BEAM POSITION MONITOR READOUT AND CONTROL IN THE SLC LINAC*

J. BOGART, N. PHINNEY, M. ROSS, D. YAFFE

Stanford Linear Accelerator Center
Stanford University, Stanford, California, 94305

ABSTRACT

A beam position monitoring system has been implemented in the first third of the SLC linac which provides a complete scan of the trajectory on a single beam pulse. The data is collected from the local micro-computers and viewed with an updating display at a console or passed on to application programs. The system must operate with interlaced beams so the scans are also interlaced, providing each user with the ability to select the beam, the update rate, and the attenuation level in the digitizing hardware.[1] In addition each user calibrates the hardware for his beam. A description of the system architecture will be presented.

1. INTRODUCTION

Beams of different currents and phasing with respect to the 360 hertz pulses of the machine travel down the SLC linac. The SLC control system architecture provides for several independent operator consoles. Different operators will, in general, want to collect beam position data for different interlaced beams. It is also often the case that a single operator will want to collect data for two or more beams, switching quickly from one to another. The SLC Beam Position Monitor (BPM) hardware and software have been designed to meet these requirements.

2. USING THE SLC BEAM POSITION MONITORS

The SLC beam position monitor hardware consists of two components: the stripline monitors in the tunnel, in which the charge to be measured is induced by the passing beam, and BPM processors (BPMP's) which process and digitize the analog stripline data. Each BPMP has three analog-to-digital converters (ADC's) and an attenuator. During normal data-taking, the *sum* ADC returns a value proportional to the strength of the beam. The other (*difference*) ADC's measure the horizontal and vertical displacement of the beam. In the linac each processor is connected to a single stripline monitor. In other parts of the SLC up to 10 stripline monitors may be multiplexed and connected to a single BPM processor. Except for the longer time intervals necessary to read the multiplexed BPM's, the difference between multiplexed and non-multiplexed BPM's is not apparent to the operator.

An operator wishing to take BPM data first sets values for several parameters. These include, among others, the geographical area over which data is to be taken, the beam code (known as the *PP*) for which data is to be taken, and the attenuation value to be written to the BPM processors. All these parameters are collected together in a logical entity known as a *Measurement Definition*. Each operator may have up to five Measurement Definitions available at one time.

The next step is to calibrate the BPMP's for the chosen beam. Calibration of the BPM processors serves at least three functions: it provides some checkout of the hardware, gives the operator an opportunity to find the correct attenuation setting for the measurements he wishes to make, and determines the characteristics of the processors (namely pedestal and gain for each ADC) while viewing the chosen beam, so that the raw measurement data may be properly interpreted. Pedestals and gains are calculated when the operator requests calibration for a particular Measurement Definition; the values remain accessible as long as the Measurement Definition exists. Disregarding for the moment a scale factor (to compensate for the differences among attenuators) and offsets, we can say

$$x_\mu = \frac{x_{\rm meas} - x_{\rm ped}}{s_{\rm meas} - s_{\rm ped}} \times \frac{s_{\rm gain} - s_{\rm ped}}{x_{\rm gain} - x_{\rm ped}} \times C$$

where the constant C depends only on the distance between strips in the stripline monitor. (Here $s_{\rm meas}$, etc., refer to the sum ADC and $x_{\rm meas}$, etc., refer to the appropriate difference ADC.) A similar formula holds for y_μ.

Now the operator is ready to take measurements. He may elect to view the data in one of two available updating displays: in numerical form or as three bar graphs (x-displacement, y-displacement, and transmittance, as in Fig. 1). Alternatively, he may invoke higher-level applications software to receive the BPM data.

3. SLC CONTROL SYSTEM OVERVIEW

In order to understand the protocol used in acquiring BPM data, it is necessary to have some familiarity with the architecture of the SLC control system.[2,3] In the SLC control system computing resources include a VAX 780 and approximately 50 micro-computers. The VAX runs a program for each operator (SLC Control Program, or *SCP*). The micros communicate directly with the hardware to be controlled, each micro being responsible for the hardware in its geographical area. The code running in each micro is divided into several "jobs" to handle different functions. In particular, a micro with beam position monitors to control will have, among others, a BPM job and a Timing[4] job. The BPM job handles operator requests for BPM functions. In doing so, it must request services relating to synchronization from the Timing job.

One micro-computer, the *Master Pattern Generator* (*MPG*), has no immediate hardware control responsibilities. Its primary function is to broadcast the upcoming beam code in order to initiate activity among klystrons and any other hardware (e.g., kicker magnets) which is controlled on a pulse-to-pulse basis. Additionally, any SCP may request the MPG to broadcast a second byte (known as the *YY*) to accompany a particular PP. It is this latter facility which is used to synchronize BPM readings.

* Work supported by the Department of Energy, contract DE-AC03-76SF00515.

0018-9499/85/1000-2101$01.00© 1985 IEEE

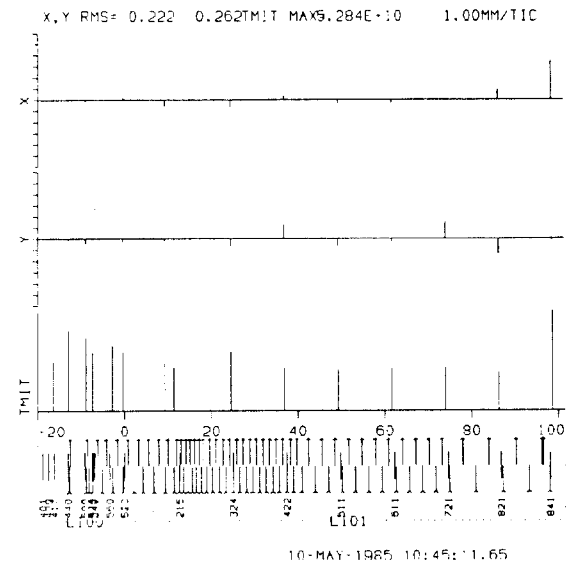

Fig. 1. Beam position monitor video display: x-position, y-position and transmittance versus z-position along the linac. Lower part of display shows positions of magnets.

4. FLOW OF DATA AND CONTROL IN BPM MEASUREMENTS

Figure 2 shows the (logical, not physical) lines of communication among computers when an operator requests BPM readings. Figure 3 shows in more detail the sequence of events in a single micro.

When an operator has a calibrated Measurement Definition and initiates data-taking, his SCP first sends a request to the BPM job in each micro in the area of interest to prepare to take data. Included in the request are, among other things, a code identifying the Measurement Definition (the BPM job uses it to find the right calibration values for this Definition) and a pair of YY's. The BPM jobs respond by preparing two camac packages, to be executed synchronously during measurement, and requesting the Timing job to associate them with the two YY's. Next the SCP sends a request to the MPG to broadcast the two YY's. The first must be broadcast at the same time as the PP of the Measurement Definition, since it is on the immediately succeeding pulse that the BPM processors will take data. The second YY need only be broadcast a short time later. The SCP now has nothing to do but wait for the data.

When it is able, the MPG broadcasts the first YY. It is seen by the Timing job in each micro. In those micros where BPM preparation has taken place, the Timing job will now execute the first camac package. It consists of packets for the Programmable Delay Units[5,6] (*PDU*'s) in the sector, and packets to set the attenuation in the BPMP's. Just before beam time the programmed PDU's gate the BPMP's. At beam time analog signals from four strips reach each BPMP and are digitized to produce counts for x-displacement, y-displacement and for the total sum signal. Soon thereafter the MPG broadcasts the

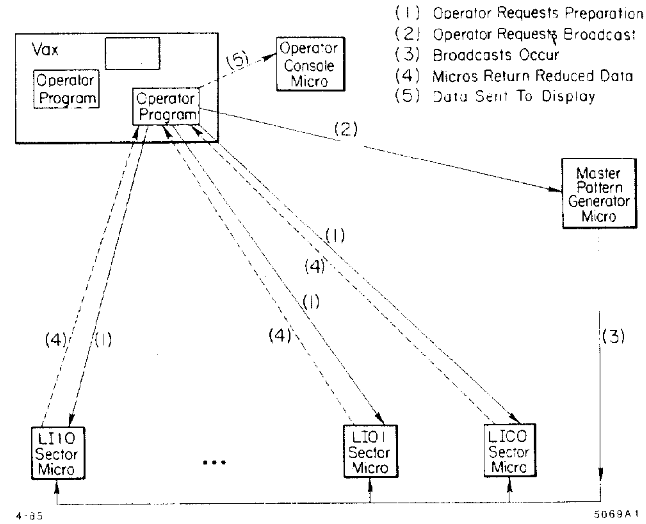

Fig. 2. Flow of data and control in beam position measurement. Sector micros prepare for measurement asynchronously. The measurement itself is synchronized by means of broadcasts by the dedicated Master Pattern Generator micro.

second YY. When the Timing job sees it the second package is executed, causing the digitized data to be read into the micro's memory. The Timing job then sends a message to the BPM job, letting it know that new data is available. The BPM job wakes up in response to the message. It reduces the data using pedestal and gain values peculiar to the beam and the BPMP, determined during a previous calibration. The BPM job then sends the processed data (x- and y-position in microns, transmittance, and status information) back to the waiting SCP. The SCP may send the data to the operator console for display or may use it in further calculations. Finally, when data-taking is over, the SCP sends a request to the BPM job to delete the camac packages and their association with the YY's.

5. REMARKS ON IMPLEMENTATION

Typically (e.g., if the data's ultimate destination is an updating display) the micros will be asked to prepare for BPM readings, data will be read for a large number of pulses, and finally the micros will be asked to clean up. It is not necessary to re-prepare the camac packages before each reading. In this way the amount of work the micro must do on a pulse-to-pulse basis is minimized.

A micro must save a substantial context (timing information, attenutation settin calibration information, etc.) for each Measurement Definition having that micro in its range; because the micro's memory is limited this context must be stored in compact form. It must also be readily accessible so as not to upset the delicate timing involved in processing data for different interlaced beams. These requirements could not be met by allocating static storage for all possible active Measurement Definitions, so space is allocated dynamically as needed.

In the areas of the SLC with multiplexed BPM's (currently the electron damping ring and its transport lines) a somewhat different protocol is needed. It is impossible to read all the BPM's on a single pulse: each multiplexer may take up to ten inputs, so any SCP wishing to read data from multiplexed BPM's will request the MPG to broadcast ten identical pairs of

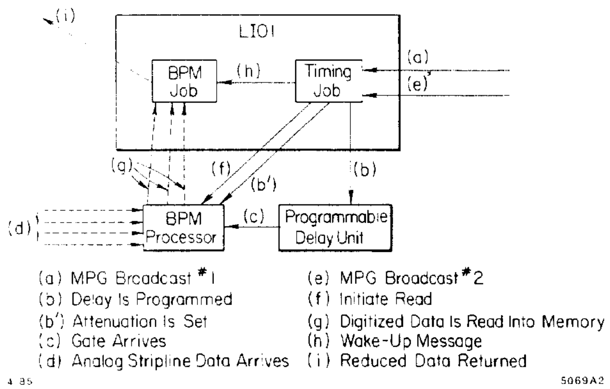

Fig. 3. Flow of data and control in beam position measurement for a single micro and its hardware. On the pulse immediately following the first Master Pattern Generator broadcast, within the gate from the Programmable Delay Unit, the signal from the striplines is digitized in the Beam Position Monitor Processor. On the pulse immediately following the second broadcast the digitized data is read into the micro's memory.

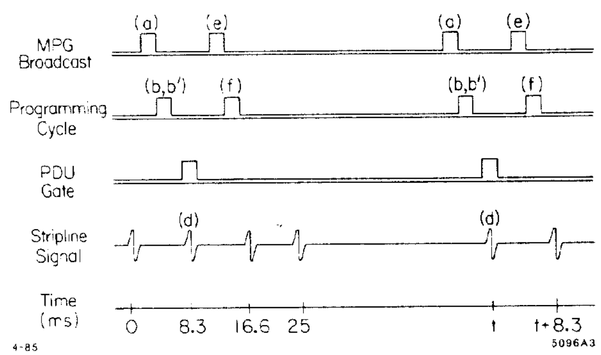

Fig. 4. Timing diagram for synchronized events surrounding beam position measurement in a micro (depicted in Fig. 3).

YY's. The first camac package must be enlarged to include a command to the multiplexers to select the proper setting. After each pair of YY's has been broadcast and before the next YY is received, several of the camac packets must be modified. In particular, the data for the multiplexer packet must be changed so that a new input will be read on the next YY, the timing data sent to the PDU must be changed (to avoid interference at the multiplexers, the peak in each input arrives at a slightly different time), and the BPM packets for reading data must be changed so that the new data does not over-write the old. The BPM job wakes up after each pair of YY's to make these changes, but only after the final pair does it transform the raw data into engineering units and send it back to the requesting SCP.

The protocol used to obtain the calibration numbers is similar to that used for measurement: the SCP asks each sector micro to prepare camac packets, then requests special broadcasts from the MPG to synchronize readings, etc. However the BPM processors must be put into special modes for measuring pedestals and gains; and for increased accuracy ten readings of each type are done and the results averaged. Hence it is not possible for a micro to interleave calibration and regular measurement readings. Under normal circumstances calibration is a rare event, so there is little harm in locking out other BPM operations in the sector micro while a calibration is in progress.

6. PERFORMANCE

To date, with non-null pulse rates up to 180 hertz, the BPM hardware and software have met all requirements: several operators have been able to take data successfully on interlaced beams over the same, disjoint, or overlapping areas of the SLC without conflict or interruption.

REFERENCES

1. J.-C. Denard et al., IEEE Trans. Nucl. Sci., NS-30, 2364 (Aug. 1983).
2. N. Phinney, K. Crook, M. Crowley-Milling, Report on the SLC Control System, Proceedings of this conference.
3. R. Melen, Design and Performance of the Stanford Linear Collider Control System, IEEE Trans. Nucl. Sci. NS-32, 230 (Feb. 1985).
4. K. Thompson, Timing System Control Software in the SLC, Proceedings of this conference.
5. L. Paffrath et al., A New Timing System for the Stanford Linear Collider, IEEE Trans. Nucl. Sci. NS-32, 84 (Feb. 1985).
6. J.E. Linstadt, A Programmable Delay Unit Incorporating a Semi-custom Integrated Circuit, Proceedings of this conference.