A Flexible Analog Memory Address List Manager for PHENIX[1]

M. N. Ericson, M. S. Musrock, C. L. Britton, Jr., J. W. Walker, A. L. Wintenberg, and G. R. Young
Oak Ridge National Laboratory, Oak Ridge, TN 37831

M. D. Allen
The University of Tennessee, Knoxville, TN 37996

—

Abstract

A programmable analog memory address list manager has been developed for use with all analog memory-based detector subsystems of PHENIX. The unit provides simultaneous read/write control, cell write-over protection for both a Level-1 trigger decision delay and digitization latency, and re-ordering of AMU addresses following conversion, at a beam crossing rate of 105 ns. Addresses are handled such that up to 5 Level-1 (LVL-1) events can be maintained in the AMU without write-over. Data tagging is implemented for handling overlapping and shared beam-event data packets. Full usage in all PHENIX analog memory-based 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. Architectural candidates for the system are discussed with emphasis on implementation implications. Details of the design are presented including application specifics, timing information, and test results from a full implementation using field programmable gate arrays (FPGAs).

I. INTRODUCTION

Analog memory units (AMUs) play an important role in many physics experiments as data storage elements [1,2,3]. Use of these elements allows proper handling of delays associated with making trigger decisions and digitization latency [3,4,5]. Published methods used to control analog memory are bit-enabling schemes [1,2] and direct addressing [4]. Bit-enabling methods use individual enables to control the write/read operation of a particular AMU cell or column of cells. In direct addressing the AMU cells are memory mapped, and the addresses are manipulated for write/read operations. Bit-enabling methods can produce unacceptable delays due to the requirement of looking ahead for unprotected cells, which is particularly a problem if FPGAs are used for implementation. Direct addressing methods require the storage and manipulation of an address list (address list management) which can require significant amounts of memory. Additionally, the memory has to be partitioned into several separately controlled blocks. However, this method lends itself nicely to the AMU-based detector subsystems of PHENIX where multiple event handling (up to 5) with a programmable number of samples and sample spacing are required.

II. PHENIX FRONT-END MODULES

This paper presents an enhanced address list manager (ALM) designed to provide a solution for all AMU-based detector subsystems of PHENIX -- namely Multiplicity Vertex Detector (MVD), EM Calorimetry (Lead Glass and Lead Scintillator), Muon Identifier (MuID), and Muon Tracker (MuTR). Fig. 1 shows a generic block diagram of the PHENIX front-end module (FEM) for these subsystem types. Each FEM contains multiple analog signal processing

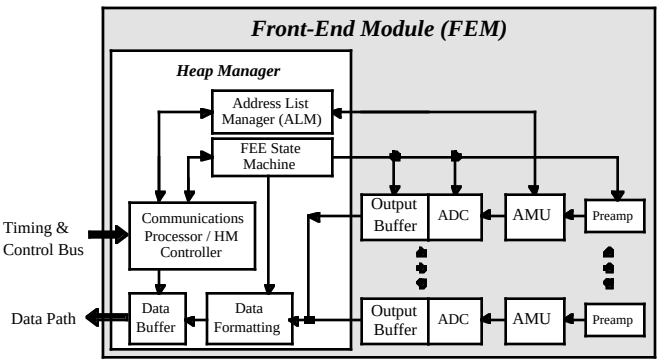

Fig. 1 Front-End Module Block Diagram

channels and sufficient control circuitry to operate as an independent system module. Timing signals and control commands are all the FEM requires to acquire data. The front-end electronics are composed of a preamplifier, AMU, and buffered ADC, that are mapped one to each detector element. All subsystems will use a voltage-write/voltage-read 64-cell deep AMU, and an 8 to 12-bit Wilkinson-type ADC per detector channel, both fabricated in 1.2µm CMOS [6]. The heap manager provides all the control for front-end sequencing, data collection and formatting, and communications functions associated with an FEM. Up to 256 detector channels will be controlled by a single heap manager. The number of associated detector channels is sub-system dependent and is related to detector pitch and physical partitioning.

[1]Research sponsored by the U.S. Department of Energy and performed at Oak Ridge National Laboratory, managed by Lockheed Martin Energy Systems, Inc. for the U.S. Department of Energy under Contract No. DE-AC05-84OR21400.

The address list manager has several programmable parameters that ensure full applicability to these 5 detector subsystems: number of samples per valid trigger, sample spacing specified as the number of beam clocks between adjacent samples, and LVL-1 trigger delay. The LVL-1 trigger delay value will be identical for all detector subsystems, but must be programmable since this value will not be exactly determined until the detector is operational. Table 1 gives the sampling requirements for the associated PHENIX detector subsystems.

Table 1
PHENIX Detector Subsystem Sampling Requirements

Detector Subsystem	*Samples per LVL-1 Accept*	*Sample Spacing in Beam Ticks*
Multiplicity Vertex Detector	2	1
EMCal		
Lead Glass	2	2-5
Lead Scintillator	2	2-5
Muon Identifier	2	2
Muon Tracker	3-4	Variable?

Besides programmability, other special functions are implemented in the ALM, including maintaining address ordering and data tagging for multiple-event separation. Each AMU must be able to store up to 5 LVL-1 qualified events and protect these events while continuously writing to unprotected AMU cells until the qualified events are digitized. By maintaining proper address control, writing to the AMU is void of dead-time. After digitization, the address is placed back into the address list in appropriate order. Address reordering helps reduce the injected noise into the AMU during cell read/write operations. When considering multiple samples and possible multiple LVL-1 events, event data packets can become highly interspersed and will require separation. Data tagging allows more straightforward separation and reconstruction of the individual beam events from overlapping and/or shared data, particularly when this function has to be performed in the FEM. The AMU controller presented in this paper performs these functions at the PHENIX beam clock rate of 9.52 MHz.

III. ALM Circuit Topology

Fig. 2 shows a high-level block diagram for the address list manager. The method employed involves the shuffling of 6-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. An additional FIFO (Sorting Memory) may be required depending on the time required to sort the address and the digitization time.

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 (MUX1) into the LVL-1 delay FIFO. When the counter outputs address 63, MUX1 is switched into 'feedback mode,' where the addresses from the Available Address FIFO are fed into the Level-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 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. Addresses that have been used for digitization are also placed in this memory following a sorting procedure that slips the address back into the appropriate location. After initialization, the Available Address FIFO must output an address each beam clock to provide the AMU write address.

Proper sizing of FIFOs and control of the LVL-1 accept are required for correct operation of the system. LVL-1 accepts are limited to 5 total events with new LVL-1 accepts occurring no sooner than 40 μs. As a result, the ALM must be able to store up to 5 events in the ADC Conversion FIFO or as many as 20 addresses, depending on the subsystem. With a maximum 40-beam-clock LVL-1 delay and 5 events in the ADC Conversion FIFO, 4 addresses are left in the Available Address FIFO. Conversely, with a minimum 32-

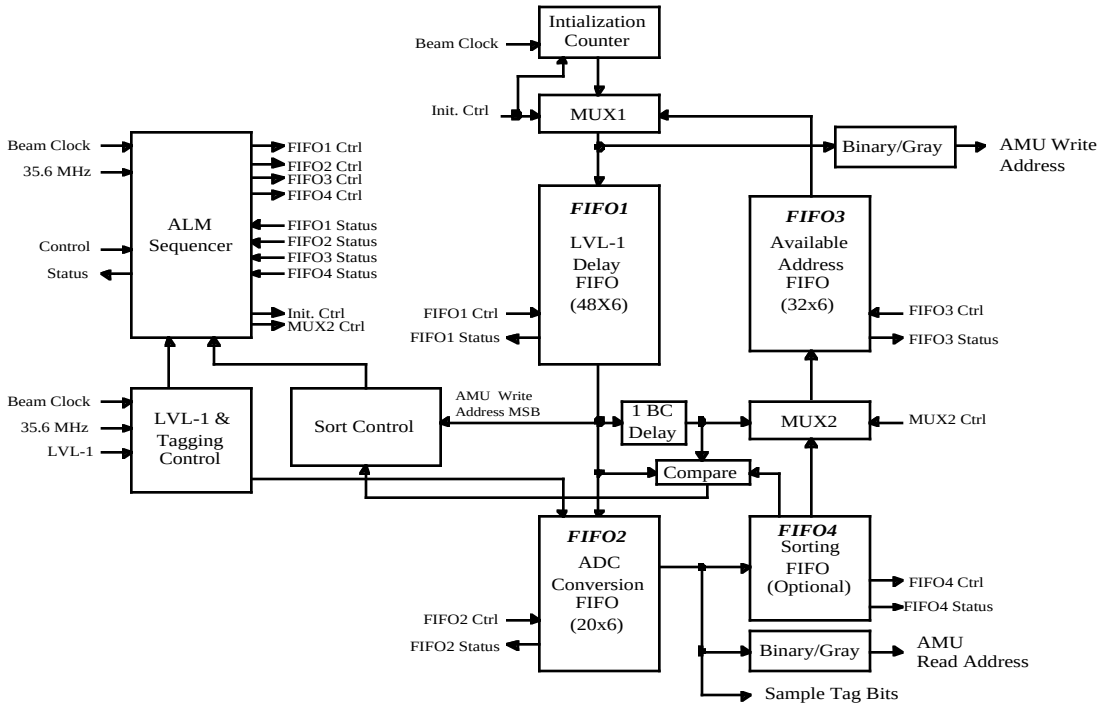

Fig. 2 Address List Manager Block Diagram

beam-clock delay and no events to be digitized, 32 addresses will be located in the Available Address FIFO. If a LVL-1 delay of > 40 beam clocks is needed, LVL-1 accepts will have to be further limited to ensure enough addresses are available to provide an AMU write address each beam clock.

Tests performed on CMOS AMU units indicate some noise benefit may be realized by using gray-coded AMU addresses. The ALM manages addresses in a binary format, and the conversion is made on both the AMU read and write addresses before being distributed to the AMU unit. Selection of the address output format (gray/binary) is a programmable function which will allow further characterization of the AMU and aid in ALM testing where binary addresses are more intuitive.

A. FPGA-Based FIFOs

As demonstrated in Fig. 2, FIFO memories play an important role in the address list manager. Due to the space limitations of all PHENIX detector subsystems, a highly integrated ALM was desired. Reconfiguration of the ALM functions make FPGA implementation more desirable than competing technologies including ASICs. SRAM-based FPGA families provide the ability to configure SRAM for use as FIFOs, SRAM blocks, and shift registers.

The base FIFO cell used in this design is shown in Fig. 3 [7]. A linear feedback shift register (LFSR) generates the address pointers for the SRAM, and a compare circuit keeps track of the relation between read and write addresses. To increase speed, the arbitration logic is configured to force a write to the SRAM block each cycle, which effectively removes one location from usable address space. The arbitration logic of the FIFO consists of two main signals, the PUSH (or write) and the POP (or read). These may be asserted simultaneously, although the PUSH has priority and will be executed first. The INPUT_RDY and OUTPUT_RDY signals may be examined to determine when data can be read from or written to the FIFO.

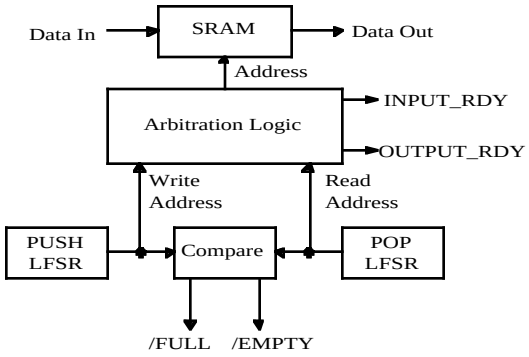

Fig. 3 Base FIFO Cell

B. Address Ordering

Re-ordering of the AMU addresses in the ALM is important to reduce the noise injected into the AMU cells from the transitioning of digital lines [8]. Referring to Fig. 2, a digital comparator is used to determine the proper placement of a binary address in the loop. Inclusion of a 1 beam-clock delay allows the comparator to look at two adjacent loop addresses and determine if the sort address fits between the

two addresses. A simple function is performed by the comparator: $addr_{LVL\text{-}1delay} > addr_{ADC} > addr_{1bcdelay}$. This algorithm is valid unless the address to be sorted is at the boundary of the addresses -- that is, it is 0 or 63, or nearer to the boundary than any other address within the available address loop. This special case is handled by monitoring the MSB of the LVL-1 Delay FIFO and a status bit to determine if there is currently an AMU address left to be sorted. If so, a compare is done to see if $addr_{ADC}$ > both $addr_{LVL\text{-}1delay}$ and $addr_{1bcdelay}$ or $addr_{ADC}$ < both $addr_{LVL\text{-}1delay}$ and $addr_{1bcdelay}$. A true condition results in a read request to the ADC conversion FIFO and an additional write request to the available address FIFO, similar to the previous case. The control logic is identical for both cases once the special cases have been handled. The most limiting beam-clock cycle is one in which one read request and two write requests are made to the available Address FIFO. Under either sorting case, this sorting procedure takes only one complete lap of the addresses.

C. Data Grouping and Tagging

LVL-1 control and data tagging are generated using the circuit shown in Fig. 4. Five serial shift registers are loaded in a rotating fashion, one for each valid LVL-1 accept, with a sampling template that identifies the number of samples and

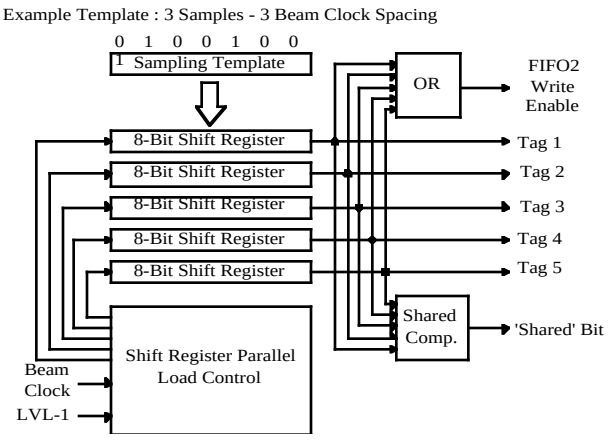

Fig. 4 LVL-1 Control and Sample Tagging

sample spacing. This provides the capability to handle as many as five overlapping or shared events. An example template given in Fig. 4 shows three samples per LVL-1 event with a spacing of three beam clocks (01001001). Every beam clock the shift registers are shifted to the right. When a '1' is output, the current AMU write address is associated with a valid event and is placed in the ADC Conversion FIFO. These output bits can also be placed in the ADC Conversion FIFO as event identifiers to help in the separation/reconstruction of data packets. This is particularly important since the data from multiple packets can be interlaced. The FIFO2 write enable is generated by performing an 'OR' on the tag bits. If multiple valid tags are generated on a single beam clock, a 'shared' bit is set and stored with the tag information. This function is significant because the contents of a single AMU cell cannot be digitized twice with sufficient accuracy. The use of this architecture utilizing sampling templates allows the data samples to be irregularly spaced.

IV. Implementation and Testing

To meet the requirements of PHENIX, all ALM and data manipulation functions must be reprogrammable in-circuit to accommodate future functional upgrades. Adhering to this requirement meant excluding ASICs from possible implementation technologies leaving programmable logic devices as the primary choice. The need to maximize integration further reduced the options to SRAM-based FPGAs that allow the incorporation of FIFO structures. Of the options available we chose the Xilinx XC4000 family.

Implementation of the ALM began with testing of 16-bit and 32-bit FIFOs. All FIFOs were incremented in sizes of 32 bits, the maximum allowed per XC4000 Xilinx configuration logic block (CLB) with individual read/write controls. The necessity of a larger 64-bit FIFO for the LVL-1 Delay FIFO requires additional logic for read/write control muxing and results in a slower speed of operation. Routed separately, individual 32-bit FIFOs easily operate at well over 40 MHz without using special routing methods.

Implementation was accomplished using schematic capture and a PC-based Viewlogic ProSeries system with XACT timing editor. The first system design using the default XACT timing specifications resulted in a maximum clock speed of 20 MHz. The design was slowed by the near-full capacity of the XC4005, which utilized 98% packed CLBs. Later designs included reassignment of the global timing and critical net timing specifications. Worst-case flip-flop to flip-flop delays (FFS_FFS) were on the order of 80 ns. Such a large delay results in little improvement in utilizing high speed parts since the 1-2 ns CLB delay improvement available becomes small in comparison to the routing delays. The current design was customized for the mid-speed XC4005-5 and reached clock speeds of 45 MHz. The identical LCA configuration was loaded into XC4005-4 and XC4005-6 parts resulting in clock speeds of approximately 40 MHz. Further synthesis effort in assigning critical nodes and time specifications should result in operating speeds on the order of 45 MHz for these devices also.

A timing diagram of the ALM is shown in Fig. 5. Besides general operation, this diagram demonstrates both normal operation and the special operations associated with address reordering. Here addresses 40, 42, and 63 have been removed from the list and are stored in the ADC Conversion FIFO. The lvl1_out and amu_wr_addr buses show absence of these addresses (40, 42, 63) that are awaiting proper re-

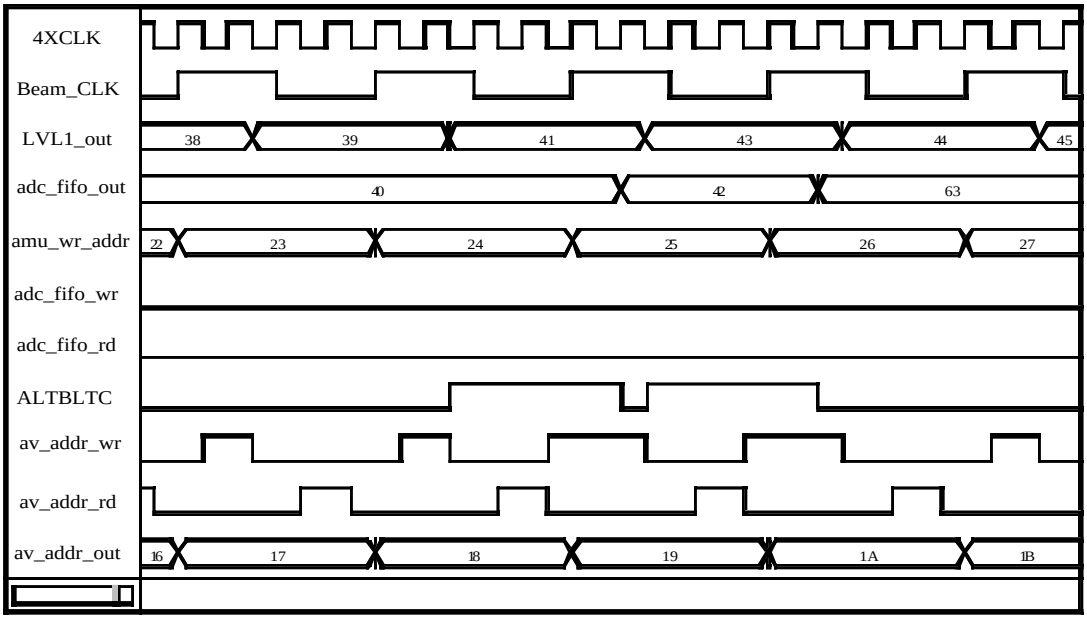

Figure 5. ALM Timing Diagram Demonstrating General Operation and Address Re-Ordering

sorting conditions. When a sorting condition is met, the adc_fifo_out address falls between the current lvl1_out and the delayed lvl1_out, and an additional write to the Available Address FIFO occurs. A read of the ADC Conversion FIFO also occurs if the FIFO is not empty. The Available Address FIFO is always read, guaranteeing a valid AMU write address each beam clock. Notice that two worst-case conditions occur on adjacent beam-clock cycles -- two writes and a read to the Available Address FIFO within one beam-clock cycle.

V. CONCLUSIONS

A programmable analog memory address list manager has been developed for generic application to all analog memory-based PHENIX detector subsystems. The unit allows programming of the LVL-1 delay latency, number of data samples per valid LVL-1 trigger, and sample spacing. Sampling parameters are programmed as a template which allows irregularly spaced samples. Cell write-over protection, address re-ordering and sample tagging are also implemented. The design was synthesized in an FPGA and operated at 45 Mhz (11.25 Mhz beam clock). Power consumption was measured as 575 mW at PHENIX beam clock rates of 9.52 MHz.

VI. REFERENCES

[1] F. Anghinolfi, et al., "Characteristics of a 'Harp' Signal Processor with Analog Memory Operated with Segmented Silicon Detectors", *IEEE Trans. Nucl. Sci*., Vol. 41, No. 4, August 1994, pp. 1130-1134.

[2] F. Anghinolfi, et al., "DYN1: a 66 MHz front end analog memory chip with first level trigger capture for use in future high luminosity particle physics experiments", *Nuclear Instruments & Methods in Physics Research Section A*, Vol. 344, 1994, pp. 173-179.

[3] 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.

[4] S. Kleinfelder, M. Levi, and O. Milgrome, "Toward a 62.5 Mhz Analog Virtual Pipeline Integrated Data Acquisition System, *Nucl. Phys.* B23A (1991) pp. 382-388.

[5] 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.

[6] O. B. Milgrome, S. A. Kleinfelder, and M. E. Levi, "A 12-Bit Analog to Digital Converter for VLSI Applications in Nuclear Science", *IEEE Trans. Nucl. Sci.*, Vol. 39, No. 4, 1992.

[7] Xilinx Application Note XAPP 044.000, "High-Performance RAM-Based FIFO", The Programmable Logic Data Book, 1994.

[8] C. L. Britton, et al., "An Analog Random Access Memory in the AVLSI-RA Process for an Interpolating Pad Chamber", to be published in *IEEE Trans. Nucl. Sci.*, 1995.