Simulation of the ALICE DAQ System with GALSIM

Jianwei Yuan, Bin Wu, Bernhard Skaali

Dept. of Physics, Univ. of Oslo, P.O.Box 1048, Blindern, 0316 Oslo, Norway

In this paper, we investigate some important issues in the DAQ system of the ALICE experiment at the Large Hadron Collider with the graphic simulator GALSIM. We concentrate on the simulation of the ALICE DAQ topology with various parameters such as the effect of message latency, the link speed. The simulation results of different models suggested for the ALICE DAQ System will also be presented.

1 INTRODUCTION

The Heavy Ion Experiment ALICE is one of the experiments at the LHC accelerator at CERN[1]. Modeling and simulation of the large scale ALICE Data Acquisition (DAQ) system is necessary in order to study the behavior of the system in detail. Therefore, a simulation package – ALSIM[2] has been developed at University of Oslo and CERN. For a better understanding of simulation work of ALICE DAQ System, an advanced version of ALSIM with graphic presentation/animation fea- tures is built up. It is called GALSIM[3]. The goals of GALSIM are to investigate the feasibility of the current ideas on the DAQ architecture, and to lay a basis for evaluating the evolution of the system architecture[2]. Furthermore, GALSIM provides a general understanding of the total system behavior and the system components, and how the physics and readout requirement influence the design of the system. It also gives us guidelines for further investigations and prototype developments for choosing the right technology for interconnects and switches, such as ATM, Fibre Channel, SCI, etc. The general architecture is performing front-end data readout, event building, data reduction and data storage. The DAQ system shall handle three estimated trigger rates; Central and Minimum Bias, both of about 40 events/second of size between 33 and 39 MBytes and between 0 and 39MBytes respectively, and Dimuon type of 1000 events/second of a size around 250KBytes [2].

This paper mainly describes the simulation results with consideration of various system parameters. We will describe the details of the software package

Preprint submitted to Elsevier Preprint *17 July 1997*

modeling environment in section 2. Section 3 investigates simulation results of different system parameters and various system topologies. The simulation of various models proposed for the DAQ System is introduced in section 4. Finally, we will conclude our simulation work.

2 MODELLING WORK

2.1 General Architecture of the ALICE DAQ System

The general architecture of the ALICE DAQ system is shown in Figure 1. The raw event data are saved in the Front End Crates (FEC) with required detector

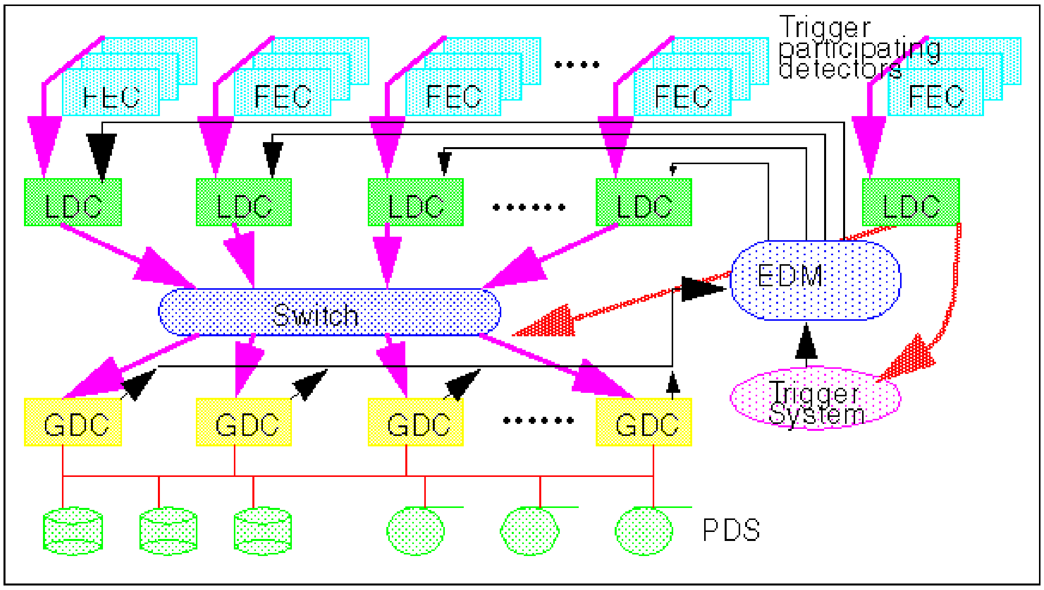


Fig. 1. Proposed general architecture of the ALICE DAQ system

granulation. These data are collected by the Local Data Concentrators (LDC) which can be implemented by dual-port memories and then pushed through the event builder–the switch. The events are finally received by Global Data Concentrators (GDC) and stored in Permanent Data Storage devices (PDS). The switch is a generic model which is described by point-to-point links. An Event Destination Manager (EDM) model controls the whole system. The size of the system and the details inside each model is defined in an ASCII configuration file.

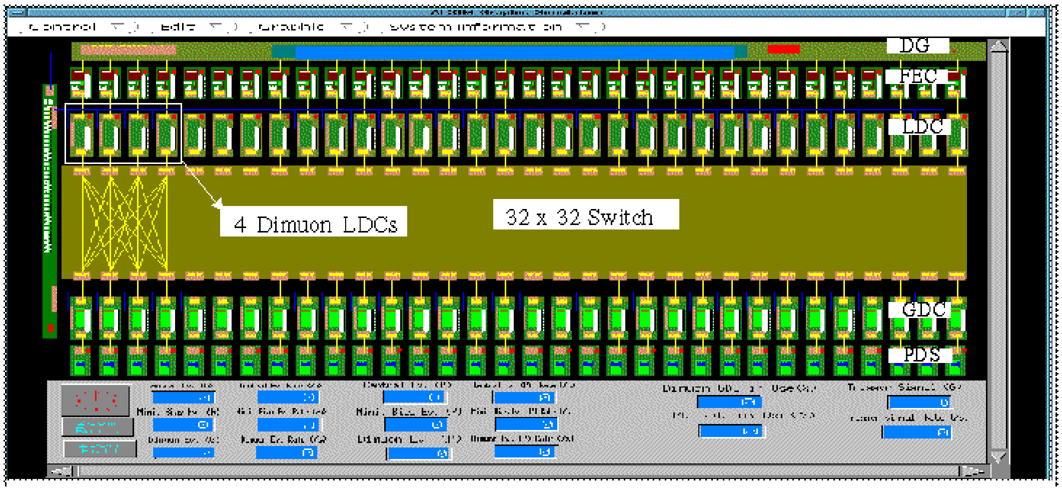


Fig. 2. The full view of the main simulation window of 32 LDC by 32 GDC

2.2 General Introduction of GALSIM

2.2.1 Objective-oriented Approach

GALSIM follows an object-oriented approach which provides a flexible and open-ended simulating environment. General speaking, like ALSIM, there is a simulation object defined in GALSIM for each of the components shown in Figure 1. In addition, a Data Generator (DG) that generates events and a graphic object representing the data message flowing on the links during the simulation are also created in GALSIM.

2.2.2 Simulation procedures

GALSIM accepts a system description in ASCII format as the input file which has already been pro- cessed by the pre-processor. The simulation results can be either dynamically shown in windows by sev- eral kinds of graphs or in ASCII format which will be transformed into input that suits many spreadsheet programs (e.g Microsoft Excel) or the CERN PAW data presentation package [4], as shown in Figure 3.

Figure 2 shows a graphic simulation environment with 32 LDCs and 32 GDCs.

Two types of events, DataTPC and Dimuon, will be generated in DGGraphicObj triggered by Central, Minimum Bias and Dimuon signals respectively. Correspondingly, LDCGraphicObjs are divided into two groups by which events they deal with, Dimuon or DataTPC. In our example of Figure 2, the number of Dimuon LDCGraphicObjs are 4 (displayed in a way that number of ports in the SwitchGraphicObj are crossed-connected).

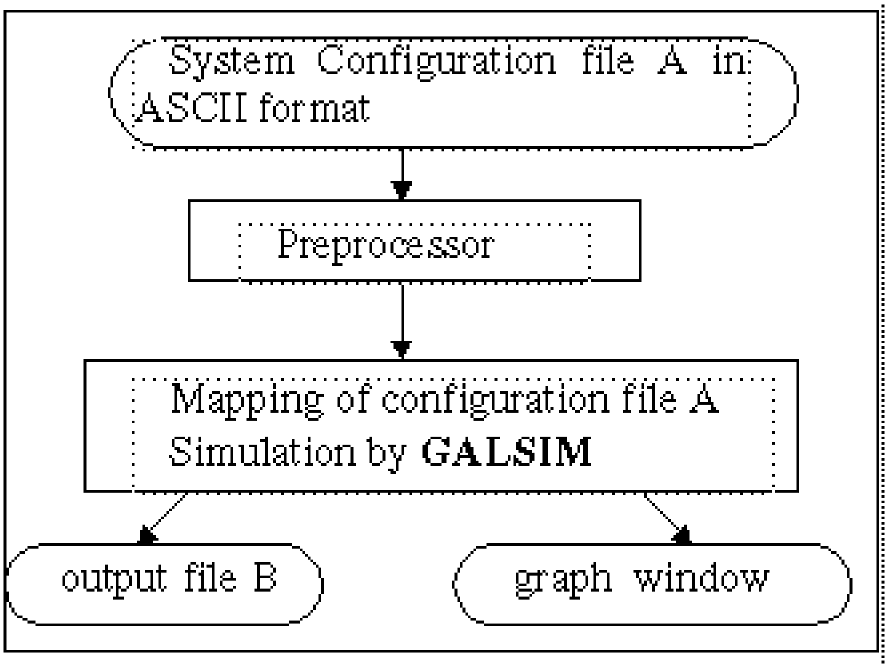


Fig. 3. The block diagram of GALSIM simulation environment

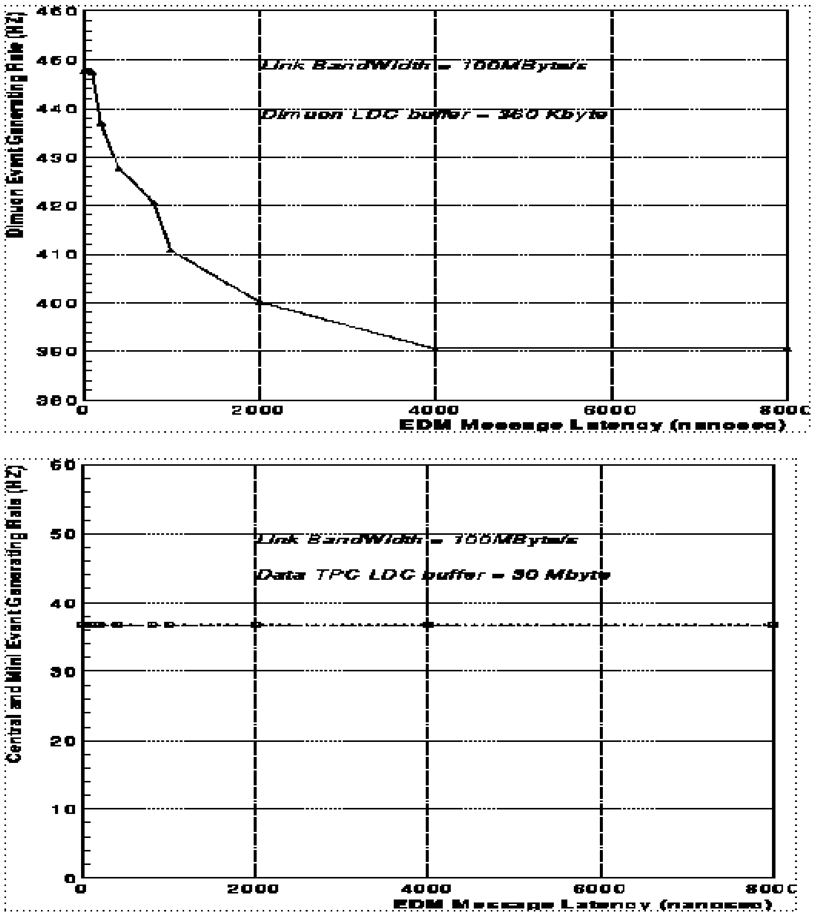
3 SIMULATION RESULT ANALYSIS

One of the most important purposes of the simulation is to test if the system configuration can match the data-assembling capacity request, and modify some system parameters to get a better result.

3.1 The Effect Of Information Message Latency

The data read-out begins with FECs reading data from sensor, which is simulated by DG producing data event and sending the split event (message) to FECs in GALSIM. After generating an event, DG will signal EDM by a trigger signal that corresponds to the type of event DG generated. EDM will then trigger each LDC to ask for data from the corresponding FEC. After finishing trigger process, EDM will check if there is any free GDC left. In case of yes, EDM will send a message, which contains the information of destination for sub-event, to each LDC. In real circuit design there are many ways to implement this message transfer technology, which will result in different message latencies. For example, the hardware point-to-point interconnection only has a latency of tens of nano seconds. Message transported through Operative System could bring us a latency of at least several micro-seconds since it will ask software to accomplish this task. In GALSIM, we will test the system performance at different values of the latency of the messages from EDM to

LDCs, using the same configuration for the system as in the last section. The simulation results are shown in Figure 4.

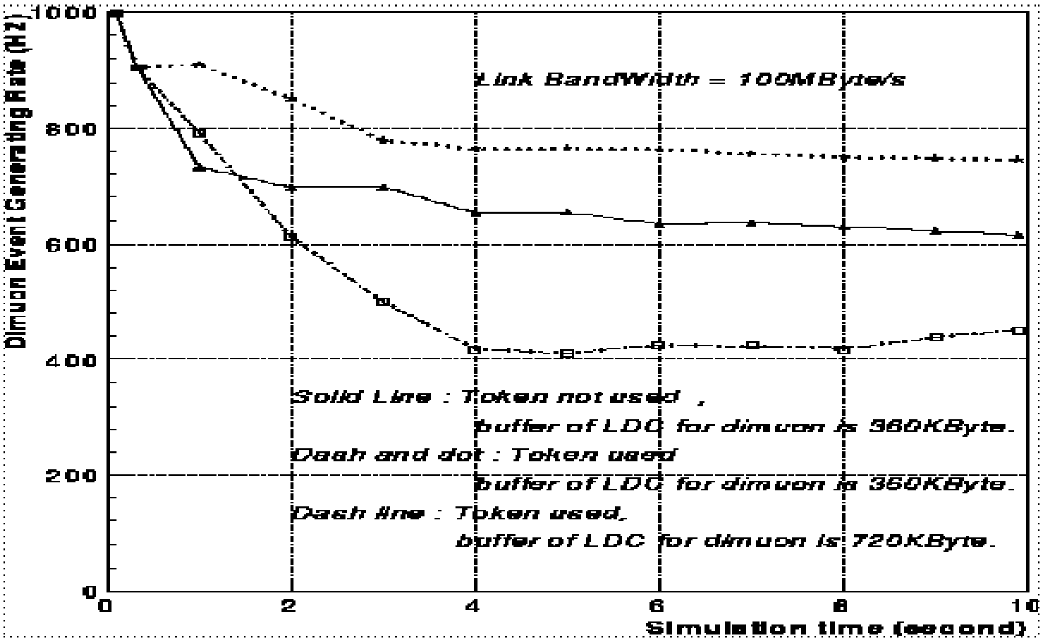


,

Fig. 4. The effect of EDM message latency on the rates of production for two different events

It is clear that higher EDM message latency will restrict Dimuon event generating rate. On the other hand, Central and Min. Bias events' rates will change little as the EDM message latency, even though as high as 8 micro seconds, is much lower compared with the time the system needs to consume such events of big sizes. So to catch up with a required Dimuon event rate, one needs to lower the EDM Message Latency to a certain level, as in the figure.
As dimuon events are very sensitive to system performance due to its high rate, the simulation results will concentrate on Dimuon events in the next of the paper.

For the event building in a large DAQ system, all the data that belong to the same event will have to be sent to the same data processor from LDCs. Tokens can be used to coordinate the data transmission among the LDCs in the system. It is especially useful in reducing traffic density which is very critical for certain modern switching technology like SCI. However, tokens may influence the system balance in term of buffer usage and traffic pattern. In GALSIM, the simulation results can be monitored.

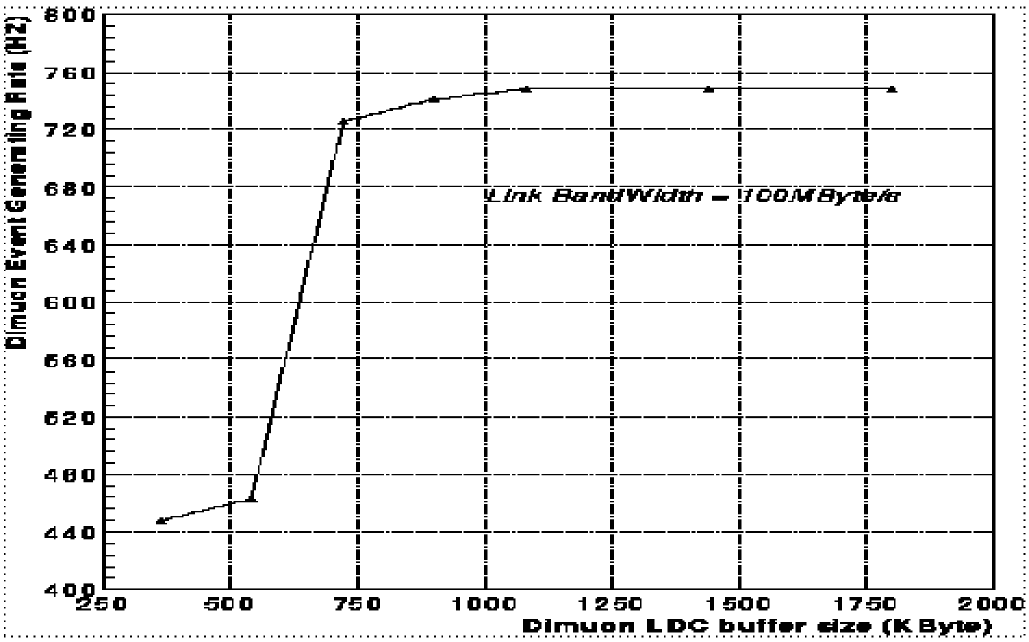


,

Fig. 5. Dimuon event generating rate under different traffic patterns and LDC buffer sizes.

Figure 5 presents the dimuon event generating rate as a function of system operation time under different traffic patterns (token used or not) and various LDC buffer size. From the figure, we can observe that the system is operated in an unstable status in the beginning of the simulation and becomes stable after about 4 seconds. Dimuon event producing rate is greatly improved if tokens are not used in the system. However, benefiting from industrial advancement, memory becomes cheaper and cheaper, the trade-off of some extra memory for adding token in the system is relatively acceptable. If we double the buffer size of LDC, the Dimuon event generating rate is greatly upgraded, higher than that of nontoken situation, as shown in the figure.

we may get an impression from Figure 5 that in order to increase the performance of the system, what we need do is just to plug some more memory cards into machines. Nevertheless, this is not the situation. Figure 6 shows the curve of Dimuon event generating rate against the buffer sizes of Dimuon LDCs during the ten-second simulation.



,

Fig. 6. Generating rate of Dimuon event as function of Dimuon LDC buffer size

It is very clear that buffer size of 360 KBytes will have poor performance in the figure. Sizes beyond 900 KBytes do not enhance the performance. By GALSIM, we find out that more and more subevents will only be accumulated in the LDC buffers, waiting for being transmitted. Meanwhile, the ports of LDCs connected to the switch continuously send data out while all GDCs for Dimuon events are already reserved or occupied (Figure 7).

Apparently, there is a "bottleneck" between LDCs and GDCs in the network. After investigation, we find that the relatively low link speed (including both bandwidth and setup delay) causes such problems. If we using a relatively high speed link in the system, the problem can be easily solved.

In GALSIM, we run the simulations of different Dimuon LDC buffer sizes at various link bandwidths, the simulation results are shown in Figure 8. When link bandwidth is increased to 110Mbyte/second, the generating rate of Dimuon events will be 1000 Hz even though the buffer size of Dimuon LDC is as low as 360 KBytes.

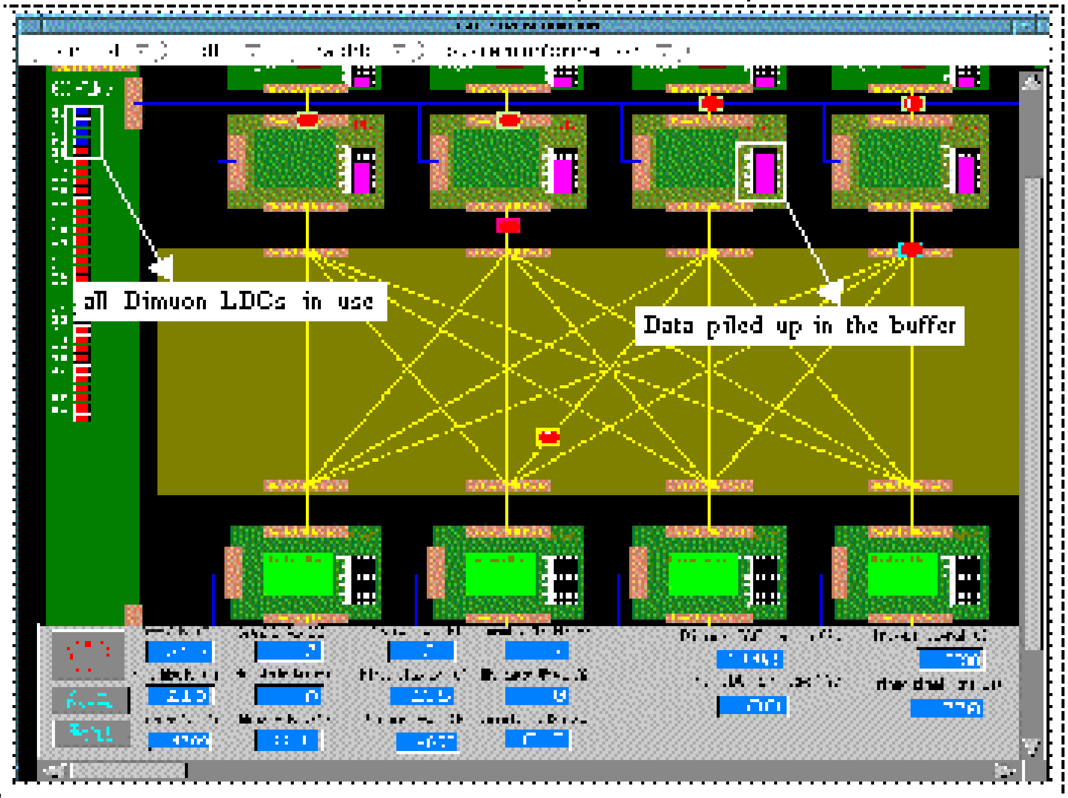


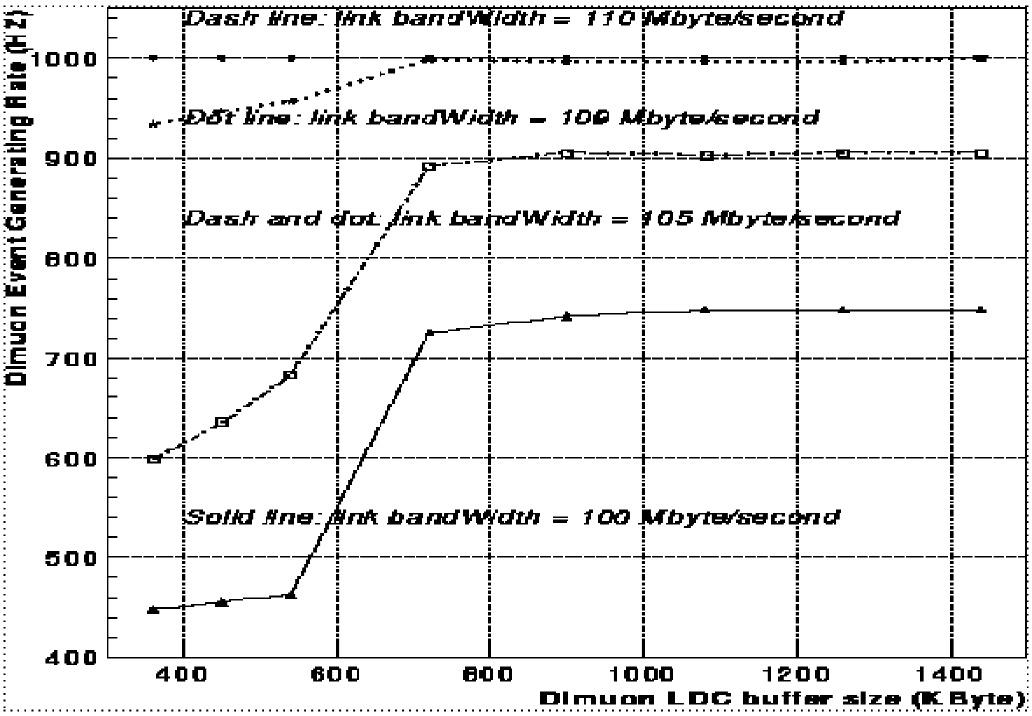
Fig. 7. The graphic presentation of dimuon part during a simulation in GALSIM of Dimuon LDC buffer size = 900KMyte

The advantage of using high speed link is not only to make up for short of buffer, but also to lessen the effect of latency of message from central control unit to other units (EDM to LDC message latency in ALICE DAQ system). Figure 9 gives us the curve of Dimuon event generating frequency as a function of EDM message latency at link bandwidth of 200 Mbyte/second.

We see (from comparing Figure 4 and Figure 9) that high speed links help to ease other restrictions that might impose on the system. One can either choose high speed links and relatively long message latency or relatively lower speed links and short message latency.

4 SIMULATION OF DIFFERENT MODELS FOR ALICE DAQ SYSTEM

Using high speed link will not only compensate for the relatively small local buffer but more importantly it can reduce the system size. As the cost of a DAQ system will depend not only on the price of each device but on the size of the whole system, the expense on the DAQ system can be greatly reduced



,

Fig. 8. Dimuon event generating rates as a function of LDC buffer sizes

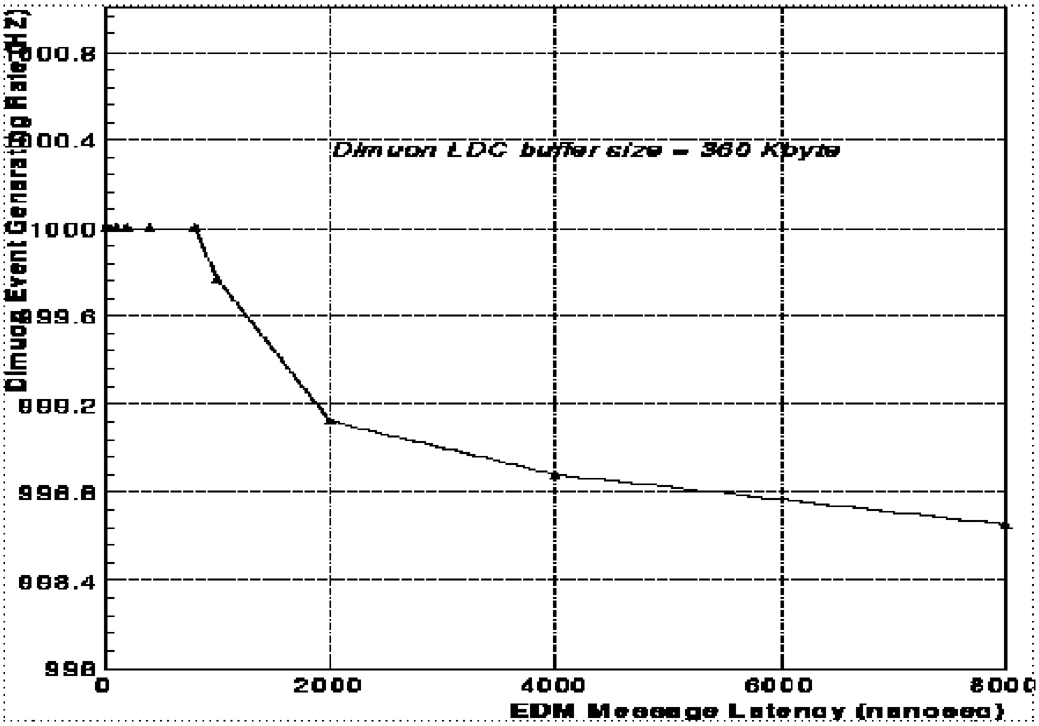
if we could minimize the system scale. In addition, control and debug can be more easily conducted on a small system. On the basis of the configuration for ALICE DAQ system given in Section II we will run simulation of several models, defined by the number of LDCs used in the system, using different links.

Model Definitions

Model A: The total number of LDCs used in the system will be 16, among which first four handle Dimuon events.
Model B: The total number of LDCs used in the system will still be 16, but first two dealing with Dimuon events.
Model C: The total number of LDCs are reduced to 8, still 2 of them are responsible for Dimuon Events.
Model D: The total number of LDCs are set to 4, still 2 of them are responsible for Dimuon Events.

Simulation Results Analysis

Just like simulation we run before, we will test the event rate as a function of link speed on the four models. The simulation results are given in Figure 10, and the simulation results of 32 LDCs by 32 GDCs with different link speeds have been illus- trated in Figure 8, called model E.



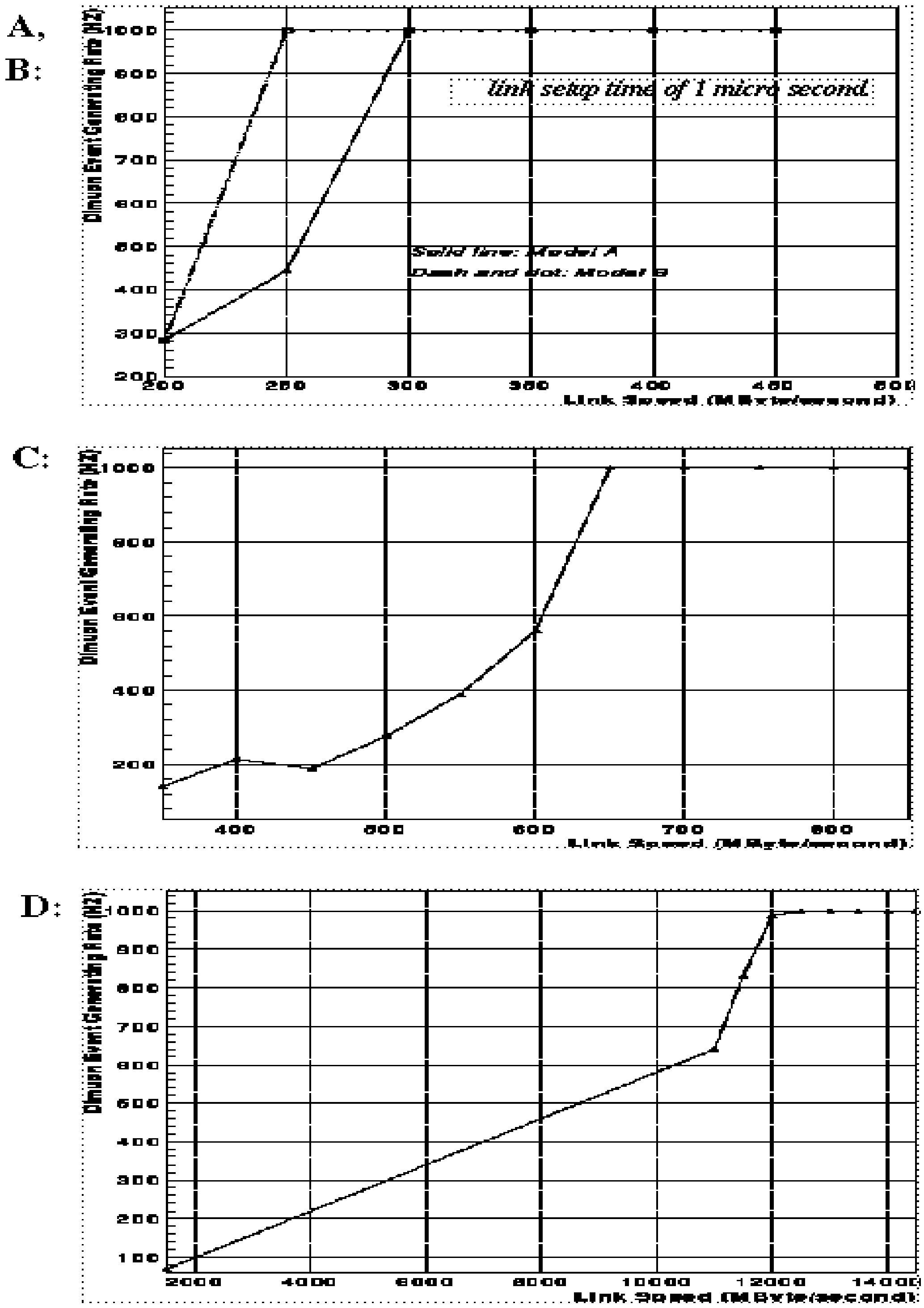
,

Fig. 9. Dimuon event generating rate versus EDM message latency

To achieve an event rate of 1000 Hz for Dimuon event, apart from model E, model B can also be a reasonable choice. Model B requires about 250 MByte/second data links while Model A requires about 300 MByte/second with the same size of system. Therefore model B is a better choice than model A. Model C is a smaller in size, but requires 650 MByte/second links which are on the cutting edge of today technology. We could not use model D until link can provide a speed of 12GBytes/second which is far beyond the current technology.

5 CONCLUSION

As presented in the paper, it is very important to use high speed links in the switch of a DAQ system since it can improve the system performance and put less constraints on other system parameters such as latency of message. With simulation by GALSIM, we can easily find the possible optimal system setup. If permitted, large memory may be installed in the local data collection equipment so as to improve the system performance.



,

Fig. 10. Dimuon events generating frequency as a function link speed for the four models.

6 ACKNOWLEDGMENT

The authors gratefully acknowledge the help from colleagues of ALSIM Modeling Group.

Jianwei Yuan is a post-graduate student at the Uni- versity of Oslo.

7 REFERENCES

[1] ALICE Collaboration, "Letter of intent for A Large Ion Collider Experiment", CERN/LHCC/ 93-16, CERN, March 1, 1993

[2] Wu, B., Yuan, J., Kvamme, B., Skaali, B., "ALSIM-Proposal for an ALICE DAQ Modeling and Simulation Project", ALICE Note 1994-34, CERN, Dec. 1994

[3] J. Yuan, B. Kvamme, B. Wu and B. Skaali, "A Graphic Presenter for A Large DAQ Simulation", 2nd International DAQ Conference, Osaka, Japan, Nov. 1996

[4] "PAW, the complete reference, version 1.14", CN Division, CERN, July 1992