A Binary Link Tracker for the BABAR Level 1 Trigger System[1]

A. Berenyi, H.K. Chen, K. Dao, S.F. Dow, S.K. Gehrig, M.S. Gill, C. Grace, R.C. Jared, J.K. Johnson, A. Karcher, D. Kasen, F.A. Kirsten, J.F. Kral, C.M. LeClerc, M.E. Levi, H. von der Lippe, T.H. Liu, K.M. Marks, A.B. Meyer, R. Minor, A.H. Montgomery and A. Romosan
E.O.Lawrence Berkeley National Laboratory, Berkeley, California 94720

Abstract

The BABAR detector at PEP-II will operate in a high-luminosity e^+e^- collider environment near the $\Upsilon(4S)$ resonance with the primary goal of studying CP violation in the B meson system. In this environment, typical physics events of interest involve multiple charged particles, and can be identified from these tracks in a fast first level (Level 1) trigger system, by reconstructing the tracks in "real time". For this purpose, a Binary Linker Tracker Module (BLTM) was designed and fabricated for the BABAR Level 1 Drift Chamber trigger system. The BLTM is responsible for linking track segments, constructed by the Track Segment Finger Modules (TSFM), into complete tracks. A single BLTM module processes a 360 MBytes/s stream of segment hit data, corresponding to information from the entire Drift Chamber, and implements a fast and robust algorithm that tolerates high hit occupancies as well as local inefficiencies of the Drift Chamber. The algorithms and the necessary control logic of the BLTM were implemented in Field Programmable Gate Arrays (FPGAs), using the VHDL hardware description language. The finished 9U$\times$400 mm board contains roughly 75,000 gates of programmable logic or about 10,000 lines of VHDL code synthesized into 5 FPGAs.

I. INTRODUCTION

A. Overview of the Level 1 Trigger

The environment of the high-luminosity PEP-II machine poses unique challenges to the trigger system for the BABAR detector which led to the adoption of a fast hardware processor-based (Level 1) trigger[1]. As shown in Figure 1, the Level 1 trigger system consists of three subsystems: the Drift Chamber Trigger, the Calorimeter Trigger and the Global Trigger. Information from the Drift Chamber and Calorimeter is processed by the respective triggers and the resulting data is sent to the Global Trigger (GLT). The algorithms of the Drift Chamber Trigger System are executed on three types of modules. First, the track segments, i.e., series of hits contiguous in space and time, are found by a set of 24 Track Segment Finder Modules (TSFMs)[2]. The track segments are then linked into complete tracks by the Binary Link Tracker Module (BLTM). A set of 8 PT Discriminator Modules (PTDMs)[3] determine if the segments are consistent with tracks of particles having a transverse momentum (P_t) greater than a specified minimum. The GLT attempts to match the angular locations of calorimeter towers and drift chamber tracks, and generates Level 1 triggers based on the results of the processing.

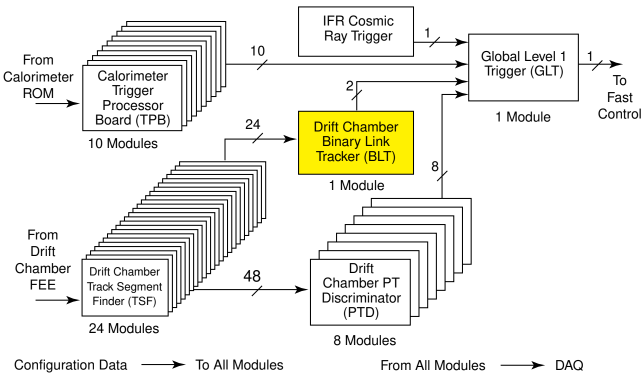

Figure 1: BABAR Level 1 Trigger System

B. Drift Chamber Input and Track Segment Finding

To support the Level 1 trigger system architecture, dedicated data paths provide the necessary information to the Level 1 trigger hardware. The Drift Chamber trigger input data is a collection of raw Drift Chamber hits. The Drift Chamber is a small-hex-cell design and contains 40 layers grouped four each in 10 radial superlayers. There are four axial superlayers (A1, A4, A7 and A10), with each pair of axial superlayers separated by two stereo superlayers (U2, V3, U5, V6, U8 and V9). Each superlayer is divided into 32 sectors in the azimuthal (ϕ) direction. A magnetic field of 1.5 T, parallel to the beam axis, forces charged particles on a circular trajectory and facilitates the measurement of their transverse momentum. The Drift Chamber front-end electronics (FEE) system [4] contributes "real-time" information (hit signals) continuously to the Drift Chamber Trigger system at a rate of 3.7 MHz, using discriminators to digitize the voltages on each sense wire into single bits [5]. The Track Segment Finder determines which of the 32 supercells in each of the 10 superlayers contain track segments of interest. By using a novel method incorporating both occupancy and drift-time information, the TSF system is able to continually search for segments in the supercells of the full 7104-wire Drift Chamber hit image at 3.7 MHz, and reports a total of 10$\times$32 bits from the 24 TSFs at 7.4 MHz to the BLT, all in coincidence[2]. Each bit indicates whether a track segment from a charged particle was found in that supercell or not.

[1]This work was supported by the Division of High Energy Physics of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098. In addition, A.B.M. was supported by the Alexander von Humboldt Foundation, Bonn, Germany.

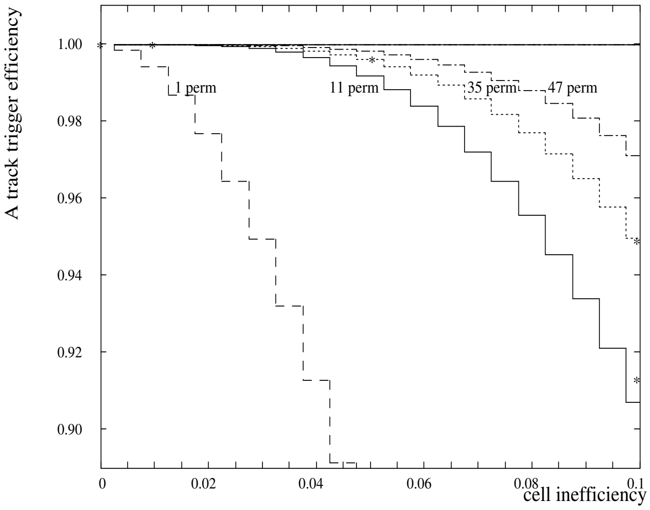

Figure 2: Calculated Trigger Efficiency for various BLT versions. The trigger efficiency calculated here is for an "A" track.

C. *Binary Link Tracker Module (BLTM)*

The BLTM receives segment hit information from all 24 TSFM at a rate of 320 bits per clock tick (7.4 MHz) and links them into complete tracks. Tracks that reach the outer layer of the drift chamber (superlayer A10) are labeled of type "A". Tracks that reach the middle layers of the drift chamber (layer U5) are labeled of type "B". It is possible for some "B" tracks to continue far enough to also become indicated as an "A" track. These radial travel distances impose implicit transverse momentum (P_t) cuts on the A and B tracks of 180 and 120 MeV respectively. The track linking algorithm uses an extension of a simple algorithm, developed for the CLEO II trigger [6]. A single BLTM module processes a 360 MByte/s stream of segment hit data, corresponding to information from the entire Drift Chamber. The resulting track information is then compressed and output to the Global Level Trigger in the form of two 16-bit words corresponding to either "A" or "B" tracks respectively. Each bit in a word represents the "track hit" state of a supercell in the designated superlayer.

One of the challenges in designing the Level 1 trigger lies in implementing efficient selection algorithms capable of handling unforeseen beam backgrounds and impaired detector performance in a robust way. The task of the BLTM is to find tracks corresponding to real charged particles in an efficient way, while minimizing the finding of "false" tracks due to spurious background hits. There are two possible causes for track finding inefficiency: dead or highly inefficient Drift Chamber supercells, and nominal Drift Chamber cell inefficiency. To protect against dead or highly inefficient supercells, a programable masking capability is provided at the input stage of the BLTM to allow the system to activate track segments from these supercells. To account for nominal cell ineffciencies, the CLEO II algorithm was modified to allow for up to two superlayers to be inefficient which further enhances the track finding efficiency without significantly increasing the background rate.

In this paper, the design and the hardware implementation

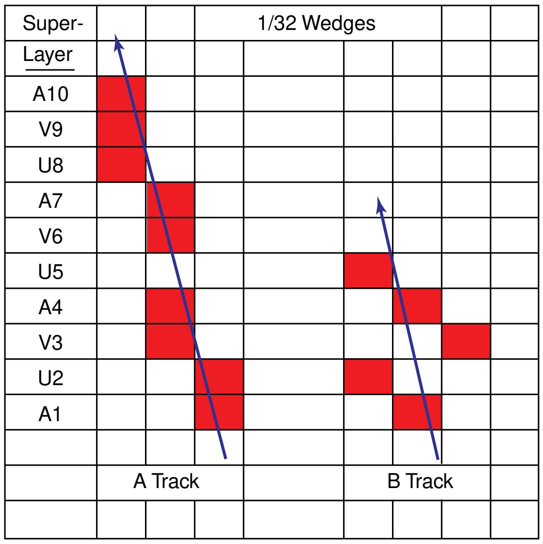

Figure 3: Track Linker Algorithm: two examples

of the Binary Link Tracker Module is described.

II. THE BINARY TRACK LINKER ALGORITHM

The BLT Module algorithm consists of three major sections: the Channel Select, the Track Linker and the Post Processor. The functionality of each section is described below.

The Channel Select consists of 320 OR gate blocks, one for each supercell. The inputs to the BLTM, consisting of segment hits found by the TSFMs, are logically "OR"ed in the Channel Select with a programmable mask pattern. This allows the system to activate track segments that correspond to dead or highly inefficient cells so that the track linking efficiency does not degrade.

The track linker algorithm starts from the innermost superlayer, A1, and moves radially outward. A track is found if (1) there is a segment hit in every layer, and (2) for a segment hit in a given superlayer there is a corresponding segment hit in the next superlayer within a certain range of supercells. The range is chosen such that it allows for track curvature in the magnetic field and dip angle variations considering the intrinsic geometry of the Drift Chamber. Wires in the axial layers are parallel to the direction of the magnetic field (z-axis), while wires in the stereo layers are oriented at an angle with respect to the z-axis. Given segments associated with the same track, the azimuthal separation between them in neighbouring superlayers depends on whether the two superlayers are both stereo layers, or one of them is an axial layer. Hence, for the stereo to axial transition the range was chosen to be 3 supercells, while it is 5 supercells for the stereo to stereo transition.

The track finding effciency was studied as a function of Drift Chamber cell inefficiency for different number of allowed missing superlayers, as shown in Figure 2. If the BLTM required all superlayers to contribute when linking a track using the algorithm described above, the track finding efficiency

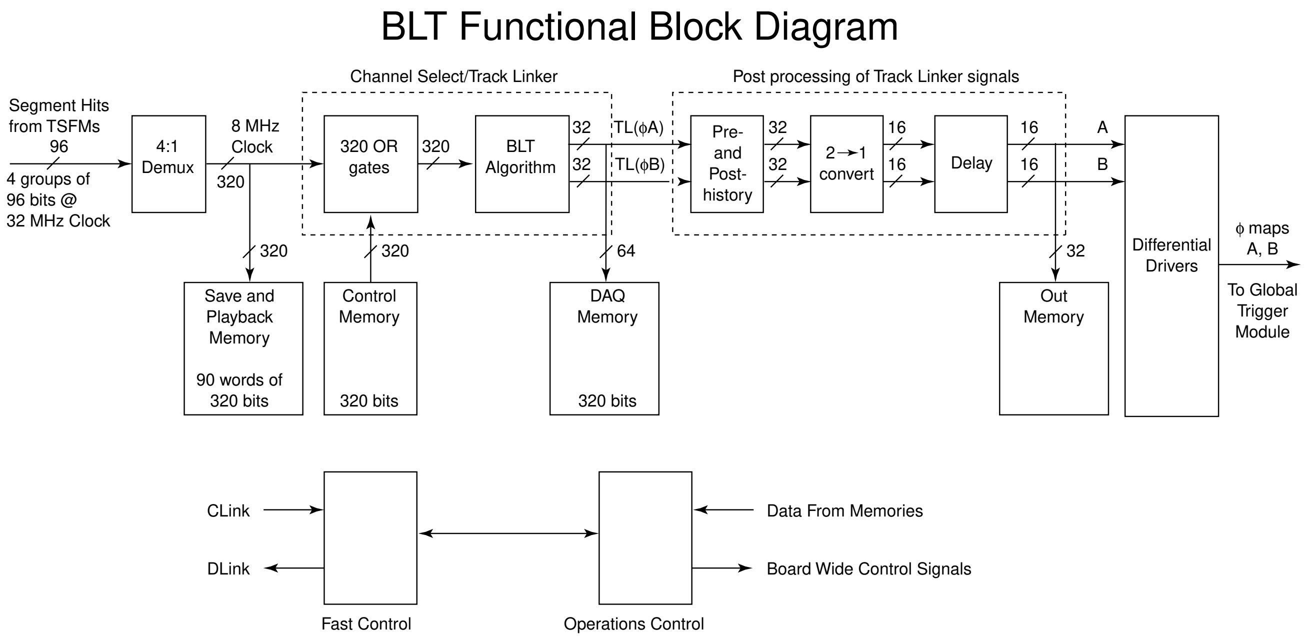

Figure 4: BLT Functional Block Diagram

would decrease rapidly with increasing cell inefficiency (see curve labeled "1 perm"). This is because the track linking efficiency of the BLTM depends on the segment finding efficiency of the TSFM which is affected by Drift Chamber cell inefficiencies.

One natural extension to the simple track finding algorithm is to allow for one or more superlayers to be skipped. If only one superlayer is skipped, resulting in 11 different allowed permutations, track finding efficiency still drops significantly with increasing cell inefficiency (see curve labeled "11 perm"). Allowing any two non-consecutive superlayers to be skipped, resulting in 47 different allowed permutations, would improve significantly the track finding efficiency even at high cell inefficienies (see curve labeled "47 perm"), but it would make the BLTM algorithm very sensitive to potential backgrounds due to the rather loose requirements. Therefore, to make the BLTM robust against cell inefficiencies as well as potential backgrounds, an "in-between" approach was chosen to include tracks for which hits in one or two superlayers are missing as long as the following criteria are satisfied: (1) Either or both of the U5 and V6 layers must have a hit, (2) Only one of layers A1, U2, V3, A4 and U5 may be missing a hit, and (3) Only one of layers V6, A7, U8, V9 and A10 may be missing a hit. This results in 35 allowed permutations (see curve labeled "35 perm"), and it gives track finding efficiencies above 95% even for 10% cell inefficiency, while it is more robust against potential backgrounds compared to the 47 permutation case.

Figure 3 shows two possible segment hit patterns, one for an "A" track and the other for a "B" track, which can be used to illustrate how the BLTM linking algorithm works. For example, in the "B" track case, starting from the segment hit in superlayer A1, the linker will then look for the nearest segments in superlayer U2 where it finds a hit. The linker will then continue to look for segments in superlayer V3, and so on until it reaches U5. At this point, a "B" track is considered found. These steps are then continued until either no more segments are found in the following superlayers satisfying the selection criteria, or it successfully links all the way to superlayer A10, to form an "A" track. The example on the left hand side of the figure shows a segment hit pattern for a valid "A" track for which the segment hit corresponding to superlayer U5 is missing.

Two 32-bit words corresponding to "A" and "B" tracks respectively are output from the Track Linker, with each bit in a word representing whether or not a track of a given type was linked through the corresponding supercell of the given superlayer. This information, which provides the ϕ location in either the A10 or U5 superlayer for each track found, is output to the Post Processor. To account for possible clock jitter, the Post Processor first extends in time each bit by one 7.4 MHz clock tick in both directions, and then it compresses the two words to 16 bits to better match the calorimeter's 20-bit ϕ resolution. The result is then sent to the GLT where information from the Calorimeter and Drift Chamber Modules is processed and Level 1 trigger accepts are issued for relevant events.

III. Hardware Implementation

The algorithms and the necessary control logic of the BLTM were implemented in Field Programmable Gate Arrays

(FPGAs), using the VHDL hardware description language. The finished 9U×400 mm board contains roughly 75,000 gates of programmable logic or about 10,000 lines of VHDL code synthesized into 5 FPGAs: Fast Control, Operations Control, Channel Select, Track Linker, and Post-Processor, as shown in Figure 4. For the actual hardware implementation, the ORCA 2C series FPGAs from Lucent Technologies were chosen. Additionally, memories of various types are used on the board for several purposes.

A. *Control Logic*

The Fast Control FPGA implements a high speed serial link responsible for remote control and configuration of the board. It is a generic chip common to all the Level 1 trigger modules. The Operation Controller interprets the commands from the Fast Control unit and coordinates the functionality of the board by implementing the control and interface logic specific to the BLTM.

B. *Channel Select and Track Linker*

The hardware implementation of the Channel Select and Track Linker is illustrated in Figure 5. The incoming segment hit information from the Segment Finders is first "OR"ed with a progrmmable mask pattern in the Channel Select, which consists of 320 gates. The resulting 10 32-bit words containing incoming segment hit signals for individual supercells from superlayers 1 through 10 (HIT 1 through HIT 10) are input to the Track Linker. The Track Linker consists of an array of logic blocks, 32 columns wide by 10 rows high. In the figure, only four columns of are shown. Each logic block is implemented as an AND-OR gate pair, as shown in Figure 5.b. For a given superlayer, segment hits from 3 or 5 consecutive supercells are "OR"ed together. The output of the OR is fed to the AND gate along with the segment hit signal from the like numbered supercell of the next layer. The output of all AND gates in the row is collected in the TRAIL field to serve as input to the OR gates for the next superlayer.

The logic to implement the complete track linker that allows missing segment hits is more complex than the basic implementation described above. It consists of 34 additional copies of the basic track linker, each copy modified so as to allow missing segment hits from certain supercells. For the complete linker, the output for a "B" track is the 32-bit "OR"ed sum of the TRAIL 5 of all copies, while the output for an "A" track is the 32-bit "OR"ed sum of all TRAIL 10's. Some copies have the logic for superlayer 5 or 10 missing. In the case of HIT 5 missing, "B" track output will use TRAIL 4. Likewise, for the case where HIT 10 missing, "A" track output will use TRAIL 9.

The actual VHDL code of the 35 track linker copies was implemented in a highly effcient manner. This allows the complete track linker to be implemented in one FPGA.

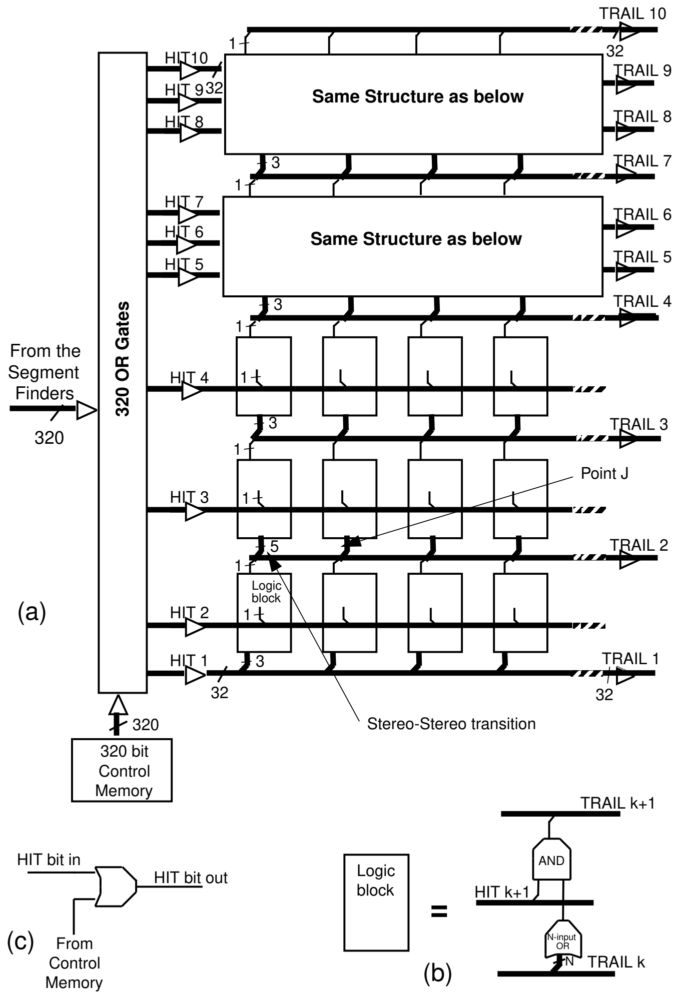

Figure 5: (a) An illustration of the structure of the basic Track Linker. The complete linker consists of 32 columns of logic blocks, of which four columns are shown. (b) Details of a logic block. (c) One of the 320 OR gates.

C. *Post Processor*

Two 32-bit words corresponding to "A" and "B" tracks from the Track Linker are output to the Post Processor FPGA, where they are modified before being sent to the Global Trigger (GLT). The modifications consist in the addition of pre- and post-history to the data to account for possible clock jitter, and the compressesion of the 32-bit words to 16 bits (2 to 1 covertion) to better match the calorimeter's 20-bit ϕ resolution. The addition of pre- and post-history is done by extending each hit in time one extra clock tick in both directions. This is to ensure that all linked tracks from a given event are in coincidence before being sent to the GLT.

D. *Memory Buffers*

A DAQ memory circular buffer is used to save data related to a particular Level 1 trigger. The data saved can be read out, allowing for later reconstruction of the processing that occurred in the BLT module during the creation of that particular trigger. The buffer has sufficient storage to hold the history during the time required for the trigger decision to be made.

Additionally, Save and Playback diagnostic memories are attached to the input and output data streams. The input/output memories allow all the algorithms and interfaces on the board to be tested. Through application of test vectors of "typical" event data from Monte Carlo simulations of the complete Level 1 trigger electronics pipeline, one can debug and test the board in a systematic way. The test results are described at the end of this paper.

IV. Design Methodology

The hardware implementation for the trigger system was designed using a top-down design methodology. The design of the trigger architecture and selection algorithms was based on simulation studies of the detailed response of the BaBar detector to physics and background events. A software model of the complete Level 1 Trigger electronics pipeline was developed for this purpose.

A major fraction of the design effort was dedicated to the implementation of the BLTM algorithms and the necessary control logic within FPGAs using the VHDL hardware description language. The design takes full advantage of state-of-the-art Computer Aided Design tools[7]. These sophisticated tools, which include routing algorithms for logical arrays and gate-level simulation greatly streamlined the design process. The finished 9U$\times$400 mm board contains roughly 75,000 gates of programmable logic or about 10,000 lines of VHDL synthesized into five FPGAs.

The BLTM was simulated prior to fabrication to ensure that the connectivity of the board was correct. The simulations also uncovered logic errors that would have been very difficult to find otherwise. The layout of the board emphasizes the careful routing and termination of critical signals such as clock lines. Also the timing of signals on the board contain large safety margins with respect to delays and setup/hold times. The FPGAs, for example, use flip-flops in the I/O blocks of the chip to re-clock all data going into or out of the chip. Finally, spare communication lines connect all FPGAs together, while LEDs and special connectors allow for easier debugging.

V. Test Results and conclusion

The BLTM prototype has undergone extensive testing of its algorithms using the memory's diagnostic play/record features. It was fed data from simulations of over 15,000 Monte Carlo events, and the results were successfully compared against the software model for the Level 1 trigger. The test bench results were in full agreement with all computer simulations of the BLTM. The module was also tested at a clock speed of 68 MHz (the nominal clock speed is 59.5 MHz) and no problems were found. There were no layout or fabrication errors found on the prototype board, allowing it to serve as the production version for the experiment. A limited system test of the 24 TSF boards together with one BLTM board was also performed in which the BLTM was able to communicate successfully with all 24 TSF boards. All the boards were installed at the BaBar detector in October 1998 for the BaBar cosmic ray commissioning test.

VI. References

[1] C.T. Day *et al.*, "The BaBar Trigger, Readout and Event Gathering System," Proceedings of the International Conference on Computing in High Energy Physics, 1995. A. Berenyi *et al.*, "Concept and Design for the Level 1 Charged Particle Trigger of the BaBar Detector," *in these proceedings*.

[2] A. Berenyi *et al.*, "Continuously Live Image Processor for Drift Chamber Track Segment Triggering," *in these proceedings*.

[3] A. Berenyi *et al.*, "A Real-Time Transverse Momentum Discriminator for the BaBar Level 1 Trigger System," *in these proceedings*.

[4] J. Albert *et al.*, "Electronics for the BaBar Central Drift Chamber", *in these proceedings*.

[5] S.F. Dow *et al.*, "Design and Performance of the ELEFANT Digitizer IC for the BaBar Drift Chamber", *in these proceedings*.

[6] K. Kinoshita, *Nucl. Instr. Methods A276 (1989)242* .

[7] Mentor Graphics Corp., Synopsys Corp. and Lucent Technologies.