FERMILAB-CONF-06-449-E

Available on CMS information server

CMS CR 2006/033

CMS Conference Report

15 June 2006

CMS DAQ Event Builder Based on Gigabit Ethernet

M. Pieri[1)], G. Maron[2)], A. Brett[3)], E. Cano[3)], S. Cittolin[3)], S. Erhan[3,4)], D. Gigi[3)], F. Glege[3)], R. Gomez-Reino Garrido[3,5)], M. Gulmini[3,2)], J. Gutleber[3)], C. Jacobs[3)], F. Meijers[3)], E. Meschi[3)], A. Oh[3)], L. Orsini[3)], L. Pollet[3)], A. Racz[3)], P. Rosinsky[3)], H. Sakulin[3)], C. Schwick[3)], J.Varela[3,6)], J. Branson[1)], S. Murray[7)], I. Suzuki[7)], R. Arcidiacono[8)], K. Sumorok[8)], V. Brigljevic[9)]

Abstract

The CMS Data Acquisition system is designed to build and filter events originating from approximately 500 data sources from the detector at a maximum Level 1 trigger rate of 100 kHz and with an aggregate throughput of 100 GByte/s. For this purpose different architectures and switch technologies have been evaluated. Events will be built in two stages: the first stage, the FED Builder, will be based on Myrinet technology and will pre-assemble groups of about 8 data sources. The next stage, the Readout Builder, will perform the building of full events. The requirement of one Readout Builder is to build events at 12.5 kHz with average size of 16 kBytes from 64 sources. In this paper we present the prospects of a Readout Builder based on TCP/IP over Gigabit Ethernet. Various Readout Builder architectures that we are considering are discussed. The results of throughput measurements and scaling performance are outlined as well as the preliminary estimates of the final performance. All these studies have been carried out at our test-bed farms that are made up of a total of 130 dual Xeon PCs interconnected with Myrinet and Gigabit Ethernet networking and switching technologies.

Presented at *CHEP06*, Mumbai, India, 13-17 February 2006

[1)] University of California, San Diego, California, USA

[2)] INFN - Laboratori Nazionali di Legnaro, Legnaro, Italy

[3)] CERN, Geneva, Switzerland

[4)] University of California, Los Angeles, California, USA

[5)] Universidad de Santiago de Compostela, Santiago, Spain

[6)] LIP, Lisbon, Portugal

[7)] FNAL, Chicago, Illinois, USA

[8)] Massachusetts Institute of Technology, Cambridge, Massachusetts, USA

[9)] Rudjer Boskovic Institute, Zagreb, Croatia

1 THE DATA ACQUISITION SYSTEM OF THE CMS EXPERIMENT

The CMS experiment is one of the 4 experiments that are being installed at LHC, the 14 TeV centre of mass energy proton-proton collider that is scheduled to start operating at CERN in the year 2007.

The DAQ system plays a key role in CMS. The beam crossing rate at LHC will be 40 MHz and it will be impossible to write all interactions to mass storage. The rate of events that will need to record for offline processing and analysis is of the order of 100 Hz. Therefore we must be able to select interesting events with a rejection power of O(10^5). This is achieved in two steps: a hardware Level-1 trigger which has a maximum accept rate of 100 kHz and a high level software trigger (HLT) with a rejection of O(10^3). In CMS all events that pass the Level-1 trigger are sent to a computer farm (Filter Farm) that performs physics selections, using the offline reconstruction software, to filter events and achieve the required output rate.

The design of the system is kept as modular as possible in order to expand the system when the luminosity increases and maintain the flexibility of implementing parts of the system when new technologies will be available or new requirements will be identified. The design of the CMS Data Acquisition System and of the High Level Trigger is described in detail in the Technical Design Report [1].

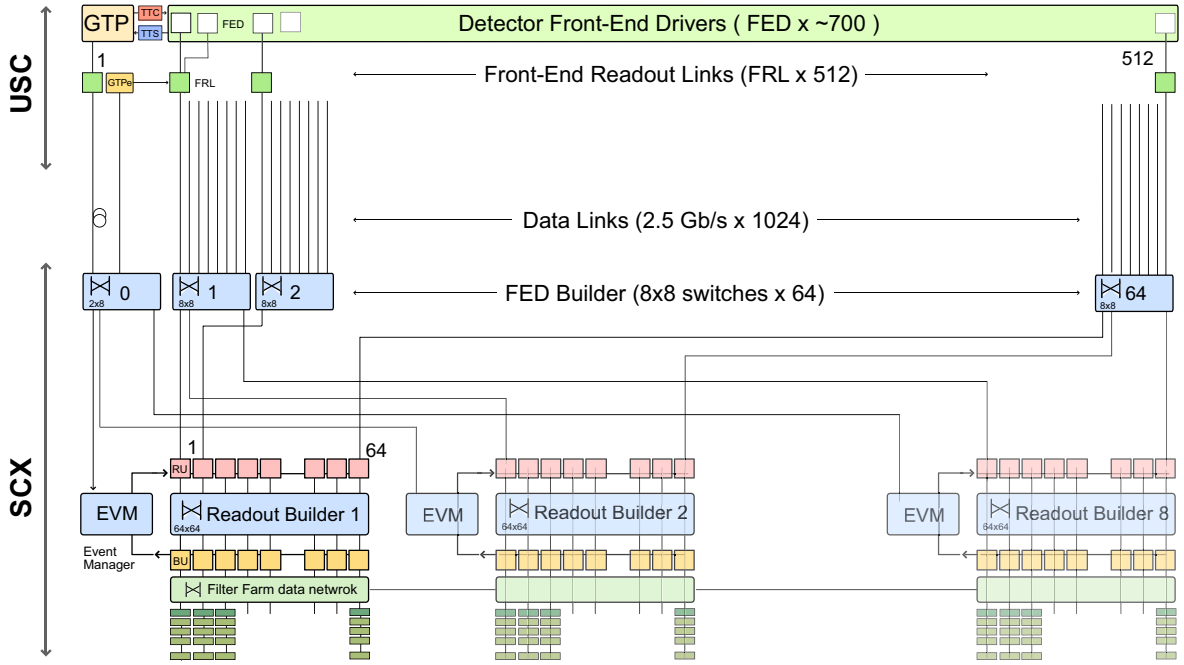

Figure 1: Sketch of the CMS DAQ system.

Figure 1 shows a simplified sketch of the CMS DAQ system. The data flow is from top to bottom in the figure. At the top are drawn the detector Front-End Drivers (FEDs) from which data are read into the 512 Front-End Readout Links (FRLs) that are able to merge the data of up to four FEDs together. The outputs of 8 FRLs are then pre-assembled in the FED builder and data are sent to one or more Readout Builders (RU Builder slices). The RU Builders are independent systems in charge of building the full events, performing the physics selections and forwarding the selected events to mass storage. The number of RU Builders in the system will grow up to 8 at the design luminosity of 10^{34} cm^{-2} sec^{-1} that will be reached after a couple of years from the start-up. Figure 2 shows a larger picture of one slice of the RU Builder and indicates its various components. In the baseline configuration a RU Builder is made up by 64 Readout Units (RU) and 64 Builder Units (BU) connected together by a switching network. An Event Manager (EVM) supervisions the data flow in the RU Builder. From the BUs all events are sent to the Filter Units (FU) through another switching network, the Filter Data Network (FDN). In the baseline configuration the number of FUs per RU Builder is 256.

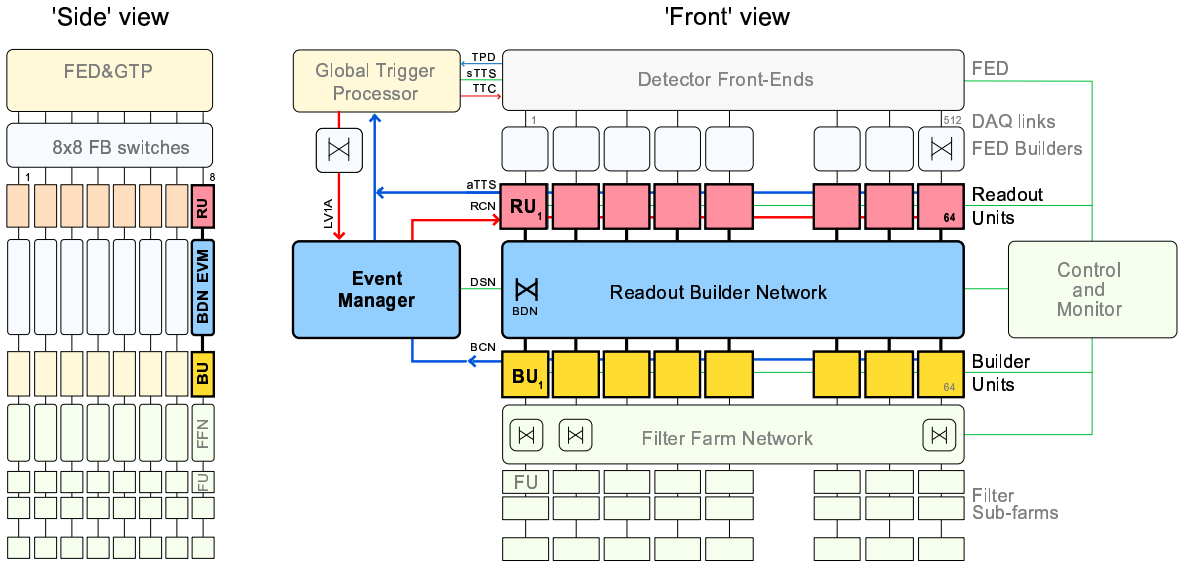

Figure 2: One Readout Builder slice.

2 EVENT BUILDER

At the design luminosity the Level-1 trigger accept rate is 100 kHz and the data size of events is approximately 1 MByte. Consequently the required performance of the RUs and the BUs is an in and out data flow of 200 MByte/s. The aggregate throughput from the detector to the Filter Farm is almost 1 Tbit/s and is very demanding in terms of networking.

2.1 Networking

At present there are two candidate network technologies for the data transfer in the event builder: Myrinet [2] and Gigabit Ethernet (GbE). They have some different characteristics: Myrinet has a data link speed of 2 Gbit/s instead of 1 Gbit/s of GbE, it has a lower latency, it implements hardware flow control and, while the Myrinet interface cards exploit the onboard CPUs, Gigabit Ethernet uses a lot of computing resources on the hosts when one uses a reliable protocol such as TCP/IP. Finally with Myrinet it is relatively easy to implement custom drivers in the on-board CPU.
For the FED Builder and the data transfer from the Front-End electronics, located in the underground cavern next to the detector, to the surface [3] Myrinet was chosen because, in addition to the aforementioned reasons, it provides fiber optic links and modest cost. The data transfer in the RU Builders can be based on Myrinet or GbE while the FDN should use Gigabit Ethernet because it will not be possible to equip all FUs with Myrinet network interfaces and also the FU input throughput is only $\sim$50 MByte/s and one inexpensive on-board GbE link is largely sufficient.

2.2 FED Builder

The FED Builder pre-assembles events fragments of an average size of 2 kBytes, coming from the FRLs, into larger fragments and distributes them to the multiple RU Builder slices. For random traffic the network efficiency is approximately 50%, due to head-of-line blocking. A throughput of 230 MByte/s per node is achieved for the nominal fragment size of 2 kBytes by using two Myrinet links. The output average fragment size is 16 kBytes but it could be increased, if necessary, to 32 kBytes by having 16$\times$16 FED Builders.

2.3 Readout Builder

For the RU Builder both networking options are still open. We have already shown that, by implementing a barrel-shifter type [4] traffic shaping, we were able to approach 100% link utilization with Myrinet [1]. With this option one Myrinet port on the RUs and BUs would be enough for the RU Builder task. The other option we are considering is the use of the TCP/IP protocol over Gigabit Ethernet. In this case, we connect the PCs in the

RU Builders with two links, referred to as 'rails' in the following, in order to reach the needed throughput of 200 MByte/s per node. We will show that already with present PCs we are able to satisfy this requirement.

3 TCP/IP OVER GIGABIT ETHERNET

The choice of TCP/IP over Gigabit Ethernet has the advantage of using completely standard hardware and software. TCP/IP is a reliable protocol and we do not need to worry about packet loss at the application level which typically occurs when operating close to wire speed. The main drawback is the considerable usage of machine resources for its operation.

In order to efficiently use TCP/IP in the Event Builder we had to optimize its use in the Linux operating system. The DAQ software is built upon the XDAQ [5] framework and the Asynchronous TCP/IP transport software package, ATCP[6], is part of it. ATCP has been developed to decouple the DAQ applications from the networking software. It also avoids the head-of-line blocking when more than one host is trying to send data to the same network interface of another host. In order to achieve good performances the following design choices were made:

i) ATCP puts all messages to be sent in different queues according to the destination and asynchronously processes them in another thread. To avoid blocking it writes(reads) into(from) a given socket until it blocks and as soon as it blocks it passes to another socket and continues.

ii) We use Ethernet Jumbo frames. By increasing the maximum transmission unit (MTU) from the standard 1500 bytes to 7000 or 8000 bytes we observe an increase in performance of approximately 50% for large fragments.

iii) We need to implement multi-rail operation. In principle it would be possible to use port bonding software such as the one provided by Linux or the Intel ANS teaming driver. Even if they seem to work well, we discarded this option because of the port bonding support requirement in the switches. What we do instead, is to use different physical networks depending on the source and destination hosts. For example in the RU-BU communication, we can send data from even hosts to even hosts through network A, from even hosts to odd hosts through network B, from odd hosts to even hosts through network B and from odd hosts to odd hosts through network A.

iv) We have different requirements when sending event data and control messages that steer the Event Builder operation, the latter need to be fast and have low throughput. There are two options in TCP/IP: to set the Nagle algorithm [7] on or off. When Nagle algorithm is on and there are not many messages in the pipeline in can happen that the message is delivered with a very large latency. On the other hand, when the Nagle algorithm is off we have observed an important decrease with time in throughput for a socket that sends small packets of variable size. The solution is to establish different connections for the data and control messages and to set Nagle algorithm on for data and off for control messages.

4 GBE EVENT BUILDER ARCHITECTURE

In this chapter we will review the performance of various GbE RU Builder architectures. All studies and integration tests for the DAQ system are carried out in the Pre-series system. It corresponds to approximately one of the eight slices of the final system and is composed of over 100 dual 2.6 GHz Xeon PCs. There are 64 PCs equipped with dual port Myrinet LANai-X and quad port Gigabit Ethernet NICs that can be operated as RUs or BUs and 16 PCs with only one additional on-board GbE interface that can be used as Filter Units or Event Manager. All of them are interconnected by means of a Force10 E1200 switch [8].

4.1 Baseline Configuration

In the baseline configuration in each RU Builder slice there are three layers of PCs: RUs, BUs and FUs (see Figure 2). The Readout Units have Myrinet input and GbE output, the BUs have GbE input and output and the FUs have only GbE input with a throughput of 1/4 compared to RUs and BUs. The Filter Unit output to mass storage is negligible. Given the relatively large consumption of CPU resources by TCP/IP compared to Myrinet, the most loaded part of the system in this configuration is the Builder Unit layer.
Figure 3 shows the results of tests in the baseline configuration. We consider three different configurations:

- 30 RUs ×30 BUs where the Builder Units discard the events as soon as they are built;
- 12 RUs ×12 BUs with all Builder Units sending data to 4 Filter Units each;
- 30 RUs ×30 BUs with 4 of the BUs connected to 4 FUs each while the others discard the events as soon as they are built. In this case the througput is measured separately for the BUs discarding events and those that send events to the Filter Units.

A throughput on the BUs of ≈170 MByte/s at the nominal fragment size of 16 kBytes is reached. We also verified that with faster PCs (3.2 GHz CPUs) we have an increase in the throughput of approximately 30%, proportional to the CPU speed increase. Further improvements are expected from the dual core CPUs that are now available.

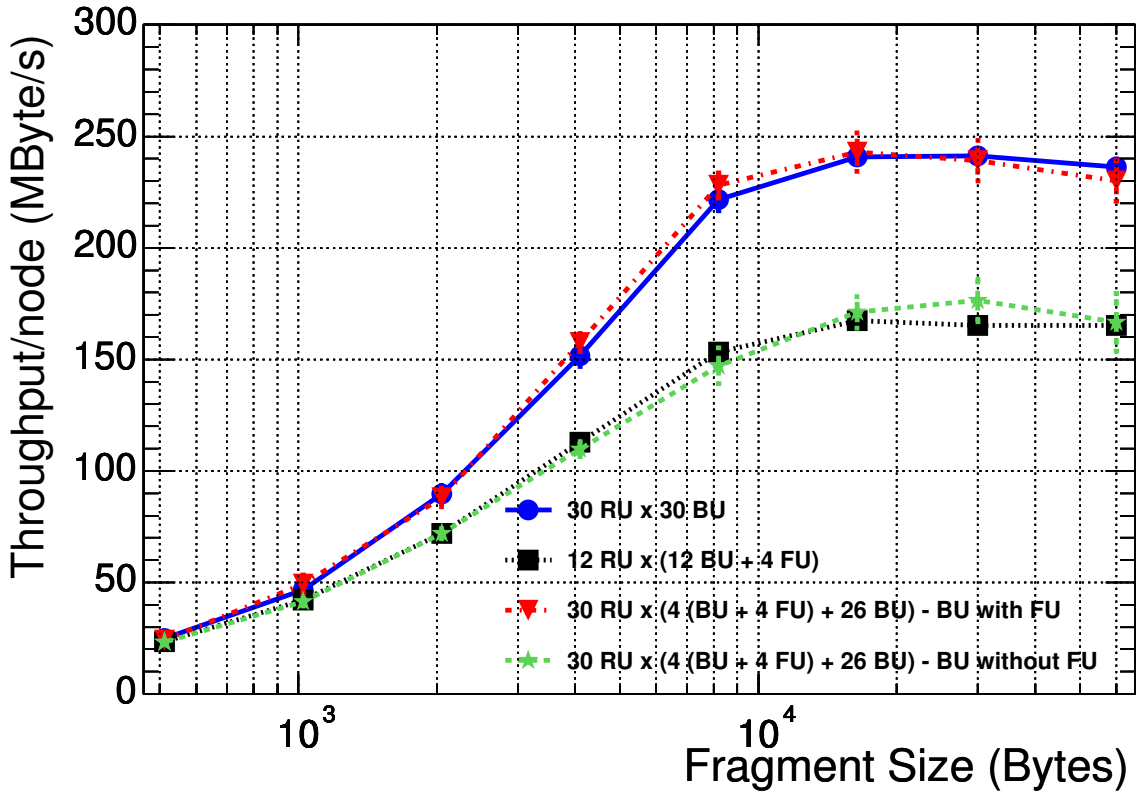

Figure 3: Throughput per node as function of the fragment size for the baseline RU Builder.

4.2 Folded Event Builder

In addition to the baseline configuration, in the DAQ TDR [1] the configuration where a RU and a BU operate on the same PC was considered ('folded' configuration). In this case only 64 PCs would be needed for the RUs and the BUs, the full duplex features of links and switches would be exploited and the throughput on each PC would be doubled. The total design throughput in each node should be 200 MByte/s Myrinet input plus 200 MByte/s GbE input and 400 MByte/s GbE output. We performed some tests of the folded RU Builder but in case of Gigabit Ethernet the throughput is still too low because of CPU limitations. Nevertheless we were able to verify the scaling of the ATCP transmission up to 60 RUs x 60 BUs, which is almost the final size of the RU Builder. Without input to the RUs and output from the BUs an in and out throughput of ≈180 MByte/s per node at the nominal fragment size of 16 kBytes was reached which is consistent with the measurements in the baseline configuration, assuming that the limitation comes from the aggregate TCP throughput per host, i.e. ∼350 MByte/s.

4.3 Trapezoidal Configuration

Considering the fact that the Builder Units are the most loaded part in the GbE RU Builder and that most of their load is related to receiving and sending data with TCP/IP, we could make a modified trapezoidal configuration for

the RU Builder, sketched in Figure 4.

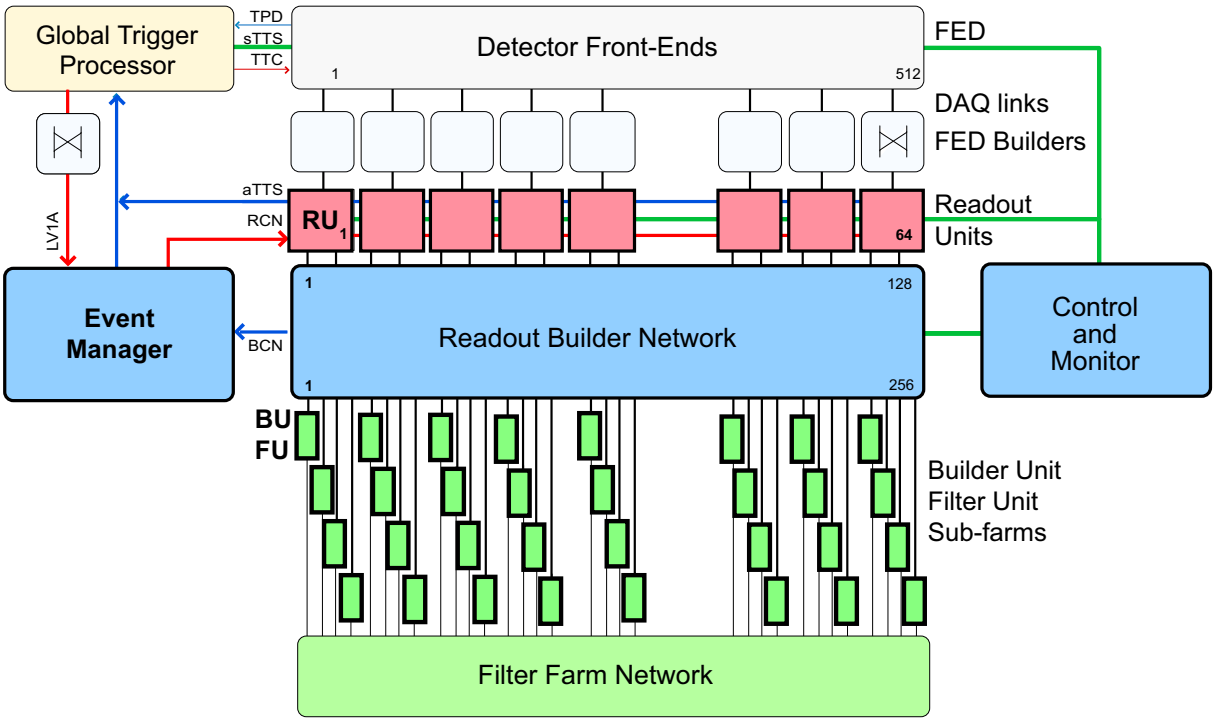

Figure 4: Sketch of the trapezoidal RU Builder.

The Builder Unit PCs layer, visible in Figure 2, is removed, the 64 Readout Units are directly connected over two rails with all the 256 Filter Unit PCs and events are built in those same PCs, referred as BU-FU. The number of connections from the RUs increases from 64 to 256 but the throughput on the BU-FUs and therefore on the switch output ports is smaller. This also reduces the requirements for the switch performance and allows for example the use of oversubscribed line-cards. The most loaded part of the system in this configuration is the Readout Unit layer.

With the pre-series system it is possible to test almost one quarter of the final RU Builder slice, a 16 RU $\times$ 60 BU-FU configuration.

Figure 5 shows the results. We reach a throughput on the RUs of 240 MByte/s for the nominal fragment size of 16 kBytes, with a network utilization of almost 100%.

To approach the full scale test in terms of number of connections from a RU to the BU-FUs we run multiple BU-FU applications (up to 4 per node). By doing this we obtained similar results up to a 16 RU $\times$ 240 virtual BU-FU configuration.

These results show that there is no or limited decrease in performance when increasing the number of output sockets from RUs to BU-FUs. The system scales with the number of BU-FUs and a trapezoidal RU Builder should satisfy the design throughput requirements.

In this trapezoidal configuration, the task of building the events is carried out by the Filter Units PCs and this slightly reduces the CPU resources available for running the HLT algorithms. On the other hand, this load is rather small and we verified that, for an input throughput of 50 MByte/s, the FUs only use about 5-10% of their CPU to build the events.

5 SUMMARY

We reviewed the requirements and the performance of the CMS two-stage DAQ Event Builder. By using TCP/IP over Gigabit Ethernet in the second stage of the Event Builder it is possible to achieve the design performance. In the trapezoidal configuration we are able to saturate 2 GbE links in our test-bed system and achieve a throughput of 240 MByte/s per node at the nominal fragment size of 16 kBytes. This is obtained using different sockets with different options, static multi-rail communication and jumbo frames.

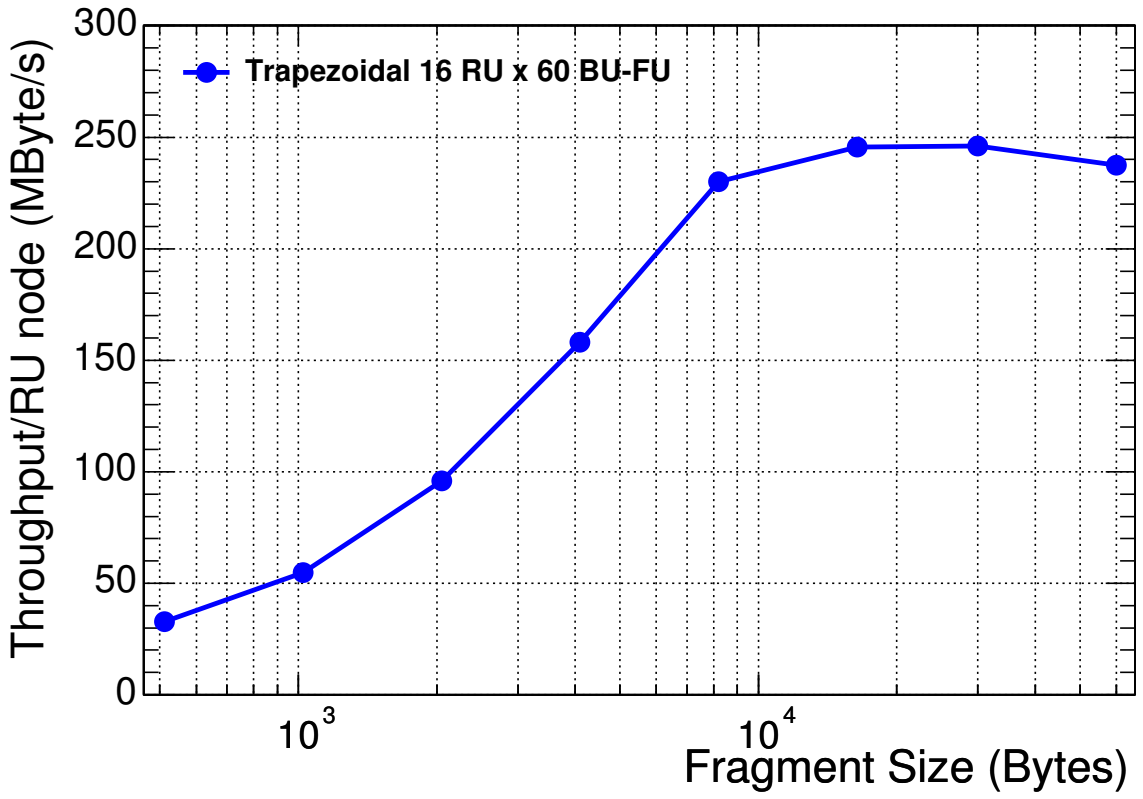

Figure 5: Throughput per Readout Unit as function of the fragment size for the trapezoidal RU Builder.

We are now in the process of finalizing the RU Builder design and we are carrying out tests of PCs in view of the purchase of the final hardware that should be ordered in summer 2006.

We will then install and commission the DAQ system that should be ready for data taking in summer 2007.

References

[1] CMS Collaboration, CERN/LHCC/2002-26, CMS TDR 6.2, 15 December 2002.

[2] Myrinet products from Myricom, Inc. Arcadia, CA, USA, see http://www.myri.com.

[3] R. Arcidiacono et al. "The 2 Tbps Data to Surface System of the CMS DATA Acquisition", 14^{th} IEEE-NPSS Real Time Conference, 2005, Stockolm, Sweden.

[4] E. Barsotti, A. Booth and M. Bowden, "Effects of Various Event Building Techniques on Data Acquisition System Architectures", Santa Fe Computing 1990:82-101.

[5] V. Brigljevic et al. "Using XDAQ in Application Scenarios of the CMS Experiment", Computing in High Energy and Nuclear Physics, 24-28 March 2003, La Jolla, USA.
See also http://xdaqwiki.cern.ch

[6] See http://xdaqwiki.cern.ch/index.php/Ptatcp

[7] R. Stevens, "Advanced Programming in the UNIX Environment", Addison Wesley, 1992.

[8] See http://www.force10networks.com