Bull. Inst. Chem. Res., Kyoto Univ., Vol. 54, No. 1, 1975

Multiparameter Data Acquisition System with a Mini-Computer

Toshihiro MIYANAGA*, Takao OHSAWA*, Seiji TANAKA*, Noboru FUJIWARA*, Shigeru KAKIGI**, Kiyoji FUKUNAGA**, and Takuji YANABU*

Received December 1, 1975

Multiparameter data acquisition system with a YHP 2100A computer was developed. Details of hardware and outlines of softwares are described. This system is now used as 3 parameter system, but expandable to 4 or more parameter system. The system is very useful for multi-particle coincidence experiment such as three body reaction. An example is shown for the detection of elastic scattering between protons and deuterons.

I. INTRODUCTION

A Yokogawa-Hewlett-Packard (YHP) 2100A mini-computer has been used as a single parameter data acquisition system in Keage Nuclear Science Laboratory. A histogram mode has been adopted in this case, and has been proved to be very convenient for a single parameter data acquisition. But if a multi-parameter data acquisition is needed, the histogram mode requires unlimited data area. Therefore, a list mode (event recorder mode) data acquisition system has been developed recently. That is because, in our laboratory, the particle-particle correlation experiments have been studied and hence the multi-parameter analysis is required.

A data acquisition system for these correlation experiments is developed by using the YHP 2100A mini-computer and a paper tape output. The analysis of the data on the paper tape is performed by using the FACOM 230-48 computer of the Institute for Chemical Research, Kyoto University.

In this paper, the details of the hardware system, the outline of the software and of the data analysis by using the FACOM 230–48 are reported. The over-all system was tested by applying to the H+d→p+d scattering and the results are also reported.

II. DESIGN AND PERFORMANCE OF THE SYSTEM

1. ADC Control System (Hardware)

A block diagram of the system including peripheral devices is shown in Fig. 1. The YHP 2100A computer has 16 bit 16 k-word core memory. The memory cycle

* 宮永俊博, 大沢孝夫, 田中誠二, 藤原 昇, 柳父琢治 : Laboratory of Nuclear Reaction, Institute for Chemical Research, Kyoto University, Kyoto.

** 柿木 茂, 福永清二 : Nuclear Science Research Facility, Institute for Chemical Research, Kyoto University, Kyoto.

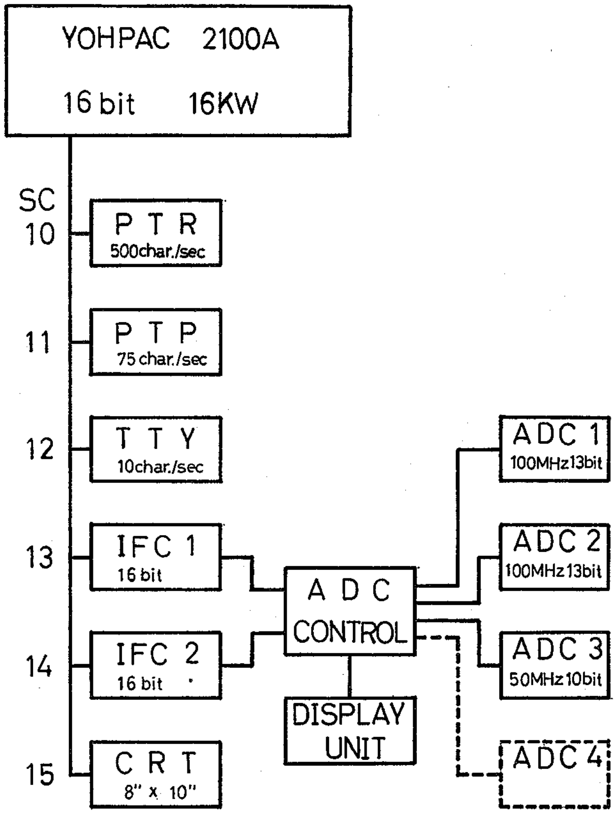

Fig. 1. Block diagram of the data acquisition system with a YHP 2100A.

time is 0.98μ sec. Peripheral devices are standard ones; paper tape reader (PTR), tape punch (PTP), teleprinter (TTY), and CRT display (CRT). The SC numbers in Fig. 1 indicate select codes of input/output slots, and the smaller number has higher priority level.

Four analog-to-digital converters (ADC) could be connected to the computer through ADC controller, but at present, three ADC's are equipped. Two of three are type NS–623 (13 bits 100MHz) of Northern Scientific, Inc. and the other is type NS–622 (10 bits 50MHz).

Figure 2 shows a block diagram of the ADC controller developed in our laboratory. Whole data lines to the computer have 32 bits information, therefore, two words are necessary for one event. The data bits are assigned to each ADC by relevant plug-in data configuration connector (DCC). According to the experimental requirements, a specified system mode is selected by using a relevant type of logic configuration connector (LCC) and by the operation of front panel switches.

The ADC controller consists of a control logic circuit and a protection buffer circuit for ADC and for IFC (general purpose interface card, YHP 16184B).

The operating status of the ADC controller is displayed by the light-emitting-

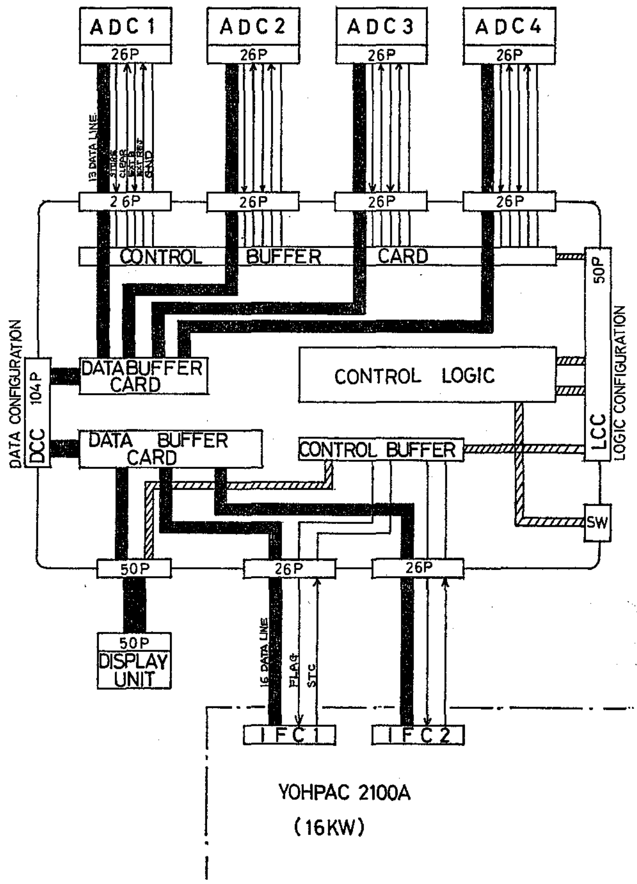

Fig. 2. Block diagram of the ADC controller.

diodes (LED) on the display unit.

The Sequence of Basic Operation of Data Transfer

Figure 3 illustrates the sequence of operations for a transfer of data from one ADC to the computer.[1,2] (1) The operation begins with a programmed instruction to set a control flip-flop (STC) and to clear a flag flip-flop (CLF). (2) The STC signal causes the IFC to issue a start command to the ADC. This signal resets the ADC and the ADC is ready.[3] Now the computer control goes to CRT display routine. (3) When the ADC receives an input signal and the conversion is complete, data ready signal (STORE) is sent from the ADC to the IFC and set the FLAG FF. (4) The FINISH signal is then sent to the CPU, and the display routine is interrupted and the computer control goes to the data taking service routine. (5) An I/O IN signal is generated in this routine and (6) the data are tansferred to the CPU. The sequence is completed and returns to (1).

The Logic of the ADC Controller

The requirements for the controller logic are; (1) When coincidence occurs

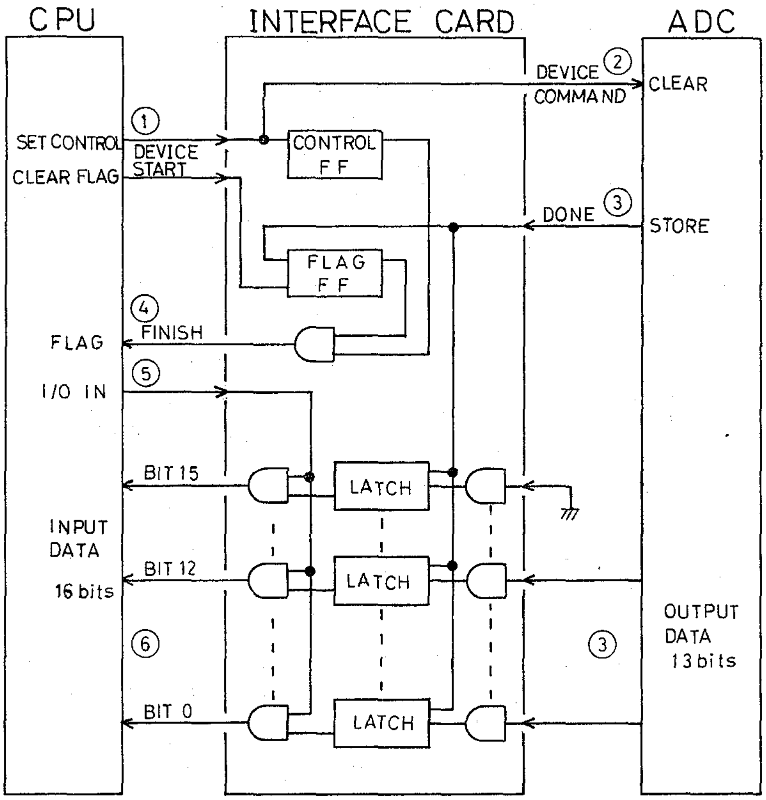

Fig. 3. Sequence of operations for a transfer of data from one ADC to the CPU.

among pulses from detectors and all ADC conversions are complete, data should be transferred to the CPU (NORMAL COINCIDENCE), (2) when there is no coincidence, data should not be transferred to the CPU (NO COINCIDENCE), (3) when the coincidence occurs and when one of ADC's resets internally due to overflow or underflow or an accidence, data should not be transferred also to the CPU (INTERNAL RESET).

Controller cycle begins with a busy signal generated in each ADC. The busy signal (EXT B), indicating that datum is being digitized, triggers each monostable multivibrator with several time constants, 0.6, 1.6, 3.9, and 9.6μ sec.

In NORMAL COINCIDENCE shown in Fig. 4, GCOINC signal is generated by coincidenced MONO pulses and triggers the CFF to hold COINC signal "high". CBS signal, ANDed signal of all ADC's busy signals, and COINC signal set CCB signal to "high". When all ADC conversions are complete, FLAG signal is generated by ANDed signal of CCB and CST, and the computer accepts the data. The CFF flip-flop and all ADC's are reset by STC signal from the CPU and the logic circuit is ready for a new cycle.

In the NO COINCIDENCE or INTERNAL RESET case, the data is invalid, and all ADC's are sequenced into the reset cycle.

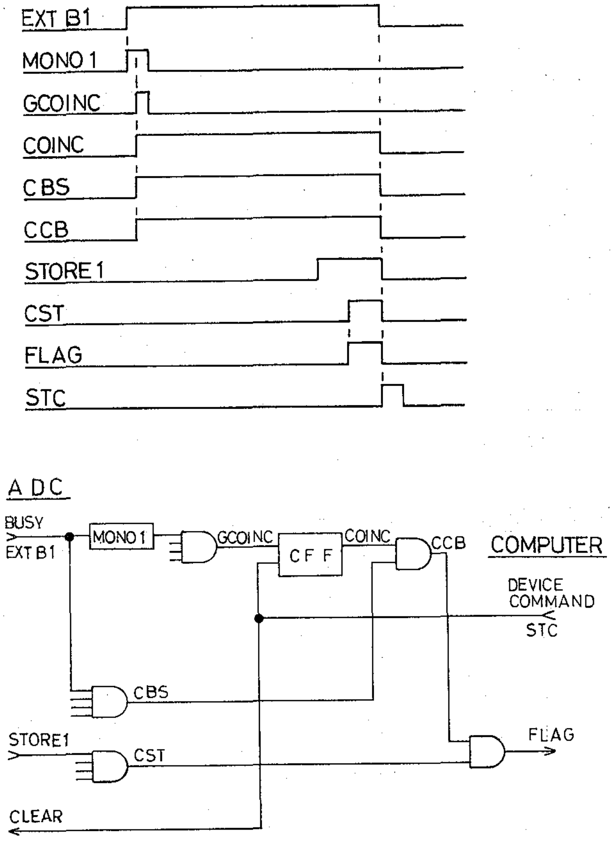

Fig. 4. Logic and the timing chart of the ADC controller in NORMAL COINCIDENCE mode.

As shown in Fig. 5 (a), if ADC3 does not receive the input signal, there is no coincidence. The logic circuit generates the EXT REJ signal to reject a conversion action of ADC1 and ADC2.

When the coincidence has occurred between ADC1 and ADC2 as shown in Fig. 5 (b) and ADC2 is reset internally, ADC1 is reset by the CLEAR1 signal generated in the logic circuit if the conversion is complete, and ADC1 is reset by the EXT REJ1 signal if the conversion is under way.

Figure 6 shows the schematic diagram of logic circuit of the ADC controller. Transistor-transistor-logic integrated circuits (TTL IC) are used. For simplicity, the circuit connected to only one ADC is shown. The switch position shown in the figure indicates that one specified ADC (ADC1) is used. In opposite position that ADC is cut off from the controller logic completely.

Figure 7 shows the photograph of ADC controller and display unit. The 6 span module and 3 span module of the AEC NIM standard are used respectively.

Some slight modification of EXT REJ logic in the 50MHz ADC was done so as to work commonly with 100MHz ADC.

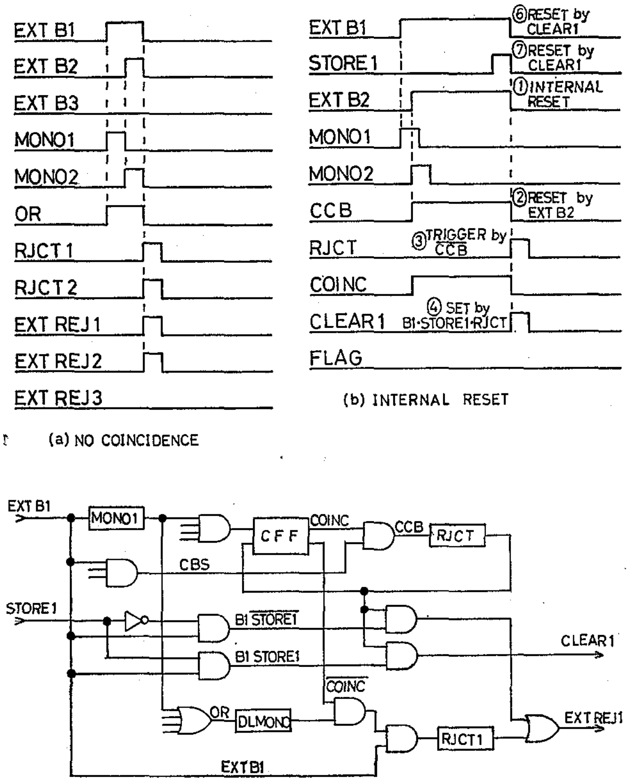

Fig. 5. Logic and the timing chart of the ADC controller in NO COINCIDENCE and INTERNAL RESET modes.

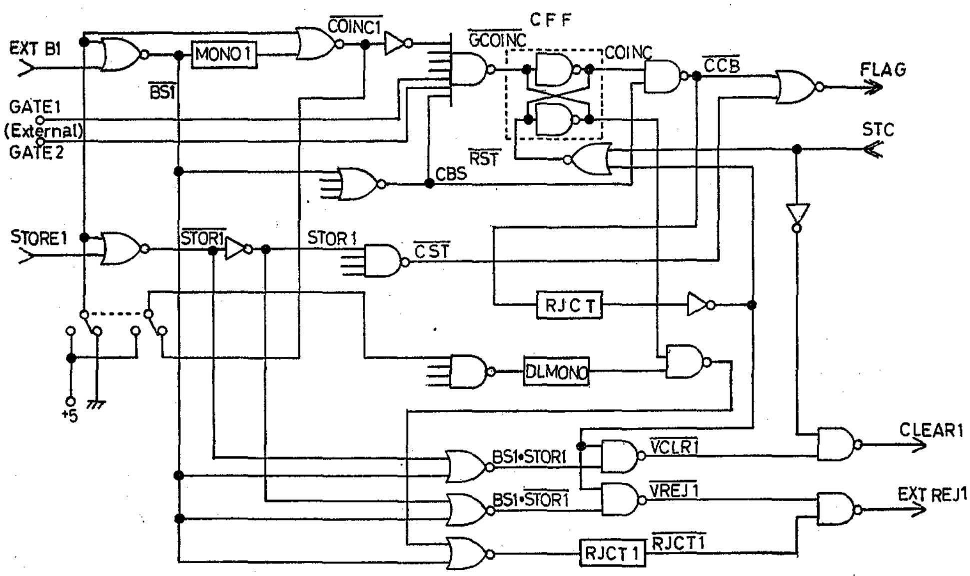

Fig. 6. Schematic diagram of the ADC controller.

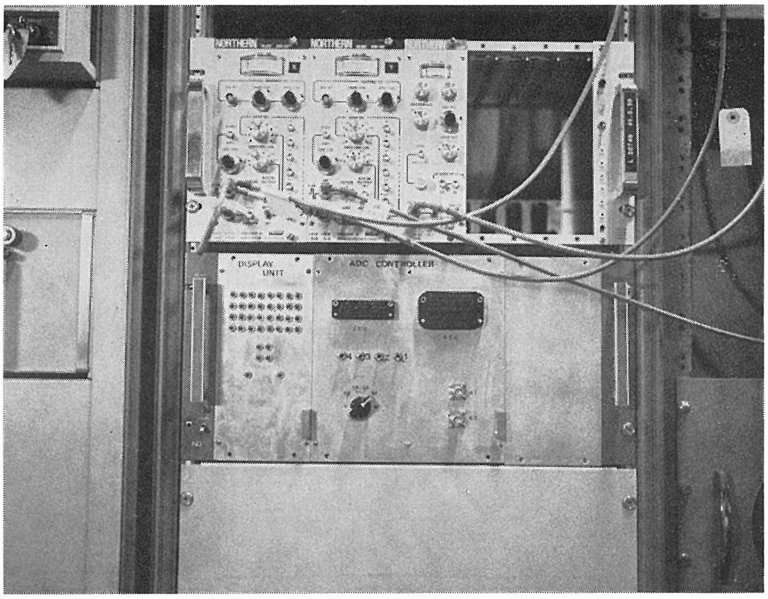

Fig. 7. ADC controller, display unit and ADC's.

2. Software for Multiparameter Event Recorder

The computer program named "Multiparameter List Mode Control" is composed of subprograms written in assembly language linkaged with FORTRAN programs. Each of subprograms has its own function, namely to store data from the ADC interfaces into core memory through CPU registers and to print out the data onto a paper tape or to display dual spectrum on the CRT. Program length is about 9k-words. During this program is running, one can always communicate with CPU through switch register to decide either ADC's are enabled or disabled. Moreover, one can decide, for example, the threshold of dual spectrum in contour representation. A flow chart is given in Fig. 8.

When ADC's receive signals and finish analog to digital conversions, flag signals are sent to CPU. These signals interrupt executing program under way and an interrupt processing part of the whole program instructs to store data from the interface data buffers into core memories. In the core memories, three blocks of data area are reserved, and one block has 508 words, corresponding to the area of 254 events. Software dead time is 51.94μsec. If one block of data area is filled, the next block is used. Contents of the previous data area are dumped onto paper tape during the CPU is free from data storing. The byte length of the tape corresponding to one block of data area is 1024 bytes and its first 8 bytes are assigned to control words and data identification.

When data storing and data dumping is not being done, this program instructs to display two dimensional spectrum on the CRT in contour representation. In this representation, the dimensions of the dislay are reduced into 64×64 irrespective

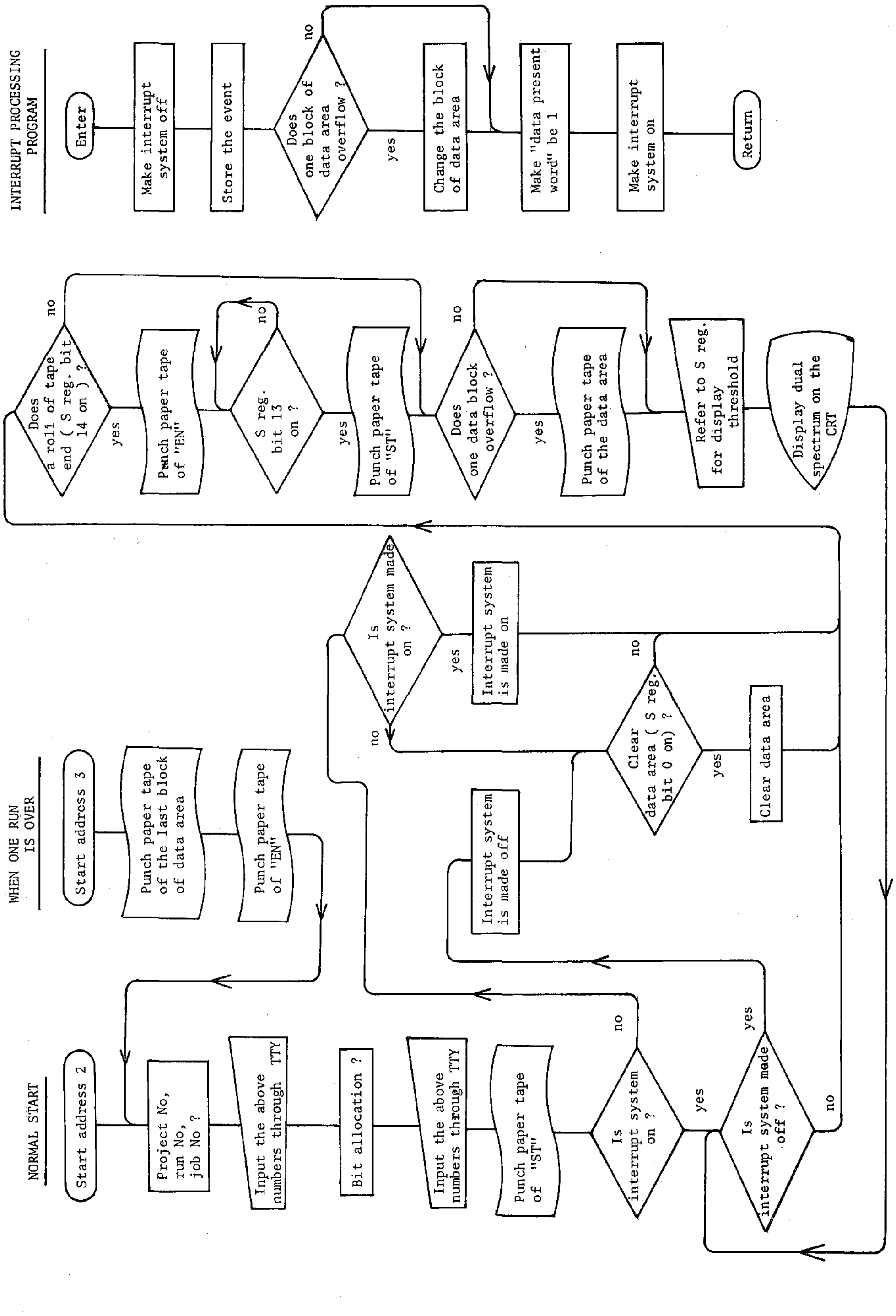

Fig. 8. Flow chart of the multiparameter event recorder program.

of original dimensions. Specified combination of display parameters are assigned by the teleprinter at the beginning of this program.

3. Data Analysis

The data obtained with this acquisition system are analysed by using the FACOM 230–48 system. The paper tape from this system are fed into the paper tape reader F749C. Then, the data are stored in the disk pack F478K of the FACOM 230–48 system and, when neccesary, the data are read out and analysed.

Two kinds of soft pragrams are necessary for the process mentioned above. The first one controls the read-out of data on the paper tape by the F749C and the write-in of data on the disk pack. The second program is used for the read-out of data from the disk and for the reduction of four parameters informations included in the paired two words. In the following, explanations are given for these programs.

(1) Read-write Program

The data on the tape are recorded binary to fit the processing by the FACOM 230–48 computer. The data on the tape are grouped into a number of sections each of which is of 512 words length. The first section is assigned to the header label and the last section to the trailer label. The read-write program control the processing of the FACOM 230–48 as follows. After detection of the header, every section is read one by one by FACOM 230–48 and then written on the designated part of the disk pack. When the trailer of the paper tape is detected, the trailer label is written on the last record of the disk and then the read-write process of a tape ends.

The FACOM 478K system has four disk packs each of which has a capacity of 100M bytes. 100 tracks are available for one project, that means about 3000K events can be stored. This memory capacity is sufficient to store the data of a series of particle-particle correlation experiments without suffering from urgent data analysis.

(2) Data Analysing Program

Each section of the data is read one by one from the disk to the core memory of FACOM 230–48 and analysed. Among 512 words of one section, the first 4 words contains the control words and the header label of the section and the remaining 508 words correspond to the data of 254 events. The header label represents the name of the data, that is, project number, job number, run number and buffer number of the experiment. These numbers are recorded on the paired two words and are reduced from these two words in the same manner as for the data reduction.

Four parameters of event are stored in the paired two words; the first word is designated with the bit 16 (most significant bit) equal to zero, and the second one the bit 16 equal to 1. The remaining 15 bits of the word can be assigned as follows, for example, 8 bits (from the bit 1 to 8) to the parameter 1 of 256 channels size and 7 bits (from the bit 9 to 15) to the parameter 2 of 128 channels size. In the FACOM 230–48 computer this word is considered as a number of 16 bits size. Then the parameter 1, in this example, is reduced from the number with dividing it by 2^8 and converted to an integer. The parameter 2 is obtained from the number of 16 bits size after subtraction of the number of parameter 1 multiplied by 2^8. The second word designated with the bit 16 equal to 1 is considered as a negative integer in the computer, Therefore this number is added by 2^{15} and subtracted by 2^{16} before analysis as the same manner as for the first word.

With these four parameter informations, one can reduct a spectrum of various types according to the request of the experimenter.

The list of these programs are described in the appendix.

4. Performance Test of the Over-All System with p-d Scattering

To test the over-all system, this system was applied to the elastic scattering experiment between protons and 13.0 MeV incident deuterons to measure the coincidence spectra of protons and deuterons. Polyethylene target was used.

The two emerging particles were detected in coincidence on opposite sides of the beam direction. The one detector is an SSD of 500μm thick and is E_1 detector, and the other detector is a counter telescope consisted of a transmission type SSD of 50μm thick, a ΔE_2 detector, and SSD of 100μm thick, E_2 detector. Three physical parameters were determined for each events; the energy of the first particle E_1, the energy of the second particle E_2 and its energy loss ΔE_2. Among 32 bits, 9 bits (512 channels), 8 bits (256 channels) and 7 bits (128 channels) were assingned to parameter E_1, E_2 and ΔE_2, respectively. The circuit block diagram is shown in Fig. 9. The number of coincident events (number of GATE) were counted by the scaler 1 and the number of transferred events from the ADC to the computer (number of Flag signal) were counted by the scaler 2.

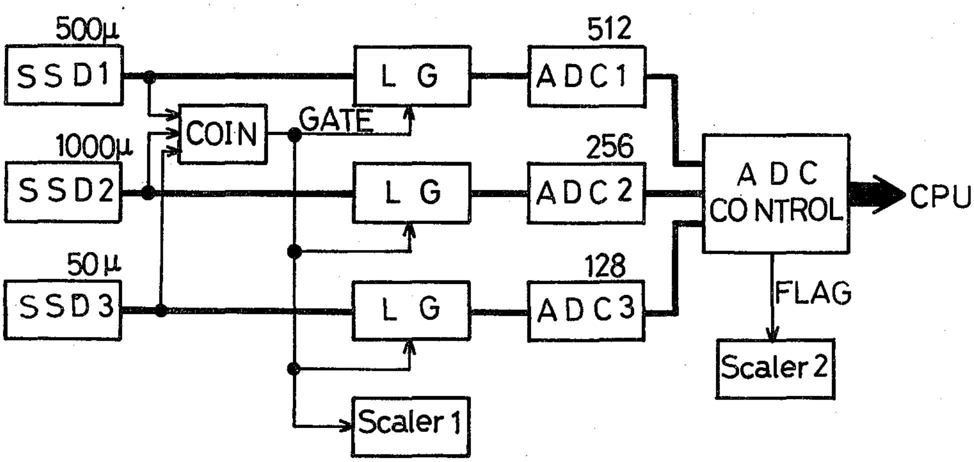

Fig. 9. Block diagram of the circuit used in the p-d scattering experiment.

Table I shows the number of coincident events, the number of transferred enents and the number of the events recorded on the paper tape and analysed by the FACOM 230–48 computer. The width of the coincidence time of the coincidence

Table I. Number of transferred events of p-d scattering

COINC TIME	GATE	FLAG	FINAL DATA
9.6 μ sec.	2596 (100)	2467 (95.0)	2468 (95.0)
3.6 μ sec.	3020 (100)	2317 (76.7)	2317 (76.7)

logic in the ADC controller is 9.6μsec or 3.6μsec. In the case of 9.6μsec, the number of transferred events are almost equal to the number of coincident events but in the case of 3.6μsec, these numbers are somewhat different with each other. This fact

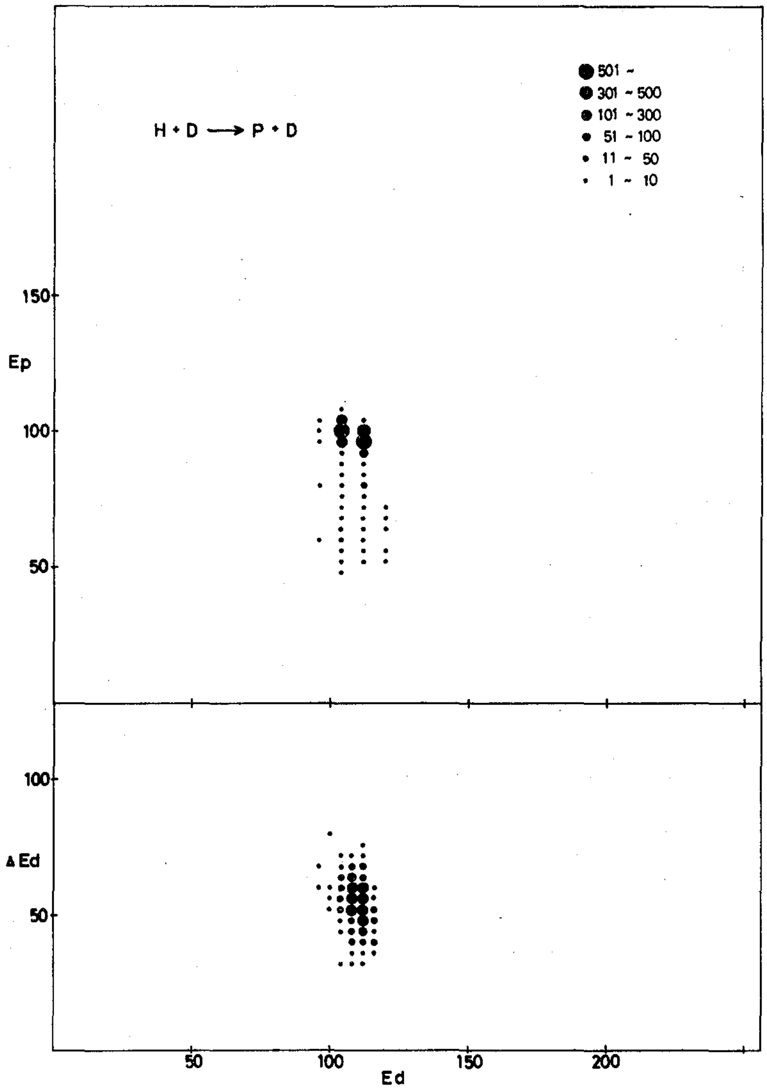

Fig. 10. Two dimensional spectra of p-d scattering.

is explained mainly by the differene of the input wave forms to ADC's. Therefore, it is better for the multi-parameter data acquisition system that the width of the coincidence time of the ADC controller is 9.6μsec or longer. The number of the events analysed by the FACOM 230–48 computer is exactly equal to the number of the transferred events in each case. A typical example of E_1–E_2 two dimensional spectrum and one of the E_2–ΔE_2 spectrum are shown in Fig. 10. These spectra were obtained by the analysis of the data on the paper tape by using the FACOM 230–48 computer. These results show that this system is applicable satisfactorily to the correlation experiments.

At present, the event rate is limited by the speed of the paper tape punch and is about 18 events/sec. Although, this limit is no objection to the correlation experiments.

With the use of a mass storage device such as a magnetic tape or a disk, event rate acceptable by this system can be enlarged extremely.

Acknowledgments

The authors are grateful to Mr. Y. Saito, Mr. Y. Kita and Dr. K. Kajiwara of the computer center of the Institute for Chemical Research for valuable suggestions and for their help in the use of FACOM 230–48. They also thank to Mr. K. Kajiya of Yokogawa-Hewlett-Packard Co. Ltd., and Mr. S. Ogawa of Seki & Company for helpful discussions on designing the ADC controller.

REFERENCES

(1) A Pocket Guide to the 2100 computer, Hewlett-Packard Company. 1972.
(2) A Pocket Guide to Interfacing the HP2100 computer, Hewlett-Packard Company. 1973.
(3) NS–623 Instruction Manual, and NS–622 Instruction Manual, Northern Scientific, Inc.

Appendix

1. Read-write progam SOURCE LIST

```
      OPTION FILE(1,1024,1024,2,PT,0),FILE(9,1022,1022,2,F)
      DIMENSION IDATA(511)
      DATA IPTHED/ZE2E3/
      DATA IPTEND/ZC5D5/
      DATA ISTEND/ZF0F0/
      DATA K/6/
      REWIND 9
      GO TO 10
    1 READ(9)(IDATA(I),I=1,511)
      WRITE(6,201)(IDATA(I),I=1,3)
  201 FORMAT(1H0,3Z5)
      IF(IDATA(1).EQ.ISTEND) GO TO 2
      GO TO 1
    2 K=K-1
   10 IF(K.EQ.0) GO TO 3
      GO TO 1
    3 READ(1)(IDATA(I),I=1,511)
      IF(IDATA(1).EQ.IPTHED) GO TO 4
      GO TO 500
    4 READ(1)(IDATA(I),I=1,511)
      IF(IDATA(1).EQ.IPTEND) GO TO 5
      WRITE(9)(IDATA(I),I=1,511)
      WRITE(6,100)(IDATA(I),I=1,3)
  100 FORMAT(1H0,3Z5)
      WRITE(6,101)(IDATA(I),I=4,511)
  101 FORMAT(1H ,26Z5)
      GO TO 4
    5 IDATA(1)=ISTEND
      WRITE(9)(IDATA(I),I=1,511)
  500 STOP
      END
```

2. Data analysing program SOURCE LIST

```
      OPTION FILE(9,1022,1022,2,F)
      DIMENSION IDATA(600)
      DIMENSION IMAT3(32,64)
      DIMENSION IMAT4(32,64)
      DIMENSION ICH1(32)
      DIMENSION ICH2(64)
      DIMENSION ICH3(32)
      DIMENSION ICH4(64)
      INTEGER*4 IBATA(600)
      INTEGER*4 IPROJN,IJOBN,IRUNN,IBUFFN,IP1,IP2,IP3,IP4
      DATA ISTAR/Z5C5C/
      DATA ISTEND/ZF0F0/
      DATA IENDN/7/
      REWIND 9
   11 DO 50 I=1,32
      DO 50 J=1,64
   50 IMAT3(I,J)=0
      DO 52 I=1,32
      DO 52 J=1,64
   52 IMAT4(I,J)=0
   10 READ(9)(IDATA(I),I=1,511)
      IF(IDATA(1).EQ.ISTEND) GO TO 300
      IF(IDATA(1).GE.0) GO TO 60
      IBATA(1)=IDATA(1)+2**16
      GO TO 61
```

(continued)

```
   60 IBATA(1)=IDATA(1)
   61 IF(IDATA(2).GE.0) GO TO 62
      IBATA(2)=IDATA(2)+2**16
      GO TO 20
   62 IBATA(2)=IDATA(2)
   20 IPROJN=IBATA(1)/2**8
      IJOBN=IBATA(1)-IPROJN*(2**8)
      IRUNN=IBATA(2)/2**8
      IBUFFN=IBATA(2)-IRUNN*(2**8)
      DO 200 I=2,255
      IBATA(2*I)=IDATA(2*I)
      IBATA(2*I+1)=IDATA(2*I+1)+2**16
      IP1=IBATA(2*I)/2**8
      IP2=IBATA(2*I)-IP1*2**8
      IP1=IP1/4+1
      IP2=IP2/4+1
      IMAT3(IP1,IP2)=IMAT3(IP1,IP2)+1
      IP3=IBATA(2*I+1)/2**9
      IP4=IBATA(2*I+1)-IP3*2**9
      IP2=IP2/2+1
      IP4=IP4/8+1
  200 IMAT4(IP2,IP4)=IMAT4(IP2,IP4)+1
      GO TO 10
  300 DO 210 I=1,32
      ISUM1=0
      ISUM2=0
      DO 220 J=1,64
      ISUM1=ISUM1+IMAT3(I,J)
  220 ISUM2=ISUM2+IMAT4(I,J)
      ICH1(I)=ISUM1
  210 ICH3(I)=ISUM2
      DO 230 J=1,64
      ISUM3=0
      ISUM4=0
      DO 240 I=1,32
      ISUM3=ISUM3+IMAT3(I,J)
  240 ISUM4=ISUM4+IMAT4(I,J)
      ICH2(J)=ISUM3
  230 ICH4(J)=ISUM4
      ISUM5=0
      ISUM6=0
      DO 250 I=1,32
      ISUM5=ISUM5+ICH1(I)
  250 ISUM6=ISUM6+ICH3(I)
      WRITE(6,100) IPROJN,IJOBN,IRUNN,IBUFFN
      DO 80 J=1,64
   80 WRITE(6,150)(IMAT3(I,J),I=1,32),ICH2(J)
      WRITE(6,150)(ICH1(I),I=1,32),ISUM5
      WRITE(6,100) IPROJN,IJOBN,IRUNN,IBUFFN
      DO 70 J=1,64
   70 WRITE(6,150)(IMAT4(I,J),I=1,32),ICH4(J)
      WRITE(6,150)(ICH3(I),I=1,32),ISUM6
  100 FORMAT(1H1,4I10)
  150 FORMAT(1H ,32I4,1I6)
      IENDN=IENDN-1
  400 IF(IENDN.EQ.0) GO TO 500
      GO TO 11
  500 STOP
      END
```