Use of an FPGA to identify electromagnetic clusters and isolated hadrons in the ATLAS level-1 calorimeter trigger

J. Garvey[a,*], S. Hillier[a], G. Mahout[a], R. Staley[a], P. Watkins[a], A. Watson[a],
R. Achenbach[b], P. Hanke[b], W. Hinderer[b], D. Kaiser[b], E.-E. Kluge[b], K. Meier[b],
O. Nix[b], K. Penno[b], K. Schmitt[b], B. Bauss[c], A. Dahlhoff[c], K. Jakobs[c],
K. Mahboubi[c], U. Schäfer[c], J. Thomas[c], T. Trefzger[c], E. Eisenhandler[d],
M.P.J. Landon[d], D. Mills[d], E. Moyse[d], P. Apostologlou[e], B.M. Barnett[e],
I.P. Brawn[e], A.O. Davis[e], J. Edwards[e], C.N.P. Gee[e], A.R. Gillman[e], R. Hatley[e],
V.J.O. Perera[e], C. Bohm[f], S. Hellman[f], S.B. Silverstein[f]

[a] School of Physics and Astronomy, University of Birmingham, Birmingham B15 2TT, UK
[b] Kirchhoff-Institut für Physik, University of Heidelberg, D-69120 Heidelberg, Germany
[c] Institut für Physik, University of Mainz, D-55099 Mainz, Germany
[d] Physics Department, Queen Mary, University of London, London E1 4NS, UK
[e] Rutherford Appleton Laboratory, Chilton, Didcot OX11 0QX, UK
[f] Fysikum, University of Stockholm, SE-106 91 Stockholm, Sweden

Received 3 December 2002; received in revised form 31 March 2003; accepted 7 April 2003

Abstract

At the full LHC design luminosity of 10^{34} cm^{-2} s^{-1}, there will be approximately 10^9 proton–proton interactions per second. The ATLAS level-1 trigger is required to have an acceptance factor of $\sim 10^{-3}$. The calorimeter trigger covers the region $|\eta| \leqslant 5.0$, and $\phi = 0$ to 2π. The distribution of transverse energy over the trigger phase space is analysed to identify candidates for electrons/photons, isolated hadrons, QCD jets and non-interacting particles. The Cluster Processor of the level-1 calorimeter trigger is designed to identify transverse energy clusters associated with the first two of these. The algorithms based on the trigger tower energies which have been designed to identify such clusters, are described here. The algorithms are evaluated using an FPGA. The reasons for the choice of the actual FPGA being used are given. The performance of the FPGA has been fully simulated, and the expected latency has been shown to be within the limits of the time allocated to the cluster trigger. These results, together with the results of measurements made with real data into a fully configured FPGA, are presented and discussed.
© 2003 Elsevier B.V. All rights reserved.

PACS: 29.40.Vj; 29.90.+r; 84.30.s

Keywords: Trigger; Calorimeter; FPGA

*Corresponding author. Tel.: +121-414-4570, +1235-44-5670; fax: +1235-44-6733, +121-414-6709.
E-mail address: j.garvey@bham.ac.uk (J. Garvey).

1. Introduction

At the full LHC design luminosity of $10^{34}\,\mathrm{cm^{-2}\,s^{-1}}$, there will be approximately 25 proton–proton interactions per bunch crossing. The ATLAS level-1 trigger is required to select bunch crossings at a level of only 1 in 1000. This is achieved by identifying interactions containing candidate high transverse energy muons using muon trigger chambers, and electrons, photons, single hadrons/taus, QCD jets, and non-interacting particles using calorimeters. The level-1 trigger must not generate any deadtime, and takes the form of real-time custom-built pipelined processors [1].

2. Trigger environment

The level-1 calorimeter trigger covers the phase space $|\eta| \leqslant 5.0$ and $\phi = 0$ to 2π. The electromagnetic cluster trigger and the isolated hadron trigger extend out only to $|\eta| \leqslant 2.5$, as this is the region in which tracking is provided by the Silicon Central Tracker (SCT), and the Transition Radiation Tracker (TRT). The trigger takes in signals from the liquid argon electromagnetic barrel and end-cap, from the hadronic tilecal barrel and extended barrel, and from the liquid argon hadronic endcap calorimeters. The calorimeter cells are combined on the detector to form trigger towers. In the liquid argon electromagnetic calorimeters, electromagnetic trigger towers are formed by adding the four samplings in depth, and forming towers covering $\Delta\eta \times \Delta\phi = 0.1 \times 0.1$. In the tilecal, which covers the range $|\eta| \leqslant 1.6$, all three depth samplings are added to form hadronic trigger towers, and in the hadronic endcap, the first three of the four depth samplings are used. The hadronic trigger towers are aligned behind the electromagnetic trigger towers and cover the same area of $\eta \times \phi$ space. The total number of trigger towers used in the trigger is 7200, of which 6400 are used in the electromagnetic cluster and isolated hadron trigger discussed in this paper. The various elements of the level-1 calorimeter trigger are indicated in Fig. 1.

The 7200 analogue pulses are carried on 60 m long dedicated copper cables from the detector front-end electronics, to the trigger crates in the underground trigger electronics room. The calorimeter trigger algorithm is evaluated using the transverse energy in the trigger towers. To reduce the dynamic range of energy pulses before digitisation, the analogue signals go into a Receiver module where a weighting is applied over bins in $\sin\theta$, to produce an approximate value of transverse energy, "E_T". These analogue pulses then enter the Preprocessor of the calorimeter trigger, where they are digitised to 10-bit precision at a frequency of 40 MHz. After bunch-crossing identification (see below), the precise value of transverse energy for each trigger tower is produced using a look-up table. The input to the look-up table is "E_T" ($= 10$ bits) and the output is the transverse energy, $E_\mathrm{T}(= 8$ bits), incorporating the exact value of θ corresponding to that particular trigger tower. The transverse energy scale is linear up to 255 GeV.

The calorimeter trigger is a pipelined processor which operates at the LHC beam crossing frequency of 40 MHz. For each beam crossing
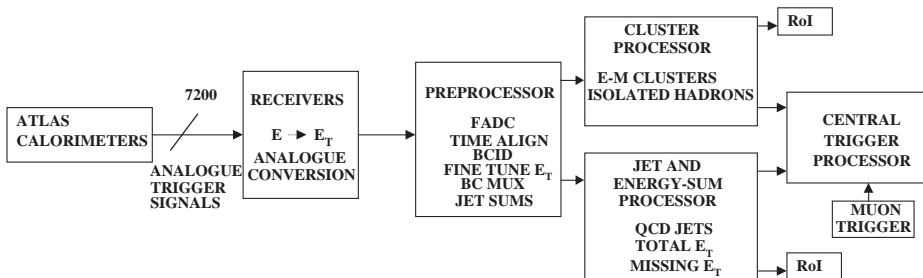

Fig. 1. Elements of the ATLAS level-1 calorimeter trigger.

the processor requires a snapshot of the pattern of transverse energy deposited in the trigger towers. The analogue pulses from the calorimeters extend over more than one 25 ns LHC clock period. The area of the pulse, which is a measure of the transverse energy, must be calculated and assigned to the beam crossing which produced the interaction. This bunch-crossing identification (BCID) is performed by a Finite Impulse Response (FIR) filter in the Preprocessor. The output from the FIR for each trigger tower consists of a 40 MHz train of digitised pulses, giving the transverse energy deposited in each successive beam crossing.

An important quantity is the overall latency of the calorimeter trigger. This is discussed in the TDR [2], where the delay between the proton–proton interaction and the production of the Level-1 accept signal by the Central Trigger Processor, corresponding to the cluster trigger, is estimated as 53.6 LHC clock periods (1.34 µs). Of this the Receivers plus the Preprocessor account for 18.1 LHC clock periods (0.453 µs), the Cluster Processor 14 LHC clock periods (0.35 µs), and the Central Trigger Processor 4 LHC clock periods (0.1 µs). Within the Cluster Processor, the functions performed by the cluster FPGA described here are expected to account for about 6 LHC clock periods. At the time of the TDR the intention was to use an ASIC for the cluster finding, but the rapid evolution of FPGA capabilities in the meantime has made them the obvious choice.

3. Trigger windows

For $\eta = 0$, the lateral dimensions of the trigger towers are approximately 10 cm × 10 cm. The Molière radius in lead is about 1 cm, leading to electromagnetic showers with lateral dimensions of a few cms. Electrons or photons entering the centre of a trigger tower will deposit the majority of their energy in one tower. However, particles entering near the join between two contiguous towers will deposit their energy in two or more towers. To obtain uniform energy response across the phase space covered by the calorimeters, the cluster energy is based on the sum of two
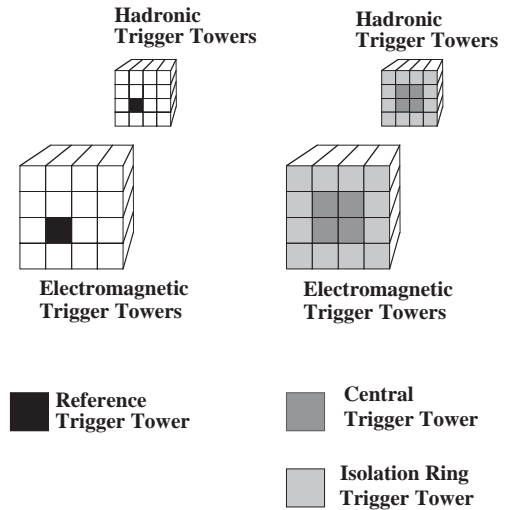

Fig. 2. Definition of trigger towers within a trigger window.

contiguous trigger towers. A slightly more accurate measurement of cluster energy would be obtained by adding three or four contiguous towers. However this decreases the rejection of QCD jets.

The elements of the trigger algorithms which have been designed for the trigger are indicated in Fig. 2. They are based on arrays of 4 × 4 electromagnetic and 4 × 4 hadronic trigger towers which are referred to as trigger windows. In each window one trigger tower is called the reference tower, as indicated in the figure. Each of the electromagnetic trigger towers is a reference tower for a trigger window. We identify the following trigger towers within each window.

- Reference trigger tower of the window. Each electromagnetic trigger tower has its own trigger window.
- Electromagnetic calorimeter central trigger towers. These are the 2 × 2 arrays of electromagnetic calorimeter trigger towers in the centre of the window.
- Electromagnetic calorimeter isolation ring trigger towers. These are the 12 electromagnetic trigger towers surrounding the central towers.
- Hadronic calorimeter central trigger towers. These are the 2 × 2 arrays of hadronic

calorimeter trigger towers in the centre of the window.

- Hadronic calorimeter isolation ring trigger towers. These are the 12 hadronic trigger towers surrounding the central towers.

4. Trigger algorithms

The calorimeter trigger described in this paper is designed to find electromagnetic clusters and isolated hadrons. A large fraction of the isolated hadron triggers will probably come from tau decays.

By combining the transverse energies of the trigger towers within the trigger window the following trigger elements are formed:

- *Electromagnetic clusters*: these are the four possible sums of two adjacent central electromagnetic trigger towers.
- *Electromagnetic isolation sum*: this is the sum of the 12 electromagnetic calorimeter isolation ring trigger towers.
- *Hadronic clusters*: these are the four possible sums of two adjacent central electromagnetic trigger towers, plus the sum of all four central hadronic trigger towers.
- *Hadronic isolation sum*: this is the sum of the 12 hadronic calorimeter isolation ring trigger towers.
- *Central hadronic isolation sum*: this is the sum of the 2×2 hadronic calorimeter central trigger towers.
- *Central Region of Interest* (RoI): this is the sum of the four central electromagnetic and four hadronic trigger towers.
- *Peripheral RoIs*: these are the sums of the additional 2×2 arrays of electromagnetic and hadronic trigger towers within the trigger window. There are eight combinations. These are shown in Fig. 3.

The formation of the trigger elements is similar for the electromagnetic trigger towers and the hadronic trigger towers in order to simplify the implementation of the trigger algorithms. From these trigger elements the electromagnetic cluster
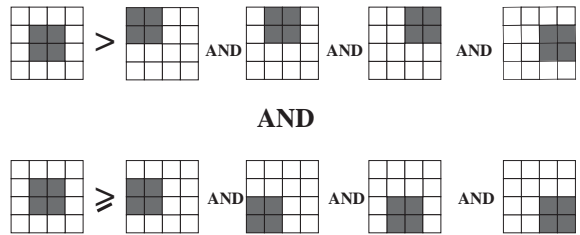

Fig. 3. The declustering requirement applied to the central RoI and the eight peripheral RoIs in each trigger window.

trigger and the isolated hadron triggers are formed.

For the electromagnetic cluster trigger the following requirements must be satisfied:

- At least one electromagnetic cluster must be above a trigger threshold.
- The electromagnetic isolation sum, the hadronic isolation sum, and the central hadronic isolation sum must all be below their specified thresholds.
- The Central RoI must be a local transverse energy maximum.

This last requirement needs some further explanation and clarification. Its purpose is to prevent cells with high transverse energy causing triggers in two adjacent trigger windows. It makes use of the fact that for any window, the central RoI clusters of the eight neighbouring windows are present, and so each window can compare its central RoI with that of its eight neighbours, even if the neighbouring windows are actually processed in a different hardware device. To avoid ambiguity due to finite resolution of digital transverse energy sums, the requirement is loosened to requiring that the central RoI be more energetic than its four peripheral RoIs at the $+\phi$ and $+\eta$ edges of the window, and at least as energetic as its four peripheral RoIs along the opposite edges. This is shown in Fig. 3.

For the isolated hadron trigger the following requirements must be satisfied:

- At least one hadronic cluster must be above a trigger threshold.

- The electromagnetic isolation sum and the hadronic isolation sum must be below their specified thresholds.
- The central RoI must be a local transverse energy maximum.

To allow a comprehensive range of trigger settings, the trigger design incorporates a total of 16 sets of threshold values. Of these eight are allocated to electromagnetic clusters, and eight can be allocated to electromagnetic cluster or isolated hadron triggers. A set of thresholds consists of the cluster threshold and the relevant isolation thresholds.

5. Implementation

The purpose of this paper is not to describe in detail the Cluster Processor of the calorimeter trigger, but rather to describe the use of the FPGAs in its implementation. However, a brief description of the overall architecture of the processor and the signal flow through it will put the role of the FPGAs into clearer focus.

The Cluster Processor has as input an array of $(\eta, \phi) = (50, 64) \times 2$ trigger towers. Each trigger tower has to be fully processed within its own trigger window. The trigger design must be factorised down to a set of discrete electronic modules, the signal dialogue between modules providing the continuous sensitivity over the complete trigger phase space. The modularity of the Cluster Processor depends on many factors, but a balance must be struck between the physical size of the modules, the number of I/O pins available, and the processing power which can be mounted on a module. The processing power required to evaluate the trigger algorithms is provided by FPGAs, for which the important parameters are gate capacity and speed.

The Cluster Processor consists of 56 Cluster Processor Modules (CPM). These are organised into four crates, each crate covering one quadrant in ϕ and $|\eta| \leqslant 2.5$, and each CPM covering $\Delta\eta = 0.4$. Most CPMs fully process 4×16 trigger towers in $\eta \times \phi$. To fully process such an array of trigger towers requires the CPM to have as input an array of dimensions (7×20). This is indicated in Fig. 4.

The additional trigger towers are required to form the trigger windows indicated in Fig. 2, the total array being 7×20 instead of 7×19 because of the BCmux mechanism described below. It is evident from Fig. 4 that within a crate the majority of trigger towers have to be fanned out to other CPMs. This is done via the crate backplane.

The modularity chosen only requires fanout to nearest neighbour CPMs, greatly simplifying the design.

In choosing an FPGA many factors have to be considered.

- Time allowed for the evaluation of the trigger algorithms.
- Processing power of the FPGA.
- Sustainable input bandwidth for data.
- Number of input pins available.
- Number of FPGAs in the processor.
- Cost and availability.

The choice of a specific FPGA has to be taken many years before the trigger is required. This is a difficult decision as the power of available FPGAs is constantly increasing. In the design of the cluster trigger, the requirement for overlapping trigger windows is essential to guarantee a uniform efficiency for cluster detection over the detector. Such a requirement needs a fanout of trigger towers to many trigger windows. This fanout would be best done inside the silicon of the FPGA. Referring to our trigger architecture in Fig. 4, the best implementation would be to take in and process all the trigger towers for one CPM on a single FPGA. There was no FPGA with a large enough number of gates to do this.

The FPGA chosen was the Xilinx Virtex chip, XCV1000E, which has 1.57 M gates, 28 k logic blocks, and a maximum of 404 I/O pins. [3] To guarantee the quality of signals transmitted over the crate backplane, we have fixed the signal bandwidth at 160 Mb s^{-1}. This input bandwidth is handled comfortably by the FPGA.

This processing capacity allows an area of trigger towers $(\eta \times \phi) = (4 \times 2)$ to be fully processed by one FPGA. To evaluate the trigger algorithms described in Section 4 requires a trigger tower array (7×5) in $\eta \times \phi$ to be available inside the FPGA. This is shown in Fig. 4. Because of the
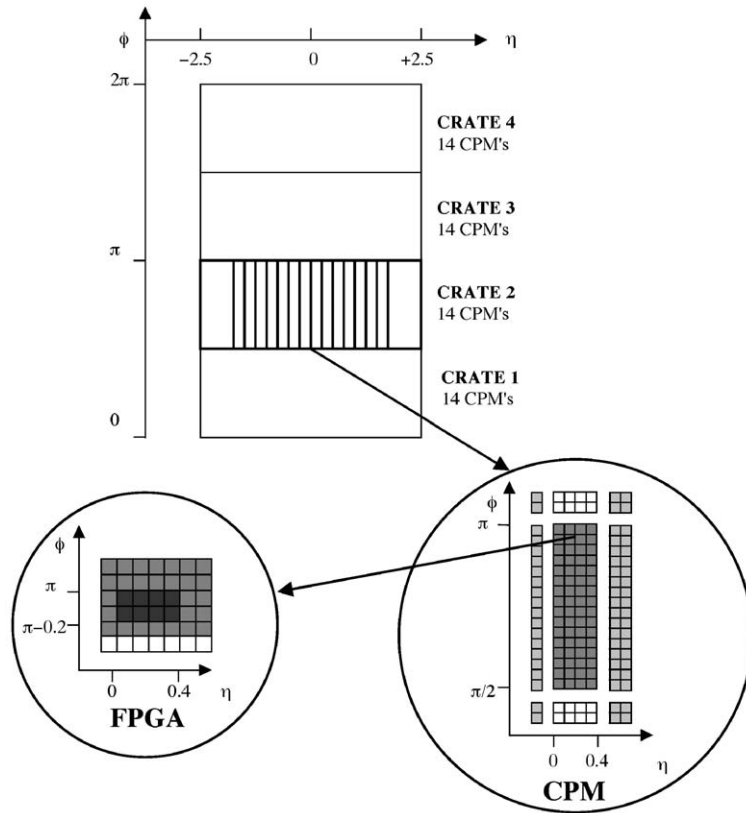
Fig. 4. Organisation of the Level-1 calorimeter trigger into electronic modules (CPM) and individual FPGAs.

BCmux, described below, the additional cells in white are also input to the FPGA. The trigger design results in a CPM containing 8 FPGAs.

6. Signals into and out from the FPGA

On the real-time path of the trigger, 108 pins of the FPGA are used for signal input and 32 pins for signal output.

The inputs into the FPGAs are the transverse energies in the electromagnetic and hadronic trigger towers, digitised to 8 bits with 1 bit = 1 GeV. The digitised pulses are produced in a Preprocessor Module (PPM), and represent the area of the calorimeter pulse. The time of arrival of the digitised pulse at the input to the FPGA is fixed relative to the LHC beam crossing which produced the energy deposition in the calorimeter.

This is achieved by the beam crossing identification logic (BCID), implemented in the PPM by the use of an FIR filter, and by a set of tunable delays.

For a given trigger tower pulse, the BCID logic cannot give an output on two consecutive beam crossings. This allows the application of a multiplexing scheme, BCmux, which results in data compression. This is indicated in Fig. 5. The BCID outputs from two trigger towers adjacent in ϕ are shown as a function of the LHC beam crossing number. Each BCID output is an 8-bit word. The two streams of 8-bit words are merged in the following way. On receipt of a non-zero pulse from either tower, an additional ninth bit is added, which is equal to 1 for the lower ϕ tower and 0 for the upper ϕ tower. If both towers have non-zero pulse heights for a given beam crossing, tower 0 pulse height is sent for that beam crossing with the BCmux bit set to 0, and tower 1 is sent on the
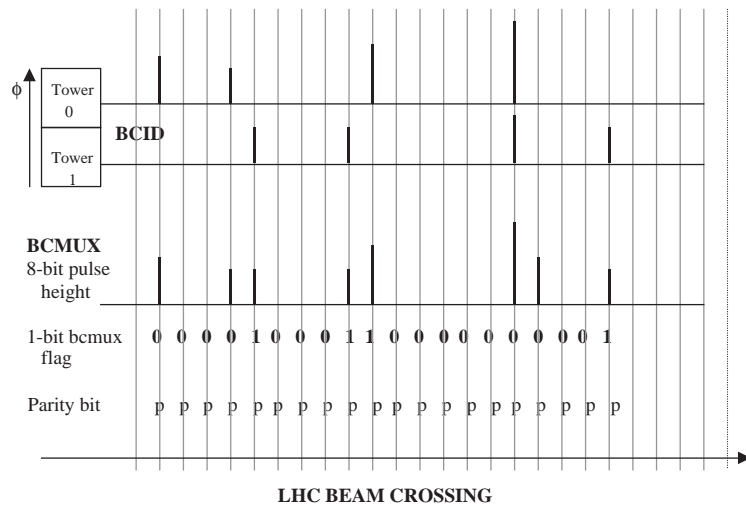
Fig. 5. Operation of the BCMUX logic, which results in data compression.

subsequent beam crossing with the BCmux bit also set to 0. The first BCmux bit indicates which tower is being sent, the second BCmux bit indicates whether it comes from the same beam crossing ($= 0$), or from the next beam crossing ($= 1$). A parity bit is added to produce a 10-bit word. This BCmuxing results in the two ϕ-adjacent trigger towers being carried from the PPM to the CPM in one 10-bit word every LHC beam crossing. To reduce the input pin count into the CPM, this 10-bit word is transferred as a serial stream at 400 Mb s^{-1}, and restored to 10-bit parallel data on arrival at the CPM [4].

The concept of trigger windows and the architecture of the Cluster Processor requires a large fan-out of trigger tower signals between adjacent CPMs, and within a CPM between the FPGAs. This is clear from Fig. 4 where the total area of trigger towers required by one CPM is shown. This fan-out is achieved over the crate backplane. To reduce the number of connections on the backplane, the fanned-out data is trans-mitted at 160 Mb s^{-1}. This is considered to be the upper limit for single-ended connections. It is also near the limit for single-ended input to the FPGA. This requires that the trigger tower signals, consisting of 10-bit parallel words at 40 MHz, are converted to 160 Mb s^{-1} serial data

streams within the CPM before input to the FPGAs, and before fan-out to neighbouring CPMs via the crate backplane. This is described below.

Within the CPM, two 10-bit words from trigger towers neighbouring in η are formed into a 20-bit word which is serialised into five 160 Mb s^{-1} data streams of four bits each, as indicated in Fig. 6. The result of the BCmuxing and serialisation is that four trigger towers are brought into the FPGA at 160 Mb s^{-1} on five input pins. Four pins are each carrying four bits of data every 25 ns, and the fifth pin is carrying the two BCmux flags and two parity bits. The trigger tower area indicated in Fig. 4 is brought into each FPGA on 108 input pins.

We give below the operations which have to be performed by the FPGA in order to evaluate the trigger algorithms described in Section 4.

- Synchronisation of the 108 160-Mb s^{-1} data streams.
- BC demultiplexing logic.
- Parallelisation of the input data to form 84 8-bit words every 25 ns. Fourteen of these 8-bit words are not used as the actual array required by the FPGA to evaluate the algo-rithms is only $(\eta \times \phi) = (7 \times 5)$. This comprises
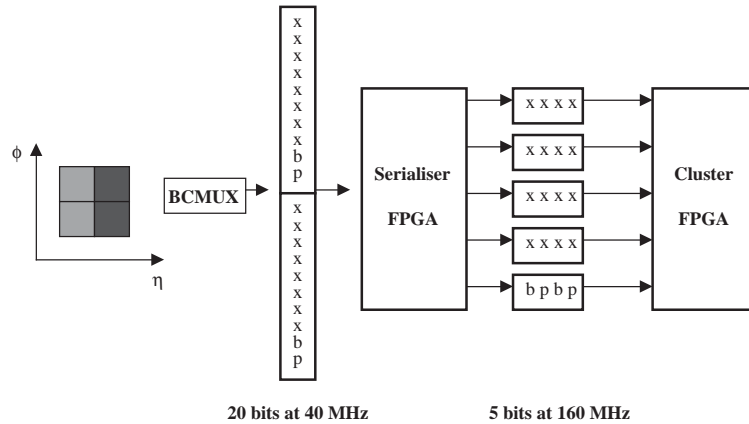
Fig. 6. Data input to the cluster FPGA. Four tower signals are input on five pins at 160 Mb s^{-1}.

35 electromagnetic transverse energies and 35 hadronic transverse energies.

- Algorithms evaluation using a clock frequency which is a multiple of 40 MHz.
- Output trigger signal generation, producing one 16-bit word from each of the two $(\eta \times \phi) = (2 \times 2)$ trigger tower areas being fully processed by the FPGA.

The sets of two 16-bit words from each FPGA are then combined to form the trigger multiplicities from all 8 FPGAs in a CPM. These are sent to merger modules which count the total multiplicity for each threshold set for all the CPMs and send this to the Central Trigger Processor.

The processing described above is all on the real-time path of the trigger. The RoI data, which will be sent to the higher level triggers when a level-1 trigger is signalled by the level-1 Central Trigger Processor, consists of the two 16-bit words from each FPGA. Each 16-bit word comes from an $(\eta \times \phi) = (2 \times 2)$ area of trigger towers. Each bit of the 16-bit word corresponds to one of the 16 sets of threshold values described in Section 4 and indicates which thresholds have been crossed. This is stored in a pipelined memory of the FPGA, to be read out when a level-1 accept is generated by the Central Trigger Processor.

7. Performance of the FPGA

The firmware for the FPGA is based on the algorithms described Section 4. The algorithm block was written in VHDL. The placement and routing is done by the Xilinx Synthesis Tool. The whole design is contained within the Mentor Graphics Renoir framework. The gate usage is about 60%. Changing the functions performed by the FPGA is in principle straightforward, requiring the production of a new firmware file, which is then easily downloaded into the FPGA via VME. In practice care has to be taken to ensure that there are sufficient resources in the chip, and that changes in latency are minimised.

The 16 sets of energy thresholds are loaded via VME into registers in the FPGA and do not require a reload of the firmware.

A full simulation of the operation of the configured FPGA has been made using pre-defined input test vectors.

The expected output for each test vector can be calculated, and this is compared with the predicted output from the simulation process. In this way the processing capabilities of candidate FPGAs are tested before a choice of FPGA is made. A vital consideration in the choice is the time taken to evaluate the trigger algorithms.

The simulation exercise predicted processing times for the evaluation of trigger algorithms in
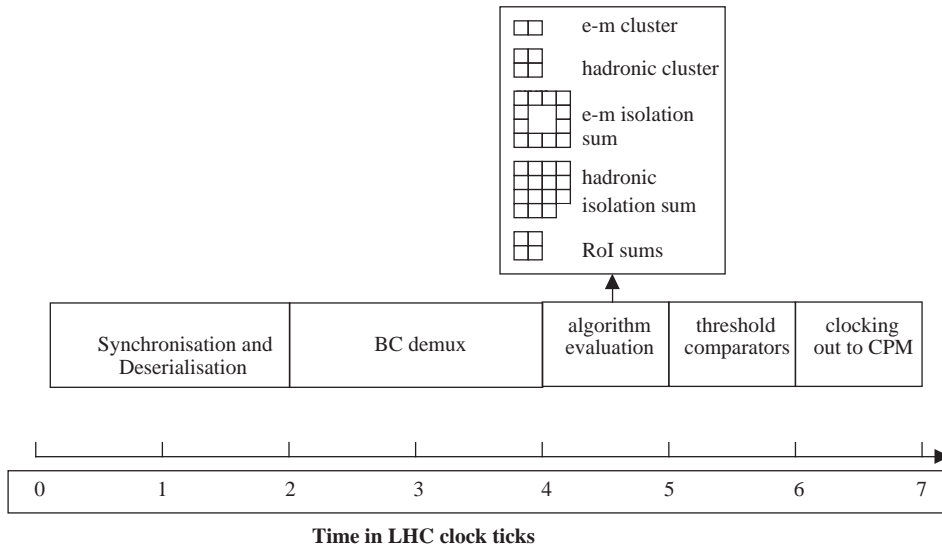
Fig. 7. The functions performed by the cluster FPGA in the level-1 calorimeter trigger, indicating the time taken for each operation.

the FPGA. These are summarised in Fig. 7. It is interesting to see that most of the time is taken up with the preparation of the data for the algorithm evaluation. The evaluation of the algorithm is done with fully parallel data, with one 8-bit word per trigger tower. The power of the FPGA is seen clearly in the time taken to evaluate the trigger algorithms. All this is completed within one LHC clock cycle.

The total latency in the cluster FPGA is simulated to be 7 LHC clock ticks, which satisfies fairly well the target set in the TDR and presented in Section 2. The value of the latency achieved with an FPGA depends on the firmware configuration of the chip. The design must be optimised for minimum latency. FPGAs are extremely versatile devices, which can be reconfigured to perform different functions, and operate with different latencies. However, the optimisation in both gate use and in speed of operation requires careful firmware design.

The simulation has been compared with measurements made on the hardware. This was first done with one fully configured FPGA mounted on a test rig, where the results of the simulation were verified for a limited range of input test data. A full-specification, 9U × 40 cm, prototype CPM has

been constructed and is now under test. A photograph is shown in Fig. 8, where the FPGAs can be seen as the eight large chips on the module. Preliminary indications are that the CPM is performing to the level predicted by the simulation.

8. Advances in technology

During the design of the architecture and the implementation of the cluster trigger design into physical electronic modules, the performance specification of many of the building blocks has improved significantly, and this is a continuing process. However, a point is reached where the implementation must be frozen, in order to allow sufficient time to guarantee the successful construction and testing of the trigger before installation into ATLAS. Since that time, significant changes have occurred in component specifications, and it is possible to gauge how rapidly this process will continue. These changes will affect the design of future first-level calorimeter triggers. We indicate below how these changes would affect the design described in this paper.

The most significant developments affecting the implementation of the cluster-trigger design are in
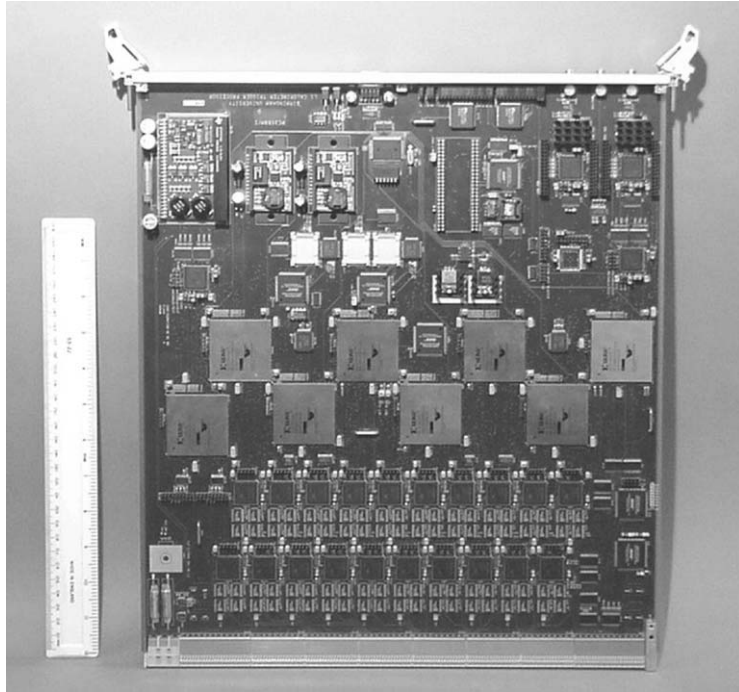
Fig. 8. Photograph of the CPM. The cluster FPGAs are the eight large chips in the middle of the board.

the speed and capacity of FPGAs. The FPGA used in our design is the Xilinx Virtex-E XCV1000E, which has 1.5 million gates. Already in the Virtex-II devices, chips with 4.7 million gates are available, with the same price per gate and a clocking speed which is double that of the Virtex-E devices.

An increase in gate capacity and speed as indicated above would have a profound effect on the implementation of the first-level cluster processor described in this paper. Much more parallel processing can be performed within one FPGA. Referring to Fig. 4, it can seen that to fully process an array of (2×4) trigger towers an area of (6×7) trigger towers has to be taken into each cluster chip. A CPM fully processes a trigger tower area 16×4, and this requires inputs from a trigger tower area of 20×7. We estimate that the area covered by one CPM could now be fully processed by one Virtex-II chip.

Such a change in implementation would result in a much simpler CPM design, with no loss of functionality. However, the number of pins carrying data in and out of the module would only be reduced slightly. With further evolution of FPGAs, the I/O pin capacity of the electronic module emerges as the limiting factor in the number of trigger towers that can be fully processed. This limitation can be overcome by increasing the bandwidth of the input serial data stream. This is possible at present, with the availability of serial links up to 2.0 GBd. They are not used in the design described here, as the power consumption required to convert to parallel data is very high. However, a significant development in this regard would be the availability of FPGAs which were able to accept input data at this bandwidth.

9. Conclusions

We have described here the use of an FPGA to find electromagnetic clusters and isolated hadrons

in the design of the Cluster Processor Module of the ATLAS level-1 calorimeter trigger. This paper has concentrated on the function of the FPGA chosen to find the electromagnetic clusters and isolated hadrons. The power of the FPGA is indicated in the time taken to evaluate all the trigger algorithms which are described, and the large number of input signals. The rapid increase in the power of available of FPGAs indicates that their use in real time applications in triggering will grow, and will in turn increase the selectivity which is possible.

References

[1] ATLAS First-Level Trigger Technical Design Report, CERN/LHCC/98-14.
[2] ATLAS First-Level Trigger Technical Design Report, CERN/LHCC/98-14, pp. 202–204, 415.
[3] Virtex^{TM}-E 1.8V Field Programmable Gate Arrays, Xilinx Data Sheet, DS022.pdf. (available from the Xilinx web site, www.xilinx.com).
[4] G. Anagnostou, et al., Study of LVDS Serial Links for the ATLAS Level-1 Calorimeter Trigger, Proceedings of the Sixth Workshop on Electronics for LHC experiments, Kraków, Poland, September 2000. CERN/LHCC/2000-10, pp. 333–337.