CAMAC EXPERIMENTAL BEAM LINE CONTROL SYSTEM

L. J. Hepinstall, E. J. Barsotti, H. C. Lau
National Accelerator Laboratory*
Batavia, Illinois

Introduction

The Laboratory has well over two miles of beam lines in the Meson, Neutrino, and Proton experimental areas. The size and complexity of these beam lines as well as the requirements imposed by multiple experimenters led to the development of a computer-controlled multi-console control system. To alleviate many of the problems associated with interfacing various experimenters' equipment to this control system, a well-defined hardware interface was necessary. A CAMAC modular instrumentation system[1] was employed in all three beam lines primarily because it is an international standard. Each of the beam lines is computer controlled using a number of CAMAC crates located in control areas scattered along each beam line. These control areas are tied together and to the computer by a fast serial communications link. Control consoles consisting of CRTs or teletypes are interfaced to CAMAC crates and may be located in any control area. The system software handles up to 12 of these remote consoles, 2 local consoles, and a variety of other tasks simultaneously through a multi-programming technique. Man-machine communication is by means of custom CRT displays and a message-oriented command language.

System Organization

Figure 1 is a block diagram of the typical serial system hardware. Each beam line has its own independent system although the three are essentially identical.

Areas along each beam line that require control equipment are called control areas. The serial communication system ties these areas together in a branched network configuration rather than as a unidirectional loop. This allows installation of additional control areas or their maintenance without necessarily shutting down the entire network. Experimenters' control stations are generally placed at the end of a branch where installation work will not disturb the rest of the control system. Repeaters in each control area provide branching capability to three additional control areas. Crates within an area are daisy-chained together.

System Hardware

Each beam line has a mini-computer equipped with 48K bytes of core memory, a 1M byte disk storage unit, a teletype, and paper tape gear. The computer's I/O buss is brought out to a modified CAMAC crate (PDC crate) to provide a convenient method of interfacing to the computer. This permits special controllers to be external to the computer which simplifies maintenance and, in the event of computer failure, simplifies switch-over to a spare computer.

A serial link controller, in the PDC crate, transmits a 70-bit serial-command data word and receives a 35-bit status-data word over the primary serial link. It operates at a 1 MHz bit rate and provides status and data parity checking. A watchdog timer terminates any transmission that fails to be acknowledged within a specified time. Block transfers over a secondary serial link provide fast data acquisition of large blocks of data from a crate. These transfers are initiated by a special command to a crate controller and operate independently of the primary serial system thereafter. Returned data is handled by a block transfer controller and passed directly to computer memory via an automatic data channel. Like the serial link controller, the block transfer controller operates at a 1 MHz bit rate and provides status and data parity checking.

The serial system is interfaced to a CAMAC dataway by either a serial branch driver in conjunction with a type A controller[2] or a serial crate controller.[3] Each of these controllers accepts the 70-bit word, checks parity, determines whether it is being addressed, performs a CAMAC dataway cycle, and returns a 35-bit word. The serial crate controller has the additional ability to generate block transfers.

A total of six coaxial cables connect each of the control areas. Two are used for the primary serial data, one for block transfer data, one for the accelerator clock,[4] and two for TV signals.

CAMAC Interface

The individual CAMAC crates contain a variety of Laboratory-designed and commercially available modules that perform the basic control functions of the beam lines. These include power supply and collimator controllers of various types, timing pulse generators, scalers, and various I/O modules. Areas requiring analog monitoring contain a multiplexed analog to digital converter that accommodates 64 differential inputs. This converter is interfaced to the control system via one of the CAMAC input modules.

A console controller module interfaces a CRT or teletype console to the CAMAC dataway. This Laboratory-designed module sends and receives characters in a full duplex mode over an RS-232-C[5] interface at a switch selectable rate of 4800 baud (CRT) or 300 baud (TTY). The controller accepts 16 bits of data (two

*Operated by Universities Research Assoc., Inc. under contract with the U.S. Atomic Energy Commission.

ASCII characters) from the CAMAC dataway and dumps them into a 64-character first-in-first-out buffer. The output of this buffer feeds the CRT or teletype at the console's maximum data rate. This buffering allows the computer to operate in a burst output mode to these relatively slow devices, thus greatly reducing the servicing requirements and the associated overhead. The module receives input characters from the console, combines them with the three other console inputs--a key switch and two knobs--and sends the result to the computer when solicited (not buffered). The software scans all console controller modules for input at a 10 Hz rate. A key switch on the controller module restricts, via software, the commands accepted from the attached console. Two 8-bit up-down counters, which are cleared when read, are each connected to a 60-count-per revolution shaft encoder. The shaft encoders are mounted near the console's keyboard and are used as knobs to adjust beam line parameters. A switch-selectable hardware multiplication factor of 1, 2, or 4 is provided to vary the coarseness of the knobs.

A buffered memory module was designed to provide a communications interface to an experimenter's computer. The module consists of two 128-byte first-in-first-out buffers (one input, one output) plus a 16-bit readable and writeable status register. When in use the experimenter's computer becomes an extension of the control system. It can be used to collect beam data or as a conversational console.

Experimental Control Hardware

When an experimenter arrives on site, he is provided with the equipment necessary to tie into and use the control system. This consists of a control rack with a CAMAC crate (including repeaters and controllers), power supply, CRT console, two knobs, and two TV monitors. CAMAC modules for receiving beam parameters, generating timing pulses, and possibly interfacing to the experimenter's computer, are installed in his crate. Miscellaneous CAMAC modules may also be provided on request. This equipment is tied into the beam line control system at the nearest control area with six coaxial cables.

Operating System

The need to support up to 15 consoles of various types while performing other real-time tasks led to the development of a multi-programming operating system. This system provides a re-entrant environment in which a variable number of tasks can run concurrently. Task control, context switching, resource management, dynamic memory allocation, I/O requests, events, timers, and statistics are all handled by the operating system.

Tasks form a single linked-list circular queue and are serviced on a first-come-first-served basis. Each task has three possible states: active (executing), ready (for execution), and waiting (for a timer or system event). Scheduling is controlled by timer and system events as well as by the active task itself. A task is allowed to run until it can no longer use the CPU (such as when it is waiting for a disk access) or when a WAIT is coded into the task. This means that some caution must be used to prevent a compute-bound task from degrading the system response. On the other hand, this scheduling algorithm does not require variables used with a single task execution period to be protected. This greatly reduces re-entrancy overhead and thus improves system response.

The operating system supports re-entrancy by providing 14 machine and pseudo-registers for each task. A single task may have several levels of register storage associated with it, organized in a push-down stack. When tasks are switched the current task's pseudo-registers are saved and linked to its last register block. The scheduler then examines the task queue for the next ready task and loads the pseudo-registers from that task's most recent register block. Control is then passed to the task and execution continues from where the task left off.

External interrupts are treated as system events which cause tasks waiting for that event to advance from the waiting to the ready state. I/O and other internal interrupts are handled by specialized service routines.

The operating system and all tasks are core resident. This is natural since the majority of tasks are active continuously and those which start and stop occasionally are so small in size that making them disk-resident is not advantageous.

Basic System Tasks

The system has five basic tasks: the master teletype console task, the paper-tape console task, the remote console scan task, the device scan task, and the beam data task. Console tasks for the master teletype and the paper tape operate in a conversational mode under control of the command processor (which will be described later).

Every five seconds the remote console scan task examines each CAMAC crate specified in a search table for the presence of a console controller module. If a controller is found and there is no console task associated with it, one is initiated. Information returned by the console controller indicates the type of console attached so that, if the console is a teletype, the console task disallows special CRT functions (such as "Page mode", described later). When a console controller or its crate is disabled or removed, the remote console scan task terminates the console task associated with the controller. This automatic startup and shut-down procedure permits consoles to be placed wherever there is a scanned CAMAC crate (almost all control areas). It also allows the system to adjust dynamically to changes in the number, type, and location of remote consoles.

The device scan task runs once very 20 seconds. It monitors all specified devices for changes in status or analog readings which exceed tolerable limits. If an abnormal condition is detected, a message describing the condition is sent over the TV system to all consoles.

The beam data task is initiated by an interrupt that signals extracted beam. This task services up to ten memory interface modules, distributes various beam parameters to specific modules along the beam line, and generates up to sixteen beam profile displays.

Command Processor

Figure 2 is a block diagram of the software system showing the interrelation of the various modules. The nucleus of the system is the command processor which directs virtually every control function. This re-entrant task provides the conversational interface for all consoles. It is the job of the command processor to respond to all commands entered at various consoles and to manage the custom displays.

The command language provides a means of controlling and monitoring all beam line devices in a general way. Differences in devices are sorted out by the device handler and are transparent to the operator. Devices are referred to by a one- to six- letter descriptive name and may be defined and named from any console. Initially a standard set of device definitions is entered via paper tape and only a few devices usually need to be entered by an operator. When a device is defined, its name is entered into a core-resident directory and its definition is written in the device descriptor file on the disk. Access to device descriptors is handled by the device access control routine. This routine maintains a pool of devices in core that are currently being used. If more than one task is using a device, only one copy of it is in core but that copy has an appropriate access count. When a device definition is no longer in use, it is returned to the device file and its space in the pool is made available to other devices.

Any device may be monitored at any console. However, it may be necessary to restrict which consoles can control a device. To achieve this, control assignment is provided for each device to define the set of logical consoles that may control it. A logical-physical console relationship permits control of blocks of devices to be shifted from one physical console to another by merely re-defining the logical-physical console relationship. The ability to change the device assignment or the logical-physical console relationship is restricted to unlocked consoles. This refers to the key switch on the console controller for each of the remote consoles. The local teletype and paper tape consoles are always unlocked.

The command processor can operate each CRT console in a "Page Mode". In this mode the screen is divided into a custom display area and a command entry area. Each console has a set of 20 pages that can be called up from disk storage and each page can display up to 15 beam line devices in its custom display area. The analog reading and digital status of each device on each displayed page is monitored once a second. When either changes, the value on the screen is updated. An operator can tailor his pages to his needs dynamically by specifying in command format what page is to be called and what devices are to appear on that page. The pages are permanently stored on the disk so that once created they may be called at any time. Either of a console's two knobs may be assigned to any device on a page by a simple command. Both knobs can be simultaneously active at each console. Devices that appear on a page are tagged as to their controlability (assigned to that console) and knob assignment.

Space limitations on the CRT screen permit only a fraction of the digital status for a device to be displayed. To solve this problem, a second form of the page mode displays a single device's status. This allows a modest amount of text to describe each status bit. Devices displayed in this page format use one of 255 text layouts that are stored on the disk.

The command language provides a means of creating, editing, and executing files that contain commands ("User files"). Each console has a set of ten command files that contain fifteen commands each. An operator may execute the commands within a file by pressing one of ten function keys on his console. This facility may be used to reset a group of scalers, to turn on a group of power supplies, or for whatever an operator wishes.

Various commands control the other system tasks such as the device scan and profile display. In addition, there are commands to display elaborate system documentation files ("Help files") at any console. These files contain complete operating instructions and recent notes on any system changes. A basic debug facility exists with a write lockout feature to prevent accidental sabotage. A serial link test program is provided to directly address a piece of CAMAC hardware. The branch, crate, slot, subaddress, and function code may be explicitly specified for a single transmission and the results may be displayed in their most basic form.

Acknowledgments

The authors wish to express their grateful appreciation for the valiant efforts of Mr. R. Dice, Mr. S. LaMantia, Mr. J. Snow, and Mr. R. Wischermann in the design, construction, and testing of the beam line control systems. We also wish to thank Mr. J. Michelsen for his valuable suggestions and his assistance in preparing this paper.

References

1. CAMAC. A Modular Instrumentation System for Data Handling. AEC Report TID-25875, July, 1972.

2. CAMAC. Organization of a Multi-Crate System. AEC Report TID-25876, March, 1972.

3. A CAMAC Crate Controller for a Serial Control System, E. J. Barsotti, this conference.

4. PRIMET: A Versatile Accelerator Timing System, C. A. Swoboda, R. E. Daniels, H. C. Lau, A. Tummillo, this conference.

5. Electronic Industries Association Standard, RS-232-C, August, 1969.

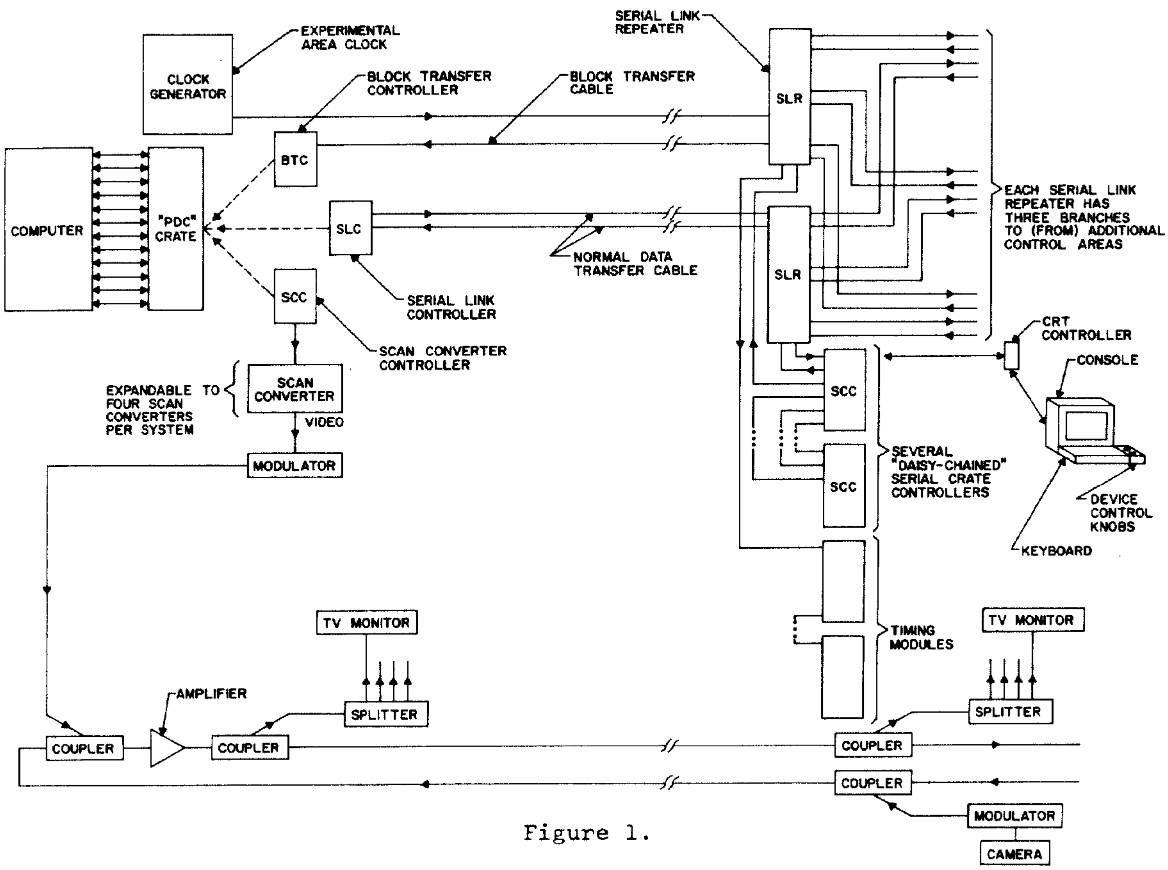

Figure 1.

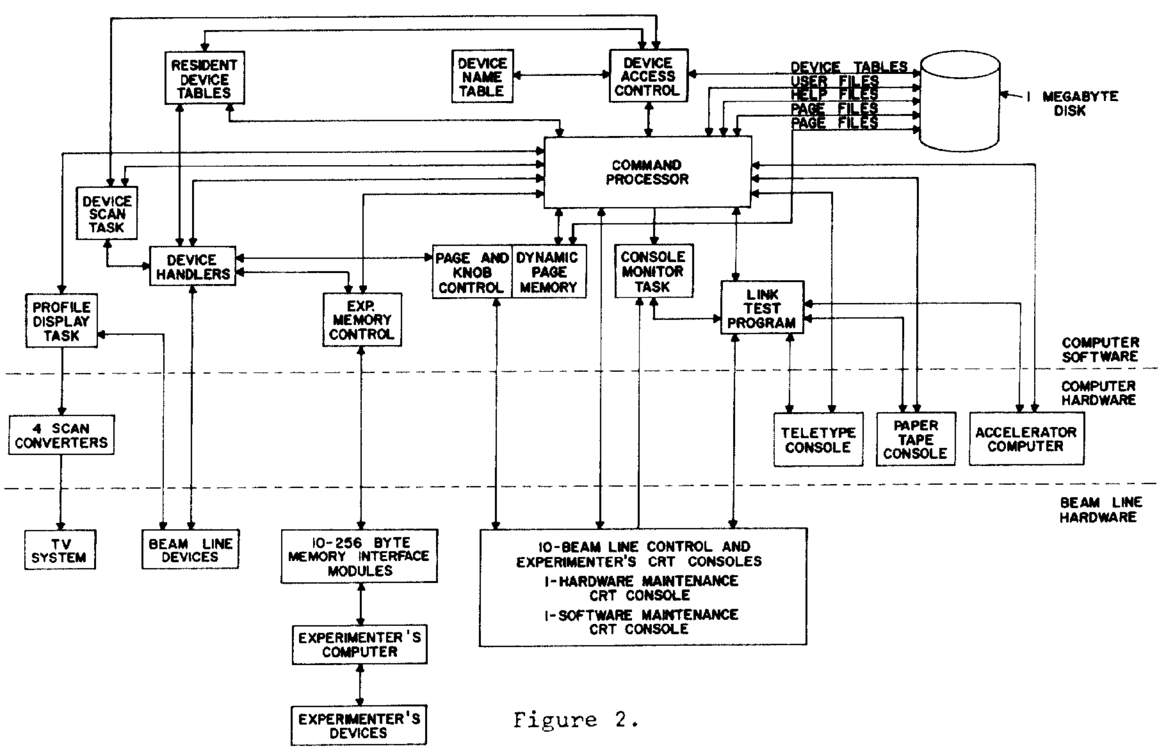

Figure 2.