Development of a Front End Controller/Heap Manager for PHENIX[1]

M. N. Ericson, M. D. Allen, M. S. Musrock, J. W. Walker, C. L. Britton, Jr.,
A. L. Wintenberg, and G. R. Young

Oak Ridge National Laboratory, Oak Ridge, TN 37831-6006

Abstract

A controller/heap manager has been designed for applicability to all detector subsystem types of PHENIX. The heap manager performs all functions associated with front end electronics control including ADC and analog memory control, data collection, command interpretation and execution, and data packet forming and communication. Interfaces to the unit consist of a timing and control bus, a serial bus, a parallel data bus, and a trigger interface. The topology developed is modular so that many functional blocks are identical for a number of subsystem types. Programmability is maximized through the use of flexible modular functions and implementation using field programmable gate arrays (FPGAs). Details of unit design and functionality will be discussed with particular detail given to subsystems having analog memory-based front end electronics. In addition, mode control, serial functions, and FPGA implementation details will be presented.

I. INTRODUCTION

Front end electronics (FEE) modules in any physics experiment require some level of control. In some systems this control resides at a higher system level through a simple control port. In other systems the control resides at the same physical level as the FEE and may vary in complexity from a simple state machine to a microprocessor-based unit. The latter approach, localized control near the detector, has become a requirement as the integrated FEE modules associated with high density, physically large detectors prominent in modern-day physics have increased the control and data acquisition requirements [1,2]. The PHENIX detector, composed of multiple detector subsystems with varying channel counts, is one such system [3].

II. FRONT END MODULE (FEM) DESCRIPTION

The front end module (FEM) is the hardware unit located nearest the detector and is comprised of FEE channels and a heap manager. The heap manager performs the following generalized functions: mode interpretation and execution, FEE timing and control, pre- and post- Level-1 (LVL-1) buffering of FEE data, and collection, packet forming, and communication of LVL-1 qualified data.

PHENIX FEMs have three classifications: Type I - AMU-based (analog memory unit) with multiple samples, Type II - Flash ADC-based with single sample, and Type III -

Flash ADC-based with multiple samples. Figure 1 shows a high level block diagram for the Type I FEM (AMU-based FEE). Mode bits are used to control the operational state of the FEM. A communications controller interprets and distributes control signals to the rest of the FEM based on the mode bit commands. An FEE state machine controls the signal processing channels including ADC sequencing. Data is collected from the FEE, formatted into a data packet with some logistics information, and transmitted over a parallel port to a DSP-based data collection module (DCM) for further data processing and reduction.
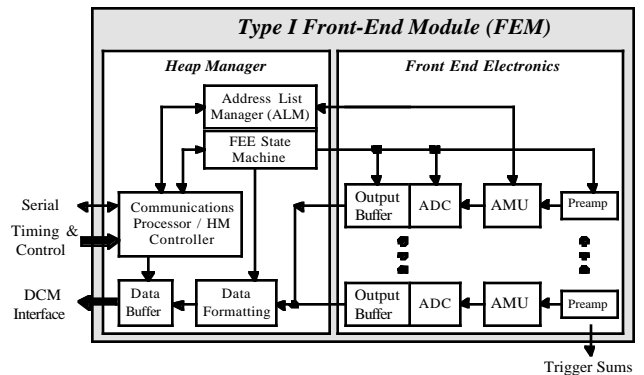

Figure 1: Generic Front End Module Block Diagram

FEMs function both synchronously and asynchronously with the beam crossings. Preamp and AMU controls must be synchronized with the beam clock while digitization, packet forming and transmission occur at a rate consistent with the required data throughput rate. Since valid events (LVL-1 qualified) can occur at rates much higher than the achievable collection rates, each FEM must be able to hold at least five qualified event data. For AMU-based FEMs, events are composed of two or more samples. These samples will not be overlapping or share individual beam crossing data. Additionally, valid LVL-1 events will occur at a rate not exceeding 25 KHz. This allows up to 40 μs for digitization, data collection and formatting, and data transmission to the DCM.

The heap manager architecture has been designed for functional modularity. Many heap manager functions are common across all FEM types. Proper partitioning of FEM functional blocks allows wide applicability across PHENIX.

A. FEM Interfaces

As demonstrated in Figure 1, four interfaces are associated with each FEM: timing & control, DCM, serial, and trigger. The timing and control interface provides the clocks, mode control bits, and LVL-1 accept signals that are synchronized with the beam. The DCM interface is a parallel interface for data retrieval from the FEM. A serial port provides means to program FEM variables and FEE control bits. The trigger interface provides a summation of FEE signals that are used in the LVL-1 decision. The signals used are highly dependent on the FEE and vary from subsystem to subsystem.

B. Mode Bits / Command Control

The PHENIX detector is a cyclic machine consisting of repeated patterns of beam events with interlaced control functions such as calibration and FEE reset operations. Mode control of each detector subsystem will be performed using a number of *mode control bits* that are supplied to each FEM. Each mode command will be sampled by the FEM on the rising edge of the beam clock and will be implemented on the following rising edge (or one beam clock period later).

In defining the mode bit definitions several objectives were considered: standardization of the command set across all subsystem types where possible, simplicity of commands, flexibility of commands for subsystem-specific functions, and developing a set that make the FEM operational state completely deterministic. A deterministic system for this application is defined as one whose functional state can be completely determined by observing the mode bits for any given beam clock (recall that the scheduler controls the FEM by issuing a mode control each beam clock). Use of a deterministic system allows one to determine the FEM operational state without knowledge of prior states. This will make both the scheduling, development, and troubleshooting tasks more efficient.

A total of 11 control bits are available for mode control, timing, and other functions. Table 1 summarizes these timing and control interface signals. Five bits are reserved for mode control (MB4-0), and three are used for providing timing signals (LVL-1, beam clock, and 4X beam clock). The remaining three bits (MB7, MB6, and MB5) are available to each subsystem and can be used or ignored depending on the control needs of each subsystem. The mode bits are output each beam clock cycle and are decoded to implement a specific command set.

The five mode bits have been split into three separate groupings, each defining a part of the operational state. MB4 is the Run/Halt Bit and defines the run status of the FEM (1=Run, 0=Halt). MB1 and MB0 are used to control the 'reset group' of commands (see Table 2). MB3 and MB2 provide another group of custom commands that are user defined (custom group). Table 2 shows the reset group mode bits and associated commands that provide a simple, flexible,

deterministic system. The custom command group is similar and provides three unique commands and a *No-op*.

Table 1
Timing & Control Interface Bit Definitions

Bit Name	Bit Definition
beam clock	beam clock - 105 ns Period
4X beam clock	4X beam clock ~ 26 ns Period
LVL-1	LVL-1 Accept
MB7	Extra Bit - No Assigned Function
MB6	Extra Bit - No Assigned Function
MB5	Extra Bit - Pulse Enable (optional)
MB4	Reserved For Command Encoding
MB3	Reserved For Command Encoding
MB2	Reserved For Command Encoding
MB1	Reserved For Command Encoding
MB0	Reserved For Command Encoding

Table 2
Mode Bit Definitions for MB1 and MB0 (Reset Group)

MB1	MB0	Command
0	0	No-op (Reset None)
0	1	Resync Reset
1	0	Initialization Reset
1	1	Special Reset

Being deterministic allows a mode command structure where each command group can produce an operation regardless of the operation specified in other groups. This allows for multiple functions to be specified at once which can be both a benefit and a problem. For instance, a *Run* can be specified with both a *Special Reset* (integrator reset perhaps) and a *Custom Command*. This makes the mode bits powerful but also allows for commands that may be incompatible if issued simultaneously. FEMs must be designed to handle invalid command sets by defaulting to a *Run* with *No-ops* for each of the other two command groups. Error conditions (incompatible or undefined group commands) should be removed from the possible command list during system checkout and initial operation.

A summary of the standard FEM commands is shown in Table 3. *Run/Halt* is used to put the FEM in a completely static state or to initiate and continue normal operation and data collection. *Init. Reset* places the FEM in an initialized state by resetting all counters and memory. *Resync Reset* provides a means to synchronously reset the beam clock counters in all FEMs to ensure event time stamp alignment. One of the custom command codes may be reserved for implementing special operations that are configured using the serial port of the FEM (*Next SL Command*). Providing this capability allows for a large number of user-supplied, custom functions that may prove worthwhile especially during system development, testing, and diagnostics.

Table 3
Mode Bit/Command Summary Listing

MB[4:0]	Command
XXXX0	Run
XXXX1	Halt
XX00X	Reset *No-op*
XX01X	*Init. Reset*
XX10X	*Resync. Reset*
XX11X	*Special Reset*
00XXX	*Custom No-op*
01XXX	*Custom Command 1*
10XXX	*Custom Command 2*
11XXX	*Custom Command 3*

All FEMs power up in the *Halt All* state. Careful design of the FEM is required so that FEMs power up gracefully and predictably.

C. Address List Manager

The address list provides simultaneous read/write control, cell write-over protection for both a LVL-1 trigger decision delay and digitization latency, and re-ordering of AMU addresses following conversion, all at a beam crossing rate of 105 ns. Addresses are handled such that up to five LVL-1 events can be maintained in the AMU without write-over. Applicability to multiple detector sub-systems is accomplished by the use of detector-specific programmable parameters -- the number of data samples per valid LVL-1 trigger and the sample spacing.

Figure 2 shows a high-level block diagram for the address list manager [4]. The method employed involves the shuffling of six-bit AMU addresses. Though this requires memory space for address holding, it provides some distinct advantages when the special functions associated with multiple event buffering, AMU cell address reporting, and data tagging are considered. The basic ALM is a recirculating loop composed of three memory units. The LVL-1 Delay FIFO is used to implement the LVL-1 trigger delay and can be programmed to be as large as 48 beam-clock ticks. After the delay period has passed, the address is placed into the ADC Conversion FIFO to await digitization if qualified, or passed to the Available Address FIFO to be re-used. The AMU addresses in both the LVL-1 Delay FIFO and the ADC Conversion FIFO are protected from being overwritten. As digitization is completed, the associated address is released and placed in proper order in the Available Address FIFO for re-use.

At initialization following a reset or power-up, a counter is used to fill the loop with addresses 0 through 63, which pass through a mux (FB MUX) into the LVL-1 Delay FIFO. When the counter outputs address 63, FB MUX is switched into 'feedback mode,' where the addresses from the Available Address FIFO are fed into the LVL-1 Delay FIFO. The input address to the LVL-1 Delay FIFO is buffered and used as the current AMU write address. A new address is always available each beam clock to prevent the overwriting of valid data.

The LVL-1 Delay FIFO acts as a programmable depth shift register. The LVL-1 delay, specified in beam clock periods, is stored in the ALM at setup. At initialization, an address is written into the FIFO each beam clock cycle. The read from the FIFO is suppressed until the number of beam clocks specified by the programmed LVL-1 delay bits has passed. This function is performed using a beam clock counter and digital comparator. Once initialized, a write and a read operation are performed each beam clock until another reset is issued.

The routing of the AMU addresses from the LVL-1 Delay FIFO is determined by the LVL-1 accept bit. If a valid LVL-1 is generated, the address is written into the ADC Conversion FIFO. This operation removes the addresses from the available address list, thus preventing them from being
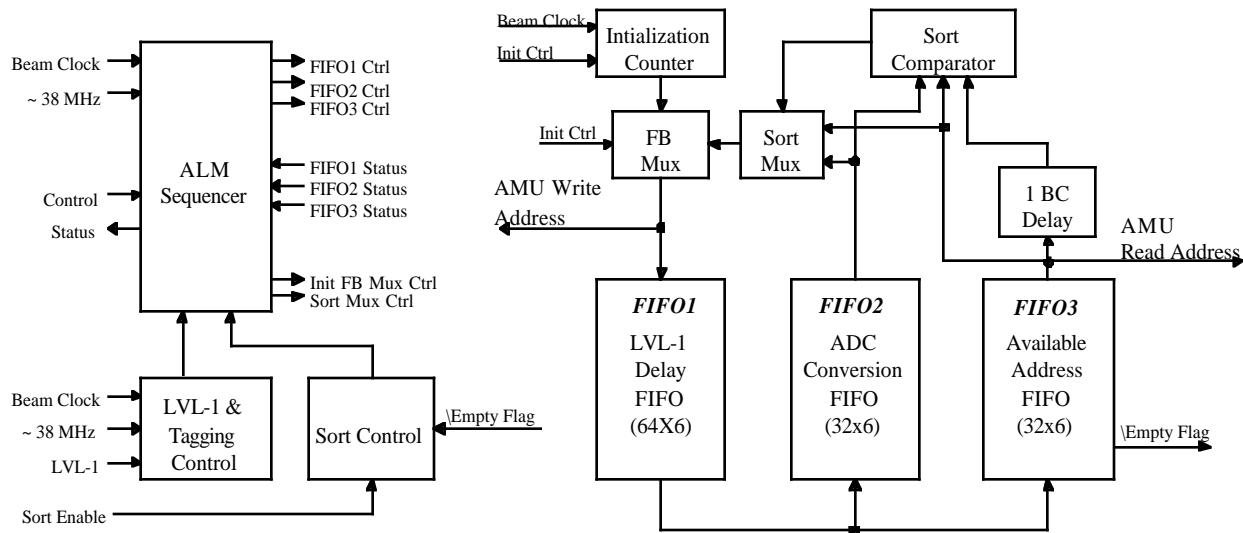

Figure 2: Block Diagram of the Address List Manager

overwritten. Other addresses associated with the same event that occur on following beam clocks are also written into this FIFO as each pops out of the LVL-1 Delay FIFO on subsequent beam clocks. Addresses that do not receive a valid LVL-1 are written into the Available Address FIFO. Reordering of addresses that have been used for digitization is accomplished using a sorting procedure that slips the address back into the appropriate location of the address loop.

D. FEM Serial Interface

The serial interface is used for configuring the FEM and can be used for monitoring and diagnostics. Various programmable parameters are accessed through this port including HM parameters (LVL-1 delay, ALM sampling parameters) and FEE parameters (channel enables, DAC settings, etc.). Figure 3 shows the basic serial architecture incorporated into each FEM. This interface uses a full readback, buffered approach that combines a set of shift registers with mirror hold registers. The interface also provides a means for programming the FPGAs during FEM initialization and reading back the DONE bit which indicates successful programming.

Among the functions accessible through the serial interface are LVL-1 delay and the LVL-1 template. The LVL-1 delay (the time between storage of an analog value in the AMU and tagging of the AMU cell via the LVL-1 accept) is programmable over a range of 32- to 48-beam clocks using four bits. The LVL-1 template uses eight bits to specify the desired sampling associated with a LVL-1 accept.
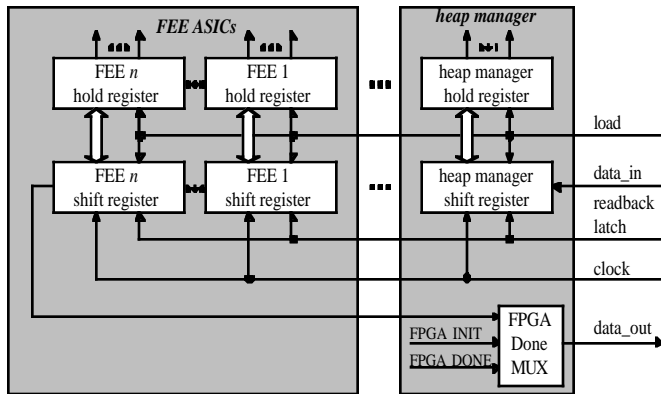
Figure 3: FEM Serial Architecture

E. Data Collection & Formatting

One primary function of the heap manager is to collect data from the FEE. For AMU-based FEMs (Type I) this consists of retrieving digitized data from a bank of ADCs. After digitization is completed each ADC register is enabled separately onto a bus. A packet is then generated which consists of a pair of header and trailer words, a unique event number, a module address, AMU cell number, the digitized data block, and a checksum. The formatting of a data packet is carried out by the heap manager controller through the use of a five input multiplexer (Data Packet MUX). Figure 4 shows how the data collection and formatting circuit is implemented.
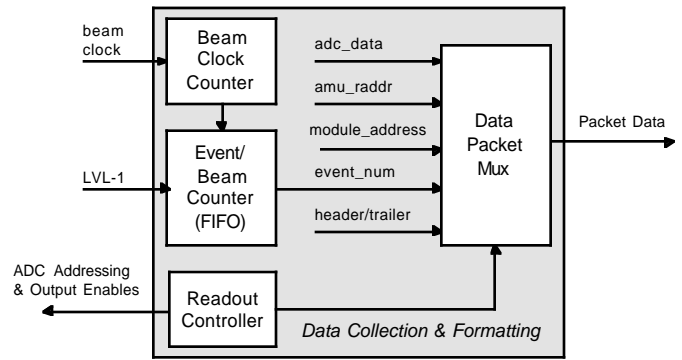
Figure 4: Data Collection and Formatting

III. SYSTEM TIMING

Data conversion, collection, and transmission is driven by the LVL-1 accept signal. Figure 5 shows the timing for the occurrence of two LVL-1 accepts. Upon receipt of a LVL-1 accept, the heap manager controller initiates an ADC conversion by reading the analog memory and initializing the ADC state machine. This process requires 3 µs. The ADC uses both edges of the conversion clock which effectively halves the required conversion time. Consequently, the ADC conversion time is the product of 1024 (10-bit ADC counter) and the 80 MHz (160 MHz internal) conversion clock period, or 7 µs. Once the ADC conversion is complete, the heap manager controller begins readout of all 256 ADC channels and adds packet formatting information, which includes the 256 ADC words as well as header and trailer information.. With buffering, one ADC conversion may be begun while data readout from a previous conversion is underway. The data is pipelined through the collection, formatting, transmission modules at a rate of one 12-bit word per beam clock. Thus, a total of 36 µs is needed to process each LVL-1 and completely transfer an event data packet to the DCM. The LVL-1 accept rate is limited to an average rate of 25 KHz so that pile-up does not occur. This allows up to 40 µs for handling the data associated with a single LVL-1 accept.
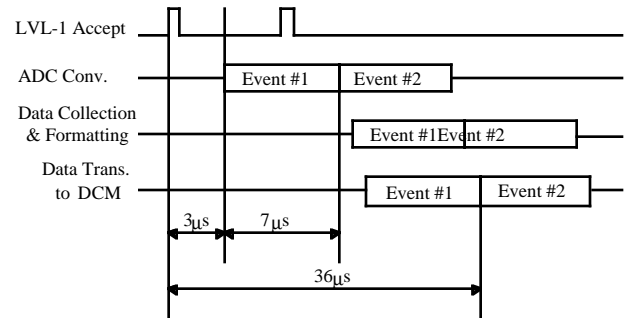
Figure 5: HM Timing diagram

For each LVL-1 accept, a pre- and post- sample are stored in the AMU for processing. Two data collection modes are available: raw and correlated. In raw mode, two data packets are generated for each LVL-1 accept and transmitted to the DCM resulting in a proportionally larger data packet. In correlator mode, the heap manager makes use of a correlator circuit in the analog memory that subtracts the amplitude of the pre-sample from the amplitude of the post-sample before

the ADC conversion takes place, resulting in the minimum data packet size. In the normal operation of PHENIX, the correlated mode will be utilized. Raw mode will be reserved for calibration and diagnostic functions.

IV. IMPLEMENTATION

A primary design requirement for the heap manager is in-circuit re-programmability, which facilitates changes that are anticipated in the early test development stages of PHENIX. The selection of Xilinx as the FPGA technology takes advantage of in-circuit programmable logic to provide the necessary resources and I/O for control of the FEE and CPU interfaces. Furthermore, the availability of RAM cells in the programmable architecture of the Xilinx families allows synthesis of FIFOs needed in the address list manager.

The Xilinx 4000E family was chosen as the platform for the construction of the heap manager. To optimize FPGA resource allocation and maximize the speed of the heap manager, the design was split into two parts: address list manager and heap manager controller. The address list manager is implemented in a single XC4005E-2, 84-pin PLCC part, and the heap manager controller is implemented in an XC4010E-2, 191-pin FPGA.

The Xilinx ProSeries software was utilized as the development platform for the heap manager. Schematic entry and VHDL were combined to produce the logic blocks used in the hierarchical heap manager design, and the XACT timing editor provided optimization necessary for minimizing delays.

A 64-channel beam test prototype of the system was developed and tested for the PHENIX Multiplicity Vertex Detector [5,6,7]. In this beam test, a heap manager controlled 64 channels of FEE (preamps, AMUs, and ADCs). Results from the beam test proved the partitioned heap manager met all functional requirements.

V. MEASUREMENT RESULTS

Continued development of a generic heap manager after the beam test resulted in a unit capable of controlling a 64-cell AMU. Both heap manager controller and address list manager were upgraded from the 16-address beam test versions to support the demands of a 256-channel, 64-address MVD subsystem. The power measurements for the fully developed heap manager are shown in Figure 6. At the specified beam clock rate of ~10 MHz, the power consumption of both FPGAs is approximately 770 mW. The unit operated properly to 4x beam clock rates exceeding 60 MHz (> 15 MHz beam clock).
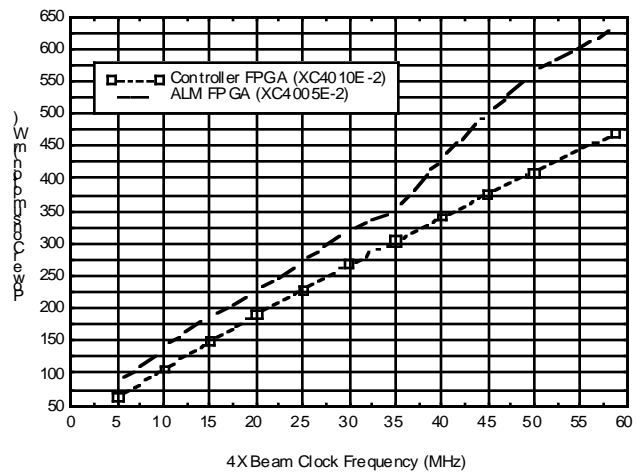

Figure 6: HM FPGA Power Vs. Clock Frequency

In addition to power and speed requirements the heap manager must have sufficient resources to allow architectural modifications after PHENIX installation. Observance of the resource allocation of both HM FPGAs demonstrates adequate resources for future design changes (see Table 4). Additionally, the speed overhead previously reported indicates that future FPGAs designs may be made denser without violating the minimum speed requirements.

Table 4
FPGA Allocation Data

Logic Blocks	ALM Allocation (Used/Total)	HM Controller Allocation (Used/Total)
I/O Pins	32/61 (52%)	126/160 (73%)
F and G Funct. Gen.	194/392 (49%)	280/800 (35%)
H Funct. Gen.	59/196 (30%)	61/400 (15%)
CLB Flip Flops	137/392 (34%)	237/800 (29%)
Total Occupied CLBs	160/196 (81%)	295/400 (73%)

VI. CONCLUSIONS

A controller/heap manager has been designed for generic application to all detector subsystem types of PHENIX. A prototype unit performs all functions associated with front-end electronics control including ADC and analog memory control, data collection, command interpretation and execution, and data packet forming and communication. The topology developed is modular so that many functional blocks are identical for a number of detector subsystem types. Programmability of function is maximized through the use of flexible modular functions and implementation using field programmable gate arrays (FPGAs). The design was partitioned into an address list manager and a heap manager controller, which were then mapped into two Xilinx FPGAs. The system operated at 4x beam clock rates up to 60 MHz, and power consumption was measured at 770 mW using a PHENIX beam clock period of 105 ns.

VII. Acknowledgments

VIII. References

[1] A. L. Wintenberg, et al., "Monolithic Circuits for the WA98 Lead Glass Calorimeter," *Conference Record of the 1994 Nuclear Science Symposium and Medical Imaging Conference*, November, 1994.

[2] A. Gara, J. A. Parsons, and W. Sippach, "Readout Electronics for the GEM Calorimeter," *Conference Record of the Second International Conference on Electronics for Future Colliders*, May, 1992.

[3] Kehoe, W. L., et al. PHENIX Conceptual Design Report. New York: Brookhaven National Laboratory, 1993.

[4] M. N. Ericson, M. S. Musrock, C. L. Britton, Jr., J. W. Walker, A. L. Wintenberg, G. R. Young, and M. D. Allen, "A Flexible Analog Memory Address List Manager for PHENIX," *IEEE Trans. Nucl. Sci.*, Vol. 43, No. 3, June 1996, pp. 1629-33.

[5] C. L. Britton, Jr., et al., "Design and Performance of Beam Test Electronics for the PHENIX Multiplicity Vertex Detector," Presented at 1996 IEEE Nuclear Science Symposium.

[6] C. L. Britton, Jr., et al., "Multiplicity-Vertex Detector Electronics for Heavy-Ion Detectors," Proceedings of the First Workshop on Electronics for LHC, Lisbon, Portugal, September 1995, pp. 84-87.

[7] J. S. Kapustinsky, et al., "A Multiplicity Vertex Detector for the PHENIX Experiment at RHIC," submitted to Proceedings of the 4th International Conference on Position Sensitive Detectors, Manchester, UK, Sept. 9-13, 1996.