Fermi National Accelerator Laboratory

Overview and Performance of the FNAL KTeV DAQ System

T. Nakaya and V. O'Dell

Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510

M. Hazumi and T. Yamanaka

Osaka University
Toyonaka, Osaka 560, Japan

November 1995

Disclaimer

OVERVIEW AND PERFORMANCE OF THE FNAL KTEV DAQ SYSTEM

T. NAKAYA, V. O'DELL

Fermi National Accelerator Laboratory, Batavia,
IL 60510, USA

M. HAZUMI, T. YAMANAKA

Osaka University, Toyonaka, Osaka 560, Japan

KTeV is a new fixed target experiment at Fermilab designed to study CP violation in the neutral kaon system. The KTeV Data Acquisition System (DAQ) is one of the highest performance DAQ's in the field of high energy physics. The sustained data throughput of the KTeV DAQ reaches 160 Mbytes/sec, and the available online level 3 processing power is 3600 Mips. In order to handle such high data throughput, the KTeV DAQ is designed around a memory matrix core where the data flow is divided and parallelized. In this paper, we present the architecture and test results of the KTeV DAQ system.

1 Introduction

KTeV consists of two fixed target experiments: FNAL E799 and E832. E799 is a search for CP violating rare K_L decays, such as $K_L \rightarrow \pi^0 e^+ e^-$, with a single event sensitivity of 10^{-11}. E832 is a dedicated experiment to measure the direct CP violation parameter ε'/ε in $K_{L,S} \rightarrow \pi\pi$ decays with a precision of 10^{-4}. Both experiments require more than an order of magnitude increase in statistics over current measurements in order to achieve the above sensitivities. Hence, a higher kaon flux and a DAQ that can cope with the increase in data rate are key for our experiments to succeed.

The Fermilab Tevatron delivers 800 GeV protons in a 23 second period, known as a "spill", and there is one such spill every 55 seconds. In KTeV we use a three level triggering scheme to cope with the high rates during the spill, where levels 1 and 2 are comprised of specifically designed hardware, and Level 3 is a software filter running on multi-processing SGI Challenge L desk-side servers. The Level 2 trigger rate is expected to be 20kHz during the spill, with an average event size of 8 kbytes, therefore, the average data flow into the DAQ during the spill is about 160 Mbytes/sec [1]. The KTeV DAQ system, which is part of the Fermilab DART project, is designed to meet the above requirements.

2 Architecture of KTeV DAQ

The KTeV DAQ can be visualized as a matrix in which each element corresponds to a memory node in VME for event storage (see Figure 1). The data from an entire spill are stored in these VME memories, which allows us to use the full spill cycle time to read out, build, filter and log events. In order to buffer data from an entire spill, the total amount of memory required is 160Mbytes/sec × 23sec(on-spill) =

6 Streams

VME | Memory | Memory | Memory | Memory | Memory | Memory → SGI Challenge

VME | Memory | Memory | Memory | Memory | Memory | Memory → SGI Challenge

4 Planes

VME | Memory | Memory | Memory | Memory | Memory | Memory → SGI Challenge

VME | Memory | Memory | Memory | Memory | Memory | Memory → SGI Challenge

RS485 lines

1st Event | Sub-event 1 | Sub-event 2 | Sub-event 3 | Sub-event 4 | Sub-event 5 | Sub-event 6
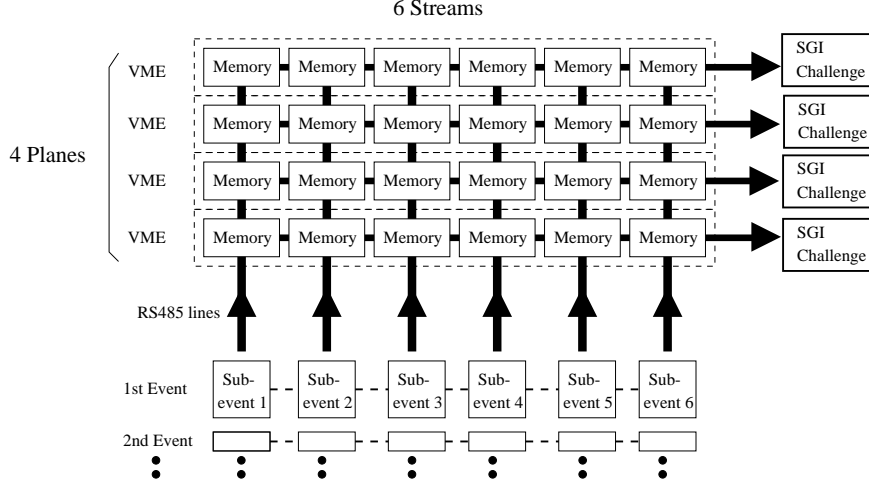
2nd Event

Figure 1: Conceptual Design of the KTeV DAQ

3.68Gbytes/spill. The data path between the front end crates and the DAQ are RS485 lines whose bandwidth is 40 Mbytes/sec/line. We use 6 RS485 lines, or 6 streams, so that we can achieve an average onspill bandwidth of 40Mbytes/sec $\times$ 6streams = 240Mbytes/sec. The 6 sub-events on 6 streams flow asynchronously into the memory nodes which are resident in a singe VME crate and thus linked via the VME bus. Each crate of VME memories is connected to the internal VME bus of the SGI Challenge machine, forming what we call a "plane". We use 3 planes for level 3 filtering and an additional plane for on-line monitoring and calibration. Each event is sent to a specific plane or planes depending on its trigger type, and the buffered events are read out, built and filtered in parallel in all four planes. Since we can readout events during the entire spill cycle (55 seconds), the data flow per plane is 160Mbytes/sec $\div$ 3(filtering planes) $\times \frac{23(\text{on-spill})}{55(\text{spill cycle})} = 22$Mbytes/sec/plane, which is well within the VME64 specification of 40Mbytes/sec. Additionally, by sending specific trigger types to specific tape drives we can do an online "split" of events for quick analysis. After events are read out into a Challenge, 8 R4400 CPU's on each Challenge fully reconstruct and process the events to remove background, filtering out 90% of the events at the level 3 trigger stage. The events that pass the level 3 trigger are built and logged onto tape. The sustained data rate out of the DAQ is about 2 kHz. Although we have chosen to use 4 planes and 6 streams (24 memory nodes) based on our throughput requirements, the numbers are flexible and extendable. We can increase our overall throughput by increasing the number of streams or planes or both.

3   Trigger timing and DA dead time

As mentioned above, the KTeV trigger system consists of three levels. The Level 1 (L1) trigger rate is expected to be around 100 kHz and is essentially deadtimeless.

The maximum L2 decision time is 2.1μsec, which means the maximum level 2 deadtime is 100kHz $\times$ 2.1μsec = 21%. The read out time for events that satisfy the L2 trigger is less than 10μsec, which adds an additional 20% dead time at a 20 kHz L2 trigger rate.

4   The DA "Building Blocks"

In this section we describe the individual DA building blocks which make up a 1 stream $\times$ 1 plane section of the KTeV DAQ.

4.1   Front End Crate

Although we have several different types of front end crates, all of the front end data in KTeV is read out through a FERA DATA bus[2]. On receipt of a L2 trigger, the data from each crate is read out and stored in the FIFO of a special readout module designed and built by Fermilab (DYC3(+)[3] or CTIRC[4]). The readout module pushes the data from its FIFO into an RS485 line, or "stream". The first crate in the stream also attaches the plane destination information to the top of its data.

4.2   Memory Node

Each memory node consists of three modules: the DM115, DC2 and DPM which is called the "DDD" [1]. The DM115 is an I/O module which receives data from the RS485 line, and pushes it into the DC2 FIFO if the event plane number matches the DM115 plane number. The DC2 is intelligent, and its main task is to transfer data from the FIFO to the DPM. The DPM is an abbreviation for a 6390 VSB/VME Dual Ported Memory made by Micro Memory, Inc. The data throughput of the DDD is 22 Mbytes/second and is limited by the VSB side of the DPM. In addition to transferring data from the DC2 FIFO to the DPM, the DC2 also has to attach the event size to the top of each subevent, make a table in the VME memory of VME addresses of all events, count the total number of events written into VME memory each spill, and finally clear the VME memory after the spill is processed. Each DDD has 224 Mbytes of memory, and the total amount of memories in the three filtering planes is about 4 Gbytes.

4.3   VME/VME interface between VME crate and Challenge

Each VME crate is connected to the internal VME bus of the Challenge by a 64 bit VME/VME interface module made by Performance Technologies, Inc. (PTI940). The maximum sustained data throughput between the VME crate and the Challenge has been measured to be 39.5 Mbytes/sec.

5   Software on Challenge

The KTeV DAQ software is constructed based on FNAL DART products [5]. The data flow in Challenge is shown in Figure 2. First, the gateway gets events from the
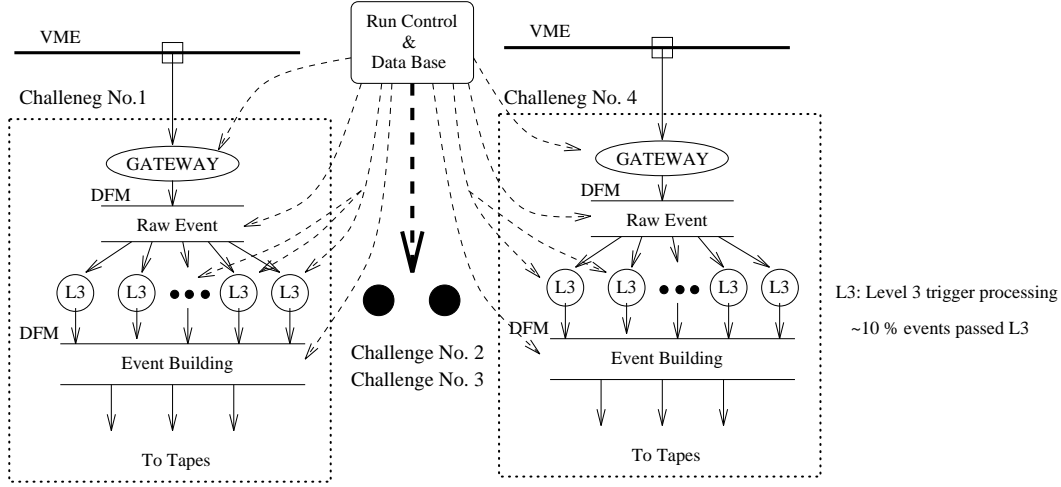
Figure 2: Data flow and DART software on the Challenges

VME memories and reads them into kernel memory in the Challenge. This kernel memory is managed by DFM, the Data Flow Maneger. The data are processed by 8 Level 3 filtering processes. The events which pass L3 filtering are built and written to tape.

In order to start and monitor all the relevent process needed for the readout on all the Challenge nodes, we use DBS (DART Bootstrap Services) which distributes system bootstrap and monitor processes on each machine. We also use the run control software DRC (DART Run Control) and OCP (Operator Control Program) and the DART configuration software DIS (DART Information Services). We use tcl [6] as a common user and program interface in all of the above software.

6 Test Reasults

We have tested the DAQ buildng block and measured its throughput capability. With one DDD, we measured 19Mbytes/sec throughput from the front end to the DDD with an event size of 1.6kbytes which is close to the KTeV event size per stream ($\sim$ 1.33 kbytes). We used FERA ADC's and DYC+, which will be used in KTeV, as the data source. For two DDD's on one stream (or 2plane $\times$ 1stream DAQ), we measured a bandwidth of 34Mbytes/sec using a round robin fashion of plane destination. For this measurement we used the KTeV Pipeline module, which will be used to read out the KTeV CsI calorimeter, in a test mode where it generated events of about 0.8 kB, which is close to the anticipated average subevent size. (Note that the bandwidth increases with increasing event size due to fixed overheads/event). Finally, we tested the plane destination logic using 3plane $\times$ 1stream DAQ. By using 3 planes, we could reach the maximum 40 Mbytes/sec bandwidth of the RS485.

In order to test the DAQ software, we used Monte Carlo (MC) events to feed the gateway instead of real events. The event format of these MC events was the same as

4

that of the raw KTeV events in VME memories, and the MC events were processed in the exact same way as the real data will be. We measured the processing time on the Challenge to be $\sim$ 3msec/event/CPU. We also tested both Exabyte 8505 and DLT 4000 as logging devices. In addition to the filtering process, we tested on-line calibration process and on-line monitoring process by using MC events.

Finally, we integrated 2 planes $\times$ 3 streams of DAQ and sent real data from the modules of sub-detector components using FERA ADC's and the KTeV Pipeline module. We measured the combined timing of reading out events from VME memories and processing them on the Challenge without filtering. The total throughput of all the overhead without filtering is over 34Mbytes/sec, well above our requirements. We also tested the robustness of our system by running it for three days at a L2 trigger rate of 5kHz.

7 Current Status and Summary

KTeV will start data taking in April, 1996. All of the parts for a 3 $\times$ 6 DAQ are installed and debugged, and the final plane of electronics will arrive in the next few weeks, allowing us to complete the final configuration by the end of the year. In addition to taking data during the experimental run, the KTeV DAQ is used for commissioning, calibrating and debugging the individual detectors that make up the KTeV experiment. We are already collecting data from the Transitional Radiation Detectors, the Photon Veto Counters, and the Drift Chambers by using a part of the KTeV DAQ. The KTeV DAQ has the capability of collecting and processing data from every component of the detector independently by partitioning the streams of a single DAQ plane.

Our test results show that the KTeV DAQ fully satisfies the KTeV input throughput requirement of 160Mbytes/sec data flow, the level 3 processing requirement of 3ms/event, and the output logging requirement of 5MB/sec. The DAQ construction is nearly complete and it should be fully installed and debugged by the end of the year, in ample time for the next FNAL fixed target run.

Acknowledgments

1. V. O'Dell and T. Yamanaka, KTeV Data Acquisition System (KTeV Internal Report), Fermilab (1994).
2. LeCroy module catalog (1995).
3. S. Hansen, S. Prodes and D. Graupman, Description of the DYC+, Fermilab Physics Department Internal Report (1995).
4. M. Qunell, KTeV CTIRC Hardware Specification, Fermilab Detector Electronics Systems Group Internal Report (1995).
5. DART documents are available at url http://fndaub.fnal.gov:8000.
6. J. Ousterhout, Tcl and the TK Toolkit, Addison Wesley Computing Series.