ORNL-3235
UC-32 – Mathematics and Computers

GETTING MULTICHANNEL ANALYZER DATA IN AND OUT OF THE IBM-7090 FOR PROCESSING

C. D. Goodman

OAK RIDGE NATIONAL LABORATORY
operated by
UNION CARBIDE CORPORATION
for the
U.S. ATOMIC ENERGY COMMISSION

DISCLAIMER

DISCLAIMER

Printed in USA. Price $0.50. Available from the
Office of Technical Services
Department of Commerce
Washington 25, D.C.

ORNL-3235

Contract No. W-7405-eng-26

ELECTRONUCLEAR RESEARCH DIVISION

GETTING MULTICHANNEL ANALYZER DATA IN AND OUT OF THE IBM-7090 FOR PROCESSING

C. D. Goodman

DATE ISSUED

DEC 12 1961

OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee
Operated by
UNION CARBIDE CORPORATION
for the
U. S. ATOMIC ENERGY COMMISSION

GETTING MULTICHANNEL ANALYZER DATA IN AND OUT OF THE IBM-7090 FOR PROCESSING

C. D. Goodman

ABSTRACT

The present method used for handling multichannel-analyzer data at the ORNL 86-Inch Cyclotron is stated. FORTRAN subroutines for reading the analyzer data into the IBM-7090 computer and for printing out the processed data and punching processed data on cards are presented.

INTRODUCTION

The accumulation of experimental nuclear data with multichannel pulse-height analyzers is now a commonplace technique. Experiments have become more complicated, and the number of channels on the available analyzers has increased to the point where machine handling of the data is imperative if the experimenter is to avoid becoming lost in the drudgery of numerical manipulations.

High-speed digital computing machines are very well suited for handling multichannel-analyzer data, and it is the purpose of this report to discuss the means of getting data in and out of such machines.

TRANSMISSION OF DATA FROM THE ANALYZER TO THE COMPUTER

The first step in machine handling of data is to obtain the analyzer output in a form that can be read by the computer or its peripheral equipment. At Oak Ridge, if the data are to be processed by the IBM-7090, the data must be supplied as punched cards for the off-line card reader, or as a magnetic tape for direct reading by the 7090. No equipment is currently available for the reading of punched paper tape except by way of tape-to-card conversion. Equipment for conversion from paper tape to magnetic tape is expected to be available in the future.

At the ORNL 86-Inch Cyclotron, data cards are produced on an IBM-24 card punch which is attached to a 400-channel analyzer. Each card contains the counts for ten channels punched in six-column fields. This information occupies the first 60 columns of the card. The channel number of the last channel on the card is punched in columns 65-67, and a deck number for identification is punched in columns 73-75. These cards are physically transported to the computer for processing.

The 20,000-channel analyzer, which is under construction, will be compatible with this card punching system; the card system will be used when the readout does not exceed a few hundred channels. When the full 20,000 channels or a major fraction of them must be read out, however, the card punch system is too slow. For this situation it is proposed to write the analyzer output directly onto magnetic tape in the same format which would be produced by card-to-tape conversion on the IBM equipment. This format will make it possible to use the usual FORTRAN "read" and "format" statements which are used for reading data from cards. The tape reel will be physically transported to the computer for processing.

Some consideration is being given to remote data transmission, and a card-to-card system is being installed on a trial basis between the ORIC building and the IBM-7090 facility.

PROGRAMMING THE IBM-7090 FOR DATA HANDLING

Most of the data reduction at the 86-Inch Cyclotron is done with the CONDAC program,[1] which is fairly complex. It is desirable to have, in addition, a means by which the experimenter, who is assumed not to be an expert programmer, can use the IBM-7090 to perform arithmetic operations on the data without calling upon a professional programmer to write a special program.

The FORTRAN system provides a simple means of writing arithmetic programs, but the input and output statements usually account for most of the programming effort. The input and output statements can, however, be written once as subroutines and used over and over again for each new problem. Two FORTRAN subroutines are presented here which take care of the input and output operations. These enable one to write data-handling programs without being concerned about format details.

THE DATAIN SUBROUTINE

The subroutine to read in the data is called DATAIN. The FORTRAN listing is shown in Fig. 1. This subroutine reads the data from a specified number of cards and arranges the data in a subscripted array suitable for subsequent processing; it reads the channel numbers from the cards so that the order of the cards within a deck is immaterial.

Two cards must precede the data deck to be read by DATAIN. The first card is a heading which may contain up to 72 Hollerith characters; the second card may contain 10 numbers to be used in the subsequent processing. These numbers are to be placed in the first ten six-column fields in the same format as the data numbers. The decimal point is assumed to be at the right-hand edge of the field unless it is explicitly punched. The first number is the number of data cards in the deck to be read. The second number is the zero offset to be added to the

---

[1] C. D. Goodman and B. D. Williams, ORNL Report 2925 (1960).

```
      SUBROUTINE DATAIN
C     READ DATA CARDS
      DIMENSION A(10,52),K(52),C(520),CHANNO(520),T(12)
      CCMMON C    ,CHANNO    ,T,P1,P2,P3,P4,P5,P6,P7,P8,NDECK,N,MINK
   49 READ INPUT TAPE 10,1,T,N,OFST,P1,P2,P3,P4,P5,
     1P6,P7,P8,(A(I,1),I#1,10),K(1),NDECK,((A(I,J),I#1,10),
     2K(J),J#2,N)
C     ORDER CHANNELS
      DC 11 J#1,N
      DC 11 I#1,10
      L#K(J)+I-9
      C(L)#A(I,J)
      FLTL#L
   11 CHANNC(L)#FLTL+OFST-1.0
      MINK#K(1)
      DC 34 I#2,N
      IF (MINK-K(I))34,34,35
   35 MINK#K(I)
   34 CONTINUE
C     BACKGROUND SUBTRACTION OR SUMMATION
   17 READ INPUT TAPE 10,5,P9,KSHIFT
      IF (P9)15,16,15
   15 READ INPUT TAPE 10,6,((A(I,J),I#1,10),K(J),J#1,N)
      DC14J#1,N
      DC14I#1,10
      L#K(J)+I+KSHIFT-9
   14 C(L)#C(L)+A(I,J)*P9
      GO TO 17
   16 RETURN
    1 FCRMAT(12A6/I6,9F6.0/10F6.0,4XI3,5XA3/(10F6.0,4XI3))
    5 FORMAT(F6.0,I6)
    6 FCRMAT(10F6.0,4XI3)
      END(1,1,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

Fig. 1. FORTRAN Listing for Subroutine DATAIN.

channel numbers on the data cards to obtain the true channel number. The other eight numbers are not used in the input program but are merely placed in common storage to be accessible for later computation. These are named P1, P2, ... P8. The use of the number in the third field (P1) is, however, restricted in that it is tested by subroutine PUTOUT to determine the choice of punched card options. For details, see the description of subroutine PUTOUT.

If several decks are to be added to or subtracted from each other, each deck after the first must be preceded by a card containing two numbers. The number in the first field is the normalization factor by which the data numbers in the following deck are to be multiplied before they are added to the first deck. This number may be positive or negative, and the decimal point is assumed to be at the right-hand edge of the field unless it is punched elsewhere.

The number in the second field is a positive or negative integer or zero which must be added to the channel numbers on the following deck to obtain correspondence with the channel numbers on the first deck. This number would be used, for example, when two spectra, taken in different subgroups of a multichannel analyzer, are added together. The sample program in Fig. 2 illustrates this point. A blank card must be placed after the last data card. Reading of this card will return control to the main program.

All decks to be combined must have the same number of cards. If it is necessary to fill out a deck with dummy cards, the first deck may be filled out with duplicates of any card. Subsequent decks may be filled out with cards which contain duplicate channel numbers but are otherwise blank. In the first deck, if two or more cards have the same channel number, the last one read takes precedence. This feature makes it easy to correct faulty data by the insertion of additional information. DATAIN, together with PUTOUT, may be used to generate a correct data deck from one in which some channels were skipped or punched twice.

After the DATAIN subroutine returns control to the main program the correctly ordered data is in common storage and may be referred to by another FORTRAN program by way of a common statement identical

```
      SUBROUTINE PUTOUT
      DIMENSION C(520),CHANNO(520),T(12),NC(10)
      COMMON C    ,CHANNO    ,T,P1,P2,P3,P4,P5,P6,P7,P8,NDECK,N,MINK
   34 L1=MINK-8
      LMAX=L1+10*N-1
   40 M=1
      WRITE OUTPUT TAPE 9,3,T,N,P1,P2,P3,P4,P5,P6,P7,P8
   41 L2=L1+9
      WRITE OUTPUT TAPE 9,4,(C(L),L=L1,L2),
     1CHANNO(L2),NDECK
      IF (P1) 46,47,48
   46 DO 12 I=1,10
      J=L1+I-1
      NC(I)=C(J)
   12 NCHAN=CHANNO(L2)
      WRITE OUTPUT TAPE 6,7,(NC(I),I=1,10),NCHAN,NDECK
      GO TO 47
   48 L3=L1+4
      WRITE OUTPUT TAPE 6,8,((C(L),CHANNO(L),L=L1,L3),NDECK),((C(L+5),
     1CHANNO(L+5),L=L1,L3),NDECK)
   47 IF (L2-LMAX) 42,43,43
   42 L1=L1+10
      M=M+1
      IF(M-23)41,40,40
   43 RETURN
    3 FORMAT(1H112A6/1H0I6,8F8.2)
    4 FORMAT(1H010F9.2,3XF5.1,2XA3)
    7 FORMAT(10I6,4XI3,5XA3,9X)
    8 FORMAT(10F7.2, 2XA3,9X)
      END(1,1,0,0,0,0,0,0,0,0,0,0,0,0,0)
```

Fig. 2. FORTRAN Listing for Subroutine PUTOUT.

to the common statement in the DATAIN subroutine. The data appear in the form of two subscripted arrays, C(L) and CHANNO(L). C(L) is the number of counts in channel number CHANNO(L). CHANNO(L) includes the specified zero offset. L is a subscript which is, in fact, the channel number, as read from the first deck, plus one.

THE PUTOUT SUBROUTINE

The PUTOUT subroutine causes the arrays C(L), CHANNO(L), and the input parameters to be printed on a page. The FORTRAN listing is shown in Fig. 3. The printout numbers are the values of all these variables after the operation of the main program. The spaces reserved for P1, P2, etc., may be used to printout the input parameters or the results of computations. The sign of P1 is examined by the program to determine the card punching option. If P1 is negative at the time it is examined by the subroutine, output cards will be punched in the same format as the input cards. If P1 is positive, output cards will be punched with alternate 7-column fields containing CHANNO(L) and C(L), five channels per card. If P1 is zero, no output cards will be punched.

The heading and the line containing the input parameters are repeated on each new page of output. The number of lines of output equals the number of cards in the first deck of input data.

A sample program using DATAIN and PUTOUT is shown with sample input data in Fig. 3. The corresponding printed output is shown in Fig. 4. A listing of the corresponding output cards is shown in Fig. 5.

```
*ID   C.D.GOODMAN    Y-12    9201-2    TEL-7398
*CHARGE(1588)
*TAPE(10,INPUT)(9,OUTPUT)(6,AUXOUT)
*EXECUTE
*TYPE(FORTRAN)

  DIMENSION C(520),CHANNO(520),T(12)
  COMMON C,CHANNO,T,P1,P2,P3,P4,P5,P6,P7,P8,NDECK
4 CALL DATAIN
  DO 1 L#1,520
  IF (C(L))3,3,2
2 C(L)#LOGF(C(L))
  GO TO 1
3 C(L)#0.0
1 CONTINUE
  CALL PUTOUT
  GO TO 4
  END(1,1,0,0,0,0,0,0,0,0,0,0,0,0,0)
*TYPE(BINARY)

THE BINARY DECKS FOR SUBROUTINE DATAIN AND SUBROUTINE PUTOUT ARE

INSERTED HERE.

*DATA

SUBTRACT BACKGROUND, TAKE LOGARITHM OF COUNTS.  CASE 1, RUN 1

   10  20.5     1

00000 18297 01442 00846 00629 00569 00544 00546 00579 00522    010      254
00526 00487 00523 00546 00525 00527 00490 00485 00510 00493    020      254
00524 00491 00497 00440 00489 00474 00489 00475 00477 00516    030      254
00486 00473 00464 00466 00503 00489 00456 00480 00477 00491    040      254
00510 00491 00517 00521 00512 00523 00502 00539 00555 00593    050      254
00583 00572 00551 00477 00487 00514 00484 00473 00500 00443    060      254
00460 00441 00434 00390 00408 00420 00434 00513 00550 00601    070      254
00470 00516 00682 00624 00432 00415 00577 01191 01202 00599    080      254
00407 00416 00374 00463 00818 00990 00603 00955 05533 16926    090      254
16247 04732 00489 00094 00060 00030 00028 00029 00030 00000    100      254
  -.2  -100
00465 05630 00117 00073 00057 00049 00039 00064 00060 00050    110      254
00052 00041 00047 00053 00073 00078 00095 00099 00096 00088    120      254
00062 00106 00172 00222 00171 00156 00185 00198 00317 00442    130      254
00191 00064 00057 00108 00103 00055 00008 00013 00016 00027    140      254
00012 00016 00004 00005 00012 00045 00187 00277 00138 00029    150      254
00015 00008 00006 00006 00002 00006 00006 00001 00000 00000    160      254
00000 00000 00000 00000 00000 00000 00000 00000 00000 00000    170      254
00000 00000 00000 00000 00000 00000 00000 00000 00000 00000    180      254
00000 00000 00002 00000 00000 00000 00000 00000 00000 00000    190      254
00000 00001 00001 00000 00000 00000 00000 00000 00000 00000    200      254
                                                                        254
```

Fig. 3. Sample Program and Data for the IBM-7090, as set up for compilation and execution on the Oak Ridge monitor. The program subtracts 0.2 times data number in second deck from the corresponding data number in the first deck and computes the natural logarithm of the difference.

SUBTRACT BACKGROUND, TAKE LOGARITHM OF COUNTS. CASE 1, RUN 1

10   1.00   -0.   -0.   -0.   -0.   -0.   -0.   -0.

0.	9.75	7.26	6.72	6.43	6.33	6.28	6.28	6.34	6.24	30.5	254
6.25	6.17	6.24	6.28	6.24	6.24	6.15	6.14	6.20	6.16	40.5	254
6.24	6.15	6.14	5.98	6.12	6.09	6.11	6.08	6.02	6.06	50.5	254
6.10	6.13	6.12	6.10	6.18	6.17	6.12	6.17	6.16	6.19	60.5	254
6.23	6.19	6.25	6.25	6.23	6.24	6.14	6.18	6.27	6.38	70.5	254
6.36	6.35	6.31	6.16	6.19	6.24	6.18	6.16	6.21	6.09	80.5	254
6.13	6.09	6.07	5.97	6.01	6.04	6.07	6.24	6.31	6.40	90.5	254
6.15	6.25	6.53	6.44	6.07	6.03	6.36	7.08	7.09	6.40	100.5	254
6.01	6.03	5.92	6.14	6.71	6.90	6.40	6.86	8.62	9.74	110.5	254
9.70	8.46	6.19	4.54	4.09	3.40	3.33	3.37	3.40	0.	120.5	254

Fig. 4. Printed Output from Sample Program of Fig. 3.

LISTING OF OUTPUT CARDS

0.	21.50	9.75	22.50	7.26	23.50	6.72	24.50	6.43	25.50	254
6.33	26.50	6.28	27.50	6.28	28.50	6.34	29.50	6.24	30.50	254
6.25	31.50	6.17	32.50	6.24	33.50	6.28	34.50	6.24	35.50	254
6.24	36.50	6.15	37.50	6.14	38.50	6.20	39.50	6.16	40.50	254
6.24	41.50	6.15	42.50	6.14	43.50	5.98	44.50	6.12	45.50	254
6.09	46.50	6.11	47.50	6.08	48.50	6.02	49.50	6.06	50.50	254
6.10	51.50	6.13	52.50	6.12	53.50	6.10	54.50	6.18	55.50	254
6.17	56.50	6.12	57.50	6.17	58.50	6.16	59.50	6.19	60.50	254
6.23	61.50	6.19	62.50	6.25	63.50	6.25	64.50	6.23	65.50	254
6.24	66.50	6.14	67.50	6.18	68.50	6.27	69.50	6.38	70.50	254
6.36	71.50	6.35	72.50	6.31	73.50	6.16	74.50	6.19	75.50	254
6.24	76.50	6.18	77.50	6.16	78.50	6.21	79.50	6.09	80.50	254
6.13	81.50	6.09	82.50	6.07	83.50	5.97	84.50	6.01	85.50	254
6.04	86.50	6.07	87.50	6.24	88.50	6.31	89.50	6.40	90.50	254
6.15	91.50	6.25	92.50	6.53	93.50	6.44	94.50	6.07	95.50	254
6.03	96.50	6.36	97.50	7.08	98.50	7.09	99.50	6.40	100.50	254
6.01	101.50	6.03	102.50	5.92	103.50	6.14	104.50	6.71	105.50	254
6.90	106.50	6.40	107.50	6.86	108.50	8.62	109.50	9.74	110.50	254
9.70	111.50	8.46	112.50	6.19	113.50	4.54	114.50	4.09	115.50	254
3.40	116.50	3.33	117.50	3.37	118.50	3.40	119.50	0.	120.50	254

Fig. 5. Listing of Output Cards from Sample Program of Fig. 3.

ORNL-3235
UC-32 — Mathematics and Computers
TID-4500 (16th ed., Rev.)

INTERNAL DISTRIBUTION

1. Biology Library
2-3. Central Research Library
4. Reactor Division Library
5. ORNL — Y-12 Technical Library, Document Reference Section
6-25. Laboratory Records Department
26. Laboratory Records, ORNL R.C.
27. J. B. Ball
28. R. S. Bender
29. G. E. Boyd
30. F. L. Culler
31. C. D. Goodman
32. J. B. Davidson
33. J. H. Frye, Jr.
34. C. B. Fulmer
35. M. L. Halbert
36. C. S. Harrill
37. A. Hollaender
38. A. S. Householder
39. F. T. Howard
40. R. J. Jones
41. R. G. Jordan (Y-12)
42. W. H. Jordan
43. M. T. Kelley
44. J. A. Lane
45. C. E. Larson
46. T. A. Lincoln
47. S. C. Lind
48. R. S. Livingston
49. J. A. Martin
50. K. Z. Morgan
51. J. P. Murray (K-25)
52. M. L. Nelson
53. E. Newman
54. G. D. O'Kelley
55. J. J. Pinajian
56. D. Phillips
57. H. E. Seagren
58. E. D. Shipley
59. M. J. Skinner
60. D. M. Smith
61. A. H. Snell
62. J. A. Swartout
63. E. H. Taylor
64. A. M. Weinberg
65. B. D. Williams (K-25)
66. R. E. Worsham
67. A. Zucker
68. H. Feshbach (consultant)
69. W. A. Fowler (consultant)
70. M. Goldhaber (consultant)
71. M. S. Livingston (consultant)
72. J. R. Richardson (consultant)
73. J. H. Van Vleck (consultant)
74. E. P. Wigner (consultant)

EXTERNAL DISTRIBUTION

75. Division of Research and Development, AEC, ORO
76. D. A. Bromley, Department of Physics, Yale University, New Haven, Connecticut
77-690. Given distribution as shown in TID-4500 (16th ed., Rev.) under Mathematics and Computers category (75 copies - OTS)