CONF 890415--2

BNL--42688

DE89 012673

MPS Data Acquisition Software System+

A.C. Saulys, A. Etkin, K.J. Foley, R.W. Hackenburg, R.S. Longacre, W.A. Love, T.W. Morris, E.D. Platner
Brookhaven National Laboratory, Upton, New York 11973
S.J. Lindenbaum
Brookhaven National Laboratory and City College of New York
C.S. Chan, M.A. Kramer
City College of New York, New York, New York 10031

To be published in
Proc. of Computing in High Energy Physics Conference
Oxford, U.K.
10-14 April 1989

April 4, 1989

+ Work supported by the U.S. Department of Energy under Contract No. DE-AC02-76CH00016.

MPS Data Acquisition Software System+

A.C. Saulys, A. Etkin, K.J. Foley, R.W. Hackenburg, R.S. Longacre, W.A. Love,
T.W. Morris, E.D. Platner
Brookhaven National Laboratory, Upton, New York 11973
S.J. Lindenbaum
Brookhaven National Laboratory and City College of New York
C.S. Chan, M.A. Kramer
City College of New York, New York, New York 10031

ABSTRACT

A description of the software for a FASTBUS based data acquisition system in use at the Brookhaven National Laboratory Multiparticle Spectrometer is presented. Data reading and formatting is done by the SLAC Scanner Processors (SSP's) resident in the FASTBUS system. A multiprocess software system on VAX computers is used to communicate with the SSP's, record the data, and monitor on-line the progress of high energy and heavy ion experiments. The structure and the performance of this system are discussed.

1. Introduction

This paper describes a data acquisition system in use at the Brookhaven National Laboratory Multiparticle Spectrometer (MPS). It is a FASTBUS based system which reads out CAMAC based devices as well by the use of a FASTBUS Branch Driver.* The host computer is a VAX-750 and it is interfaced to the FASTBUS by means of a computer interface module** (CFI) to VAX Unibus.

+ This research was supported by the U.S. Department of Energy under Contract No. DE-AC02-76CH00016

* STRUCK 320 module

** STRUCK 300 module

2. Hardware Layout

Even though this paper is meant to describe the software aspects of the data acquisition system, a minimal description of the hardware is required. A block diagram of this system is shown in Fig. 1.

The VAX-750 host computer is connected to the Computer Interface module (CFI) in FASTBUS crate 0 by means of a specially adapted DRE11-C UNIBUS module. The data recording medium is a 6250 bpi magnetic tape capable of speeds of 125 inches per second.

The present FASTBUS configuration consists of three FASTBUS crates. Each crate contains a SLAC Scanner Processor (SSP). The SSP's are used as intelligent masters in FASTBUS. They are programmable modules with their own CPU's and data memories [1]. The cable segment port on the SSP's allows the SSP to be used as a Buffered-Segment Interconnect (BSI). This feature gives us the ability to assemble a multicrate FASTBUS data acquisition system and provides us with essentially direct access to all the crates, even though there is only one CFI in one FASTBUS crate (crate 0). This crate also contains the FASTBUS Branch Driver (FBD) for accessing the CAMAC branch through FASTBUS and a Lecroy 1892, a 4 megabyte FASTBUS memory module. Since we operate with a fixed target accelerator, the memory module allows us to buffer data during beam spills and read it into the VAX host between spills for recording on magnetic tape.

Detector systems routinely connected to the CAMAC and FASTBUS front end modules include: proportional wire chambers, scinitillator hodoscopes, drift chambers, time projection chambers, lead glass arrays, various ADC's and latches.

3. SSP Software

The SSP program DAQ is written in VAX-11 MACRO assembly language format. This may seem a surprising approach at first, since the SSP CPU has been designed to emulate IBM system/370 instructions. In fact, since we use a VAX computer as a host for the data acquisition system, we find it convenient to assemble the SSP software on a VAX. Since the MACRO assembler language has provision for externally defined "macros", a special macro library has been written for the SSP that allows SSP MACRO code to have a similar appearance to IBM assembler language. This approach has served us well, since when we started putting together our data acquisition system we had no access to an IBM system/370. Besides the set of IBM and Fastbus I/O instructions contained in the library, there are also some higher level macro definitions which facilitate such things as data definition, subroutine linkage and repetitive loops. The VAX-11 MACRO assembler and the SSP macro library taken together form the BNL Cross Assembler for the SSP. This cross assembler produces SSP machine code that is already in separated form (SSP program memory and data memory are two different entities). There are important distinctions between the BNL Cross Assembler and the IBM assembler, which will not be discussed here. Manuals detailing the writing of code are available at BNL.

The SSP program consists of a main program and a number of subroutines in a set of MACRO files. These subroutines with a memory definition file are assembled into an object library for the VAX linker. After assembling, linking, and loading, instruction statements produce code that is loaded into the SSP's program memory (PM), while data statements produce code that is loaded into SSP's data memory (DM). This is accomplished through the use of program sections or PSECT's, which permit the assembler to produce separated code. The PSECT's are collected into defined clusters at linking time by the VAX linker. This facilitates the downloading of the SSP's program memory and data memory separately.

For various reasons, including overall simplicity, all SSP's contain identical code (the same DAQ program). The actual differences in the execution of the individual SSP's, according to the needs of each crate's devices, is implemented through various device and descriptor lists, resident in the SSP's data memories. These lists can be downloaded from the VAX host at the beginning of any run. This allows us to modify the FASTBUS configuration by simply editing a list. We can add or remove crates and modules as required to accomodate data from different sets of apparatus. The data acquisition system is thus more flexible and easier to change than if the SSP code itself had to be reprogramed. Each SSP has a DM word, SSPID, with which the SSP can know its own identity. This allows the distinction to be made between the allowed set of operations for the SSP in the crate with the CFI (SSP0) and the SSP's connected by a cable segment.

In addition to the DAQ program a set of SSP-resident routines are loaded which permit user written software on the VAX host to access FASTBUS slaves in other FASTBUS segments, be they cable or crate, through a chain of SSP's in much the same way as if there were real FASTBUS segment interconnects (SI's) joining the segments. This system is called the Buffered Segment Interconnect (BSI) sytem. At present, there is no user-definable routing table or adjustable timeout capability, but a fairly flexible default routing table has been defined which permits access to any slave on a segment which is separated from the CFI module's segment by as many as two intervening segments (for a total of four segments being involved). The default routing table is very simple; the 32 bits of the primary address word are broken up into four bytes. Only the lowest-order byte is a true geographical FASTBUS address, the other three comprising the "segment identifier". With true SI's the user assigns a unique segment identifier to each route that he intends to use, and downloads the SI's with that information [2]. With the BSI routines, however, the 3 bytes of the segment identifier are used as geographical addresses. The originating master (be it host-computer with FASTBUS interface, or an SSP) looks at the highest order byte (of these three), and if it is non-zero, uses that byte as the geographical address of an SSP through which the transaction will be "routed". The originating master then strips off this highest non-zero byte and passes the new primary address on to the SSP indicated by that byte. That SSP in turn does the same, until there are no non-zero bytes in the three-byte segment identifier. The

SSP at the end of the line directly engages the slave. The way the BSI routines are set up in the host permits the programmer to call the same routines for slaves in the segment in which the host has direct access as for slaves in other segments.

4. Host Software

The VAX-host data acquisition software system consists of three separate processes:

1) DAFAB
2) DAFABCON
3) DAFNET

This software system is designed to run as a multiprocess system with minimal operator intervention and terminal usage. These processes are constructed from several software libraries: CFI access code, SSP access code, tape manipulation library, and command parsing library. The CFI access code is the old CERN FASTBUS library release version of August 1985. The FUFISUBS library was modified at BNL to improve recovery from CFI hardware hangs. The SSP access code was all written at BNL. The tape manipulation library is a combination of a set of VAX FORTRAN routines using VAX/VMS system services and a DECUS library of special utilities. The command parsing library COMAC has been written at BNL for general on-line systems use and has been described in a previous publication [3].

4.1 DAFAB

The main data acquisition process DAFAB runs as a spawned process with no terminal attached. It only needs an error output device. The function of DAFAB is to detect when the SSP has finished filling the FASTBUS memory, to read out the memory, and to write it on magnetic tape. At the beginning of each run DAFAB transmits all the parameters describing the run conditions to the SSP and signals the SSP to start the data taking run. It also writes a copy of the device list that the

SSP uses, any other parameters specified by the operator, and any operator comments to the magnetic tape at the beginning of the run. DAFAB normally cycles in a loop testing a data ready flag in the FASTBUS memory every 100 milliseconds and for any commands passed on by the controlling process DAFABCON. These commands are passed by means of Common Event Flags with access provided by the VAX/VMS System Services. When the SSP sets the data ready flag in FASTBUS memory at the the end of its spill, DAFAB reads a memory directory set up by the SSP. This directory contains the pointers to the beginning of each SSP buffer. It then reads the memory in blocks of 120 kilobytes or less, the block size determined by the maximum integral number of SSP buffers that can be fit into 120 kilobytes. Since the CFI has a long fixed access time (about 30 ms.), this blocking optimizes the data transfer rate to the VAX-host. The data is then reblocked for tape writing with tape block size of up to 32 kbytes for IBM tape transport compatability. The tapes are written in image binary format and in a multibuffered "QIO" mode to maximize the effective tape writing speed. After each FASTBUS memory read DAFAB tests for an Event Flag requesting a sample of data for monitoring purposes. If the flag is set, then it makes a copy of the current data buffer in a Global Section which is shared with the DAFNET process.

4.2 DAFABCON

The DAFABCON process is the controlling process of the data acquisition system. It interacts with the operator in determining the operation mode of the data acquisition. It loads the intelligence and the list of operations in FASTBUS (the device list) into the SSP's. It communicates with the operator using the COMAC dialogue [3]. Using COMAC, a dialogue can be set up in a command file for any data acquisition conditions. Using this command file an operator needs very little knowledge of the data acquisition system operation to start up the system. All he has to do is to execute the command file from a terminal.

4.3 DAFNET

DAFNET is a small process activated when a user on any DECNET node calls the SYS$ASSIGN system service routine to execute a command file resident on the data acquisition host computer. The DECNET link is set up in the transparent task-to-task mode. Once the DAFNET process is activated it waits with the SYS$QIOW service routine for a message from the process requesting the data buffer. It then sets a Common Event Flag requesting a fill of its buffer by the DAFAB process. Upon acknowledgement of completion from DAFAB it writes the data buffer to the requesting process using the SYS$QIOW system service routine. It then waits for the request of the next data buffer.

5. System Operation

During the accelerator beam spill the SSP's go through their device readout cycle in the following way:

1) When an event of interest is detected by the fast electronics trigger logic, a trigger pulse is sent to the main SSP (SSP0).
2) SSP0 disables any further triggers and reads a module in a CAMAC crate to obtain a trigger classification.
3) SSP0 passes trigger class information to the other SSP's and issues commands to start up the other SSP's on the cable segment.
4) Each SSP reads out the modules specified in its own device list as downloaded in the beginning of the run. Each trigger class has its own, separate device readout list, and each SSP has its own set of these lists. The data from each module is organized by the SSP's into a buffer, complete with headers containing word counts and status words for each module.
5) When all SSP's on the cable segment have completed their device readouts, SSP0 reads their formatted data and appends this data to its own buffer, making a single buffer containing the data from all devices read out by all

the SSP's. This with an additional header of run, date, time, tape number and total word count information forms an event buffer. Several events can be (and usually are) formed into a bigger buffer for efficiency in data block transfers. These multi-event buffers contain a header with run information as well as a pointer list to the beginning of each event buffer. Each multi-event SSP buffer is written to the FASTBUS memory.

6) At the end of each event SSPO re-enables triggers from the fast trigger logic and goes back to step 1.

7) When SSPO receives information that the end of the beam spill has occured or the FASTBUS memory is full, it sets a flag in the FASTBUS memory indicating that the memory is ready to be read out by the VAX host. It then halts, waiting for the VAX to read out the memory and restart SSPO.

6. System Performance

This data acquisition system has been in use at the BNL MPS since the beginning of 1988 for some heavy ion data-taking using a 3-module Time Projection Chamber (TPC) system [4]. It will be used for data recording of other experiments planned for the MPS apparatus this year. Similar implementations of this system are in use by other experimenters at BNL.

The system has been timed to record approximately 300 kilobytes per second with an accelerator cycle time of 3 seconds and a beam spill time of 1 second. Since the system does not read the FASTBUS memory during the beam spill, this transfer rate corresponds to effective recording speed of 450 kilobytes per second. This data recording speed is consistent with the CFI data transfer speed. We find this data acquisition system quite flexible and easy to use for the experiments we have run and plan to run with the MPS in the near future.

FIGURE CAPTIONS

Figure 1. MPS data acquisition system hardware block diagram.

REFERENCES

[1] H. Brafman, T. Glanzman, A. J. Lankford, J. Olsen, and L. Paffrath, IEEE Transactions on Nuclear Science, Vol. NS-32, No. 1, (Feb. 1985) pp. 336.

[2] U.S. NIM Committee, FASTBUS: Modular High Speed Data Acquisition and Control System for High Energy Physics and Other Applications, (Dec. 1983) DOE/ER-0189.

[3] A. C. Saulys, The BNL Multiparticle Spectrometer Software, Computer Physics Communications 38, pp. 319 (1985).

[4] William A. Love, Some TPC Measurements in an Oxygen Beam at the AGS. Paper presented at the VII International Conference on Ultra Relativistic Nucleus-Nucleus Collisions (Quark Matter 88), Lenox, Massachusetts, September 26-30, 1988 (to be published in Nuclear Physics A, Special Issue); A. Etkin et al., Modular TPC's for Relativistic Heavy Ion Experiments. Paper presented at the Wire Chamber Conference, Vienna, Austria, February 13-17, 1989 (to be published in Nuclear Instruments and Methods in Physics Research).

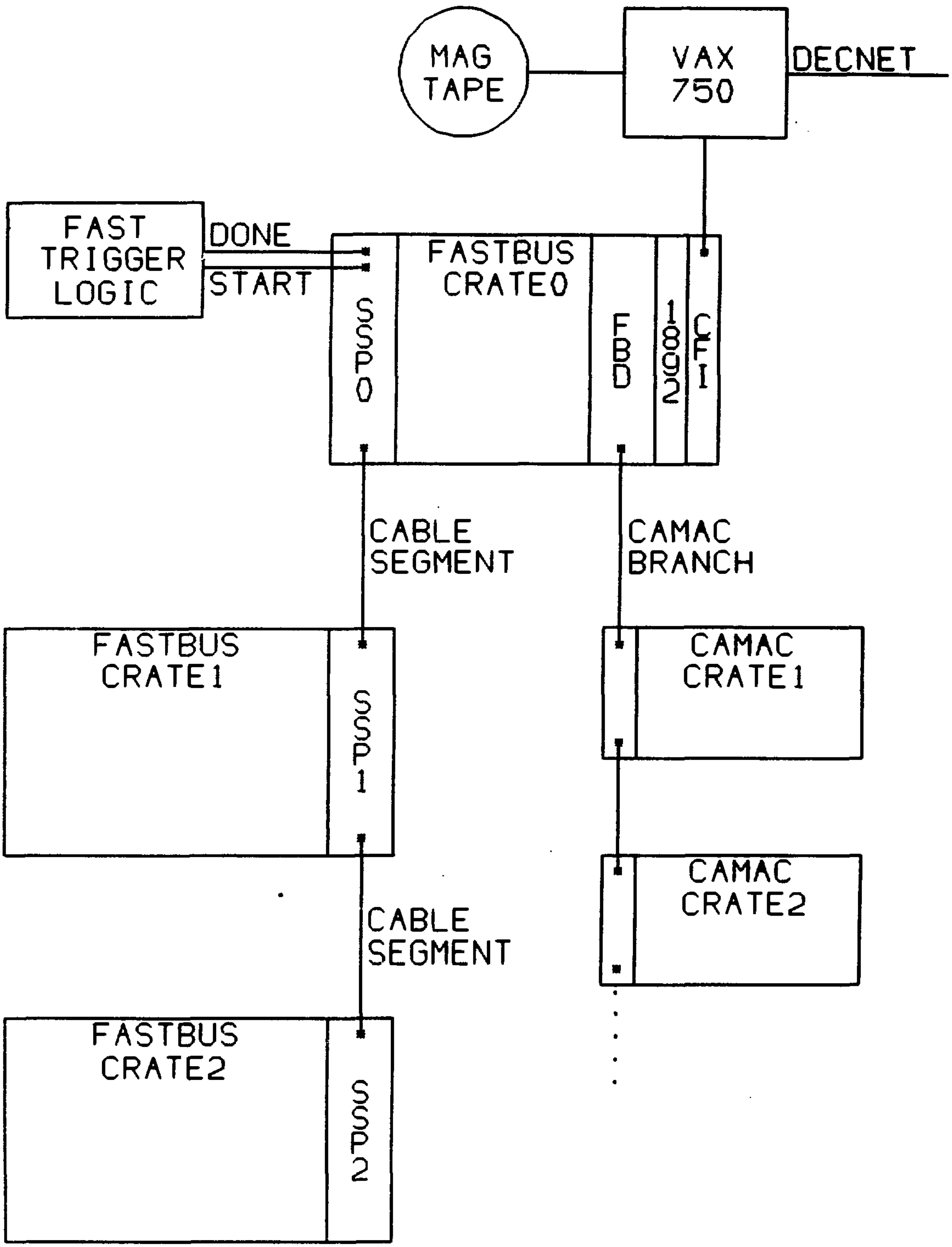
MAG TAPE
VAX 750
DECNET
FAST TRIGGER LOGIC
DONE
START
FASTBUS CRATE0
SSP0
FBD
1892
CFI
CABLE SEGMENT
CAMAC BRANCH
FASTBUS CRATE1
SSP1
CAMAC CRATE1
CAMAC CRATE2
CABLE SEGMENT
FASTBUS CRATE2
SSP2

Figure 1