LC-Note XXXX

Data Handling for the LC Detector from the front end to full reconstruction DRAFT 1

G. Eckerlin, A.R. Gillman, J. Goldstein, P. LeDu

March 28, 2003

Scope of this document

In the framework of the future linear collider research and development will be needed both on detector and machine technologies and on the readout of the detector components. In many cases existing readout technologies are used for the prototype systems, but in some areas new developments have started and for the final detectors new concepts are being developed. In fact the time available should be used to elaborate on new technologies handling large data volumes. This should not be limited to techniques used in HEP, may be new ideas can be adapted from other fields.

The purpose of this document is to guide the various R&D groups towards possible common solutions for the prototype systems as well as for a future design without any prejudice in terms of new technologies which may come up. First the document will try to define the boarder lines between front end readout and central readout components and then define generic building blocks and interfaces between them. The aim is to ease common developments and reuse of components where possible. By defining distinct blocks with clear interfaces, development can be done by different R&D groups in parallel without loosing the ability to use common central readout components at the earliest possible stage. By this approach is should also be possible to incorporate new technologies later on with minimal efforts. Finally the document will point out possible R&D areas for the different readout components and tries to show examples of technologies which may be available in near future.

A working group is proposed, which should coordinate and foster corresponding R&D work and members from different R&D group dealing with readout components are encouraged to join.

1 introduction

This section will introduce the subject, the following points will be raised and a figure like Fig.1 demonstrates the evolution in this field.

- history and experience with current systems
- state of the art systems
- new recent developments

see separate document written by Patrick LeDu.

2 data handling model

This section describes the data handling in general, which is logically divided into 4 parts:

- the front end readout
- the central data collection
- the event building and data processing
- and the data storage for the final analysis.

The last part is not covered by this document.

2.1 The front end readout

see separate document by Patrick LeDu

- functional description
- building blocks
- block diagram
- requirements
- some examples

A more detailed figure of the functional block diagram (Fig.2) will be included here.

2.1.1 Examples

As an examples of currently proposed front end readout systems the vertex detector readout is shortly described here. See Ref.[2] and [3] for further details.

The vertex detector readout

by Toni Gillman and Joel Goldstein.

The proposed vertex detector for an experiment at the Linear Collider is an 800 Mpixel silicon device, with sensors mounted on mechanical ladders arranged into five barrels with radii from 15 mm to 60 mm. The material budget is 0.120 microns, giving excellent impact parameter resolution over a large momentum range. Although different pixel technology options are currently the subject of extensive R&D, only a design based on column parallel CCDs is discussed here. Similarly for brevity, only the (more demanding) Tesla environment is used as an example.

Integrating data from over 2820 bunches in each train at Tesla would result in an excessively high hit density, so there have to be multiple readouts within the train: twenty for the innermost layer and five for the remainder. This produces data from a total of over four billion pixels per train.

A custom front-end readout chip is bump-bonded to the CCDs at each end of a ladder. Each chip contains not only an amplifier and FADC for each column of pixels, but also the digital processing for noise suppression (Extended Row Filtering - ERF), cluster finding and zero suppression, as well as a local memory cache sufficient to retain all the hit data from one bunch train (¡1 Mbyte) until the ladder is read out (see Figure 3).

A schematic of a possible detector readout scheme is shown in Figure 4. At each end, all of the ladders in a given layer are connected via a ring bus to an external serial link. Since the expected total data volume even for the innermost layer, which contains the largest hit density, is only about 6 Mbytes per bunch train per end (assuming 2x2 clustering), commercial 400 Mbit/s LVDS serial links would be adequate to read out all layers in the 200 msec inter-train gap.

A single DAQ interface card is situated at each end of the detector, outside of the sensitive region. This card has an FPGA to format and multiplex the serial bitstream data from the individual layers, and sufficient memory to retain data from at least one bunch train (10Mbytes per end). When requested by the central DAQ, this memory is read out via an optical transmitter driving a single fibre at ¡ 1Gbit/s. The card will also distribute the control signals to the detector.

Figure 3 - Schematic of front-end chip using seeded 2x2 clustering
Figure 4 - Readout scheme for one end of the vertex detector

2.2 Central data collection

After the treatment of the data in the front end readout with the detector specific electronics, all data should be avaliable in a format such that a common data handling of all sub systems should be possible. We assume that the data will be read by a central readout unit from the front end system which buffers the data for further formatting and processing. The readout unit should have an intelligent interface to adapt future changes of the front end format if needed. After receiving the data a input buffer large enough to decouple the front end system from the data collection system is forseen. The size of the buffer has still to be specified and should be kept flexible. The data are then further processed and reformatted if needed by local processing in the readout unit and stored in an output buffer which is able to serve as a local storage for test purposes. This output buffer will be interfaced to the event building network and the final data processing. Industry standard technologies should be used for the output buffer and its interface to the event building. The size of the buffer again has to be large enough to decouple the final event building from the proceeding steps.

2.3 Event building and data processing

The event building will then be done by reading the data of all subsystems for a complete train from the output buffers of the different readout units into a common processing unit. The processing unit itself will not be specified in detail at that moment, but for simplicity a PC-farm node might be used as an example. The event building will have the full information of all subdetectors and all bunch crossings in one train.

It is likely that the event building will be divided into several logical steps. It is assumed that the data are formated in the readout units such that no further reformatting is needed otherwise this would be done first.

The next logical step would then be to do a full reconstruction of the complete train and all subsystems. Some detector parts include several bunch crossings into one picture and need to look at these bunches in parallel, others may run through the bunch crossings sequentially, or even having several processes in parallel. How to treat this in the most efficient way has to be studied in detail using realistic Monte Carlo and reconstruction programs.

The next step would be a finder which one can think of having several processes in parallel dealing with the data to identify bunches of interest

to the specific physics, calibration or monitoring needs. A physics student, machine expert or detector expert who needs specific data would specify the selection criteria running in one of the finders or if no suitable finder frame exists write a finder process which will then run to select the data for him.

The finally tagged bunches of interest will then be signaled for storing on permanent media. For each finder classification one could foresee a separate budget, or bandwidth in case not all data of a specific finder is needed. For example a minimum bias finder would select otherwise rejected events, but only a small fixed fraction would be taken for crosschecks or tuning.

The signaled event will then be send by the data logging network to permanent storage. The data logging network might be physically the same network as the event building network it is logically treated here as a separate network to be more general. All data not sent to permanent storage could if the buffers on the processing node is chosen large enough be kept temporary until a defined time is elapsed or a garbage collection is needed to free buffer space. The time these data are kept depends largely on the buffer size chosen and this in turn will certainly be defined by the technology available then.

2.4 detector control

see separate document written by Patrick LeDu.

2.5 calibration and other issues

see separate document written by Patrick LeDu.

3 boundaries and interfaces

With the data handling model described above the four different layers are clearly defined. The front end electronics, the central data handling, the full event processing and the data analysis of the permanently stored data. The last layer will not be further specified here, this document ends at the permanent storage but it has to be noted that there should not be a difference in the code that runs on the stored data and the code that runs in the finder steps. There is no 'online' or 'offline' in the old sense in the proposed scheme. It might well be that certain technologies developed for the GRID will be implemented on the last layer. The basic building blocks of the first three layers will be specified and the interface between all four different layers will be defined.

3.1 building blocks

The front end readout layer is naturally determined by the different detector technologies and will not be common. Nevertheless several common parts will be used, and to ease data treatment later on, the output of the front end electronics should be unified as much as possible.

The front end readout should include the detector signal treatment with preamplifier, shaper, digitizing, as well as the local treatment of the detector data like sparcification, first level buffering and multiplexing to reduce the amount of data lines coming from the detector as much as possible. All these functionalities should be built on the detector itself or very near to it. This requires high integration and low power consumption electronics to reduce heat load as well as dead material.

The central data handling layer will read the data from the front end readout through a link, which will interface the two layers. It will consist of an intelligent interface card that receives the data from the link, formats the data and stores it in a input data buffer. The data format received by the interface card has to be programmable to allow different detectors to use the same interface card and to allow for future changes of the data format. The data will then optionally be further processed and again stored in an output data buffer. If no further processing is needed both data buffers might be identical and unnecessary moving of data should be avoided. The output data buffer should have a standard industry interface.

The full event processing as the third layer will read the data of all detectors for all bunch crossings of a full train from the different readout units involved. The readout will be done via a event building network which is the interface between the event processing and the central readout units. The full event processing will then process the full data, selects and sends the data to permanent storage. It will most likely be a farm of processors each with its own interface to the network, a data input buffer, a data output buffer and an interface to the data logging network. The input and output buffer may be identical as well as the interfaces for input and output.

In addition to the four layers an overall process control will be needed, which may just be a subset of processes in the different layers or dedicated hardware connected to the different layers via the network. A central clock distribution and synchronization with the colliding bunches has to be provided for the front end readout. From then on the systems do not need further synchronization with the machine itself, but run starts, stops and initialization has to signaled at least.

3.2 interfaces

The four different layers need to be interfaced such that a future change in technology for one layer will allow to keep the other layers as is or at least keep the changes in the other layers minimal.

The interface of the front end readout to the central data handling units will most probably be a serial line. At the moment this is assumed to be a serial optical link with 10Gbit/sec or more bandwidth and the data will be send in digital form. Is 16bit wide sufficient ?

The interface of the central readout units to the full event processing is a industry standard network, the event building network. It should be non blocking, scalable and it should be possible to partition it maybe even hardware wise.

The interface of the event processing to the permanent storage, the data logging network, is probably the same kind or even physically the same network.

3.3 general considerations

All layers have to be designed such that a part or the whole detector can be included or excluded from a data taking stream by software configuration. Several data taking streams should be possible, with each stream having a part or all of a layer included. It might be advantageous if a part of a detector could be included in more than one data taking stream at a time, (eg. physics data taking and calibration run at the same time). If one would build the readout system at the moment the front end interface would probably be a 10Gbit optical fiber and the event building network would probably be Gigabit Ethernet with TCP/IP or alternatively Myrinet. Both options have been tested for the CMS Full Event builder and showed sufficient performance for the requirements expected (Ref.[1]).

4 future and outlook

This section will give a short outlook to future developments by raising some of the following points.

- future techniques for front end data links and networking
- identification of possible R&D areas
- tools, adapt future developments, rules and guidelines

see separate document written by Patrick LeDu.

5 conclusion

This document is an attempt to give R&D groups an idea of the different readout layers, their functionality and interfaces. It is meant to help guiding towards a common readout system and to keep the efforts spent on the different issues at its minimum. In addition some areas were additional R&D on readout systems may be worth are shown. Some new technologies coming up in the next years may change the way a high energy physics experiment will be readout and its data being processed.

Given the complexity of the subject, the number of R&D groups and the time scales involved it is clear that this document cannot be the final word on the readout, it can only be a starting point and should be developed further with new ideas coming up in the next years. For this purpose a working group on 'data handling' is proposed which should keep this document up to date and encourage the R&D groups to use new technologies if available.

The working group should meet 3-4 times per year and a new release of this document or more detailed documents on specific parts should be written and discussed within this group. The group should also define and develop common tools and test systems, as well as try to develop and define standards.

In the coming 2-3 years a sufficient set of tools and test systems should be made available for the R&D efforts in the different groups.

References

[1] CMS, The TriDAQ Project, Technical Design Report Volume 2[CMS Collaboration] , CERN/LHCC 2002-26.

[2] http://hep.ph.liv.ac.uk/ green/lcfi/home.html

[3] http://polywww.in2p3.fr/flc/calice.html

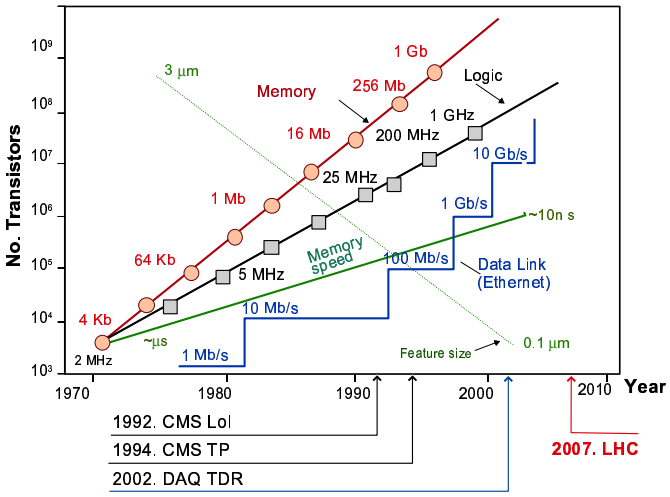

Figure 1: Evolution of computing technology (from CMS - TriDAS, TDR, Vol.2)

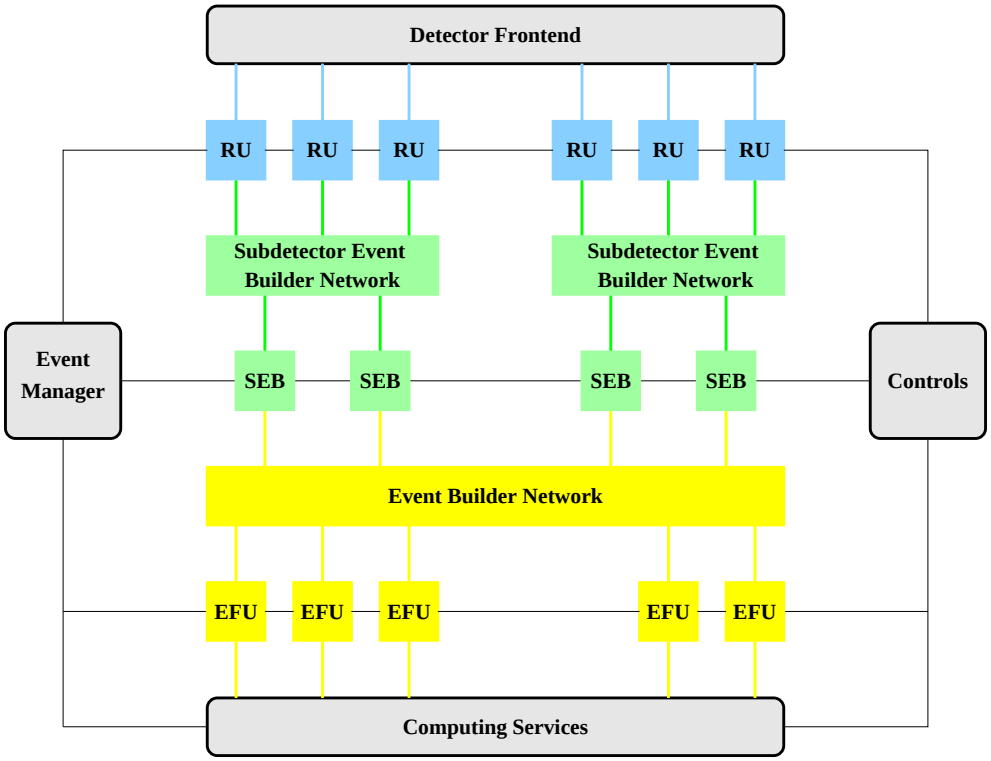

Figure 2: Functional block diagram of the DAQ system

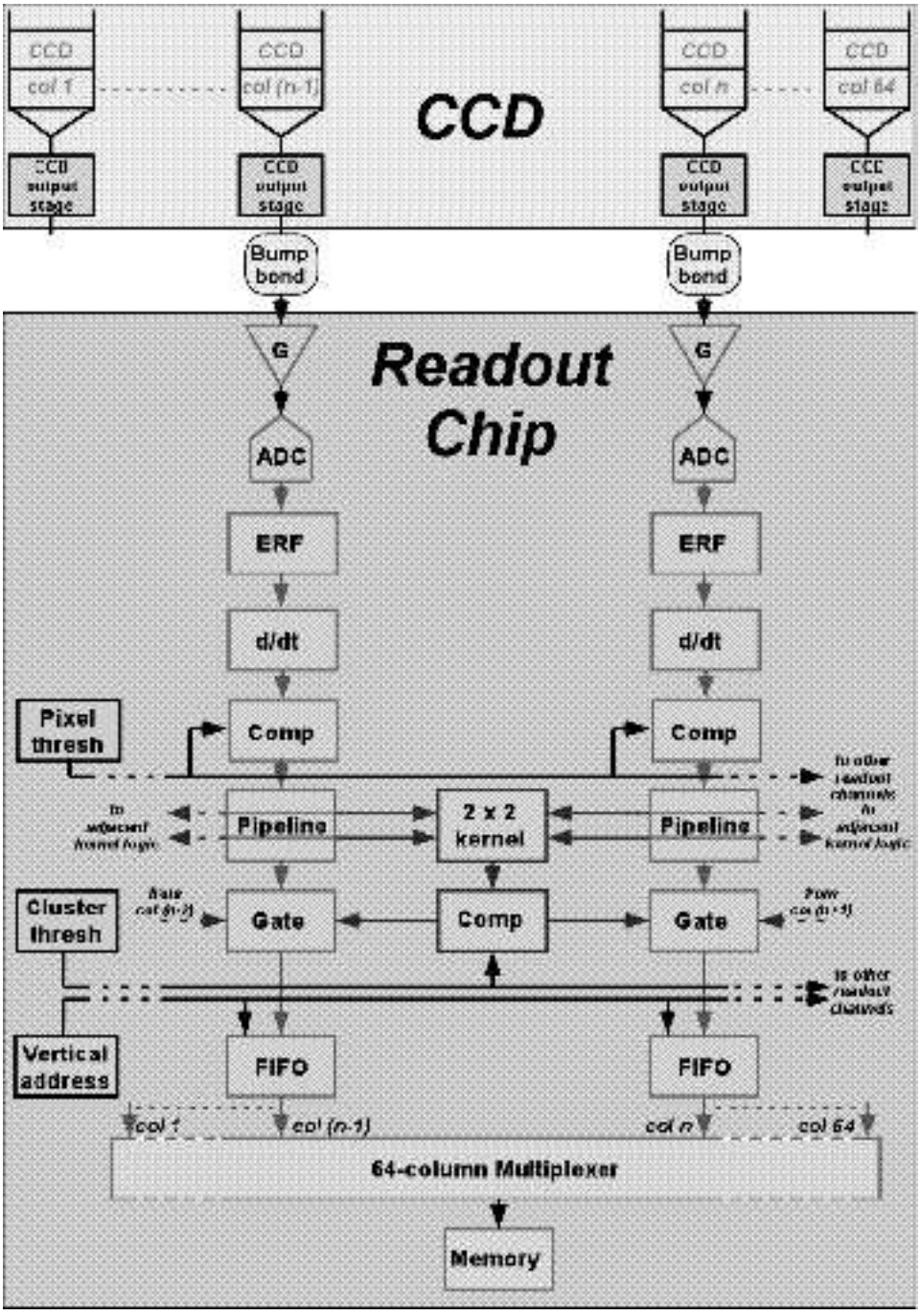

Figure 3: Vertex DAQ

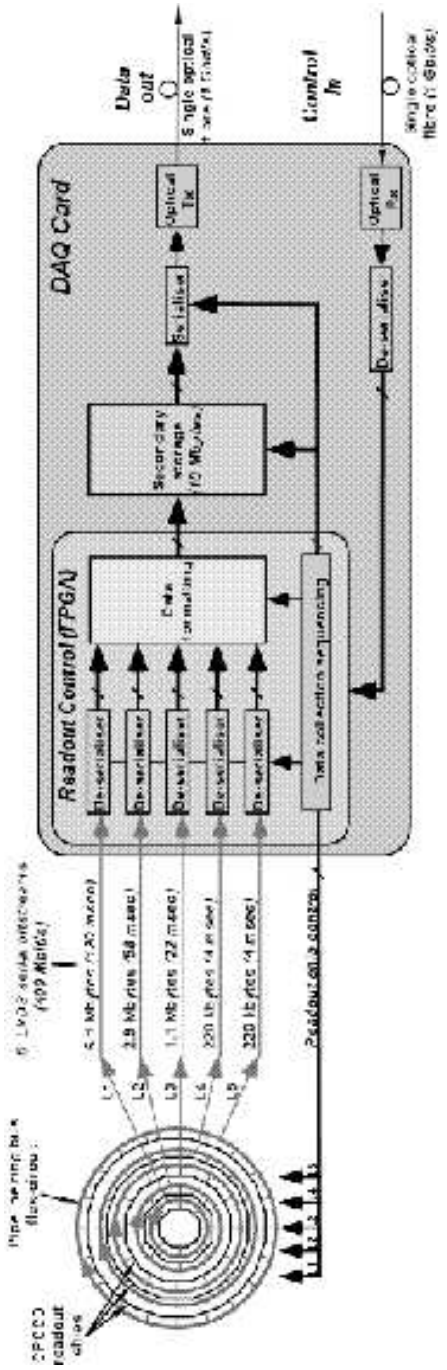

Figure 4: Vertex DAQ