# Rapid 3D Track Reconstruction with the BABAR Trigger Upgrade

S. J. Bailey, G. W. Brandenburg, N. Felt, T. Fries, S. Harder, M. Morii, J. N. Oliver, N. B. Sinev, E. Won

Abstract—A new hardware trigger system based on tracks detected by a stereo drift chamber has been developed for the *BABAR* experiment at the Stanford Linear Accelerator Center. The  $z_0 p_T$  Discriminator (ZPD) is capable of fast, 3-dimensional reconstruction of charged particle tracks and provides rejection of background events due to beam particles interacting with the beam pipe at the first-level trigger. Over 1 gigabyte of data is processed per second by each ZPD module. Rapid track reconstruction has been realized using Xilinx Virtex-II FPGAs.

Index Terms—BABAR, level-1 trigger, tracking, ZPD

#### I. INTRODUCTION

THE BABAR Experiment, operating since 1999 at the Stanford Linear Accelerator Center, studies physics of *B* mesons produced in collisions of electrons and positrons with a center-of-mass energy of 10.58 GeV. The data acquisition system for the experiment is driven by a multi-level trigger system. The first hardware trigger system is called the Level 1 (L1) Trigger. A L1 Accept initiates data transfer via optical fibers that connect the front-end electronics to the off-detector processors where further data reduction takes place. The latency of the L1 Trigger is fixed at 12  $\mu$ s, determined by the length of the data FIFO in the front-end electronics.

The rate at which *BABAR* can record data is primarily limited by the bandwidth of the GLink optical fibers between the drift chamber front-end and off-detector electronics. The data acquisition system was designed for a sustained L1 rate of 1 kHz; the rate saturates around 3.5 kHz for typical drift chamber occupancies. The L1 rate is rising as the accelerator continues to improve its luminosity. It is therefore necessary to upgrade the L1 Trigger system to reduce the background rate while maintaining high efficiency for the physics signal events.

The small cross-section of the  $e^+e^-$  collisions means that, even with the expected luminosity increase, the physics event rate remains less than 100 Hz. Fig. 1 shows the distribution of  $z_0$ , the position of the track's closest approach to the beamline (*z*-axis) measured from the nominal interaction point, for the tracks in the events that were accepted by the current L1 Trigger. In addition to the peak near  $z_0 = 0$  produced by the  $e^+e^-$  collisions, many tracks are detected with distances of 10 cm or larger from the interaction point. These tracks, produced by beam particles and synchrotron radiation photons hitting the beamline components near the experiment, comprise the bulk of the current L1 rate but are not useful for physics measurements.



**Fig. 1** Origin along the beamline (*z*-axis) of drift chamber tracks which pass the current Drift Chamber Trigger. The peak at  $z_0 = 0$  is from  $e^+e^-$  collisions; the remaining tracks are primarily background tracks from beam particle collisions with beamline components.

The current *BABAR* detector and trigger systems are described in [1-5]. The L1 Drift Chamber Trigger (DCT) receives inputs from a cylindrical small-cell drift chamber with 40 concentric layers arranged in 10 superlayers of 4 layers each. Four of the ten superlayers are strung parallel to the beam axis, while the remaining six are given 40-70 mrad stereo angles. The *z* information provided by this arrangement is used in the software trigger and in the offline event reconstruction, but not in the current L1 DCT, which finds tracks only in the 2D projection. The existing DCT is therefore unable to take advantage of the  $z_0$  distribution shown in Fig. 1 to reduce the background rate.

The goal of the *BABAR* L1 Trigger upgrade is to reduce the background trigger rate by implementing a 3D tracking capability in the L1 DCT. Eight new  $z_0 p_T$  Discriminator (ZPD) boards reconstruct 3D track information and make trigger decisions based upon the track's curvature and its  $z_0$ 

Manuscript received 24 November 2003.

S.J.Bailey, G.W.Brandenburg, N.Felt, T.Fries, S.Harder, M.Morii, J.N.Oliver, and E.Won are with the High Energy Physics Laboratory, Harvard University, Cambridge, MA 02318.

N.B.Sinev is with the Department of Physics, University of Oregon, Eugene, OR 97403.

origin with respect to the interaction point.

#### II. TRIGGER OVERVIEW

A block diagram of the upgraded trigger system is shown in Fig. 2 below.



Fig. 2 Block diagram of the upgraded BABAR drift chamber trigger system.

Hit information from the 7104 sense wires of the *BABAR* Drift Chamber is transmitted to the L1 DCT every 269 ns. Since this time sample is shorter than the ~600 ns total drift time, each event is spread over a few consecutive clock cycles. The Track Segment Finder (TSF) modules [2] use the time development of hit patterns over 3 clock cycles in 1776 overlapping groups of 8 wires each to identify track segments. Configurable look up tables convert the hit patterns for each group (8 wires times 2 bits per wire) into segment locations and weights. The segment locations estimate the position of tracks in each superlayer with a resolution of ~800  $\mu$ m in comparison to the offline reconstructed track trajectories. The weights are used to select the best 3 segments per superlayer per 1/16 in azimuth to send to the ZPD modules, which perform 3D track reconstruction.

The Drift Chamber is organized into octants and superlayers as shown in Fig. 3 below. Superlayer designations A, U, and V correspond to axial, right hand stereo, and left hand stereo respectively.



**Fig. 3** End view of one half of the BaBar Drift Chamber showing the axial (A) and stereo (U and V) superlayer structure. An example track with 8 segments is shown. The superlayer octants with potential seed segments for one ZPD are highlighted. In this example, a neighboring ZPD would find the same track using the A7 seed instead of the A10 seed.

Each ZPD receives segment data from three octants. Seed segments in axial superlayers A10 and A7 of the central octant of each ZPD serve as a starting point for the tracking algorithm which searches for tracks which come from the origin and pass through a seed segment. The two adjacent octants provide tracking coverage for low  $p_{\rm T}$  tracks whose curvature extends outside the central octant. The two side octants are the central octants of neighboring ZPDs.

The tracks are fit for their  $z_0$ , curvature, and dip angle. These parameters are used to select signal events which have tracks from  $e^+e^-$  collisions.

The current L1 DCT uses TSF modules that report detailed segment information only for the axial superlayers. A set of 24 updated TSF modules that send the full segment information from both the axial and the stereo superlayers are being built to replace the current modules.

Additional trigger modules use electromagnetic calorimeter information to make trigger decisions based upon neutral particles. A Global Trigger board combines information from each L1 Trigger subsystem to make the L1 Trigger decision.

## III. Z<sub>0</sub> P<sub>T</sub> DISCRIMINATOR (ZPD)

Organization of the ZPD is shown in Fig. 4 below.



Fig. 4 Block diagram of the ZPD dataflow, showing each of the FPGAs.

Each ZPD receives data from 9 TSFs as single ended standard 3.3V CMOS levels via an interface board on a 153 bit wide, 60 MHz bus for a total data rate of ~10 GB/s. Each bit serially transmits a track segment location and a 3 bit error code indicating the quality of the segment position estimate. A frame bit from each TSF indicates the start of a new serial segment "packet". These packets are continuously reformatted and transmitted along a 75 bit multi-drop LVDS bus, called the "MegaBus" to six "Finder-Fitter" FPGAs (Xilinx XC2V4000, 1152 pin BGA). The MegaBus transmits its data in "Double Data Rate" (DDR) format in which the data change at both edges of the 60 MHz clock for an effective 120 MHz transfer rate. Electrically, the MegaBus consists of pairs of 6 mil traces on top of each other and surrounded top and bottom by ground planes in the circuit board. This configuration results in a 60 Ohm differential transmission line terminated at both ends. This results in very clean data transitions which have been tested to over 160 Mb/s (30% above spec) on each line with no bit errors seen in ~10<sup>13</sup> bits per line. One such pair fits between the vias in the 1 mm ball grid array of the Finder-Fitter FPGAs. The ZPD is a twelve layer 9U VME printed circuit board, the number of layers driven primarily by the need to accommodate the MegaBus differential traces and ground planes.

There are six Finder-Fitters on the MegaBus, each being responsible for two of the twelve seed segments. Three of the Finder/Fitters process A10 seed segments and three process A7 seed segments. Each group of three Finder/Fitters uses identical firmware and a geographic address is used to distinguish them. Fit results from the Finder/Fitter FPGAs are combined by a Decision Module FPGA which selects signal events that have tracks originating near  $z_0 = 0$ . The ZPD logic has a latency of 2.2 µs. Firmware designs stored on a Compact Flash card are downloaded to the FPGAs using Xilinx SystemACE CF.

## A. ZPD Track Finder Algorithm

The ZPD Track Finder algorithm uses a Hough transform [6] to search for charged particle tracks which originate from the  $e^+e^-$  interaction point and pass through a seed segment in axial superlayers A7 or A10. For each seed segment, 260 different curvature and dip angle possibilities are considered in parallel. The origin, seed segment location, curvature and dip angle hypothesis determine the trajectory of a potential track. The best option is selected based upon the number of superlayers which have segments consistent with each track parameter hypothesis.

Each Finder searches for tracks using two different seed segments which are processed serially using the same pipelined logic. The first step of the algorithm adjusts each segment azimuthal location (phi) by subtracting the seed phi so that the subsequent algorithm is independent of an overall azimuthal rotation.

A Pattern Recognition Matrix (PRM) records how many superlayers have segments which are consistent with the track parameter hypothesis for each bin. The bin with the most hits is selected as the most likely track trajectory. If the maximum spans several bins, the track parameter values are averaged over those bins. The best track option must have no more than 2 superlayers with missing segments at a radius closer than the seed superlayer.

A look up table converts the track parameters into the expected phi positions of segments along the track. The segments closest to the expected phi value are associated with the track and passed to the Fitter algorithm for the full 3D track parameter fit.

The Finder algorithm is implemented in Xilinx Virtex-II 4000 FPGAs. Individual steps of the algorithm are area constrained within the FPGA in order to assist the place and

route software. The PRM logic dominates the resource usage of the Finder-Fitter FPGAs and care was taken in order to minimize the resource usage of the basic elements which are replicated thousands of times. Dual port block RAMs internal to the Virtex-II FPGAs are used for input segment buffering and look up tables. The segments arriving on the MegaBus are written to these memories at 120 MHz and read at 60 MHz, processing up to 26 segments at a time. Individual block RAMs are location constrained to assist the place and route software in achieving the timing constraints.

A C++ program which simulates the Finder algorithm was used to optimize the number of track hypotheses considered and make design choices related to algorithm performance versus resource usage. The current design uses 260 hypotheses which provides sufficient performance while still fitting within affordable FPGAs. This program also outputs VHDL code containing the PRM constants which map segment phi values into PRM bins. The VHDL code was designed to minimize the differences between A7 and A10 Finders and the logic for either is selected by a single generic constant at the time of synthesis.

# B. ZPD Track Fitter Algorithm

The Fitter algorithm receives a set of up to 10 spatial points, one in each superlayer, from the Finder algorithm, and attempts to fit a helix to them. This is done in a 2-stage, noniterative process that ensures efficient hardware implementation with a fixed latency.

In the first stage, the points are projected on the *x*-*y* plane at z=0 and are fitted to an arc. Each of the three pairs of neighboring stereo points (U2 and V3, U5 and V6, and U8 and V9) are combined to cancel the effects of the stereo angles, so that they provide an effectively axial measurement. The fit starts from an arc that passes the interaction point and the seed (A7 or A10) segment, with the curvature determined by the Finder. The residuals at each spatial point and their first derivatives over the curvature are calculated. Newton's method is applied once to improve the value of the curvature. Iterative application of this procedure would improve the accuracy of the curvature only marginally at a significant cost in latency.

In the second stage, the residual of each stereo point with respect to the 2-dimensional arc is converted into a z measurement. These (up to 6) spatial points form a straight line on the cylinder defined by the arc. A simple linear least-square fit gives the  $z_0$  and the dip angle. The error on the  $z_0$  is estimated from the number of spatial points available to the Fitter.

The Fitter algorithm is implemented in the same FPGAs as the Finder, utilizing the dedicated multipliers and block RAM. The highly pipelined algorithm runs with a 60 MHz clock and requires the dedicated resources to meet timing requirements. The block RAM is used as a lookup table where algorithm values are stored. The critical path in the design is a multiplication with a value from a lookup table. The dedicated multipliers and block RAM are physically close in the FPGA to minimize delays. The remaining algorithm logic is constrained to be in the same area to further minimize delays.

#### C. Decision Module

The Decision Module FPGA collects fitted track parameter information from each of the Finder-Fitter FPGAs and makes trigger decisions. Four independent decision bits are reported to the global trigger based upon cuts in the number of missing segments, curvature, dip angle,  $z_0$ , and the error on  $z_0$ . If any track passes the set of cuts, the event is accepted by the ZPD. An additional 4 decision bits are stored in the output data stream for testing new cut parameters. The cut parameters are stored in memory which is configured at the beginning of a data-taking run.

### D. Diagnostic Memories

Six diagnostic memories at various places throughout the dataflow provide debugging capabilities for the ZPDs. These memories are implemented using dedicated block RAMs within the Virtex-II FPGAs and can store 64 packets of information. These can either record the datastream or be used to play data to downstream pieces of the data processing pipeline. These memories have been used to test the hardware, debug the firmware implementation, and compare the ZPD results with the C++ simulation code.

#### E. Implementation Tools

The CAE tools used to design the ZPD were provided by Mentor Graphics. All of the firmware was written in VHDL using "HDL Designer Series" to produce a well laid out hierarchical design. The design implementation relied heavily on accurate VHDL simulations using ModelSim. Leonardo Spectrum was used for VHDL synthesis. Xilinx ISE was used to place and route the FPGAs.

#### IV. PERFORMANCE

The on-board interconnects among the FPGAs are validated by a boundary scan. The MegaBus, which carries the largest amount of data, has been tested using dedicated FPGA firmware that transmitted rotating bit patterns. This allowed a stress-test the bus up to  $\sim 10^{15}$  bits with no errors and with higher clock frequencies than the nominal double data rate 60 MHz. The input segment bus has been tested with  $\sim 10^{13}$ bits with no errors.

More complex functionalities have been tested using the diagnostic memories. Real data from the current trigger system as well as simulated input data was transmitted through various steps of the algorithm and the results were recorded. These results were compared with the C++ ZPD simulation program which was used to develop the algorithm and study its expected performance. The hardware has a bit-wise match with the simulation for over 10,000 events tested.

Using a quarter of the entire upgrade system, the ZPD performance was measured with real data using a fiber optic splitter to obtain Drift Chamber input data while the current DCT was running. The ZPD has a 95% efficiency for

reconstructing tracks which were found by the offline reconstruction with  $p_T > 300$  MeV/c. The measured  $z_0$  resolution is 4 cm, consistent with expectations from Monte Carlo simulations. This is sufficient to distinguish signal tracks at  $z_0 = 0$  from background tracks at  $|z_0| > 10$  cm.

Background data and Monte Carlo simulations of signal events were used to study the ZPD signal and background selection efficiency. Fig. 5 shows the ZPD selection efficiency for signal and background events which pass the current L1 trigger as a function of the  $|z_0|$  cut applied. Triangles show the efficiency for generic signal events, while circles show the efficiency for events where one of the two *B* mesons decays  $B \rightarrow \pi^0 \pi^0$  which results in charged tracks only from the other *B*. Squares show the efficiency for background events. A simple cut of  $|z_{0,\min}| < 10$  cm can reduce the background rate by 50% while maintaining a >99.5% signal event efficiency. More sophisticated cuts which combine track  $p_T$  and track multiplicity information can further reduce the background rate while maintaining high signal efficiency.



**Fig. 5** ZPD selection efficiency for signal (circles and triangles) and background (squares) events which pass the current L1 trigger as a function of the  $|z_0|$  cut applied. A cut of  $|z_{0,min}| < 10$  cm is sufficient to reduce the background trigger rate by 50% while maintaining a >99.5% efficiency for signal events.

#### REFERENCES

- B. Aubert, A. Bazan, A. Boucham, D. Boutigny, I. De Bonis, J. Favier et al., (the BABAR Collaboration), "The BABAR Detector," Nucl. Instrum. Methods, vol. A479, pp. 1-116 Feb./Mar. 2002.
- [2] A. Berenyi, H.K. Chen, K. Dao, S.F. Dow, S.K. Gehrig, M.S. Gill, et al., "Continuously Live Image Processor for Drift Chamber Track Segment Triggering," *IEEE Trans. Nucl. Sci.*, vol. 46, pp. 348 – 353, June 1999.
- [3] —, "A Binary Link Tracker for the BABAR Level 1 Trigger System," IEEE Trans. Nucl. Sci., vol. 46, pp. 928 - 932, June 1999.
- [4] —. (1998, Nov.). Design and Implementation of the Level 1 Charged Particle Trigger for the BABAR Detector. Presented at IEEE 1998 NSS.
   [Online]. Available: http://www.slac.stanford.edu/BFROOT/www/Detector/Trigger/98/ieee\_
- papers/dct-ieee.ps
  [5] —. (1998, Nov.). A Real-Time Transverse Momentum Discriminator for the BABAR Level 1 Trigger System. Presented at IEEE 1998 NSS.
   [Online]. Available: http://www.slac.stanford.edu/BFROOT/www/Detector/Trigger/98/ieee\_ papers/ptd-ieee.ps
- [6] P.V.C. Hough, "A Method and Means for Recognizing Complex Patterns", U.S. Patent 3069654, 1962.