FERMILAB-PUB-07-118-CD

NO*v*A DAQ, System Architecture, Data Combiner and Timing System

R. Kwarciany*, K. Biery*, G. Cooper*, S. Foulkes*, G. Guglielmo*,
B. Haynes*, V. Pavlicek*, L. Piccoli*, M. Votava*

*Fermilab, United States Department of Energy, Computing Division,
Kirk and Wilson Streets, Batavia, IL 60510, USA

***Abstract* - NOvA (E929) is a long baseline experiment that will search for neutrino oscillations. There will be one detector near the beam source at Fermilab, and one detector in northern Minnesota. The DAQ system for the far detector collects over-threshold hits from over 450,000 channels of scintillator readouts, sorts the time-stamped data packets and archives selected time periods of data for transmission and processing. While a simple point-to-point protocol is used for the first level of data collection, Ethernet was chosen as the fabric for the rest of the DAQ. The packet time-stamp and overall system synchronization is based on two common-view GPS trained clock oscillators, one at each site. The present design cost-effectively satisfies the experiment's moderate speed and data volume requirements.**

I. Introduction

Members of the NO*v*A Collaboration have created a preliminary design for a new experiment to study neutrino oscillations using the existing neutrino beam at Fermilab. The main goal of the experiment is to measure the probability for muon neutrino to electron neutrino oscillations $(\nu_\mu \rightarrow \nu_e)$ down to a value ten times more precise than the existing experimental limit. To do this, NO*v*A proposes to build and install two particle detectors optimized for identification of electron neutrino (ν_e) interactions. The "Near" detector is a 210 ton liquid scintillator detector placed in the existing NuMI tunnel at Fermilab. The "Far" detector is an 18 kiloton liquid scintillator detector placed 810 km North-Northwest of Fermilab in Ash River, Minnesota.

Neutrino interactions in the detectors are detected as light pulses in the scintillator. The NOvA detectors, both far and near, consist of scintillation particles suspended in mineral oil within a PVC tube. An wavelength shifting optical fiber is routed inside the PVC tube to collect the light, and an Avalanche Photo Diode (APD) attached to the fiber converts the light pulse into electrical signals. The Front End Board (FEB) will discriminate and time tag the signals. A timing system will synchronize and provide the time base to all the DAQ electronics modules. FEBs will transmit the over-threshold signals to the Data Combiner Module (DCM). The DCM will collect data from up to 64 FEBs and transmit packets of data onto the DAQ Ethernet network. A small farm of computers will buffer the data, build requested events based on the time tags and archive that data. The selected data will be transmitted to the data processing center at Fermilab.

II. DAQ Architecture

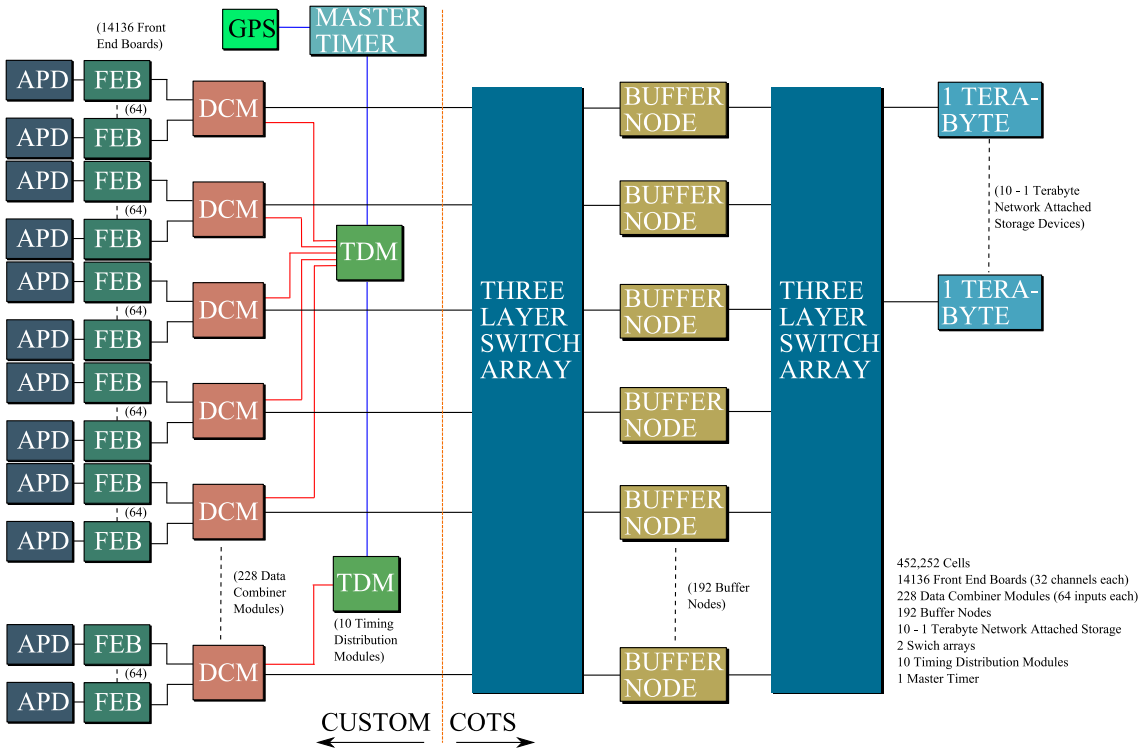

Fig. 1. NOvA Data Acquisition System

The signal from each channel of the NOvA detector consists of the digitized output of an APD. 32 channel APD arrays are each connected to an FEB, where the data is digitized, and transmitted to the Data Acquisition System at a maximum design data rate of 1 Mbps. The NOvA DAQ system collects this data from FEBs arrayed on the detectors and stores complete events into mass storage sorted by timestamp. To minimize cost and development time, it was deemed desirable to use commercial Ethernet hardware for the bulk of the DAQ system. The DCM is a custom module that has been designed to collect data from 64 FEB modules using point-to-point data links. The DCM will concatenate and packetize the data, then output it over Ethernet at a 1 Gbps rate. A three tier array of commercial gigabit Ethernet switches then routes data for events to a farm of commercial CPU nodes (Buffer Nodes) that buffer data and build the events. Complete events are passed to mass storage through a second switch array.

All modules on the detector are kept time synchronized by the NOvA Timing and Control System (TCS). A master clock module at each detector receives time information from a precision GPS receiver. Beam spill timing information is transmitted from Fermilab to the far detector in Minnesota via public internet. Precision synchronization of the far detector with the near detector is possible due to the accuracy of the GPS based clock system, and the availability of the spill timing information received from Fermilab. The master clock module distributes timing and control information to all DCMs on the detectors via Timing Distribution Boxes (TDB). TDBs and DCMs implement timing learning capability to compensate for cable length differences, allowing the entire system to be accurately synchronized.

III. Data Combiner Module

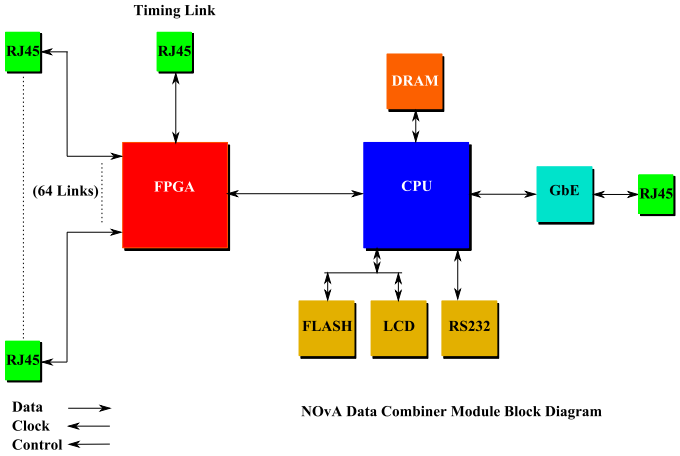

Fig. 2. Data Combiner Module Block Diagram

Data Combiner Modules are detector mounted stand-alone intelligent devices with an embedded CPU running Linux. The CPU pulls data from a buffer implemented in an FPGA. This buffer is continuously filled with serial data collected from up to 64 FEB modules via one of the pairs of wires in the CAT5e front end link cable. The link deserializers, first stage buffers, and data concatenation logic are also implemented in the FPGA. DCMs receive timing and control information from the NOvA Timing Distribution System over a dedicated link. This information is transmitted to the FEBs over the remaining three pairs of wires in the cable. DCMs are connected to the Gigabit Ethernet DAQ network with copper CAT5e links, and are positioned on the detector as necessary to keep the FEB data cable lengths manageable.

At boot time, the flash based bootloader downloads the DCM Linux image, and then boots Linux. A Linux boot script then downloads the configuration file for the FPGA device and configures the FPGA. Once the FPGA is configured, the DAQ application software and DCM configuration information is downloaded. The main DAQ application running on the processor is responsible for packetizing and buffering the data until it can be output over the DAQ network to a buffer node CPU. Included in the DCM configuration data will be a list of buffer nodes and information defining how data is distributed to the nodes. In general, the DCMs will send event data to each buffer node in a round-robin rotation, allowing for efficient use of the DAQ network, its switches, and the buffer nodes themselves.

Since DCMs will be distributed about the detectors, they will be packaged in 3U rack mountable enclosures, and will be mounted adjacent to the power distribution nodes. This ensures that cables to the FEBs will be reasonably short, and can be routed near the power cables. FEB, Timing Link, and Gigabit Ethernet connectors on the DCM are all RJ-45 jacks, and CAT5e cables with RJ-45 plugs are used for all non-power connections.

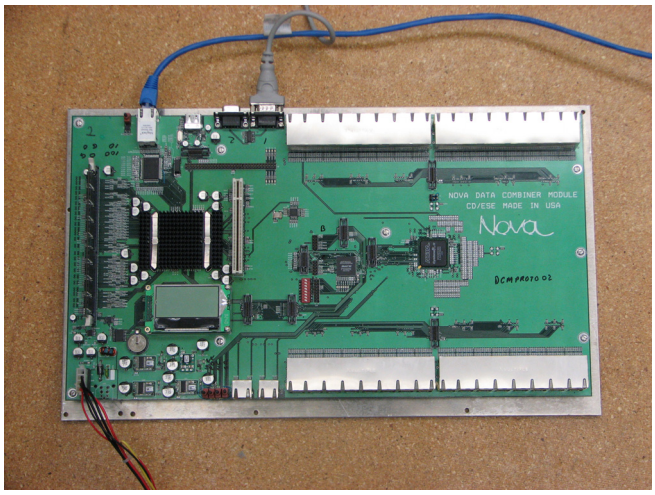

Fig. 3. Data Combiner Module Prototype

IV. Data Acquisition Network

The DAQ network is based on commercially available Gigabit Ethernet hardware. The largest generally available commercial switch modules available at the time of purchase will be used to build a three layered switch array to allow all 228 DCMs to communicate with any of the 192 Buffer Nodes as if the array was actually one large switch module.

To minimize blocking at the switch, DCMs will be programmed to send data to different Buffer Nodes on a rotating basis. At system startup, each DCM will be configured with a list of buffer nodes, and the order in which to write to them. Each DCM will also be assigned a different starting buffer node such that when all DCMs begin writing data, they will all be writing to different Buffer Nodes, allowing for maximum data transmission efficiency through the switch array.

V. Buffer Nodes

The DAQ system will contain a large collection, close to 200 nodes, of compute nodes running Linux, which will serve as a memory buffer, or buffer farm, for potential event data. Data will be held in memory on these nodes while awaiting a trigger decision from the global trigger system. The primary data trigger will be composed of a delayed signal from the Fermilab site, 810 kilometers away, that is translated into a 30 usec time window bracketing the 10 usec live beam time. Because the offset between the actual beam arrival and the arrival of the beam trigger (from Fermilab) is variable due to internet uncertainties, the farm is designed to buffer data for a minimum of 20 seconds. The nodes in the buffer farm are designed to be one large circular buffer. Data on each DCM is accumulated for several milliseconds and then routed over TCP/IP to a node in the buffer farm. Data from all DCMs collected during that same time period will also be sent to the same node, although the actual sending of the data may be slightly delayed between different DCMs to avoid high data rates into a single node. Data from the next few milliseconds will be sent to the next buffer node in the list in the same manner, cycling through all of the buffer nodes in a circular pattern. In addition to buffering the data, the buffer farm also allows for monitoring algorithms to run over the buffered data. Upon receiving a trigger window the buffer farm nodes will search their data buffers for matching data and route it to a data logger process on a separate node. The data buffers inside each buffer farm node are treated as circular memory.

VI. Data Storage

Ten terabyte Network Attached Storage (NAS) devices will be available to the data logger nodes through a dedicated network switch array. This switch array will be based on commercial Gigabit network hardware and will also provide a connection to Run Control for all modules in the system. The design data rate for the experiment is only 370 Gigabytes per year, but there could be as much as 36 Terabytes per year of calibration data.

VII. System Software

The DAQ system software provides the user interface to the DAQ system as well as the infrastructure for integrating the various hardware subsystems into a unified operational system. There are three major functional areas in the DAQ software: data handling and selection; DAQ system control; and quality and performance monitoring. A simplified diagram of how these functional areas are connected through the flow of information is shown in Fig. 4.

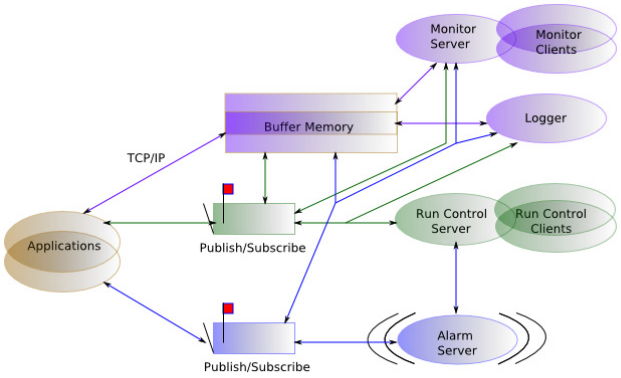

Fig. 4. Overview of information flow in the system software. Purple lines represent data flow over TCP/IP. Green lines represent control flow and blue alarm flow in a publish/subscribe model using the Nova DAQ Messaging System.

At the heart of the system are the data handling functions. There are several distinct functions related to data handling. The first is to provide routing of data between various stages of the system, which is done with client/server connections over TCP/IP. The second is to provide a large memory buffer pool for data waiting for a trigger decision. This is accomplished through processes which maintain time ordered data in a circular memory buffer on each of the buffer nodes. The buffering is necessary because the beam trigger may be delayed for multiple seconds as it travels from Fermilab to the far detector site. To aid in finding the data within a beam trigger window based on timestamps in the large buffer (hundreds of MB), a lookup table of coarse timestamps are used to quickly seek to the buffer region where data with matching timestamps is stored. Finally the data handling functions provide aggregation and writing of data to files for later long term storage in the mass storage system.

Control of the DAQ system is provided through control messages sent from the Run Control applications and the rest of the DAQ system. The control messages will be provided by the Nova DAQ Messaging System. This messaging system provides a publish/subscribe API and abstraction for the choice of underlying transport mechanism. The goal of the abstraction is to allow replacement of the underlying transport mechanism without impacting application level code. Initially the messaging system will use EPICS with patches to change its delivery model to avoid dropped messages in a high rate environment [3]. The Run Control processes provide the user control interface for the Data Acquisition system and will allow partitioning the hardware into parallel separate DAQ systems, each of which can be controlled by a separate Run Control client process. Partitioning of the DAQ hardware allows part of the detector to be commissioned or tested in parallel with production running. The core of Run Control will provide centralized services for multiple user interfaces and manage the DAQ hardware resources for the users.

Data quality and performance monitoring insures the proper functioning of the system and the integrity and quality of the data collected. The system is composed of automated and interactive processes for monitoring. On the automated side, processes that detect a problem can raise an alarm which will be processed by the alarm server. As with control messages, the alarm messaging infrastructure is provided by the Nova DAQ Messaging System. The alarm server has the following primary functions: log alarms; serve alarm messages to interested processes; filter alarms and route selected alarms to the Run Control system. The alarm server will also provide an interface to the hardware alarm and monitoring system, which will be EPICS based. There are two types of monitoring processes. The first monitors data quality by processing a subset of the data. Depending on the specific monitoring process, results will be available interactively in a user display. The second type of monitoring process evaluates the performance of the DAQ system. Information concerning data rates, errors and other performance metrics at various points in the system is collected and sent to a monitor server. The monitor server will use a database to provide persistent buffering of monitoring information. This approach shields the primary data handling processes from user monitoring requests and simplifies the API for user code by providing one connection and one format for monitor information.

VIII. DAQ Timing System

The Timing and Command Distribution System provides the timing infrastructure and distributes commands to the entire detector system from the run control computer. Synchronization of the detector timing system with the neutrino beam production at Fermilab is based on two common view GPS trained oscillators. The timing commands (or run control commands) are distributed on a bidirectional high speed serial link along the length of the detector with a far-end serial loopback. This loopback permits the timing system to be designed to self-compensate for propagation delays. Removal of the propagation delays insures simultaneous timing command arrival at each DCM on the detector, and with an accuracy of better than one clock cycle (163.84 MHz). The Master Timer Box (MTB) contains a Global Positioning System (GPS) trained clock generator (163.84 MHz) to synchronize the timing system at Fermilab and the far detector in Northern Minnesota. The GPS trained clock is the time-base for generating the timing commands at a fixed repetition rate around the serial loop shown in Fig. 5. All non-timing controls commands to the system, usually reads & writes, are queued and interlaced between the timing commands at a lower priority.

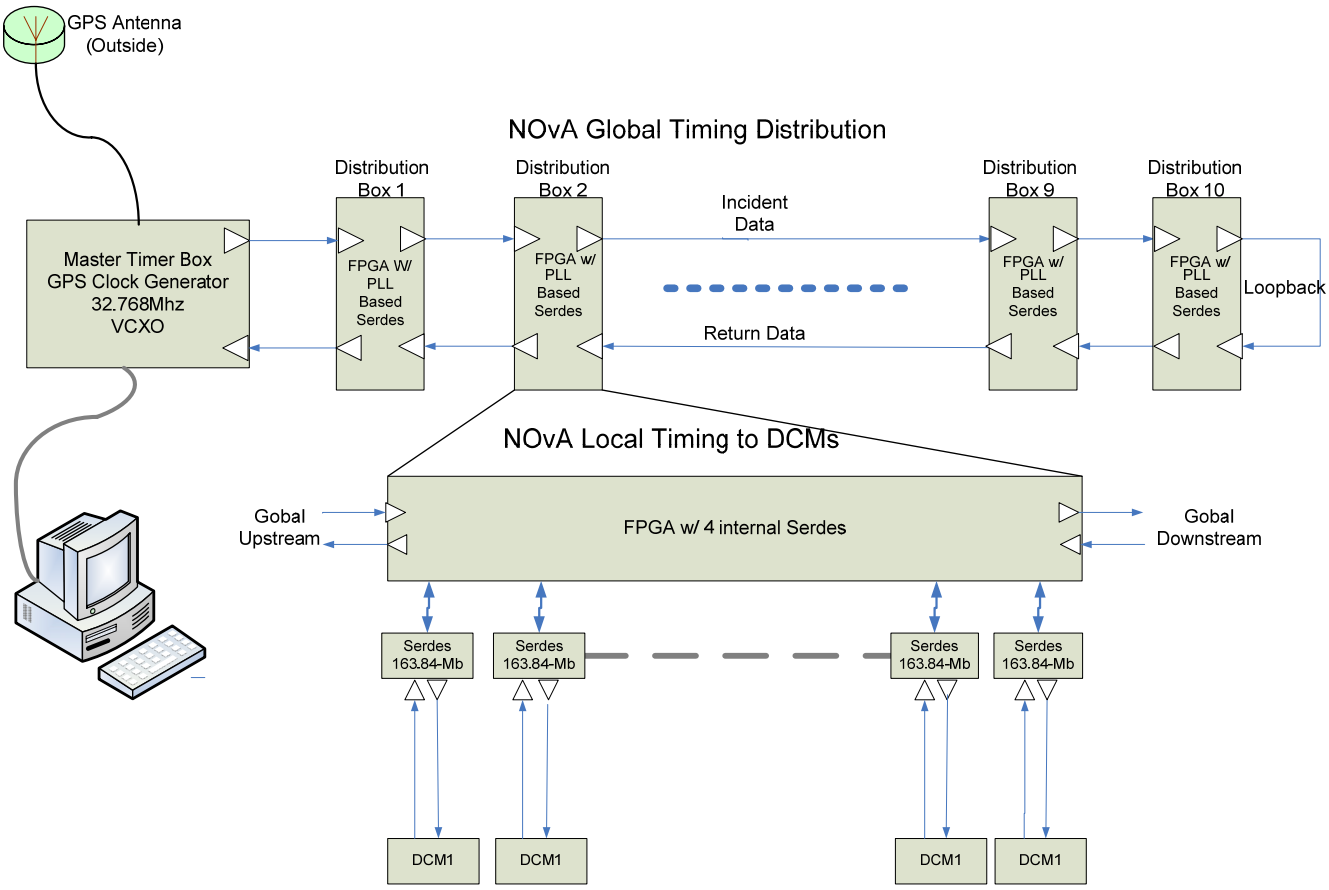

Fig. 5. Block Diagram of Global and Local Timing Distribution System

The ten Timing Distribution Boxes (TDB) make up the timing backbone of the detector, with the Master Timer Box at the near end as shown above in Fig. 5. The self compensation of the TDB backbone is based on the round trip time-of-flight (TOF) of a timing command through each TDB to the far end of the cable loop and back. The reference point for all of the timing is the far endpoint of the backbone loop. Each TDB measures the TOF independently and its measured value is related to its position along the backbone. The TOF measurement starts on the first arrival of the incident timing command and ends on the receipt of the return (or reflected) signal as in Fig. 6. This time-of-flight value is divided by two to give the specific TOF delay for each TDB across the entire backbone. A synchronization trigger output is generated by each TDB on the next timing command after counting the TOF/2 delay. All TDBs perform the same relative timing delay measurement synchronizing each TDB to the loop-back at the far end of the backbone. This method calibrates the propagation delay out of the TDB system. The cables between TDBs can be any length without effecting the timing synchronization. The synchronized trigger signal generated at all TDBs causes the timing command to be issued on all of the DCM links along the backbone simultaneously. The MTB also synchronizes to the far end reference point at the DCM in the same way.

Compensation of the propagation delay on each of the TDB to DCM links is performed similarly to the TOF/2 measurement of the detector backbone except there are multiple loops requiring the longest TOF/2 to be determined and used in all of the timing calculations. Rather than determining the longest TOF/2 and broadcasting it, an alternative method is to predetermine a delay parameter that is slightly greater than the longest actual measurement of all of the DCM links. The actual measured TDB to DCM round trip TOF is subtracted from this delay parameter in the timing calculation for each DCM port. Long cables will have short added delays and short cables will have long added delays such that all the DCMs receive timing signals at the same time. The TDB still compensates for cable lengths differences but the reference point is artificially longer than the longest link cable. An advantage to this method is that communicating the TOF parameters between TDBs about the TDB to DCM links is unnecessary and the single predetermined delay can be simply hard coded in the firmware.

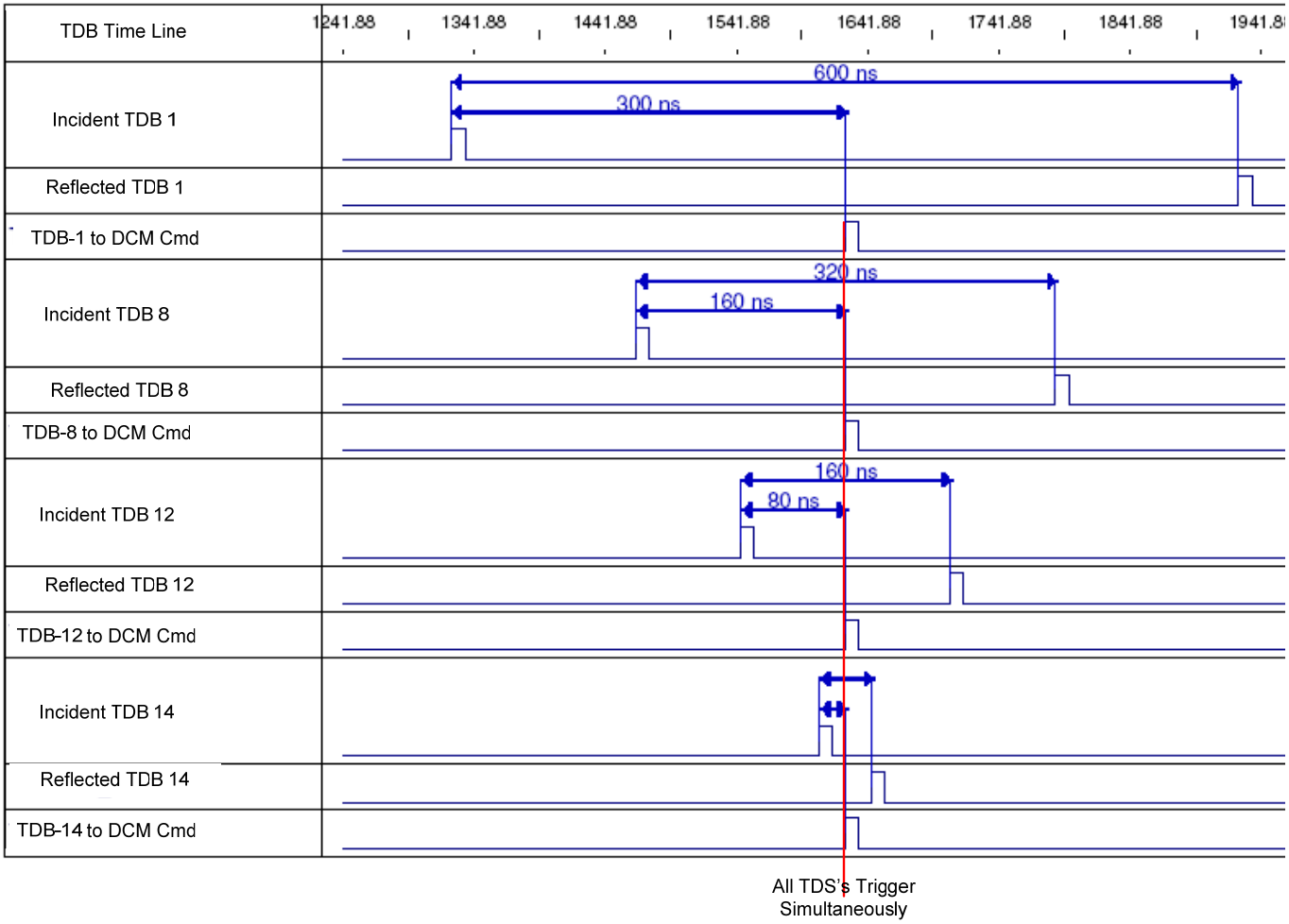

Fig. 6. Command Time-of-Flight and Synchronized Trigger at TDBs

IX. Future Plans

Pre-production DCMs are scheduled to be built in late 2007. The pre-production DCMs will differ from the prototype modules based on our experience with them to date.

Development of the timing system components is planned to begin in late 2007.

The experiment will have a CD-2/3a review in mid-2007 and if successful will move toward detector construction in 2011.

X. Conclusions

Version 1 of the Prototype DCM hardware is working, with one fully assembled and two partially assembled (CPU section only) boards. Firmware and software development is ongoing with approximately 80% of the firmware development finished and operating system porting complete.

Conceptual design of the Timing system is complete and prototype development is expected to begin later this year.

System software architectural development is essentially complete, and code development has begun. A small array of prototype buffer nodes have been assembled and are currently being used for development.

The DAQ and Timing System design cost-effectively satisfies the experiment's moderate speed and data volume requirements.

Acknowledgments

Special thanks are extended to Mark Bowden at Fermi National Accelerator Lab for his system architectural development work on this project.

References

[1] NO*v*A Document 1208, "DRAFT Technical Design Report Chapters 13 & 14, Electronics and DAQ", 01 March 2007, L. Mualem, et al., NOvA (E929) Document Database.

[2] NO*v*A Document 1145, "Preliminary Draft TDR, October 2006, Updated Nov 1", 01 November 2006, J. W. Cooper, et al, NOvA (E929) Document Database.

[3] K. Biery, A System for Exchanging Control Messages and Replies in the NOvA Data Acquisition", RT2007 Conference Record, in press.