A MICROPROCESSOR-BASED PREACCELERATOR CONTROL SYSTEM

R. W. Goodwin, R. F. Kocanda and M. F. Shea
Fermi National Accelerator Laboratory*
Batavia, IL 60510

Summary

A modular microprocessor system is used to control the Fermilab high voltage test facility. Operationally, the system has been patterned after typical Fermilab accelerator control systems. The operator can select pages of analog parameters and digital status to be displayed on an alphanumeric screen; analog readings or settings may be displayed either in engineering units or in volts; devices are selected for on/off or analog control by cursor position; an incremental shaft-encoder knob adjusts the value of analog parameters and digital time delays. Settings of parameters can also be entered from a 16-key keyboard and transmitted to the device. Two microprocessor systems are used--one in the high voltage equipment dome and one at ground potential. Communication between the processors is via two serial, digital, fiber optic light links.

Introduction

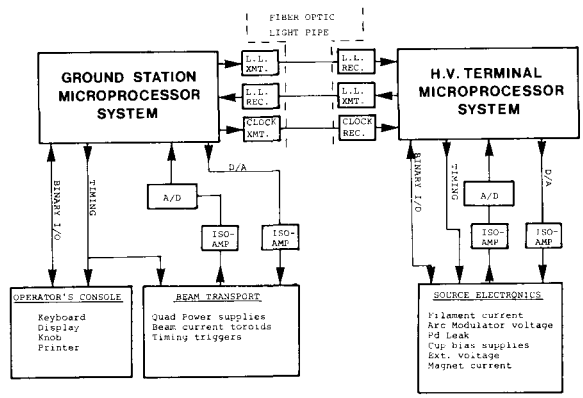
The Fermilab high voltage test facility is in the original Linac laboratory building located about 3 miles from the accelerator. This facility is maintained to support the ion source and column development program. Because of its location, this system could not be conveniently integrated into the Linac control system, so other alternatives were considered to fulfill the control requirements including a scaled down Xerox Sigma II system, minicomputers and microprocessor systems. Any solution short of installing a nearly duplicate Linac control system required a new software package. An analysis of the system parameters indicated that a microprocessor could provide all the necessary control, monitoring and display functions. We chose to implement the control of this facility with a microprocessor-based system.

The Hardware

General Description

The control system for the preaccelerator includes functions to be performed both at the ground-based beam transport line and operator's console, and in the high-voltage terminal where the ion-source electronics are located. At each location all the standard control system functions are needed: analog input and output, digital sensing and control, and generation of variable delay timing pulses.

The system is implemented as shown in Fig. 1 with two integrated microprocessor-I/O systems connected together by three fiber optic light links - a timing link, and a data link from the ground station to the high voltage dome, and a return data link from the dome to the ground station. The dome system controls and monitors the source electronics; the ground station controls and monitors the beam transport line and the operator's console. Each station includes a 12-bit, 16-channel A/D converter, several 12-bit D/A converters, digital delay timers and binary I/O cards.



Block diagram of the microprocessor-based control system

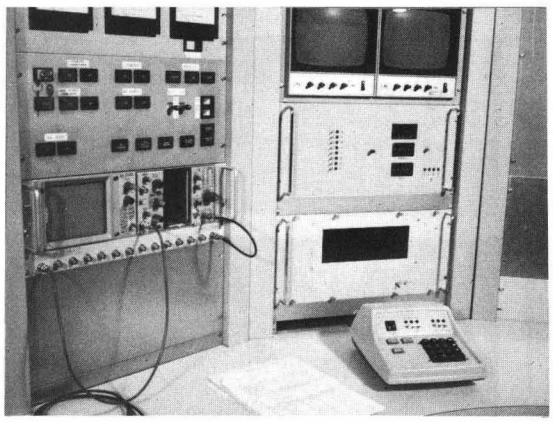
Fig. 1

The preaccelerator control racks are shown in Fig. 2. The system is operated through the small box on the console that contains several dedicated buttons and switches, a 16-key keyboard for data entry and a shaft encoder knob for adjusting analog and timing parameters. Figure 3 is a photograph of this control box. Data are displayed for the operator on an 8-line by 32 character Burroughs' Self Scan panel.

The Microprocessor System

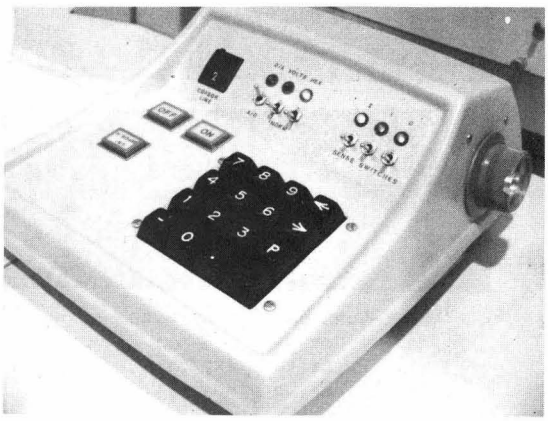
The microprocessor system itself has been described elsewhere,[1] so only a short description is included here for completeness. This system built around the M6800 CPU is designed as a bus-oriented card file that accepts up to sixteen 4 1/2 in. x 6 1/2 in. printed circuit cards on one-half inch centers. An operational system contains a CPU card, a controller, enough memory cards to contain the program for the application, and an assortment of both analog and digital I/O cards to interface with the external equipment. These cards are accessed by the CPU as though they are memory locations so that a single instruction transfers data to and from registers on the I/O card. The cards are designed to uniformly interface with the printed card-file backplane that carries the address, data and control signals between individual cards and the CPU. Signals to external equipment are accessed by ribbon cable connectors on the front of the cards. The card-file and its associated power supplies are mounted in a chassis that has interface connectors on the back panel and a front panel that allows inspection of memory and loading memory locations manually. The two microprocessor chassis are identical, and each is connected to an "expansion chassis" that provides space for future expansion of the system. Most of the interface cards needed for the preaccelerator system are part of a family of compatible cards designed to work with the

*Operated by Universities Research Association Inc. under contract with the Energy Research and Development Administration.



Photograph of the preaccelerator control console

Fig. 2

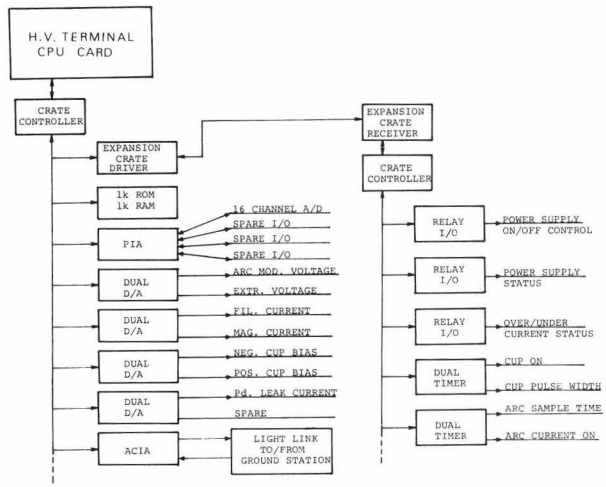


Photograph of the operator's control box

Fig. 3

microprocessor system. Special purpose cards can be fabricated using kluge cards that contain the interface to the backplane bus. The relay I/O cards were implemented in this way.

Figure 4 is a block diagram of the high voltage terminal system. This figure shows the interface cards and the external equipment controlled by them. The interface cards include dual 12-bit D/A, asynchronous communication interface adapter, relay isolated binary I/O, dual 16-bit preset timers and TTL compatible digital I/O. The TTL compatible I/O is from a Peripheral Interface Adapter (PIA), a single 40-pin integrated circuit that has two 8-bit bytes of programmable I/O along with 4 control bits. The direction (input or output) of the data bits and the behavior of the control bits is determined by the program. The versatility of this device eliminates much of the custom design previously associated with interfacing a computer to a variety of external equipment. As an example, a single binary interface card containing four PIA chips positions the display cursor, writes data to the display, controls and reads the 16-channel A/D converter, reads the console sense switches, the 16-key keyboard, the vertical cursor position switch, and the 8-bit up-down counter associated with the shaft encoder knob.



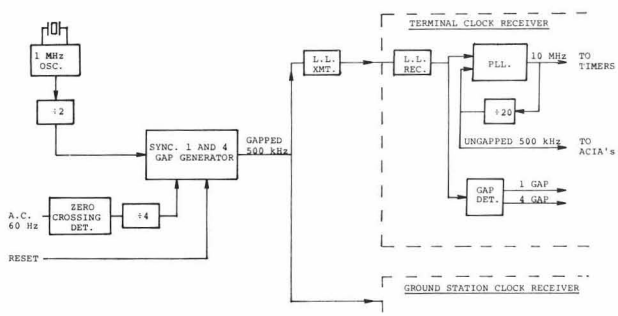
High voltage terminal system block diagram

Fig. 4

Data Transmission and Timing

Data are transmitted serially between microprocessors through fiber optic light links. An asynchronous communication interface adapter (ACIA) card provides the interface to TTL compatible light link transmitters and receivers. The ACIA performs the parallel-to-serial conversion and generates the parity bit for each data byte transmitted. It also does the serial-to parallel conversion and the data overrun and framing error checks for each data byte received.

Figure 5 shows the multipurpose timing system. A one megahertz crystal clock is divided by two and driven out to two clock receiver modules, one at the ground station and one in the high voltage terminal. This frequency is multiplied by a factor of 20 using a phase locked loop with a divide-by-20 scaler inside the loop. The resulting 10 MHz frequency is the time base for the delay timers. Two other pieces of information are inserted onto the 500 kHz pulse train by synchronously removing either one pulse or four pulses. A single pulse is deleted from the pulse train to mark the beginning of a beam cycle. It is used to reset the delay timers. This "one gap" is inserted by every fourth pulse from a line zero-crossing detector to produce a



Timing system diagram

Fig. 5

cycle rate of 15 Hz. Four pulses are deleted from the pulse train to initiate a "Reset" for the microprocessor in the dome. The missing pulses are detected by the clock receiver modules. The gapped 500 kHz pulse train is the input for the phase locked loop but an ungapped 500 kHz is also available at the output of the divide-by-20 scaler. This ungapped 500 kHz is used for the external clock to operate the ACIA circuits in a synchronous mode. The synchronous mode increases the baud rate from 50 kHz to 500 kHz. The 500 kHz frequency was chosen to satisfy the frequency limits of the ACIA and light links without requiring too large a multiplying factor in the phase-locked loop. The clock receivers at both stations are identical. All the analog and digital data are transmitted to the ground station each cycle. The transmission time for 56 bytes is 5.3 ms including the ground station software overhead needed for error checking and storing the data.

Timers

Variable delayed trigger pulses are generated by dual-channel timer cards. Each channel is a 16-bit scaler that counts either the 1 MHz CPU clock or an external clock. The count is compared with a 16-bit register loaded by the CPU and a pulse is generated when the counter equals the value in the register. The cycle is initiated by an external pulse, a pulse from the CPU or the output of another timer. Using a 10 MHz time base a total delay of 6.5 ms with 100 ns resolution is possible. The phase-locked-loop frequency multiplying technique results in very small jitter between the two timing system.

Operating Characteristics

General Features

The organization and operator interface to this system has been patterned after typical Fermilab control systems.[2] The operator can display an index that lists the displays that are available. There are three types of displays: parameter pages, a binary status display and a memory dump. Examples of these are shown in Fig. 6.

The Parameter Display

The majority of the capability of the system is contained within the parameter display. From the present total of 52 parameters, the operator may group any eight together within one of the pages P1 through P6. Each line of the display contains a description of the device, a six character number field and a three character units field. A two digit number entered at the beginning of the line, followed by pressing "Keyboard Interrupt", places the device on the present line. Settings of controllable devices may be changed either by entering the desired setting in the number field and pressing Keyboard Interrupt or by turning the shaft encoder knob with the cursor positioned on that line. For some of the devices in the dome, pressing ON or OFF will initiate that action for the device on the cursor line. The value displayed in the number field is usually the A/D reading averaged over 8 pulses, scaled to display in engineering units and updated every 0.8 sec. Toggle switches on the console can

(a)

```
0 INDEX          P8
P1.SOURCE #1     P9 MEMORY DUMP
P2.SOURCE #2
P3.QUADS
P4.TOROIDS
P5.PARAM
P6.PARAM
P7 STATUS
```

(b)

```
-QTM1 ENDS              120.9 A
-QTM1 CENTER            246.7 A
-ARC SAMPLE TIME         31.4  US
-POSITIVE CUP BIAS      107.2*V
-NEGATIVE CUP BIAS     -102.1*V
-CUP ON TIME             28.4  US
-CUP OFF TIME            34.7  US
-TDATA SAMPLE TIME       29.6  US
```

(c)

```
EXTRACTOR OV O
*ARC MOD
.MAGNET
*FILAMENT  UV OC
*CUP BIAS
 PRESSURE  UP OP
 COOLING   NF OT
```

(d)

```
FD20 U  0FD7  14CE  FFFF  FF8E
FD28    00FF  9100  FF91  02FF
FD30    9200  FF92  02CE  00FF
FD38    FF8E  06CE  55FF  FF95
```

Self-Scan panel displays
a. Index
b. Parameter display
c. Binary status
d. Memory Dump

Fig. 6

cause the number field to display either A/D or where appropriate the D/A values. These can be given in engineering units, volts, or hexadecimal. Controllable devices are identified by a dash at the start of the line. A star following the number field indicates that the device is off. Once the operator arranges a page of parameters, he can leave the page and return to it at a later time.

Status Display and Memory Dump

The Status Display provides a means of examining all the digital status of devices in the high voltage terminal. ON/OFF commands can be sent to equipment in the dome from this page. Such a page is needed to show status of equipment that has no analog readout associated with it, or equipment that has several bits of binary status.

The memory dump page is a program that displays 32 bytes of memory starting at any location selected by the operator. Two bytes of data may be entered manually on the display and loaded into memory by pressing the Keyboard interrupt. A console sense switch directs the program to display data from the "Upstairs" system (U) or the "Downstairs" system (D). Other sense switches allow changing the display to show the next 32 bytes, the previous 32 bytes, or to display in A/D volts rather than hexadecimal. The display is updated as

required at 15 Hz. This page is a diagnostic aid to the engineer as well as the programmer because I/O is addressed as memory.

The Software

Program Preparation

The software for the entire system was written in the assembly language of the M6800 Microprocessor, assembled by a cross-assembler running on the Linac 16-bit control computer and programmed into 2708 UV-eraseable PROMs using a similar 6800 system with a PROM programming card. Because the read-only memories are organized as separate 1K byte chips, it is natural to organize the software in modules which fit in the separate chips. To facilitate communication between modules, the entry points to subroutines within each module are implemented as a series of jump instructions placed at the beginning of the module. Thus changes can be made by assembling only those program modules affected.

Ground Station Software

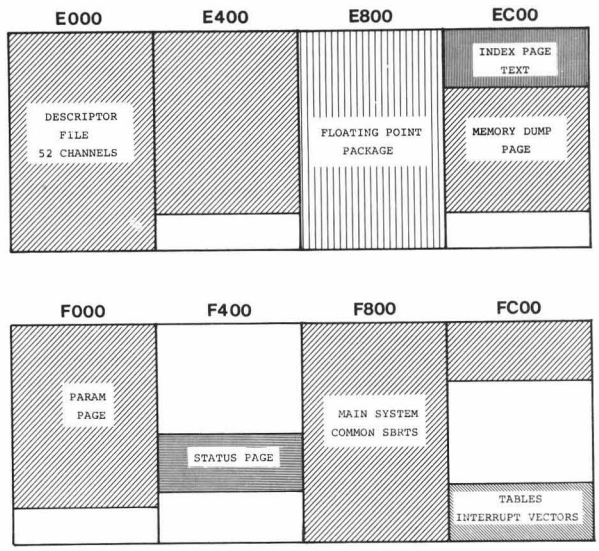
The ground station software is organized into modules as shown in Fig. 7. The total program storage is 8K bytes, programmed into 8 read-only memory chips. The modules include tables, display programs, and common subroutines.

The descriptor file contains 36 bytes of information for each data channel. This includes text to describe the device, engineering units designation, full-scale in terms of those units, D/A control address and on-off control address.

The heartbeat of this system is the interrupt signal that occurs at a frequency of 15 Hz, the pulse repetition rate of the preaccelerator. Upon receipt of this interrupt the ground station sends a request for data to the dome system. While the dome system is reading its data, the ground system reads its own A/D data. If a dome station raise/lower request is included in the 26-word data pool received, the appropriate setting is sent to the dome station. (This method of allowing local control in the dome was chosen so that the ground station always knows the current settings of all D/A's and timers). If the ground station front panel interrupt latch is set, one data word is sent to the address specified by the front panel switches (either in the ground station or in the dome system as specified by a toggle switch). All 15 Hz interrupt activity is completed in about 8 ms.

The background loop notices that a 15 Hz interrupt has taken place and processes 16-key keyboard activity to allow numeric entry on the Self Scan display, left-right cursor control or request of a new display page. Next, lever-wheel activity is checked to allow cursor line positioning. The knob processor watches for changes in the knob up-down counter and stores differences to be acted upon by the current display program. Then the display program is called at the appropriate entry point as described below. Upon completion of display program activity, the front panel is updated and the background loop waits for another interrupt.

Each display program provides four entry points via a jump table. These are used for the initialization call when the page is first called up, the termination call when a new page is requested, the keyboard interrupt entry when the keyboard interrupt button is pressed, and a 15 Hz entry called if none of the other calls is pending. A page table includes for each page a pointer to the display program entry point jump table and a pointer to private RAM storage that may be used, for example, to remember which devices are displayed on each line of the display.



Ground station program memory map

Fig. 7

The floating point package provides all the subroutines necessary for doing basic floating point calculations. A floating point accumulator is maintained to hold one of the parameters and the result. A four byte format is used for floating point numbers, where the first three bytes are a tow's-complement signed mantissa and the last byte is a signed binary exponent. This corresponds to more than 6 decimal digits of precision and a range of 10^{38}. Subroutines are provided to load or store the floating accumulator, add, subtract, multiply, divide, convert between integer and floating point, and convert between 6-character ascii format and floating point. Typical times in ms are .4 for add and subtract, 1.5 for multiply and divide, 4 for encoding, and 10 for decoding ascii format. The entire package resides in 1K bytes of PROM and uses 12 bytes of RAM for the floating accumulator and temporary working storage.

Dome Station Software

The dome system software is much simpler than the ground station since there is no operator's console or display provided other than the front panel hexadecimal 4-digit display, address and data hex switches, and toggle switches and binary leds. Two raise/lower pushbutton switches allow for local adjustment of D/A's and timers in the dome. The data switches are set to indicate a channel number to be adjusted and an increment to apply to the

current setting. If either the raise or lower button is depressed when the request for data comes from the ground station, then the channel number and requested increment or decrement is included in the data pool returned. Another front panel switch allows displaying memory data in decimal volts assuming the data is in A/D format (left-adjusted signed binary fraction of 10 volts).

The dome system uses no interrupts but merely spends most of its time updating its front panel and waiting for a command record to appear on the light link.

The format of data records transmitted between the two stations consists of 8-bit bytes transmitted via ACIA's as described above. The first byte in a record is a header byte describing the type of data to follow. The next byte gives the number of 16-bit data words in the record. The data words are next, followed by a checksum byte and final zero byte. When the dome station receives a header byte it first checks a link command table to see if it is a valid header byte, to get the address of the memory buffer into which the data is to be read, and to get the address of the command routine to be called upon successful transfer of the data. The number of words is checked to be sure it doesn't exceed the size of the buffer. Then the data is read into the buffer and the checksum is verified. Finally the command routine is called. Types of command facilities provided are 1) read all dome data and return data pool; 2) make one-byte or two-byte settings; 3) select one address for passing dome data word to ground front panel display; and, 4) read and return selected block of memory.

Power-On and Restart

Many features have been included in the software to minimize the chances of inadvertently misadjusting source parameters. When the microprocessor is first powered up, it sends settings of zero to all the D/A converters in the system. At this time the arrangement of devices on the parameter pages is restored to groupings that are stored in the read-only memory. If either microprocessor is reset, the last recorded D/A values are transmitted to the D/A's. The ground station differentiates between these two cases by examining a word in a memory location. If the appropriate value is found, microprocessor assumes it was a normal reset, not a power-on reset. In that case the stored values are considered valid and transmitted to the D/A's. When the dome station is reset it directs the ground station to restore the D/A settings in the dome using the raise/lower facility described above with a zero increment.

Results and Conclusions

Work began on the system described above during December of 1975 and the first use of the system with beam occurred during March of 1976. All the functions described here were available at that time though the external equipment was not configured to accept the remote ON-OFF capability or the delay timers in the terminal. The system performed as designed and provided the operator with the controls and displays needed to operate the test facility.

The major problem with the system has been caused by the unusually severe electrical noise environment around the test facility at the times of preaccelerator arc-downs. Although this sparking has caused little damage to the electronics, it does normally require a restart of the system following a major arc-down. Two PIA's that drive the operator's console failed before we improved the shielding of the interconnecting cables. Work is in progress to improve the shielding and isolation of signals connected to the microprocessor chassis. A similar system was tested at the operating preaccelerator where the shielding is much better. This prototype system did not require a reset following an arc-down.

The use of a modular microprocessor-I/O system greatly simplified the hardware construction and checkout, allowed for many checks on the validity of communications between the two systems, and provided flexibility for future expansion. We plan to add a 30-character per second printer to the system to produce hard-copy listings of operating parameters. A separate microprocessor-controlled emittance measuring system is being assembled for the negative ion source development program. With the addition of a link to the Linac control computer, this entire system can act as a satellite computer while maintaining the local control capability described above.

Acknowledgments

The authors would like to thank Phil Livdahl, Don Young, Curt Owen and Cy Curtis for their encouragement and support of this work. We would also like to acknowledge the contributions of Ellery Cook and Jawahar Ticku who helped with the fabrication and installation of the system.

References

1. R. W. Goodwin, R. F. Kocanda and M. F. Shea, "A Method for Implementing Microprocessor Controlled Systems", IEEE Trans, Nucl. Sci. NS-23, No. 1, 297-300(1976).

2. R. E. Daniels, R. W. Goodwin, and M. R. Storm, "The NAL Computer Control System", IEEE Trans. Nucl. Sci. NS-20, No. 3. 505-509(1973)

DISCUSSION

R.L. Witkover, BNL: Can you give an estimate of the cost of this system?

Shea: Commercially available interface cards similar to those used here cost about $200 each, so the dome system, including the chassis and power supplies, would be about $4000.

F. Selph, LBL: How difficult is it to make changes in the program? How many people understand the programming language?

Goodwin: Only one man was needed to write the program. Minor changes such as adding devices or changing full scales involve modifying the descriptor file table which merely requires reprogramming one or two memory chips, a procedure which anyone can do with minimal instruction. Of course, more extensive changes require more detailed understanding of the program itself. The programming language used is the standard M6800 assembly language.