Fermi National Accelerator Laboratory

FNAL/C--91/317

DE92 004119

Control and Data Acquisition Electronics for the CDF Silicon Vertex Detector

K. Turner, C. Nelson, T. Shaw and T. Wesson

Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510

November 1991

* Presented at the *IEEE Nuclear Science Symposium*, Santa Fe, New Mexico, November 2-9, 1991.

Disclaimer

Control and Data Acquisition Electronics for the CDF Silicon Vertex Detector

K. J. Turner, C. A. Nelson, T. M. Shaw and T. R. Wesson,
Fermi National Accelerator Laboratory[1]

Abstract

A control and data acquisition system has been designed for the CDF Silicon Vertex Detector (SVX) at Fermilab. The system controls the operation of the SVX Rev D integrated circuit[1] (SVX IC) that is used to instrument a 46,000 microstrip silicon detector[2]. The system consists of a Fastbus Sequencer, a Crate Controller and Digitizer modules.

I. INTRODUCTION

The CDF Silicon Vertex Detector is instrumented with the SVX IC that manages 128 channels of silicon microstrip detector. The detector is constructed as two barrels, each with twelve wedges[3]. The 1,920 channels per wedge are handled by 15 SVX ICs that share a common readout and control bus. Twelve signals control the sample and hold operations within the amplifiers, as well as the event readout. Sample and hold operation is synchronized with the 3.5 usec beam crossing interval.

The SVX data is delivered to the CDF Data Acquisition network[4], which is a mature Fastbus based system with well defined hardware, software and event formats. A Fastbus broadcast message initiates readout of the data into local buffering that can accommodate four events. Prior to acquisition by the Event Builder, header information is attached to each event and a signal is returned to the system indicating that readout is complete.

The scope of this paper is a description of the hardware design for the CDF SVX data acquisition electronics. The System Architecture section discusses the method chosen for integrating the SVX Detector system into the CDF Data Acquisition network and synchronization with the Fermilab collider beam crossings. The system modules section describes the Sequencer, Controller and Digitizer. A separate section is devoted to error detection and diagnostics. The software tools[5] that were developed over the course of the project as well as the commissioning [6] of the completed system are discussed in other papers presented at this symposium.

II. ARCHITECTURE

A. *The System*

Some of the factors that determined the system design were the detector architecture, the operation of the SVX IC and its need for a variety of clocking sequences, the required readout speed and the existing CDF data acquisition network. Existing equipment and methods were used when appropriate. In the minimum system, a Fastbus Sequencer controls an SVX Rabbit Crate[7] containing up to 8 Digitizers and is interfaced to the CDF data acquisition system thru a SLAC Scanner Processor (SSP) as illustrated in Figure 1. The CDF SVX detector is instrumented with four Fastbus crates and four SVX Rabbit crates each with 6 Digitizers to accommodate the 24 SVX wedges.

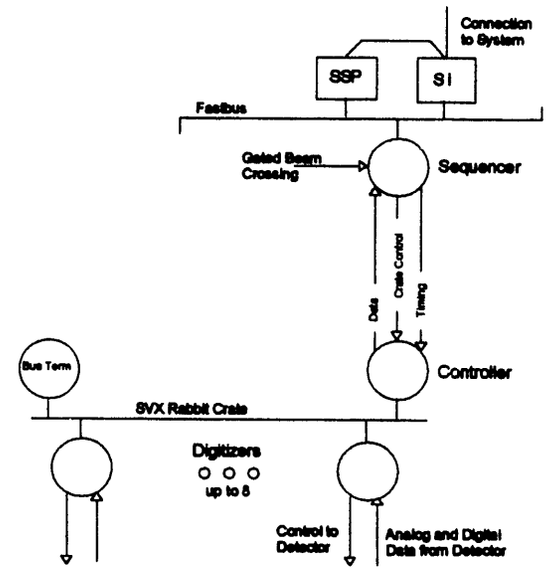

Fig. 1. CDF SVX System Diagram

The standard interface to the CDF data acquisition system is provided by an SSP, a commercial product that is used extensively in the CDF network. It is a programmable Fastbus master that can reformat the event data and attach header information. Additionally, it has adequate memory to accommodate the required four event buffers.

The Sequencer[8], a Fastbus slave, was designed to provide more capabilities than an earlier CAMAC system[9]. It is a programmable module that provides the clocking signals necessary to operate the SVX IC and contains storage for one event as a pipeline to the SSPs. The Fastbus crates that contain the SSPs and Sequencers reside in the CDF control room.

The Controllers and Digitizers[10] are housed in SVX Rabbit Crates on the CDF Central Detector. Rabbit was chosen as a package design that was familiar and readily available within CDF. Mounting areas for the crates and power supplies were already in place, having been anticipated

[1] Operated by the Universities Research Association under contract from the U.S. Department of Energy.

early in the design of the CDF central detector. The SVX Rabbit Crate protocol is a redefinition of the backplane signals provided for in the CDF Rabbit protocol.

The Controller provides the interface between the Digitizers and the Sequencer. The Digitizers are connected to the Detector and are readout by the Controller.

B. Control and Data Flow

The SVX Rabbit system is reminiscent of CAMAC. The Controller occupies the right-most position in the crate and reads from, or writes to, the Digitizers in response to control functions and addresses. A single crate can contain up to eight Digitizers.

Five cables connect the Controllers to the Sequencer. The cables have 18 twisted pairs inside a common ground shield. Signals are transmitted via differential ECL, in one direction only. The SVX Timing cable, the Write-Data cable and Command cable carry signals from the Sequencer to the Controller, while the two Read-Data cables carry data back to the Sequencer. Data writes are 16 bits and data reads are 32 bits where the upper 16 bits identify the source of the data.

Data transfers are generated with "FNA" commands similar to CAMAC and consisting of a 3 bit function code F, a 4 bit module position address N and a 4 bit subaddress A. The Sequencer can read from and write to both the Digitizers and the Controller. It is also possible to direct the read information out a front panel port on the Digitizers so as to operate with an online monitoring system under development in Italy. The primary mode for reading out wedge data in the fully implemented system is via automatic scans in which the Controller reads the wedge data register in each of its Digitizers reporting the presence of data. This will result in as many as 6 words of data depending on the number of Digitizers responding with data.

During normal beam crossings, the SVX timing signals pass from the Sequencer through the Controller and Digitizer to the wedge. During readout, the timing signal drivers turn off, and the digitizer accepts the Chip ID and Channel ID along with the analog data from the wedge. The Sequencer also issues a convert signal telling the ADC on the Digitizers when to begin the hold and digitize cycle. Once the data for a channel has been prepared by the Digitizers, readout can commence. The Controller is commanded by the Sequencer to initiate a scan read. This is an operation that involves a multiple word read of the crate causing the Digitizers, that have data, to respond. During a scan, the Controller only sends data if the Digitizer replies. The SVX IC's will normally be programmed to provide sparsified data which will cause the Digitizers to stop responding as each wedge becomes empty. The Sequencer monitors the presence of data in the wedges and terminates the readout after all the wedges have been emptied. The 32 bit Read-Data path from the Digitizer to the Sequencer carries data during a read and the presence of data status from each Digitizer when a read is not in progress. Both the data and its status can be masked off. The data from any wedge can be masked in the Controller and data status is maskable in the Sequencer.

III. SYSTEM MODULES

A. The Sequencer

The block diagram of the Sequencer is shown in Figure 2. At the heart of the Sequencer is a microprogram engine. It consists of a micro sequencer chip, a program memory array and pipeline register and produces the clock patterns required to operate the SVX IC. The Event Memory and End of Buffer and Memory Management logic provide for temporary storage of the data for one event. All registers and memories are accessible thru Fastbus.

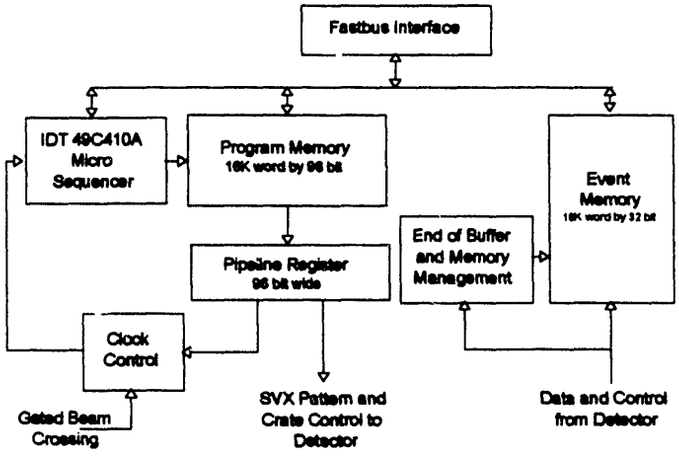

Fig. 2. Sequencer Block Diagram

The IDT 49C410A[11] micro sequencer contains a 16 bit program counter, 33 deep stack and register that can be used for loop control. External logic provides for 32 branch conditions in their true and complement form as well as an unconditional branch. Instructions from the program memory are executed in the pipeline register. This register can be thought of as a set of fields that describe functions to be performed during the present instruction execution. Some of the pipeline register fields control how the micro sequencer generates the address of the next instruction. This address is used to point to the program memory location of the next instruction to be executed. That instruction is loaded into the pipeline register to begin the cycle again. The other fields in the pipeline register are used to generate the signals that control the SVX IC operation. In this way transitions on SVX control lines can occur as often as each instruction is executed.

The program memory is a 16K word by 96 bit array and has three functional pieces as illustrated in Table I. The pipeline register is also 96 bits in width, with a pipeline register bit for each bit of the program memory array. The Microprogram Control determines which location in the program memory to get the next instruction from. The Crate

Control allows read/write access to any register in the SVX Rabbit Crates, and the SVX Pattern Control governs the timing transitions for the SVX IC.

Table I
Instruction Fields

Microprogram Control	Crate Control	SVX Pattern Control
Micro Instr(3:0)	Sub Adr(3:0)	Vernier Ctrl(3:0)
Done	Slot(3:0)	NIM(3:0)
XQT Hold	Function(2:0)	*Vernier Adjustable*
Seq Function(2:0)	Test(3:0)	NIM(3:0)
Branch Ctrl(6:0)	Data(15:0)	Digitizer(5:0)
Branch Adr(15:0)	XQT	Pattern(11:0)

The Sequencer provides the timing signals necessary to operate the SVX IC and these signals must be synchronous with the Beam Crossing. Additionally, these signals must be tailored to the requirements for each phase of sampling and acquiring/rejecting an event as well as calibration and monitoring of the detector. The clock frequency for the microprogram engine is 12.5 MHz (80 ns period). The timing signals that operate the SVX IC are controlled directly by the program running in the engine. To eliminate timing jitter associated with the sequencer clock, the clock must be synchronized with every beam crossing.

To accomplish this, the Sequencer's clock is started by an external signal that is related to the beam crossing, and it is stopped before the next beam crossing signal. During the time when the sequencer's clock is operating, the internal program is generating the SVX IC timing pattern appropriate for sampling an event. The Sequencer program will execute 43 instructions during the 3.5 usec interval between beam crossings, which includes the overhead of stopping and restarting the clock.

It is necessary to control the widths of some of the timing pattern lines to within 10 ns over a period of 1usec. Since the edges of these lines are determined by the 80 ns clock period a method for delaying its arrival was developed and is illustrated in Figure 3. The SVX Pattern Control field of the instruction controls a delay line circuit that produces clocks for the pattern portion of the pipeline register. A three bit field of the instruction selects one of 8 possible delay taps for use in clocking the SVX Pattern.

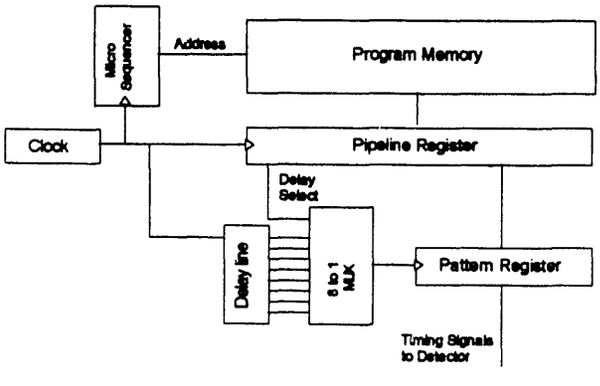

Fig. 3. Pattern Timing Block Diagram

The relationship between the instruction clock, the delayed pattern clock and the resulting timing signal is illustrated in Figure 4. In that the instruction being executed determines the delay for the pattern clock, a delay of zero results in a 10 ns relative shift for all timing signals with respect to the instruction being executed. If a 5ns delay is selected, the pattern clock is delayed by 5 ns as is the timing signal. This allows both the leading and trailing edges of a timing signal to be adjusted in 5 ns increments across a range of 35ns. When both leading and trailing edges are considered, this provides for 70ns of width correction on the total pulse width.

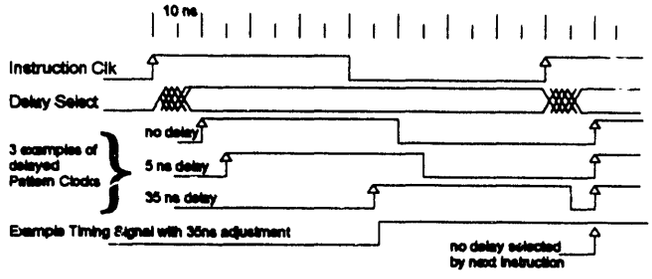

Fig. 4. Delayed Clock Timing

The Event Memory is separated into eight 2K buffers each holding the data from a specific wedge. Each 32 bit data word from the Digitizer contains a wedge identifier in the upper bits that is used to route the data to the appropriate buffer in the Event Memory. Each time a word is written to a wedge buffer, a register associated with that buffer is loaded with the physical address of that word in the Event Memory. This set of registers, called the End of Buffer Registers, will always contain the address of the last location that was written. When the event is readout by the SSP, these registers are used to issue a Fastbus end of buffer response.

B. The Controller

The Controller interfaces the Sequencer to an SVX Rabbit Crate of up to eight Digitizers. A block diagram of the Controller is shown in Figure 5. It provides read and write access to all registers in the crate and passes the timing signals for the detector from the Sequencer to the Digitizers. The function, slot and subaddress information from the Sequencer is used to direct single word read and write operations as well as multi-word readout scans and broadcast operations. Digitizer data as well as status indicating the presence of data is issued to the Sequencer upon request. The front panel has five high density 36-pin connectors, three for receiving and another two for transmitting. Information from the SVX Timing Cable is passed directly to the RABBIT backplane.

The function codes allow data, from a read operation, to be routed to the Sequencer or to a front panel port or to both. In a write to the Digitizers, control information along with the Write-Data from the Sequencer is driven onto the RABBIT backplane and the appropriate addressing and control lines set to cause the specified register on the addressed Digitizer to

accept the data. In a read operation, the Read-Data cable will be driven by the Controller along with a signal to tell the Sequencer that the data is valid. Data can come from either the Digitizers or from internal Controller Registers. It is also possible to direct the read information to a front panel port on the Digitizers. In addition to single word operations, the Controller can be commanded to scan the data from an entire crate eliminating the need to directly address a Digitizer for each word of data; a crate scan pushes data at a 12.5 MHz rate.

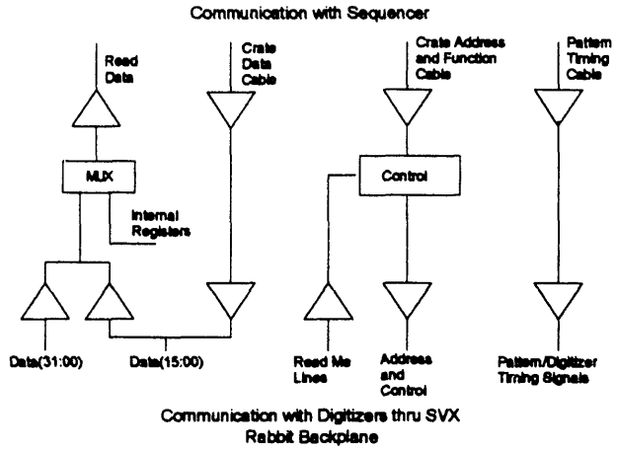

Fig. 5. Controller Block Diagram

C. *The Digitizer*

The Digitizer passes the clocking signals from the Sequencer to the detector and receives digital data and analog signals in return. It contains registers for storage of digital data and a section for processing and conversion of the analog data. All of the digital information is formed into a 32 bit word that can be readout by the Controller.

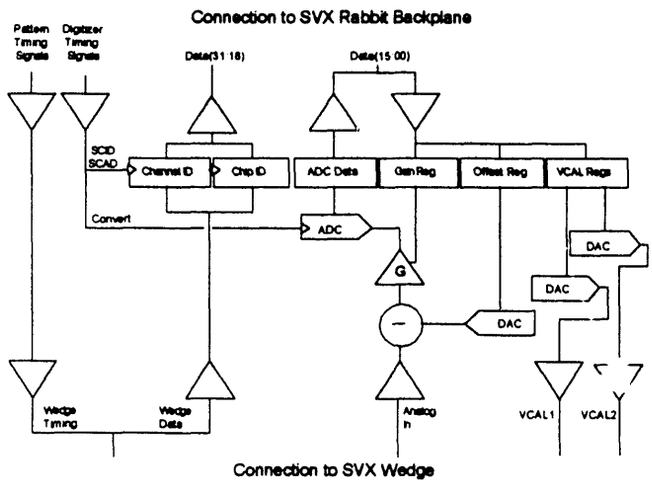

Fig. 6. Digitizer Block Diagram

As shown in the block diagram in Figure 6, the Digitizer contains a 12 bit ADC, three 12 bit DACs and registers for data, control and status. The ADC is a Datel 132S, with built in sample and hold, and an overall throughput of 2 MHz. A large offset voltage that accompanies the signals from the detector must be subtracted before any gain adjustment can be applied. One of the three DACs supplies a voltage for this purpose. The programmable gain can be adjusted over a range of 15 equal steps. Each SVX wedge contains 2 calibration pulsers, each controlled by a separate calibration voltage. The pulsers are fired by one of the timing signals supplied by the Sequencer. Two of the three DACs supply these separate calibration voltages.

IV. ERROR DETECTION AND DIAGNOSTICS

Since access to the SVX Rabbit electronics is limited, identification of the source of errors played an important role in the system design. Some of the capabilities of the system aimed at problem diagnosis are discussed here.

A. *Data Integrity*

Because parity is not included within the SVX readout system, the Digitizer provides a set of pattern words along with the event data that give a snapshot of the health of the data link anywhere along its path. Short lived intermittent problems will not be detected with this method, however stuck bits and hard cable or backplane failures will be found quickly. Even though manual scanning of data records is clearly not the method of choice for system integration diagnostic techniques, when it becomes necessary, visually distinct patterns in the data have been a valuable aid. This method of detecting interconnection failures has become preferred over the detection of parity on every transfer. Checksum and error detect/correct methods were not pursued due to the overall loss of speed that would be incurred.

During the readout process, the data from each wedge is interleaved with its neighbors. The Digitizer attaches a unique wedge identifier to each data word. This identifier is used by the Sequencer to route the data into specific wedge buffers. The wedge identifier can be used at any point between the Digitizer and offline analysis as a check that the data paths were properly setup throughout the entire data acquisition process.

Once the integrity of the data path is suspect, a Sequencer program can be used to verify the operation of the Program Memory and buffers associated with issuing commands to the Controller and also verify that the Event Memory can correctly accept data every 80 ns, by connecting the Sequencer's transmit cables into its own receivers connectors. Further checks of registers in the Controller and Digitizer the can usually pinpoint the module or cable that is in error.

In addition to digitizing the analog input from the wedge, the Digitizer's ADC can be used to digitize the 3 on board DACs and the crate's power supply voltages. In that none of these can be considered a standard or reference, part of the checkout procedure for the Digitizer includes setting the ADC's

offset and gain with an external reference. The ADC's gain and offset will be adjusted so that the response is within a few counts of nominal over the full range.

B. Timing Integrity

Ultimately the Sequencer's clock determines the timing for the measurements taken on each event and variations in its frequency or variations in the external signals that synchronize it with the beam crossing are only detectable through careful observation of the data. Some intermittent timing problems could mistakenly be attributed to noise or instabilities with the detector's cooling or power and remain unsolved. The Sequencer has a circuit that monitors the number of instructions executed between each beam synchronization signal. If this number is ever not equal to the correct number of instructions an error is indicated and the information is included in the event data. Both variations in the time between crossings and variations with the Sequencer's timing can be detected with this monitor.

The operation of the SVX IC is such that it is desirable to monitor time periods that are not within the direct control of the SVX electronics. The Sequencer has a 32 bit counter that is operable under program control and can be used to dynamically account for these times during normal running conditions. The counter can be incremented on every instruction independent of any other instruction actions which gives a basic sampling granularity of 80 ns. The contents of the counter can be easily include as a part of the event data.

The bus that connects a Digitizer with the wedge is the last place that the timing pattern that controls the SVX IC can observed. This area is within a radiation enclosure and is not accessible while the accelerator is operating. The Digitizer can be set in a mode that places the signals from this bus on the data lines thru the Controller to the Sequencer. Once in the Sequencer, they can be written to the Event Memory. More importantly, since these signals contain timing information, they can be observed directly through a port on the Sequencer's front panel.

V. IMPLEMENTATION TOOLS

A. Programmable Logic Devices

The control logic for the Sequencer and Controller designs was implemented with Programmable Logic Devices, PLDs. The Fastbus interface on the Sequencer was implemented with 5 PLDs and a total of 59 PLDs were used in the entire design. The parts were restricted to 5 ns propagation delay 20L8s and 22V10s with 29MA16's used for a portion of the design requiring multiple individually clocked latches. The Controller was implemented with 13 ECL PLDs; the 1016R8 and 1016P8 each with 6 ns propagation delay. DATA I/O Corp.'s PLD design software, ABEL, was used to write the PLD equations and Digital Equipment Corp.'s Code Management System, CMS, was used to manage the entire PLD design. The use of this tool aided us tremendously however to better manage a digital control project of this size an integrated PLD management system is needed that addresses the same issues as ABEL for an integrated multi PLD control design. To our knowledge this has not yet been pursued by the industry.

B. Zip Module Fabrication

In order to conserve space on the Sequencer board a ZIP module was designed to accommodate a byte wide slice of the Program Memory, Pipeline Register and associated three-state buffers. In the final design the module was also used for the Event Memory array. The .79" x 3.11" boards are constructed as ZIP modules that have 63 pins on a .1" grid that are offset by .05" in a zizag pattern from side to side. In production quantities of 300 modules, a single module costs $70.

A ZIP module tester was built that operates from a CAMAC Crate controlled by an IBM. The tester accommodates 16 modules and is operated by a diagnostic package that was written in Basic. The software has not been optimized but when compiled and operated on a 16 MHz IBM SX a complete test of the 16K x 8 memory, 8 bit register and four data paths takes 8.5 hours for 16 ZIP modules. Out of an initial 80 modules produced in house there was one failure.

C. Module Statistics

Table II lists a few statistics about the physical implementation of each module. At the time of this writing, the final production version of each design is either complete or very close to complete. The board density along with the number of physical layers is listed to indicate the relative complexity of the printed circuit board. Density is computed as square inches per equivalent 14 pin IC's (EIC), where EIC is the number of pads (excluding vias) divided by 14. Since EIC does not include area, this number is a measure of routability rather than an indication of whether the parts will fit on the board. The highest density listed is .4, and .8 is the lowest density.

Table II
Module Statistics

	Sequencer	Controller	Digitizer
Cost/brd	$7,500	$1,200	$800
Develop	2.5 years	1.5 years	2.5 years
People	2 Eng/1 Tech	1 Eng/1 Tech	1 Eng/1 Tech
Status	4rd Rev - final	5th Rev - final	3rd Rev - final
Power	96W/5V@17A	60W/-5V@10A	36W/±5V@6A
Density	.4 in^2 / EIC	.7 in^2 / EIC	.8 in^2 / EIC
#layers	9 (5 sig, 4 pwr)	6 (2 sig, 4 pwr)	4(2 sig, 2 pwr)
Logic	TTL - F	ECL 10KH	TTL - F

Table III lists a few timing parameters for each module. The AS-AK and DS-DK times listed for the Sequencer refer to the address connection time and data transfer latency for Fastbus operations. The Controller Crate Scan time is for a scan read of eight Digitizers. The Sequencer Crate Access time is the instruction overhead for any operation between the

Sequencer and Controller. It includes the instructions for initiating the action and detecting that the operation is complete. There is an additional overhead of 850 ns for the round trip propagation delay associated with the 250 foot cable between the Sequencer and Controller.

Table III
Module Specific Timing

Sequencer		Controller		Digitizer	
AS-AK	25 ns	Write	80 ns	ADC	2 MHz
DS-DK	55 ns	Read single	200 ns	DAC	3 usec - 1/2ct
Crate Access	320 ns	Crate Scan	900 ns		
Clock	100 MHz				
Program Clk	80 ns				

VI. CONCLUSIONS

The design of the data acquisition system for the CDF Silicon Vertex Detector, the system architecture and system modules has been presented. The Fastbus Sequencer, a programmable timing generator for controlling the SVX Rev D integrated circuit is described along with the Controller and Digitizer modules that reside in SVX Rabbit crates. A method for providing timing patterns to the wedge that allows pulse width control to within 10 ns over a period of 1 usec is presented. Diagnostics abilities that provide mechanisms for understanding data and timing integrity are discussed. A variety of module specifications, timing and costs are given.

The system was designed to help integrate a new detector into an existing system and provide the flexibility necessary to operate the SVX Rev D integrated circuit within the 3.5 usec beam crossing intervals.

This system design and its ultimate success was in part due to close communication between the system designers and the users. Features such as the timing monitor in the Sequencer were added only after the system designer realized that errors in this area could only be detected in the data and that intermittent problems would probably be blamed on poor detector performance or instabilities in the Digitizer. Design specifications are written at beginning of a project, however the user and designer cannot part company for periods of the time punctuated by the normal design milestones of prototype, pre-production and production phases of a module. Steady frequent interaction is required to succeed with the design of a new detector system.

VII. REFERENCES

[1] S. A. Kleinfelder et al, "A Flexible 128 Channel Silicon Strip Detector Instrumentation Integrated Circuit with Sparse Data Readout", *IEEE Trans. on Nucl. Science* 35, No.1, 171, 1988.

[2] W.C. Carithers et al, "The CDF SVX: A Silicon Vertex Detector for a Hadron Collider", *Nucl. Inst. and Meth.* A289 (1990) pp. 388.

[3] W. C. Wester et al, "The CDF Silicon Vertex Detector: An Overview and Test Results", paper submitted to this symposium.

[4] E. Barsotti et al, "Fastbus Data Acquisition For CDF", *Nucl. Inst. and Meth.* A269 (1988) pp. 82-92.

[5] S. M. Tkaczyk and M. W. Bailey, "System Software Design for the CDF Silicon Vertex Detector", paper submitted to this symposium.

[6] S. M. Tkaczyk et al, "Commissioning of the Control and Data Acquisition Electronics for the CDF Silicon Vertex Detector", paper submitted to this symposium.

[7] G. Drake et al, "CDF Front End Electronics: The Rabbit System", *Nucl. Inst. and Meth.* A269 (1988) pp. 68-81. Rabbit was chosen as an available package. This paper is included for those interested in Rabbit, however it is not necessary for understanding the SVX implementation.

[8] K. J. Turner et al, "CDF SVX Sequencer ", Fermi National Accelerator Laboratory, internal design document, 1991.

[9] F. A. Kirsten and C. Haber, "A Versatile, Programmable Control and Data Acquisition System for Complex Integrated Circuits", *IEEE Transactions on Nuclear Science* , 1989.

[10] C. A. Nelson and T. M. Shaw, "Manual for the SVX Digitizer and Controller Modules", Fermi National Accelerator Laboratory, internal design document, 1991.

[11] Integrated Device Technology, Inc., "IDT 49C410 16-Bit CMOS Microprogram Sequencer", Logic Data Book (1990-91).

END

DATE FILMED

01/24/92