FPHX: a New Silicon Strip Readout Chip for the PHENIX Experiment at RHIC

James R. Hoff, *Member, IEEE,* Tom N. Zimmerman, Raymond J. Yarema, *Life Member, IEEE,* Jon S. Kapustinsky, and Melynda L. Brookes

***Abstract*–The FPHX chip is a silicon strip readout chip developed at Fermilab for use in the FVTX Detector of the PHENIX experiment at RHIC. Each front end consists of an integrator which is AC coupled to a shaper, followed by a discriminator and a 3-bit analog-to-digital converter. The backend is a novel architecture in two stages that permits dead-timeless operation and high-speed readout with very low latency. A slow controller provides an interface for all on-chip programmable functions. This chip has been fabricated in the 0.25um TSMC process. All functionality including the analog front-end, the digital back-end, and the slow controller has been verified experimentally.**

I. Introduction

THE Forward Silicon Vertex Tracker (FVTX) is a silicon strip detector being developed as an upgrade to the PHENIX experiment at RHIC[1]. Its purpose is to improve secondary vertex tracking and thereby, through a distance of closest approach (DCA) technique, develop cuts to separate prompt muons and long-decay particles from short-lived heavy quark mesons. The detector consists of four planes of strip panels at low incidence angle to the beam line and at both forward and backward rapidity. Each panel is formed from 48 wedges and each wedge contains two columns of parallel strips. The strip pitch is 75 um. Each strip is normal to a radius drawn outward from the beamline and each strip subtends an arc of 3.75 degrees regardless of its distance from the beamline. The wedge design is shown in Fig. 1.

The FPHX is a custom integrated circuit designed at Fermilab as a read out chip for the FVTX. The unique requirements of the FVTX make this chip essential. The mechanical structure of the wedge requires the front end amplifier to be capable of handling strips that range from 3.45mm long at the smallest radius to 11.2mm long at the outer radius. This translates to input capacitances that vary from approximately 0.5pF to 2.0pF. Moreover, the mechanical structure of the wedge implies that there is very little room on the High Density Interconnect (HDI) for parallel buses, bias lines, and support circuitry. Therefore, self-biasing and data serialization are fundamental to the design. To achieve the required DCA resolution of better than 200 um [2] a three bit Flash ADC is necessary per channel. Finally, the

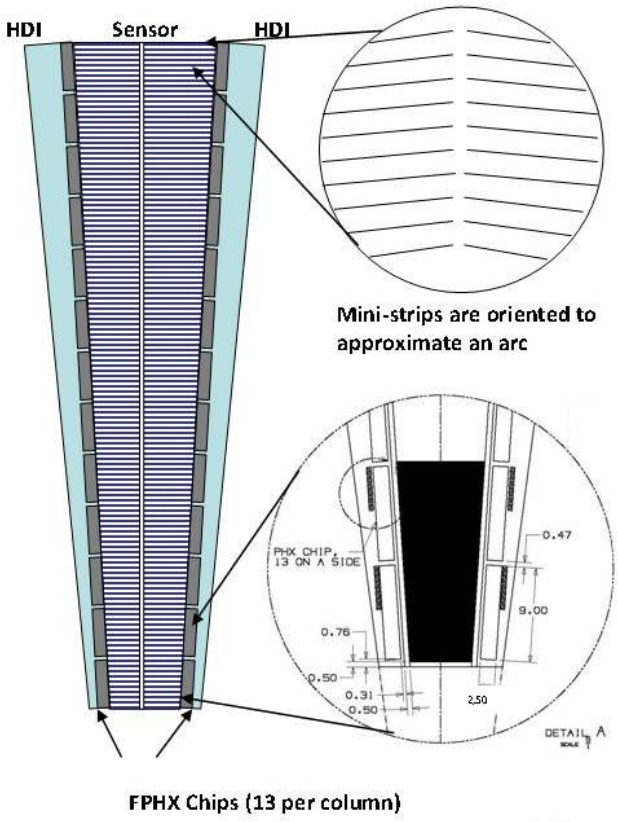

Fig. 1: The FVTX Web Structure.

desire to have the FVTX capable of participating in the Level 1 trigger means that readout must be deadtime-less, there is little tolerance for latency, and the architecture must be data push. The highest occupancy events in Au+Au collisions, which have an average of four hits per chip, must be read out in four clock cycles or less. At the same time, larger events are assumed to be significant and the FPHX must be able to deal with them and their long read out times, even if that means that the data cannot be made available in time for the Level 1 trigger.

Section II will discuss the analog front end. Section III will cover the digital architecture, including the Slow Control programming interface. Finally, Section IV will talk about fabrication and test results.

Manuscript received November 12, 2009. This work was supported in part by the U.S. Department of Energy under Contract No. DE-AC02-07CH11359.

J. R. Hoff, T. N. Zimmerman and R. J. Yarema are with the Fermi National Accelerator Laboratory, Batavia, IL 60187 USA (telephone: 630-840-2398, e-mail: jimhoff @fnal.gov).

J. S. Kapustinsky and M. L. Brookes, are with Los Alamos National Laboratory, Los Alamos, NM 87545 USA.

II. Front End

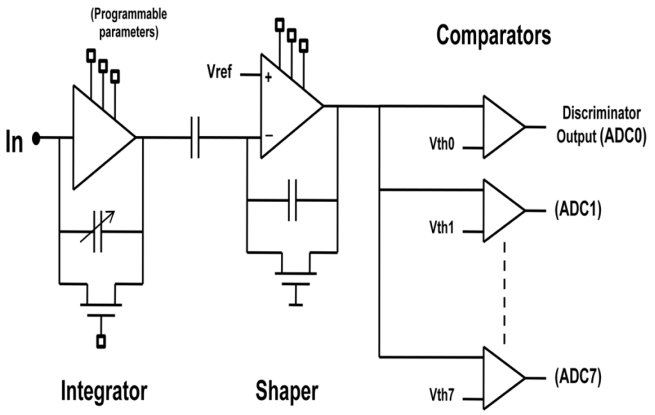

Fig. 2: The FPHX front end (one channel)

The analog front end portion of the FPHX integrates and shapes the detector signals. In each of 128 identical channels, the shaped signal feeds a threshold discriminator and a simple 3-bit flash ADC. A simplified block diagram is shown in Fig. 2. The system is designed and optimized for unipolar (positive) input signals and performs CR-RC shaping. When the integrator receives a charge input impulse, it produces a fast (relative to the shaper peaking time) output step followed by a programmable discharge time, giving the "CR" portion of the CR-RC response. The integrator bias level, gain, and bandwidth are programmable in order to allow optimization of noise and response time for a variety of input capacitance values. The shaper has a fixed voltage gain (of approximately 5) and limits the signal bandwidth, which determines the signal rise time and therefore the peaking time. The shaping time can be adjusted via the programmable shaper bias.

The shaper DC baseline is set with a programmable reference voltage (Vref), and the shaper output pulse feeds eight programmable-threshold comparators. The comparator thresholds are referenced to the shaper baseline Vref. The first comparator is used as a hit discriminator. The other seven comparators, along with the hit discriminator, form a simple thermometer-code flash ADC.

III. Back End

The PHENIX experiment has a 106ns event period and hit occupancy is expected to be relatively low. However, as stated previously, the FVTX would like to participate in the Level 1 trigger, so the FPHX must be deadtime-less, it must be data push, and it must have low latency. Space on the HDI is very limited so data sparsification is essential and data serialization is a requirement. Typical events generate four hits or less per chip, but they must be read out in four clock cycles or less. Noise hits will add to the required bandwidth, but they cannot be permitted to adversely affect the flow of events. Finally, large events are assumed significant, so the architecture has to be flexible enough to deal with them.

A. Output Word

Fig. 3 shows the output word produced by the FPHX. The seven bits of address information (Location[6:0]) are required

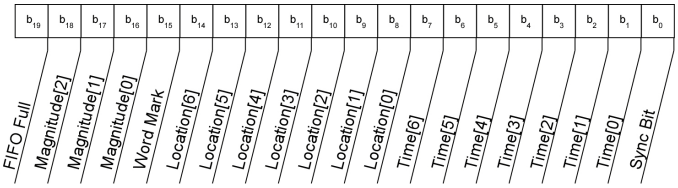

Fig. 3: The Output Word

by the fact that there are 128 channels per FPHX chip. The three bits of ADC information (Magnitude[2:0]) and the seven bits of time information (Time[6:0]) are required by the FVTX system. No chip ID information is required because all communication between a FPHX chip and its DAQ logic is point-to-point, so the chip address is hard-wired. The Sync Bit and the Word Mark result from the need for synchronization on the serial lines. When there is no data to output, sync words are output by the chip for the obvious purpose of data synchronization with downstream logic. Sync words are 19 zeros followed by a 1 (the Sync Bit) and no data word can be permitted to be identical to a sync word. Therefore, during data transmission, the Word Mark Bit is set to a Logical 1 so that a zero magnitude hit on location zero at time zero is not mistaken for a Sync Word. Finally, FIFO full is a status bit sent from an FPHX chip to alert its DAQ logic to the possibility that data is being lost inside the chip.

In short, this 20-bit word is the smallest possible output word that satisfies the FVTX system requirements. Given the prerequisite of four hits in four clock periods, the absolute minimum serial bit frequency for one serial line would be 80 bits of data in 424ns or approximately 189 MHz.

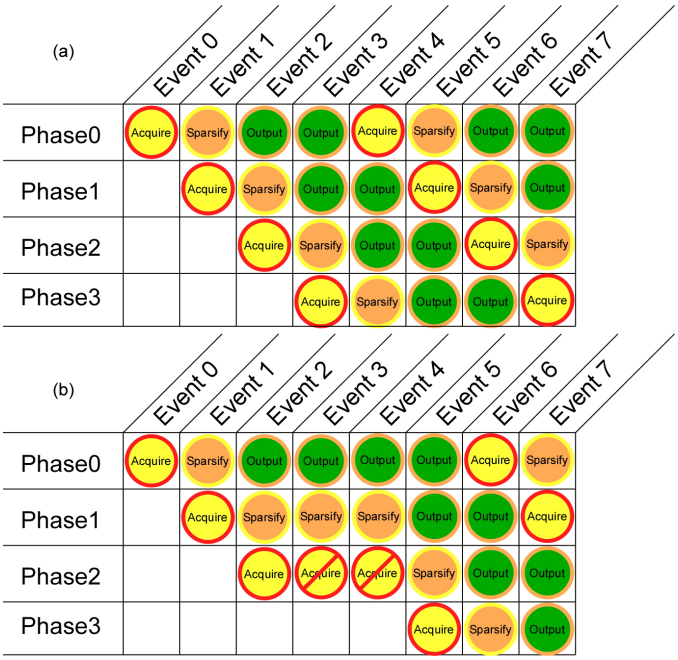

Fig. 4: Four-phase Operation. (a) shows typical event processing. (b) shows the processing of a large event (Event 1 in Phase 1) that forces the blocking of data acquisition during Events 3 and 4. Event 2 is still cleanly and accurately acquired by the chip as is Event 5.

Unfortunately, this leaves no time for data acquisition, which, by definition, takes one full event period, or for sparsification, which takes a variable amount of time depending on the number of hits. Extremely high serial bit rates would leave more time for sparsification, but they could also pollute the HDI with high-frequency noise that could couple into the analog front ends. Slower serial bit rates would require more parallel data lines to meet the required data rate. However, space on the HDI is very limited. In the end, a compromise was made and two serial lines operating at a serial bit frequency of 189 MHz was selected. This means that for a typical event of four hit channels that must be read out in four clock periods, one period is reserved for hit acquisition and two periods are reserved for data transmission, leaving only one period for sparsification. Obviously, since the system must be deadtime-less, hit acquisition must happen while sparsification and serialization of previous events is ongoing. Less obvious are the facts that

1. A channel hit in one event period must immediately be available to begin its output in the next period, and
2. Sparsification of the entire 128 channel array must be fast enough so that the sparsification of one event can be followed immediately in the next event period by the sparsification of the next event.

These requirements lead to the idea of a four-phase, four datapath architecture.

B. Four-phase/Four-datapath Architecture

N-phase/N-datapath Architecture is a well-established technique for achieving high-rate, deadtime-less operation [3]. In a single datapath architecture, data is acquired by the channel and then processed. Typically, while data is being processed, no new data can be acquired. In a multi-datapath architecture, data from one event is acquired by one datapath and then processed in that datapath. However, while data is being processed in one datapath, new data from another event can be acquired in another datapath. This is shown in Fig. 4(a).

It is obvious from the figure that Output Mode processing can pile-up. For example, during Event 3, both the Phase 0 and Phase 1 logic are in Output Mode. This is not a problem, however, because, as stated previously, activity is relatively rare, and most events have less than four hit channels per chip. However, this does illustrate the need for a FIFO between the Sparsification Mode and the Output Mode to mitigate any data pile-up that might occur.

What makes the Four-phase/Four-datapath architecture of FPHX somewhat unique is the Sparsification Mode data processing. Events can be almost arbitrarily large and sparsification cannot be made arbitrarily fast. Instead, the capacity to continue sparsification past the typical single event period limit must be built into the system. This is illustrated in Fig. 4(b). Event 1 is a large event. The Phase 1 datapath begins to sparsify it during Event 2 but the sparsification is not

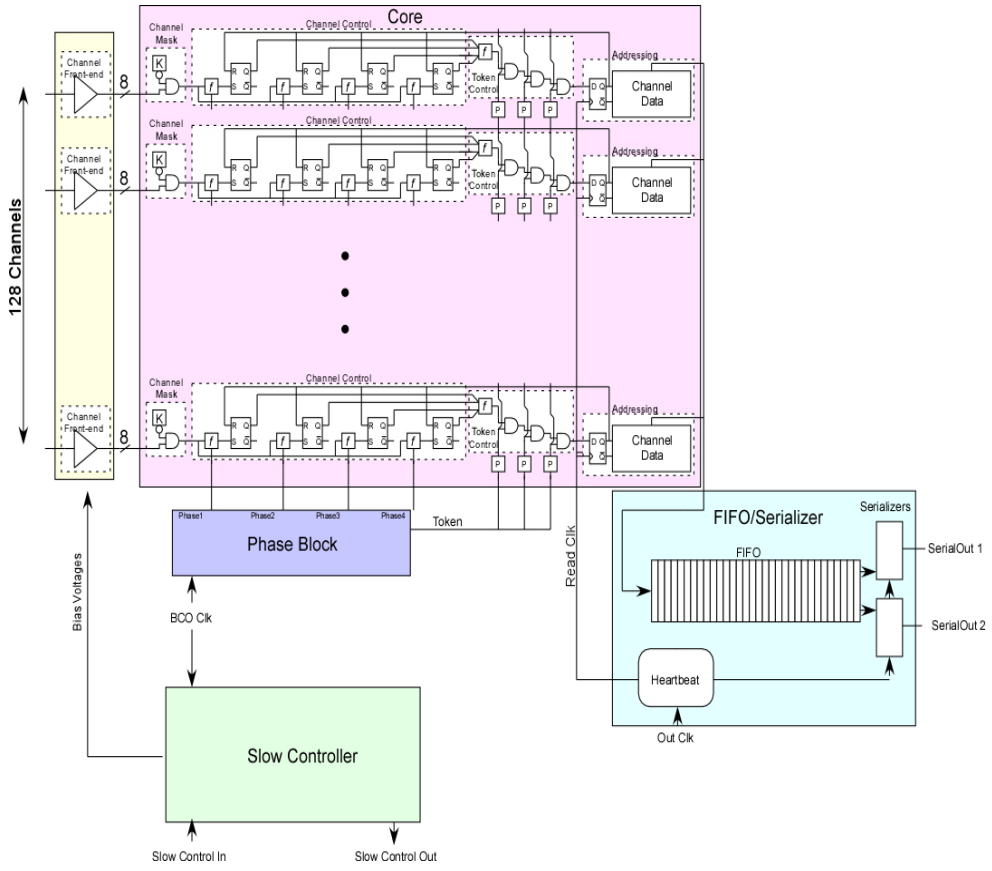

Fig. 5: The Two-Stage Architecture and Support Logic

complete by the end of the Event 2 period. Hit channels from Event 2 have been properly acquired by the Phase 2 logic, but the datapath cannot be switched until Event 1 is fully sparsified. If permitted, any hit channels acquired during Event 3 would be indistinguishable from those properly acquired in Event 2. Therefore, the Phase Change Logic must block data acquisition whenever extended sparsification interrupts datapath switching.

C. Two-Stage Readout Architecture

Fig. 5 shows a block diagram of the final architecture of the FPHX readout. The first stage, labeled "Core" is the four-phase/four-datapath design that handles Data Acquisition Mode and Sparsification Mode. The Phase Block uses an Event Clock matched to the beam cross-over period of the PHENIX experiment. It provides the four phase switching signals as well as the four-phased token. The Phase Block also contains the Phase Change Logic that senses the completion of the Sparsification Mode and determines if the Sparsification Mode needs to be extended and, consequently, if datapath switching and data acquisition should be allowed or blocked. The second stage, labeled "FIFO/Serializer" contains a 32-hit deep FIFO for mitigating data pile-up. This FIFO feeds two serializers that are each capable of outputting one 20-bit word per event period. The serializer uses an Output Clock that is frequency matched to twenty times the Event Clock period.

The second stage also contains the Heartbeat logic that uses the Event Clock and the Output clock to produce the Read Clock and the Grab Clock that govern the transfer of data from the first stage to the second stage. Basically, the Read Clock and the Grab Clock each pulse three or six times per Event Clock period depending on whether the user has programmed the FPHX to use one or both serializers. At the rising edge of each Read Clock, the data of one hit channel is output from Stage 1 to Stage 2. At the rising edge of each Grab Clock, that data is latched into the FIFO. Sparsification Mode is finished when there are no more channels to read out on the rising edge of a Grab Clock.

D. Token Passing

The FVTX is such that there is a near uniform distribution of hit probability across any given FPHX chip and, because of noise, there can be no assumption of clustering. Therefore, the first hit channel can be anywhere in the 128 channel array. The last hit channel can be anywhere in the array. Finally, the distance between any two hit channels can be as little as zero to as many as 126 channels. Therefore, the logic used for sparsification must be extremely fast. The FPHX uses the three-tiered token-passing scheme shown in Fig. 6.

The 128 channel array in the Core of Fig. 5 can be logically divided into four Banks of 32 channels each. Each Bank of 32 can be further divided into four Blocks of 8 channels each. The signal that marks the transition from Acquire Mode to Sparsification Mode becomes the token for a particular datapath. That signal is split three ways. It is driven to the bottom of the lowest Bank of 32 as the Tier 3 token. It is driven to the bottom of the lowest Block of 8 in each Bank of 32 as the Tier 2 Token and it is driven to the bottom of each

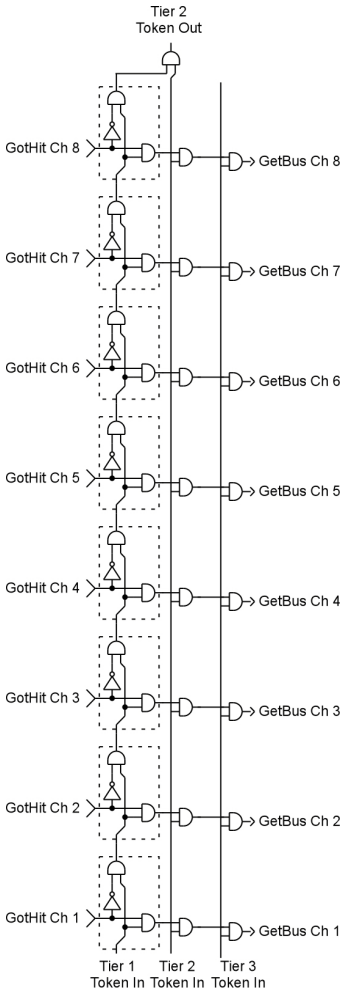

Fig. 6: An 8-channel Block of the three-tiered token passing scheme

Block of 8 in the entire array as the Tier 1 token. Each of these tokens is allowed to drift up until it encounters the first hit Channel (Tier 1), Block (Tier 2), or Bank (Tier 3). If there are no hit channels in a Block of 8, that Tier 1 token combines with the Tier 2 token and is passed to the next Block of 8 as its Tier 2 token (See the top-most AND gate in Fig. 6). If there are no hit channels in a Bank of 32, that Tier 2 token combines with the Tier 3 token and is passed to the next Bank of 32 as its Tier 3 token (with logic identical to the top-most AND gate in Fig. 6). When a particular channel has all three tokens, then on the next rising edge of the Read Clock, it outputs its data onto the Stage 1 to Stage 2 bus.

The resultant token passing scheme is capable of jumping across any possible channel combination in less than 5ns worst case.

E. Slow Controller

A Slow Control interface was designed for the FHPX for use in programming its various features. It has the following properties and capabilities.

- 3-wire interface: Shift In, Shift Out, and Shift Clock.
- 32-bit Slow Controller Word (see Fig. 7).
- Each register assumed to have 8-bits.

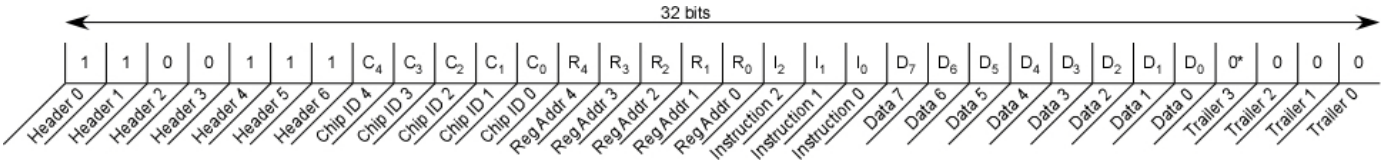

Fig. 7: The Slow Controller Word

- 31 unique chip IDs and one Wild Card (via 5 chip ID bits in the Slow Controller Word)
- 31 unique register IDs and one Wild Card (via 5 register address (regAddr) bits in the Slow Controller Word).
- Different Commands can be sent to each register. Any register can be Set (all bits to 1), Reset (all bits to 0), forced to its Default value, or Written to (commands come via 3 Instruction bits and data comes via the 8 Data bits).
- Each command reads out the current register value through the Shift Out line
- Multiple control words can be sent one after another

In the FPHX, the Event Clock is used as the Shift Clock. The chip IDs are set through pads on the periphery. The "wild cards" allow a user to send commands to all chips and/or all registers simultaneously.

IV. Fabrication and Test Results

The FPHX was first submitted in June of 2008 to the MOSIS service. It was fabricated in the TSMC 0.25 um CMOS mixed-signal process (CM025). This first submission was a fully operational chip including front end, back end, programming interface and LVDS drivers and receivers. Input transistors of different width-to-length ratios were placed in different channels to permit experimental verification of the optimal length and width. The chip was fully functional and only two errors were reported – a slightly larger than expected discriminator threshold voltage distribution and an error on the back end that improperly activated the Phase Change logic, causing hit channels to report an erroneous time stamp under limited circumstances.

A second prototype was submitted in May of 2009. Like the first, it was fabricated in the TSMC 0.25 um CMOS mixed-signal process (CM025) and like the first it was a fully functional chip. The final input transistor configuration was chosen from test results on the first prototype. The threshold dispersion was tightened through a layout modification and the Phase Change logic was corrected.

Testing of the two prototypes has been completed and the second prototype is considered error free. A brief overview of the test results is as follows:

- Integrator and shaper responses are according to specifications
- gain verified 50, 66, 100 or 200 mV/fC
- dynamic range > 800 mV (non-linear from ~800mV to 1V)
- nominal peaking time 60 ns
- noise on single chip (no sensor) meets specs ~200 electrons + ~ 70 electrons/pF
- no measureable crosstalk to neighbor channels with large signal input
- all programmable bit functions were verified
- FPHX operates in a predictable and stable mode over a wide range of setup parameters
- Power ~ 140 uW/ch analog plus ~ 300 uW/ch digital
- Comparator response linear to ~ 1%
- Digital Backend performs as designed in all respects.

An engineering run has been submitted to MOSIS which should provide the experiment with all required chips plus spares. FPHX Wafers will be tested and FVTX wedges will be fabricated at Fermilab in 2010.

Acknowledgment

The authors would like to thank Sergey Butsyk of the University of New Mexico, and David Winter and Eric Mannel of Nevis Laboratories and Columbia University for endless testing and Albert Dyer of Fermilab for equally endless hours of wire bonding.

References

[1] A. K. Purwar, "DAQ for Forward Silicon Tracker Upgrade for PHENIX", 15th IEEE-NPSS Real-Time Conference, April 29, 2007-May 4, 2007

[2] W. A. Zajc, *et al*, "PROPOSAL for a Forward Silicon Vertex Tracker (FVTX) for the PHENIX Experiment", available online at http://www.phenix.bnl.gov/WWW/fvtx/home/Documents/fvtx-proposal_11-1-07.doc

[3] R. J. Yarema, G. W. Foster, J. Hoff, M. Sarraj, T. Zimmerman, "A Fast, Wide Range Charge Integrator and Encoder ASIC for Photomultiplier Tubes", *IEEE Trans. Nucl. Sci.*, vol. 40, no. 4, pp. 750-752, Aug. 1993.