A FAST PROCESSOR FOR DI-LEPTON TRIGGERS

P. Kostarakis, and S. Katsanevas
University of Athens,
Solonos 104
Athens, Greece

E. Barsotti, B. Cox, J. Enagonio, M. Haldeman,
W. Haynes, C. Kerns, H. Smith, T. Soszynski, J. Stoffel,
K. Treptow, F. Turkot, and R. Wagner
Fermi National Accelerator Laboratory
P.O. Box 500
Batavia, Illinois 60510 USA

H. Areti, S. Conetti, and P. Lebrun
Mc Gill University
Montreal, Quebec, H3A 2T8, Canada

As a new application of the Fermilab ECL-CAMAC logic modules, a fast trigger processor was developed for Fermilab experiment E-537, aiming to measure the high mass di-muon production by antiprotons. The processor matches the hit information received from drift chambers and scintillation counters, to find candidate muon tracks and determine their directions and momenta. The found tracks are then paired to compute and invariant mass: when the computed mass falls within the desired range, the event is accepted. The whole process is accomplished in times of the order of 5 to 10 microseconds, while achieving a trigger rate reduction of up to a factor of ten.

INTRODUCTION

A general purpose trigger processor system[1] was developed at Fermilab as a means of filtering events in high rate experiments, prior to acquisition by an online computer. The system consists of a set of general utility logic and arithmetic modules built of fast ECL integrated circuits. The modules resides in CAMAC-like crates with a backplane modified to carry ECL singnals. Each crate communicates with the host computer via a standard A-1 CAMAC crate controller. A CAMAC-ECL translator module sits in station 23 to convert the CAMAC signals to ECL and transmit them to the rest of the crate.

Individual modules are linked to each other via front panel connectors carring ECL differential signals on twist-and-flat ribbon cables or on twisted pair 2-pin LEMOs. This feature allows easy reconfiguration of a system or redefinition of the function of individual modules. Without describing every single module, we will recall that one of the key components of the system is the Memory Lookup Unit (MLU). This module consists of 16 X 4K (or 1K) ECL RAMS (∼ 60 nsec access time), which are used to store pre-computed tables of logical or arithmetical functions, allowing the fastest possible computational capability. Apart from its modularity, one of the main features of the system is its asynchronous mode of operation: each module is activated by a set of INPUT READY lines, and produces its own OUTPUT READY, allowing parallel processing and flow optimisation of activities requiring different times to be completed.

THE EXPERIMENT

Fermilab experiment E-537[2]   (Athens-Fermilab-McGill-Michigan-People's Republic of
China collaboration) was the second experiment to employ the ECL-CAMAC processor system.
As a proof of the versatility of the original design, in this new application we employed
the standard set of modules with the addition of a few general purpose modules[3]
Two test modules were designed for the experiment in order to simulate the real-time
flow of data in the form expected by the processor.

The experiment, shown in Fig. 1, measures the production of high-mass di-muons by
pions and antiprotons. A negative beam enriched in antiprotons[4],   tagged by Cerenkov
and scintillation counters and momentum analyzed by multiwire proportional chambers,
strikes a nuclear target. This target is followed by a 60" copper absorber: reaction
products surviving the absorber are tracked and momentum analyzed by a set of drift
chambers in front and in the back of a large aperture magnet. Further tagging is provided
in the vertical and horizontal direction by a set of scintillation counters (charged
particle hodoscope CPX and CPY) placed behind the last drift chamber. Finally, the
particles hit a set of two iron and one concrete walls. Behind each wall a plane of
scintillation counters detects penetrating particles: a muon is defined as a triple
coincidence of three aligned counters, one from each plane. The low level trigger requires
a beam particle in coincidence with at least two hits in the CPX hodoscope and two muon
signals originating from two different quadrants.

In addition to the spurious triggers due to halo,decay muons and hadron punch-through,
a large class of unwanted events arises from the spectrum of low mass dimuons ($M_{2\mu} < 3$ Gev),
which dominates the dimuon cross section. Most of the schemes that can be devised to
depress such contribution have the undesired property of decreasing as well as distorting
the acceptance of the high mass events: A true dimuon mass computation is the most
desirable way of filtering the trigger. The no loss and no distortion requirements are
particularly important since the experiment is intended to make the absolute measurement
of a small cross section. The trigger processor scheme for the experiment was designed to
provide a fast, good resolution calculation of the di-muon mass.

GENERAL SCHEME OF THE PROCESSOR

The processor's view of the experiment is shown in fig. 2: The three drift chamber
X planes behind the magnets plus the CPX,CPY and mu triple coincidences are used by the
processor. A list of the relevant parameters of these detectors is given in table 1.

The invariant mass of a two particle system is approximately given by: $M \simeq (p_1 p_2)^{1/2} \theta$
with p_1 and p_2 the momenta of the two particles and θ their opening angle at production.
The processor uses the information of the drift chambers x planes behind the magnet to
find tracks and compute their slopes and intercepts at the magnet's center. This
information allows to compute the bend angle (inversely proportional to the momentum),
making use of the bend plane and point target approximations. The combined values of
the two intercepts at the bend plane uniquely determines the opening angle θ_x in the
xz plane. The drift chambers do not provide any direct measurement in the yz,non bend,
plane (each drift chamber module contains x,u and v planes, withstereo angle 16.7 degrees).
Since a computation of the θ_y opening angle via the chamber's information would have
required a costly duplication of the track-finding subsystem for the u and v planes, or
a time consuming sequential processing of tracks from the different planes, it was

decided to make use of the CPY hodoscope hits (fig. 2b) to determine, with sufficient accuracy, the opening angle θ_y for each di-muon candidate in the non-bend plane. The computed momenta and angles are combined to calculate the value of the di-muon mass.

An important concept in the design of the processor was that of "roads". For every possible (out of the total of 60) triple coincidence mu signal, it can be recognized that, if the track is originating from the target and its momentum is consistent with the one of the dimuons accepted by the spectrometer, only a subset of the drift chamber's wires are of interest for the track search. As detailed in the next section, the system was designed so that for every triple coincidence signal (road), only the hits contained in the appropriate region in the chamber were transmitted to the processor, thus minimizing the time required for the track search. Such a feature was even more important in view of another requirement we imposed on the processor. In order for the total efficiency of the track finding not to be a strong function of individual chamber inefficiency, (fig. 3), we chose to define a candidate track as given by only a pair of hits in two separate x planes. This weak track requirement would then be strengthened by asking for a confirmation from the CPX and the mu triple coincidences: the first and last chambers would be searched for tracks first and, in case of failure, the other two chamber pair combinations would be tried.

To perform a track search over the whole chamber, in the presence of spurious hits badly degrades the time performance of the processor, since the track finding time is proportional to the product of the number of hits in the two chambers.

The imposition of the roads requirement results in a considerable speeding-up of the process.

INTERFACE TO THE PROCESSOR

A special interface, whose block diagram is shown in Fig. 4, was required to organize the data from the detectors in the desired form and to transmit it to the processor. The interface is centered around the "Unit Boxes": each of these modules receives the signals from up to 64 data sources (drift chamber wires or scintillation counters) together with a trigger gate. Input levels that are present simultaneously with the arrival of the gate are recorded in a first bank of 64 latches. These latches are connected through a set of gates (each gate controlling 4 lines) to a second bank of latches, whose outputs are funneled into a priority encoder. The sequence of operations is shown in Fig. 5: For every low level fast trigger generated by the NIM logic, the drift chamber wires and scintillation counters that were hit are recorded in the Unit Boxes (two different types of Unit Boxes were made for chambers or counters, differing in the accepted input level, ECL or NIM respectively). Simultaneously, the signals from the mu triple coincidencies are presented, together with the trigger gate, to the "Road Box", which contains a single set of 64 latches connected to a priority encoder. The outputs of the Road Box encoder are sent as an address to a set of "Memory Boxes", each consisting of 3 banks of 256 x 16 RAMs, loadable through CAMAC by the host computer. A first "petition" for data is announced by the same trigger gate, (suitably delayed) sending a "PEAT" signal to the Memory Boxes: This causes a 3 x 16 bit pattern, the one corresponding to the road address originating from the Road Box, to appear at the output of the Memory Boxes. The patterns are sent to the Unit Box gates, causing only the hits contained within the road to be transmitted from the first to the second latch bank in

the Unit Boxes. The contents of these latches are then sent, through the priority encoder, to the trigger processor's Stacks. The flow of data is controlled by the Unit Clock Boxes which, in addition to combining the data from up to 3 Unit Boxes, also contain an adjustable clock, to be matched to the acceptance rate of the Stacks (~10 MHZ). As soon as the data starts arriving, the actual processor examines it in search of good tracks; if none is found, the processor sends, through the Road Clock Box, a "REPEAT" signal to the Memory Boxes. The same data is retransmitted to the processor, which this time analyzes it looking at a different chamber pair. The third REPEAT (or the first one if only two chambers had data within the current road) are transformed by the Road Clock Box into an ADVANCE signal to the Road Box; this causes the next road to appear at the Road Box output, and the corresponding bit pattern to arise from the Memory Boxes. A subsequent PEAT signal (generated by the Road Clock Box as a delayed ADVANCE) activates the processing of this second road. ADVANCE requests can also be generated directly by the Road Clock Box when less then two chambers with data are present, or by the processor itself, whenever good tracks are found in given road. The process continues until all roads (in general only 2, due to the 2 mu requirement in the trigger) are exhausted.

THE ECL-CAMAC PROCESSOR

A simplified block diagram of the E-537 trigger processor is shown in Fig. 6. A more detailed description of the modules appearing in the diagram and mentioned in the following text can be found in Ref. 1. The first module encountered by the drift chamber data flowing from the Unit Clock Boxes is a Router. This unit serially receives in its three inputs the data from the three drift chambers and only transmits two, in order to cycle through the three possible 2-chamber combinations. The switching of inputs is controlled by a two-bit pattern generated by the Road Clock Box. The data from the Router is fed into two separate Stacks, which are interrogated by a Do Loop Controller. All possible two-hit combinations are sequentially extracted from the Stacks and sent to the Track Finder, a simplified version of the one described in Ref. 1. The track slope and intercept computed by the Track Finder logic are sent to a sequence of two 1K Memory Lookup Units and one Arithmetic Logic Unit. These modules have the task of correcting the track parameters to account for the chamber pair used, project the track back to the magnet's center, and provide veto bits to signal slope too steep or track missing the magnet aperture. Non-vetoed tracks, defined by their slope and their intercept at the magnet's center, are stored into a set of two Stacks, able to record up to 64 candidate tracks. Units for the track parameters are wire spacings (3/4"), so as to match the precision contained in the input data. The candidate track Stacks act as an intermediate buffer that allows the process of track finding and track verification (the next step of the processor) to occur simultaneously. While the track finding loop proceeds to the next pair of hits, the found track is read from the stack and sent into three parallel channels, where the track validity will be examined. In the first channel a 4K MLU computes the bending angle of the track (inversely proportional to the track momentum) and corrects it for energy loss in the absorber. A veto flag will be set if the computed momentum is less than 4 GeV (at least 5.5 GeV are required for a track originating from the target to reach the last plane of mu counters). In the next two channels 2 4K MLUs are used to extrapolate the tracks to the position of the mu plane and the CPX hodoscope respectively. The track is accepted if it points at the right muon

counter (whose value for the current road is available from the Road Box), and to a CPX counter that had a hit in the current event. This last check is performed by presenting the computed CPX counter number to a Hit Array, in which a bit had been set for every bit transmitted by the CPX Unit Box. As the total number of bits defining a good track exceeds 16, tracks are stored in a set of two Stacks. The quantities recorded are the track inverse momentum, its intercept at the magnet center, a two-bit number identifying the quandrant to which the track belongs (to be used in the vertical opening angle evaluation) and a sequential road counter (used in the next stage to avoid pairing tracks belonging to the same road). Two more stacks are provided to store tracks arising from the next road. In the less frequent case of events with more than two roads, tracks belonging to the third and following roads are stored in both sets of Stacks. This procedure is a good compromise between keeping down the number of required Stacks and minimizing the number of unnecessary track pairings in the mass calculation loop. This loop, the last stage of the processor, is driven by a Do Loop Controller identical to the one governing the track finding. The pairings can be started as soon as the first road has been fully processed and at least the first track from the second road has been found; another level of parallel processing is thus provided. All possible pair of tracks are exteacted from the Stacks and sent through a set of 4 4K MLUs, programmed to calculate respectively the product of the two momenta (more precisely $(p_1 p_2)^{-1/2}$), the opening angle in the bend plane θ_x, the 3-dimensional opening angle θ, and the mass M of the dimuon system. The 3-dimensional opening angle is obtained by combining the computed θ_x, with the θ_y available from a separate section of the processor. In the current implementation, only the two CPY hits that are furthest apart are used to give an opening angle so that the answer may be incorrect when more than two counters were hit. In the future we expect to perform a more exact θ_y evaluation by imposing that the CPY hits used to compute the opening angle belong to the actual quadrants pointed at by the reconstructed tracks. This more elaborate scheme is expected to improve the rejection factor by a factor of two. The final answer of the processor is a yes or a no, depending upon the computed masses being larger than a value set in the mass MLU. An early event rejection is also possible when no two roads with good tracks are found.

IMPLEMENTATION AND PERFORMANCE

After the general scheme for the processor had been laid out, a FORTRAN program was written to simulate the processor operation in every detail. Monte Carlo generated events were fed to the program, so that the expected resolution (Fig. 8), the best MLU programming, the road configuration, etc. could be studied. In a later stage, actual events from test run data tapes were processed by the simulator, to provide a figure for the expected rejection rate of unwanted events. Another software simulation, written in the interactive language FORTH (5), was developed in parallel with the assembly and interconnection of the processor's various modules: events of increasing complexity were presented to the processor as it was being assembled, allowing a gradual debugging of its performance. This task was accomplished through the use of two test modules, capable of being loaded via CAMAC with fake events and of downloading them to the processor at the same rate and with the same protocol as expected in the experiment. The final phase of testing consisted of loading the test modules with real events from data tapes, and comparing event by event the processor results with the ones provided by the FORTRAN

simulator. When attached to the experiment, the processor performance was first moni-
tored by recording its answer for every trigger, in parallel with the normal data
acquisition. At a later stage, the consistency of operation was checked by scaling a
set of internal conditions (number of tracks rejected and accepted, mass loops, events
rejected, etc.). In addition a pre-scaled trigger would inhibit the processor response
and force an exhaustive read-out of all the processor Stacks, so that the correctness
of operation could be continuously monitored. The time employed by the processor to
analyze the events was measured to average between 5 and 10 microseconds, according to
the specific beam and trigger conditions. Aside from the experimental conditions, the
rejection factor is a function of the cut-off value for the mass. Figure 8 shows the
spectrum of masses computed by the processor for the events containing at least one track
in each of two roads (~ 40% of the total triggers). In the cases in which more than
two tracks are found, the largest calculated mass value is entered in the plot. A cut-
off value of 2 GeV was found to produce (in agreement with the simulation) negligible
loss of good events in the region of interest ($M_{2\mu}$ > 3 GeV, see Fig. 10), while providing
a total rejeciton rate ranging between 5 and 10, the exact value being a function of
the spill structure.

ACKNOWLEDGEMENTS

TABLE I

DETECTOR NAME	TYPE	NUMBER OF ELEMENTS	CELL OR COUNTER DIMENSION	TOTAL AREA COVERED
DC4	Drift Chamber	124	(2×118) cm^2	2.9 m^2
DC5	Drift Chamber	176	(2×168) cm^2	5.9 m^2
DC6	Drift Chamber	176	(2×168) cm^2	5.9 m^2
CPX	Scintillation Counter Hodoscope	184	(4×100) cm^2	7.4 m^2
CPY	Scintillation Counter Hodoscope	48	(8×200) cm^2	7.7 m^2
$\mu 1$	Scintillation Counter Hodoscope	60	(20×145) cm^2	17.4 m^2

NOTES AND REFERENCES

1. E. Barsotti et al., IEEE Trans. of Nucl. Scie., 26, 686 and ECL-CAMAC Trigger Processor System Documentation, Fermilab TM-821 (1978).

2. R. M. Baltrusaitis et al., Proposal to Study $\bar{p}N$ Interactions in the p-West High Intensity Laboratory, Fermilab E-537, January '78.

3. New modules designed for this experiment were:
   ECL-24   Test Module
   ECL-25   Test Module
   ECL-26   Router
   ECL-27   Comparator
   ECL-28   Arithmetic Logic Unit
   ECL-29   Track Finder
   ECL-30   Latch
   ECL-31   Hit Array

4. B. Cox, Fermilab Report 79/1, 0090.01, January '79.

5. H. W. Hammond and M. S. Ewing, FORTH Programming for the PDP-11, DECUS No. 11-232.
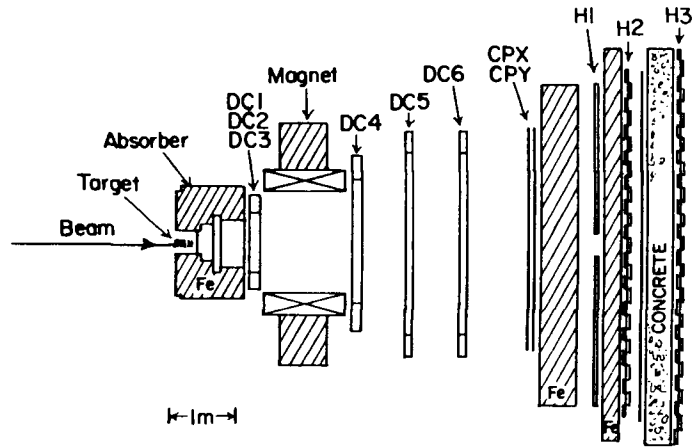
Fig. 1   E537 π, $\bar{p}$ spectrometer
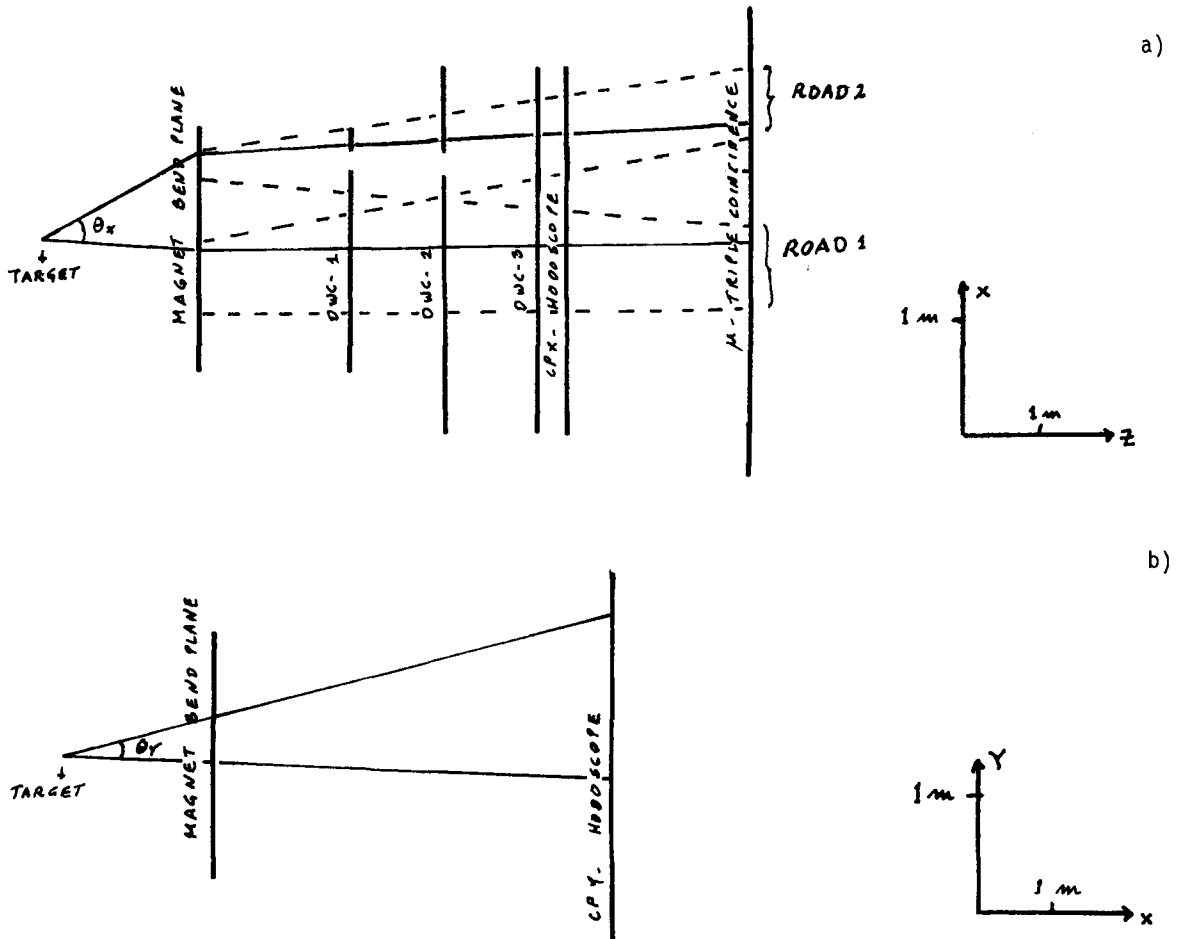
Fig. 2   Processor's view of the experiment.
The roads are shown by dashed lines in Fig. 2a).
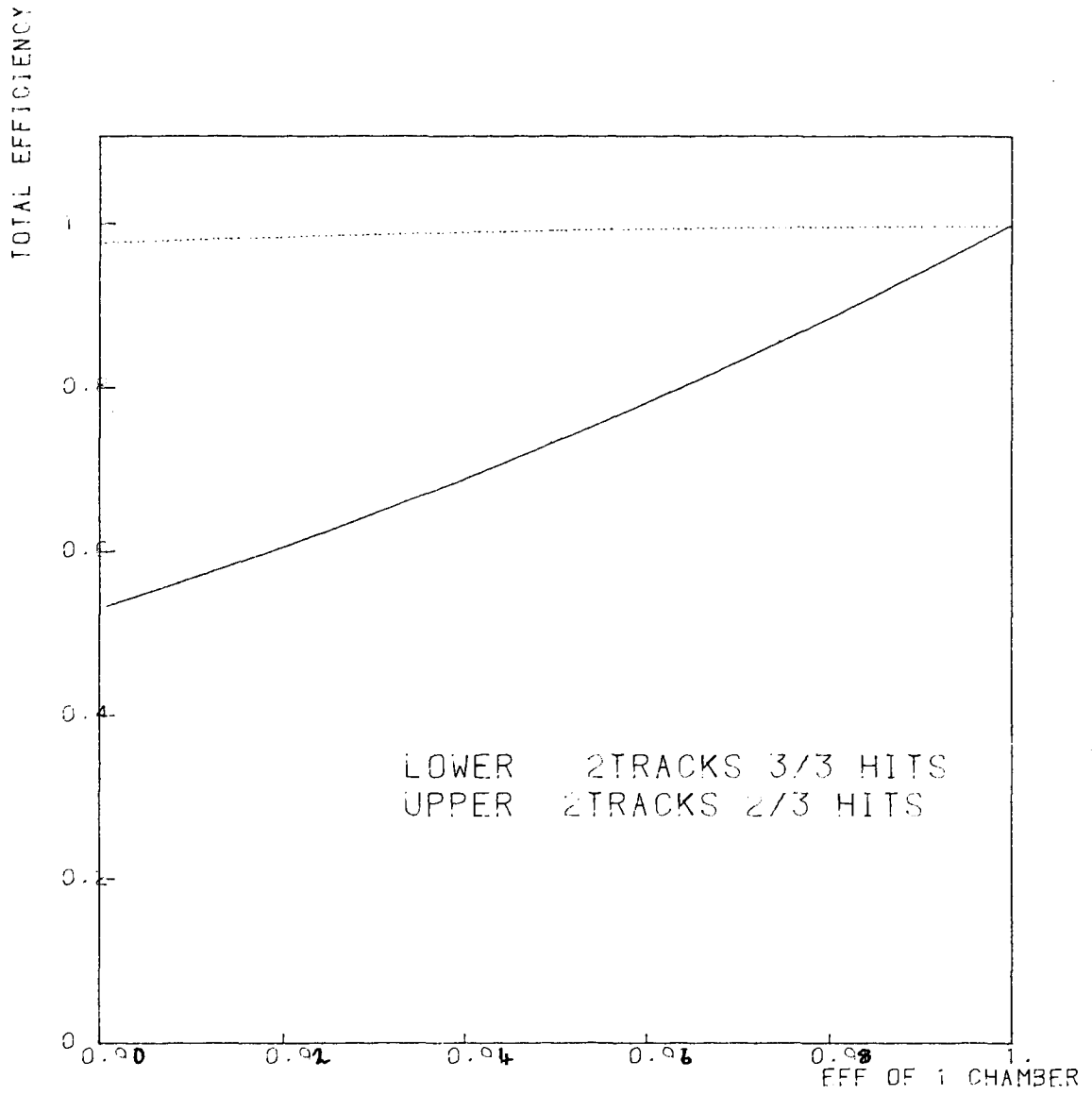
Fig. 3 Total track finding efficiency
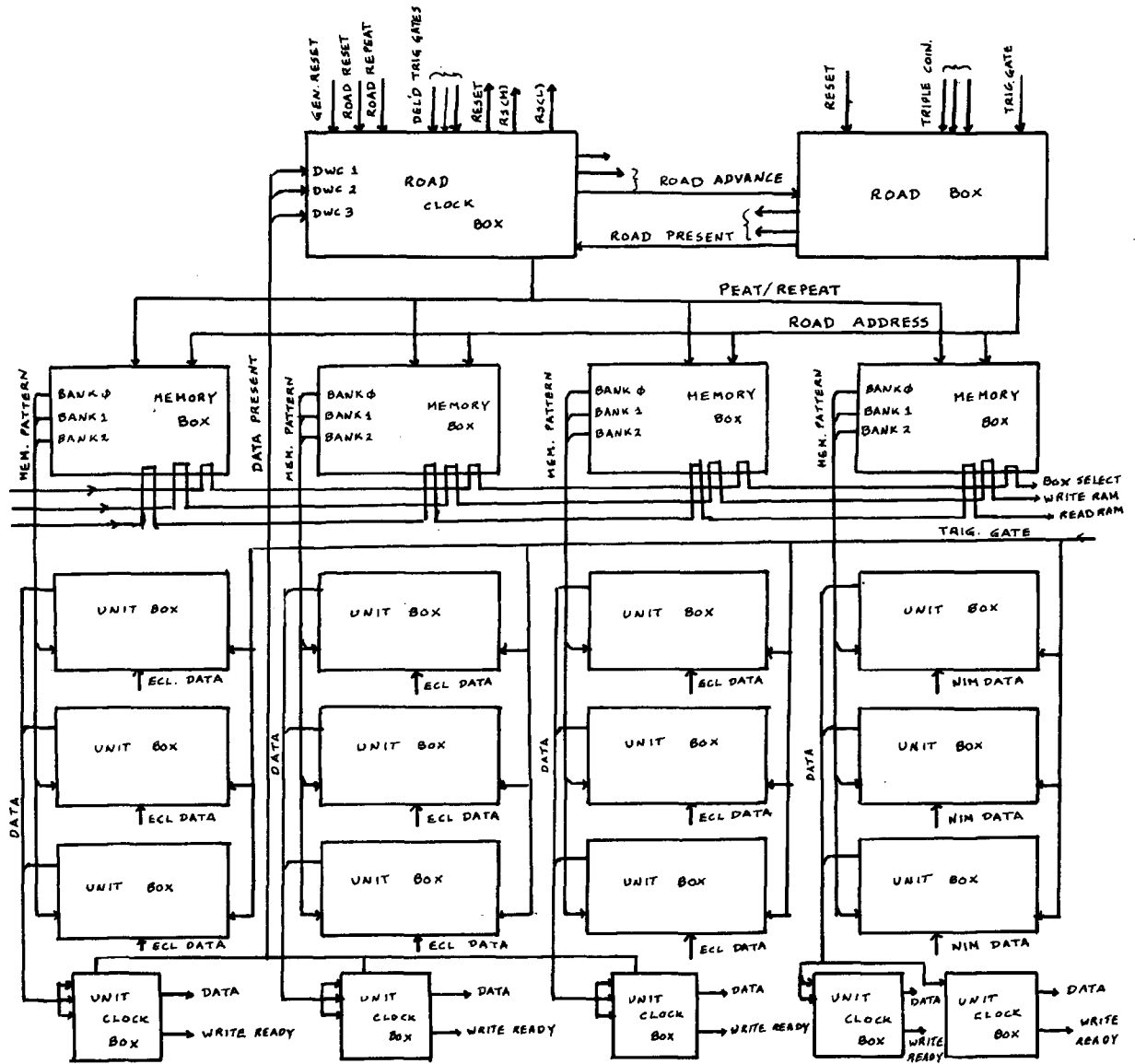
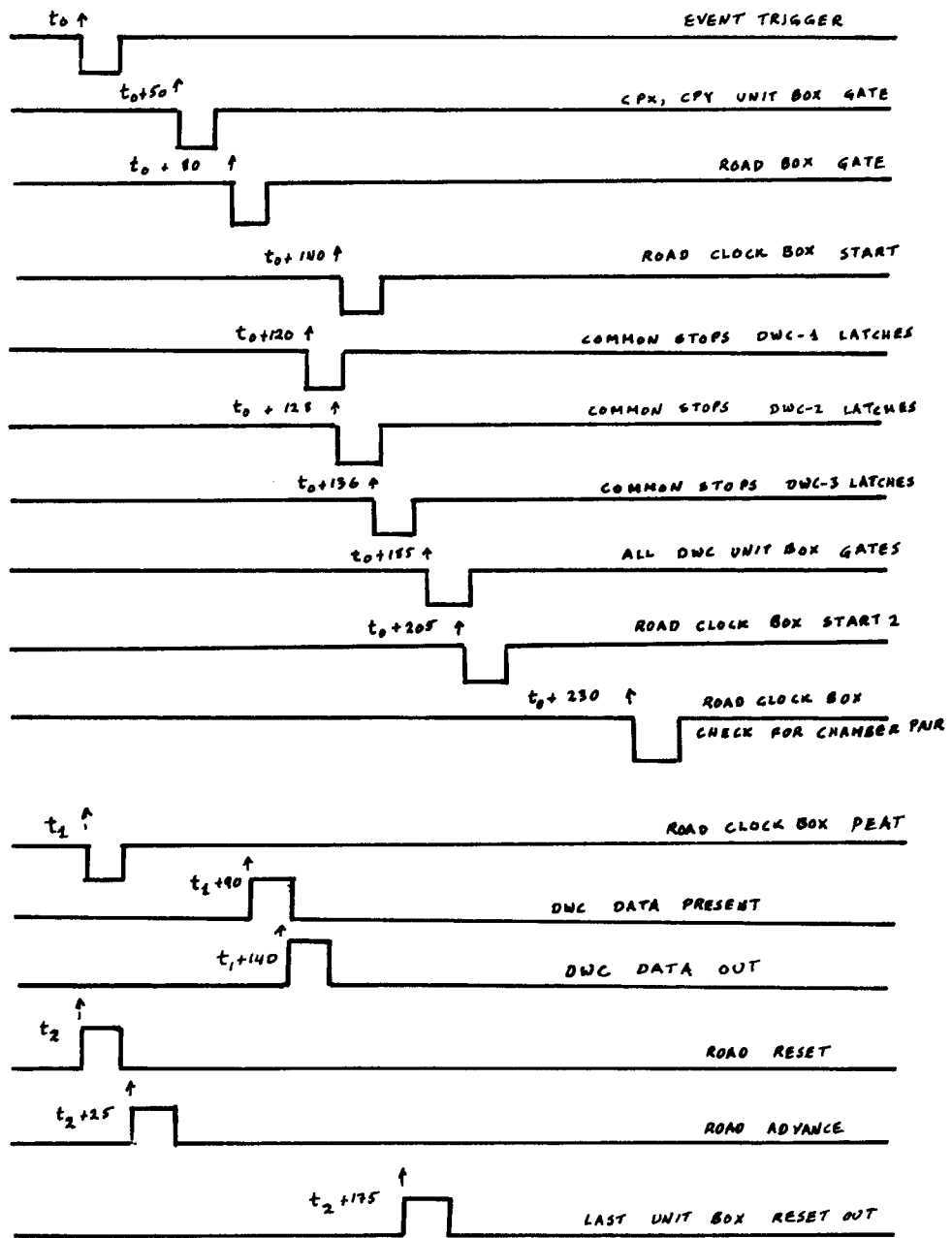Fig. 4 Block diagram of the processor interface

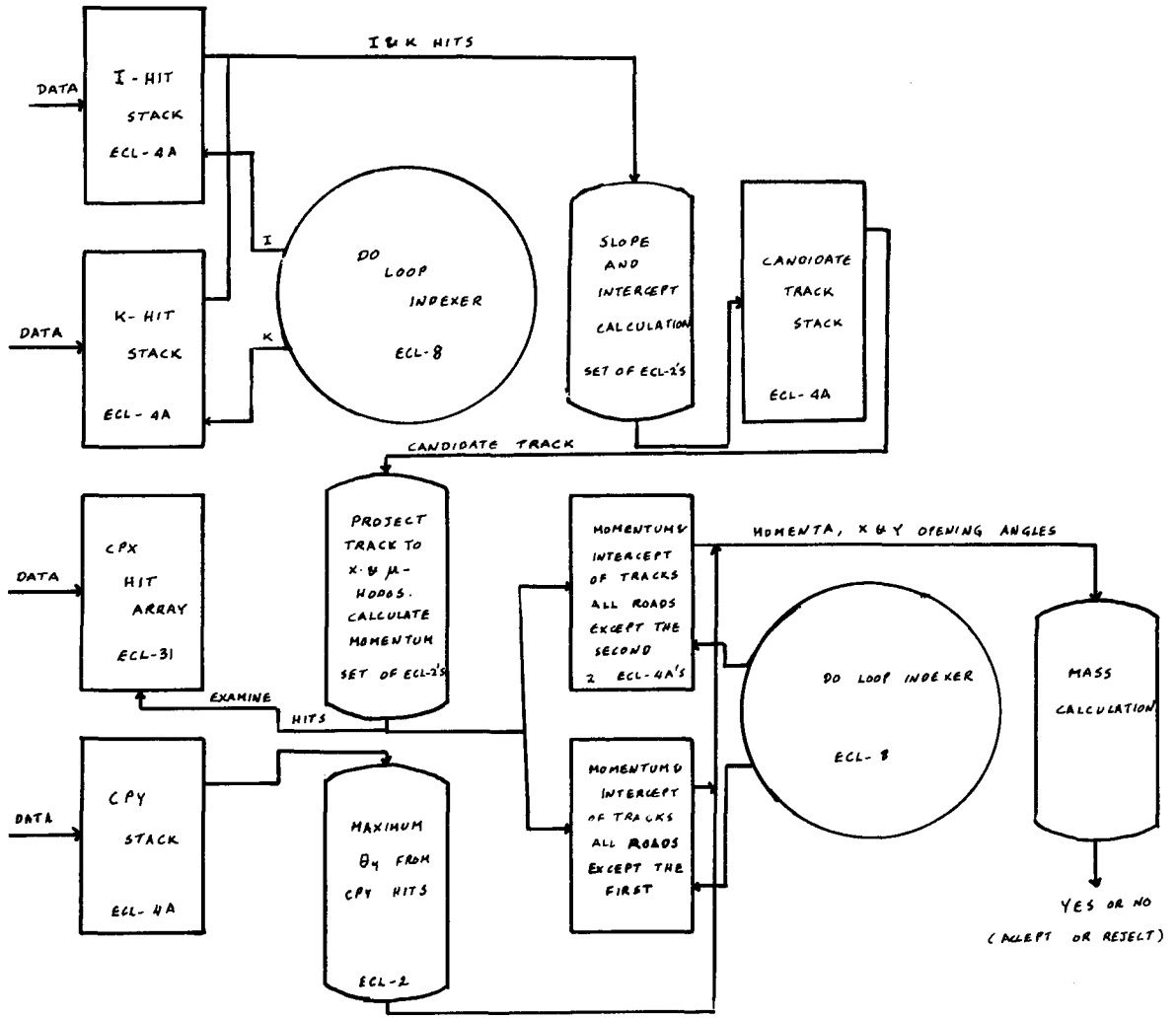Fig. 5 Timing diagram of the processor interface
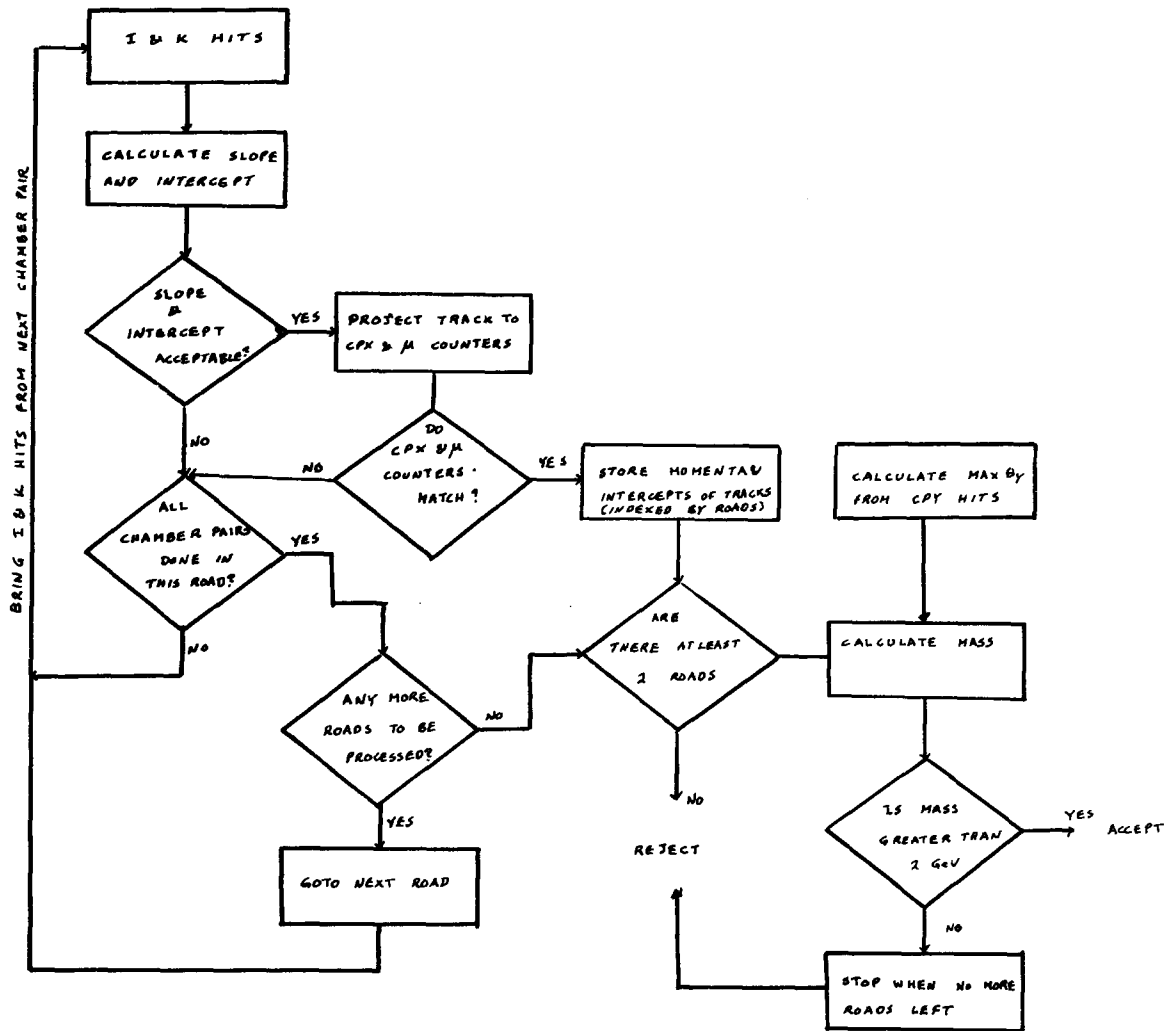
Fig. 6 Trigger processor block diagram

Fig. 7   Trigger processor flow chart

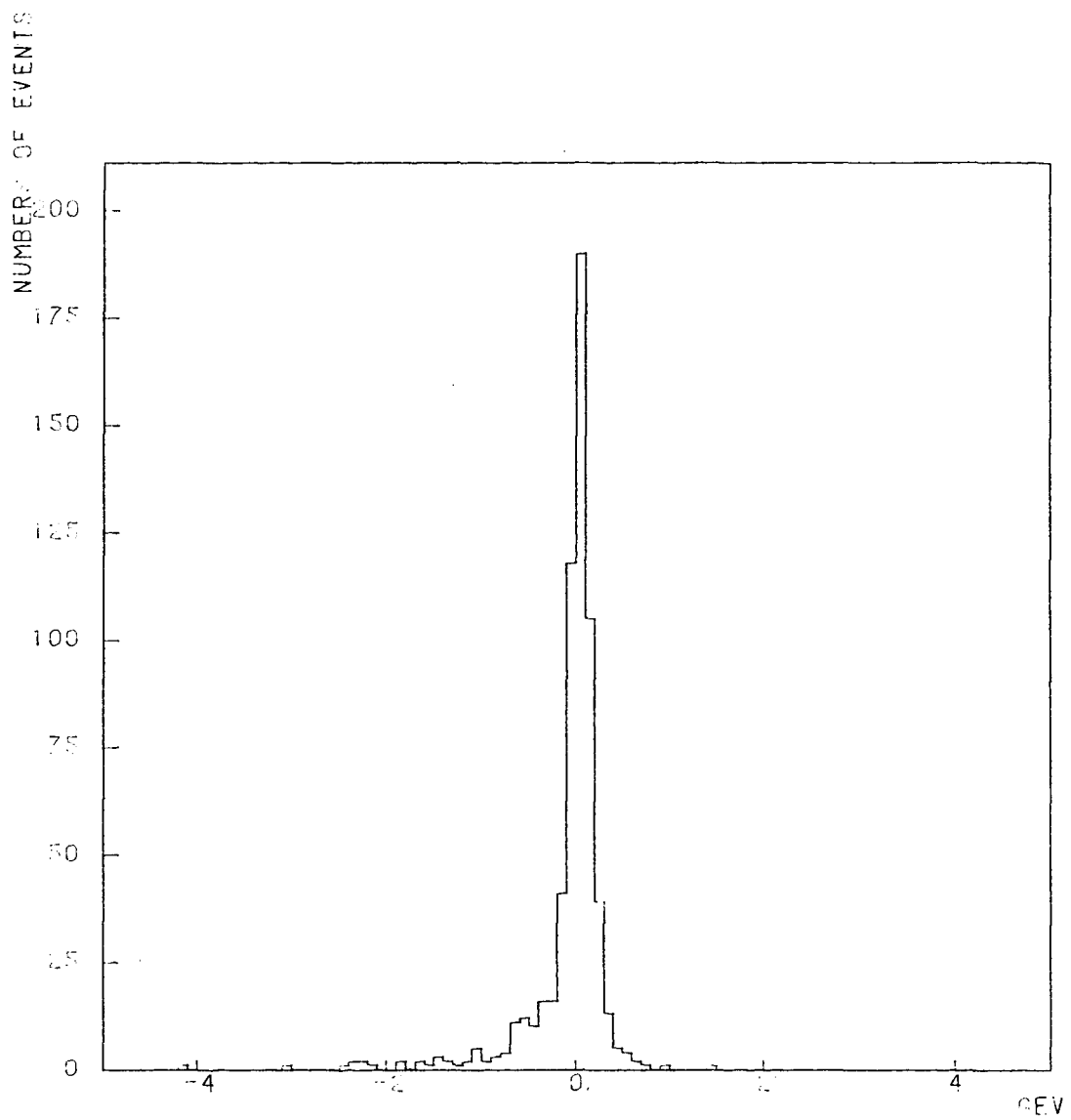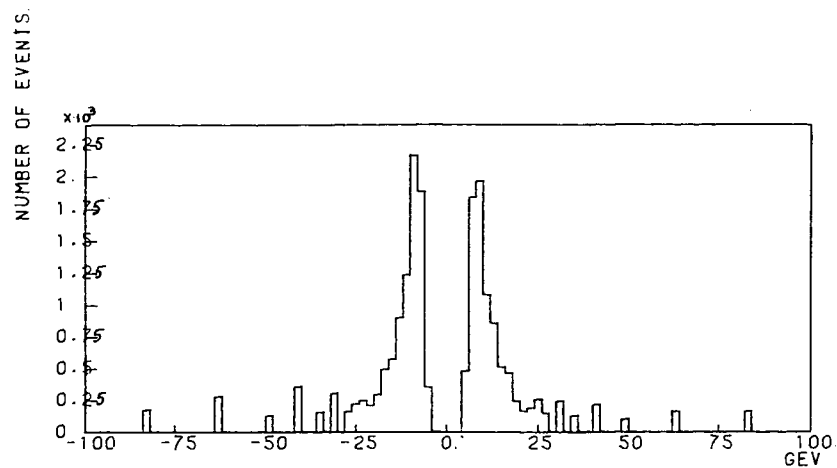Fig. 8 Mass resolution - TP results - Tape DG0274 (13000 events)
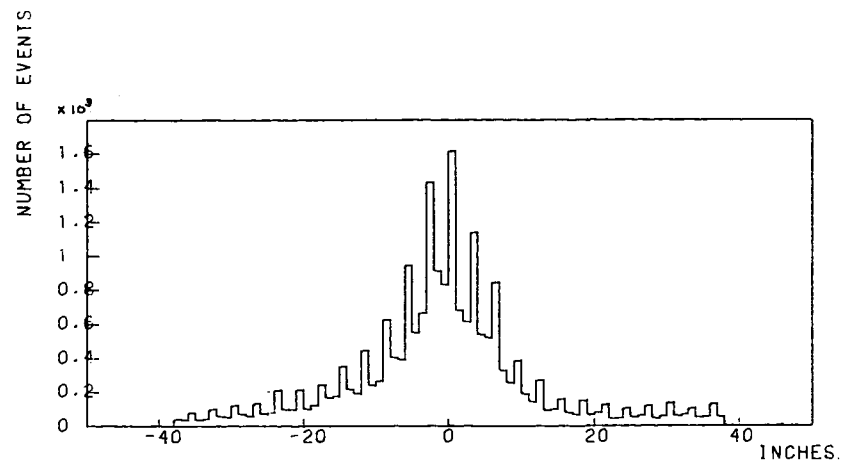
Fig. 9 a) TP momenta

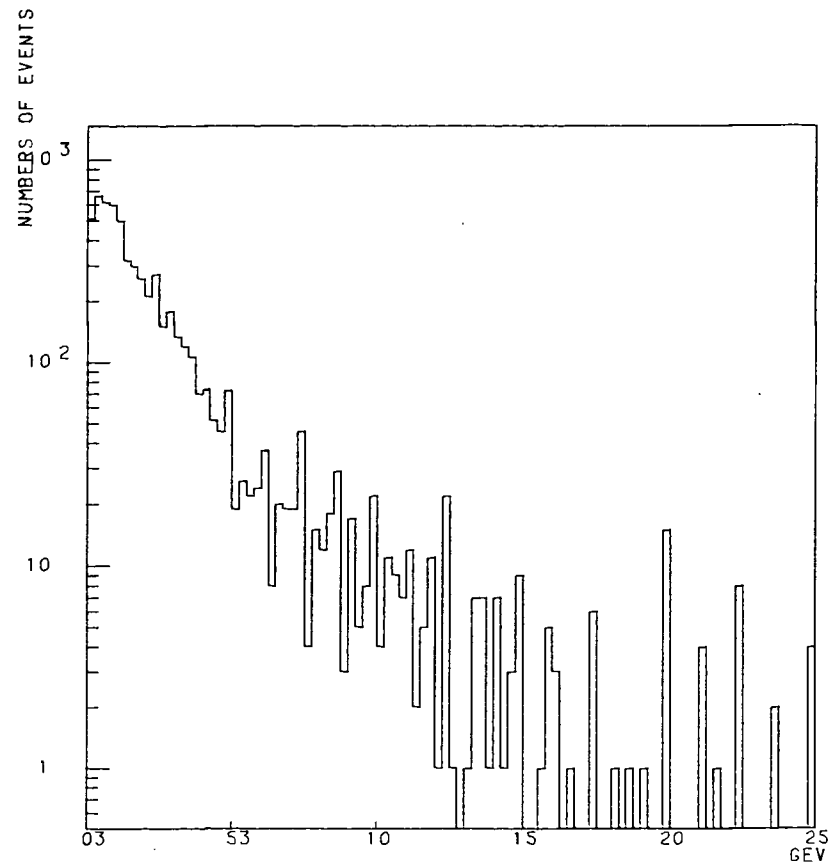Fig. 9 b) TP intercepts - TP results -
Tape DG0274 (13000 events)

Fig. 9 c) TP maximum mass - TP results -
Tape DG0274 (13000 events)

NUMBERS OF EVENTS

--- UNCUT

— TP=YES

Fig. 10 a) Mass - TP results -
Tape DG0274 (13000 events)

NUMBERS OF EVENTS

Fig. 10 b) Mass (TP = NO) 4

NUMBERS OF EVENTS

Fig. 10 c) Mass (TP = YES) - TP results -
Tape DG0274 (13000 events)