PROCEEDINGS OF SCIENCE

FPGA implementation of a histogram-based parent bunch crossing identification for the Drift Tubes chambers of the CMS experiment

Fabio Montecassiano
Istituto Nazionale di Fisica Nucleare, Sezione di Padova
E-mail: fabio.montecassiano@pd.infn.it

Nicola Pozzobon[*]
Università degli Studi di Padova
Istituto Nazionale di Fisica Nucleare, Sezione di Padova
E-mail: nicola.pozzobon@pd.infn.it

Pierluigi Zotto
Università degli Studi di Padova
Istituto Nazionale di Fisica Nucleare, Sezione di Padova
E-mail: pierluigi.zotto@pd.infn.it

on behalf of the CMS Collaboration

The first running implementation on FPGA of a histogram-based trigger primitive generator for the CMS Drift Tubes at the High Luminosity LHC is presented. The foreseen architecture requires that raw charge collection times, measured for each tube by means of a TDC, are processed in the back-end to generate trigger primitives, identifying the parent bunch crossing and measuring the track parameters. We review the design of a parent bunch crossing evaluation, its implementation on FPGAs of the Xilinx UltraScale family by means of High-Level Synthesis, and the performance of a demonstrator board of such a trigger.

Topical Workshop on Electronics for Particle Physics TWEPP2019
2-6 September 2019
Santiago de Compostela - Spain

[*] Speaker.

1. Introduction

The Drift Tubes (DT) chambers of the Compact Muon Solenoid (CMS) experiment at the CERN LHC are gaseous detectors used in the barrel region of the CMS spectrometer for muon tracking and triggering at pseudorapidities $|\eta| < 1.2$ [1]. A standard chamber is composed of three super-layers (SL), each composed of four layers of staggered rectangular drift tubes. In view of the High Luminosity-LHC (HL-LHC) program, the front-end electronics will be upgraded introducing an asynchronous transfer of the primary TDC signal, *i.e.* the charge collection times measured by the TDC in the LHC orbit time frame, to the back-end. This requires a new approach to the Level 1 Trigger Primitives Generator (L1 TPG) [2], currently based on a synchronous time sampling [3].

Parent bunch crossing identification and track fitting are the requirements for muon triggers at the LHC. Hence we designed and simulated a Hough Transform-based trigger in C programming language, using a hardware-oriented style. Such a trigger is divided into a parent bunch crossing time (t_0) identification block, called Majority Mean-Timer (MMT), and a track identification block, embedding a Compact Hough Transform [4, 5, 6]. Both blocks are based on a Hough Transform-like approach, but they have different implementations mainly because of the algebraic form of the track equations in the MMT. Hereafter we present the first implementation of the MMT on FPGAs of the Xilinx UltraScale family [7]. We used High-Level Synthesis (HLS) tools in order to minimise the effort on hardware implementation of the algorithm. The optimal balance between resources and latency was obtained by implementing a highly parallelised algorithm and factorising the high-granularity multi-dimensional histograms to one-dimensional ones.

2. Algorithm overview and implementation

The parent bunch crossing identification relies on the mean-timing concept [8], which, so far, has been mainly used in drift velocity calibration [9]. Let t_j be the charge collection time in channel j measured in the LHC orbit frame, $T_j = t_j - t_0$ the reconstructed drift time in the same channel, and assume that the muon track is described by a straight line. Such a line through three points in three different layers results in a linear combination of the three T_js and the maximum drift time $T_{\max}$ in a DT cell. The muon crossing time t_0 can be computed directly, as explained in [5]:

$$a_1T_1 + a_2T_2 + a_3T_3 = bT_{\max} \quad \longrightarrow \quad (a_1 + a_2 + a_3) \times t_0 = a_1t_1 + a_2t_2 + a_3t_3 - bT_{\max}$$

provided that $a_1 + a_2 + a_3 \neq 0$. The coefficients $\{a_1, a_2, a_3, b\}$ in the linear combination depend on the geometry and on which side with respect to each wire the muon track is supposed to have crossed the cell. The regular layout of DT cells ensures that all coefficients are integer numbers. Since only few neighbouring wires should be processed to propose a candidate, a chamber is divided into macro-cells, as defined in [4], including the full information for any muon track.

The algorithm was implemented with Xilinx Vivado HLS code developed from the original emulator of the algorithm in C language, according to the Figure 1 diagram. A ‘hit collector’ module gathers, decodes, and feeds TDC hits to a ‘hit driver’ module which assigns them to their position within a macro-cell. The ‘hit collector’ features a paged scheduling scheme, which ensures that all TDC hits from the same muon are processed together at least once. A ‘super-layer processor’ evaluates all possible combinations of hits and side assumptions (patterns), for each group of active channels in a macro-cell. After computing all relevant equations and providing a t_0

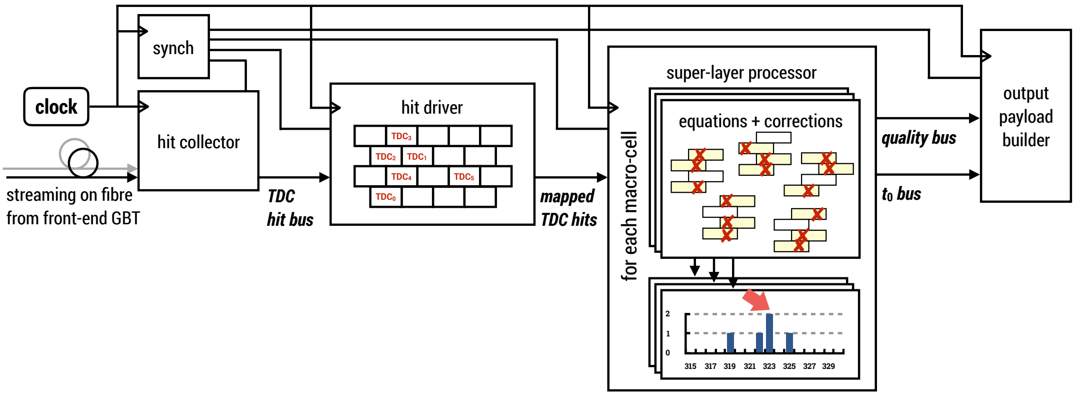

Figure 1: Block diagram of the MMT algorithm implemented in an FPGA. In the 'super-layer processor' block, an example of active channels is shown: only some associated patterns, deliberately including some patterns incompatible with each other, are drawn. The 'super-layer processor' includes a set of corrections to cope with non-uniform drift field.

candidate for each pattern, a histogram is real-time populated. The output t_0 for each macro-cell is the most frequent one in the histogram. A high quality (low quality) flag is assigned to candidates consistent with the alignment of 4 (3) hits within a SL. The output t_0 and quality are delivered to the payload builder for a given number of macro-cells. The charge collection times t_js are represented with a least significant bit corresponding to 3.125 ns in the algorithm, *i.e.* one eighth of the 25 ns LHC bunch crossing interval, matching the DT single hit resolution, and four times better than the current L1 DT TPG, while t_0 is currently determined with a 25 ns precision.

A first implementation was evaluated on a speed grade -3 Xilinx XCVU440 FPGA, clocked at 200 MHz, using realistic TDC hits from simulated muons at HL-LHC rate. Negligible mismatch is found with respect to the C emulator, including inefficiencies and pile-up. The amount of resources needed is approximately 16,000 Look-Up Tables (LUT) for a macro-cell of 18 wires, without using any hard-wired multiplier blocks (DSP). The best latency obtained is less than 500 ns, corresponding to 20 LHC bunch crossings. The current design, still provisional and open to improvements, requires an amount of resources comparable to the ASIC processors of the current L1 DT TPG and is compatible with the L1 decision latency budget [10]. The expected performance was also evaluated within the CMS simulation framework, where the muon crossing time t_0 is discretely distributed in multiples of 25 ns [11]. The efficiency of high quality triggers ranges between 85% and 90% if information from only one SL is available, and increases up to 98-99% when both $r\phi$ SLs are used. All figures are consistent with the legacy L1 DT TPG.

3. Performance evaluation with cosmic muons

The same algorithm was used to collect cosmic muon data at the INFN Legnaro National Laboratories in August 2019 with a setup made of four reduced-area DT SLs, called mini-chambers. Such mini-chambers share the same technology with the CMS DT chambers and are arranged in a tower configuration with alternate orientation, being indexed 0 to 3 starting from the lowest one. Two Xilinx VC707 boards, used to prototype the new TDC Phase II DT boards (OBDT) [2], collect data from two mini-chambers each, and provide unfiltered TDC streaming on fibre using the GBT protocol [12]. A Xilinx KCU1500 board, featuring a XCKU115 FPGA, speed grade -2, hosts the back-end firmware which implements two GBT de-serializer modules and saves the TDC streams on a host computer mass storage, via Direct Memory Access through PCIe. The algorithm was interfaced to the TDC streams, and deployed on the XCKU115 FPGA. The input TDC stream and the trigger output are merged and stored via PCIe into a single dataset. The footprint per macro-cell is unchanged: the implementation of the 7 macro-cells needed to process a full mini-chamber requires 22% of FPGA LUTs. Timing was closed at 160 MHz by choice: higher frequencies could be achieved with slight algorithm optimisation, longer placement and routing effort, which are beyond the scope of the test.

The hardware trigger is used to process only data from mini-chamber 2, but, in fact, all mini-chambers are used in the evaluation. Indeed, the software emulator is used to determine the expected triggers for each mini-chamber, having the full set of input data stored by the back-end. A candidate muon is tagged by the coincidence of two high quality emulated triggers in mini-chambers 0 and 3, or 1 and 3, since $96.95 \pm 0.02\%$ of such coincidences are in-time with the parent muon in simulations. Redundant triggers are removed from both the emulator and hardware trigger outputs. The algorithm proved to be robust against the scheduling phase, thanks to the chosen time-hermetic paged scheme of the 'hit collector' module.

The setup ran, collecting data without interruptions, for 25 hours and 45 minutes, while the trigger payload was written 8,705,737 times. No error was reported by the monitoring tools of the algorithm firmware and GBT. The agreement between emulated triggers and actual hardware triggers on mini-chamber 2 was measured to be 98.3% for the output t_0, and 99.4% for the trigger quality. The selected trigger is correct, but associated to different macro-cells due to their superposition, in less than 0.3% of cases. Figure 2 shows the measured performance of the muon crossing time assignment, which must be interpreted knowing that t_0 is output with a 25 ns precision. A

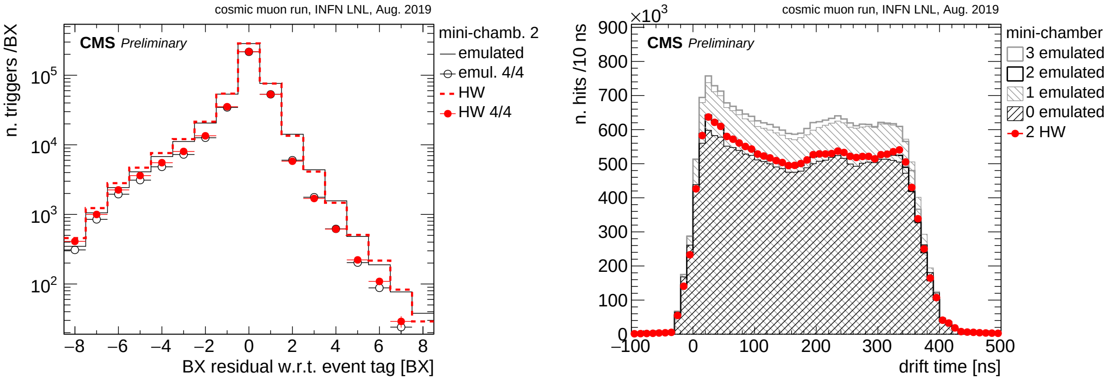

Figure 2: (Left) Trigger time residual distributions measured with cosmic muons using a telescope of reduced-area super-layers and a continuous DAQ setup: the reference time is given by coincidences of high quality candidates in external mini-chambers; the high quality triggers subset (4/4) is shown with circles. (Right) Drift time distributions obtained from triggered events: distributions are built using the crossing time produced by the trigger emulator in all mini-chambers and by the hardware trigger in mini-chamber 2; the emulated trigger distribution in mini-chamber 2 is hardly visible being hidden by the red markers.

slight excess of hardware triggers is found in mini-chamber 2 with respect to the emulator.The time residuals of hardware triggers with respect to the high quality coincidences of emulated triggers in external mini-chambers is shown on the left. The relative population of in-time triggers is $58.70 \pm 0.07\%$ of total, and $85.40 \pm 0.05\%$ of high quality triggers. The drift time $T_j = t_j - t_0$ distributions, with crossing time t_0 assigned by emulated triggers in all mini-chambers and hardware triggers in mini-chamber 2, is shown on the right. All distributions are consistent with the non-bunched asynchronous nature of cosmic muons.

4. Conclusions and outlook

We successfully interfaced a histogram-based parent bunch crossing identification for the DT chambers of the CMS experiment to a realistic Phase II front-end, and used it to process cosmic muon data in real time. We measured a consistent performance with expectations, showing the algorithm is robust against the synchronisation of the hit collection module. Important planned improvements are already in the design phase, in particular a hit collector module which will feature a dynamic geographical assignment of input data to macro-cells in order to reduce both area and latency. An output precision better than 25 ns can be obtained with a finer granularity histogram because of the newer, more compact design. Porting of the Compact Hough Transform-based design to an FPGA has already started, with the design advantage that the existing architecture, already developed for the MMT bunch crossing identification, can be re-used.

Acknowledgments

We thank our colleagues working on the LEMMA and CMS 40 MHz Scouting projects in Padova, who provided us with the mini-chambers setup and infrastructure, described in Section 3.

References

[1] CMS Collaboration, *Journal of Instrumentation* **3** (2008) S08004.

[2] CMS Collaboration, CERN-LHCC-2017-012, CMS-TDR-016.

[3] P. Arce *et al.*, *Nucl. Instr. Meth. A* **534** (2004) 441–485.

[4] N. Pozzobon, F. Montecassiano and P. Zotto, *Nucl. Instr. Meth. A* **834** (2016) 81–97.

[5] N. Pozzobon, P. Zotto and F. Montecassiano, *Eur. Pys. J. Web Conf.* **127** (2016) 00012.

[6] N. Pozzobon, F. Montecassiano and P. Zotto, *IEEE Trans. Nucl. Sci.* **64** (2017) 1474–1479.

[7] https://www.xilinx.com/products/silicon-devices/fpga.html

[8] F. Gasparini *et al.*, *Nucl. Instr. Meth. A* **336** (1993) 91–97.

[9] G. Abbiendi *et al.*, *JINST* **4** (2009) P05002

[10] CMS Collaboration, CERN-LHCC-2017-013, CMS-TDR-017.

[11] CMS Muon Group, approved performance plots of the MMT-CHT DT TPG with simulated muons https://twiki.cern.ch/twiki/bin/view/CMSPublic/HistogramBasedDTTPGLocal2019Sim

[12] P. Moreira *et al.*, TWEPP 2009 – Topical Workshop on Electronics for Particle Physics, Paris, France, 21 - 25 Sept. 2009, pp.342-346 (CERN-2009-006)