EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN-EP/84-151

12 November 1984

A FAST DATA COLLECTION AND ANALYSIS SYSTEM FOR A MAGNETIC SPECTROMETER

P. Denes^{*)}

Department of Physics and Astronomy, University of New Mexico,
Albuquerque, N. Mex., USA

ABSTRACT

The data collection and pre-processing system used in LEAR experiment PS183 is described.

(Submitted to Nuclear Instruments and Methods in Physics Research)

^{*)} Present address: CERN, Geneva, Switzerland.
Supported by DOE contract DE-AC04-76ER03347.

Experiment PS183 [1] is performing stopping antiproton-proton spectroscopy at the CERN Low-Energy Antiproton Ring (LEAR). The apparatus consists of a single-arm magnetic spectrometer possessing sufficient geometrical acceptance and detection elements to act as both a photon pair spectrometer and a single-charged-track spectrometer, in the region of roughly 0 to 1 GeV/c. The high trigger rate (30,000 to 100,000 stopping antiprotons per second) as well as the relative complexity of the apparatus necessitates a data collection system that is rapid as well as economically feasible.

To satisfy these requirements and provide a system that is reliable and very easy to use, an interconnected multiprocessor complex has been constructed, centred around the LeCroy 4800-series CAB microprocessor. The CAB is a fast, on-board CAMAC computer based on the AM2901 arithmetic logic unit and AM2910 microprogram sequencer, and on the useful arithmetic options of a TRW 1010 J TDC multiplier-accumulator and an AM25S10-based shifter array. Besides the obvious speed advantage of the CAB (the microinstruction execution time is 200 ns), the immediate CAMAC dataway (for the crate controller version) and parallel branch highway (for the branch highway driver version) access makes the CAB an ideal front-end processor for particle physics experiments. The CAB has been complemented, in the PS183 system, with a PDP-11/60 for writing tapes and performing off-line analysis, and with an HP9826. The HP9826 is used not only as a CAB development machine, but also as an electronics tester and graphics device during off-line analysis, and as a system supervisor and checker while the experiment is running.

The experimental apparatus, shown in Fig. 1, implements proportional wire chambers (B, C1, C2, D1, D2, E1, E2, Z) using the PCOS III readout system; drift chambers (NDC, RDC, PDC) using the 4290 readout system; and scintillators. The antiproton beam passes through two beam-defining chambers (XY, XY), a trigger scintillator S, and impinges on a LH_2 target. Interaction products may subsequently enter the spectrometer magnet, each track forming one of three basic topologies, an R track (the track enters the spectrometer, and reflects back out of the front of the magnet); a T track [the track is trapped in the spectrometer, hitting one of the side (D or E) chambers]; or a P track (the track penetrates through the spectrometer Z and the PDC chambers). Roughly 2% of the incident antiprotons produce triggers, the bulk of which are single charged tracks, the remainder being multiple charged tracks or e^+e^- pairs produced by photon conversion in a thin lead radiator placed directly in front of the B chamber. Track momenta are then reconstructed based on the chamber information, and vertex information is obtained by using a combination of the NDC, the RDC, the beam, and proportional chambers.

The trigger is formed in a straightforward way: the coincidence of an S and a Q counter hit and of a hit in any of the other scintillators generates a 'pretrigger' which serves to clock a two-stage programmable logic unit system. The first stage essentially encodes hodoscope multiplicity; the second stage determines whether the pattern forms a valid trigger, and if so, issues a trigger and encoded event topology number. The programmable logic units are down-loaded by the HP9826 whenever a new trigger configuration is desired.

The data collection system is organized as follows. There is a master CAMAC crate housing a CAB BG* branch highway driver and a CAB CG* crate controller. Following this there are several CAMAC crates containing data collection elements, each crate having both an A2 and a GPIB controller, allowing simultaneous CAB and HP9826 access. The primary crate contains the single-board interface/buffers used to control the chamber systems, and the subsequent crates contain ADCs, TDCs, scalars, etc. The PDP-11/60 is connected to CAB C through a Kinetic Systems 2051 branch highway driver, and the HP9826 is connected to both CABs through a GPIB interface. An additional interconnection is provided between the CAB system and the HP9826 through serial data transmission buffers, which are loaded by CAB C on a per demand

basis with a single event, enabling the HP9826 to produce 'diagnostic event snapshots'. In addition, an IBM PC is driven by the HP9826 to provide remote beam chamber and scaler information for use in the beam magnet control room.

The CAB B performs all the data collection. Since the CAB event throughput (> 2000 events per second) is far greater than the tape writing speed (~ 300 events per second with two 1600 bpi, 75 ips drives) an intermediate buffering device (CAB C) is implemented to improve over-all performance. Each of the possible event topologies is allocated a 1 kword buffer in a 4302 extension memory connected to CAB C. Incoming events from CAB B (approximately 100 words long) are then stacked in these buffers for subsequent block transfer to the PDP-11/60. The use of a 4302 is not only necessary in order to increase the CAB memory size from the on-board 4 kwords, but is also quite convenient for memory-to-memory block transfers owing to the independent indexing of on-board and extension memories (on-board memory indexing is clumsy, since there is only a single, loadable, write-only memory index register).

Two basic modes of CAB B to CAB C transfer are used: for data that one wishes to ensure are written on tape (such as rare-event topologies or scalers) a demand transfer mode is used, in which CAB B will wait indefinitely for CAB C to become ready for data transfer. For more copious events, a queued transfer mode is used, in which CAB C must be ready at the end of the event, or the event is discarded. In either case, the transfer synchronization is controlled entirely by a single latch connected to a front panel input and output of each CAB. Each CAB can sense the state of the latch, which is set by CAB B and reset by CAB C. The CAB B detects that CAB C is free by the absence of the latch output. Once this state occurs, CAB B loads its dataway communications register with a transfer word count and sets the latch. The CAB C loads the word count, and the data are transferred at 1 MHz; immediately following this, CAB B is released. The integrity of the timing is maintained by CAB B, which senses the transfer status through the C bit of its dataway communications register. (As far as CAB C is concerned, CAB B is simply another CAMAC module, hence the data must be valid in the fashion of the CAMAC dataway standard.)

The data transfer to tape is conceptually simple, as the collection and formatting of the data is performed by the CAB system. On interrupt (branch demand), the PDP-11/60 prepares for DMA (Q-stop) transfer from the CAB, and then DMA to tape. This sequence thus operates almost entirely in a non-processor-request mode, leaving the PDP-11/60 processor free to make the usual diagnostic histograms (chamber profiles, TDC spectra, etc.). Since the actual transfer of the data through the 2051 operates at far less than bus cycle speed, and owing to the high operating system (RSX-11M) overhead associated with interrupts, an obvious boost in performance, which nonetheless retains the advantages of incorporating a UNIBUS PDP-11, could be obtained by adding to the CAB-CAB system an additional CAMAC processor (such as the CES 2180 J-11-based microprocessor) connected to the PDP-11/60 by a DR11-W/DRV11-B DMA interprocessor link.

Analysis of the collected data consists, essentially, of identifying an event topology, making a few simple cuts (based on particle identification or vertex location), and reconstructing a momentum. The high speed of the CAB makes it a logical on-line event pre-processor. Owing to the arithmetic options and powerful AM2901 instruction set, rapid integer addition, subtraction, multiplication, and logic may be performed. However, any floating-point operations must be performed entirely by software emulation and are therefore quite time-consuming.

Track identification makes use of the PLU-returned topology code, which is strobed into an input register read by the CAB, and is available for subsequent use as a task dispatch address. The wire chamber data, which become valid long before any of the other digitized data, are sorted and loaded into chamber stacks. The first word of each stack is a stack pointer—that is,

the chamber plane multiplicity. The resulting multiplicities are then concatenated to form a single word. Based on the hardware topology code, only a few chamber multiplicity patterns are valid. These are readily checked by masking the multiplicity word, and then comparing with a check word. Once the topology has been identified, and since the chamber stacks have already been loaded, a track vector is formed by simply selecting the appropriate hits.

In principle, then, to reconstruct a track momentum is trivial. Since the track momentum is uniquely specified by three coordinates, the hits of the track vector would simply point to a location in a large momentum table. However, owing to the large number of wires, the amount of memory required to do this would be prohibitively large. Various interpolation and correction schemes are therefore used to allow coarser binding of the wires (such as looking up a central momentum and its gradient, and forming a first-order correction based on the deviation of the hits from the bin centre). The entire reconstruction process, including histogramming the final result, is performed in less than 60 microseconds per track. Subsequent cuts, such as vertex or particle type, are then rapidly performed, since they tend to involve only simple calculations.

Since there does not yet exist a high-level language for programming the CAB, some practical examples may be useful to give a flavour of the type of programming required to use it. A Q-stop block transfer may be performed at full CAMAC speed, since the time required to issue the CAMAC instruction is only one microcycle (200 ns), leaving free time for memory indexing:

```

      .
      .
      .

Read:  VAR    FNA(0,1,0)          ; Define the CAMAC instruction
      .
      .
      .

QSCAN: MOVE   Read      CAMAC    ; Initiate the dataway or branch operation
      INC     B0A0      B0        ; Register 0 is used as the memory index.
                                      ; This instruction places the current value
                                      ; of R0 in the ALU output register, increments it,
                                      ; and stores the result in R0
      MOVE    ALU        MEMX     ; Load the memory index register
      BRFC    .           ; Wait for the dataway cycle to finish
      MOVE    DATAWAY @BUFFER    ; Store the data
      BRFQ    QSCAN       ; Repeat if Q

```

Notice that if (auto-indexing) extension memory were used, the two instructions performing memory indexing would not be required. A simple, general-purpose crate controller emulator may be constructed using the AM2910 program index register to point to different FNA routines:

```

      .
      .
      .

TASKS: VAR    READ          ; Set up a task dispatch table
      VAR    CONTROL
      VAR    WRITE
      VAR    CONTROL

```

```

.
.
.
IDLE:  BRNL . ; Wait for BHD command (GL 24)
        MOVE CCR SHIFTER #2; Get the 2 MS bits of the CAMAC
        ; Command Register (FFFF FxxN NNNN AAAA)
        MOVE SHIFTER/H MEMX ; and place them in the program
        MOVE @TASKS PCX ; counter index register
        @BR
.
.
.
READ:  MOVE CCR CAMAC
        BRFC .
        MOVE DATAWAY BRANCH ; Transmit 16 bits to the BHD
        BR DONE
WRITE:  MOVE BRANCH DATAWAY ; Transmit 16 bits from the BHD
        MOVE CCR CAMAC
        BR DONE
CONTROL: MOVE CCR CAMAC
        BR DONE
DONE  G BX ; Generate a Branch X
        BRNQ . + # 2
        G BQ
        G BTB ; The CAB protocol requires the
        ; CAB to generate the branch timing
        ; signal BTB,
        BRFL . ; and wait for the driver signal
        ; BTA to become unasserted
        BR IDLE

```

Since CAB program starting and stopping may be controlled from the branch highway, it is easier to have separate program sections generating CAMAC C, Z, I, etc., than to attempt to perform these functions by TYPE A CC emulation.

In conclusion, it should be pointed out that the high performance of the PS183 system, whilst achieved primarily by fast, microprogrammed microprocessors, benefits enormously from its multiprocessor environment. Each of the three different types of computers utilized—fast microprocessor, versatile technical desk-top, standard miniframe—have been assigned those tasks to which they are optimally suited. A system based solely on fast microprocessors, whilst ideal for data collection, is difficult to debug and use. The inclusion of a multi-interface, easily programmed desk-top adds the ability for numerous real-time diagnostic checks, as well as a trivially simple user interface. Finally, it should be noted that the same system provides a powerful data analysis tool: events off-line-analysed by the PDP-11/60 are readily sorted and histogrammed by the CAB system for eventual graphic presentation by the HP9826.

REFERENCE

- [1] A. Angelopoulos et al., Preprint CERN-EP/84-47 (1984).

Figure caption

Fig. 1 : The PS183 detector.

Fig. 1