K0882-0104

CENTRAL INSTITUTE OF PHYSICS

*) INSTITUTE FOR PHYSICS AND NUCLEAR ENGINEERING

Bucharest, P.O.Box MG-6, ROMANIA

**) GSI - DARMSTADT, FRG

NP-59-1987 May

ADVANCED IPNE DATA ACQUISITION SYSTEM[+)]

*) M.DUMA, D.MOISA, M.PETROVICI, I.BERCEANU, M.IVASCU, G.PASCOVICI, V.SIMION, E.OSVATH

and

**) R.BOCK, A.GOBBI, K.D.HILDENBRAND, J.LYNEN, W.F.J.MÜLLER, M.BEESKOW

Abstract: A complex and flexible data acquisition system has been developed in order to run relative complex experiments at our acceleration system -ALIGATOR.

AIDA programme has been carried out on a small PDP - 11/34 computer and is based on a CAMAC hardware. The main hardware and software features are presented.

+) Work supported by the State Committee for Nuclear Energy under the contract no. 62-86-30.
Partly supported by the scientific exchange agreement between Romania and Federal Republic of Germany.

1. INTRODUCTION

A multiparametric set-up - DRACULA - realized in order to study reaction mechanisms at IPNE-acceleration system - ALIGATOR, has been presented in a previous progress raport [1].

The main argument in the favour of a new type of data acquisition system and some of the requirements which should be fulfilled by this have been listed.

As a consequence, the Advanced IPNE Data Acquisition system has been developed and implemented.

For AIDA's hardware a CAMAC system has been chosen as long as it offers an unsurpassed flexibility at relatively low cost. Even if its speed is not too high, nevertheless is completely adequate for a medium size experiment. For a large experiment is used a logic signal for every type of event. This signal gates the corresponding ADC's, makes the interrupt to the computer and is used to define the corresponding event type.

Under such conditions, all ADC's and TDC's can be treated as slaves.

AIDA's software is a system of programs designed to facilitate data acquisition associated with experiments performed at ALIGATOR (especially for DRACULA, see ref.[1]).

It allows an experimentalist to solve standard problems using commands which enable manipulation of complex data structure. As a consequence, the acquisition of data for experiments performed at IPNE using DRACULA arragement or other set-ups of similar complexity is realized by AIDA running on a PDP-11/34 experiment computer and the data analysis is performed by ANALIZ (see ref. [1]) on PDP-11/34 or CORAL-4012.

Section 2 gives an outline of the hardware structure of AIDA. The general structure will be presented in more detail in the case of operation of a large position sensitive ionisation chamber - the main detection and identification system of DRACULA.

In Section 3 is presented AIDA's software, its block diagram and a detailed description of the commands.

The concluding remarks are presented in Section 4.

2. AIDA's HARDWARE STRUCTURE

The complexity of experiments prepared to be performed at ALIGATOR [1] requires much more flexibility and higher performance in data acquisition.

Using the experience gained in this field at other experimental facilities [2, 3, 4] a similar solution has been chosen.

Data acquisition has to have the following functions:

i) conversion of the analog signals created from physical detection systems into digital numbers.

ii) accumulation of these data and storage them on a magnetic tape or disk.

A description of the AIDA's hardware is the subject of this section.

As already mentioned in Introduction, AIDA is based on a CAMAC hardware. Such a solution gives the possibility to run complex experiments based on different type of detectors, some or them consisting of several counters.

With a logic signal associated to every event type one can use several simultaneous triggers. This gives the possibility to run in parallel different type of experiments. The different counting rates can be ballanced via so called Scale Down

Units. If the experimental set-up has detectors consisting of several counters of the same type, then a multiplicity trigger gives the simplest way to handle such a device.

The possible trigger signals are used to generate a LAM(via a dedicated Event Box) to the computer (in our case a PDP-11/34 experiment computer). Once a LAM accepted, via the software the computer reads the state of the Pattern Unit (PU). This information is used to select the corresponding CAMAC locations and their read - out sequence by the specific software written for a given experiment.

This is the way in which a variable event length is constructed and stored in two 4kW buffers which operate in a ping-pong mode.

Every full buffer is thrown on a magnetic tape by a special subroutine supported by AIDA.

The acquisition system is restarted by a single CAMAC command. A general scheme of the trigger logic for a complex experimental arrangement can be seen in Figure 1.

An example of a dedicated configuration of AIDA realized for DRACULA operation can be followed in Figure 2.

The logic structure of the system is schematically shown in Figures 2, 3 and 4.

In Figure 5 are presented the time signals from o-grid to TDC (TD 811) used to obtain x-position of the detected fragments by IC.

The time diagram presented in figure 6 gives the sequence of the time signals for accepted and rejected events.

In order to reduce the effective time used by the computer for acquisition, an auxilliary crate controller J-11 is forseen to be implemented in AIDA's hardware.

For the moment the event rate which is not to high due to reading out of events via software does not seem to be a limiting factor for the experimnts which are going to use AIDA.

For special purposes a DMA tranfer is in our view.

3. AIDA's SOFTWARE

AIDA's software contains a system of programs designed to facilitate the data acquisition associated with complex experiments which are using the hardware described above.

The manager program AIDA supervises other two working programs ACHIZI and PRELUC as well as the display procedures.

Written in FIV language, AIDA takes 32 kW of the computer capacity.

AIDA organizes two distinct data types:

- list mode data (event by event) and
- spectra (accumulated events).

Well known, a list mode data consist of sequences of events. An event is a set of numerical values associated with a "physical event"; all the parameters measured in connection with the "physical event". An event contains several parameters correlated in time. A parameter is a numerical value resulted after conversion of an analog signal delivered by a detection system. The parameters could represent physical coincidences (a coincidence experiment) or different values associated to a single physical event (all the parameters from a position sensitive ionization chamber for example)

To keep track of the correlations between the different parameters for later analysis, they are usually written event-by-event onto a storage device.

Spectra may be accumulated in the memory of the experiment computer (PDP-11/34 in our case).

Figure 7 shows an event-format for a general complex experiment.

AIDA's general scheme can be followed in Figure 8.

Written in MACRO-11, ACHIZI realizes the dialogue with the CAMAC crate, reads the CAMAC locations, builts the events with variable length, transmits events to PRELUC and stores the events in a list mode in two 4W buffers operated in a ping-pong mode.

Once a buffer is filled,the events will start to be written in the empty one and the first will be thrown on a magnetic tape. When PRELUC is active and ready to analyze a new event, ACHIZI will transmit to it a new event.

A block diagram of ACHIZI is presented in Figure 9.

PRELUC is foresean to realize a rough data analysis with user written routines. Only part of the stored events are analyzed in order to have a check of the operation of different detectors and to choose the best experimental conditions and in the same time to avoid a competition with ACHIZI. In such a way the dead time of the system is not influenced by running ACHIZI and PRELUC in parallel. It creates a special zone in the computer memory where one-dimensional spectra may be accumulated.

Under the commands accepted by AIDA, PRELUC is processing the events following the definitions included in other two subroutines UNIONL and BIDONL.

These subroutines are written for each specific experimental configuration and the AIDA environment is created again using the commands indicated in AIDA.CMD.

The main AIDA's commands are the following:

> X - start data taking without writting onto tape. One should see the lights in the various data way displays blinking. This is a check that data are being acquired.

> IB - a tape is initialized - a file is opened and tagward with specific comments can be introduced. At this command the tape is automatically positioned. The above information are written in the header of each file where the starting time will be also inserted.

> X _ A - start data taking with writing onto tape.

> S _ N - stop writing on magtape but continuing to acquire data without closing the file on tape.

> S _ C - stop acquisition, closing the file.

> XP- the analysis routine PRELUC is started.

> SP- the analysis routine PRELUC is stoped.

The main display process is automatically created by starting the system. It is connected to the main console and has allocated a graphic terminal.

> XL _ n - this command gives the possibility to have a live-mode display of a two-dimensional spectra number n.

> SL- stop the live-mode display.

> D _ n,m - under this command a zone m of the one-dimensional spectra n is displayed.

>M - a marker will appear on the screen.

> E - expand command.

> ? - help command. A list of the all commands will appear on the screen.

> CTRL/Z or PA - AIDA is temporary stopes and restarted by RES-AIDA - restart AIDA.

> ST - stop AIDA.

There exists a partition specially created, the same for all three programs, used to sincronize the activity of the two working programs ACHIZI and PRELUC.

4. CONCLUSIONS

The effort invested to implement a versatil and high-efficient experimental set-up (DRACULA) [1] for nuclear reaction mechanism studies has been pursued for AIDA realization.

Taking the advantage of previous experience in the field [2] , AIDA programme has been carried out on a small computer PDP- 11/34 and is based on a CAMAC hardware. This solution offers a high flexibility at reasonable cost.

A specific signal for every type of detector gives the possibility to select the events of interest early enough in the data flow.

Such a procedure reduces the total amount of harware processed data and supresses the coordinates which do not contain information.

Data visualization can be performed using graphic terminals.

One-dimensional spectra and high resolution two-dimensional scatter plots are used during the setting-up, run and calibration of a given experiment.

DRACULA in combination with AIDA open the possibility to performe different type of experiments in heavy ion collision between heavy targets (A = 30-60) and light projectiles at incident energies E ≥ 5 MeV/n, avâilable at ALIGATOR - the acceleration system of our department.

R E F E R E N C E S

[1]. M.Petrovici, V.Simion, I.Berceanu, A.Demian, M.Duma, M.Ivascu, D.Moisa, E.Osvath, G.Pascovici, R.Bock, A.Gobbi, K.D.Hildenbrand, U.Lynen, H.Sann, G.Augustinski, H.Beeskow, H.W.Daues, IPNE- Preprint NP-55-1986.

[2]. U.Lynen, H.Stelzer, A.Gobbi, H.Sann, A.Olmi, Nucl. Instr. and Meth. 162 (1979) 657.

[3]. W.F.J. Müller - private communication.

[4]. EDAS - GSI Vol, I-III.

FIGURE CAPTIONS

Fig. 1 A general schemeof the trigger logic for a complex experimental arrangement based on different type of detectors some of them consisting of se· ral counters.

Fig. 2 Block diagram for hardware - oriented data system for DRACULA;(E1,E2,E3,ER)↑ and (E1,E2,E3,ER)↓ are the ΔE and E_{res} signals corresponding to the anodes of the position sensitive ionization chamber splitted in a given way [see Ref. 1].
Σx (-1) - sum inverting amplifier, TRX-(TTL-NIM) translator, ADC (AD 811).

Fig. 3 The trigger logic for the big ionization chamber itself.
K-prompt cathode signal, BP-pattern unit 16P-2047, EV-event box.

Fig. 4 The logic structure for FC (fast clear), Restart, clear and Reset signals.

Fig. 5 A diagram of the time signals from θ -grid to TDC (TD 811)used to obtain x-position of the detected fragments by the IC.

Fig. 6 A suggestive time diagram with the sequence of the time signals for accepted and rejected events.

Fig. 7 An event-format for a general, complex experiment.

Fig. 8 Data flow in the AIDA program.

Fig. 9 A block diagram of ACHIZI program.

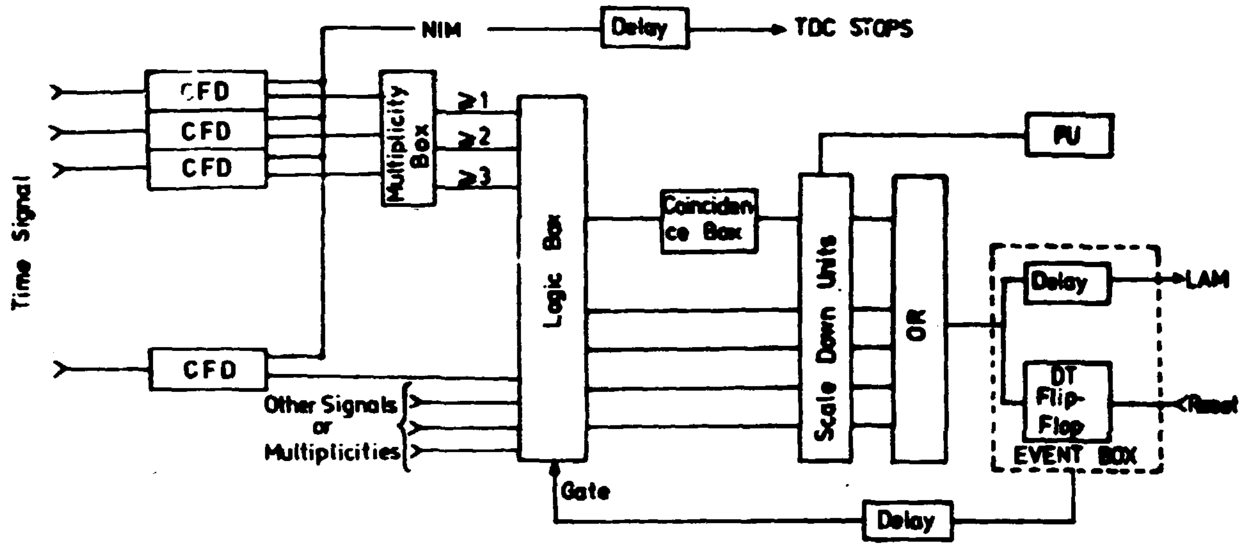

NIM
Delay
TDC STOPS
Time Signal
CFD
CFD
CFD
CFD
Multiplicity Box
≥1
≥2
≥3
Logic Box
Coincidence Box
PU
Scale Down Units
OR
Delay
LAM
DT Flip-Flop
Reset
EVENT BOX
Other Signals or Multiplicities
Gate
Delay

FIG.1

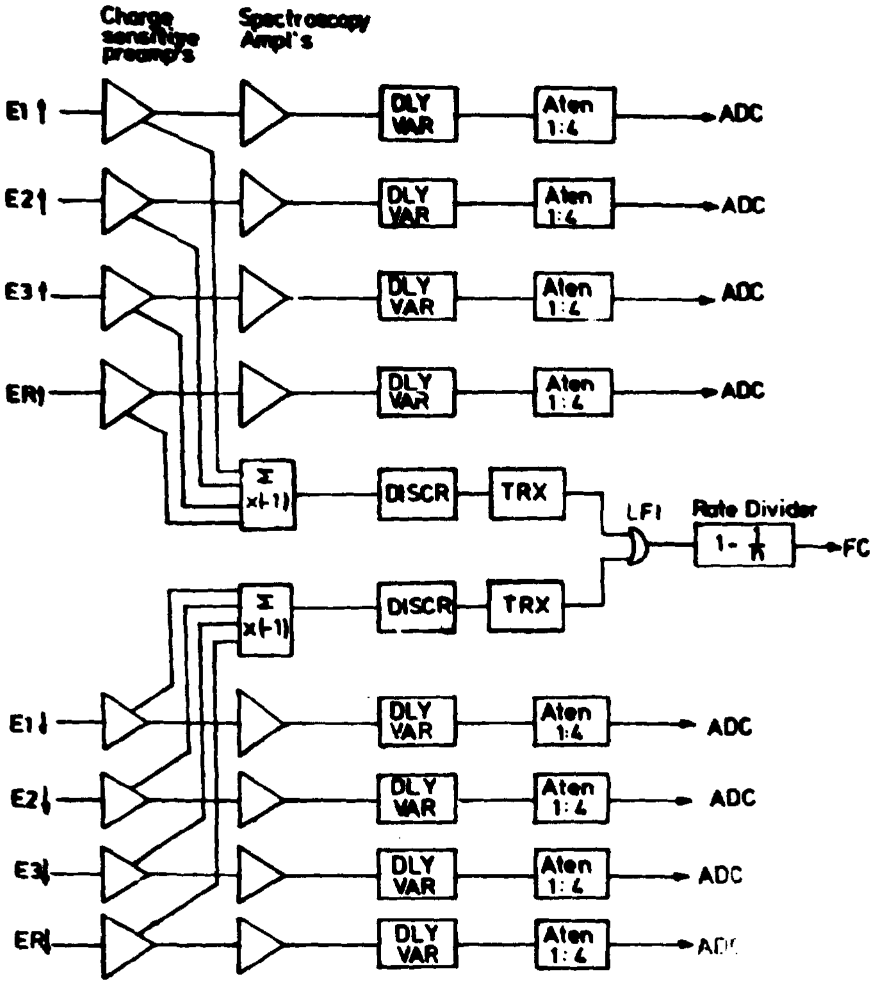
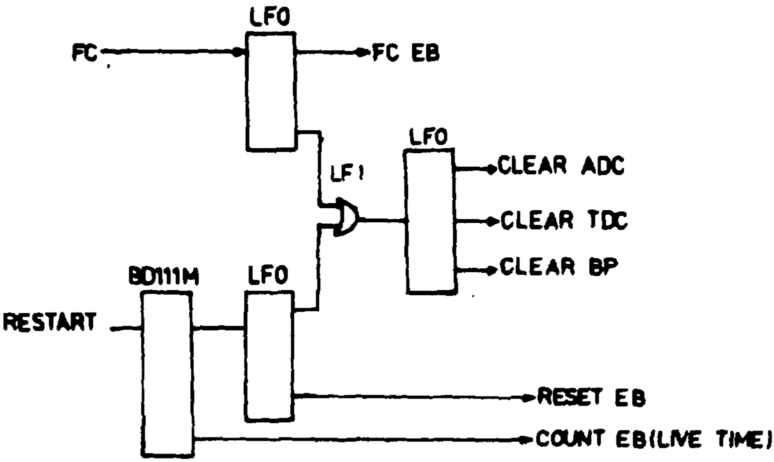
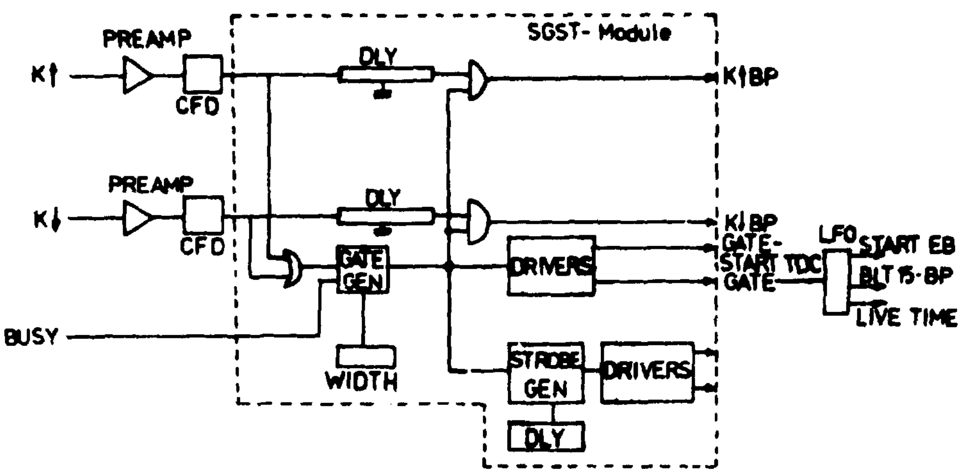

Charge sensitive preamp's
Spectroscopy Ampl's
E1↑
E2↑
E3↑
ER↑
DLY VAR
Aten 1:4
ADC
Σ x(-1)
DISCR
TRX
LFI
Rate Divider
FC
E1↓
E2↓
E3↓
ER↓

FIG.2



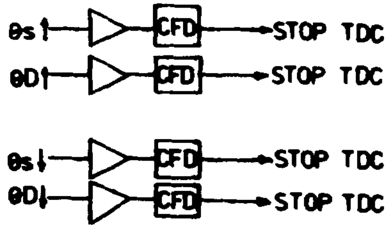
LFO
FC
FC EB
LFO
LF1
CLEAR ADC
CLEAR TDC
CLEAR BP
BD111M
LFO
RESTART
RESET EB
COUNT EB(LIVE TIME)

FIG.3



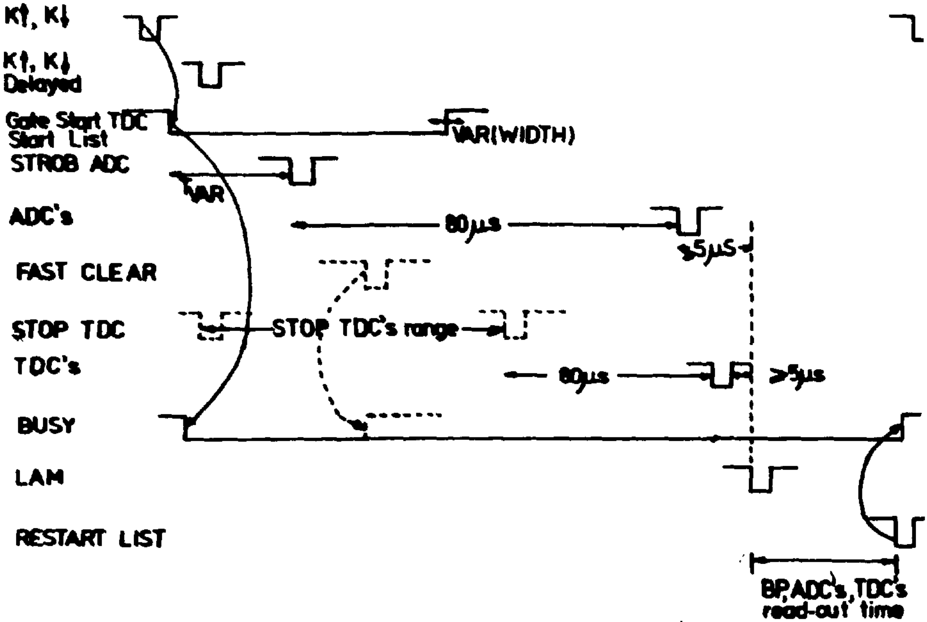
SGST- Module
PREAMP
K↑
CFD
DLY
K↑BP
PREAMP
K↓
CFD
DLY
K↓BP
GATE-
START TDC
GATE
LFO
START EB
BLT 15-BP
LIVE TIME
GATE GEN
DRIVERS
BUSY
WIDTH
STROBE GEN
DRIVERS
DLY

FIG. 4



θs↑
θD↑
θs↓
θD↓
CFD
STOP TDC

FIG. 5



K↑, K↓
K↑, K↓ Delayed
Gate Start TDC Start List
VAR(WIDTH)
STROB ADC
VAR
ADC's
80μs
≥5μS
FAST CLEAR
STOP TDC
STOP TDC's range
TDC's
80μs
≥5μs
BUSY
LAM
RESTART LIST
BP,ADC's,TDC's read-out time

FIG. 6

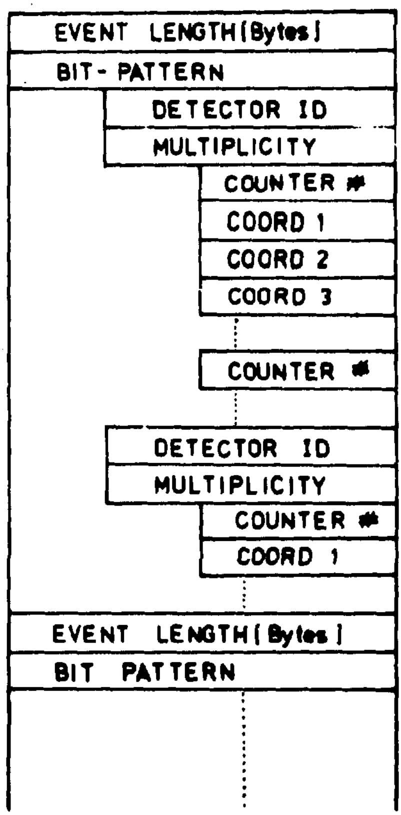

EVENT LENGTH(Bytes)
BIT- PATTERN
DETECTOR ID
MULTIPLICITY
COUNTER #
COORD 1
COORD 2
COORD 3
COUNTER #
DETECTOR ID
MULTIPLICITY
COUNTER #
COORD 1
EVENT LENGTH(Bytes)
BIT PATTERN

FIG.7


NO

FIG.9

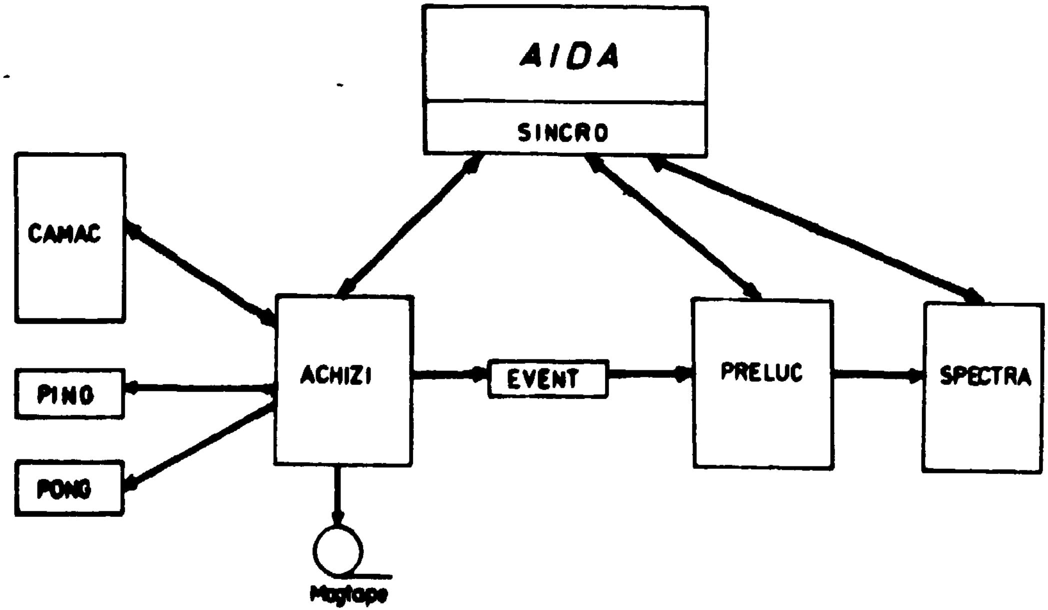
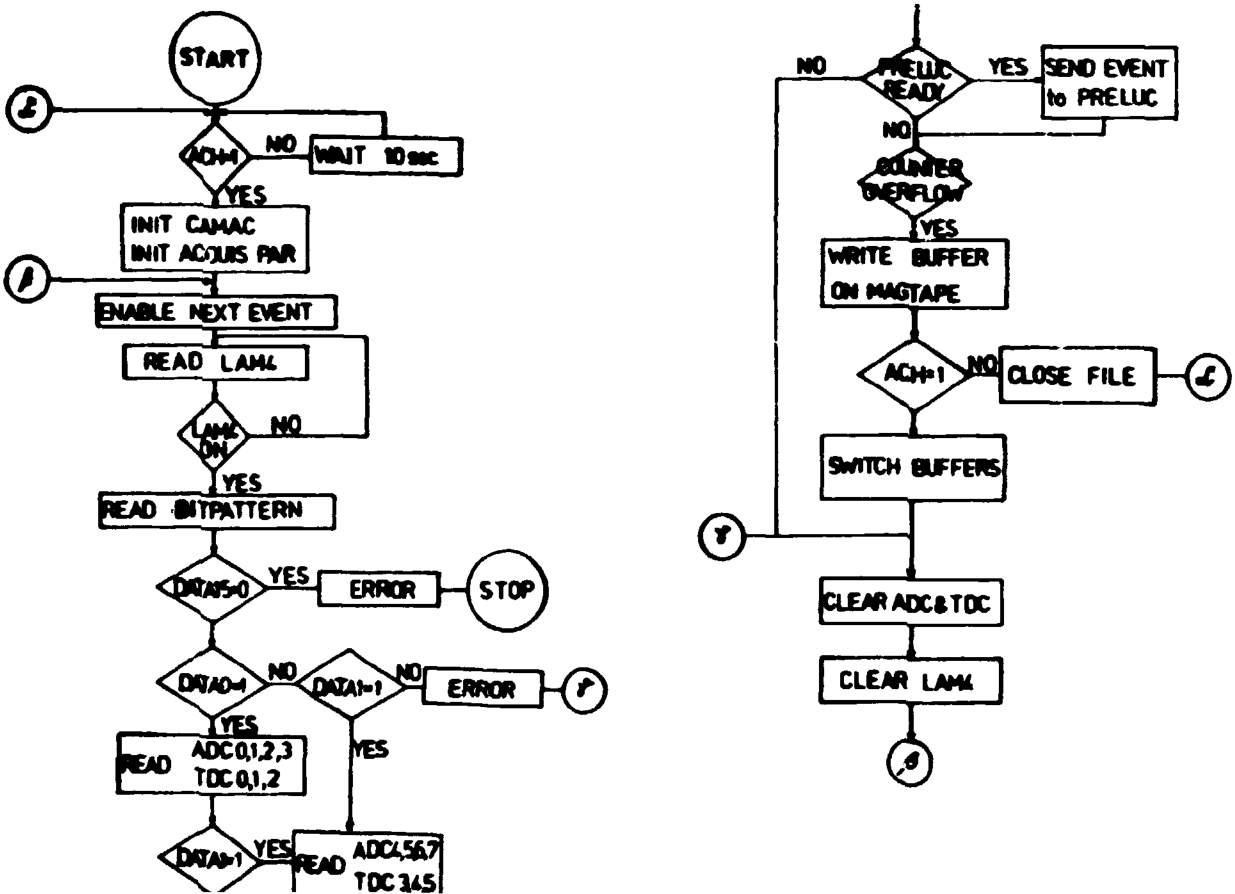

AIDA
SINCRO
CAMAC
PINO
PONG
ACHIZI
EVENT
PRELUC
SPECTRA
Magtape

FIG. 8



START
WAIT 10 sec
ACH=1
NO
YES
INIT CAMAC
INIT ACQUIS PAR
ENABLE NEXT EVENT
READ LAM4
LAM4 ON
READ BITPATTERN
DATA15=0
ERROR
STOP
DATA0=1
DATA1=1
READ ADC 0,1,2,3 TDC 0,1,2
DATA1=1
READ ADC4,5,6,7 TDC 3,4,5
PRELUC READY
SEND EVENT to PRELUC
COUNTER OVERFLOW
WRITE BUFFER ON MAGTAPE
ACH=1
CLOSE FILE
SWITCH BUFFERS
CLEAR ADC&TDC
CLEAR LAM4