High Level Trigger System for the LHC ALICE Experiment

H. Helstrup[1], J. Lien[1], V. Lindenstruth[2], D. Röhrich[3], B. Skaali[4], T. Steinbeck[2], K. Ullaland[3], A. Vestbø[3], and A. Wiebalck[2]
for the ALICE Collaboration

[1] Bergen College, P.O. Box 7030, 5020 Bergen, Norway
[2] Kirchhoff Institute for Physics, University of Heidelberg, Schröderstrasse 90, 69120 Heidelberg, Germany
[3] Department of Physics, University of Bergen, Allegaten 55, 5007 Bergen, Norway
[4] Department of Physics, University of Oslo, P.O. Box 1048 Blindern, 0316 Oslo, Norway

Abstract. The ALICE experiment at the Large Hadron Collider (LHC) will produce a data size of up to 75 MByte/event at an event rate of up to 200 Hz resulting in a data rate of ~15 GByte/s. Online processing of the data is necessary in order to select interesting events or sub-events (high-level trigger), or to compress data efficiently be modeling techniques. Both require a fast parallel pattern recognition. Processing this data at a bandwidth of 10-20 GByte/s requires a massive parallel computing system. One possible solution to process the detector data at such rates is a farm of clustered SMP-nodes based on off-the-shelf PCs, and connected by a high bandwidth, low latency network.

1 Introduction

The ALICE experiment [1] at the upcoming Large Hadron Collider (LHC) at CERN will investigate Pb-Pb collisions at a center of mass energy of about 5.5 TeV per nucleon pair. The main detector is a 3-dimensional tracking device, the Time Projection Chamber (TPC), which is specially suited for handling large multiplicities. The detector is readout by 570,000 ADC-channels, producing a data size of ~75 MByte/event [3]. The event rate is limited by the bandwidth of the permanent storage system. Without any further reduction or compression ALICE will be able to take events up to 20 Hz. Higher rates are possible by either selecting interesting events and sub-events (high-level trigger), or compressing the data efficiently by modeling techniques. Both high-level triggering and data compression requires pattern recognition to be performed online. In order to process the detector information of 10-20 GByte/s a massive parallel computing system (high-level trigger system) is needed. The construction of such a system includes development of both hardware and software optimized for a high throughput data analysis and compression.

P.M.A. Sloot et al. (Eds.): ICCS 2002, LNCS 2329, pp. 494–502, 2002.

2 Functionality

From a trigger point of view the detectors in ALICE can be divided into two categories: fast and slow. Fast detectors provide information for the trigger system at every LHC crossing. Decisions at trigger 0,1 and 2 are made using information from these detectors. The slow detectors (such as the TPC) are tracking drift detectors and need longer time span after the collisions to deliver their data. Their slowness is compensated for by the detailed information they provide. The ALICE High-Level Trigger (HLT) system is intended to take advantage of this information in order to reduce the data rate as far as possible to have reasonable taping costs. The data is then recorded onto an archive-quality medium for subsequent offline analysis.

A key component of the HLT system is the ability to process the raw data performing track pattern recognition in realtime. About 20,000 particles per interaction each produce about 150 clusters in the detectors. These signals has to be read out, processed, recognized and grouped into track segments. Based on the detector information – clusters and tracks – data reduction can be done in different ways:

– Generation and application of a software trigger capable of selecting interesting events from the input data stream ("High-Level Trigger")

– Reduction in the size of the event data by selecting sub-events

– Reduction in the size of the event data by compression techniques

2.1 Event rate reduction

The ALICE TPC detector will be able to run at a rate up to 200 Hz for heavy ion collisions, and at up to 1 kHz for p-p collisions [3]. In order to increment the statistical significance of rare processes, dedicated triggers can select candidate events or sub-events. By analyzing tracking information from different detectors online, selective or partial readout of the relevant detectors can be performed, thus reducing the event rate. The tasks of such a high-level trigger are (sub)event selections based upon the online reconstructed track parameters of the particles, e.g. to select events which contains e^+–e^- candidates coming from a quarkonium decay. In the case of low multiplicity events from p-p collisions the online pattern recognition system can be used to remove pile-up events from the data stream.

2.2 Realtime data compression

Data compression techniques can be divided into two major categories: lossy and lossless. Lossy compression concedes a certain loss of accuracy in exchange for greatly increased compression. Lossless compression consists of those techniques guaranteed to generate an exact duplicate of the input data stream after a compress/expand cycle.

General lossless or slightly lossy methods like entropy and vector quantizers can compress tracking detector data only by factor 2-3 [2]. The best compression method is to find a good model for the raw data and to transform the data into an efficient representation. By online pattern recognition and a compressed data representation an event size reduction by a factor of 15 can be achieved [3]. The information is stored as model parameters and (small) deviations from the model. The results are coded using Huffman and vector quantization algorithms. All correlations in the data have to be incorporated into the model. The precise knowledge about the detector performance, i.e. analog noise of the detector and the quantization noise, is necessary to create a minimum information model of the data. Realtime pattern recognition and feature extraction are the input to the model.

3 Online pattern recognition

Both high-level triggering and data modeling require a fast parallel pattern recognition. The data modeling scheme is based on the fact that the information content of the TPC are tracks, which can be represented by models of clusters and track parameters. Fig. 1 shows a thin slice of the TPC detector, clusters are aligned along the trajectories (helices). The local track model is a helix; the knowledge of the track parameters helps to describe the shape of the clusters in a simple model [5] [6]. The pattern recognition reconstructs clusters and associates them with local track segments. Note that track recognition at this time can be redundant, i.e. clusters can belong to more than one track and track segments can overlap. Once the pattern recognition is completed, the track can be represented by helix parameters.

3.1 Cluster finder and track follower

In the STAR [7] experiment the online tracking is divided into two steps: [8] Cluster finding and track finding. A cluster finder searches for local maxima in the raw ADC-data. If an isolated cluster is found, the centroid position in time and pad direction is calculated. In the case of overlapping clusters simple unfolding procedures separate charge distributions. Due to the missing track model there are no cluster models available, therefore the centroid determination is biased. The list of space points is given to the track finder, which combines the cluster centroids to form track segments.

A simple cluster finder and a track follower which applies conformal mapping [9] in order to speed up fitting routines has been adapted to ALICE TPC data. This method has shown to work efficiently only in the case of low multiplicity Pb-Pb collisions and for p-p collisions, where the amount of overlapping clusters are small.

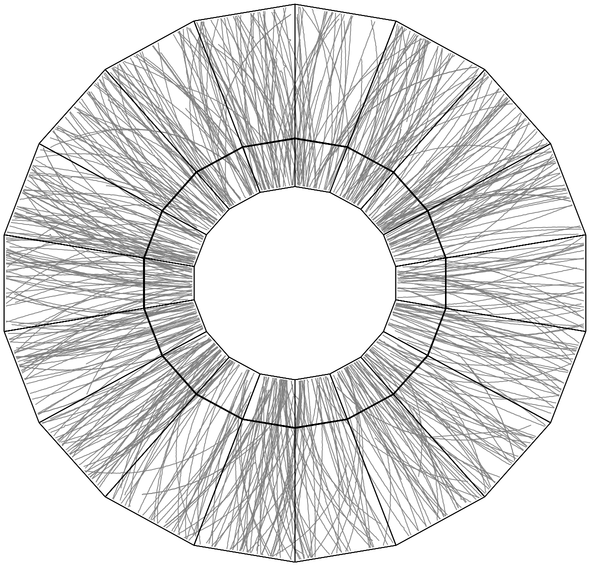

Fig. 1. Simulated Pb-Pb collision at LHC. Only a small fraction of the tracks in the ALICE TPC detector is shown

3.2 Hough transform

For the high cluster densities that are expected for heavy ion collisions at LHC, an alternative is being developed. In this case track segments finding is performed directly on the raw data level, using a special case of the Hough transformation technique.

The Hough transform is a standard tool in image analysis that allows recognition of global patterns in a image space by recognition of local patterns (ideally a point) in a transformed parameter space. The basic idea is to find curves that can be parametrized in a suitable parameter space. In its original form one determines a curve in parameter space for a given signal corresponding to all possible tracks with a given parametric form it could possibly belong to. All such curves belonging to the different signals are drawn in parameter space. This space is then discretized and entries are histogrammed – one divides parameter space up into boxes and counts the number of curves in each box. If the peaks in the histogram exceeds a given threshold, the corresponding parameter values define a potential track.

In ALICE, the transformation is done assuming the tracks follow circles in the transversal plane. In order to simplify the transformation, the detector volume is divided into sub-volumes in pseudorapidity, η. If one restricts the analysis to tracks starting from a common interaction point (assumed to be the vertex), the track model is characterized by only two parameters; the emission angle with the beam axis ϕ_0 and the curvature κ. The transformation is performed using the equation:

$$\kappa = \frac{2}{R} sin(\phi - \phi_0) \tag{1}$$

where R and ϕ are the polar coordinates of the point to be transformed. Each

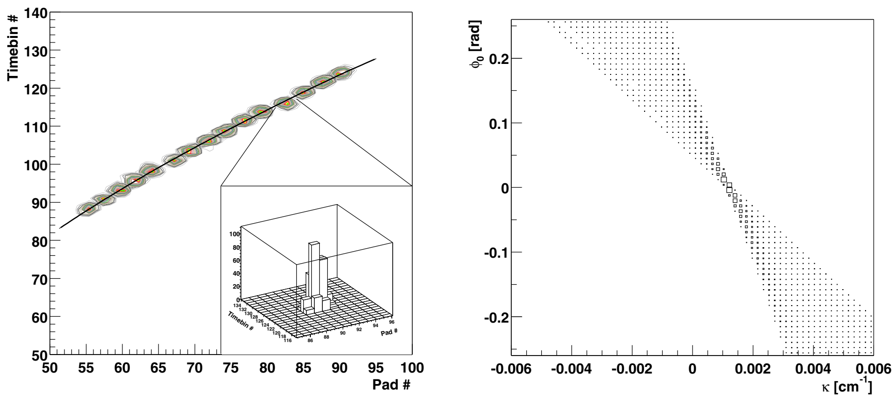

Fig. 2. On the left is shown clusters and associated track segments over 15 successive padrows in the TPC. The right figure shows a corresponding histogram resulting from a Hough transform applied on the signals produced by the track. Each pixel (padrow,timebin, pad) on the track is transformed into a quantised curve in parameter space (κ,ϕ_0), and the corresponding bins are incremented with the ADC-value of the pixels. The intersection of these curves is located at the parameters of the track

active pixel (ADC-value above threshold) transforms into a sinusoidal line extending over the whole ϕ_0-range in the parameter space, see Fig. 2. The corresponding bins in the histogram are incremented with the ADC-value of the transformed pixel. The super-position of all these point transforms produce a maxima at the circle parameter of the track. The track recognition is now done by searching for local maxima in the parameter space. Once the track parameters are known, cluster finding on the raw ADC-data can be performed by a straight forward unfolding of the clusters.

3.3 FPGA implementation

In order to minimize the CPU-load on the HLT processors, as much as possible of the pattern recognition will be implemented in dedicated hardware located on the PCI receiver cards in the first layer of the HLT system (see Fig. 5). Fast isolated cluster finding and the Hough transformation are simple algorithms that will be implemented into one or several FPGAs.

Fig. 3 shows a sketch of the architecture of the PCI receiver cards. One possible implementation scenario is the existing ALTERA Excalibur series of FPGAs, which include a choice of an ARM or MIPS embedded processor. The design will allow for most of the pattern recognition steps to be implemented in the FPGA. The internal SRAM can be extended by external high-speed SRAM if necessary. All additional logic, including the PCI interface, can also be implemented in the FPGA.

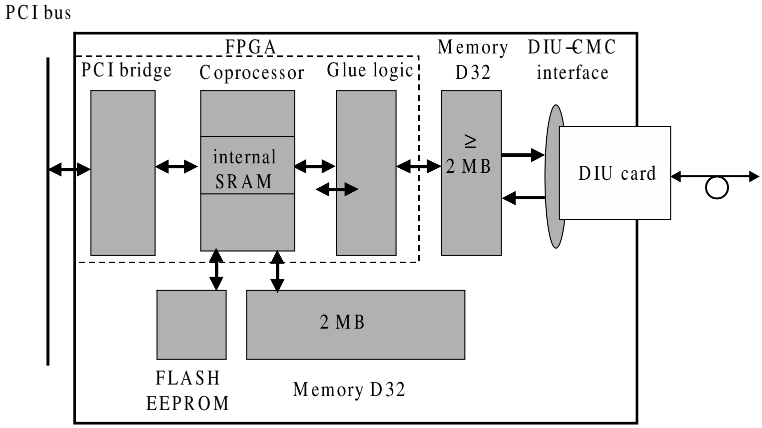

Fig. 3. PCI Receiver Card. The DIU is the interface to the optical detector link

Fig. 4 shows a block diagram of a possible implementation of the Hough transformation. In order to gain speed the raw input data will be processed time multiplexed and in parallel. The input is 10 bit raw ADC data coming from the front-end electronics of the detector. Every nonzero pixel is sorted according to its pseudorapidity, η, which is provided using a look-up table stored in the internal memory of the FPGA. In addition, the polar coordinates are provided using a combination of look-up tables and calculation algorithms. The actual transformation is done in a address calculator which calculates the addresses of counters which is incremented with a 8 bit representation of the ADC value of the pixel. Each counter corresponds to a bin in the histogram in (κ,ϕ_0)-space. There are maximum 128 histograms on each card, and the size of each histogram is $64\times64\times12$ bits.

The peaks in each of the histograms will be found by means of firmware algorithms on the FPGA, and the resulting list of track candidates will be shipped over PCI to the host for further processing. Alternatively the histograms may also be shipped to the host, and the peak finding performed on the CPU.

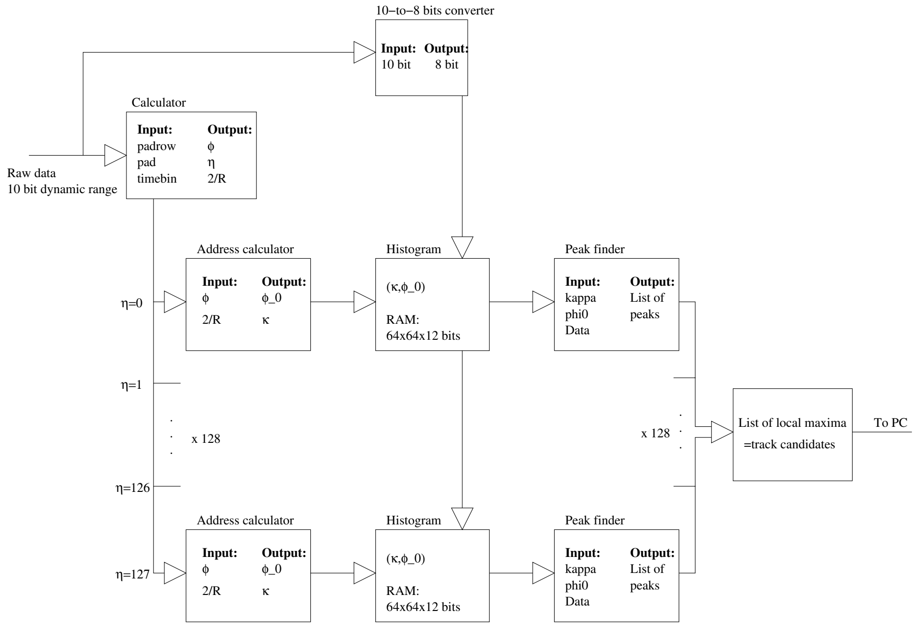

Fig. 4. A possible FPGA implementation of the Hough transform

4 Architecture

The ALICE HLT system will be located in the data flow after the front-end electronics of the detector and before the event-building network. A farm of clustered SMP-nodes, based on off-the-shelf PCs and connected with a high bandwidth, low latency network provide the necessary computing power for online pattern recognition and data compression. The system nodes are interfaced to the front-end electronics of the detector via their internal PCI-bus.

The hierarchy of the farm has to be adapted to both the parallelism in the data flow and to the complexity of the pattern recognition scheme. Fig. 5 shows a sketch of the foreseen hierarchy of the system. The TPC detector consists of 36 sectors, each sector being divided into 6 sub-sectors. The data from each sub-sector are transferred via an optical fiber from the detector into 216 receiver nodes of the PC farm. A hierarchical network interconnects all the receiver processors. Each sector is processed in parallel, results are then merged in a higher level. The first layer of nodes receive the data from the detector and performs the pre-processing task, i.e. cluster and track seed finding on the sub-sector level. The next two levels of nodes exploit the local neighborhood: track segment finding on sector level. Finally all local results are collected from the sectors and combined on a global level: track segment merging and final track fitting. The total farm will consist of 500-1000 nodes, partially equipped with FPGA co-processors.

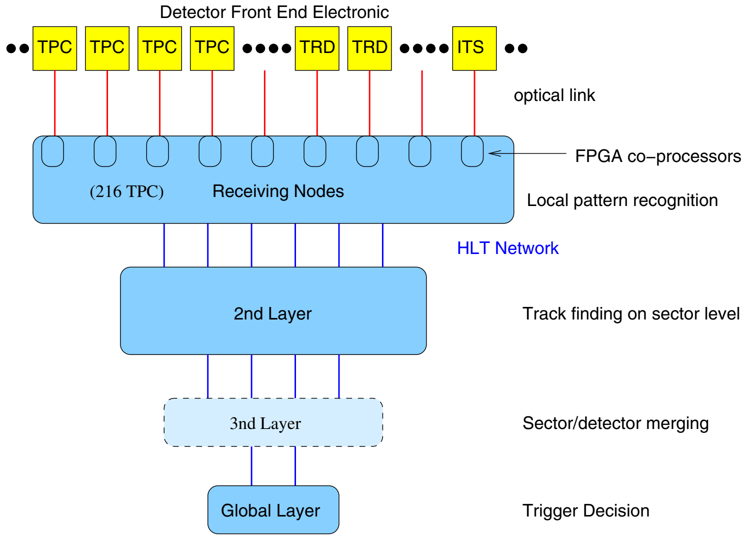

Fig. 5. Architecture of the HLT-system. The data from each sub-detector are transferred via optical fiber to the receiver processors. Results of the local processing on the receiver nodes are merged at a higher level. At the final global level all the processed data are merged, and the trigger decision can be made

The farm has to be fault tolerant and scalable. Possible network technology candidates are amongst others SCI [4] and Gigabit-Ethernet. The operating system will be Linux, using a generic message passing API for communication with the exception of data transfers, which use directly remote DMA.

5 Summary

The High-Level Trigger system for the ALICE experiment at the LHC accelerator has to process data at a rate of 10-20 GByte/s. The architecture of a PC cluster and data analysis software which fulfills this requirement are currently under study. The information from the online pattern recognition can be used for data compression, to derive physical quantities (e.g. momentum, impact parameters etc.) for High-Level Trigger decisions and for online monitoring.

References

1. ALICE collaboration, *Technical Proposal*, CERN/LHCC/95-71 (1995).
2. J. Berger et al., *TPC Data Compression*, Subm. to Nucl. Instrum. Methods Phys. Res., A
3. ALICE collaboration, *Technical Design Report of the Time Projection Chamber*, CERN/LHCC 2000-001 (2000).

4. SCI (Scalable Coherent Interface) Standard Compliant, ANSI/IEEE 1596-1992.
5. H. Appelshäuser, PhD thesis, University of Frankfurt 1996.
6. J. Günter, PhD thesis, University of Frankfurt 1998.
7. STAR collaboration, *STAR Conceptual Design Report*, Lawrence Berkeley Laboratory, University of California, PUB-5347 (1992).
8. C. Adler et al., *The Proposed Level-3 Trigger System for STAR*, IEEE Transactions on Nuclear Science, Vol. 47, No.2 (2000).
9. P. Yepes, *A Fast Track Pattern Recognition*, Nucl. Instrum. Meth. **A380** (1996) 582.