Fermi National Accelerator Laboratory

FERMILAB-Conf-89/136

The PAN-DA Data Acquisition System *

Don Petravick, David Berg, Eileen Berman, Mark Bernett, Penelope Constanta-Fanourakis,
Terry Dorries, Margaret Haire, Ken Kaczar, Bryan MacKinnon, Carmenita Moore,
Tom Nicinski, Gene Oleynik, Ruth Pordes, Gary Sergey, Margaret Votava, and Vicky White
Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510 U.S.A.

May 1989

* Presented at ''Real Time Computer Applications in Nuclear, Particle, and Plasma Physics,'' Williamsburg, Virginia, May 16-19, 1989.

Operated by Universities Research Association, Inc., under contract with the United States Department of Energy

THE PAN-DA DATA ACQUISITION SYSTEM (*)

Don Petravick, David Berg, Eileen Berman, Mark Bernett, Penelope Constanta-Fanourakis, Terry Dorries, Margaret Haire, Ken Kaczar, Bryan Mackinnon, Carmenita Moore, Tom Nicinski, Gene Oleynik, Ruth Pordes, Gary Sergey, Margaret Votava, Vicky White

Online and Data Acquisition Software Groups
Fermi National Accelerator Laboratory
Batavia, Illinois 60510

Abstract

The Online and Data Acquisition software groups at Fermi National Accelerator Laboratory have extended the VAXONLINE data acquisition package to include a VME based data path. The resulting environment, PAN-DA, provides a high throughput for logging, filtering, formatting and selecting events.

I. HISTORY

The VAXONLINE data acquisition system has succeeded in providing a high quality distributed online environment for VMS at high energy physics experiments at Fermilab [1]. VAXONLINE is now being extended so that the main data flow is through a low-cost, high performance VME system. Ancillary data flows for monitoring and calibration are supported to both VAXONLINE and high performance, cost effective UNIX based computer systems.

Practical limits of CPU time and Q-bus speed effectively limit the data taking speed of the VMS based VAXONLINE to about 1/2 Megabyte/second. Upcoming experiments at Fermilab are demanding data rates of a few megabytes per second.

II. RATIONALE

A new generation of front end readout controllers and event builders provide data rates in this range. These devices include the Fermilab Smart (CAMAC) Crate Controller (SCC), the Fermilab FASTBUS Smart Crate Controller (FSSC), and the Struck General Purpose Master (GPM) (Figure 1). None of these devices connect to VAX/VMS in a manner that is appropriate to an experiment's main datapath.

(*)sponsored by DOE Contract No.DE-AC02-76CH03000

Since the last fixed-target run (March, 1988) workstations have become ubiquitous at experiments. While most run VAX/VMS, they do not play the multi-purpose, multi-tasking role that is well supported by the VAXONLINE Event Pool [2]. Instead, these machines are typically dedicated to single tasks. Although not many experiments have workstations that run operating systems other that VMS, it is clear that UNIX-based computers are assuming an important role.

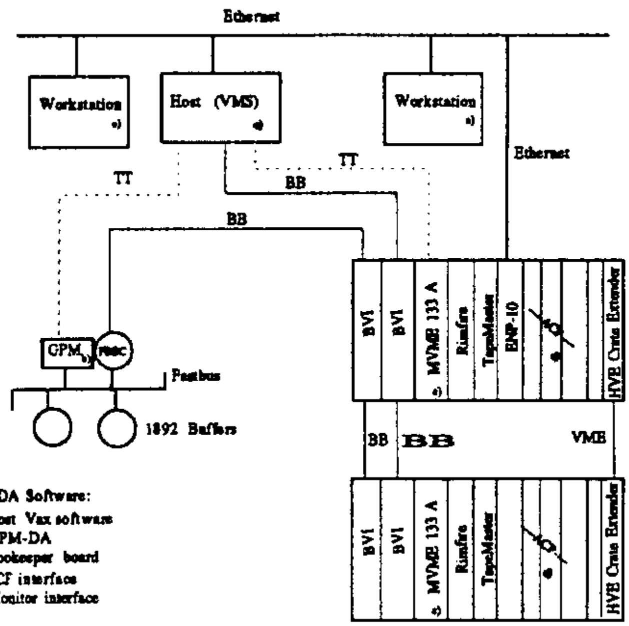

Fig. 1 PAN-DA Configuration Using GPM

Experiments' need for increased online processing power has led to the demand for computing of sufficiently low cost and high power such that it may be applied to each event in the data stream. The Fermilab ACP has a series of computers that are appropriate for incorporation into a VME based data acquisition system.

In order to move an experiment's datapath into a high throughput environment and to supply a software framework for a number of backend computers of various types, we are developing PAN-DA, a federation of several software packages and hardware modules. This paper will describe in detail many of the components and underlying software which are being used in this system.

Thus, the philosophy behind this software effort is to encourage the migration of an experiments final data path from VAX/Q-bus to VME; to accommodate workstations of many types in an experiment's analysis and monitoring framework; and to allow the inclusion of a processor farm in the data acquisition system.

III. VME HARDWARE MODULES

The PAN-DA VME system uses commercial modules for access to Ethernet, 9 track tape, and 8mm cartridge tape. Modules developed by the Fermilab Advanced Computer Program (ACP) provide access to FASTBUS and form the basis of a processor farm. RBUF modules, developed by the Fermilab Electrical Engineering Department, form the interface from CAMAC.

This activity is coordinated by a low cost computer suitable for real time applications, the Motorola MVME133A-20 32 bit Monoboard Microcomputer. This computer runs the real-time software environment, SYS68K, which we have assembled [3].

A CMC ENP-10 Ethernet node processor incorporates the LANCE Ethernet chip set and a Motorola 68010 processor. This module contains firmware which implements the TCP/IP Ethernet protocol. The ENP-10, as well as a pSOS software driver for it, is described in detail in a companion paper given at this conference [4].

Data may be logged to STC-2925 9 track tape drives, driven by the CIPRICO TAPEMASTER 3000 controller. The system supports parallel logging to multiple tape drives.

Exabyte 8mm tape drives are accessed through a VME to SCSI adapter, the CIPRICO RIMFIRE 3510. An 8mm tape cartridge may hold two gigabytes of data (the equivalent of ten conventional 9 track tapes recorded at 6250 b.p.i.) and is comparable in size to an audio cassette tape. Data may be recorded at 245,000 bytes per second. While up to seven Exabyte drives can be connected to single RIMFIRE adaptor, four drives logging in parallel will achieve the maximum SCSI banwidth, one megabyte of data each second. These devices are described in a companion paper given at this conference [5].

ACP 68020 computers are ubiquitous at Fermilab and will provide the basis for executing online filtering algorithms during the next fixed target run. The ACP nodes run ABSOFT FORTRAN under the LUNI operating system, hosted from a VMS system.

ACP R3000 computers, which will be available in the next few months, will provide significantly more computing power. As these become available, we plan to incorporate them into the PAN-DA system. Because these machines run UNIX, they integrate well into our overall design.

Access from FASTBUS is over the ACP Branch Bus. Experiment 687, Figure 1, will use a Struck GPM to push data from FASTBUS over Branch Bus into a number of ACP 68020 processors. Access to Branch Bus is provided by the BVI module, which is a Branch Bus slave and a VME master. The GPM is described in a companion paper given at this conference [6].

The primary readout mechanism for CAMAC is through the Fermilab Smart Crate Controller. The SCC interfaces to VME via the RBUF module, which may be readout under control of the MVME133 or ACP 68020 computers [7].

IV. PAN-DA SYSTEM COORDINATION

The MVME133A computer coordinates data acquisition activity in the VME crate. It supervises data flow and control information amongst the VME hardware components and ancillary computer and workstations.

Centralization of this activity gives PAN-DA relative independence from the other devices in the VME crate. The MVME133A serves as a standard communication point to the host and additional back end computers.

A. Software Environment

The software environment on this board is SYS68K, which we have assembled from a purchased real-time kernel pSOS (Software Components Group), and Microtec high level language (C and PASCAL) cross compilers.

Additional effort was required to make these commercial products suitable for our applications. First, it was necessary to port the pSOS kernel to the MVME133A. Additionally, the high level language run time library (which contains routines like C library function malloc()) had to be made re-entrant so it might be called in a multitasking environment without the danger of corrupting its underlying data structures. While Software Components Group supplies a debugger, pROBE, it was necessary to extend the debugging facilities on the board. Specifically, code has been added to check for stack overflow and misuse of the floating point co-processor.

Three tools were added which are aimed directly at understanding the behavior of the board, and the programs on it, in a running experiment. A run time message reporting facility connects with the COURIER facility in VAXONLINE [8]. A run time trace facility was added [3]. We have learned from our experience with VAXONLINE that it may take days to locate a flaw in software. Obtaining a complete snapshot preserves the maximum amount of information, while permitting experiments to continue acquiring data. A Static Debugging tool has been developed, which allows the entire context of a board to be transferred to and saved on a VAX, where it can be examined at leisure using the Microtec XRAY debugger.

The SYS68K environment supports additional boards as well. Our implementation of this environment is fully described in a companion paper given at this conference [9].

B. Data Flow Tasks

Because the MVME133A runs a multi tasking kernel, each data flow is directed by a single task.

1. A tape logging process locates buffers of events to be logged and drives either the TAPEMASTER controller if 9 track tape is used or the RIMFIRE adaptor if 8mm tape is used.

2. A GPM pathfinder task locates buffers to be filled with data originating in FASTBUS. This task builds a list which is then used by the GPM.

3. The Host Data Hoist locates events wanted for analysis by monitoring computers and workstations.

4. A Data Acceptor injects event data that originate outside of the main data path into the logging stream. A typical application would be to place end of spill events and calibration items onto tape.

5. A Run Control task receives control information from the host machine and serves to coordinate the data acquisition within the VME crate.

V. CONNECTIVITY

In order to keep SYS68K environment simple, it was designed to be subordinate to a host machine. The primary connection to the host computer for the MVME133A is Ethernet. Any task running may establish a connection across the Ethernet network. A pSOS ENP-10 driver and socket level interprocess communication routines have been implemented.

This allows the MVME133A to be connected to VAX/VMS (where we support the SRI MultiNet TCP/IP system) and to UNIX.

Since this board is controlled from a network, and PAN-DA seeks to integrate a number of diverse workstations and computers, a Remote Procedure Execution (RPX) package with underlying support for TCP/IP has been implemented for pSOS, UNIX and VMS.

The RPX package addresses many of the problems associated with a distributed system composed of different types of computers. Its effect, from a programmer's point of view, is to make the system appear as if implemented on a single computer. Each component of the system merely maintains a procedure call interface.

It is advantageous for us to use RPX wherever possible. An interface implemented with RPX is automatically available on all of the target computers. The RPX package itself deals with differences in binary representation and hides the actual communications protocol used. RPX forms a framework for migrating applications with machine dependencies to new computers. The RPX software package is fully described in a companion paper given at this conference [10].

The dataflow tasks are controlled via a remote procedure execution interface, enabling the host computer to run either VMS or UNIX. An example control program for VMS will be supplied, but the individual task's interface could be used if distributed control is desired. Since a substantial amount of communications to the coordinating MVME133A are implemented via RPX, we anticipate that its functions can be distributed amongst several processors, if a single board proves inadequate, or to a new processor, such as the ACP R3000.

We have successfully migrated the analysis portion of the CDF offline package to a MIPS R120 computer and left the operating system dependent portion of the code (the VMS database) running under VMS. Should the database be ported to RISC/OS (a UNIXesque operating system), it would be trivial to remove the remote procedure calls; this application would be then become local.

The RPX package allows communication over serial lines as well as Ethernet.

VI. SUMMARY

The full PAN-DA system will be used at E687 during the next fixed target run at Fermilab. Since PAN-DA is a modular system, several experiments are already using some of its software components.

For many applications we support at Fermilab, PAN-DA represents a significant extension of established high speed front end readout systems and the VAXONLINE monitoring environment. The system makes use of new economical data storage devices, high speed microprocessor technology (RISC), and operating systems other than VMS.

VII. REFERENCES

[1] V.White, P.Heinicke, E.Berman, P.Constanta-Fanourakis, B.MacKinnon, C.Moore, T.Nicinski, D.Petravick, R.Pordes, L.Quigg, "The VAXONLINE Software System at Fermilab" IEEE Transactions on Nuclear Science, Vol.NS-34, No.4, August 1987.

[2] D.Quarrie, "Data Acquistion System - Programmers reference manual", CDF Note 166 (Fermilab Computing Department Note PN-251)

[3] B.MacKinnon, D.Berg, "SYS68K - The compound product for pSOS/68K development", Fermilab Computing Department Note PN-387.

[4] M.Bernett, G.Oleynik, "Uniform Communications Software Using TCP/IP"

[5] S.Bracker, P.Constanta-Fanourakis, G.Hockney, K.Kaczar, G.Oleynik, D.Petravick, M.Votava, V.White, "Exabyte Helical Scan Devices at Fermilab"

[6] T.Dorries, M.Haire, C.Moore, R.Pordes, M.Votava, "FEREAD *Front End Readout Software for the* Fermilab PAN-DA Data Acquisition System"

[7] M.Bernett, P.Heinicke, B.Mackinnon, S.Wickert, "SCC-OS Smart Crate Controller Operating System Users Reference Guide", Fermilab Computing Department Note PN-314.

[8] B.MacKinnon, T.Nicinski, "VAXONLINE Message COURIER System, Reference Manual and User's Guide", Fermilab Computing Department Note PN-311.

[9] D.Berg, P.Heinicke, B.MacKinnon, G.Oleynik, T.Nicinski, "A Real Time Integrated Environment for Motorola 680xx-based VME and FASTBUS Modules"

[10] E.Berman, D.Petravick, G.Sergey, " "Remote Procedure Execution Software for Distributed Systems"