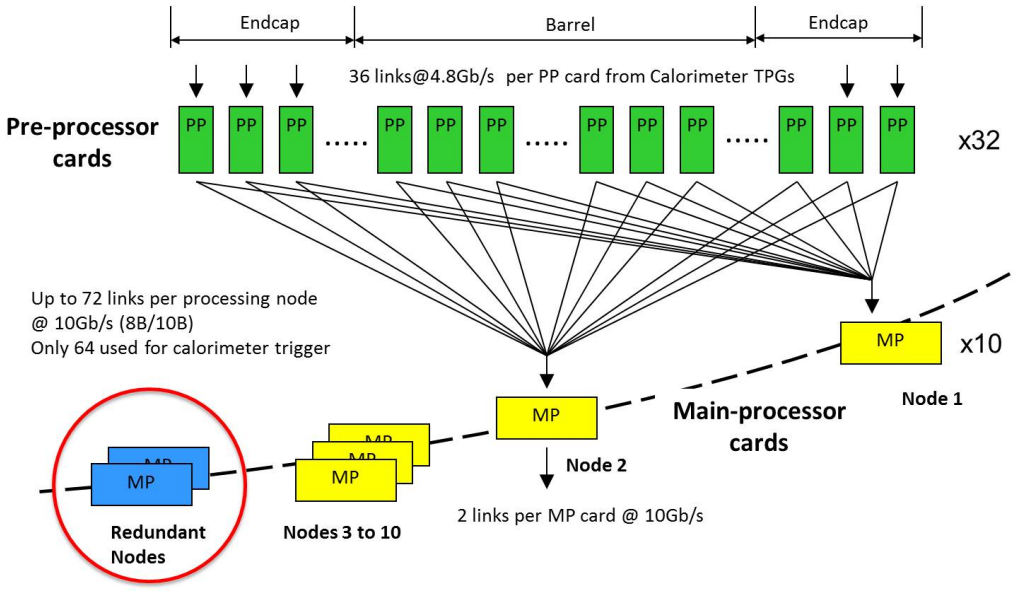

Figure 2. There are 32 Pre Processor cards that span the length of the detector. These receive ECAL & HCAL data and time multiplex it so that each bunch crossing is sent to one of 10 Main Processor nodes. There are 2 spare Main-Processor nodes available.

In normal operation, an AMC, located in the redundant MCH slot for connectivity purposes and referred to as AMC13, would supply the LHC clock, fast control/feedback and serve as a DAQ hub [7]. Prototypes were still under test when the demonstration system was commissioned and thus the 40 MHz clock was distributed to AMCs via a front panel input on the MCH and fast control was distributed via external cables. Fast feedback was not required and DAQ was via the local GbE.

2.1 Processing Card: MINI-T5-R2

The MINI-T5-R2 processing card contains a large Xilinx Virtex 5 FPGA (TX240T) and has 32 input / 20 output optical links operating at 5 Gb/s. The details of the card are described elsewhere [8] except that in the R2 revision the dual 80 way Samtec headers have been removed and replaced with dual 72 Mb QDRII SRAMs, each with separate 18 bit wide R/W ports operating at 500 Mb/s per pin. The choice of QDR rather than DDR RAM was motivated by the requirement that trigger may need random access to the memory with fixed latency (i.e. waiting for a DDR refresh cycle to complete would not be acceptable). The capability to store multiple firmware configuration images and to warm boot to them has been added via the Virtex 5 ICAP interface. A safe image is stored at the base of the PROM in case of a configuration failure. It is activated by power cycling the card.

2.2 Ethernet Communication: IPbus

The primary control path on MicroTCA is Gb/s Ethernet. The dominant method of communication between Ethernet based devices is TCP/IP, but to fully exploit the Gb/s bandwidth available on port-0 requires a powerful CPU, associated peripherals (i.e. RAM and flash memory) and a TCP/IP

software stack. While this capability may be required for some systems for many ~100 Mb/s is sufficient, particularly given that 12 cards in a crate share a single GbE connection and in CMS Ethernet would just be for control. Trigger, readout and DAQ will be served by their own dedicated links. An alternative is a soft core CPU located within an FPGA; however, these typically only achieve data rates of ~10 Mb/s and add complexity. For CMS, there is also the requirement that system configurations are stored in some permanent manner (usually a database), quickly retrieved and used. Lastly, the latency of an Ethernet transaction is substantial (~1ms), which must be accommodated.

Consequently, there was a desire to develop an IP core that; was reliable and relatively simple; had reasonable bandwidth (~100 Mb/s), particularly for register access; was portable from one card to the next across different manufacturers and their different devices; and was not too onerous to implement. Lastly the design should be modular so that it could easily be ported from one underlying transport mechanism to another without too much effort and without changing the user interface.

This led to the development of IPbus, a protocol for placing many transactions (i.e. Read/ Write, Bulk Read/Write, etc) into a single packet with a user interface that explicitly defined when a packet was transmitted. While making the programming more complicated it enables the programmer to write his code in a sensible manner for a packet based communication medium with a large latency. For example, imagine the programmer wishes to check the number of CRC errors on incoming detector links. In this instance the programmer would read back all CRC error counters in a single packet and then check them, rather than using a packet to read back the CRC error counter for the first link, checking it and then repeating this for each link. IPbus is therefore a protocol for the concatenation of multiple commands.

IPbus is currently implemented as a firmware core that uses UDP/IP, but there is no reason why it could not use TCP/IP. In the latter case the segmentation of large packets into smaller ones would not be performed by the IPbus software suite, but by the TCP/IP protocol itself. The ability of TCP/IP to have multiple small packets in flight simultaneously would allow a much greater throughput, whereas in UDP the arrival of each individual packet is verified before the next one is transmitted.

The IPbus software suite consists of MicroHAL, a C++ user interface that can model and access the firmware in a hierarchical fashion; and the ControlHub, a single point of access to the hardware that is written in Erlang. Erlang scales transparently across multiple CPU cores and automatically splits large read/write requests into appropriately sized Ethernet packets. At present the core supports standard Ethernet 1.5 kB packets (i.e. the IEEE 802 standard) and achieves a throughput of 40 Mb/s. The performance reaches 150+ Mb/s if jumbo frames (up to 9 kB) are used. The system has been tested on a system comprising 3 MicroHAL PCs, 1 Control Hub PC and 20 IPbus clients. Approximately 1 in 189 million packets are lost without the UDP retry mechanism which is currently being implemented. All unnecessary network protocols were switched off (spanning tree, etc).

IPbus also provides a back door to the main FPGA bus via IPMI commands issued to the MMC controller. This is useful for setting up Ethernet (e.g. assigning IP addresses) and as fall back in case of Ethernet issues. The core is designed primarily for fairly powerful FPGAs, but will fit within a small Spartan FPGA (e.g. Xilinx SP601 development board, Spartan 6 XC6LX16-CS324

with resource usage of 7% FFs, 18% LUTs and 25% BRAM).

2.3 Algorithms

The demonstrator system (figure 3) simulates approximately 1/4 of the input to the CMS Calorimeter Trigger (~1 Tb/s). It achieves this with 6 MINI-T5-R2 cards, of which 4 are used to simulate 24 Pre-Processors and 2 are used as Main-Processors. It is possible to simulate 6 Pre-Processors in a single MINI-T5-R2 card because in the demonstrator system there are no input links from HCAL & ECAL and there are 2, rather than 12, Main Processors (i.e. serial links that would have been used for the other 10 Main Processors can be reused for the additional Pre Processors). In each Pre-Processor, BRAMs (32 kbit dual port memory) are filled with patterns and read out simultaneously. The data are driven through the time multiplexer in each Pre Processor and for 2 of the 12 Main Processors the data are directed to a serial link output.

A patch panel connects the 24 Pre Processors distributed within the 4 cards to the 2 Main Processors. The events are aligned and passed to the algorithm. DAQ capture blocks allow the events to be captured both before and after the algorithm on receipt of a trigger. These operate very differently from conventional pipeline memories in CMS, in which the data are retrieved upon receipt of a trigger that arrives a fixed period after a Level-1 trigger. In these pipelines the trigger bx is calculated and the logic searches for the event within the memory system.

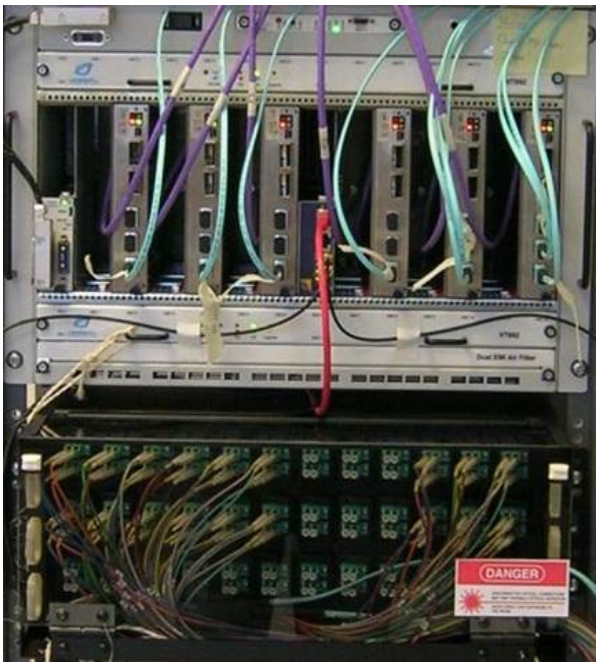

Figure 3. The demonstrator system. The 4 MINI-T5 cards each containing 6 Pre Processors are shown on the left. They drive 24 links to each of the 2 Main Processors which are shown on the right. The MCH is visible in the bottom centre of the crate.

The demonstrator system differs from the final system only slightly. Data are time-multiplexed across η (i.e. from end to end of CMS) rather than in ϕ (i.e. rings around CMS), which is the baseline for the final system, but the concept remains the same. The number of bits per tower is compressed from the maximum 24 bits available from the proposed 4.8 Gb/s ECAL & HCAL links to 16 bits; however, this is still significantly more than the 10 bits initially suggested for an upgraded Calorimeter Trigger.

The detector is simulated by loading FPGA memories (BRAMs) with detector data and then reading these out.

The e/γ algorithm implemented [2] spans 24 towers in ϕ and all of η, corresponding to 1/3 of CMS. Including all SerDes control, link alignment and the Ethernet interface the design uses 13% of LUTs and 15% of registers in the Virtex 5 TX240T FPGA. The final system target FPGA is the XC7VX690T, which has 6 times as many FFs and 3 times as many LUTs. Scaling the resource

usage from the demonstrator system to the final system indicates a resource usage of 17% for LUTs (the most critical resource) without the jet algorithm implemented.

3. Conclusion

A new approach to first-level triggering at LHC experiments has been demonstrated, using a prototype system based on time multiplexing of detector trigger data. This enables, in the case of the CMS detector, the trigger primitive data from the entire calorimeter for a specific bunch crossing (40 MHz) to be processed in a single processing node (FPGA). The data rate exceeds 5 Tb/s. This architecture avoids the data-sharing connections between processing nodes that occur in a conventional system in which each node operates on only a portion of the detector with limited overlap between nodes.

Furthermore, the trigger algorithms within each processing node have access to the full resolution of the trigger primitives data, providing maximum algorithm flexibility. The described system has been designed for the CMS Calorimeter, but because the system does not apply any data reduction until the main processing stage, it is completely generic and can be used for other applications. The system is essentially a very flexible 5 Tb/s real-time image processor operating with a latency of 1 µs, and as such has received interest from other fields (defence).

The system hardware is based on MicroTCA. Communication to the AMC cards is via IPbus; a Hardware Access Library (HAL) that provides robust access to the hardware via standard IP-over-Ethernet networks. Similarly to the hardware, the software follows a modular design approach. It exhibits efficient use of the network through automatic concatenation, packetisation and queuing of multiple commands and data streams; reliable access over unreliable network protocols via built in error detection and retry capability; and a simple interface for integration with hardware control applications. The software makes use of robust telecommunication technology (Erlang) to provide a scalable system that has been extensively tested at a realistic scale. The client is implemented directly in firmware without use of a soft or hard CPU, thus simplifying FPGA design and making the design very portable and easy to implement; it includes a simple SoC bus and controller which may be interfaced to a large variety of custom or open firmware cores.

The system resides in a standard, redundant, telecom MicroTCA crate. The primary slot contains a standard commercial MCH for Ethernet and IPMI. The redundancy is not used, but the connectivity of the redundant slot is exploited with an AMC13 card that provides an interface to the experiment.

A full implementation of the CMS first-level Calorimeter Trigger system will require the bandwidth capability of the newer Xilinx Virtex 7 series FPGAs to remain within the stringent latency constraints of CMS; however, the described prototype system, built around Xilinx Virtex 5 series FPGAs, demonstrates the core firmware and software functionality of a full vertical slice (i.e. time-multiplexing, algorithms, DAQ and de-multiplexing).

Acknowledgments

We would like to thank Sarah Greenwood (Imperial College) for layout of the MINI-T5 card and STFC for financial support.

References

[1] The CMS Collaboration, S Chatrchyan et al., *The CMS experiment at the CERN LHC*, 2008 *JINST* **3** S08004

[2] *CMS Regional Calorimeter Trigger Upgrade: Hardware and Firmware Proposals and Development* 2010 CMS-IN-2010-035.

[3] J. Jones., *CMS: A Future Trigger Architecture, CMS Upgrade Workshop, Fermi National Lab, USA, 28 - 30 Oct 2009*.

[4] G. Iles & A. Rose et al., *A Time-Multiplexed Calorimeter Trigger for CMS with Addendum*, 2011 CMS-IN-2011-008.

[5] http://vadatech.com

[6] http://www.nateurope.com

[7] http://www.amc13.info

[8] G. Iles et al., *A demonstrator for a level-1 trigger system based on MicroTCA technology and 5Gb/s optical links.*, TWEPP-10: Topical Workshop on Electronics for Particle Physics, Aachen, Germany, 20 - 24 Sep 2010