Online energy reconstruction on ARM for the ATLAS TileCal sROD co-processing unit

Mitchell A. Cox and Bruce Mellado

School of Physics, University of the Witwatersrand, Johannesburg 2050, South Africa

E-mail: mitchell.cox@students.wits.ac.za

Abstract. Modern Big Science projects such as the Large Hadron Collider at CERN generate enormous amounts of raw data which presents a serious computing challenge. After planned upgrades in 2022, the data output from the ATLAS Hadronic Tile Calorimeter (TileCal) will increase by 200 times to over 40 Tb/s. This increase requires more advanced processing on the raw data in order to harness a larger quantity of good quality physics data. An algorithm called Optimal Filtering (OF) is currently used in the TileCal front-end for online energy reconstruction of the digitised photo-multiplier tube signals and is currently implemented on Digital Signal Processors (DSPs) and Field Programmable Gate Arrays (FPGAs) which are difficult to program and are expensive. It is proposed that a cost-effective, high data throughput and general purpose Processing Unit (PU) can be developed by using several commodity ARM processors while maintaining minimal software design difficulty for the end-user. This PU could be used for a variety of high-level algorithms other than OF on the high data throughput raw data to combat the issue of out of time pile-up and for online data quality testing. OF and histogram algorithms have been implemented in C++ and several ARM platforms have been tested and shown to have good CPU to external I/O balance.

1. Introduction

Projects such as the Large Hadron Collider (LHC) generate enormous amounts of raw data which presents a serious computing challenge. After planned Phase-II upgrades in 2023, the raw data output from the ATLAS Hadronic Tile Calorimeter (TileCal) will increase by 200 times to over 40 Tb/s (Terabits/s) [1, 2]. It is infeasible to store this data for offline computation and so an online triggering system is used to reduce the quantity of data before storage.

The LHC Run 0, as of 2012 had a peak luminosity of 7.7×10^{33} cm^{-2}s^{-1} with a center of mass energy of $\sqrt{s} = 8$ TeV. The Phase-II upgrades in 2023 assume a maximum instantaneous luminosity of 7×10^{34} cm^{-2}s^{-1} [3].

The average number of interactions per bunch crossing also known as “in time pileup”, $\langle\mu\rangle$, can be calculated. As of 2012, $\langle\mu\rangle = 20.7$. This is assumed to increase to $\langle\mu\rangle = 200$ for Phase-II [3]. By increasing the amount and also the precision of the data read out from TileCal, more sophisticated algorithms may be used for energy reconstruction in a high pile-up environment.

The sROD (Super Read out Driver) is a FPGA-based device which is located in the back-end. It is the interface to the upgraded TileCal front-end read-out electronics and is responsible for its management and control. The sROD will become the new Level-0 trigger, performing energy reconstruction and then reducing the data rate from the 40 MHz bunch crossing rate to 500 kHz

[2, 3]. The upgraded Level-1 trigger will then reduce the data to 200 kHz as opposed to the current 100 kHz. A diagram of the upgraded TileCal read out architecture is shown in Figure 1.

Photomultiplier tubes (PMTs) are linked to layers of scintillator in TileCal by wavelength shifting optical fibres. The electrical signal that is produced by a PMT when a particle interacts with the scintillator is stretched, amplified and sampled with an analog to digital converter operating at 40 MSa/s (Million Samples per Second). Seven samples are taken of each pulse. Figure 2 shows an ideal pulse shape with arbitrary amplitude and phase shift with seven samples. From the figure it is clear that out of time pileup can dramatically influence the shape of the pulse. It is the job of the filtering algorithm to negate this. Optimal Filtering (OF) is currently used to perform energy reconstruction on the raw data [4].

The OF algorithm simply finds two dot products of these seven samples and two pre-calculated "weights" vectors. Each vector is highly specific and the result is the peak amplitude of the pulse as well as the phase shift of the pulse from the expected time.

Under higher pile up conditions OF loses accuracy and so a third calculation is done to estimate the quality factor of the results. If this quality factor is low then the raw data is stored for later offline processing with a more accurate algorithm.

Several new energy reconstruction algorithms are in development and still more are expected in years to come, each with advantages and disadvantages. A Matched Filter based technique shows similar performance to OF, and there is also a signal response deconvolution based technique shows promise [5, 6]. The deconvolution based technique cannot be implemented on the existing read out electronics due to lack of processing power.

An ARM System on Chip (SoC) based co-processing unit (PU) for the TileCal sROD is being developed [7]. ARM is an alternative CPU architecture to Intel x86 and is commonly used in mobile devices due to its energy efficiency and low cost. This PU can be used to monitor the raw data that passes through the sROD and to verify the sROD energy reconstruction data quality. Since the PU is based on general purpose processors, it is easy to implement new algorithms and functionality using a programming language such as C or even Java. A more ambitious use for the PU could be to update the energy reconstruction parameters used by the sROD in real-time or to take over energy reconstruction in its entirety.

The results of a C++ OF algorithm on ARM are presented in Section 2 and the implementation of a simple PCI-Express CPU interconnect and preliminary performance is given in Section 3. Section 4 concludes with a brief discussion of future work.

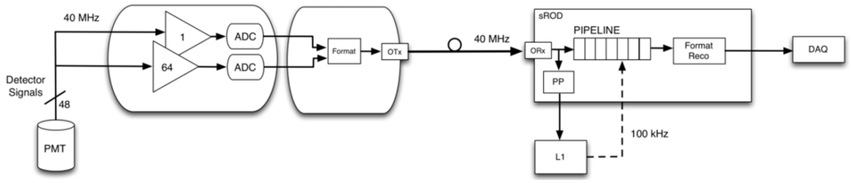

Figure 1: ATLAS TileCal upgraded read out architecture showing the sROD [2].

2. Optimal Filtering on ARM

Optimal Filtering (OF) is a simple algorithm that solves Equation 1 and 2 to find the peak amplitude, A, and the time shift, τ, given two weights vectors, a_i and b_i and the samples vector S_i [8]. n is equal to 7 in the existing TileCal implementation. The pedestal or noise floor, p, can be estimated as a constant or calculated with its own set of weights.

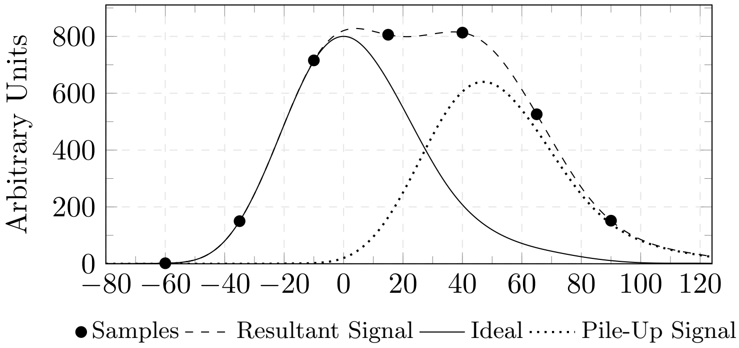

Figure 2: Ideal filtered PMT pulse showing seven samples.

$$A = \sum_{i=1}^{n} a_i(S_i - p) \qquad (1) \qquad\qquad \tau = \frac{1}{A}\sum_{i=1}^{n} b_i(S_i - p) \qquad (2)$$

This algorithm was implemented using the EIGEN C++ library which is one of the highest performing linear algebra libraries that also supports ARM [9]. Vector sizes of eight were used to correspond to the current TileCal read-out padded with an additional zero, and two dot products were performed with random weights and data to represent the amplitude and phase filters. A CPU is able to perform the dot product more efficiently with a vector size of eight because it fits into its Single Instruction Multiple Data (SIMD) registers better.

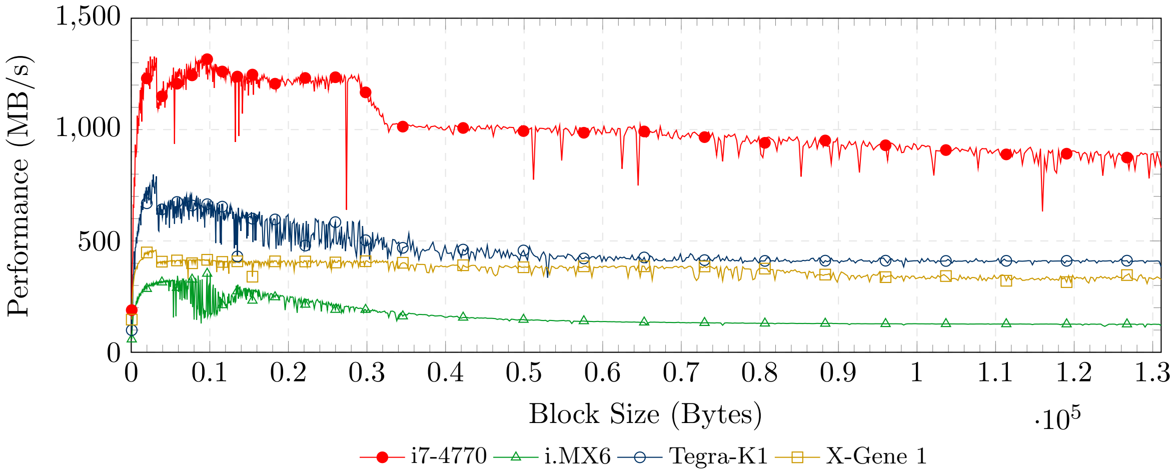

Figure 3: Single thread performance of an OF implementation with seven samples and two outputs with varying block size. An Intel i7-4770 CPU and three ARM platforms are shown. The results are normalised to a 1 GHz CPU clock.

Figure 3 shows the performance which has been converted to MegaBytes per second (MB/s) of filtered data where one megabyte is equal to 1024 × 1024 bytes. A high-end Intel i7-4770 CPU has been compared to an ARM Cortex-A9 (Freescale i.MX6), an ARM Cortex-A15 (NVIDIA Tegra-K1) and an ARMv8 (APM X-Gene 1) platform. All results have been normalised to 1 GHz. The i7 normally runs at 3.5 GHz "turbo boost", the i.MX6 at 1 GHz, the Tegra-K1 at 2.3 GHz and the X-Gene at 2.4 GHz.

Batches of sample and weight vectors were combined into blocks for more efficient processing. Block sizes from 32 B (one set of samples) to 128 kB were tested to find the maximum performance of the OF algorithm on the different platforms.

It is clear that the Intel performance is better than the ARM platforms. It is interesting to note that the performance of the Tegra-K1 is better than the X-Gene. This should not be the case but it was found that the compiler is not yet mature for the new ARMv8 architecture. Using GCC (GNU C Compiler) 4.9 instead of GCC 4.8 makes approximately 10% performance difference but for uniformity GCC 4.8.2 was used on all platforms.

The Wandboard platform (i.MX6 SoC - similar to the SoC found in a Samsung Galaxy SIII smart phone, for example) is capable of a peak of 350 MB/s and the other platforms all achieve over 1 GB/s of OF throughput. This is typically more than the external I/O available on the SoCs. It is important to note that the results are for a single thread and so if multiple cores are used then the results will scale accordingly because the algorithm can be run in parallel on separate data.

3. PCI-Express External I/O

The external I/O interface of a computing system should be well balanced with the available processing power. Because this is highly dependent on the algorithms used, benchmarking is important. Based on the results in Section 2, even the lowest performing system, the Wandboard, requires up to 350 MB/s external I/O for good system balance. Gigabit Ethernet - which is almost always available on a SoC - only allows up to 125 MB/s. This is clearly insufficient for optimal system balance in this application.

PCI-Express is a high-bandwidth external I/O interface that is energy efficient and simple to implement for the system designer as it only requires several PCB traces and very little special and potentially expensive hardware. For the connection to the sROD, PCIe can be used directly between the sROD and ARM SoCs.

PCI-Express throughput tests for the lower end i.MX6 SoC have been performed and are described in detail in another paper [7]. A summary of the results is shown in Table 1. The theoretical maximum throughput available on the SoC is 500 MB/s but based on the test results only slightly more than half of that is attainable on the i.MX6 because the Direct Memory Access (DMA) is not usable for generic PCIe data transfers.

Table 1: PCI-Express throughput results of an i.MX6 pair.

	Memory Copy	**DMA (EP)**	**DMA (RC)**
Read (MB/s)	$94.8 \pm 1.1\%$	$174.1 \pm 0.3\%$	$236.4 \pm 0.2\%$
Write (MB/s)	$283.3 \pm 0.3\%$	$352.2 \pm 0.3\%$	$357.9 \pm 0.4\%$

The X-Gene and Tegra-K1 system on chips have more advanced PCI-Express controllers which do support DMA. A test system for these platforms has not been built but in theory there should be no significant issues. The theoretical data throughputs for these platforms is approximately 15 GB/s and 2 GB/s respectively.

4. Discussion, Conclusions and Future Work

High data throughput computing is required for projects such as the LHC which produce enormous amounts of raw data. A general purpose ARM System on Chip based processing unit is being developed which will be used as a co-processor to the sROD to help mitigate the energy reconstruction issues caused by pile-up under higher luminosity operation of the LHC.

An OF algorithm was implemented and tested on ARM Cortex-A9, A15 and X-Gene (similar to Cortex-A57) SoCs. The slowest SoC, the Cortex-A9, achieved 350 MiB/s throughput on the algorithm. The Tegra-K1 peaked at 1.8 GB/s at the maximum clock speed with the X-Gene being slower because of the lack of compiler optimisation.

A PCI-Express interface will be used for the raw data transfer between the sROD and the PU. 2.8 GB/s of data throughput is required to sustain the raw data from the sROD prototype [2]. Initial throughput measurements presented for a pair of Freescale i.MX6 quad-core Cortex-A9 SoCs is 283 MB/s for the available x1 link. The theoretical data throughput of the Tegra-K1 and X-Gene SoCs are 2 GB/s and 15 GB/s respectively.

With multi threaded operation, the Tegra-K1 could sustain 1.8×4 GB/s but is limited by its 2 GB/s maximum PCIe throughput. A more sophisticated filtering algorithm may be effective given the excess of CPU processing power for the OF algorithm. The X-Gene could sustain eight times the measured throughput and so there would theoretically be no I/O bottleneck.

Testing still needs to be done on more complex filtering algorithms as well as algorithms for other sROD co-processor uses such as histograms.

Acknowledgements

The financial assistance of the National Research Foundation (NRF) towards this research is hereby acknowledged. We would also like to acknowledge the School of Physics, the Faculty of Science and the Research Office at the University of the Witwatersrand, Johannesburg.

References

[1] The ATLAS Collaboration, *Readiness of the ATLAS Tile Calorimeter for LHC collisions*, *The European Physical Journal C* **70** (Dec., 2010) 1193–1236.

[2] F. Carrió et al., *The sROD module for the ATLAS Tile Calorimeter Phase-II Upgrade Demonstrator*, *Journal of Instrumentation* **9** (Feb., 2014) C02019–C02019.

[3] The ATLAS Collaboration, *Letter of Intent for the Phase-II Upgrade of the ATLAS Experiment*, Tech. Rep. CERN-LHCC-2012-022. LHCC-I-023, CERN, Geneva, Dec., 2012.

[4] E. Fullana et al., *Optimal Filtering in the ATLAS Hadronic Tile Calorimeter*, 2005.

[5] B. Peralva, *The TileCal energy reconstruction for collision data using the matched filter*, in *2013 IEEE Nuclear Science Symposium and Medical Imaging Conference (2013 NSS/MIC)*, pp. 1–6, IEEE, oct, 2013.

[6] L. M. A. Filho, D. O. Damazio, and A. S. Cerqueira, *Calorimeter Signal Response Deconvolution for Online Energy Estimation in Presence of Pile-up*, *ATLAS Internal Note* (Sept., 2012).

[7] M. A. Cox, R. Reed, and B. Mellado, *The development of a general purpose ARM-based processing unit for the ATLAS TileCal sROD*, *Journal of Instrumentation* **10** (Jan., 2015) C01007–C01007.

[8] W. Cleland and E. Stern, *Signal processing considerations for liquid ionization calorimeters in a high rate environment*, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **338** (Jan., 1994) 467–497.

[9] B. Jacob and G. Guennebaud, *Eigen C++ Library*, 2014. http://eigen.tuxfamily.org.