SDC-92-203

SDC
SOLENOIDAL DETECTOR NOTES

EFFICIENT DATA TRANSMISSION FROM SILICON WAFER STRIP DETECTORS

February 1992

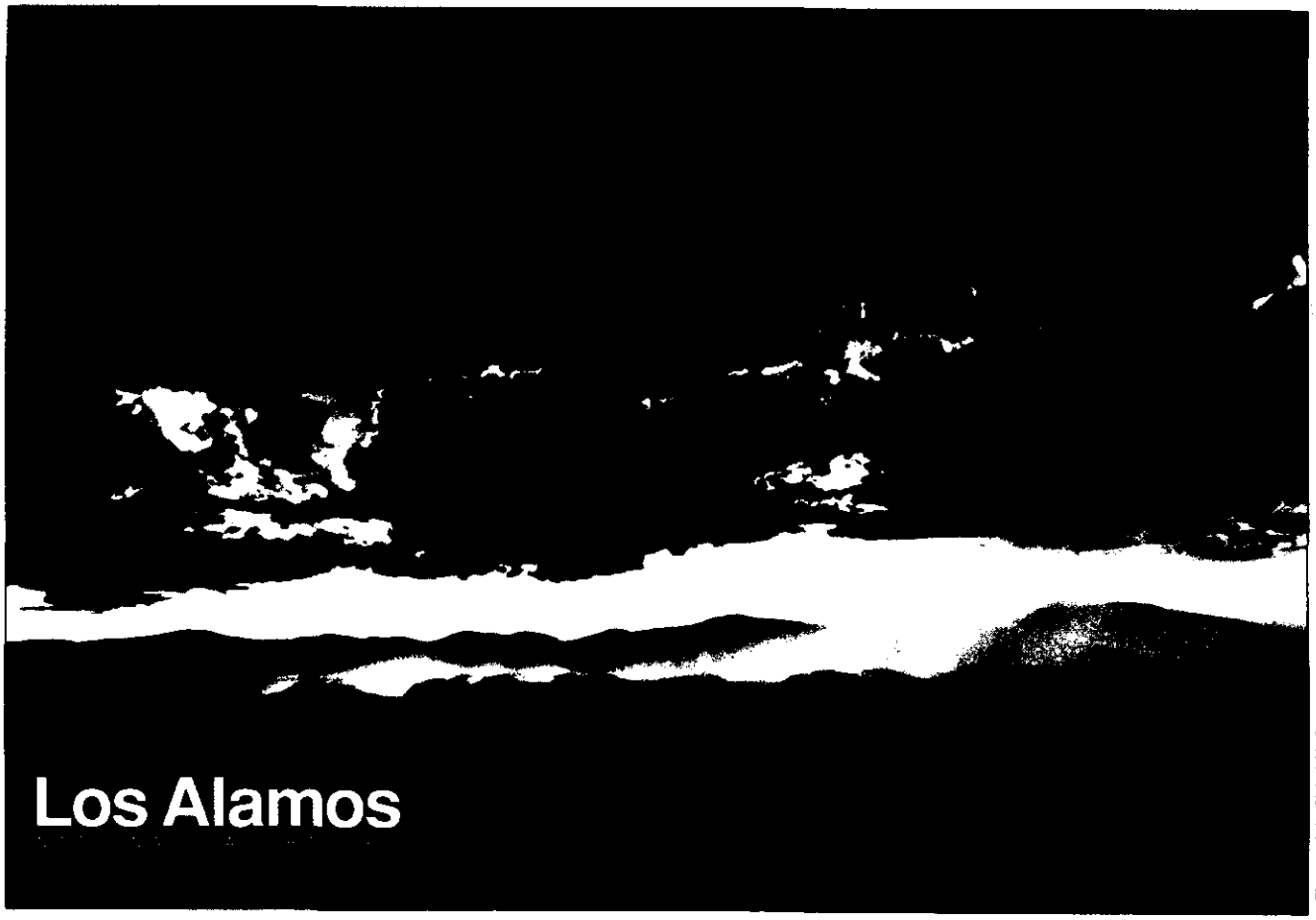
Bradly J. Cooke, Klaus S. Lackner, Andrea P.T. Palounek, David H. Sharp,
Larry Winter and Hans-Joachim Ziock
Los Alamos National Laboratory

LA-UR-91-3903

Efficient Data Transmission from Silicon Wafer Strip Detectors

Bradly J. Cooke, Klaus S. Lackner, Andrea P. T. Palounek, David H. Sharp, Larry Winter, and Hans-Joachim Ziock

Los Alamos National Laboratory
Los Alamos, NM 87545



Photograph: by Chris J. Lindberg

Efficient Data Transmission from Silicon Wafer Strip Detectors

Bradly J. Cooke, Klaus S. Lackner, Andrea P. T. Palounek,
David H. Sharp, Larry Winter and Hans-Joachim Ziock

Los Alamos National Laboratory
Los Alamos, NM 87545

Abstract

An architecture for on-wafer processing is proposed for central silicon-strip tracker systems as they are currently designed for high energy physics experiments at the SSC, and for heavy ion experiments at RHIC. The data compression achievable with on-wafer processing would make it possible to transmit all data generated to the outside of the detector system. A set of data which completely describes the state of the wafer for low occupancy events and which contains important statistical information for more complex events can be transmitted immediately. This information could be used in early trigger decisions. Additional data packages which complete the description of the state of the wafer vary in size and are sent through a second channel. By buffering this channel the required bandwidth can be kept far below the peak data rates which occur in rare but interesting events.

I. Introduction

The purpose of this paper is to propose an architecture which accomplishes significant data processing in the core of detectors under consideration for use at the SSC [1–3] and at RHIC [4].

The principal function of the central tracking system is to measure precisely the trajectories of particles generated in the interaction region. The detectors considered here are silicon strip detectors on silicon wafers. Track localization is accomplished by using a large number of small wafers ($\mathcal{O}(10^4)$). A wafer, however, is large compared to the localization required for each trajectory. Higher resolution in direction is achieved by partitioning each wafer into a large number of strips. In some designs each wafer consists of two layers of strips on the opposite wafer surfaces, offset by a small angle. In this case it is possible, at least for an isolated track, to determine the intersection point completely.

Even though the event rate in the wafers under consideration is high (up to 60 MHz), the number of hits per wafer per event is small, typically of order one or less. However, these low hit rates are interspersed with sudden bursts of high multiplicity (50 and higher) in jet events at the SSC [5, 6] and with central heavy ion collisions in the RHIC detector [4].

The raw data flux generated in these detectors can be reduced dramatically because most events are not of interest and can be discarded once recognized. This is accomplished with a trigger. Current designs do not utilize the information from the central track detector in the initial (level I) trigger decision because, in its unprocessed form, the data flux cannot be transmitted in the available time.

Making the information from the central core silicon tracker subsystem available prior to a level I trigger decision will have several advantages for the detector design. Most importantly, this information could add significantly to the efficiency of the level I trigger and thereby help to reduce the data stream at the earliest possible level. We consider a number of potential filters which are based solely on data from the central tracker. These are listed roughly in order of increasing computational effort.

- Very stiff jets. High multiplicity jets with narrow cross sectional areas can be extracted from single wafer information. A simple cross-check with other wafers on the same radial line can rule out an unusually large response of a wafer to a single track.
- Very stiff single particle tracks. Massively parallel road building algorithms [7] can be used efficiently to find stiff tracks of single particles.
- Tagging of back-to-back stiff jets and particle tracks.
- Tagging semileptonic events by correlating jets with nearby stiff single particle tracks, e.g. muons.
- Vertex reconstruction This should put tagging of heavy quark events within reach of the level I trigger system.
- Missing momentum. For a small number of very stiff jet or particle tracks it may be possible to register the presence or absence of large amounts of missing momentum.
- Fast detection of multiple beam interactions in a single event cycle by locating the primary vertices.

Another class of filters can be designed by combining silicon tracker information with data from other subsystems.

- Separation of large electromagnetic energy deposition in a calorimeter into photon and electron events. One can distinguish between these events by searching for the corresponding track in the silicon-wafer subsystem.
- Improving timing resolution. Many of the other detector subsystems lack the timing resolution necessary to identify uniquely the event cycle which generated a particular signature. As a result these event features make poor level I triggers. This situation could be significantly improved by making use of correlations with data from the silicon tracker system.

Finally we note that for some designs currently under consideration, direct readout of all data may be unavoidable. For example, if some of the descoped detector designs rely exclusively on the end caps of the silicon tracker for particle detection in the forward and backward direction, a data read out prior to the level I trigger decision would appear to be mandatory.

II. The Processing Architecture

Our scheme for preprocessing data on individual wafers in the detector core is based on the following ideas [8]. Data are separated into a prompt component and a delayed component. The prompt component which would be available for the level I trigger would contain statistical information such as the number of hits on the wafer and the centroid of the position of their distribution, or a coarse resolution map of all hits. For events with only a limited number of hits, the prompt channel may provide a complete description of the state of the wafer. The delayed component would carry any information needed to complete the description of the event.

The delayed channel is accessed through a queue of variable length. Hence it is not possible to predict the precise transmission time for delayed data. Low multiplicity events do not strain the delayed channel. High multiplicity events are sufficiently rare that they do not clog the channel. Prompt and delayed information are recombined on the outside of the detector to give a complete record of the event.

III. On-Wafer Processing at the SSC

A. Data Rates

Here and in the following sections, we discuss a concrete example which is based on the preliminary SDC design for a central silicon tracker. The analysis can easily be applied to other designs.

Based on a machine cycle of 16 ns, and a design luminosity of $\mathcal{L} = 10^{33}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$, the number of particle tracks crossing a single wafer 12 cm by 3.2 cm in area during one machine cycle can be estimated to be

$$N_{\mathrm{tracks}} \approx \frac{60\mathrm{cm}^2}{r_\perp^2},$$

with $r_\perp$ the distance from the beam-axis. Included are particle trapping in the 2 Tesla magnetic field as well as secondary tracks from photon conversions [9]. At a minimum radius of 9 cm one therefore expects 0.75 tracks per wafer per event cycle.

The double-sided wafers are divided on both sides into 640 strips. The orientations of the strips on the two sides differ by 5 milliradians allowing for some spatial resolution along the average strip direction. Each particle hitting a wafer will cause a small cluster of strips to respond on either side of the wafer. A simple description of the state of the wafer is given by the state of all strips. For simplicity

Fig. 1: Example of strip encoding. The long and short bars represent the state of the strips on a wafer. The sequence of numbers below the bars is transmitted. These numbers are the numbers of consecutive strips in the same state. Note that we include two "phantom" strips, one on each end, which are taken to be in the "off" state. These strips make it possible to identify unambiguously the state of the strips at the edge of the wafer.

we shall assume in the following that no statistical correlation between the front and the back data of the wafer is attempted and that for data compression purposes we treat the raw data as a string of single bits each reflecting the state of a single strip. The first half of the string reflects the state on one side of the wafer, the second half the state on the other side.

B. Bandwidth Requirements after Compression

To estimate the bandwidth requirements for transmitting the state of the wafer for every event cycle, we begin by estimating the data compression which might be achieved. Instead of sending the bit string representing individual strips we propose to send a string of numbers which represent the numbers of consecutive strips in the same state (Fig. 1). In effect we give the distances between consecutive cluster edges. For a low number of hits this will lead to an encoding which is significantly shorter than the 1280 bits required for the raw input. We can estimate the length of this encoding as function of the number of hits and the distribution of cluster widths. For a random distribution of hits with a hit probability of $\langle n_{\mathrm{hit}}\rangle = 0.75$ and a cluster width given by a distribution generated with GEANT [5, 10–12] (Fig. 2) we obtain the distribution of edge distances shown in Fig. 3. An encoding using $b = -\log_2(Prob(d))$ bits for the distance d requires an average $\langle b_{\mathrm{edge}}\rangle = 6.6$ bits for the encoding of a single distance. If distances were uncorrelated this would indeed be the optimum encoding. Considering that each hit generates two clusters, one on the front and one on the back of the wafer, a simple estimate yields $(4\langle n_{\mathrm{hit}}\rangle + 1)\langle b_{\mathrm{edge}}\rangle = 26.5$ bits for the average event which agrees with our numerical simulation. Doubling the average number of tracks on a wafer, for example due to an increase in luminosity, would lead to an optimal encoding using 6.8 bits per distance and 47.6 bits per event.

This analysis indicates that for a luminosity of $\mathcal{L} = 10^{33}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$ the average data transmission rate at optimal compression is approximately 1.7 Gbit/s compared to a raw data stream at 80 Gbit/s. This estimate sets a lower limit for the data transmission rate.

The bandwidth of the data channels must exceed the average data transmission rate for three reasons: statistical fluctuations in the hit rate, jets, and redundancy for error correction. There is a trade-off between excess bandwidt

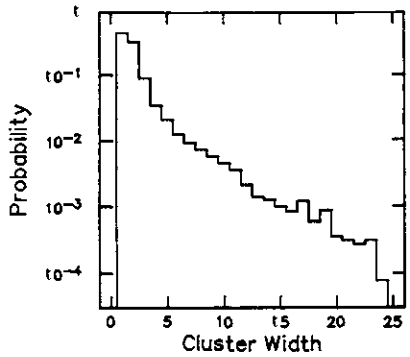


Fig. 2: Distribution of cluster widths.

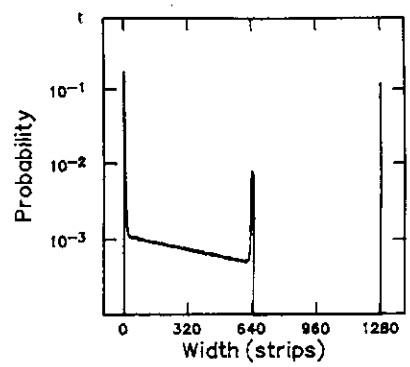


Fig. 3: Probability of occurrence of different widths of clusters and inter-cluster regions. We assume that the cluster widths are distributed as in Fig. 2 and that uncorrelated tracks hit a doubled sided wafer 640 strips wide. The hit frequency follows from a Poisson distribution with a mean of 0.75. The stereo angle of 5 milliradians is included in the Monte Carlo simulation. The narrow peak at $d = 1$ corresponds to the width distribution, the peak at $d = 640$ is caused by single track events which result in a cluster on the front and back side of the wafer approximately 640 strips apart. $d = 1280$ correspond to zero track events which constitute 47% of the total.

to process peak data rates and buffering requirements.

For practical reasons it is desirable to transmit the prompt information through a single fiber; a second fiber may be used for the delayed channel. For low occupancy rates, it is the fixed block size of this prompt channel which sets the minimum bandwidth of a single fiber. At higher occupancy it is the average transmission rate of the delayed channel which determines the bandwidth of a single fiber.

Given that the prompt channel is unbuffered, the bandwidth must allow for the maximum number of prompt bits per event cycle. At a minimum, the prompt channel must transmit the number of tracks in order to determine what is transmitted in the delayed channel. In addition, the prompt channel should transmit information which makes it possible to describe a single track completely, complete the description of two tracks on one side of the wafer, or give the center locations of up to four tracks on one side of the wafer. For more than four tracks the information given may be the centroid of the distribution, together with the total number of hits and the left most and right most edge. Alternatively, one could include a coarse grained description of the state of the wafer by performing a logical-or operation on blocks of 32 strips at a time thereby achieving a 20 bit summary of the state of the wafer. This information could be used for stiff track reconstruction prior to a level I trigger [7, 13] As can be seen from Table 1, this would require a block size of 44 bits leaving another 4 bits for error recovery in a 3 Gbit prompt channel.

Table I: Utilization of a Prompt Channel

0 Tracks	$1 + 3$ bits for N_{track}	4 bits
1 Track	$1 + 3$ bits for N_{1track} 4×10 bits for edges on both sides of wafer	44 bits
2 Tracks	$1 + 3$ bits for N_{track} 4×10 bits for edges on one side of wafer	44 bits
3 Tracks	$1 + 3$ bits for N_{track} 3×10 bits for cluster centers on one side of wafer 10 bits for total number of hit strips	44 bits
4 Tracks	$1 + 3$ bits for N_{track} 4×10 bits for cluster centers on one side of wafer	44 bits
≥ 5 Tracks	$1 + 10$ bits for N_{track} 10 bits for centroid position on one side of wafer 20 bits for coarse grained image of one side of wafer or for number of hits and width of distribution	41 bits

The prompt channel carries a large fraction of the information. Using an optimal encoding for all events with two or more tracks and leaving out the distance from the last cluster edge to the end of the wafer, which can be deduced from the number of tracks already transmitted by the prompt channel, one finds that the delayed channel has to carry 10.2 bits per event cycle. This is less than 40% of the optimally compressed data flow in the absence a prompt channel. In a 3 Gbit/s channel it leaves 38 bits per event cycle for the transmission of excess data from fluctuations in the data rate or from jet events.

The peak data rates are completely dominated by jets. The buffer size has to be sufficient to store the track information generated on a single wafer for one or two jet events. This requires storage for approximately 100 tracks [5, 6]. The time necessary to transmit the buffered information to the outside depends on the excess bandwidth. With an excess of 38 bits per cycle in the delayed channel, a buffer filled with two 50 track events can be emptied in 47 event cycles. This corresponds to 17.6 bits per track which is lower than for the average track because overlap between tracks reduces the number of clusters and because the average distance between clusters is smaller leading to a shorter encoding. Statistical fluctuations from the Poisson distributed background are significantly smaller and therefore do not change this estimate. The dependence of the minimum required bandwidth on the mean occupancy of the wafer is shown in Figure 4.

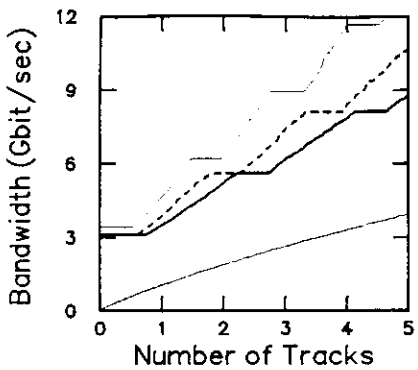


Fig. 4: The minimum required data bandwidth per channel as a function of the average number of tracks traversing a wafer in a single event cycle. The heavy line corresponds to an optimal design. The light lines indicate an upper and lower bound to this estimate. The lower limit is the mean compressed data flux evenly divided between the two channels: the upper limit corresponds to an encoding using 11 bits per edge plus an additional 10% redundancy for error correction. The dashed line is a practical compromise which corresponds to an encoding requiring twice the number of significant bits for each number. The prompt and delayed channel are assumed to have the same bandwidth. The prompt channel transmits the number of tracks, and for an event with a small number of clusters, it is used to transmit the complete state of the wafer. The bandwidth required for transmitting an event with n_c clusters through the prompt channel is given by $(40\,n_c + 10)$ bits per event cycle. The delayed channel is required to have an excess bandwidth of 2.5 Gbit or 40 bits per event cycle to cope with peak data rates. This would make it possible to transmit the edge information for 100 excess tracks in approximately 50 to 100 event cycles. The lower limit shown by a light line does not include any excess bandwidth to allow for fluctuations. For an event with more clusters than can be completely described by the prompt channel, the delayed channel transmits all the edge information leaving out only the last check sum which can be deduced from the information in the prompt channel. For the heavy line it is assumed that cluster edges are encoded using an optimal encoding scheme. Events completely described in the prompt channel are omitted from the delayed channel. The plateaus in the required bandwidth correspond to regimes in which the delayed channel is not saturated because the prompt channel carries nearly all the information. At each successive plateau, the bandwidth of the prompt channel has increased sufficiently to accommodate yet another track: consequently the bandwidth requirements on the delayed channel drop. The starting level corresponds to a prompt channel which can transmit complete information on a single track.

In practice one cannot expect to achieve the optimal data compression. A conservative estimate is based on a fixed length encoding for each distance requiring 11 bits per distance or edge. With this assumption, the delayed channel transmits on average 17.3 bits per event and has roughly 32 bits available to accommodate events with large numbers of tracks. It would take approximately 110 event cycles to transmit an excess of 100 tracks caused by 2 events with 50 tracks each. This corresponds to 35 rather than 44 bits per track because of track overlap.

A nearly optimal encoding can be achieved with two different fixed length encodings. For distances up to eight one uses a three bit encoding: for larger numbers an eleven bit encoding. In each case, there is an additional bit flagging the length of the encoding. This leads to an average bit rate in the delayed channel of 11.8 bits and would require only 50 event cycles for a pile up of 100 tracks to be transmitted to the outside.

IV. Example of an Encoding Algorithm

Here we present a simple encoding scheme which we believe can be implemented in hardware based on currently available technology. The encoding is carried out as follows [14]. The first step is edge detection. To detect an edge between two strips requires a single exclusive-or operation which can be performed in a single time cycle. The output is a vector of single bits, with one more element than the original strip vector. The data are encoded by counting the number of strips between any two edges in a serial fashion. Parallelization is achieved by processing up to twenty substrings in parallel and combining them in a final step. The output is comprised of two bitstreams, one representing the binary representation of the number of consecutive strips in the same state, the other of equal length is simply used to mark the beginning of each number in the first string. Thus, the resulting encoding requires twice the number of significant bits in a binary representation of the distances between cluster edges discussed in the previous section. As can be seen in Figure 4, it is noticeably better than a fixed-length encoding because the average number of significant binary digits turns out to be less than half the maximum required one. It requires more bandwidth than an optimal encoding, or an encoding using two fixed-length formats. The major advantage of this variable length encoding is that processing of the input stream always progresses from a longer encoding to a shorter one. No additional data storage has to be provided to deal with a number of closely spaced clusters as would be necessary in a fixed length encoding scheme. Furthermore, this encoding contains sufficient redundancy to allow for effective error correction without additional bandwidth.

By processing twenty segments of 64 strips each in parallel, the encoding should be completed in approximately 100 processing cycles. Even if processing cycles coincide with event cycles, this is fast enough to extract the prompt channel information after the compression and still arrive in time for use in a level I trigger. After the compression, the wafer state information is buffered for transmission through the delayed channel. This buffer is only required to hold about 100 tracks and is therefore, at 4000 bits, significantly smaller than the buffer requirements in a conventional design which can be estimated to be 300 kbit. Even including the data storage implicit in the data processing pipeline, 250 kbit, the effective memory requirement of this architecture is smaller.

In summary, with this choice of an encoding and a pair of fiber-optic channels operating in the 4 to 6 Gbit/s range, the SDC silicon tracker could operate without reliance on an external trigger and could itself contribute to the level I filter.

V. Discussion and Conclusion

We have demonstrated that the complete digital information generated on a silicon wafer for use in the SDC silicon tracker system can be compressed by as much as a factor of 40. An even larger data reduction could be achieved if on-wafer processing went beyond data compression by eliminating less critical information. For example, if a consensus could be reached that cluster width information is not important, simple algorithms could be used to generate the cluster centroid positions rather than cluster edges. (Extremely wide clusters could be encoded through multiple centroids, retaining some of the cluster width information.) This would reduce the data flow by another 20 to 30%.

The on-chip storage associated with the required processing is less than what is required for simply holding the data for $4\,\mu s$ to wait for a level I trigger. Hence, data compression would already facilitate conventional on-wafer data storage and data transmission controlled by an external trigger.

More importantly, the large data compression makes it possible to transmit all data generated in the silicon subsystem to the outside of the detector, obviating the need for an external trigger. The silicon tracker would reach its full potential by contributing significantly to the level I data filter. Because of the high quality of the tracker data, it would become possible to utilize heretofore inaccessible trigger signatures like secondary vertices from heavy quark decays.

By separating the data into a prompt and a delayed component, it is possible to guarantee a fixed arrival time for the most important information while at the same time allowing for significant buffering in the delayed channel, thereby compensating for fluctuations in the wafer occupancy. This separation comes at price; the effective compression for both channels achieved in our example is about a factor of ten. A variation of this architecture would lead to a hybrid data transmission for which the prompt channel would contribute to the level I filter, whereas the delayed channel would be controlled by it.

The bandwidth requirements for data transmission are large. Transmission rates of 3 to 6 Gbit/s for each of the two channels on a wafer require optical fibers and optical modulators. While optical modulators with bandwidths well above 6 Gbit/s have been reported [15–17], and radiation hardness of some optical modulators has been demonstrated [18], further research on the development of small, low power, radiation hard modulators is needed. An effort to design and develop the necessary hardware is currently underway and we have reported on its present status [14].

In spite of the demands on the technology, the benefits of a self-contained data transmission architecture appear to be well worth the effort. In particular, the ability to contribute to early data filtering would greatly enhance the utility of the silicon tracker subsystem and the SDC detector as a whole.

VI. References

[1] G. H. Trilling *et al.*, "Solenoidal Detector Collaboration Expression of Interest," SDC-90-00085, May, 1990.

[2] "Letter of Intent by the Solenoidal Detector Collaboration," SDC-90-00151, November 1990.

[3] GEM Collaboration, "An Expression of Interest to Construct a Major SSC Detector," SSC EOI-0020, July, 1991.

[4] John P. Sullivan, Jan G. Boissevain, Dan Fox, Barbara V. Jacak, Jon S. Kapustinsky, Michael J. Leitch, Patrick L. McGaughey, Joel Moss, Walter E. Sondheim & Hubert van Hecke, *The Vertex Dector for the Lepton/Photon Collaboration*, Los Alamos National Laboratory, 1991, (in preparation).

[5] W. W. Kinnison, D. M. Lee & H.-J. Ziock, "Monte Carlo Simulations of a Central Tracking Sub-System," in *Symposium on Detector Research and Development for the Superconducting Super Collider*, Tom Dombeck, Valerie Kelly & George P. Yost, eds., World Scientific, Fort Worth, Texas, October, 1990.

[6] J. DeWitt, D. Dorfan, A. Litke, H. Sadrozinski, A. Seiden & A. Weinstein, "Tracking Requirements for the SSC and Silicon Strip Development," in *Proceedings of the Summer Study on High Energy Physics in the 1990s*, Sharon Jensen, ed., World Scientific, Snowmass, Colorado, 1988.

[7] R. M. Farber, W. W. Kinnison & A. S. Lapedes, "A Parallel Non-Neural Trigger Tracker for the SSC," LAUR-91-607, 1990.

[8] Klaus S. Lackner, Vernon D. Sandberg & David H. Sharp, "Data Processing at the SSC with Structured Neural Nets," in *Symposium on Detector Research and Development for the Superconducting Super Collider*, Tom Dombeck, Valerie Kelly & George P. Yost, eds., World Scientific, Fort Worth, Texas, October, 1990.

[9] A. Seiden, "Tracking Strategies for Magnetic Detectors at High Luminosity and High Energy," Santa Cruz Institute for Particle Physics, University of California, Santa Cruz, SCIPP 91/17, May, 1991.

[10] R. Brun, F. Bryant, M. Maire, A. C. McPherson & P. Zanarini, "GEANT 3," CERN Report, DD/EE/84-1, 1984.

[11] C. Adolphsen, R. G. Jacobsen, V. Lüth, G. Gratta, L. LaBarga, M. Litke, A. S. Schwarz, M. Turala, C. Zaccardelli, A. Breakstone, C. J. Kenney, S. I. Parker, B. A. Barnett, P. Dauncey, D. C. Drewer & J. A. J. Matthews, "The Mark II Silicon Strip Vertex Detector," SLAC-PUB-5443, August, 1991.

[12] W. C. Sailor, H.-J. Ziock, W. W. Kinnison & K. Holzscheiter, "A model for the performance of silicon microstrip detectors," *Nucl. Inst. Meth.* (1991), (to appear).

[13] Richard Nickerson, Oxford, private communication, 1991.

[14] B. J. Cooke, K. S. Lackner, C. L. Winter & H. Ziock, "Superconducting Super Collider Data Compression and Driver/Modulator Architecture," 1991, (These proceedings).

[15] C. M. Gee, G. D. Thurmond & H. W. Yen, "17-GHz bandwidth electro-optic modulator," *Appl. Phys. Letr.* 43 (1983), 998–1000.

[16] R. L. Jungerman, C. Johnsen, D. J. McQuate, K. Salomaa, M. P. Zurkowski, R. C. Bray, G. Conrad, D. Cropper & P. Hernday, "High-Speed (20 GHz) Optiocal Modulator for Application in Instrumentation," *J. of Lighwave Tech.* 8 (1990), 1363–1370.

[17] High-Speed 8:1 Multiplexer and 1:8 Demultiplexer Implemented with AlGaAs/GaAs HBT's, R. B. Nubling, James Yu, Keh-Chung Wang, Peter M. Asbeck, Neng-Haung Sheng, Mau-Chung F. Chang, R. L. Pierson, G. J. Sullivan, Mark A. McDonald, A. J. Price & A. D. M. Chen, *IEEE Journal of Solid State Circuits* 26 (1991), 1354–1361.

[18] Mark Lowry, LLNL, private communication, 1991.