A TRIGGER PROCESSOR FOR ARGUS

H.D. Schulz and H.J. Stuckenberg

Deutsches Elektronen-Synchrotron DESY, Hamburg, Germany

ABSTRACT

The ARGUS detector at the electron-positron storage ring DORIS consists of various particle detectors arranged in cylindrical symmetry in and around an axial magnetic field. A fast secondary trigger processor has been designed to find and count circular tracks in the r-ϕ-plane of the detector that originate from the interaction point. The processor serves as a filter in the trigger system of the ARGUS detector and is expected to reduce the rate of background triggers by two orders of magnitude. The processor hardware is built in ECL to perform simple operations with very high speed and partly in parallel. The track finding process takes about 9 μs plus 3 μs for each encountered track element. Control information is stored in memories that may be loaded via CAMAC from the online computer allowing easy adaption of the processor to different experimental conditions.

INTRODUCTION

At the 5 GeV electron positron storage ring DORIS in Hamburg a new third generation 4π-detector is set up which is called ARGUS. It consists of a large solenoid magnet, centered on the interaction point and filled with a series of detectors and is designed to track and identify the particles generated from a positron electron collision. As a particle leaves the interaction volume it passes through the following detector parts (see Fig. 1) :

(1) The beam pipe

(2) A large cylindrical drift chamber consisting of 36 layers of drift cells. Eighteen of these have their signal wires spanned parallel to the cylinder axis (0^o-wires) and 18 are stereo layers. The number of cells in a layer varies from 60 on the inside to 264 on the outside.

(3) A layer of scintillation counters for time-of-flight measurements (TOF-counters).

(4) A layer of lead-scintillator-sandwich shower counters.

(5) The magnet coil and the iron yoke used as hadron absorber.

(6) A layer of proportional chambers to detect muons.

Some particles will emerge at small angles to the beam axis. These pass through only some of the cylindrical drift chamber layers and then through additional TOF and shower counters in the cylinder endcap region.

The ARGUS trigger will be driven from two sources :

1) The total energy trigger is derived from the total energy deposited in the shower counter system. Its decision time fits well within the 1 μs time between bunch crossings at DORIS and does not produce any deadtime.
2) The charged particle trigger fires, if a minimum number of charged tracks is coming from the interaction volume into the detector. Detection of tracks in ARGUS implies pattern recognition at least within the r-ϕ-plane of the drift chamber. Since this is not possible to achieve within the DORIS bunch-to-bunch time of 1 μs, the charged particle trigger utilizes a two step decision process.

The first step decision or primary trigger is derived from fast coincidences between adjacent groups of TOF and shower counters. It decides at each bunch crossing time within less than 0.5 μs whether an event occurred which is of further interest. It is sensitive to local track elements far from the interaction point and its rate is estimated to vary between 100 s^{-1} and 1000 s^{-1}. With an average readin time of an event into the online computer of 50 ms these rates are at least two orders of magnitude higher than can be handled by the data acquisition and analysis system.

A second step decision is essential. It is made by an electronic processor called 'Little Track Finder', the design and performance of which is the topic of this paper. The track finder is started by the primary trigger and uses data from the 0^{o}-drift wires and the TOF counters to find and classify circular tracks in the r-ϕ-plane of the ARGUS detector. This process takes more than the time between bunch crossings and some deadtime is introduced into the detector performance. With a maximum primary trigger rate of 1000 s^{-1} and allowance for 5% deadtime an acceptable processing time for the track finder is 50 μs. Within this time the track finder logic is expected to reduce the number of accepted events to about 1 s^{-1} that can be reasonably handled by the data acquisition and analysis system.

The task of the trackfinder is now well defined :

1) To identify circular tracks from the interaction volume in the r-ϕ-plane of the detector by looking into data from the drift chamber and the TOF counters.
2) To count these tracks and make a trigger decision within about 50 μs.
3) To store all information collected about the tracks of an event for possible later use by an additional software filter in the online computer.

Problems of a similar kind have been encountered at many large detectors in the past and have found various solutions [1)], ranging from fully parallel hard wired pattern recognition down to fully sequential programmed analysis, the former being extremely fast but inflexible, the latter being most flexible but slow. In our case we decided to make a fair compromise between speed and easy adaptability to varying experimental conditions. We designed a processor that works partly in parallel and partly sequential under software control. The intelligence should be in memories rather than in cabling.

PRINCIPLE OF OPERATION

The design of the trackfinder is strongly influenced by the properties of the ARGUS drift chamber. This is characterized by a constant high density of rather small (18x18 mm^2) drift cells. Thus a very good track definition is already obtained from just using the wire hits as such, and ignoring the drift time information. For instance the angular resolution for straight tracks in ϕ is then still 0.7^o. The hit information is easily available from the LeCroy TDC modules 4291B which are connected to the ARGUS drift wires. They contain the information whether a wire had a hit during the sensitive time interval in so called hit registers which can be accessed via a backplane connector.

The basic idea of the track finder is the following :

- Transfer the hit information of selected layers of the ARGUS drift chamber into bitwise addressable memories to make it available to sequential data processing. This transfer is done in parallel strobed with each primary trigger.
- Have another memory filled before the experiment starts with a preestablished list of all possible good tracks. Each entry in this list consists of a sequence of wire numbers, one number for each layer, which together describes a possible good track in the detector. Such an entry is called a mask.
- Have a processor that sequentially compares the entries in the mask list with the contents of the wire memory each time a primary trigger occurred. The comparison is done in parallel for the layers participating in a mask. If there is a match detected between a mask and the wire hit pattern a track counter is incremented. In case the track counter exceeds a preset threshold the secondary trigger is set. If the list of good tracks is exhausted and the track counter did not exceed the threshold a reset command is sent to the detector.
- One will notice that each wire participates in many possible tracks. This fact can be used to considerably speed up the search of the track finder. Once it is detected that a wire has no hit, it is of no use to check on all the rest of the masks containing also this same wire. A clever incrementation scheme for the mask address will skip over masks containing this same wire. This feature of the track finder is the key point for a fast sequential search and saves a factor of 10 in the processing time.

In the actual design of the track finder some features had to be added to the basic idea.

Because the number of masks has to be finite and should be preferably small, one mask must cover a certain region of good tracks in space and momentum as shown in Fig. 2. This region extends well over a few neighbouring cells in some layers. Each mask list entry therefore does consist not only of one wire number per layer, but to each wire number are added 3 bits which contain the number of consecutive wires that have to be ORed together to cover the mask region completely. The processor hardware must correctly execute this OR-prescription in the comparison.

Instead of asking for a full coincidence between mask and wire hits it is sometimes advantageous to define a match differently, e.g. to allow for one layer to be missing to account for a certain chamber inefficiency. There is quite some flexibility in the definition of a match in the track finder since up to 64 alternative patterns for this definition can be loaded into one of its memories.

To avoid double counting of one track which passes through cell boundaries or which together with some random hit would give rise to two matches, an 'antichatter interval' can be defined such that from subsequent matches the second one or even also the third one will not be recognized.

The described principle of the track finder can be extended to cover also tracks that leave the detector through the endcaps. The necessary minor extensions to the logic will be implemented but they are not described in this paper.

SEQUENCE OF OPERATIONS

A block diagram of the processor is shown in Fig. 3. The main components have already been introduced above. One notices that no CPU is present. Indeed the actual processing unit is not much more than a 50 Mc clock that controls memory cycles and some simple logic circuits.

Let us go briefly through the sequence of operations.

Prior to the start of an experimental run the track finder registers are loaded via CAMAC from the online computer of the experiment. The data sets that are loaded to the mask memories (MMB), the increment memory (IM) and the contents addressable memory (CAM) of the track finder are prepared by a special mask generation program that runs on a large computer. The simple inputs to this program represent the physical requirements of the experiment : Properties of the layers connected to the track finder, size and ordering of masks, definition of good tracks, and the trigger threshold. The control information loaded into the track finder remains unchanged during an experimental run.

Each primary trigger will initialize the track finder and will strobe the contents of the TDC-hitregisters into the wire memory boards (WIB). At the same time the mask address register (MAR) points to the first mask on the MMBs. There is one MMB for each layer. Three bits of one mask memory word define the WIB of the layer, one out of eight, that is to be addressed. Another 8 bits are the address of a 256 word x 32 bit-PROM. Logically 5 of the 8 bits describe a base wire number on the WIB and the 3 other bits contain the number of consecutive wires that have to be ORed together with the base wire. The PROM decodes this information by activating those of its 32 output lines that are described in the 8 bit input code. The output lines of the PROM strobe the contents of the addressed wire memory cells into one bit of the CAM-register (CAMR) thus effectively performing an OR over the addressed cells.

The problem of ORing drift wires across WIB boundaries has been solved by adding an 'overflow bit' to the mask information. If set, this bit activates via some simple gate logic the proper lines on the next WIB also. There is also the problem of ORing wires accross the 2π-boundary of a layer, that is from the highest to the lowest wire numbers of a layer. This is solved by filling unused inputs following the last wire on the last WIB of a layer in parallel with the first wires of the first board, thus achieving the necessary wrap-around.

After the WIB gating the CAMR contains the hit pattern of the first mask, each bit corresponding to one layer of the detector. This hit pattern is now compared to the predefined patterns of good tracks stored in a 64 word CAM. If there is a match the track counter TC is incremented and the contents of the MAR as well as the hit pattern from the CAMR are loaded on their stacks MST and HST, respectively. If the track counter exceeds the value in the predefined track threshold reigster the secondary trigger output enables the readin of the event into the online computer.

Three predefined consecutive bits from the CAMR and the 9 least significant bits from the actual MAR value are taken to fetch the new mask address incrementation value from the IM. (To limit the size of the IM, use is made here of the 4-fold symmetry of the ARGUS central detector.) The increment value is added to the actual MAR, thereby starting a new mask cycle.

Essential for the function of the MAR incrementation scheme is a systematic ordering of masks in the mask list. This list is logically divided into groups. Each group is characterized by the same wire number in some reference layer. There is also a subdivision within each group according to the same wire numbers in two layers adjacent to the first reference layer. The three CAMR-bits from the first reference layer and its two adjacent layers are used to control the MAR incrementation. For simplicity, let us assume that an acceptable track should have a coincidence in all layers. If then the first mask of a group is checked with the result that the reference layer has no hit one may skip over the rest of the masks of the group since no match is possible with any of them. If the reference layer is hit, but the next layer is not, then there may be a skip over a subgroup and so on with the third layer. The calculated values in the IM correspond to the ordering of masks and ensure the proper incrementation values. If, due to inefficiencies a majority coincidence defines a track, the algorithm describing the incrementation steps as a function of the response of the reference layers is somewhat different, but the principle is still applicable and very useful.

The mask cycles continue until the MAR contents exceeds a preset maximum value, that is until the mask list is exhausted. If the trigger threshold has not been reached up to now, a reset signal is sent to the detector electronic. If the threshold had been reached a LAM is generated to inform the online computer that the data block ocntaining the track information TC, MST and HST is ready to be read via CAMAC. These data can be used further in some software filter routine or will be added to the event record.

PERFORMANCE

The time needed for one mask cycle is 150 ns. The processing time of an event depends on the number of track elements that are encountered in the three reference layers during the mask cycles, since they determine the incrementation steps of the MAR. There is a free choice of the position of the three adjacent reference layers within the detector, since only the ordering of masks by software is involved. The choice will be made in such a way that the processing time is minimized with the actual background conditions taken into account. The processing time of an event is made up of two parts. The first part is constant and independent of whether any wires have been hit. This is the time required to check in each mask group whether the reference layers have hits which would mean that the group deserves closer inspection. With 60 cells in the reference layer this gives 60 mask cycles or 9 μs of basic processing time. The second contribution depends on how often hits are encountered in the reference layers, inducing the track finder to step through the respective mask group in detail. With 20 masks in a group it takes 3 μs to step through it. Thus with a total number of 1200 masks the processing time of an event amounts to 9 μs basic time plus 3 μs per each encountered track element in the reference layers. From Monte-Carlo simulations of background tracks we got a typical processing time of about 20 μs which is very encouraging.

Additional computer simulations of the track finder response to Monte Carlo generated tracks have shown the properties of different mask sets. Assuming drift chambers with 100% efficiency we calculated the track detection efficiency to be 100% for tracks which approach the nominal interaction point closer than d = 3 cm and with transverse momentum larger than 130 MeV/c. This efficiency then drops almost linearly from 100% to 0% between d = 3 cm and d = 5 cm. These numbers apply to the barrel region of the ARGUS detector, that is to 75% of 4π. After extension of the track finder to include also the endcap region, the accepted solid angle will rise to 91% of 4π.

The effect of a drift chamber inefficiency on track detection can be minimized by allowing one layer to be missing on a good track. For a chamber inefficiency of 1% and 10 selected layers in a track the tracking efficiency rises from 90% to 99%.

At the same time the introduction of an 'antichatter interval' of 2 (see above) reduces the probability of counting a single track twice from 5% to 0.3%. Further improvements are possible by widening the 'antichatter interval' to ignore 2 of 3 subsequent matches or by using an additional software filter in the online computer.

PACKAGING

The track finder logic is housed in three CAMAC size crates with special bus connections between them. The logic is done in ECL to obtain high speed operation. The power consumption of the device is 1 kW.

The track finder consists of 4 types of modules :

1) The wire input boards (WIBs)
2) The mask memory boards (MMBs)
3) The CAM memory boards (CAMs)
4) The logic boards

All WIBs are identical having 32 wire inputs each. Up to 8 WIBs are grouped together to cover all inputs from one detector layer. To each layer belongs one MBB, all of them are identical containing RAMs of 2k x 12 bits to store the mask information. There are 12 groups foreseen so far with a total of 12 MMBs and 54 WIBs. Together with the two CAM-boards these modules fill 90% of the crate space.

The actual logic boards take only 3 slots of the crates. The logic accomodates up to 16 detector layers. Since only 12 are used until now there is more space for future additional layers, for instance beam pipe proportional chambers and other detector components.

The connection of the track finder to the CAMAC branch is accomplished by a pair of modules, one residing in the track finder crate, the other in a normal CAMAC crate. The track finder looks like a simple CAMAC address to the computer.

TESTING

Easy and self-contained testing facilities are of primary importance for any trigger processor. To test the track finder alone it is possible to disable the inputs from the TDCs to the WIBs by a CAMAC command and to access with read and write all registers of the track finder from the online computer via CAMAC. This includes also the WIBs which can be written with predefined hit patterns to test the response of the track finder. Also the clock can be cycled under software control to follow the mask cycle in detail. These features proved to be extremely helpful during the checkout procedure of the device.

Another type of testing is the verification of the quality and performance of the software datasets that are loaded into the memories of the track finder. These tests can be easily and thoroughly done by simulating the track finder hardware on a large computer and by studying its response with different input data.

STATUS

All parts of the track finder have been produced by the firm Dr. B. Struck, Tangstedt and have been delivered to DESY. The system test is under way and has been quite satisfactory so far. An extension of the track finder to also detect tracks that leave the detector through the endcaps has been specified and will be ordered soon.

SUMMARY

The properties of the trigger processor for ARGUS can be summarized in a few statements:

- The processor finds and classifies circular tracks in the r-ϕ-plane of the ARGUS drift chamber by looking for wire hits in the 0^o-drift-layers. It is sensitive to particles from the interaction region only.
- It is expected to reduce the trigger rate by two orders of magnitude to a few per second.
- The processing time for simulated background events with a total number of 1200 masks defined is of the order of 20 µs. This introduces less than 5% deadtime into the detector performance.
- The track detection efficiency is 100% for fully efficient drift chamber and for charged particles with transverse momentum larger than 130 MeV/c if the particle approaches the nominal interaction point closer than 3 cm. The efficiency drops to zero for particles with a distance larger than 5 cm from the interaction point.
- Even with a drift chamber inefficiency of 1% and 10 drift layers connected to the track finder the track detection efficiency can be 99% for particles from the interaction region.
- At the same time the probability to count a single track twice can be kept lower than 1% by sacrificing to resolve parallel tracks close to each other in space.
- The track finder can be easily adapted to changing experimental conditions by differently loading its memories.
- The track finder can be easily and thoroughly tested from the online computer via CAMAC.

The combination of serial and parallel gating logic to check more than 1000 masks in a time of about 20 µs by comparing the address of hits to the stored library of possible track coordinates seems to be an excellent compromise between speed and cost. Essential for the speed of the device is the use of a precalculated skip logic which shortens the effective processing time by an order of magnitude. Very new developments of fast ECL bitslice processors are providing the chance to use such processors in future applications of our design.

ACKNOWLEDGEMENT

The authors wish to thank Dr. W. Schmidt-Parzefall for stimulating discussions and Dr. E. McCliment, who did early work on the computer simulation. Thanks are also due to H. Quehl from DESY and R. Ölschläger from the firm Dr. B. Struck for their excellent support in the construction of the device.

REFERENCES

[1] see for instance the review talk of
P. Waloschek, International Conference on Experimentation at LEP, Uppsala, 16-20 June 1980, published as DESY 80/114, Nov. 1980, or
G. Franke, "Triggers and Decision Processors", DESY 80/109, Nov. 1980.

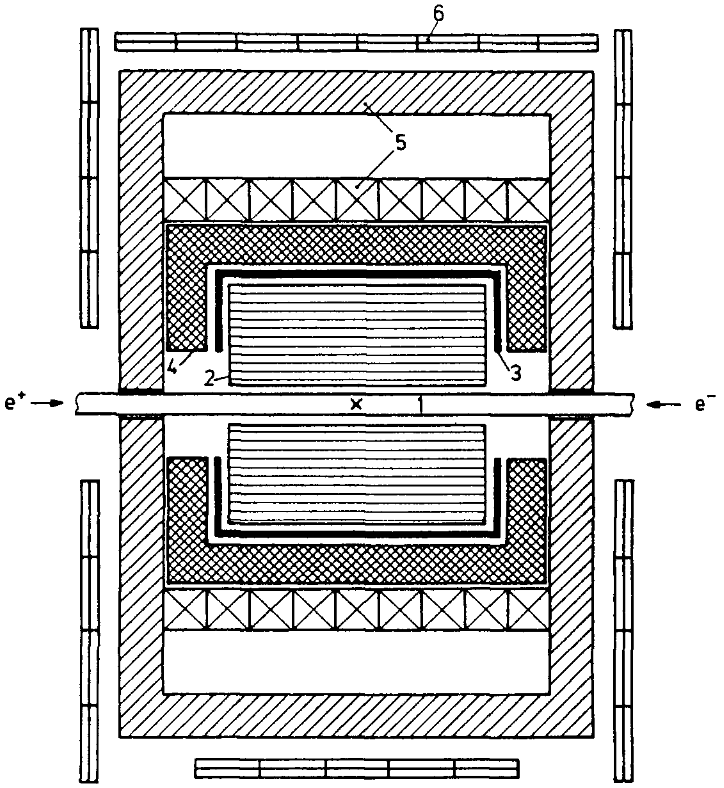

Fig. 1 ARGUS detector (schematic)

(1) Beam pipe
(2) Drift chamber
(3) Time-of-flight counters
(4) Shower counters
(5) Hadron filter (magnet coil and yoke)
(6) Proportional chambers for muon detection

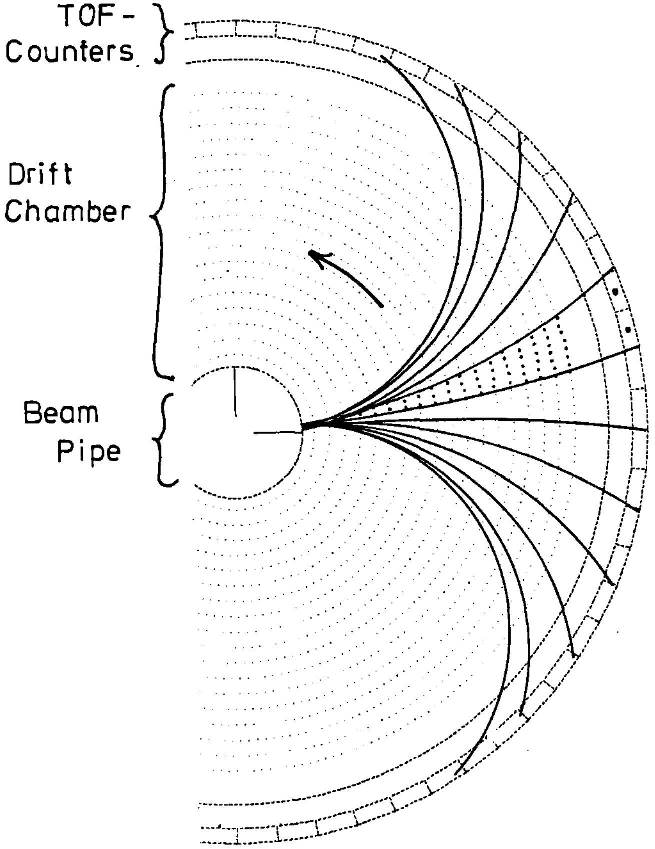

Fig. 2 Definition of masks

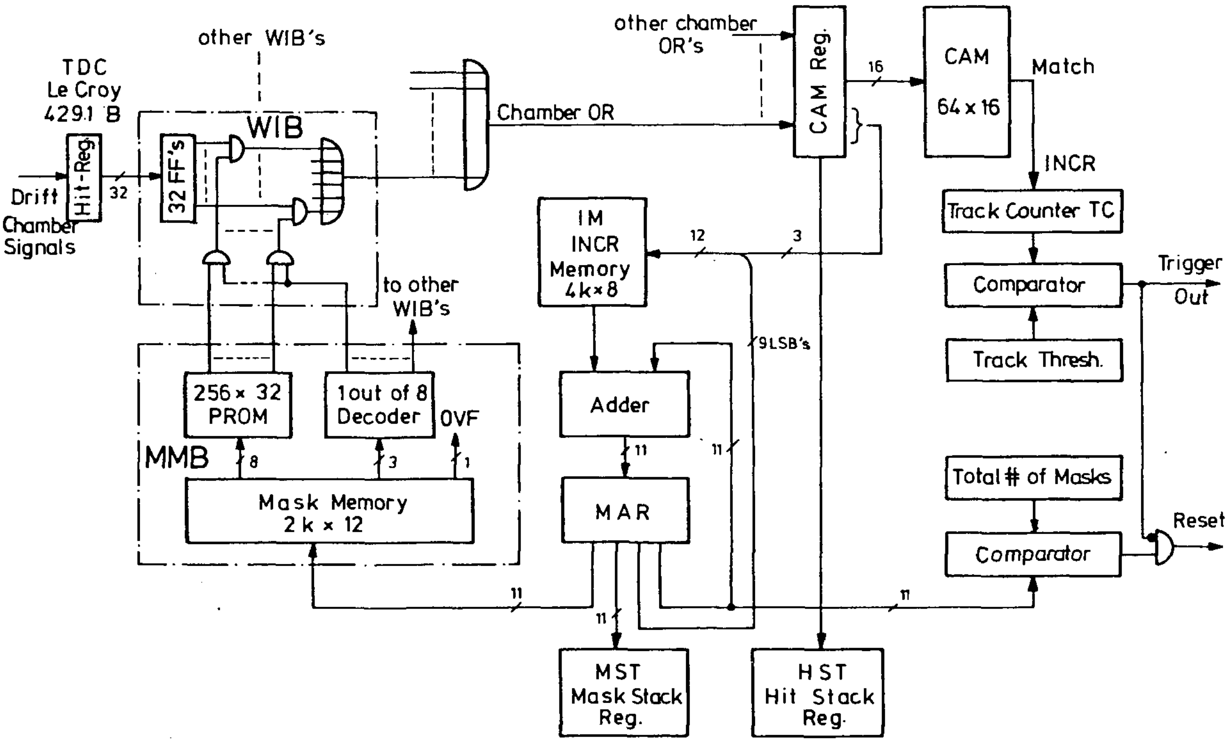

Fig. 3 Block diagram of track finder