X-735-66-385

NASA TM X- 55619

COMPUTER PROGRAM FOR ANALYZING BATTERY PERFORMANCE DATA

GPO PRICE $ ____________

CFSTI PRICE(S) $ ____________

Hard copy (HC) 6.00

Microfiche (MF) 1.25

ff 653 July 65

BY

G. HALPERT

NASA

GODDARD SPACE FLIGHT CENTER
GREENBELT, MARYLAND

RQ7-42348

X-735-66-385

COMPUTER PROGRAM

FOR

ANALYZING BATTERY PERFORMANCE DATA

by

G. Halpert
Goddard Space Flight Center
Greenbelt, Maryland

CONTENTS

COMPUTER PROGRAM

FOR

ANALYZING BATTERY PERFORMANCE DATA

by

G. Halpert
Goddard Space Flight Center
Greenbelt, Maryland

N67-13164

ABSTRACT

A system of computer programs (routines) for processing and analyzing battery and other time-dependent data is presented. The first sections describe the format of the data on punched cards and the mechanism for converting the cards to magnetic storage tape. The third and major section describes the individual analysis routines, instructions to the analyst and computer operator on their use, and the type of output from each. These routines include a means for reducing data, performing mathematical operations, listing, plotting in graphical form, and statistically analyzing the data. Programmers information is found in the Appendices.

author

COMPUTER PROGRAM

FOR

ANALYZING BATTERY PERFORMANCE DATA

by

G. Halpert
Goddard Space Flight Center

INTRODUCTION

A System of Battery Analysis Programs (SBAP) has performed satisfactorily in the processing and the analyzing of battery performance data. This report which describes the programs was specifically written so that the battery engineer or analyst with little or no knowledge in computer operations can put the system into practical use.

The need for the SBAP became evident when evaluating the test results of the nickel-cadmium battery research project (NAS-5-3027) initiated by GSFC in 1963. In this project twelve 10-cell nickel cadmium batteries were tested for prolonged periods in related but different types of operations. Performance characteristics such as current, cell voltages and average power were measured during the charge and discharge phases. Because of the complexity of the test, an IBM 1710 computer was used to control the operations and record the measurements. To reduce the volume of data, measurements made during the charge and discharge phases were recorded only when certain parameters were exceeded. Measurements representing compiled data were recorded at the end of every phase.

The mathematical and statistical data evaluation using the SBAP assists in the determination of relationships between battery operation and battery life, which is the primary objectives of the Battery Research Project. In writing the SBAP, an effort was made to make it flexible enough to be used by anyone with battery data or for any data of a similar time-dependent nature.

The first part of this report describes the arrangement (format) of the battery test data on the punched cards. The information provided in this section should be considered by the battery engineer before designing a battery test if he intends to utilize a computer program for subsequent analysis of the resulting data. The second part describes the SBAP, the five routines comprising it, what the functions of the routines are, how they are used, and what instructions should be given to the computer operator in order to process them. Appendix A supplies information needed to transfer raw data from cards to magnetic tapes. Appendices B and C contain programming information such as source statements and flow charts for all routines mentioned.

The programs were written for the IBM 7094 computer system at GSFC and are usable with other computers containing a comparable storage capacity, and the ability to utilize FORTRAN II language. The computer time required for each processing depends on the routines used, the quantity of data extracted from the raw data tapes, and the operations to be performed. The average time for each routine is given under operator instructions.

INPUT DATA AND FORMAT REQUIRED

In order to analyze battery performance or similar types of time-dependent data, the data must be punched onto cards in a specific order (format) so the computer can process and store the data on magnetic storage tape. In the process, the data is sorted, edited, and brought up to date. The raw data on tape in binary form are then ready to be used as input data to the SBAP.

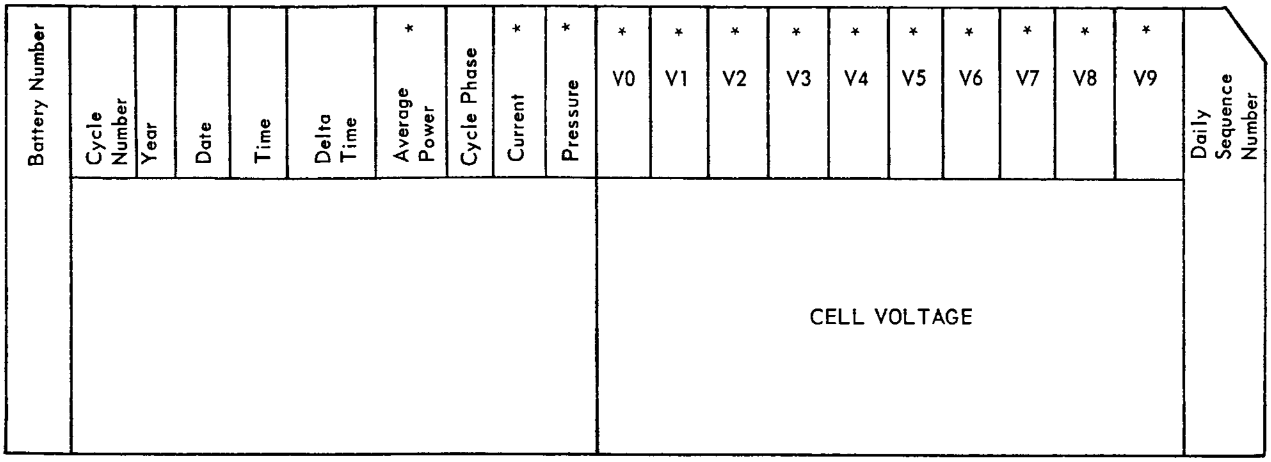
There are two types of data available from a performance test: outcard data, obtained during the charge and discharge phase, and endcard data, taken at the end of the phase and representing a compilation of the data. A separate tape is required for each type. Depending on the analyst's request an endcard raw data tape, an outcard raw data tape, or a combination of the two is supplied as input to the battery analysis program.

Outcard Format

An outcard is divided into 21 groups, and each represents a specific measurement. The name of the group, the number of digits allowed, the location of the decimal point, and a sample and its meaning are given in Figure 1. An important consideration for those who are designing a test is to decide which information should be included. Any outcard type measurement can be punched into the card in a space allotted for a group with an asterisk, provided that the number of digits and the position of the decimal point are the same as in the group it replaced. For example, if ten temperature measurements are made on one cell using thermocouples, the millivolt values can be punched into the spaces reserved for V0 to V9.

Endcard Format

Examples of endcards are given in Figures 2 and 3. Two endcards are provided to accommodate additional data. Included are the group names, the number of digits allowed, the location of the decimal point, and the sample and its meaning. As with outcards, any measurements of endcard data can be punched into the spaces reserved for the asterisked groups. The number of digits and decimal point restrictions must be taken into account.



OUTCARD

No.	Groups	Digit and decimal designation	Sample	Units or meaning
1	Battery No.	XX	01	Battery 1
2	Cycle No.	XXXX	2765	Cycle 2765
3	Year	X	6	1966
4	Day	XXX	365	Dec. 31
5	Time	XX.XXXX		Hours
6	Delta Elapsed Time	X.XXXX		Hours
7	Avg. Instantaneous power*	XX.XXX		Watts
8	Phase	X	0	Charge
9	Current*	XX.XXX		Amperes
10	Pressure*	XX.XX		Millivolts
11	Voltage of cell 0*	X.XXX		Volts
12	Voltage of cell 1*	X.XXX		Volts
13	Voltage of cell 2*	X.XXX		Volts
14	Voltage of cell 3*	X.XXX		Volts
15	Voltage of cell 4*	X.XXX		Volts
16	Voltage of cell 5*	X.XXX		Volts
17	Voltage of cell 6*	X.XXX		Volts
18	Voltage of cell 7*	X.XXX		Volts
19	Voltage of cell 8*	X.XXX		Volts
20	Voltage of cell 9*	X.XXX		Volts
21	Sequence No.	XXXX	0032	32nd Card.

Note 1: Each of the groups labeled with an asterisk can be replaced by another group if the new data has the same number of digits and the same decimal point location. The new group must also be an outcard data group (described in text).

Note 2: The sequence numbers for outcards must start with "0000" or "3000." The card to tape editing program checks these numbers to assure that outcard data is put on an outcard data tape.

Figure 1—Format of Outcarcs

Battery Number	Cycle Number	Year	Date	Time Duration	* Ampere Minutes	* Watt Minutes	* Energy Efficiency	* Ampere Hours	* Watt Hours	* Average Current	* Minimum Current	* Maximum Current	* Average Power	* Minimum Power	* Maximum Hours	* Current Efficiency	Endcard No.	Cycle Phase	Daily Sequence Number

ENDCARD NO. 1

Groups	Digit and decimal designation	Sample	Units or Meaning
Battery No.	XX	01	Battery 1
Cycle No.	XXXX	2765	Cycle 2765
Year	X	6	1966
Date	XXX	365	Dec. 31
Time duration	XX.XXXX		Hours
Ampere minutes*	XXX.XXX		Amp. minutes
Watt minutes*	XXX.XXX		Watt minutes
Efficiency (Energy)*	XX.X		
Ampere hours*	XX.XXX		Amp hours
Watt hours*	XX.XXX		Watt hours
Average current*	XX.XXX		Amperes
Minimum current*	XX.XXX		Amperes
Maximum current*	XX.XXX		Amperes
Average power*	XX.XXX		Watts
Minimum power*	XX.XXX		Watts
Maximum power*	XX.XXX		Watts
Efficiency (Current)*	XX.X		Amperes
Endcard No.	X	1	Endcard 1
Phase	X	1	Discharge
Sequence No.	XXXX	6004	9th card

Note 1: Each of the groups designated with an asterisk can be replaced by another group if the new data has the same number of digits and same decimal point location. The new group must be endcard data.

Note 2: The sequence numbers for endcards must start with "6000" or "8000." The card to tape editing program checks these numbers to assure that the endcard data is put on an endcard data tape.

Figure 2—Format of endcard 1

Battery Number	Cycle Number	Year	Date	* E I Ratio	* C0	* C1	* C2	* C3	* C4	* C5	* C6	* C7	* C8	* C9	Max. Variance	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	* Real Time	Endcard No.	Cycle Phase	Daily Sequence Number
					CELL CYCLE COUNT											CELL ACTIVITY													

ENDCARD NO. 2

Group	Digit and decimal designation	Sample	Unit or Meaning
Battery No.	XX	05	Battery 5
Cycle No.	XXXX	2765	Cycle 2765
Year	X	6	1966
Date	XXX	365	Dec. 31
E/I ratio $= \frac{\text{V average}}{\text{I average}}$ coefficient*	X.XXX		Coef. Resistance
Cycle count cell #0*	XXXX.		
Cycle count cell #1*	XXXX.		
Cycle count cell #2*	XXXX.		
Cycle count cell #3*	XXXX.		
Cycle count cell #4*	XXXX.		
Cycle count cell #5*	XXXX.		
Cycle count cell #6*	XXXX.		
Cycle count cell #7*	XXXX.		
Cycle count cell #8*	XXXX.		
Cycle count cell #9*	XXXX.		
Maximum variance	XX.XXX		
Cell activity	XXXXXXXXXX		All cells
Real time phase completion*	XX.XXX	1322	Hours
Endcard No.	X	2	Endcard 2
Phase	X	0	Charge
Sequence No.	XXXX	8004	4th Card

Note 1: Each of the groups labeled with an asterisk can be replaced by another group if the new data has the same number of digits and the same decimal point location.

Note 2: The sequence numbers for endcards must start with a "6000" or "8000." The card to tape editing program checks these numbers to assure that the endcard data is put on an endcard data tape.

Figure 3—Format of endcard 2

CARD TO TAPE CONVERSION

In order to use the data on outcards and/or endcards in the SBAP, it must be stored on magnetic storage tape in a binary form. This is done by processing the outcards and endcards separately, using a set of card-to-tape editing routines. The routines are listed in Tables 1 and 2.

Table 1
Card to tape editing program for outcards.

A-1	Card to tape - rough routine
B-1	Sort routine
C-1	Alpha check routine
D-1	Preliminary analysis
E-1	Delete and duplicator elimination routine
F-1	BCD to binary routine
G-1	Binary merge routine

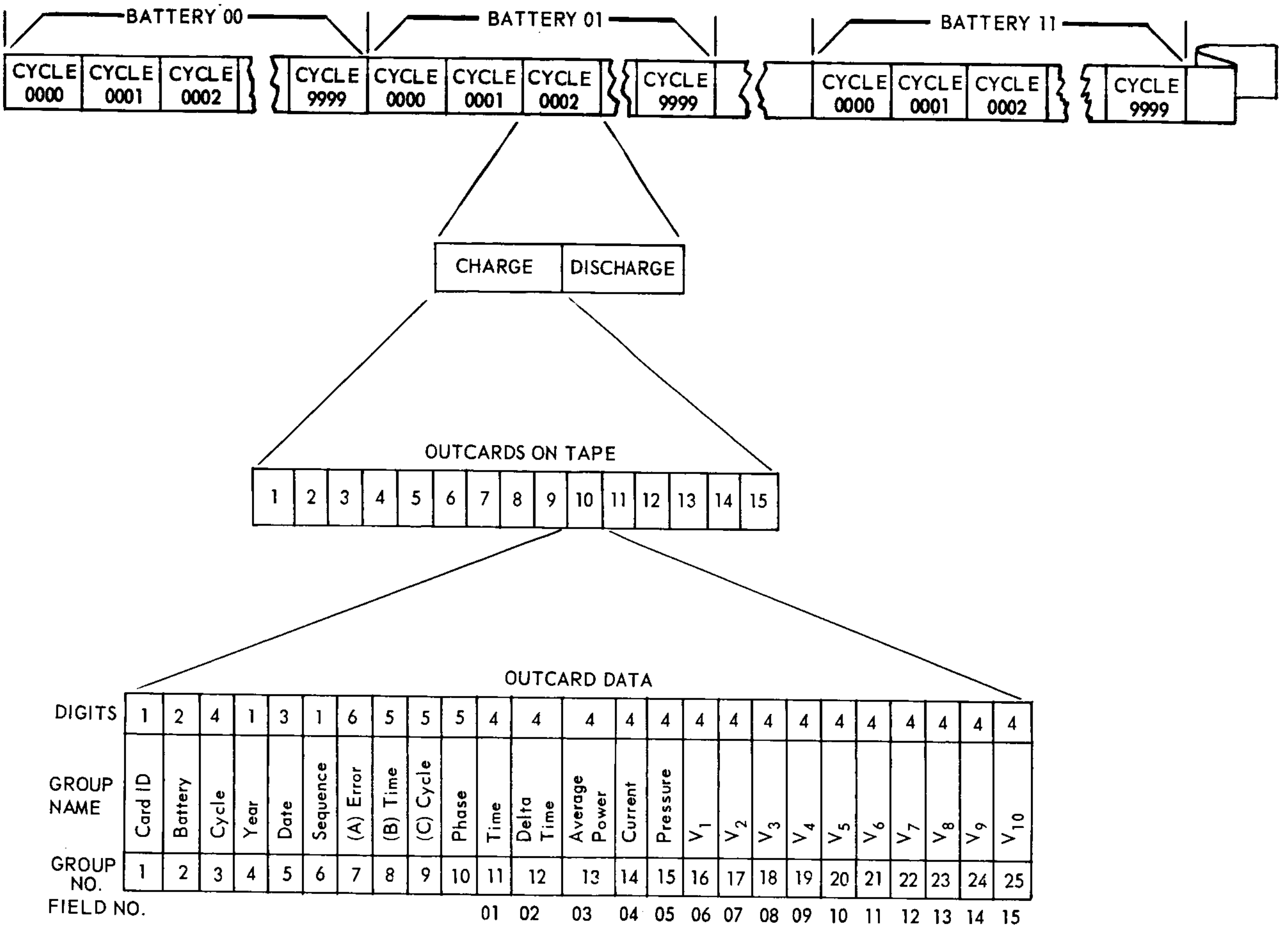
Table 2
Card to tape editing program for endcards.

A-2	Card to tape - rough routine (Endcard 1 and endcard 2 separately)
B-2	Sort routine (Endcard 1 and 2 separately)
C-2	Alpha check routine (Endcards 1 and 2 separately)
D-2	Preliminary analysis routine (Endcard 1 only)
E-2	Delete and duplication elimination routine (Endcard 1 only)
F-2	BCD to binary routine (Endcard 1 and 2 separately)
G-2	Binary merge routine (Endcard 1 and 2 together)

The objective of these routines is to transfer the data to magnetic tape, sort it, arrange it in the proper sequence, test it for validity, code the invalid or incomplete data, and merge the data in binary form with existing data. The routines, instructions, flow charts, and source deck statements appear in appendix A.

Outcard Raw Data Tape Format

When the conversion from outcards to an outcard raw-data tape is completed, the tape will contain the same information as the cards but in a slightly different arrangement of groups and in a particular sequence as illustrated in Figure 4.



DIGITS	1	2	4	1	3	1	6	5	5	5	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
GROUP NAME	Card ID	Battery	Cycle	Year	Date	Sequence	(A) Error	(B) Time	(C) Cycle	Phase	Time	Delta Time	Average Power	Current	Pressure	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9	V_{10}
GROUP NO.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
FIELD NO.											01	02	03	04	05	06	07	08	09	10	11	12	13	14	15

Figure 4—Outcard data tape format

The tape at the top of the page shows the data arranged in sequence by battery. Thus, all data pertinent to the first battery (battery 00) is recorded first on the tape, all data pertinent to the second battery (battery 01) is recorded next, and etc. Data recorded for each battery are divided into cycles; each cycle is subdivided into a charge or discharge phase. The data from each phase are subdivided into approximately 25 data groups. Groups 11 to 25 (designated fields 1 to 15) are abstracted from the outcard raw-data tape at the request of the analyst and used in the analysis.

Groups 1-10 are only checked as the computer processes the data and are never abstracted from the tape.

Endcard Raw Data Tape Format

The endcard data is arranged on the endcard raw-data tape by battery, by cycle, and by phase, respectively (Figure 5).

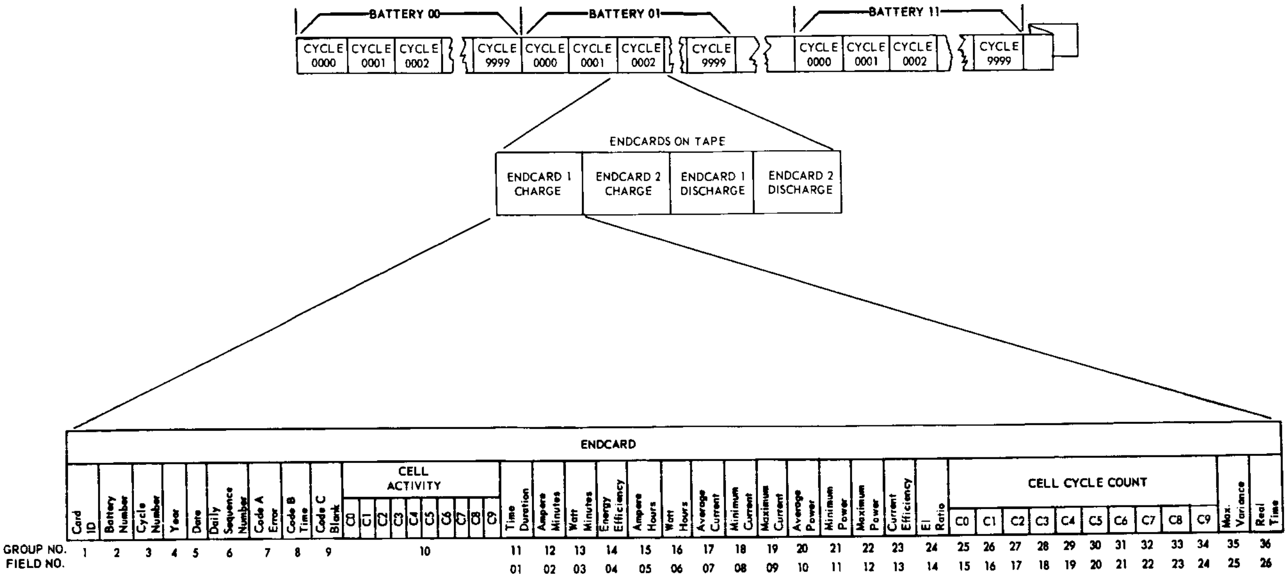


Figure 5—Endcard data tape format

The data from endcard 1 and endcard 2 were merged to form one large endcard on the tape. There are 36 groups on this endcard, and each group contains one item of information. As in the case of outcards, the first 10 groups are checked by the program while the computer is processing the data but never retrieved. The rest of the groups (11-36) are designated fields 1 to 26. These fields are specified by the analyst when requesting data to be abstracted from the raw-data tapes and are used in the analysis.

SYSTEM OF BATTERY ANALYSIS PROGRAMS (SBAP)

The SBAP consists of five programs which are used to process raw data input and to print-out desired information in a suitable form. The overall program sequence is illustrated in Figure 6.

As pointed out in the first part of this volume, raw data input consists of outcard and endcard data tapes which contain encoded measurements made during battery charge and discharge cycles. Outcard data are obtained during a charge or discharge phase, endcard data are taken at the end of each charge and discharge phase.

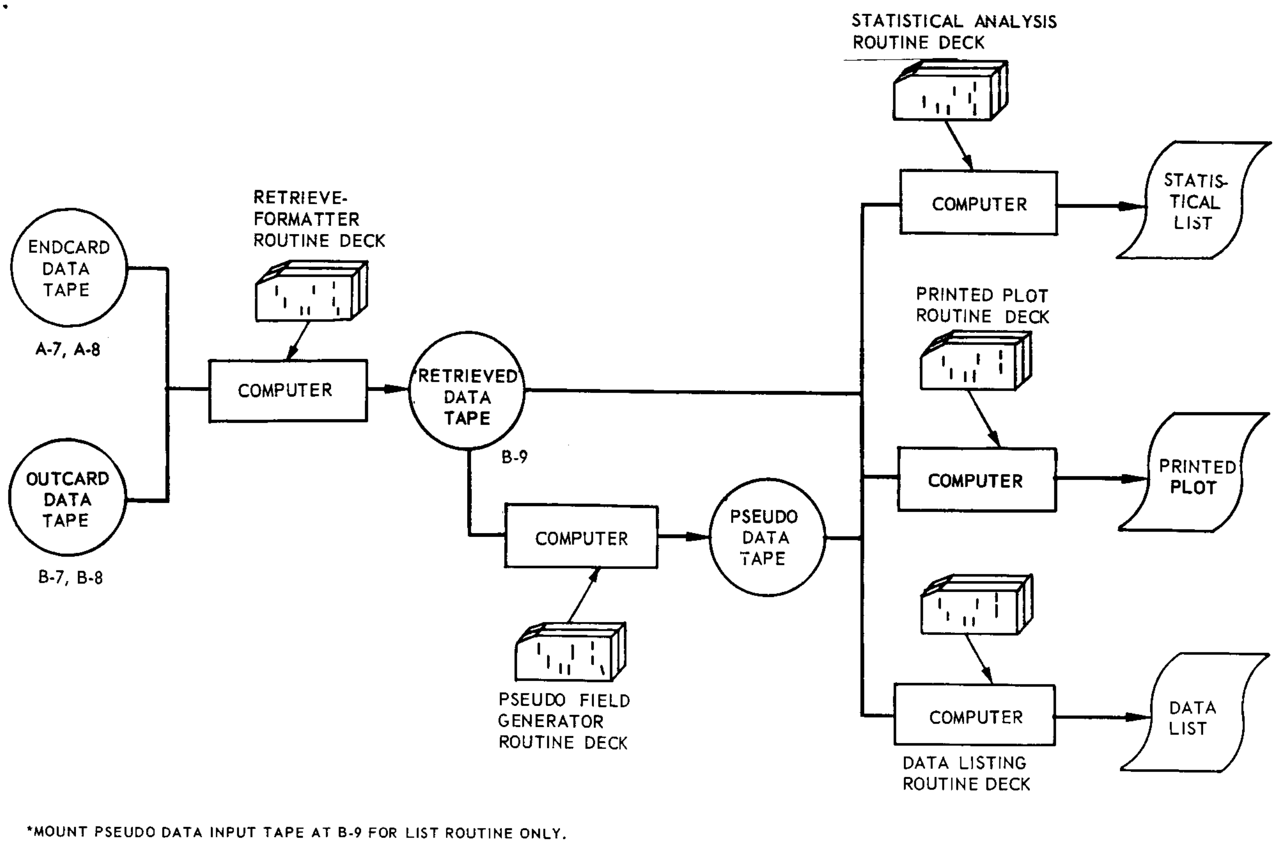

STATISTICAL ANALYSIS ROUTINE DECK
COMPUTER
STATIS-TICAL LIST
RETRIEVE-FORMATTER ROUTINE DECK
ENDCARD DATA TAPE
A-7, A-8
COMPUTER
RETRIEVED DATA TAPE
B-9
OUTCARD DATA TAPE
B-7, B-8
PRINTED PLOT ROUTINE DECK
COMPUTER
PRINTED PLOT
COMPUTER
PSEUDO DATA TAPE
PSEUDO FIELD GENERATOR ROUTINE DECK
COMPUTER
DATA LIST
DATA LISTING ROUTINE DECK
*MOUNT PSEUDO DATA INPUT TAPE AT B-9 FOR LIST ROUTINE ONLY.

Figure 6—Battery analysis program functional flow diagram

The five routines are:

1. A Retrieve-Formatter Routine to extract specified data from the endcard and outcard data tapes, and to record it in a more suitable form on a retrieved-data tape.

2. A Pseudo Generator Routine for mathematical operations on the retrieved-formatted data which records the results on a pseudo-data tape.

3. A Data List Routine to obtain a listing (printout) of the data as it appears on the retrieve or pseudo tape.

4. A Statistical Analysis Routine for the data using such results as mean, variance, standard deviation, linear correlation, and multiple stepwise linear regression.

5. A Printed Plot Routine which presents data in graphical form.

In order to process a routine, a tape containing the data and the routine card deck must be supplied. The decks are complete except for control cards which must be added by the program user. The control cards determine the results obtained by specifying the type of data to be abstracted, and the statistical and mathematical operations to be performed.

Recorded data which might be incomplete because the cycle was cut short or might contain errors but are still usable can be requested and will be coded with a flag word describing what the problem is. However, it will not be supplied unless specifically requested.

The time required for a run of one routine depends on the amount of data requested and the operations to be performed. The average request for one routine is 2 minutes.

Data Retrieve-Formatter Routine

The retrieve-formatter routine whose functional diagram appears in Figure 7 is the first step in the analysis. The purpose is to select specific data fields from endcard or outcard raw data tapes and record this data on a retrieved data tape. The format of the retrieved data will be suitable for processing in the other routines of the program. An off-line history tape is simultaneously prepared which can be listed — to ascertain error or data loss.

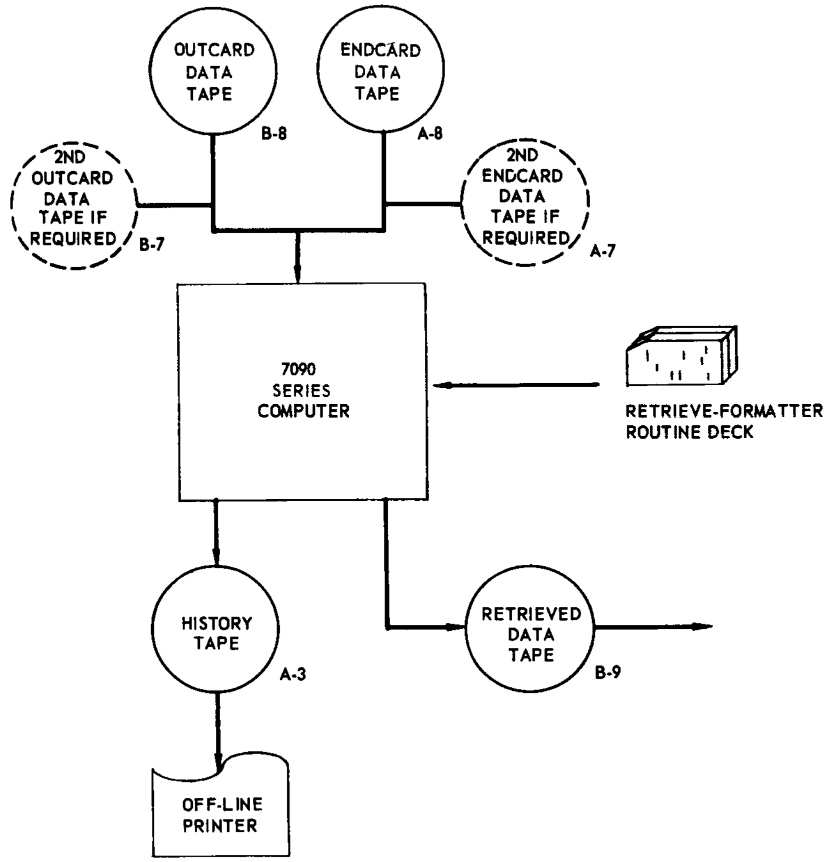


Figure 7—Retrieve formatter routine functional flow diagram

An endcard raw data tape is the only input tape required as an input when the endcard data is desired in the analysis. If outcard data is desired, both tapes are required because the outcard data on a particular phase (charge or discharge) is referenced to the total number of ampere hours (field 9) and the total time duration (field 5) of that phase appearing on the endcard data tape. If total ampere hours and total time duration are identical for the same phase on every cycle, an additional control card with the proper information can be added to the routine deck and only the outcard raw data tape is needed.

The retrieve-formatter routine deck, with one or more control cards added, is used to control the computer program so that only the fields requested are retrieved from the input data (raw data) tape. If one raw data endcard or outcard tape does not contain all the data available, a second tape of each can be used provided the cycle numbers on the outcard tapes and endcard tapes correspond.

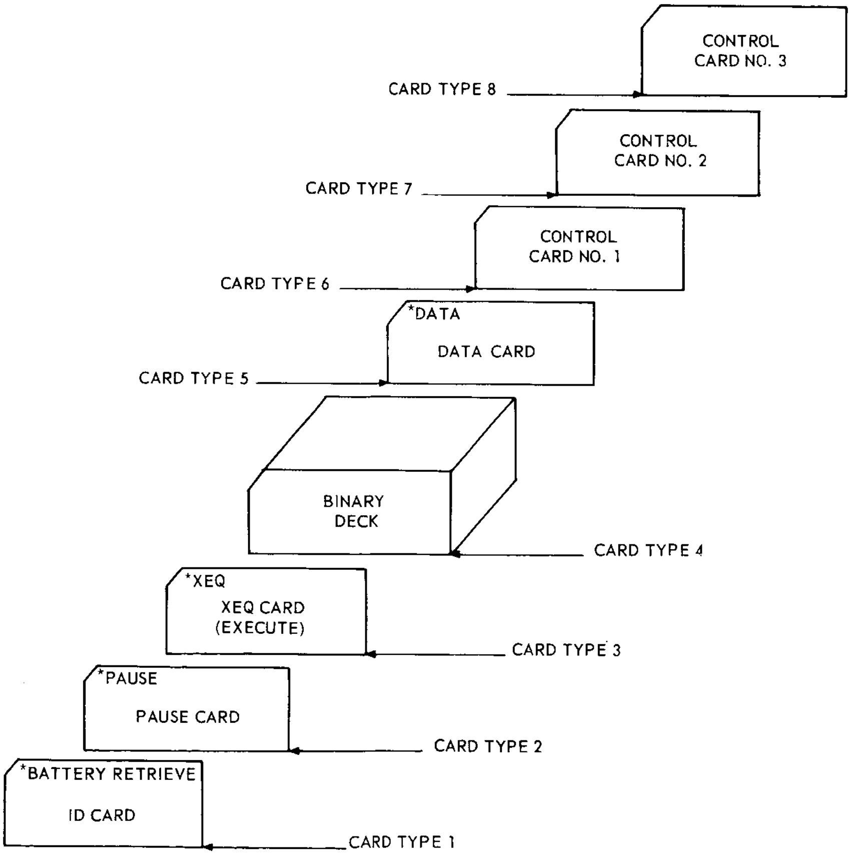


Figure 8—Retrieve formatter routine card deck arrangement

Card deck arrangement. The cards are arranged in the sequence shown in Figure 8. The cards types are:

Card Type	Description
1	Analyst identification card – asterisk in column 1 and any appropriate statement in columns 7-72
2	Pause card – asterisk in column 1 and the letters PAUSE in columns 7-11
3	Execute card – asterisk in column 1 and the letters XEQ in columns 7-9
4	Binary routine deck
5	Data card – asterisk in column 1 and the letters DATA in columns 7-10
6	Control card 1
7	Control card 2
8	Control card 3

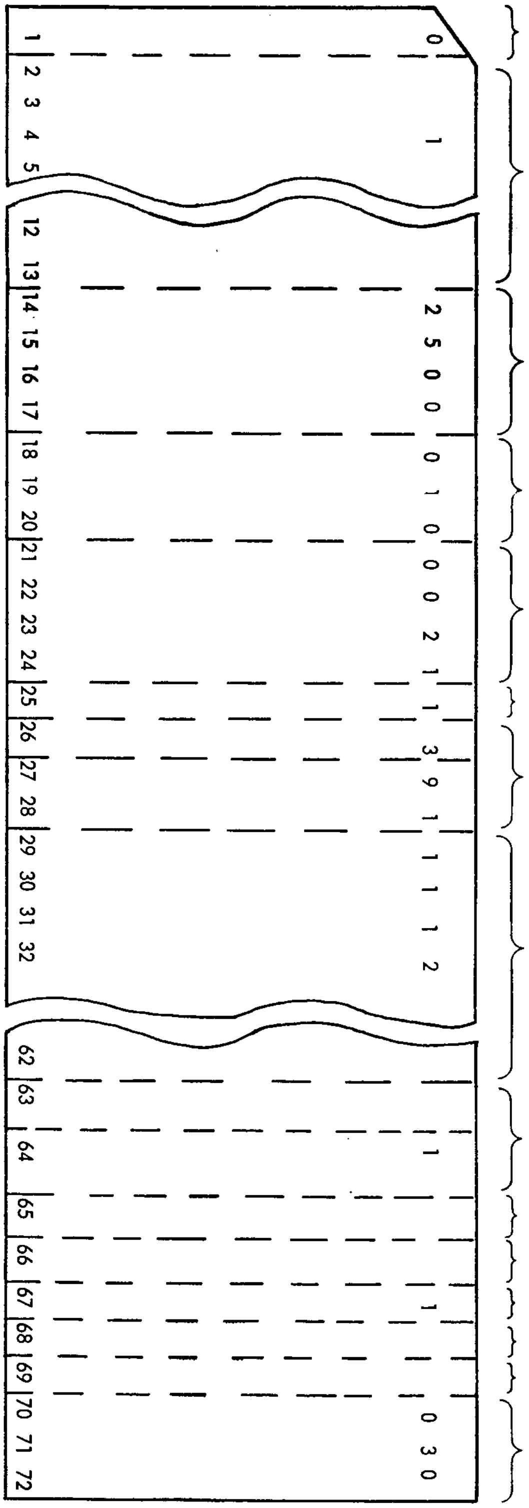
Types 2-5 are the same for every retrieve-formatter request. Type 1 can be changed at the descretion of the analyst. Types 6-8 contain the request, and will control the routine.

Control cards. Each request requires from one to three control cards. The first card selects the batteries and fields to be retrieved, the cycle range and increments, and raw data input tape. The second card provides six additional cycle ranges, and the third is used when the outcard raw data tape is processed by itself. In order to process more than one request on a single computer run, additional sets of control cards can be added to the end of the deck. A description of each control card is given below.

Control card 1. The information punched into this card contains the request of the analyst for abstracting data from the endcard data and/or outcard data tape. It can be used without control card 2 if no additional cycle ranges are required, and without control card 3 if the outcard data tape only is not required.

Control Card No. 1 (Figure 9)

Item	Control card columns	Description
1	1	A "1" punch in column 1 of the control card will select endcard data. A "0" punch in this column will select outcard data. Figure 9 shows the selection of outcard data.
2	2-13	A selection of data on one or more batteries is obtained by entering a "1" punch in the appropriate battery identification columns 2-13. The illustration shows the selection of batteries 4, 8 and 9.
3	14-17	Items 3, 4, and 5 are listed separately but are used in conjunction with one another. Item 3 specifies the first cycle number requested. Cycle 2500 is selected in the illustration.



Card Type	Description
1	Outcard or endcard data
2	Battery identification
3	Starting cycle
4	Interval between cycles
5	Number of increments
6	Phase
7	Time code
8	Field definition
9	Outcard data tape only
10	Least square fit-% of cycle
11	Statistical or non-statistical
12	Additional cycle range
13	Invalid data request
14	Cell cycle (outcard only)
15	Expired day

Figure 9—Retrieve-formatter routine control card 1

Item	Control card columns	Description
4	18-20	The minimum distance between cycles specifies the minimum number of cycles between the actual desired cycles. Starting with cycle 2500 and the minimum distance between cycles of 10, the following cycles if found would appear on the retrieved data tape: 2500, 2510, 2520 . . . when a cycle cannot be found the closest higher cycle is used.
5	21-24	Item 5 specifies the maximum number of cycles desired. As illustrated, 21 increments of 10 cycles each would appear on the retrieved data tape: 2500, 2510, 2530 ... 2690, 2700. The cycle range is between 2500 and 2700. The program will extract data only within this range, even though the data may be sparse and cycle 2700 is reached before 21 cycles have been retrieved.
6	25	The half cycle specification is actually a specification of charge or discharge phase. A "0" punch in this column denotes the charge phase of the cycle. A "1" punch denotes the discharge phase (Figure 9), and a "2" punch denotes that the requested fields will be extracted for both phases. A negative cycle appearing on the retrieval tape indicates data is extracted from the discharge phase.
7	26-28	The time code columns of the card, columns 26 through 28, apply only to the outcards. The digit in column 26 may be "0", "1", "2" or "3". A "0" specifies all data is requested for those cycles. A "1" specifies a request for a third order least squares polynomial fit of the selected fields corresponding to $Y = Ax^3 + Bx^2 + Cx + D$. Delta time will always be the x-coordinate and must never be one of the requested fields. The Y-coordinate may be any of the fields (maximum of 5), except delta time. The output from a least squares fit request will contain standard deviation and four coefficients (A, B, C and D) for each field requested. In order for this to occur a "1" must also be punched in column 65 (item 10). A "2" or "3" in column 26 is used in conjunction with columns 27 and 28. Columns 27 and 28 specify the actual percentage of the cycle (time duration) that data are desired. The illustration shows 91 percent of the cycle. A "2" in column 26 would specify the first 91 percent of the cycle and a "3" in this column would specify the last 91 percent of the cycle. A "0" in column 26 would specify 100 percent of data in the cycle. The last 91 percent of the cycle is illustrated. When there is a "1" in column 65 there will be a least squares polynomial fit of the specified percentage of the cycle requested.
8	29-62	The various fields of the endcard and outcard data tapes are called out by field number (Table 3). There are 15 separate outcard fields and

Table 3

Definition of field numbers for retrieve formatter routine

Field numbers	Actual outcard field	Actual endcard field
F01	Time	Time duration
F02	Elapsed Time	Amp minutes
F03	Average Instant. Power	Watt minutes
F04	Current	Energy efficiency
F05	Pressure	Amp hours
F06	Voltage of cell 0	Watt hours
F07	Voltage of cell 1	Avg. current
F08	Voltage of cell 2	Min. current
F09	Voltage of cell 3	Max. current
F10	Voltage of cell 4	Avg. power
F11	Voltage of cell 5	Min. power
F12	Voltage of cell 6	Max. power
F13	Voltage of cell 7	Current efficiency
F14	Voltage of cell 8	$\frac{\text{Avg. voltage}}{\text{Avg. current}}$
F15	Voltage of cell 9	Cycle of cell 0
F16		Cycle of cell 1
F17		Cycle of cell 2
F18		Cycle of cell 3
F19		Cycle of cell 4
F20		Cycle of cell 5
F21		Cycle of cell 6
F22		Cycle of cell 7
F23		Cycle of cell 8
F24		Cycle of cell 9
F25		Maximum variation
F26		R. time

Item	Control card controls	Description
8 (continued)		26 endcard fields noted in the table. To select any parameter such as voltage, pressure or current it will be necessary to specify its two-digit numbered field. The first field illustrated on the sample control card is field 11, "voltage" for cell 5. All remaining fields desired, up to 16, are likewise selected by 2 digit numbers. The fields desired need not be sequentially numbered. Endcard fields differ from outcard fields and can only be distinguished in the retrieve-formatter routine by requesting the outcard raw data tape (0 in column 1 of control card 1) or endcard raw data tape (1 in column 1 of control card 1).

Item	Control card columns	Description
9	64	A "0" in column 64 is required when only an outcard raw data tape is supplied as input. It requires that control card 3, which contains total ampere hours and total time duration for charge and discharge, be included in the deck. A "1" or blank in this column will require both endcard and outcard raw data tape input for an outcard data request, and an endcard raw data tape only as input for an endcard data request.
10	65	A "1" in column 65 is used in conjunction with item 7 (column 26) to specify a least squares polynomial fit. A "0" or blank in this column is a request for the actual raw data.
11	66	Column 66 can be used to describe a statistical ("1") or nonstatistical ("0") request. A "1" in this column will cause the retrieve routine to stop abstracting data when a blank or -0 value is found on the raw data tape. A "0" in this column will allow the routine to continue to abstract until the cycle range request is satisfied.
12	67	A digit "1" in column 67 indicates that additional "cycle ranges" are necessary. Control card 2 must be included with the routine deck. The illustration shows additional cycle ranges are necessary.
13	68	A digit "1" in column 68 of the control card will cause all invalid data within the request to be written on the retrieved data tape, otherwise, all invalid data will be bypassed.
14	69	In certain cases it may be desirable to specify a particular cell cycle instead of a battery cycle. A "1" punch in column 69 will give the analyst this option. The actual cell desired (01 - 09) will be punched in the first field definition (Columns 29 and 30).
15	70-72	To provide for later checking or sorting, the analyst will enter the expired day of the year in columns 70, 71 and 72.

Control card 2 (Figure 10)

The first cycle range is requested on control card 1 from columns 14-24 (items 3-5). Up to six additional ranges can be specified with control card 2 illustrated in Figure 10.

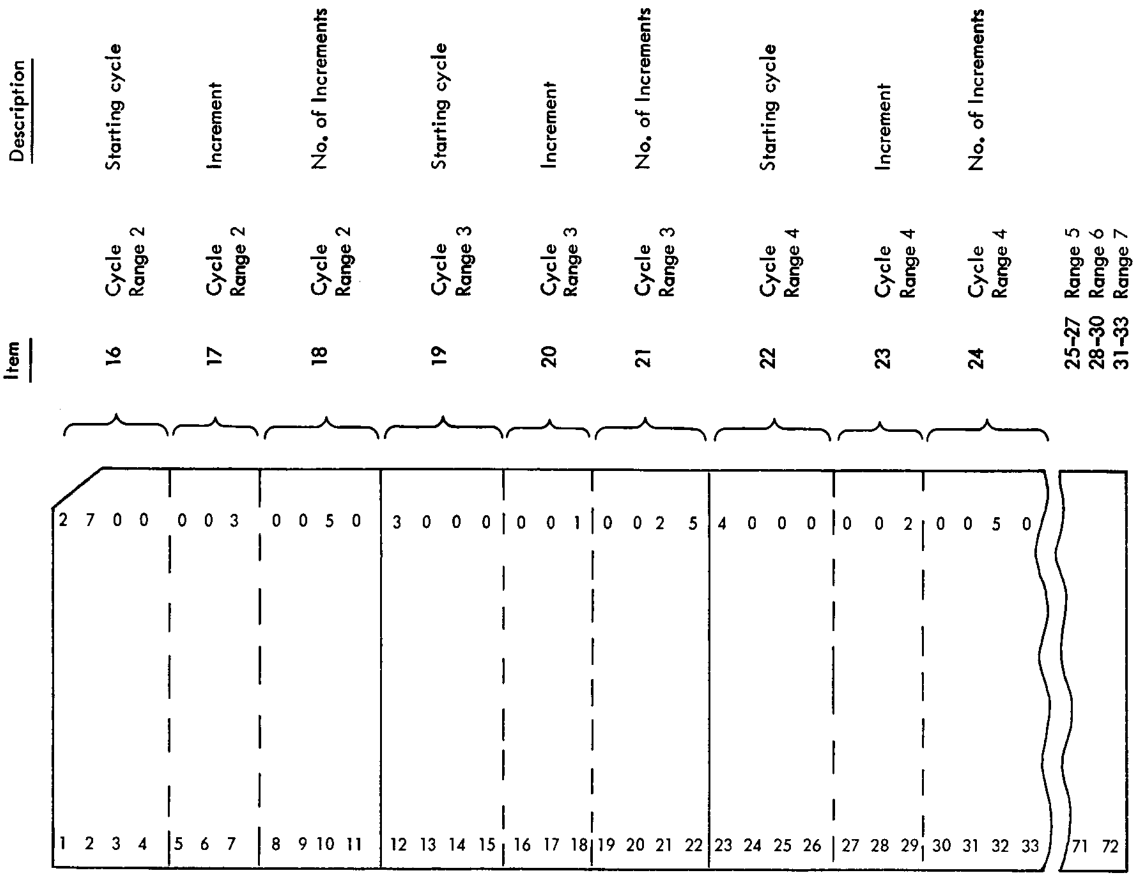

Item
Description
16 Cycle Range 2 Starting cycle
17 Cycle Range 2 Increment
18 Cycle Range 2 No. of Increments
19 Cycle Range 3 Starting cycle
20 Cycle Range 3 Increment
21 Cycle Range 3 No. of Increments
22 Cycle Range 4 Starting cycle
23 Cycle Range 4 Increment
24 Cycle Range 4 No. of Increments
25-27 Range 5
28-30 Range 6
31-33 Range 7
2 7 0 0
0 0 3
0 0 5 0
3 0 0 0
0 0 1
0 0 2 5
4 0 0 0
0 0 2
0 0 5 0
1 2 3 4
5 6 7
8 9 10 11
12 13 14 15
16 17 18
19 20 21 22
23 24 25 26
27 28 29
30 31 32 33
71 72

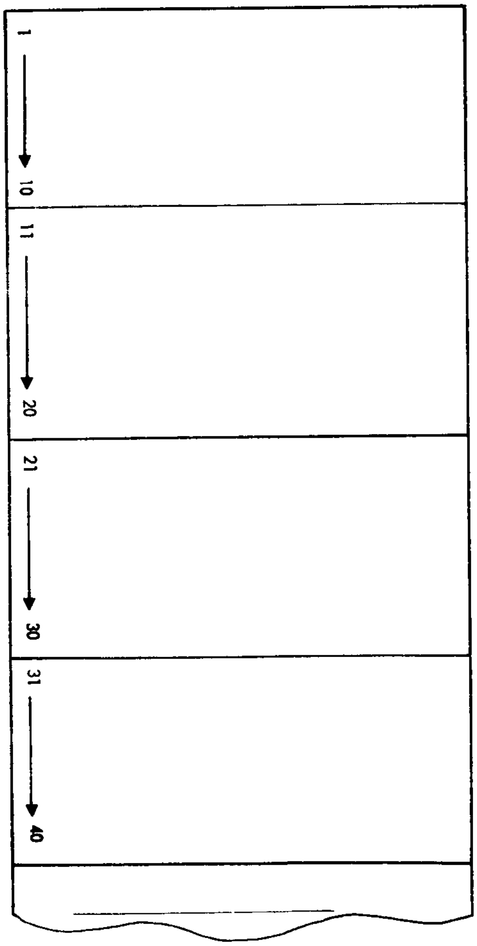
Figure 10—Retrieve formatter routine control card 2

Item	Control card column	Description
16	1-4	This is the starting cycle number. It has the same description as item 3 on control card 1, except that it is the second range.
17	5-7	The increment desired is punched into columns 5-7. Item 17 has the same description as item 4 in control card 1, except that it refers to the second cycle range.
18	8-11	The number of increments is punched into columns 8-11. It has the same description as item 5 in control card 1 except that it is the second cycle range.

Item	Control card columns	Description
19-21	12-22	Same as items 16-18 except cycle range 3
22-24	23-33	Same as items 16-18 except cycle range 4
25-27	34-44	Same as items 16-18 except cycle range 5
28-30	45-55	Same as items 16-18 except cycle range 6
31-33	56-66	Same as Items 16-18 except cycle range 7

Control card 3 (Figure 11)

Item	Description
34	Columns 1-10 contain the total ampere hours of charge. The value can appear anywhere in these columns but a decimal point must be included.



Item	Description
34	Amp hrs charge
35	Amp hrs discharge
36	Time duration charge
37	Time duration discharge

Figure 11—Retrieve formatter routine control card 3

Item	Description
35	Columns 11-20. Same as Item 22 except for discharge.
36	Columns 21-30 contain total delta (elapsed) time — in hours of charge. The restriction is the same as item 23.
37	Columns 31-40. Same as item 24 except for discharge.

Instructions to computer operator. The following instructions are given to the operator when submitting the data tapes and routine card deck for processing.

1. Mount endcard binary data tape on logical tape setting of A-8. If an additional endcard tape is necessary to accommodate the raw data, the second tape is mounted on logical tape setting A-7.

2. Mount outcard binary data tape on logical tape setting B-8. If an additional outcard tape is necessary to accommodate the raw data, the second tape is mounted on logical tape setting B-7. Caution — The second endcard and outcard tape must begin with the same cycle of the same battery.

3. The retrieved binary data tape has a logical tape setting of B-9 (Output).

4. The off-line (history tape) will have a logical tape setting of A-3. A printout of this tape is requested (the list routine is not required).

5. For additional requests the operator is instructed to press START upon completion of run.

6. Fortran II — EXECUTE

7. Average machine time for a run is 2 minutes.

Pseudo Generator Routine

The purpose of the pseudo generator routine is to enable the analyst to obtain cell, battery, and other related parameters not present on the retrieved or raw data tapes. The method of generating pseudo data tapes is illustrated in Figure 12. The input required is a retrieved data tape or another pseudo data tape. The pseudo generator routine deck including two sets of one to sixteen pseudo equation cards supplied by the analyst are used to control the routine. The resulting output is a pseudo data tape containing the defined pseudo fields appended to the original retrieved data fields. The original request of the retrieve-formatter routine and the defined pseudo equations are listed.

Card deck arrangement. The pseudo field generator deck arrangement appears in Figure 13. The card types appearing in the figure are described below.

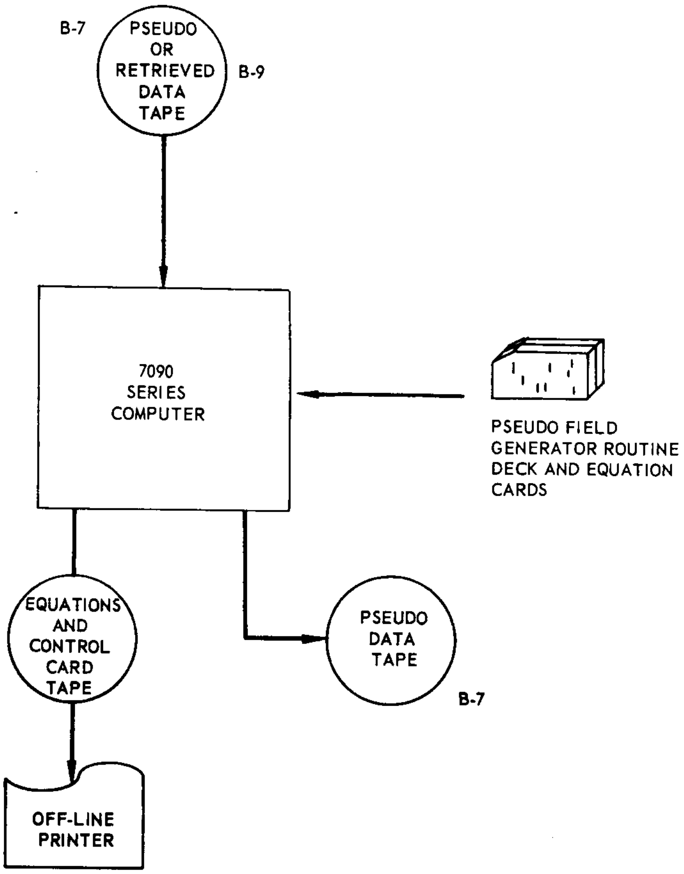

B-7
PSEUDO OR RETRIEVED DATA TAPE
B-9
7090 SERIES COMPUTER
PSEUDO FIELD GENERATOR ROUTINE DECK AND EQUATION CARDS
EQUATIONS AND CONTROL CARD TAPE
PSEUDO DATA TAPE
B-7
OFF-LINE PRINTER

Figure 12—Pseudo field generator routine functional flow diagram

Card type Description

1 Run identification card — must have an asterisk punched in column 1 — any other information may be punched in columns 2-72.

2 Pause card — has an asterisk punched in column 1 and the word PAUSE punched in columns 7-11.

3 Execute card — has an asterisk punched in column 1 and XEQ punches in columns 7-9.

4 This is a set of 7 cards which constitute the control information of the FORTRAN source language subroutine mentioned in the input section.

5 Pseudo equations start card — has a C punched in column 1 and the words PSEUDO EQUATIONS START HERE punched in columns 7-34. This card is used to indicate to the analyst where he must insert the pseudo field equation cards.

Card type	Description
6	The set of pseudo field equation cards is to be inserted by the analyst.
7	Pseudo equations endcard – has a C punched in column 1 and the words PSEUDO EQUATIONS STOP HERE punched in columns 7-33. The card is used to indicate to the analyst where he must insert the pseudo field defining cards.
8	Return card – has the word RETURN punched in columns 7-12. This is another control card of the FORTRAN source language subroutine.
9	Endcard – has the word END punched in columns 7-9. This card is the terminal card of the subroutine.
10	A set of binary punched cards which constitute the main body of the pseudo field generator program.
11	Data card -- has an asterisk punched in column 1 and the word DATA punched in columns 7-10. This card is used to indicate where the analyst is to insert the second set of the pseudo field defining cards.

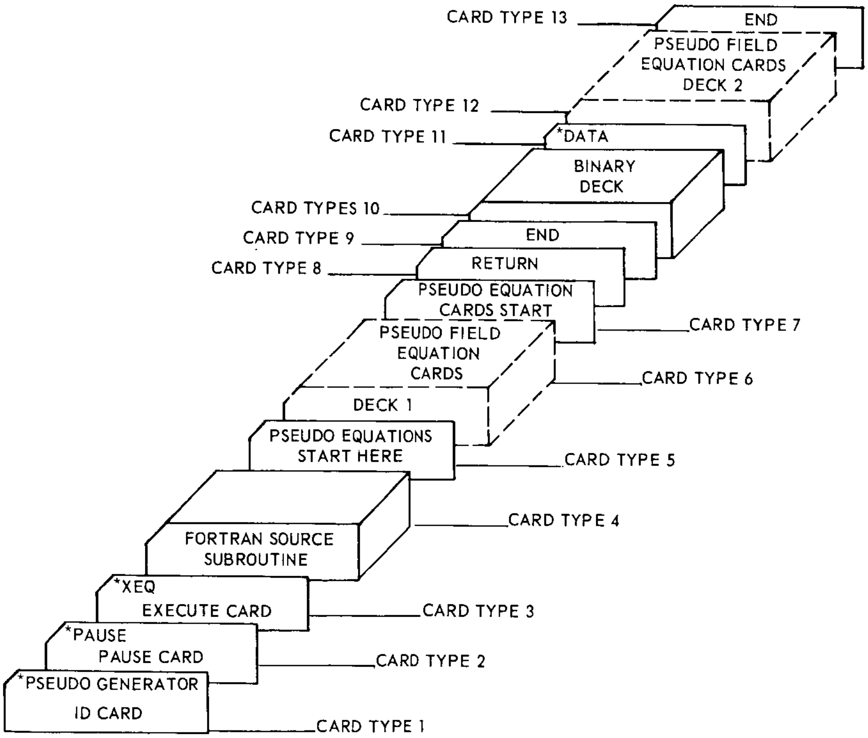


Figure 13—Pseudo field generator routine card deck arrangement

Card type	Description
12	The second set of the pseudo field defining equations to be inserted by the analyst.
13	Endcard — has the word END punched in columns 7-9. This card is the terminal card of the entire program.

Card types 1, 6 and 12 are prepared by the analyst and put into the proper position in the card deck prior to submitting the routine deck to the computer operator. All other types remain unchanged for every run.

Card type 1 contains a statement which identifies the specific request for the analyst. It does not take part in the routine and is later printed out as punched.

Types 6 and 12 are identical sets of pseudo equation cards which must be defined by the analyst. A comment (identification) card may be inserted among the pseudo field equation cards.

Pseudo field equation cards. The analyst can request from one to sixteen pseudo fields using this routine. Each pseudo field equation must be punched on a separate card. Three examples of pseudo field equations are given in Figure 14. Note that columns 1 to 6 are blank. The letter P must appear in column 7 followed by the pseudo field number (1 in Example 1), an equal sign (=) and then the equation. Columns 73 to 80 may contain any identifying symbols or code desired by the analyst.

The equations are FORTRAN II statements which means that all operations valid for FORTRAN II can be used. Some of these are listed in Table 4 along with the symbolic language.

The fields available from the raw data tape and from the retrieve-formatter tape, together with their symbolic field names, appear in Table 3. The method used for referencing the desired retrieved data fields and pseudo fields in the pseudo field equations is described using the examples in Figure 14.

Example 1

After using an endcard raw data tape as input to the retrieve-formatter routine, a retrieved data tape was generated. This tape serves as input to the pseudo field generator routine. In the example, pseudo field 1 (PI) is set equal to the result of data field 2 (amp minutes) divided by the sum of data field 6 (watt hours) and data field 4 (energy efficiency). Pseudo field 1 (PI) will be calculated using these specific fields for every cycle on the retrieved data tape.

Example 2

This example is similar to example 1 except that in addition to the inclusion of a data field (F01) in the equation, a pseudo field, the value for cycle (cyc), and a constant (7.24) were included. In the equation pseudo field 2 (P2) is set equal to 7.24 plus the quotient of pseudo field 1 (P1) times

Table 4
Symbols Used in FORTRAN II Language

Operation	Symbol
1. Exponentation	**
2. Division	/
3. Multiplication	*
4. Addition	+
5. Subtraction	-
6. Sine (ARG)	SINF (ARG)
7. Cosine (ARG)	COSF (ARG)
8. Arctangent (ARG)	ATANG (ARG)
9. Natural Log (ARG)	LOGF (ARG)
10. Log_{10} (ARG)	LOG10F (ARG)
11. $e^{(ARG)}$	EXPF (ARG)
12. $\sqrt{ARG}$	SQRTF (ARG)
13. |ARG|	ABSF (ARG)

NOTES: a. If the operations used in the expression are not explicitly ordered by parentheses, the order of the first five operations will be as shown.

b. The (ARG) above must be any legitimate arithmetic expression.

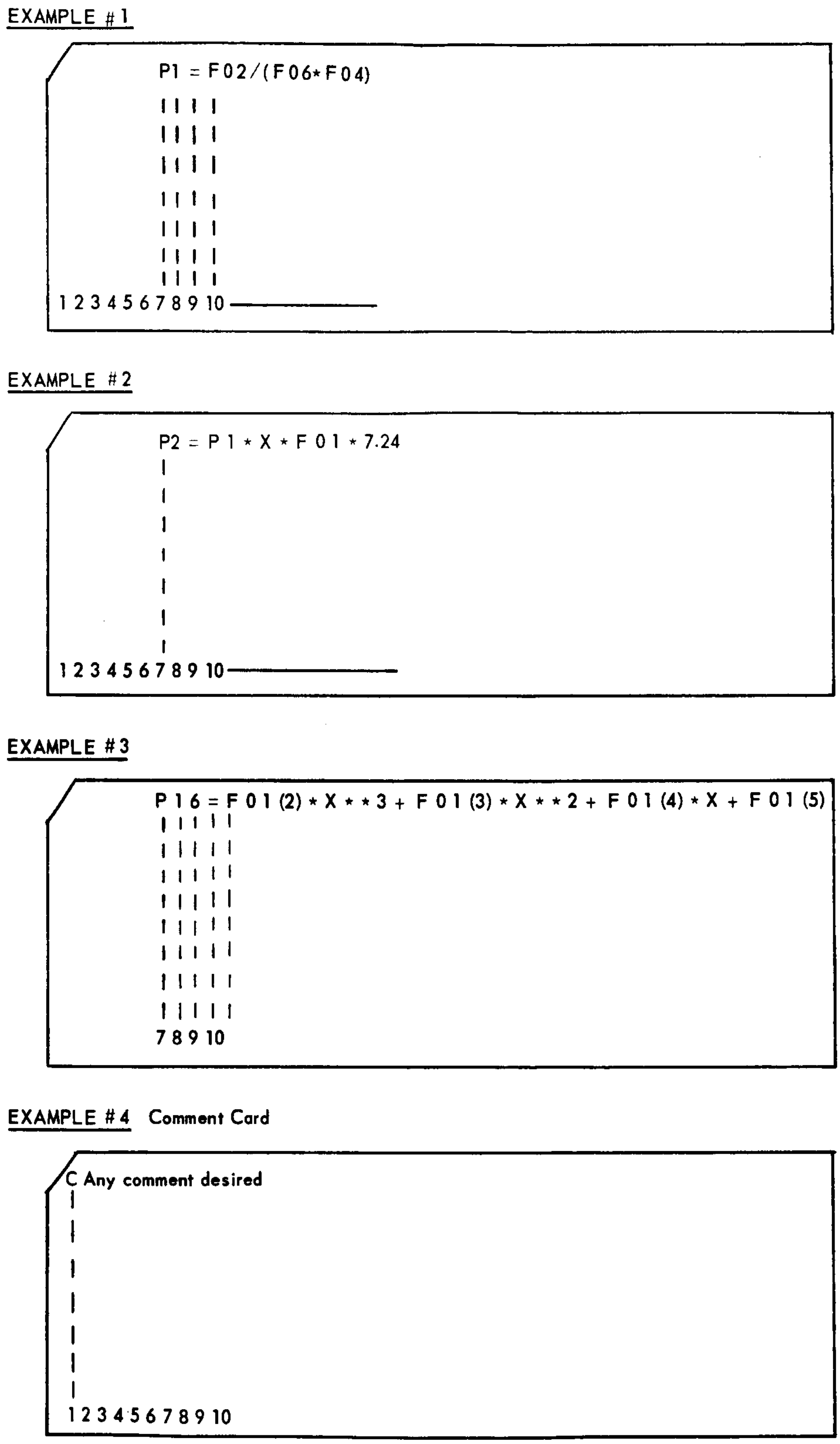
c. For the availability and symbols of other operations see any FORTRAN II instruction manual.

cycle number times data field 1 (time duration). Any constant may be used provided it contains a decimal point and is in the range of -1×10^{-36} to $+ 1 \times 10^{36}$.

Example 3 - least squares fit of the data

An option of the retrieve-formatter routine allows the analyst to request a least squares polynomial fit of the outcard data. In this case, as noted earlier, the retrieved tape contains the standard deviation, the coefficients of the polynomial and the time duration for each cycle in place of the actual single-valved fields. The polynomial which uses the coefficients is

$$P = Ax^3 + Bx^2 + Cx + D,$$



EXAMPLE #1

```
P1 = F02/(F06*F04)
```

EXAMPLE #2

```
P2 = P 1 * X * F 0 1 * 7.24
```

EXAMPLE #3

```
P 1 6 = F 0 1 (2) * X * * 3 + F 0 1 (3) * X * * 2 + F 0 1 (4) * X + F 0 1 (5)
```

EXAMPLE #4 Comment Card

```
C Any comment desired
```

Figure 14—Examples of pseudo field equation cards

where A, B, C, D are the coefficients and X is the time described in Table 5. The field designations for these coefficients are included in the table.

In order to calculate the value for this polynomial at a given time, the pseudo equation card in example 3 must be supplied. In example 3 pseudo 16 (P16) is set equal to the result of the A coefficient of data field 1. (F01 (2))times the time cubed (X ** 3), plus the B coefficient, (F01 (2)) times the time squared (X ** 2) and etc., X represents the total time. If the value of the field at a particular time were required, the specific time would be inserted in the place of X. If the value of the field were required at the midpoint of the cycle, the symbol (X/2) would be inserted in place of X.

Table 5
Symbolic names, least square fit data

Symbolic name	Actual outcard data
X	Time duration for all fields in a particular cycle
F01 (1)	The standard deviation of the polynomial fitted to the outcard data of field number 01
F01 (2) = A	The coefficient of the X^3 term of the polynomial
F01 (3) = B	The coefficient of the X^2 term of the polynomial
F01 (4) = C	The coefficient of the X term of the polynomial
F01 (5) = D	The constant term of the polynomial

F01 is the number of the data field requested for least square fit

Example 4 — comment Cards

Example 4 in Figure 14 illustrates the format of the comment card. The letter C must be inserted in column 1 and the actual comment begins in column 7. The comment card is similar to the run identification card because it has no functional use except to interject appropriate comments among the actual field-defining equations. It was noted that the use of this card was optional.

Because of the nature of the pseudo generator routine, the analyst is not bound to insert only pseudo field defining cards into the FORTRAN source language subroutine. Any legal FORTRAN statement with the exception of a COMMON statement could be used, which means that a subroutine could be written in FORTRAN II language, and submitted in place of or included among the pseudo field equations if desired.

The details of other type statements are not mentioned because they are not to be used by the analyst. However, they can be found in any FORTRAN II instruction manual.

Instructions to the operator. The following instructions are given to the computer operator when submitting the retrieve or pseudo data tape and the pseudo field generator card deck.

1. If a retrieved data tape is used as input, mount on a logical tape setting of B-9. If another pseudo field tape is used as input, mount on a logical tape setting of B-7.

2. The output tape of the pseudo field generator routine has a logical setting of B-7.

3. FORTRAN II — compile and execute.

4. The time required for computer usage depends on the number and type of pseudo field equations. The average request would probably be 2 minutes.

Data List Routine

The data list routine is used to print out the data in tabular form of either a retrieved data tape or pseudo data tape as in Figure 15. An example of a retrieved data tape with least squares fit option as shown in Figure 16. The listing of a pseudo data tape (Figure 17) is also shown. A non-least squares retrieved data list would be identical to the pseudo data list without the pseudo fields.

Card deck arrangement. The arrangement of the data list deck appears in Figure 18. The cards required are described. It will be noted that no control cards are required. The deck is the same for every request.

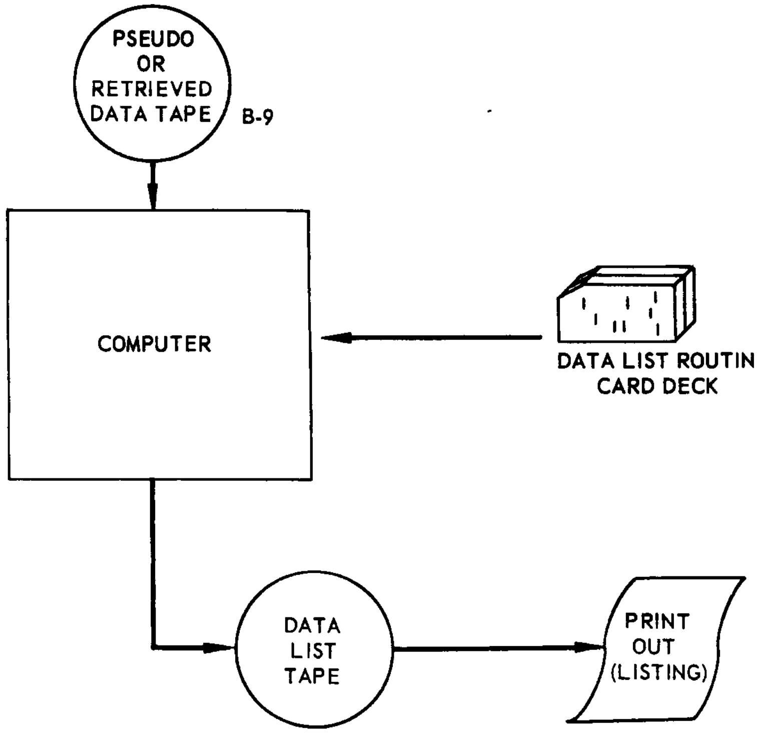
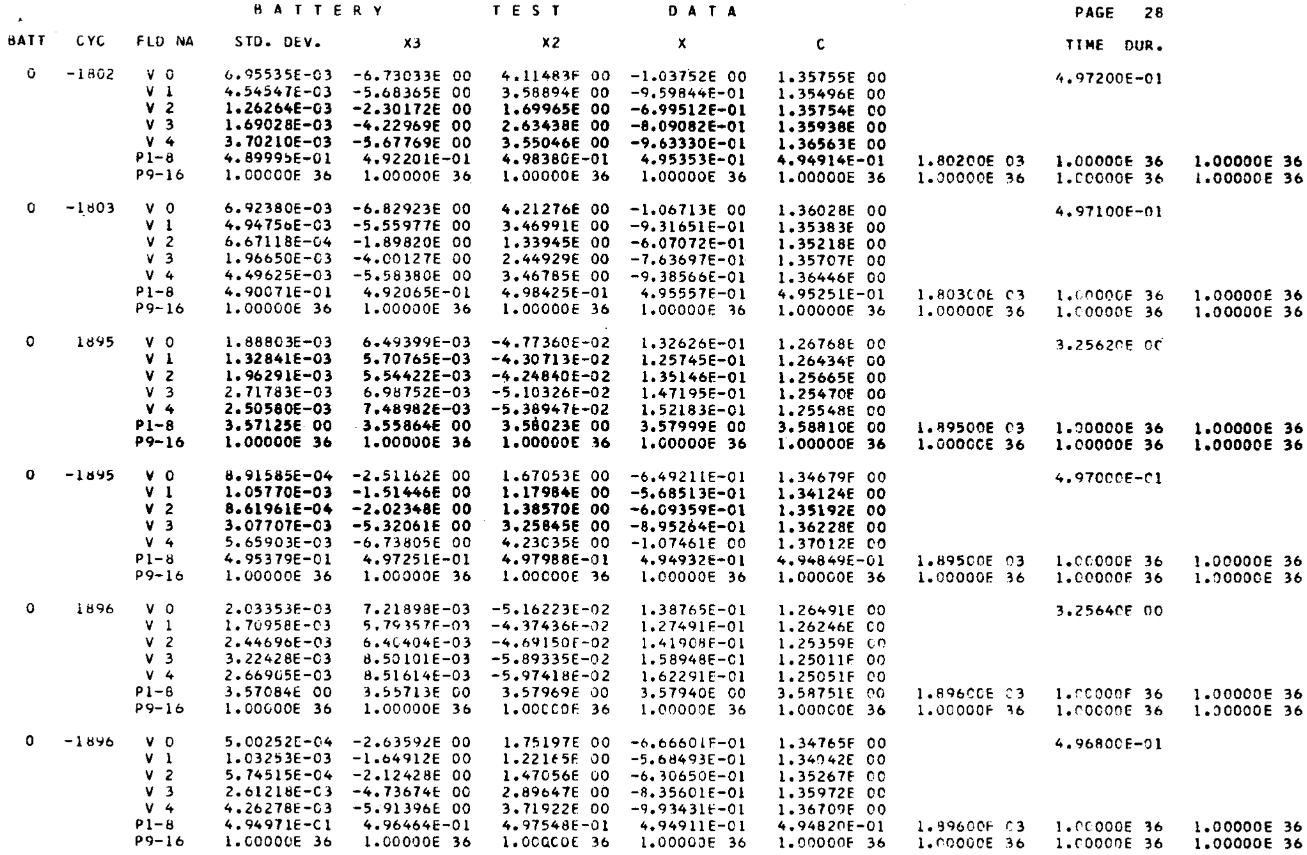
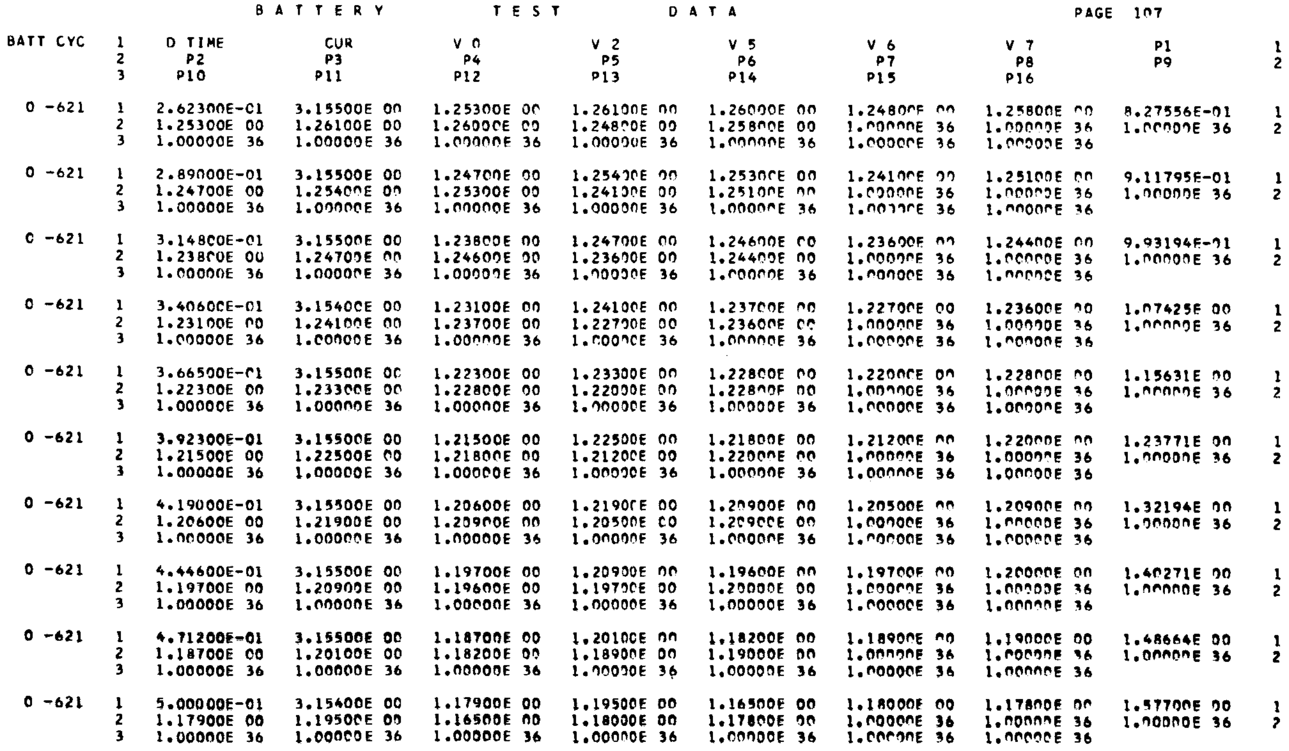


Figure 15—Data list routine functional flow diagram



```
                         B A T T E R Y             T E S T            D A T A                                         PAGE   28

BATT   CYC     FLD NA     STD. DEV.         X3              X2              X              C                           TIME  DUR.

 0   -1802      V 0     6.95535E-03  -6.73033E 00   4.11483F 00  -1.03752E 00   1.35755E 00                         4.97200E-01
                V 1     4.54547E-03  -5.68365E 00   3.58894E 00  -9.59844E-01   1.35496E 00
                V 2     1.26264E-03  -2.30172E 00   1.69965E 00  -6.99512E-01   1.35754E 00
                V 3     1.69028E-03  -4.22969E 00   2.63438E 00  -8.09082E-01   1.35938E 00
                V 4     3.70210E-03  -5.67769E 00   3.55046E 00  -9.63330E-01   1.36563E 00
               P1-8     4.89995E-01   4.92201E-01   4.98380E-01   4.95353E-01   4.94914E-01   1.80200E 03   1.00000E 36   1.00000E 36
               P9-16    1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000F 36   1.00000E 36

 0   -1803      V 0     6.92380E-03  -6.82923E 00   4.21276E 00  -1.06713E 00   1.36028E 00                         4.97100E-01
                V 1     4.94756E-03  -5.55977E 00   3.46991E 00  -9.31651E-01   1.35383E 00
                V 2     6.67118E-04  -1.89820E 00   1.33945E 00  -6.07072E-01   1.35218E 00
                V 3     1.96650E-03  -4.00127E 00   2.44929E 00  -7.63697E-01   1.35707E 00
                V 4     4.49625E-03  -5.58380E 00   3.46785E 00  -9.38566E-01   1.36446E 00
               P1-8     4.90071E-01   4.92065E-01   4.98425E-01   4.95557E-01   4.95251E-01   1.80300E 03   1.00000E 36   1.00000E 36
               P9-16    1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36

 0    1895      V 0     1.88803E-03   6.49399E-03  -4.77360E-02   1.32626E-01   1.26768E 00                         3.25620E 00
                V 1     1.32841E-03   5.70765E-03  -4.30713E-02   1.25745E-01   1.26434F 00
                V 2     1.96291E-03   5.54422E-03  -4.24840E-02   1.35146E-01   1.25665E 00
                V 3     2.71783E-03   6.98752E-03  -5.10326E-02   1.47195E-01   1.25470E 00
                V 4     2.50580E-03   7.48982E-03  -5.38947E-02   1.52183E-01   1.25548E 00
               P1-8     3.57125E 00   3.55864E 00   3.58023E 00   3.57999E 00   3.58810E 00   1.89500E 03   1.00000E 36   1.00000E 36
               P9-16    1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36

 0   -1895      V 0     8.91585E-04  -2.51162E 00   1.67053E 00  -6.49211E-01   1.34679F 00                         4.97000E-01
                V 1     1.05770E-03  -1.51446E 00   1.17984E 00  -5.68513E-01   1.34124E 00
                V 2     8.61961E-04  -2.02348E 00   1.38570E 00  -6.09359E-01   1.35192E 00
                V 3     3.07707E-03  -5.32061E 00   3.25845E 00  -8.95264E-01   1.36228E 00
                V 4     5.65903E-03  -6.73805E 00   4.23035E 00  -1.07461E 00   1.37012E 00
               P1-8     4.95379E-01   4.97251E-01   4.97988E-01   4.94932E-01   4.94849E-01   1.89500E 03   1.00000F 36   1.00000E 36
               P9-16    1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000F 36   1.00000F 36   1.00000E 36

 0    1896      V 0     2.03353E-03   7.21898E-03  -5.16223E-02   1.38765E-01   1.26491E 00                         3.25640E 00
                V 1     1.70958E-03   5.79357F-03  -4.37436E-02   1.27491F-01   1.26246E 00
                V 2     2.44696E-03   6.40404E-03  -4.69150F-02   1.41908F-01   1.25359E 00
                V 3     3.22428E-03   8.50101E-03  -5.89335E-02   1.58948E-01   1.25011F 00
                V 4     2.66905E-03   8.51614E-03  -5.97418E-02   1.62291E-01   1.25051F 00
               P1-8     3.57084E 00   3.55713E 00   3.57969E 00   3.57940E 00   3.58751E 00   1.89600E 03   1.00000F 36   1.00000E 36
               P9-16    1.00000E 36   1.00000E 36   1.00000F 36   1.00000E 36   1.00000E 36   1.00000F 36   1.00000E 36   1.00000E 36

 0   -1896      V 0     5.00252E-04  -2.63592E 00   1.75197E 00  -6.66601F-01   1.34765F 00                         4.96800E-01
                V 1     1.03253E-03  -1.64912E 00   1.22165F 00  -5.68493E-01   1.34042E 00
                V 2     5.74515E-04  -2.12428E 00   1.47056E 00  -6.30650E-01   1.35267E 00
                V 3     2.61218E-03  -4.73674E 00   2.89647E 00  -8.35601E-01   1.35972E 00
                V 4     4.26278E-03  -5.91396E 00   3.71922E 00  -9.93431E-01   1.36709F 00
               P1-8     4.94971E-01   4.96464E-01   4.97548E-01   4.94911E-01   4.94820E-01   1.89600F 03   1.00000E 36   1.00000E 36
               P9-16    1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000F 36   1.00000E 36   1.00000E 36   1.00000E 36
```

Figure 16 — Sample printout of pseudo data tape — least squares fit request



```
                         B A T T E R Y             T E S T            D A T A                                   PAGE  107

BATT CYC    1     D TIME          CUR           V 0           V 2           V 5           V 6           V 7            P1            1
            2       P2             P3            P4            P5            P6            P7            P8            P9            2
            3       P10            P11           P12           P13           P14           P15           P16

  0 -621    1   2.62300E-01   3.15500E 00   1.25300E 00   1.26100E 00   1.26000E 00   1.24800F 00   1.25800E 00   8.27556E-01   1
            2   1.25300E 00   1.26100E 00   1.26000E 00   1.24800E 00   1.25800E 00   1.00000E 36   1.00000F 36   1.00000E 36   2
            3   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36

  0 -621    1   2.89000E-01   3.15500E 00   1.24700E 00   1.25400E 00   1.25300E 00   1.24100E 00   1.25100E 00   9.11795E-01   1
            2   1.24700E 00   1.25400E 00   1.25300E 00   1.24100E 00   1.25100E 00   1.00000E 36   1.00000E 36   1.00000E 36   2
            3   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36

  0 -621    1   3.14800E-01   3.15500E 00   1.23800E 00   1.24700E 00   1.24600E 00   1.23600F 00   1.24400E 00   9.93194F-01   1
            2   1.23800E 00   1.24700E 00   1.24600E 00   1.23600E 00   1.24400E 00   1.00000E 36   1.00000E 36   1.00000E 36   2
            3   1.00000E 36   1.00000E 36   1.00000E 36   1.00000F 36   1.00000E 36   1.00000E 36   1.00000E 36

  0 -621    1   3.40600E-01   3.15400E 00   1.23100E 00   1.24100E 00   1.23700E 00   1.22700E 00   1.23600E 00   1.07425F 00   1
            2   1.23100E 00   1.24100E 00   1.23700E 00   1.22700E 00   1.23600E 00   1.00000E 36   1.00000E 36   1.00000E 36   2
            3   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36

  0 -621    1   3.66500E-01   3.15500E 00   1.22300E 00   1.23300E 00   1.22800E 00   1.22000E 00   1.22800E 00   1.15631E 00   1
            2   1.22300E 00   1.23300E 00   1.22800E 00   1.22000E 00   1.22800F 00   1.00000E 36   1.00000E 36   1.00000E 36   2
            3   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36

  0 -621    1   3.92300E-01   3.15500E 00   1.21500E 00   1.22500E 00   1.21800E 00   1.21200E 00   1.22000E 00   1.23771E 00   1
            2   1.21500E 00   1.22500E 00   1.21800E 00   1.21200E 00   1.22000E 00   1.00000E 36   1.00000E 36   1.00000E 36   2
            3   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000F 36   1.00000E 36   1.00000E 36

  0 -621    1   4.19000E-01   3.15500E 00   1.20600E 00   1.21900E 00   1.20900F 00   1.20500E 00   1.20900E 00   1.32194E 00   1
            2   1.20600E 00   1.21900E 00   1.20900E 00   1.20500E 00   1.20900E 00   1.00000E 36   1.00000E 36   1.00000E 36   2
            3   1.00000E 36   1.00000E 36   1.00000E 36   1.00000F 36   1.00000E 36   1.00000E 36   1.00000E 36

  0 -621    1   4.44600E-01   3.15500E 00   1.19700E 00   1.20900E 00   1.19600E 00   1.19700F 00   1.20000E 00   1.40271E 00   1
            2   1.19700E 00   1.20900E 00   1.19600E 00   1.19700E 00   1.20000E 00   1.00000E 36   1.00000E 36   1.00000E 36   2
            3   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36

  0 -621    1   4.71200E-01   3.15500E 00   1.18700E 00   1.20100E 00   1.18200E 00   1.18900E 00   1.19000E 00   1.48664E 00   1
            2   1.18700E 00   1.20100E 00   1.18200E 00   1.18900E 00   1.19000E 00   1.00000F 36   1.00000E 36   1.00000E 36   2
            3   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36

  0 -621    1   5.00000E-01   3.15400E 00   1.17900E 00   1.19500E 00   1.16500F 00   1.18000F 00   1.17800E 00   1.57700E 00   1
            2   1.17900E 00   1.19500E 00   1.16500E 00   1.18000E 00   1.17800E 00   1.00000E 36   1.00000E 36   1.00000E 36   ?
            3   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36   1.00000E 36
```

Figure 17—Sample printout of pseudo data tape — actual data request

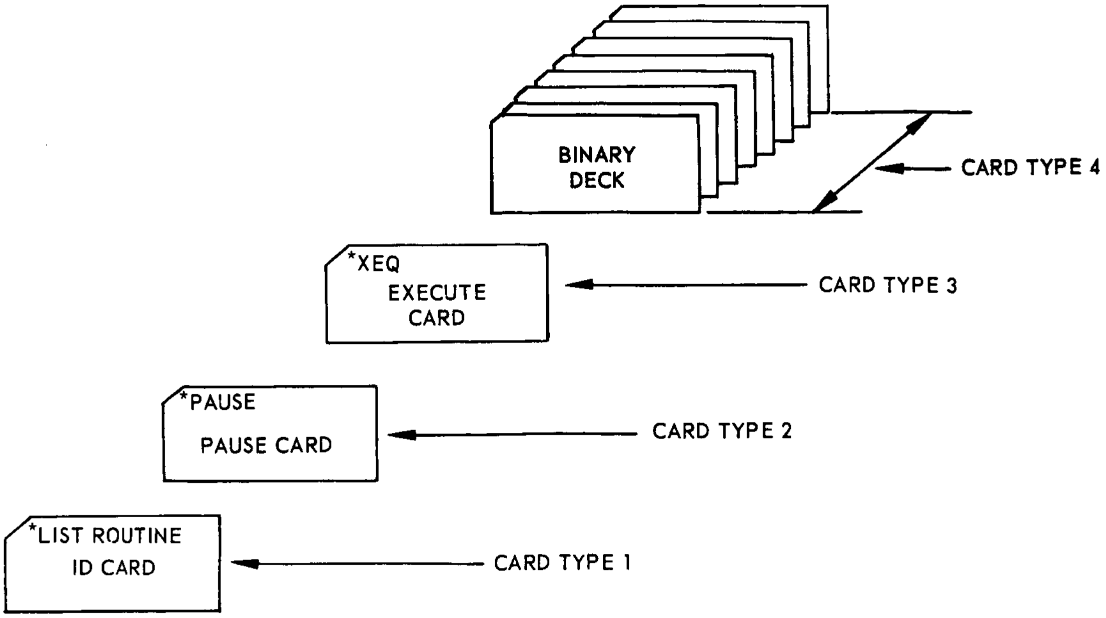


Figure 18—Data list routine card deck arrangement

Instructions to the operator

1. Mount input tape (retrieve data or pseudo data) on logical setting of B-9.

2. Use 132 print position paper.

3. Multiple tapes may be printed by pressing start after completion of each run.

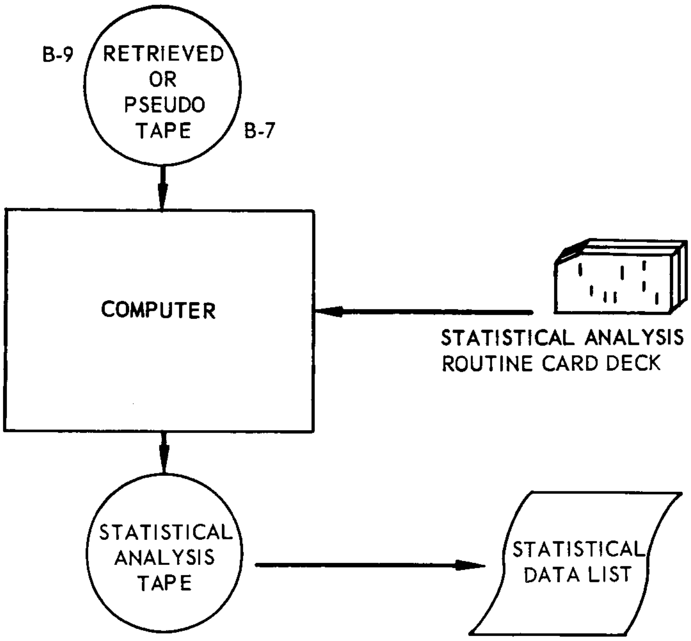
4. FORTRAN II – Execute.

5. Average machine time for a run is 2 minutes.

Statistical Analysis Routine

The statistical analysis routine is used to perform statistical calculations involving the fields and pseudo fields on the retrieved or pseudo field data tapes, and print out the results in suitable data listings (Figure 19). The analyst may select from a retrieved data or pseudo data input tape the fields and/or pseudo fields to be processed and the statistical calculations to be made.

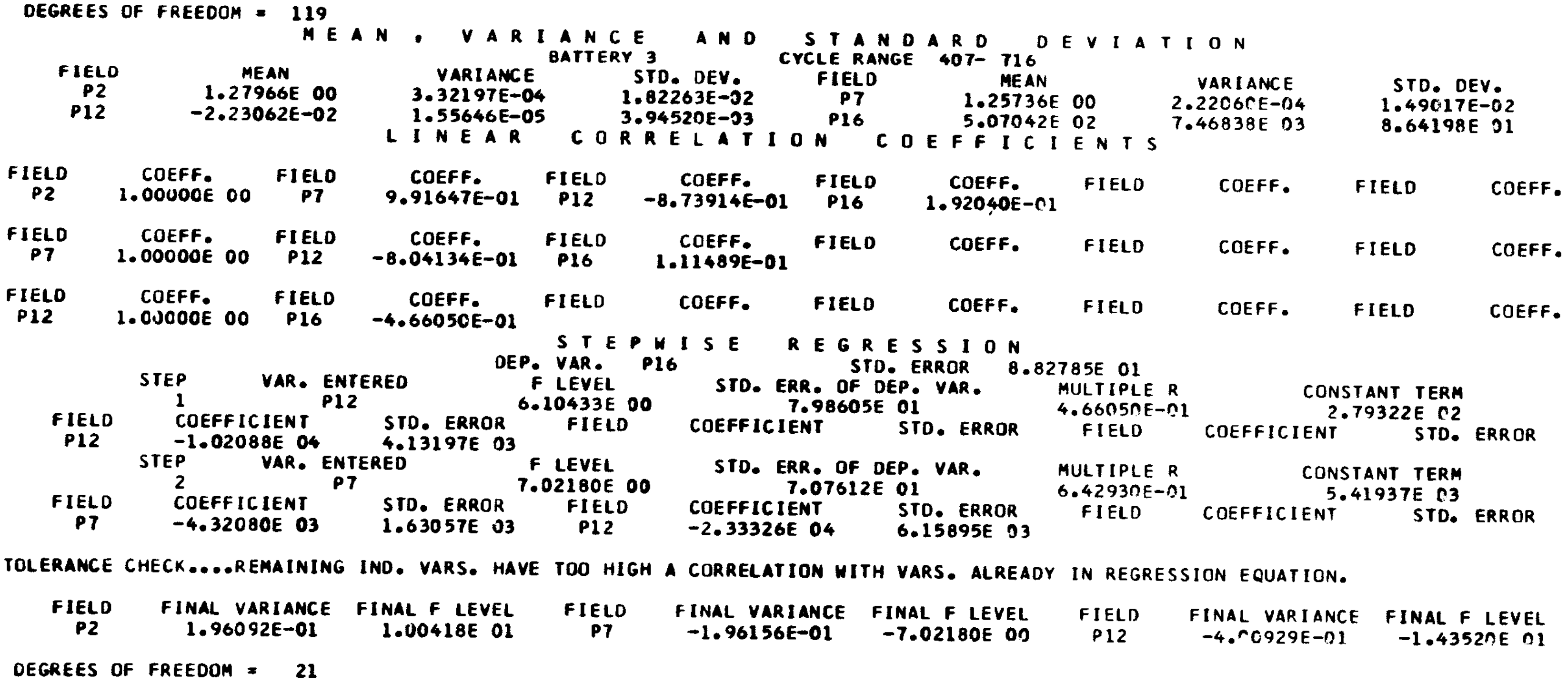
Figure 19—Statistical analysis routine functional flow diagram


B-9
RETRIEVED OR PSEUDO TAPE
B-7
COMPUTER
STATISTICAL ANALYSIS ROUTINE CARD DECK
STATISTICAL ANALYSIS TAPE
STATISTICAL DATA LIST

The routine card deck, with a control card added by the analyst, is used to control the computer operation. The selected fields are operated on statistically in three steps. In the first step, the mean, variance, and standard deviation of the selected fields are calculated. In the second step, a matrix of linear correlation coefficients is calculated. In the third step, the coefficients of the multiple linear regression equation expresses the dependence of one of the variables (the one specified by the analyst via input control card data), considered as the dependent variable on the others, considered as independent variables. The program computes the regression coefficients by "entering" the independent variables into the equation one at a time. At each "entrance," the program removes from the equation a variable whose contribution to the variance is insignificant, or, if no variable can be removed, enters into the equation that variable which produces the largest reduction in the "unexplained" variance of the dependent variables. The results of each step are printed out at the conclusion. Printed out as a heading are the retrieve request, the pseudo request, and the statistical request. Figure 20 is an example of a printout of a statistical analysis request when all three steps are requested. The headings are eliminated.

Card deck arrangement. The deck of input cards in proper order is illustrated in Figure 21. The cards types are:

Card type	Description
1	Identification card - asterisk in column 7 and run identifying comment in columns 7-72.
2	Pause card. Asterisk in column 1 and the letters PAUSE in columns 7-11.
3	Execute card. Asterisk in column 1 and the letters XEQ in columns 7-9.
4	Binary program deck.



```
DEGREES OF FREEDOM =  119
                     M E A N  ,   V A R I A N C E    A N D    S T A N D A R D    D E V I A T I O N
                                                 BATTERY 3            CYCLE RANGE  407-  716
  FIELD          MEAN            VARIANCE        STD. DEV.        FIELD          MEAN             VARIANCE        STD. DEV.
   P2        1.27966E 00       3.32197E-04      1.82263E-02        P7         1.25736E 00       2.22060E-04     1.49017E-02
   P12      -2.23062E-02       1.55646E-05      3.94520E-03        P16        5.07042E 02       7.46838E 03     8.64198E 01
                                L I N E A R    C O R R E L A T I O N    C O E F F I C I E N T S

FIELD    COEFF.     FIELD     COEFF.      FIELD     COEFF.       FIELD     COEFF.      FIELD     COEFF.     FIELD     COEFF.
 P2   1.00000E 00    P7   9.91647E-01      P12   -8.73914E-01     P16   1.92040E-01

FIELD    COEFF.     FIELD     COEFF.      FIELD     COEFF.       FIELD     COEFF.      FIELD     COEFF.     FIELD     COEFF.
 P7   1.00000E 00    P12  -8.04134E-01     P16    1.11489E-01

FIELD    COEFF.     FIELD     COEFF.      FIELD     COEFF.       FIELD     COEFF.      FIELD     COEFF.     FIELD     COEFF.
 P12  1.00000E 00    P16  -4.66050E-01
                                         S T E P W I S E    R E G R E S S I O N
                                   DEP. VAR.   P16                 STD. ERROR   8.82785E 01
          STEP       VAR. ENTERED          F LEVEL         STD. ERR. OF DEP. VAR.        MULTIPLE R            CONSTANT TERM
           1             P12             6.10433E 00           7.98605E 01               4.66050E-01             2.79322E 02
   FIELD      COEFFICIENT        STD. ERROR       FIELD      COEFFICIENT       STD. ERROR      FIELD     COEFFICIENT       STD. ERROR
   P12       -1.02088E 04        4.13197E 03
          STEP       VAR. ENTERED          F LEVEL         STD. ERR. OF DEP. VAR.        MULTIPLE R            CONSTANT TERM
           2             P7              7.02180E 00           7.07612E 01               6.42930E-01             5.41937E 03
   FIELD      COEFFICIENT        STD. ERROR       FIELD      COEFFICIENT       STD. ERROR      FIELD     COEFFICIENT       STD. ERROR
   P7        -4.32080E 03        1.63057E 03       P12      -2.33326E 04       6.15895E 03

TOLERANCE CHECK....REMAINING IND. VARS. HAVE TOO HIGH A CORRELATION WITH VARS. ALREADY IN REGRESSION EQUATION.

   FIELD    FINAL VARIANCE   FINAL F LEVEL     FIELD    FINAL VARIANCE   FINAL F LEVEL     FIELD    FINAL VARIANCE   FINAL F LEVEL
    P2       1.96092E-01      1.00418E 01       P7      -1.96156E-01     -7.02180E 00       P12     -4.00929E-01     -1.43520E 01

DEGREES OF FREEDOM =   21
```

Figure 20—Sample printout of statistical analysis tape

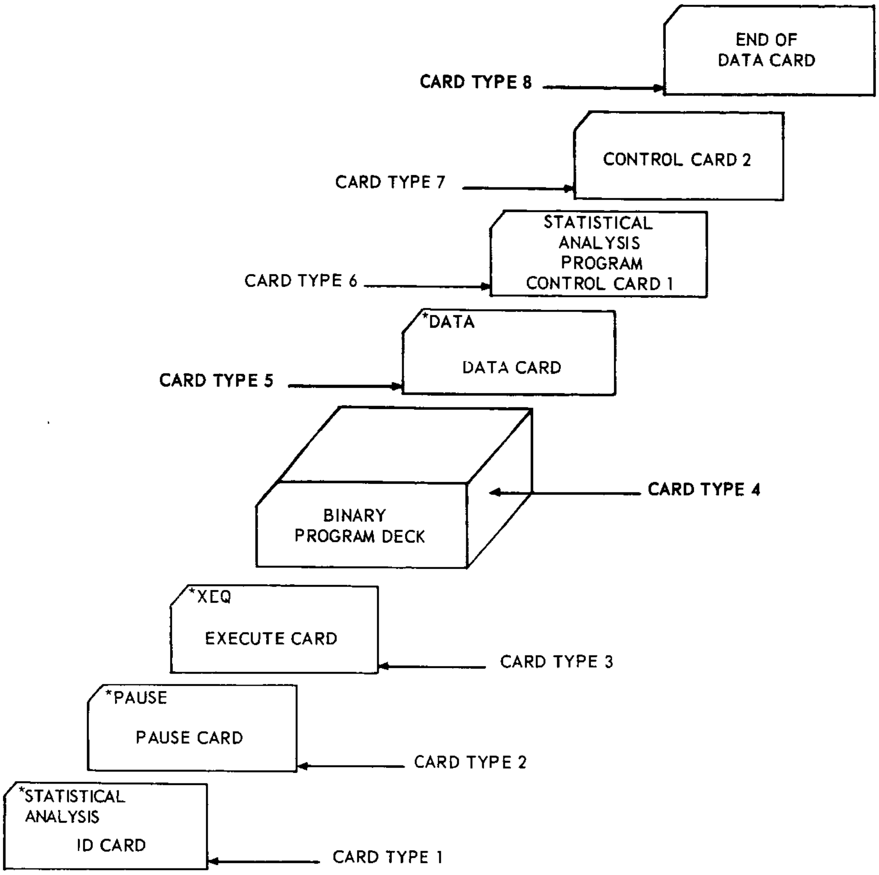
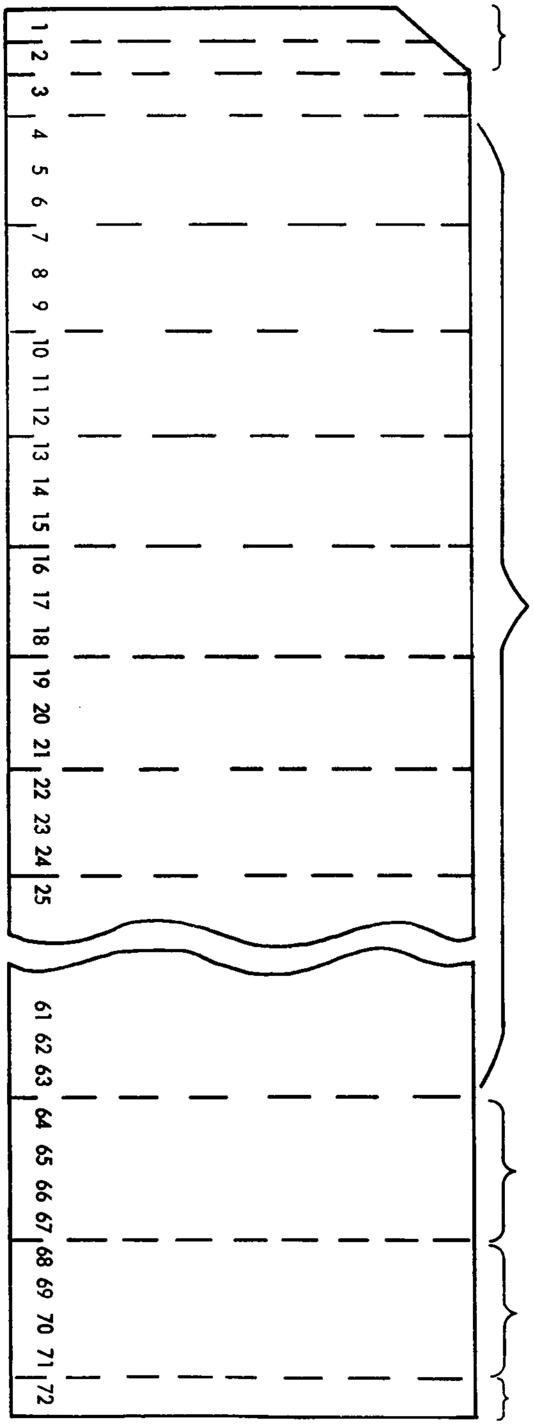


Figure 21—Statistical analysis routine card deck arrangement

Card type	Description
5	Data card. Asterisk in column 1 and the letters DATA in columns 7-10.
6	Statistical analysis control card No. 1.
7	Statistical analysis control card No. 2.
8	End of data card - a card with a 9 in column 1.

Card types 6 and 7 must be supplied by the analyst. The other types appear whenever a request for a statistical analysis is made.

Control cards. The first control card allows the analyst to select the fields to be processed and the statistical operations to be performed. At least one control card must be prepared by the analyst for the statistical analysis routine to work. To run multiple requests, an additional set of control cards must be added after control card 2 prior to the end of the data card.



Item	Description
1	Run tape
2	Data option
3	Dependent variable
4	Independent variable field and pseudo field specification
5	F level entry
6	F level removal
7	Addition field indicator

Figure 22—Statistical analysis routine control card

Control card 1

			Description	
Item	Column	Digit in column	Input tape	Operations
1	1		In column 1 is specified the type of statistical calculations, and the retrieved or pseudo data input tape. The following are the choices the analyst can make when requesting a statistical analysis:	
		1	Retrieved tape	Mean, variance, and standard deviation.
		2	Retrieved tape	Mean, variance, standard deviation, and linear correlation.
		3	Retrieved tape	Mean, variance, standard deviation, linear correlation, and multiple stepwise linear regression.
		4	Pseudo field tape	Same as 1.
		5	Pseudo field tape	Same as 2.
		6	Pseudo field tape	Same as 3.
2	2		Column 2 describes the fields or pseudo fields to be processed.	
			Field - pseudo field continuation requested	
		1	All fields and pseudo fields on input tape.	
		2	All fields and specified pseudo fields. (See items 3 and 4)	
		3	All pseudo fields and specified fields. (See items 3 and 4)	
		4	Only specified fields and specified pseudo fields. (See items 3 and 4)	
3	3-5		Columns 3-5 contain the field or pseudo field designated to be the dependent variable. It is specified in the form of a field number (F01) for retrieved data, and a pseudo field number (P01) for pseudo field data which is consistent with items 1 and 2.	
4	7-63		Columns 7 through 63 contain all fields and pseudo fields to be processed as independent variables in the statistical analysis routine. Columns 7-63 are divided into segments of three columns, each for fields specified by F01 and pseudo fields P01, etc.	
5	64-67		Columns 64 through 67 contain the "entry" F level value to be used in conjunction with a multiple regression type run (item 1 = 3 or 6).	
6	68-71		Columns 68 through 71 contain the "removal" F level value to be used in conjunction with a multiple regression type run (item 1 = 3 or 6).	
7	72		Column 72 contains the additional input control card indicator. A value of zero means no additional control card follows. A non-zero value indicates the presence of an additional control card.	

Control card 2

Additional independent variable fields can be requested by using a second control card. Columns 1-60 of the second card are divided into segments of three columns each. The additional fields and pseudo fields are designated in the same manner as described in item 4.

Instructions to the computer operator

(1) Mount input retrieved tape on a logical setting of B-9. Input pseudo tape is mounted on a logical tape setting of B-7.

(2) Use 132 print position paper.

(3) Multiple tapes may be printed by pressing "START" after completion of each run.

(4) FORTRAN II - EXECUTE.

(5) Machine time for the average run is estimated at 2 minutes. (Since the nature of the program tends to produce considerable output, a cutoff limit of 500 pages per run was added to the program.)

Printed Plot Routine

The printed plot routine is used to abstract the information requested by the analyst in the retrieve-formatter and pseudo field generator routines, and to present the results in a printed plot form (Figure 23). The routine card deck, with control cards included by the analyst, controls the

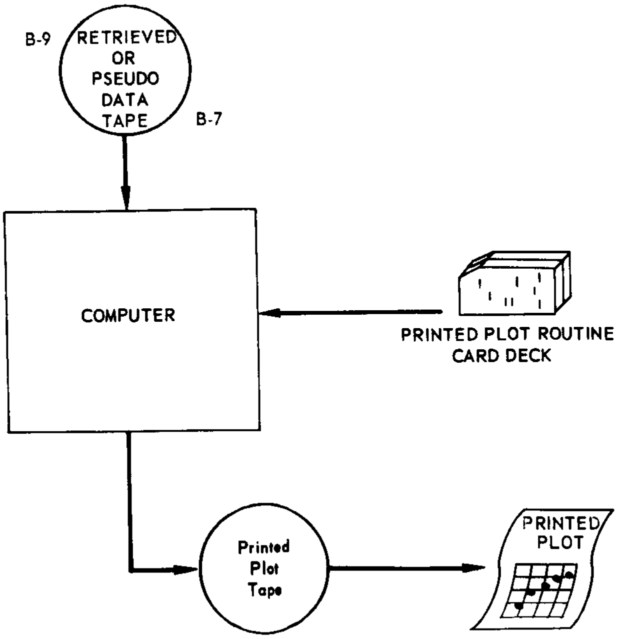


Figure 23—Printed plot routine functional flow diagram

computer so that the data is extracted from the input tape, recorded on an intermediate output tape, and then printed out in the plot form with dimensions. Also printed are the retrieve-formatter request statements, the pseudo generator request statements, and the printed plot request statements.

Figure 24 is a sample of a printed plot. The term printed plot refers to the fact that the data points are printed by an IBM 1460 computer which presents the curve in a discontinuous format similar to that of a multiprint recorder. Either numerals, letters or symbols can be used to designate data points.

Any five variables (Y_1, Y_2, Y_3, Y_4, Y_5) corresponding to any retrieve or pseudo fields can be plotted vs. any other variable (X) which can be a retrieve or pseudo field. The analyst has the capability of defining the coordinates and their sensitivity, setting the size of the grid spacing, and the total width and length of the graph image. The maximum allowed for one continuous plot is 500 lines (8 pages).

Card deck arrangement. The card deck arrangement is illustrated in Figure 25. The following are the card types required and a brief description of each:

Card type	Description
1	Identification card. Asterisk in column 1 and run identification numbers in columns 7-72.
2	Pause card. Asterisk in column 1 and the letters PAUSE in columns 7-11.
3	Execute card. Asterisk in column 1 and the letters XEQ in columns 7-9.
4	Binary routine deck.
5	Data card. Asterisk in column 1 and the letters DATA in columns 7-10.
6	Control card 1.
7	Control card 2.
8	Control card 3.

Card types 2-5 are always the same for every run in which a printer plot is requested. Type 1 is a statement defined by the analyst for identification purposes which does not affect the routine. Types 6-8 control the routine and are to be supplied by the analyst. The three control cards are required for each request of the printer plot. For successive requests, additional groups of the three control cards are placed behind the first group of three.

Control cards. Control card 1 chooses the input tape, selects the scale factors, and determines the X and Y variables to be plotted. Control card 2 sets the Y variable maximum and minimum and determines the direction of X and Y. Control card 3 sets the X variables maximum and minimum, and the number and value of ranges to be plotted.

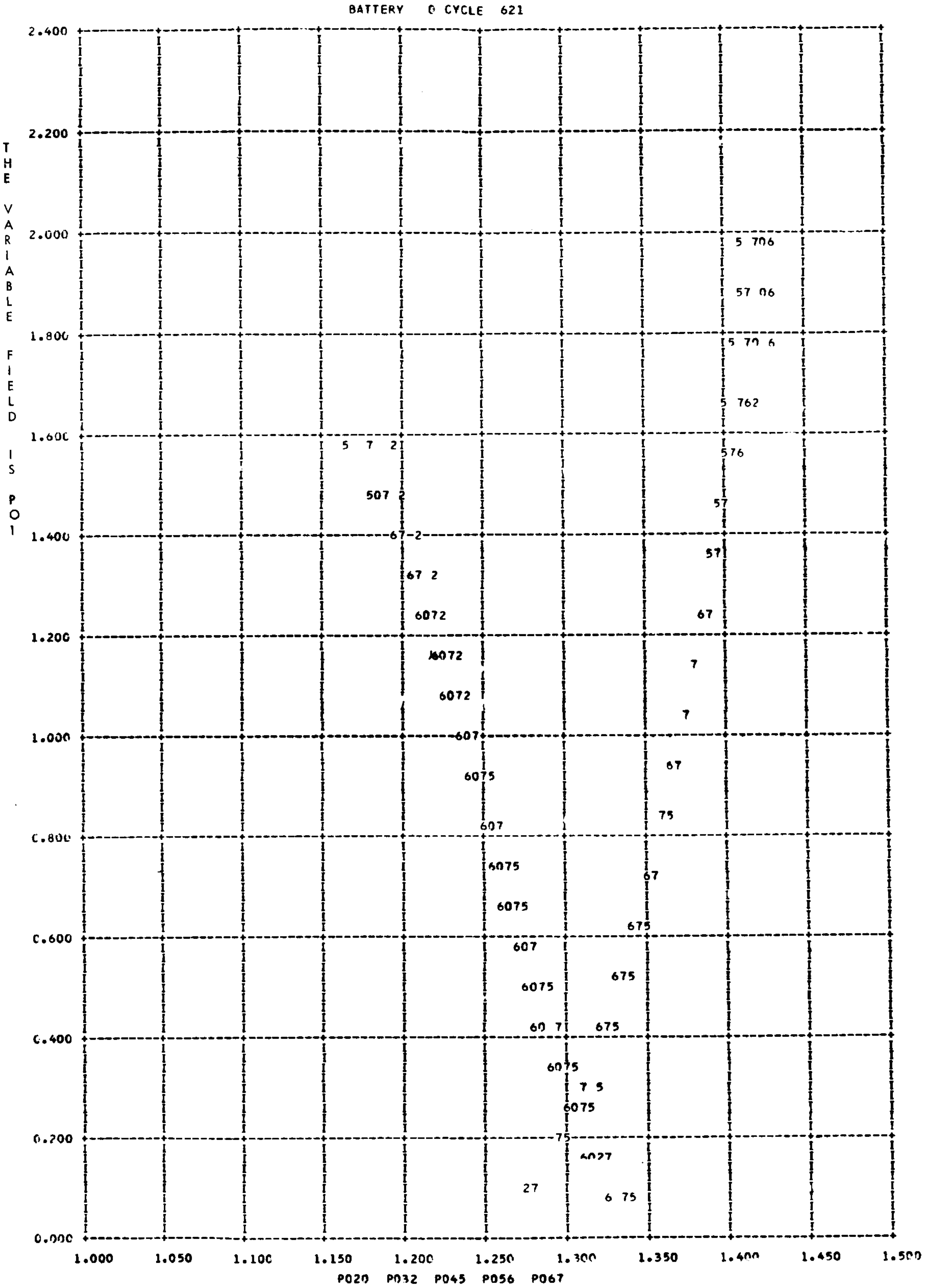


Figure 24—Sample printout of printed plot tape

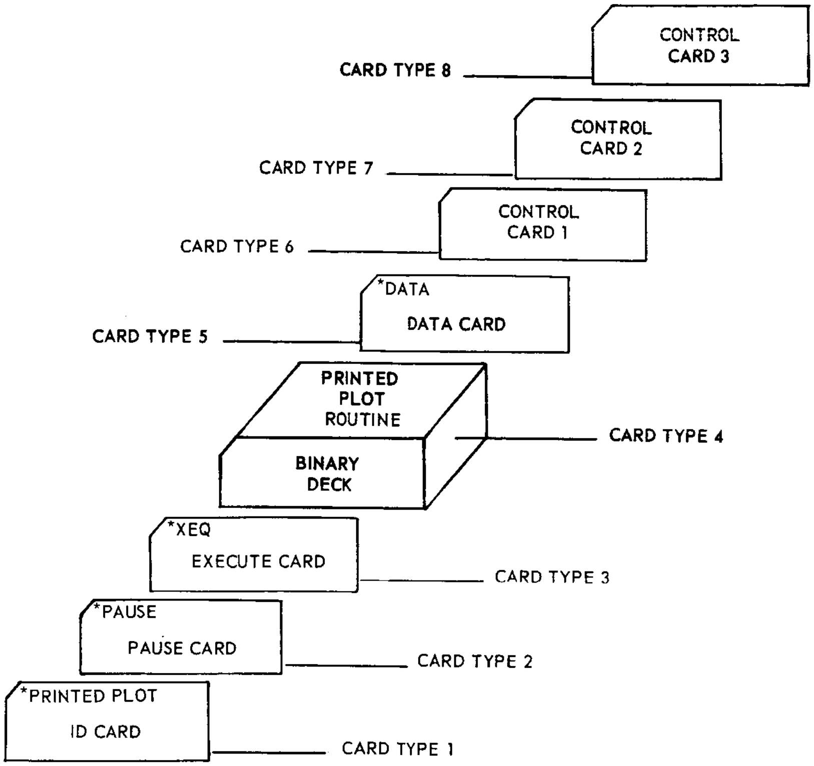
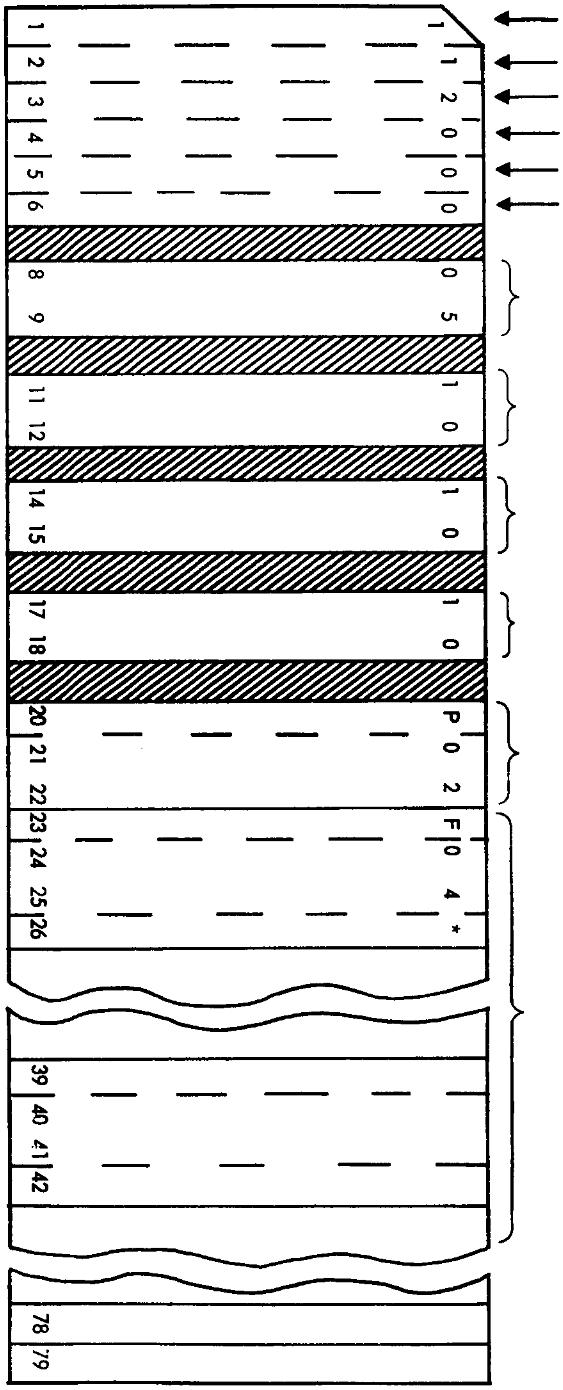


Figure 25—Printed plot routine card deck arrangement

Control card 1 (Figure 26)

Item	Control card column	Description
1	1	A "0" in column 1 selects a retrieved data tape. A "1" in column 1 selects a pseudo data tape.
2	2	To use actual scale factors and three places to the right of the decimal for both X and Y, punch a "0" in column 2, and ignore items 3-6 (columns 3-6). (A "0" punch means actual scale factors, and three places to the right of the decimal point corresponding to 0, 3, 0, 3 in columns 3-6). For any change punch a 1 in column 2.
3	3	The numeral punched in column 3 is equal to I where printed values of the ordinate (Y) are 10^{I} times the actual values. A zero in this column as shown in the figure means the actual values are to be printed along the ordinate.
4	4	The numeral punched in column 4 is equal to J where printed values of the ordinate (Y) will have J digits following the decimal point. A "2" punched in this column requires two places following the decimal so that the value along the ordinate will have two digits X.XX.



Item	Description
1	Run type
2	Actual scale factors
3	Y ordinate multiplier
4	Y ordinate decade
5	X abscissa multiplier
6	X abscissa decade
7	Horizontal lines
8	Horizontal spaces
9	Vertical lines
10	Vertical spaces
11	X Variable field
12	Y Variable fields

Figure 26—Printed plot routine control card

Item	Control card column	Description
5	5	Column 5 is the same as item 3 except referenced to the abscissa (X). A "0" is shown in the figure.
6	6	Column 6 is the same as item 4 except references to the abscissa (X). A "0" is shown in the figure which indicates no digits to the right of the decimal point for X.
7	8-9	Columns 8 and 9 denote the number of horizontal grid lines in the graph image. The example shows 5.
8	11-12	Columns 11 and 12 are used to denote the number of spaces between the horizontal grid lines. The example shows 10. NOTE: Item 7 times item 8 is the number of horizontal lines for printing the value of a variable. For a one page plot the value is 50.
9	14-15	Columns 14 and 15 are used to denote the number of vertical grid lines in the graph image. The example shows 10.
10	17-18	Columns 17 and 18 are used to denote the number of spaces between the vertical grid lines. The example shows 10. NOTE: Item 9 times item 10 is the number of vertical lines for printing the value of the variable. For a 1 page plot the value is 100.
11	20-22	Columns 20, 21, and 22 are where the analyst places the X variable. Column 20 must be an "F" or "P" indicating a field or a pseudo field. Columns 21 and 22 contain the actual field number. The example shows that the "X" value is a pseudo field and for purposes of illustration was set equal to cycle in a pseudo field generator routine request.
12	23-26 27-30 31-34 35-38 39-42	The Y variables are placed starting in columns 23 through 26 for the first "Y," thru columns 39-42 for the last or 5th "Y". From one to five, "Y's" may be plotted. The format for the control card for Y values is the same as for the X value except that the last character of each "Y" field requested is the plotting character. The example shows that the value to be plotted for "Y_1" is Field 4 (energy efficiency using an endcard data tape) and the plotting character is an asterisk.

Control card 2 (Figure 27)

Item	Control card column	Description
13	1-10	Y MINIMUM - columns 1-10 contain the minimum value of all the Y variables for a given run. The value does not have to start in column 1

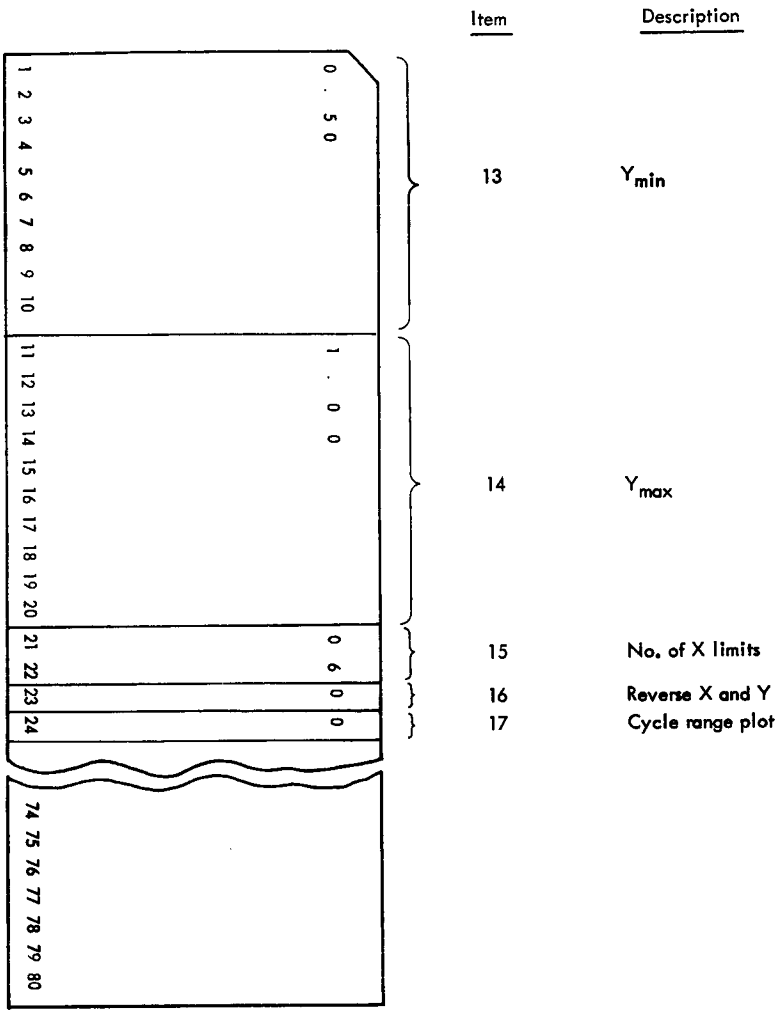


Figure 27—Printed plot routine control card 2

Item — Control card column — Description

13 (continued) — but must be contained in columns 1-10. A decimal point must be present. In the example, the minimum value of F04 (energy efficiency) is 0.50.

14 — 11-20 — Y MAXIMUM - columns 11-20 contain the maximum value of all the Y variables for a given run. Format is the same as Y MINIMUM. The example shows Y MAXIMUM to be 1.00. Y MINIMUM and Y MAXIMUM will appear along the ordinate grid lines outside the edge of the graph.

15 — 21-22 — Columns 21 and 22 contain the number of X coordinate limits to be read, from control card 3 (Figure 27). Every plot has two limits - an X maximum and X minimum. The sample control card in Figure 26 shows item 15 equal to 06 because there are 3 pairs of X maximum and X minimum

Item	Control card column	Description
15 (continued)		from control card 3 to be read. If a second control 3 were necessary for additional ranges, the number of fields used would be included in the count for item 15.
16	23	If column 23 contains a zero then the X variable will appear along the abscissa, and the y variables will appear along the ordinate. Any non-zero value in column 23 will reverse the above statement (Figure 28).
17	24	This is used only for endcard or outcard least square fit requests, where each plot is a cycle range. If column 24 contains a zero as in the example then all the pairs of ranges on control card 3 are reduced to a single range. If column 24 contains a "1" punch, then each pair of ranges on control card 3 is equivalent to a complete range on the tape. All ranges on the tape to be plotted must appear on control card 3 if the "1" punch in column 24 option is used, and must be in the same sequence as requested in the retrieve-formatter-routine.

Control card 3 (Figure 29)

This control card describes the minimum and maximum values of X for each plot (range).

Item	Columns	Description
18	1-10	These columns contain the minimum value of X (X min) including decimal point for the first range.
19	11-20	These columns contain the maximum value of X (X max) including a decimal point for the first range.
20	21-30	Same as item 18 except it applies to range 2.
21	31-40	Same as item 19 except it applies to range 2.
22	41-50	Same as item 19 except it applies to range 3.
23	51-60	Same as item 19 except it applies to range 3.

The example in Figure 29 shows the first cycle range is 0 to 100, the second range is 100 to 200 and the third is 200 to 300.

An additional control card 3 can be used if additional ranges are required. A maximum of seven are allowed. In the sample control cards (Figures 25-27), a 1 appears in column 23 of

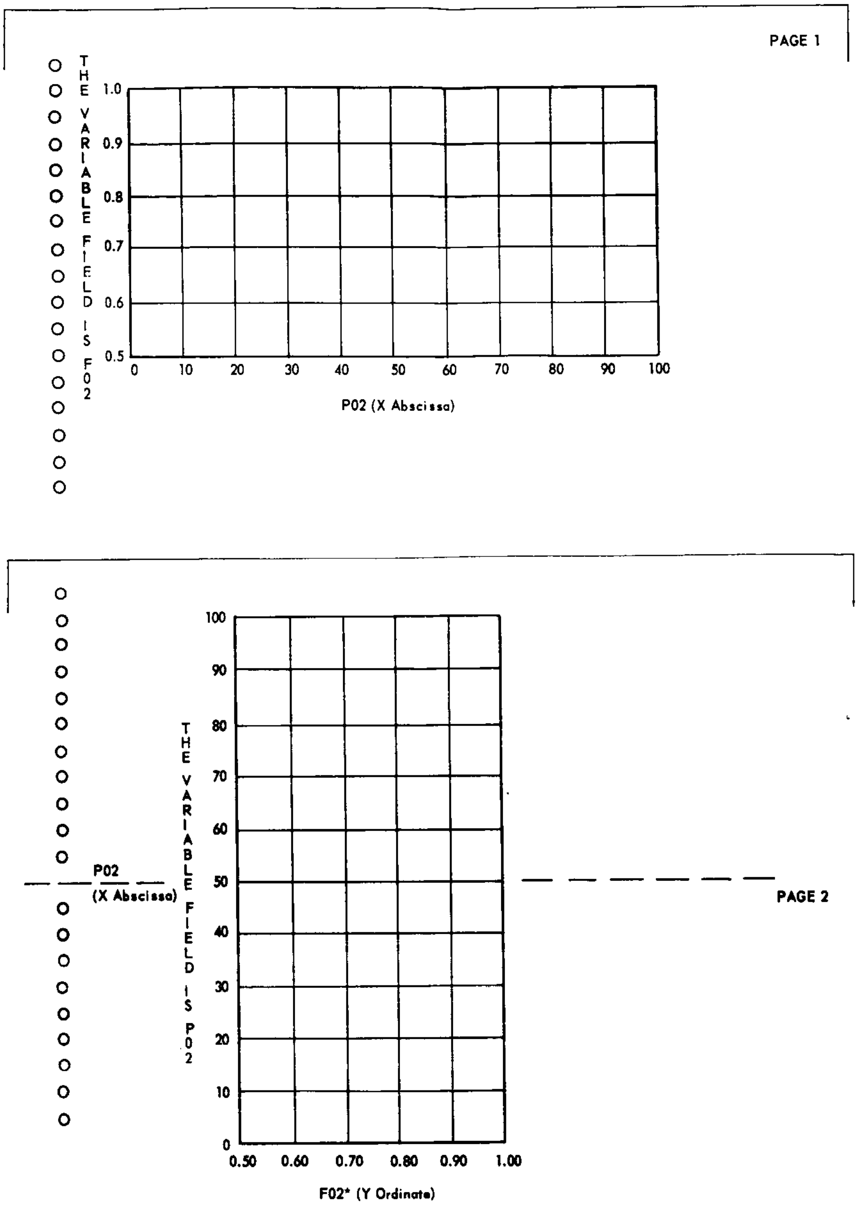
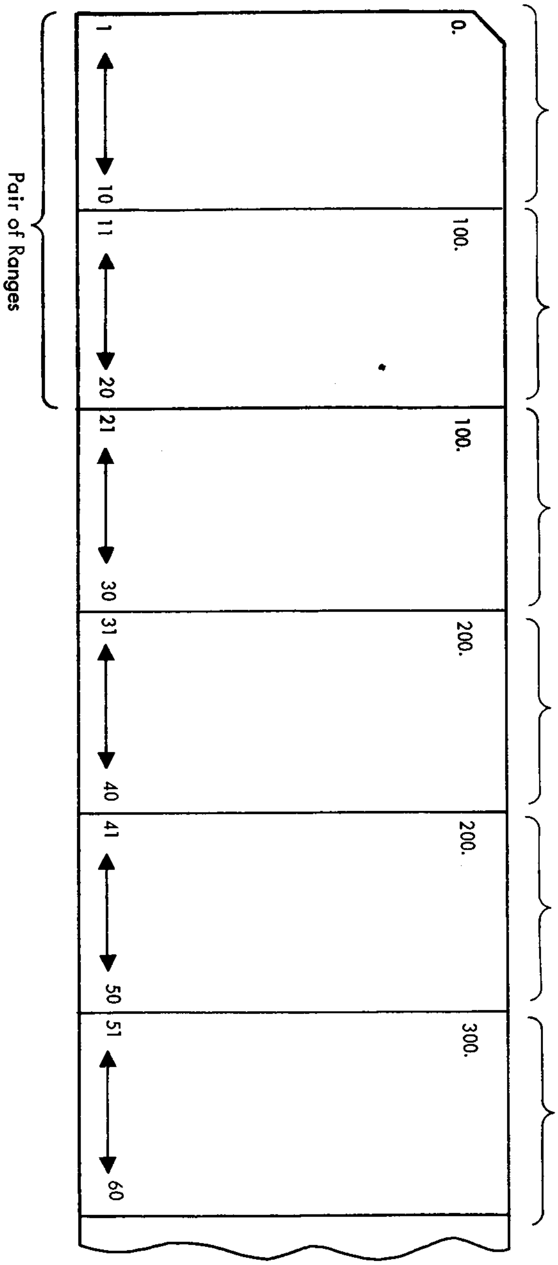


Figure 28—Printed plot routine coordinate arrangement



Item	Description
18	X_{min} range 1
19	X_{max} range 1
20	X_{min} range 2
21	X_{max} range 2
22	X_{min} range 3
23	X_{min} range 3

Figure 29—Printed plot routine control card 3

control card 2. Only one range is required with X_{min} equal to 0 and X_{max} equal to 300. Item 15 (control card 2) would have to be changed to 02.

There are certain restrictions in this routine which must be considered when the analyst is preparing a run:

1. When plotting endcard data, whether retrieved or pseudo, the program plots by cycle range. The number of cycles in a cycle range must not exceed 500.

2. A tape to be plotted with least square fit fields must be a pseudo tape, and only pseudo fields may be referenced.

3. The internal core image array is set at 7000 locations. Therefore, "N" must not exceed this limit.

   N = (Item 9 times item 8 + 1) × P,
   where P = (Item 11 times item 10 + 1)/6 rounded out to the nearest integer.

4. Items 8-11 cannot be set equal to 0.

5. Item 14 (Y_{min}) must be less than item 15 (Y_{max}).

Instructions to the operator. The following instructions must be given to the operator when submitting the routine card deck for analysis.

1. Mount retrieved data tape on logical tape setting B-9 or mount pseudo data tape on logical setting of B-7.

2. Use 132 print position paper.

3. Process multiple runs (additional sets of control cards 1-3) by pressing START after each run.

4. FORTRAN II - EXECUTE

5. Average machine time for a run is 2 minutes.

6. Request TONAP Carriage Tape

ACKNOWLEDGEMENTS

The author wishes to acknowledge the fine effort of Leslie Long, Al Flinn, Leo Kossa, Sheldon Wiseman and Bob Amos of the Martin Company who wrote the programs. Also a very special thanks is given to Joseph Sherfey of GSFC for his valuable support and to Lloyd Green of GSFC for advice on computer technology.

BIBLIOGRAPHY

1. Final Report Contract NAS-5-9755-1 Baltimore, Md.: Martin Company, May 1965.

2. Final Report Contract NAS-5-9755-2 Baltimore, Md.: Martin Company, May 1965.

3. Card to Tape EDITING PROGRAMS, Special Report Contract NAS-5-3027, Baltimore, Md.: Martin Company.

4. Green, L. G., Specifications for the Battery Analysis System, Aeronomy and Meteorology Division, GSFC, November 2, 1964.

5. Segal, D., CUPCRRG Share Subroutine, University of Pennsylvania, Philadelphia, Pa., July 10, 1964.

6. Ralston, A. and Wilf, Herbert S., Mathematical Methods for Digital Computers, Chapter 17, New York; John Wiley, 1960.

7. Carnattan, B. and Evans, L., Share Subroutine, University of Michigan, Ann Arbor, Michigan, March 1, 1961.

APPENDIX A

CARD TO TAPE EDITING PROGRAM

APPENDIX A

CARD TO TAPE EDITING PROGRAM

The card to tape editing program will take the raw data cards and record them on magnetic tape. This is followed by a series of tests in which this data is sorted and checked for validity. Any cycles found to be invalid are then deleted or have special codes placed on them. The new data becomes the endcard and outcard raw raw data tapes, and are merged in binary form with the other data already on tape to become up-to-date endcard and outcard raw data tapes.

Appendix A

1. Instructions for Use

General order of programs to run when merging new data with old data.

A. Cards to tape
   1. Outcards
   2. Endcard #2
   3. Endcard #1

B. Sort routine
   1. Outcards
   2. Endcard #2
   3. Endcard #1

C. Alpha check routine
   1. Outcards
   2. Endcard #2
   3. Endcard #1

D. Preliminary analysis routine
   1. Outcards
   2. Endcard #1

E. Delete and duplication elimination routine
   1. Outcards
   2. Endcard #1

F. BCD to binary routine
   1. Outcard
   2. Endcard #1 and Endcard #2

G. Binary merge routine
   1. Outcards
   2. Endcards

Note: See Flowcharts for merging endcards and outcards on Pages 2 and 3.

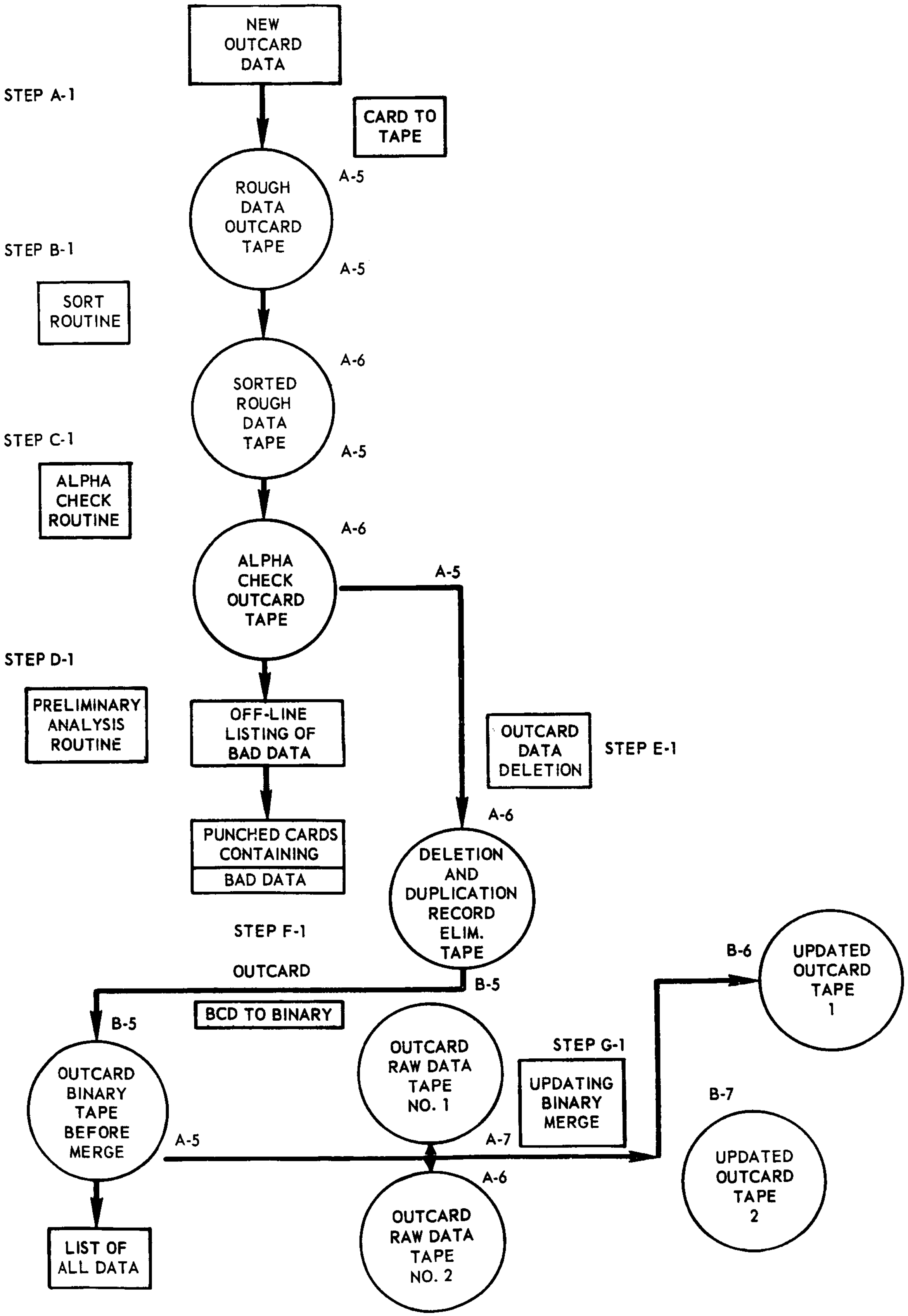


Figure A-1—Outcard Card to Tape Editing Programs

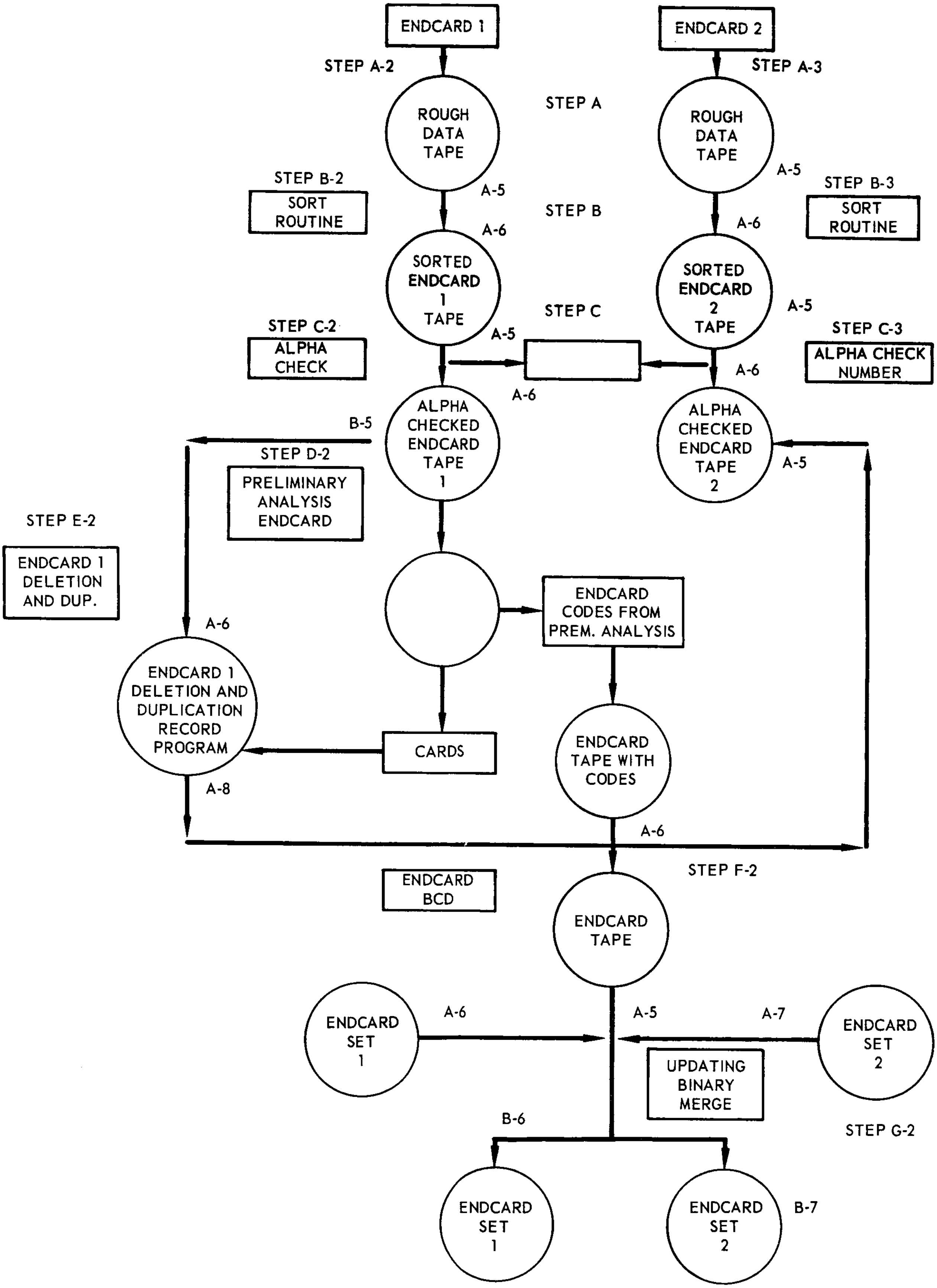


Figure A-2–Endcard Card to Tape Editing Programs

Steps A-1, A-2 and A-3

1. Title: **Preparation of data cards to be put on tape.**

2. General description:

   A. Data to be limited to a maximum of one 2400 foot tape or less. This is approximately 15 boxes of cards.

   B. Use separate tapes for the following:

      2. Endcard #1
      3. Endcard #2
      1. Outcards

   C. Make sure on update runs that no cycles from new data are already on the old data tape. Also make sure that the last cycle on each battery is complete for the new data to be merged.

   D. **See illustration on Page 7 of data card formats before recording on tape. Also see Pages** 8, 9, 10 for more detail on the data card formats.

3. Operating Procedure: **Cards are put on tape using an IBM 1401 Computer FARGO Program.**

Steps B-1, B-2 and B-3

1. Title: **Data sort program**

2. General description:

The purpose of the data sort program is to assure that all the data that has been put on tape is in order, according to battery number and cycle number.

3. **Input:**

Use the raw data tape (A5), also use a 3 scratch tapes (A7), (B5) and (B6).

4. **Output:**

Generates output tape (A6). Examples of print out are attached.

5. **Operating procedures:**

Attach a job card and computer operations tape log card to front of binary deck.

6. **Special considerations:**

a. Call out on job card to use IBSYS system.

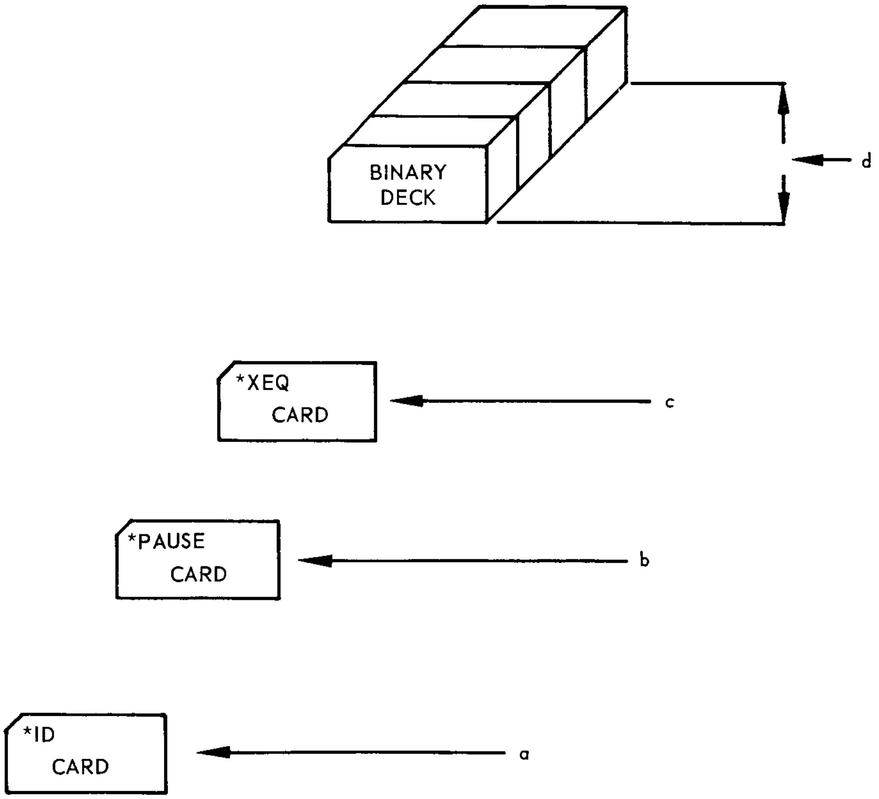
7. Sample deck set-up

a. Identification card. An asterisk punched in column 1 and in columns 7-72 a program description is punched.

b. Pause card. An asterisk punched in column 1 and PAUSE punched in columns 7-11.

c. Execute card. An asterisk punched in column 1 and XEQ punched in columns 7-9.

d. Binary deck.



NOTE: This setup is same for endcard #1, endcard #2 and outcards.

Figure A-3—Example deck setup - sort program

8. Example of on-line listing for SORT – B-1, B-2 and B-3:

```
$ID

$Pause

Oper. action pause

.. Continuing

          Remark endcard 1 data sort

Prepare units as follows..    *EP06

Unit A5       Input tape
Unit B5       Merge tape
Unit B6       Merge tape
Unit A6       Merge tape
Unit A7       Merge tape
Unit BO       Checkpoint tape

Press start when all units are property prepared and in ready status.

Oper. action pause

.. Continuing

Unit A6       closing output reel No. 0001  *OUO1


End of sort
```

Steps C-1, C-2 and C-3

1. Title: Alpha check program

2. General description:

This program will eliminate any card images that contain alpha characters. This program is used to place an additional record at very end of output tape that contains 999999 in columns 1-6. This is a last card signal for the next series of programs.

3. Input:

Use the tape generated from data sort (A5).

4. Output:

Generates output tape (A6). Examples of print-out are attached.

5. Operating procedures:

Attach a job card and computer operations tape log card to front of binary dec.

6. Sample deck setup

Same as for data sort, Page A-5.

Note: Endcard #1 tape and endcard #2 tape should be run separately.

7. Example of on-line listing for alpha check – C-1, C-2 and C-3:

```
*Alpha check program for endcards 1

*Pause

*XEQ

Execution

Last A5.  processing complete

        999999

Prepare A6 if processing incomplete.
```

Step D-1 (for Endcards – See Step D-2, page A-13)

1. Title: Outcard preliminary analysis program

2. General description:

This program computes the following for analysis:

a. Volts hours over a range of start and end time of half cycle.
b. Number of cards present on a half cycle.

c. Average voltage for first and last card on half cycle.

d. Delta time of first and last card punched during half cycle.

3. Input:

Use tape generated from outcard alpha check (A5).

4. Output:

Generates an off line listing from tape (A3) to all outcard cycles.

5. Operating procedures:

Attach a job card to front of binary deck.

6. Special considerations:

a. From the off-line listing that was generated it is possible to analyze any out-of-limit conditions. Also to determine if any cycles are invalid and should be deleted.

b. An out-of-limit condition may actually be valid data. Cycles of this nature will be recorded according to the condition, coded and used later in the BCD to binary conversion program.

7. Sample deck setup:

Same as for data sort (Figure A-3).

8. Example of on-line listing preliminary analysis for outcards – D-1:

```
*   Preliminary analysis for outcards

*   Pause

*   XEQ

    Execution
    927 Lines output this job.
```

Also tape A-3 lists all cycles out of limits.

Step E-1

1. Title: Outcard delete and duplication record elimination program.

2. General description:

This program deletes any duplicate records which may have accidently been loaded on the original magnetic tape, and deletes any invalid records located by the preliminary analysis routine.

3. Input:

a. Use the outcard tape generated from outcard alpha check (A5).

b. Any cycles to be deleted will be punched on IBM cards and placed directly behind a "*data card." This is loaded directly behind the source program and the last card has a "99" punched in columns 5 & 6 to signal end of data. This card has to be loaded directly behind a "data card" even if no cycles are to be deleted. Below is example of card format.

Column	Data	Description
1	X	0 Complete cycle 1 Half cycle
5 & 6	XX	Battery number
8 - 11	XXXX	Cycle number
14	X	Phase if needed
24	0	Signifies outcards

4. Output:

a. Generates output tape (A6)

b. Off-line listing from tape (A3) will list any duplicate records that were found and any records which were deleted.

5. Operating procedure:

Attach a job card and computer operations tape log card to the front of the binary deck. Same type as made out for endcards.

6. Sample deck setup:

This will be the same as used for endcards, (Figure A-4, Page A-16 for example of deck setup), (Page A-15 for explanation of this setup).

7. Examples of on-line listing outcard delete and dup record elimination – E-1:

```
*    Outcard delete and dup record elimination

*    Pause

*    XEQ

     Execution

     12539  Lines output this job.
```

Step F-1

1. Title: Outcard BCD to binary program

2. General description:

This program generates a binary tape with batteries 0-11 on it. It records on this tape the format as shown on Page 41 for outcards. This program makes some final validity checks before writing the cycles up on binary tape. If the cycles do not meet the requirements, it will not write this cycle up on tape. Any cycles which are not accepted are listed on the output tape (A3) with the battery and the cycle number, also the reason for not being accepted. Below are some of the checks made.

a. Checks to see if the card is an outcard by sequence number.

b. Checks to see if there are 5 or more cards per half cycle.

This program puts codes on all cycles for which a code card was made. Also uses the endcard code tape and places the last bad endcard cycle number in code "C" on every outcard cycle until the next bad endcard cycle number is equal to or greater than the outcard cycle number and then this number is used in code "C".

3. Input:

a. This program uses (3) input tapes.

i. Output tape of outcard delete program (B5).
ii. Outcard tape with codes (A7).
iii. Endcard tape with codes (A6).

b. Outcard codes which were made up from listing of outcard preliminary analysis program are written up on separate tape. Format for punched cards is as follows:

i.

Column	Data	Description
5 & 6	XX	Battery number
9 - 12	XXXX	Cycle number
14 & 15	XX	Code number
16 - 18	XXX	Time if any
24	0	Signifies outcard

ii. These cards have to be in order by battery and cycle number before going on tape.

iii. Last card has "99" punched in columns 5 & 6 to signal last card.

4. Output:

   a. Generates binary output tape (A8) containing batteries 0-11.

   b. Generates off-line listings from tape (A3) of all outcard data that is on this update run. Request (2) listings of this tape.

5. Operating procedures:

   Attach a job card and computer operations tape log card to the front of the binary deck.

6. Special considerations:

   a. Before running this program, a tape with endcard codes must be ready.
   b. Include the last cycle coded from each battery from the last update run.

7. Sample deck setup

   Same as for the data sort (Figure A-3).

8. Example of on-line listing outcard BCD to binary – F-1:

```
*    Outcard BCD to binary

*    Pause

*    XEQ

     Execution

     12691  Lines output this job.
```

Step G-1

1. Title: Outcard binary merge

2. General description:

This program will merge all the new data with the old. It generates (2) binary tapes, one with batteries 0-7, and the other with batteries 8-11.

3. Input:

This program uses (3) input tapes.

a. Output tape of outcard BCD to binary (A5)
b. Old binary outcard tape containing batteries 9-7 (B6)
c. Old binary outcard tape containing batteries 8-11 (B7)

4. Output:

Generates (2) output tapes.

a. Updated binary outcard tape with batteries 0-7 (B6)
b. Updated binary outcard tape with batteries 8-11 (B7)

5. Operating procedures:

Attach a job card and (2) computer operations tape cards to the front of the binary deck.

6. Special considerations:

Duplicate the (2) binary tapes for spares, in case anything happens to the originals.

7. Sample deck setup:

Same as data sort (Figure A-3).

8. Example of on-line listing outcard binary merge - G-1:

```
*   Outcard binary merge

*   Pause

*   XEQ

    Execution

    End of job

    13 Lines output this job.

*   End tape
```

Step D-2

1. Title: Endcard #1 preliminary analysis program

2. General description:

This program summarizes any out of limit condition on any card image on the input tape. The following are the fields it will check and if they are not in limits of the parameters set up, they will be listed on tape A3 for off-line listings.

a. Charge time
b. Discharge time
c. Energy efficiency
d. Current efficiency
e. Watt hours charge
f. Watt hours discharge

Should it be necessary to modify the out of limit parameters, it can be easily accomplished by modification of the source program constants. See example with source program.

3. Input:

Use tape generated from alpha check (A5).

4. Output:

Does not generate any output tape, just an off-line listing from tape A3. See attached example of this listing.

5. Operating procedures:

Attach a job card to the front of the binary deck.

6. Special considerations:

a. Do not use endcard #2 tape with this program.

b. From the A3 listing, it is possible to analyze the out of limit conditions, to determine the validity of the information. Any invalid cycles will be recorded and are to be used in the delete program.

c. An out of limit condition may actually be valid data. Cycles of this nature will be recorded according to the condition, and used later in the BCD to binary conversion program.

7 Sample deck setup:

Same as the data sort (Figure A-3).

8. Example of on-line listing for endcard preliminary analysis – D-2:

```
*    Preliminary analysis for endcards 1

*    Pause

*    XEQ

     Begin compilation

     Execution

     1382 Lines output this job.
```

Step E-2

1. Title: Endcard #1 delete and duplication record elimination

2. General description:

This program deletes any duplicate records which may have accidently been loaded on the original magnetic tape. Also deletes any invalid records located by the preliminary analysis routine.

3. Input:

a. Use the endcard #1 generated from the alpha sort (A5).

b. Any cycles to be deleted will be punched on IBM cards and placed behind a "*data card." The last card has a 99 in columns 5 and 6 to signal the end of the data. This card has to be the last card even if no cycles are to be deleted. Below is example of card format.

Column	Data	Description
1	0	Identification
5 & 6	XX	Battery No.
8 - 11	XXXX	Cycle No.
24	E	Signifies endcard

4. Output:

Generates output tape (A-6). Examples of print out are attached.

5. Operating procedure:

Attach a job card and computer operations tape log card to the front of the binary deck.

6. Special considerations:

Do not use endcard #2 tape with this program.

7. Sample deck setup (Figure A-4):

a. Identification card. An asterisk punched in column 1 and a program description punched in columns 7-72.

b. Pause card. An asterisk punched in column 1 and PAUSE punched in columns 7-11.

c. Execute card. An asterisk punched in column 1 and XEQ punched in columns 7-9.

d. Binary deck.

e. Data card. An asterisk punched in column 1 and DATA punched in columns 7-10.

f. Delete cards.

g. Last card with 99 punched in columns 5 and 6.

8. Example of on-line listing for endcard delete and dup record elimination - E-2:

```
*    Endcard 1 delete and dup record elimination

*    Pause

*    XEQ

     Execution

     10732 Lines output this job.
```

Step F-2

1. Title: Endcard BCD to binary merge

2. General description:

This program merges endcard #1 and endcard #2 together and generates a binary tape with batteries 0-11. It records on tape the format as shown on Page 28 for endcards. This also makes some final validity checks before merging endcard #1 and endcard #2. If the cycle does not meet all the parameters shown below it will not write this cycle on tape. Any cycles which are not accepted are listed on A3 output by battery and cycle number and the reason not accepted.

a. Checks to see if card is an endcard.
b. Checks for (2) cards per half cycle.

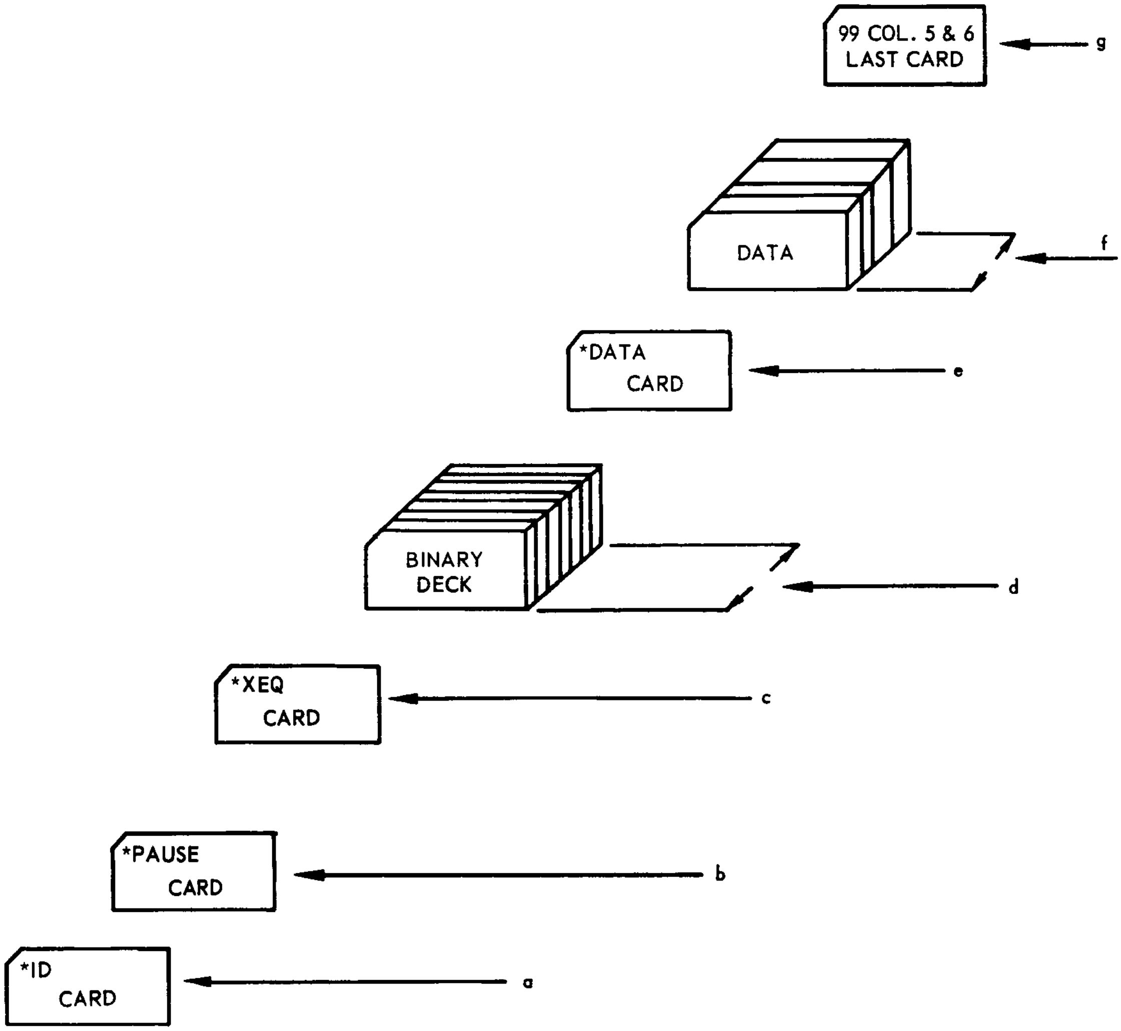


Figure 5.1–Sample deck setup — delete program endcard #1

c. Checks for (4) cards per cycle and each (4) cards, checks for following:

1. Same battery number.
2. Same cycle number.
3. Cell activity agrees on charge and discharge.

d. Deletes cycle if there is zero efficiency on endcard #1

3. Input:

a. This program uses (3) input tapes.

i. Endcard #1 tape from output of delete and duplication record elimination (A8).

ii. Endcard #2 alpha check tape (B5).

iii. Endcard tape with codes (A6).

b. Endcard codes which were made up from listing of endcard analysis program are written up on separate tape. Format for punched cards to use on code is as follows:

i.

Column	Data	Description
5 & 6	XX	Battery number
9 - 12	XXXX	Cycle number
14 & 15	XX	Code number
16 - 18	XXX	Time if any
24	E	Signifies endcard

ii. These cards have to be in order by battery and cycle number.

iii. Last card has 99 punched in columns 5 and 6 to signal last card.

4. Output:

a. Generates binary output tape (A7) containing batteries 0-11.

b. Generates off-line listing from tape (A3) of all endcard data that is up on tape. Example of print out are attached. Request (2) listings of this tape.

c. Codes all endcard cycles that are requested.

5. Operating procedures:

Attach a job card and computer operations tape log card to the front of the binary deck.

6. Sample deck setup:

Same as for the data sort (Figure A-3).

7. Example of on-line listing for endcard BCD to binary - F-2:

```
*    Endcard BCD to binary

*    Pause

*    XEQ

     EXECUTION
```

Total number of cycles processed – 5202.

End of tape writing A3 . Change and push start.

31986 Lines output this job.

Step G-2

1. Title: Endcard binary merge program

2. General description:

This program merges all the new data with the old data. It generates (2) binary tapes, one with batteries 0-7, and the other binary tape with batteries 8-11.

3. Input:

This program uses (3) input tapes.

a. Output tape of endcard BCD to binary (B8).
b. Binard endcard tape containing old data from batteries 0-7 (A6).
c. Binary endcard tape containing old data from batteries 8-11 (A7).

4. Output:

Generates (2) output tapes:

a. Updated binary endcard tape with batteries 0-7 (B6).
b. Updated binary endcard tape with batteries 8-11 (B7).

5. Operation procedures:

Attach a job card and (2) computer operations tape log cards to the front of the binary deck.

6. Sample deck setup:

Same as for the data sort (Figure A-3).

7. Special considerations:

Duplicate the (2) binary tapes for spares, in case anything happens to the originals.

8. Example of on-line listing for endcard binary merge – G-2:

```
*   Endcard binary merge

*   Pause

*   XEQ
    Execution

End of job

13 Lines output this job.
```

APPENDIX B

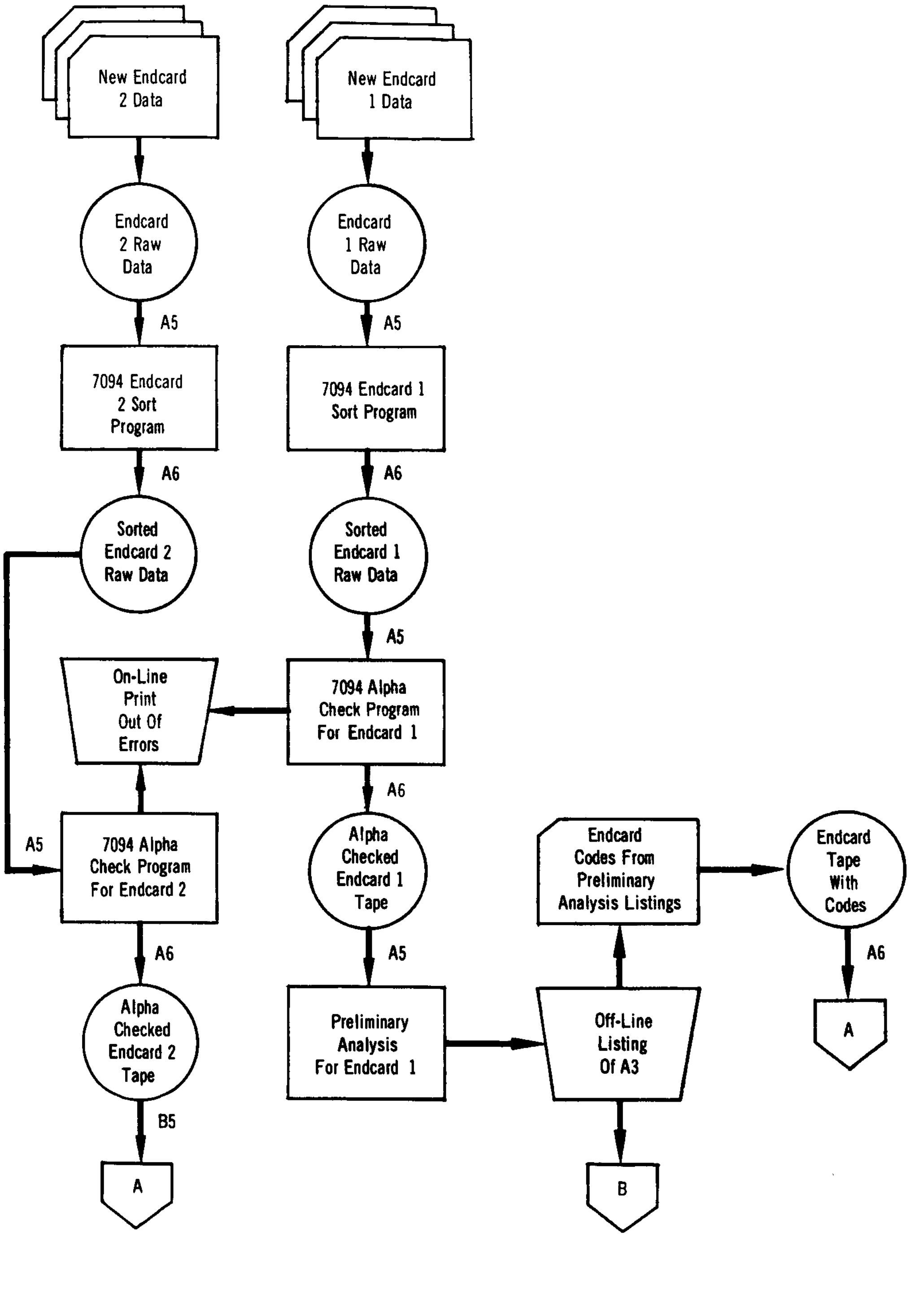
FLOW CHARTS

and

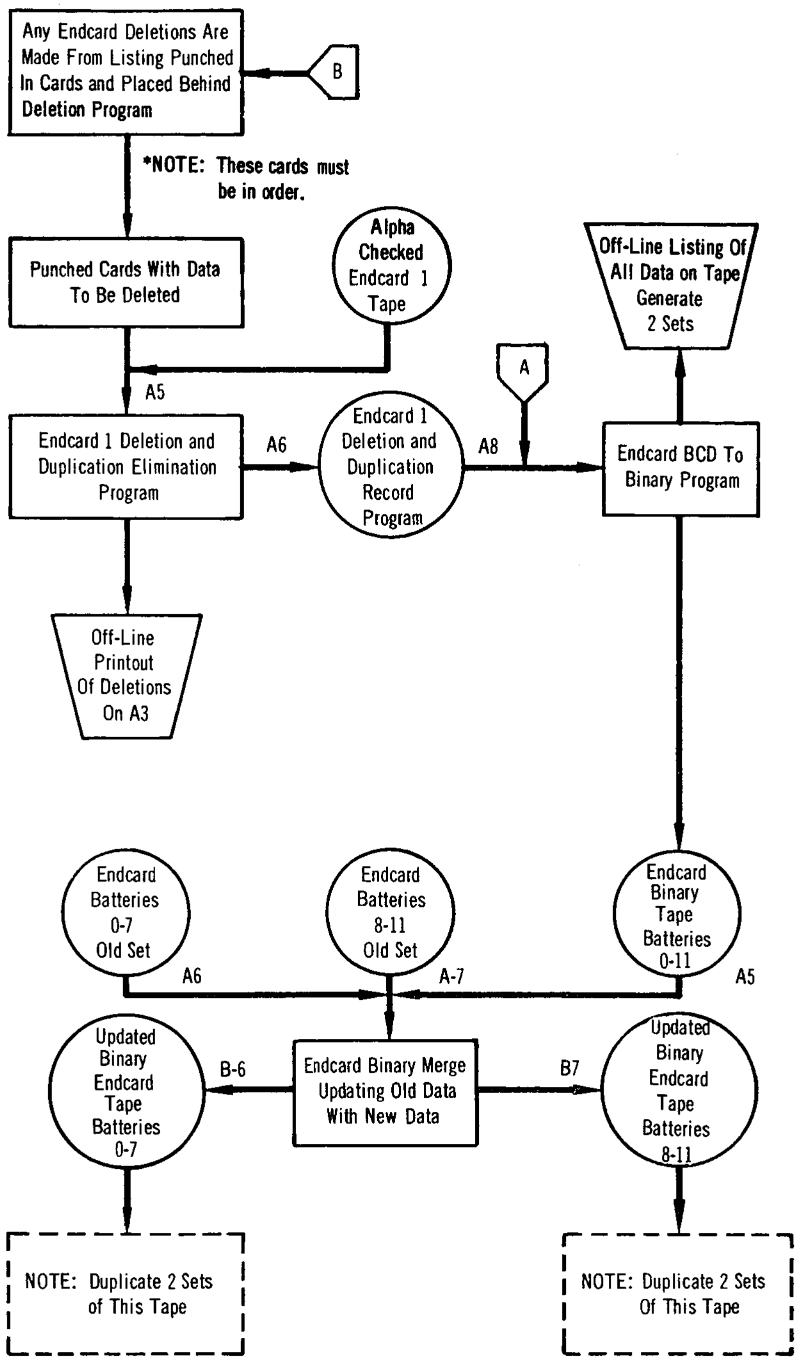
SOURCE DECK STATEMENTS

FOR CARD TO TAPE EDITING PROGRAM

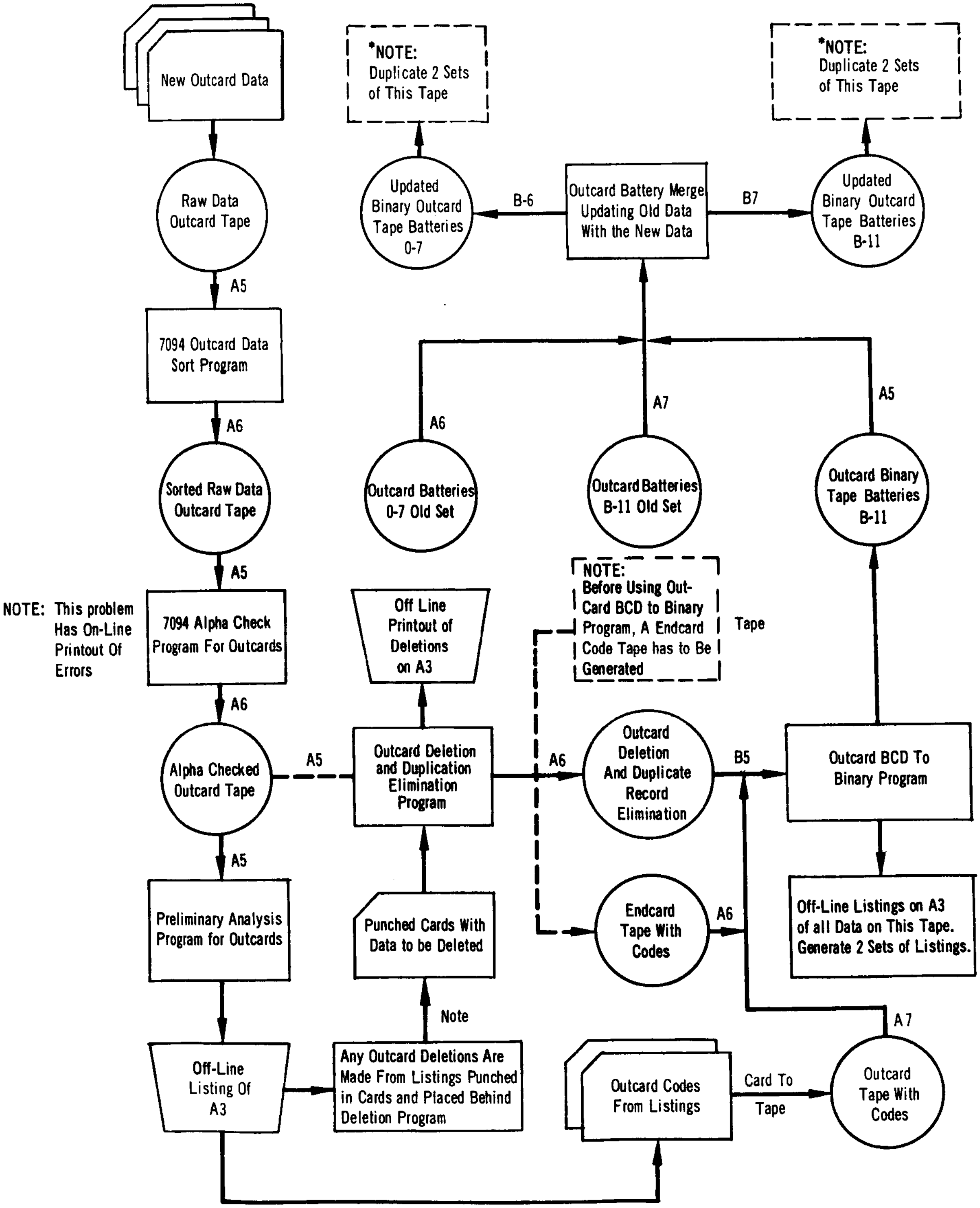
NEW ENDCARD DATA MERGE


New Endcard 2 Data
New Endcard 1 Data
Endcard 2 Raw Data
Endcard 1 Raw Data
A5
A5
7094 Endcard 2 Sort Program
7094 Endcard 1 Sort Program
A6
A6
Sorted Endcard 2 Raw Data
Sorted Endcard 1 Raw Data
A5
On-Line Print Out Of Errors
7094 Alpha Check Program For Endcard 1
A6
A5
7094 Alpha Check Program For Endcard 2
Alpha Checked Endcard 1 Tape
Endcard Codes From Preliminary Analysis Listings
Endcard Tape With Codes
A6
A5
A6
Alpha Checked Endcard 2 Tape
Preliminary Analysis For Endcard 1
Off-Line Listing Of A3
A
B5
A
B

NEW ENDCARD DATA MERGE


Any Endcard Deletions Are Made From Listing Punched In Cards and Placed Behind Deletion Program
B
*NOTE: These cards must be in order.
Punched Cards With Data To Be Deleted
Alpha Checked Endcard 1 Tape
Off-Line Listing Of All Data on Tape Generate 2 Sets
A
A5
Endcard 1 Deletion and Duplication Elimination Program
A6
Endcard 1 Deletion and Duplication Record Program
A8
Endcard BCD To Binary Program
Off-Line Printout Of Deletions On A3
Endcard Batteries 0-7 Old Set
Endcard Batteries 8-11 Old Set
Endcard Binary Tape Batteries 0-11
A6
A-7
A5
Updated Binary Endcard Tape Batteries 0-7
B-6
Endcard Binary Merge Updating Old Data With New Data
B7
Updated Binary Endcard Tape Batteries 8-11
NOTE: Duplicate 2 Sets of This Tape
NOTE: Duplicate 2 Sets Of This Tape

NEW OUTCARD DATA MERGE


New Outcard Data
Raw Data Outcard Tape
A5
7094 Outcard Data Sort Program
A6
Sorted Raw Data Outcard Tape
A5
NOTE: This problem Has On-Line Printout Of Errors
7094 Alpha Check Program For Outcards
A6
Alpha Checked Outcard Tape
A5
A5
Preliminary Analysis Program for Outcards
Off-Line Listing Of A3
Any Outcard Deletions Are Made From Listings Punched in Cards and Placed Behind Deletion Program
Note
Punched Cards With Data to be Deleted
Outcard Deletion and Duplication Elimination Program
Off Line Printout of Deletions on A3
*NOTE: Duplicate 2 Sets of This Tape
Updated Binary Outcard Tape Batteries 0-7
B-6
Outcard Battery Merge Updating Old Data With the New Data
B7
Updated Binary Outcard Tape Batteries B-11
*NOTE: Duplicate 2 Sets of This Tape
A6
Outcard Batteries 0-7 Old Set
A7
Outcard Batteries B-11 Old Set
A5
Outcard Binary Tape Batteries B-11
NOTE: Before Using Out-Card BCD to Binary Program, A Endcard Code Tape has to Be Generated
Tape
A6
Outcard Deletion And Duplicate Record Elimination
B5
Outcard BCD To Binary Program
Off-Line Listings on A3 of all Data on This Tape. Generate 2 Sets of Listings.
Endcard Tape With Codes
A6
A7
Outcard Codes From Listings
Card To Tape
Outcard Tape With Codes

```
$IBSYS
$ID                                                         STEP B-1
$PAUSE
$EXECUTE       SORT
      FILE,INPUT/1,BLO/14,REE/1,MOD/D,DENSITY/H
      FILE,OUT,BLO/14,MOD/D,DEN/H
      RECORD,TYPE/F,LEN/(14),FIELD/(6,10,5,5,1)
      SORT,FILE/1,ORDER/2,FIE/(1A,5D,3A),SEQ/S
      CHANNELS,INP/A,MERGE/(A,B)
      OPT,TAP
      REMARK OUTCARD DATA SORT
      END
$FMSYS
```

```
*    OUTCARD ALPHA TEST                 STEP C-1
*     PAUSE
*     XEQ
*     CARDS COLUMN
*     FORMAP
*     FAP
 INIT  ETTA
       NOP
       AXT     14,1
       STZ     WKAR+14,1
       TIX     *-1,1,1
 READ  RTDA    5
       RCHA    COMM1
       TCOA    *
       TRCA    RDERR
       TEFA    END5
       TRA     VALCK
 RDERR AXT     9,2
 RDBSP BSRA    5
       RTDA    5
       RCHA    COMM1
       TCOA    *
       TRCA    *+2
       TRA     RDERR-2
       TIX     RDBSP,2,1
       TSX     $(SPH),4
       PZE     MESSA,,1
       TSX     $(FIL),4
       CALL    $EXIT
 VALCK AXT     14,1
       AXT     6,2
       CAL     WKAR+14,1
       SLW     TEMP
 CK1   ANA     MASK
       LAS     SIXTY
       TRA     ERROR
       TRA     CK2
```

```
        LAS     NINE
        TRA     ERROR
        TRA     CK2
CK2     CAL     TEMP
        ARS     6
        SLW     TEMP
        TIX     CK1,2,1
        TIX     VALCK+1,1,1
        TRA     WRITE
ERROR   CAL     WKAR
        SLW     MESSG+2
        CAL     TEMP
        SLW     MESSG+5
        TSX     $(SPH),4
        PZE     MESSG,,1
        TSX     $(FIL),4
        TSX     $(SPH),4
        PZE     MESSB,,1
        TSX     $(FIL),4
        TRA     INIT
WRITE   WTDA    6
        RCHA    COMM1
        TCOA    *
        ETTA
        TRA     END6
        TRCA    WIERR
        TRA     INIT
END5    LXA     SAM,1
        TIX     *+2,1,1
        TRA     LAST
        SXA     SAM,1
        RUNA    5
        TSX     $(SPH),4
        PZE     MESSD,,1
        TSX     $(FIL),4
        HTR     READ
LAST    RUNA    5
        TSX     $(SPH),4
        PZE     MESSE,,1
        TSX     $(FIL),4
        AXT     1,1
        SXA     SAM,1
END6    CAL     CODE
        SLW     WKAR
        WTDA    6
        RCHA    COMM1
        TCOA    *
        WEFA    6
        RUNA    6
        CAL     WKAR
        SLW     MESSG+2
        TSX     $(SPH),4
        PZE     MESSG,,1
        TSX     $(FIL),4
        TSX     $(SPH),4
        PZE     MESSF,,1
        TSX     $(FIL),4
        HTR     INIT
WIERR   AXT     9,2
WIBSP   BSRA    6
        WTDA    6
        RCHA    COMM1
```

```
       TCOA    *
       TRCA    *+2
       TRA     INIT
       BSRA    6
       WTDA    6
       TCOA    *
       TRCA    WZ
       TIX     WIBSP+1,2,1
WZ     TSX     $(SPH),4
       PZE     MESSC,,1
       TSX     $(FIL),4
       CALL    $EXIT
 MESSA BCI     7,(35H READ ERROR ENCOUNTERED, CALL EXIT.)
 MESSB BCI     7,(35H INVALID CHARACTER IN RECORD ABOVE.)
 MESSC BCI     7,(36H WRITE ERROR ENCOUNTERED, CALL EXIT.)
 MESSD BCI     7,(29H PREPARE NEW A5. PRESS START.)
 MESSE BCI     7,(29H LAST A5.PEOCESSING COMPLETE.)
 MESSF BCI     7,(37H PREPARE A6 IF PROCESSING INCOMPLETE.)
 MESSG BCI     7,(35H                                   )
 WKAR  BSS     14
 SIXTY OCT     60
 CODE  OCT     111111111111
 MASK  OCT     77
 NINE  OCT     11
 TEMP  PZE
 SAM   PZE     1
 COMM1 IOCD    WKAR,,14
       END


*     PRELIMINARY ANALYSIS FOR OUTCARDS                STEP D-1
*
*     PAUSE
*     XEQ
*     CARDS COLUMN
C****                   OUTCARD VALIDITY ANALYSIS
C**** INITIALIZATION
      DIMENSION V(10)
    1 ISW=1
      NSTNO=1
      ISTNO=1
      L=5
C**** FIRST INPUT TAPE DRIVE L
      REWIND L
C**** HEADINGS
    2 WRITE OUTPUT TAPE 3,100
      WRITE OUTPUT TAPE 3,99
      LINE=3
      GO TO (3,10,14),NSTNO
C**** FORMAT STATEMENTS
   99 FORMAT (116H0     BAT    CYCLE    DAY    Y    SEQ    NO CARDS    V
     1ULT-HOURS    START-TIM   END-TIM   START-ET    END-ET    PHASE)
  100 FORMAT (126H1   P  R  E  L  I  M  I  N  A  R  Y     O  U  T  C  A
     1  R  D      S  U  M  M  A  R  Y     A  N  D     A  N  A  L  Y  S
     2  I  S)
  101 FORMAT (I2,I4,I1,I3,6X,F5.4,5X,I1,9X,10F4.3,I1)
  102 FORMAT (1H )
  103 FORMAT (1H ,3I8,I5,I6,I10,F15.4,F13.4,F11.4,F10.3,F11.3,6X,A3)
  104 FORMAT (12H0 END OF JOB)
C**** READ RECORD
    3 READ INPUT TAPE L,101,NBAT,NCYC,NY,NDAY,TIME,NPHAS,V(1),V(2),V(3),
     1V(4),V(5),V(6),V(7),V(8),V(9),V(10),NSEQ
```

```
      GO TO (15,4),ISTNO
    4 IF (NBAT-99)5,21,21
    5 IF (NPHAS-IPHAS)8,6,8
    6 IF (NCYC-ICYC)8,7,8
    7 IF (NBAT-IBAT)8,15,8
    8 IF (LINE-55)10,9,9
    9 NSTNO=2
      GO TO 2
   10 IF (IPHAS)50,50,60
   50 FPHAS=3HCHG
      GO TO 70
   60 FPHAS=3HDIS
   70 WRITE OUTPUT TAPE 3,103,IBAT,ICYC,IDAY,IY,ISEQ,NOCD,VH,TINT,TAXT,
     1TINE,TAXE,FPHAS
      LINE=LINE+1
      IF (NBAT-IBAT)11,12,11
   11 NSTNO=3
      GO TO 2
   12 IF (NCYC-ICYC)13,14,13
   13 WRITE OUTPUT TAPE 3,102
      LINE=LINE+1
   14 ISW=1
   15 ATCEL=0.0
      AVER=0.0
      DO 17 I=1,10
      IF (V(I))16,17,16
   16 ATCEL=ATCEL+1.0
      AVER=AVER+V(I)
   17 CONTINUE
      AVER=AVER/ATCEL
C**** FIRST CARD OF PHASE OR HALF CYCLE
      IF (ISW)19,19,18
   18 TINT=TIME
      TINE=AVER
      ISEQ=NSEQ
      IDAY=NDAY
      IBAT=NBAT
      ICYC=NCYC
      IY=NY
      NOCD=0
      VH=0.0
      IPHAS=NPHAS
      ISW=0
      GO TO 20
C**** REMAINDING CARDS OF PHASE OR CYCLE
   19 DELTI=TIME-TAXT
      VH=VH+DELTI*((TAXE+AVER)/2.0)
   20 TAXT=TIME
      TAXE=AVER
      NOCD=NOCD+1
      ISTNO=2
      GO TO 3
C**** END OF TAPE ROUTINE
   21 REWIND L
      WRITE OUTPUT TAPE 3,104
      CALL EXIT
      END


*     OUTCARD DELETE AND DUP RECORD ELIMINATION     STEP E-1
*     PAUSE
*     CARDS COLUMN
```

```
C..... OUTCARD  DELETE AND DUPLICATE RECORD ELIMINATION PROGRAM
C.... BELOW IS FORMAT EXPLANATION FOR RECORDS TO BE DELETED
C *1* COL 1 CONTAINS A 0 IF ENTIRE CYCLE IS TO BE DELETED
C *2* COL 1 CONTAINS A 1 IF PHASE OF CYCLE IS TO BE DELETED ONLY (OUTCARDS
C *3* COL 5 AND 6 CONTAINS BATTERY NUMBER
C *4* COL 8 TO 11 CONTAINS CYCLE NUMBER
C *5* COL 14 CONTAINS PHASE (IF COL 1 CONTAINS 1),0=CHG,1=DISCHARGE
C *6* LAST CARD SIGNAL CONTAINS 99 IN COL 5 AND 6 (THIS ENDS ALL DELETIONS)
C *7* THIS DATA WILL BE PLACED BEHIND DELETE PROGRAM
C.... INITIALIZATION
      DIMENSION NN(21),MM(21)
    2 READ INPUT TAPE 2,100,ID,IBAT,ICYC,IPHAS
    1 REWIND 5
      REWIND 6
      WRITE OUTPUT TAPE 3,103
      ISW=0
C.... FORMAT STATEMENTS
  100 FORMAT (I1,I5,I5,I3)
  101 FORMAT (I2,I4,I1,I3,I5,1X,2I5,I1,I5,12I4)
  102 FORMAT (27H0DUPLICATE RECORD DELETED  ,I2,I4,I1,I3,I5,1X,2I5,I1,I5
     1,12I4)
  103 FORMAT (1H1,45X,32HD E L E T I O N    P R O G R A M)
  104 FORMAT (13H0  END OF JOB)
  105 FORMAT (27H0REQUESTED RECORD DELETED  ,I2,I4,I1,I3,I5,1X,2I5,I1,I5
     1,12I4)
  106 FORMAT (36H0REQUEST FOR DELETION OUT OF ORDER  ,4I6)
  107 FORMAT (47H0COULD NOT LOCATE RECORD FOR DELETION REQUEST  ,2I6)
    3 READ INPUT TAPE 5,101,(NN(I),I=1,21)
    4 READ INPUT TAPE 5,101,(MM(I),I=1,21)
      DO 5 I=1,21
      IF (NN(I)-MM(I))9,5,9
    5 CONTINUE
      WRITE OUTPUT TAPE 3,102,(NN(I),I=1,21)
    6 DO 7 I=1,21
      NN(I)=MM(I)
    7 CONTINUE
      IF (MM(1)-99)4,8,8
    8 WRITE OUTPUT TAPE 6,101,(MM(I),I=1,21)
      WRITE OUTPUT TAPE 3,104
      GO TO 22
    9 IF (IBAT-99)11,10,10
   10 WRITE OUTPUT TAPE 6,101,(NN(I),I=1,21)
      GO TO 6
   11 IF (IBAT-NN(1))16,12,10
   12 IF (ICYC-NN(2))16,13,10
   13 IF (ID)15,15,14
   14 IF (IPHAS-NN(8))10,15,10
   15 WRITE OUTPUT TAPE 3,105,(NN(I),I=1,21)
      ISW=1
      GO TO 6
   16 IF (ISW)21,21,17
   17 READ INPUT TAPE 2,100,ID,IIBAT,IICYC,IPHAS
      ISW=0
      IF (IBAT-IIBAT)19,18,20
   18 IF (ICYC-IICYC)19,20,20
   19 IBAT=IIBAT
      ICYC=IICYC
      GO TO 9
   20 WRITE OUTPUT TAPE 3,106,IBAT,IIBAT,ICYC,IICYC
      GO TO 22
   21 WRITE OUTPUT TAPE 3,107,IBAT,ICYC
      GO TO 17
   22 REWIND 5
      END FILE 6
      REWIND 6
      CALL EXIT
      END
*     DATA


*     OUTCARD BCD TO BINARY                               STEP F-1
*     PAUSE
*     XEQ
*     PAUSE
*     CARDS COLUMN
C        OUTCARD VALIDITY AND BINARY CONVERSION PROGRAM
C       TAPE NO  3 IS FOR ERROR MESSAGES
C       TAPE NO  6 IS FOR ENDCARD CODES
C       TAPE NO  7 IS FOR OUTCARD CODES
C       TAPE NO  8 IS FOR OUTCARD OUTPUT
C       TAPE NO 15 IF FOR OUTCARD INPUT
      DIMENSION IMOD(10),FMOC(16,350),FMOD(16,550),IA(21)
      REWIND 8
      NUBAT=0
      REWIND 6
      REWIND 7
      REWIND 15
      REWIND 6
      IND=0
      NOCHG=0
      NODIS=0
      ICNT=0
C....    TAPE A6 CONTAINS ENDCARD CODES
    2 READ INPUT TAPE 6,101,KEDBAT,KEDCYC
  101 FORMAT (I6,I6)
      READ INPUT TAPE 7,102,KODBAT,KODCYC,KOD,KODTIM
  102 FORMAT (I6,I6,I3,I3)
C....    TAPE B15 CONTAINS OUTCARD TAPE
    6 READ INPUT TAPE 15,103,(IA(I),I=1,21)
  103 FORMAT (I2,I4,I1,I3,I5,1X,2I5,I1,I5,11I4,I1)
C....    DETERMINE IF THIS IS AN OUTCARD
      IF (IA(8)-1)3,3,4
    3 IF (IA(21)-4)5,5,4
C....    ILLEGAL OUTCARD ROUTINE
    4 WRITE OUTPUT TAPE 3,104
  104 FORMAT (47HOCARD BELOW CONTAINS ILLEGAL PHASE OR SEQUENCE.)
      WRITE OUTPUT TAPE 3,105,(IA(I),I=1,21)
  105 FORMAT (1H ,I4,I6,I3,I5,I8,2I7,I3,I7,11I6,I3)
      GO TO 6
    5 IMOD(9)=0
    7 NUBAT=IA(1)
      NUCYC=IA(2)
C....    DETERMINE IF CYCLE SHOULD HAVE OUTCARD CODE
    8 IF (KODBAT-99)9,11,11
    9 IF (KODBAT-IA(1))13,10,11
   10 IF (KODCYC-IA(2))13,12,11
   12 IMOD(7)=KOD
      IMOD(8)=KODTIM
      READ INPUT TAPE 7,102,KODBAT,KODCYC,KOD,KODTIM
      GO TO 14
   11 IMOD(7)=0
      IMOD(8)=0
      GO TO 14
   13 WRITE OUTPUT TAPE 3,106
  106 FORMAT (44HCOUTCARD DATA DOES NOT EXIST FOR CODE BELOW.)
      WRITE OUTPUT TAPE 3,102,KODBAT,KODCYC,KOD,KODTIM
      READ INPUT TAPE 7,102,KODBAT,KODCYC,KOD,KODTIM
      GO TO 8
C....   FILL IMOD ARRAY WITH BAT,CYC,YEAR,DAY AND SEQ
   14 IMOD(1)=0
      DO 15 I=2,5
      IMOD(I)=IA(I-1)
   15 CONTINUE
      IMOD(6)=IA(21)
C....   CHECK TO SEE IF OUTCARD SHOULD HAVE LAST PREVIOUS BAD ENDCARD
C....     CYCLE NUMBER CHANGED OR WROTE IN AT THIS TIME
   16 IF (KEDBAT-99)17,21,21
   17 IF (KEDBAT-IA(1))19,18,21
   18 IF (KEDCYC-IA(2))20,20,21
   19 READ INPUT TAPE 6,101,KEDBAT,KEDCYC
      GO TO 16
   20 IMOD(9)=KEDCYC
      READ INPUT TAPE 6,101,KEDBAT,KEDCYC
      GO TO 16
   21 IF (IA(8))22,22,23
C....   PACKING ARRAY FOR CHARGE HALF
   22 NOCHG=NOCHG+1
      FMOC(1,NOCHG)=IA(8)
      FMOC(2,NOCHG)=IA(5)
      FMOC(2,NOCHG)=FMOC(2,NOCHG)*.001
      FMOC(3,NOCHG)=IA(6)
      FMOC(3,NOCHG)=FMOC(3,NOCHG  .0001
      FMOC(4,NOCHG)=IA(7)
      FMOC(4,NOCHG)=FMOC(4,NOCHG)*.001
      FMOC(5,NOCHG)=IA(9)
      FMOC(5,NOCHG)=FMOC(5,NOCHG)*.001
      FMOC(6,NOCHG)=IA(10)
      FMOC(6,NOCHG)=FMOC(6,NOCHG)*.01
      DO 24 K=7,16
      FMOC(K,NOCHG)=IA(K+4)
      FMOC(K,NOCHG)=FMOC(K,NOCHG)*.001
   24 CONTINUE
      GO TO 25
C....   PACKING ARRAY FOR DISCHARGE HALF
   23 NODIS=NODIS+1
      FMOD(1,NODIS)=IA(8)
      FMOD(2,NODIS)=IA(5)
      FMOD(2,NODIS)=FMOD(2,NODIS)*.001
      FMOD(3,NODIS)=IA(6)
      FMOD(3,NODIS)=FMOD(3,NODIS)*.0001
      FMOD(4,NODIS)=IA(7)
      FMOD(4,NODIS)=FMOD(4,NODIS)*.001
      FMOD(5,NODIS)=IA(9)
      FMOD(5,NODIS)=FMOD(5,NODIS)*.001
      FMOD(6,NODIS)=IA(10)
      FMOD(6,NODIS)=FMOD(6,NODIS)*.01
      DO 26 K=7,16
      FMOD(K,NODIS)=IA(K+4)
      FMOD(K,NODIS)=FMOD(K,NODIS)*.001
   26 CONTINUE
C....   READING NEW OUTCARD FROM TAPE
   25 READ INPUT TAPE 15,103,(IA(I),I=1,21)
      IF (IA(1)-99)27,41,41
C....    CHECK TO SEE IF CARD READ IN IS A OUTCARD
   27 IF (IA(8)-1)28,28,29
   28 IF (IA(21)-4)30,30,29
```

```
C...      ILLEGAL OUTCARD ROUTINE
   29 WRITE OUTPUT TAPE 3,104
      WRITE OUTPUT TAPE 3,105,(IA(I),I=1,21)
      GO TO 25
C....      CHECK TO SEE IF A NEW BATTERY NUMBER IS ON OUTCARD
   30 IF (IA(1)-NUBAT)33,31,33
   31 IF (IA(2)-NUCYC)33,32,33
   32 IF (IA(8))22,22,23
C....     NEW BATNO HAS BEEN READ IN, START VALIDITY CHECK
   33 IF (NOCHG)60,61,60
   60 IF (NOCHG-5)38,66,66
   61 IF (IMUD(7)-21)38,66,62
   62 IF (IMOD(7)-23)38,66,63
   63 IF (IMOD(7)-26)38,66,38
   66 IF (NODIS)67,68,67
   67 IF (NODIS-5)38,39,39
   68 IF (IMOD(7)-21)38,39,69
   69 IF (IMUD(7)-23)38,39,70
   70 IF (IMUD(7)-26)38,39,38
C....     ERROR ROUTINE FOR CYCLES WITH LESS THAN 5 CARDS
   38 WRITE OUTPUT TAPE 3,109
  109 FORMAT (52HOFOLLOWING CYCLE HAS LESS THAN 5 CARDS AND NOT CODED)
      WRITE OUTPUT TAPE 3,110,IMOD(2),IMOD(3)
  110 FORMAT (14HOBATTERY NO.  ,I2,12H CYCLE NO.  ,I4)
      GO TO 50
C....    TAPE 8 IS USED FOR OUTPUT TAPE
   39 ICNT=NOCHG+NODIS
      IF (NOCHG)76,76,77
   76 WRITE TAPE 8,IND,ICNT,(IMOD(I),I=1,9),((FMOD(I,J),I=1,16),J=1,NODI
     1S)
      GO TO 50
   77 IF (NODIS)78,78,75
   78 WRITE TAPE 8,IND,ICNT,(IMOD(I),I=1,9),((FMOC(I,J),I=1,16),J=1,NOCH
     1G)
      GO TO 50
   75 WRITE TAPE 8,IND,ICNT,(IMOD(I),I=1,9),((FMOC(I,J),I=1,16),J=1,NOCH
     1G),((FMOD(K,L),K=1,16),L=1,NODIS)
   50 ICNT=0
      NOCHG=0
      NODIS=0
      IF (IA(1)-NUBAT)42,43,42
   42 IMOD(9)=0
   43 IF (IND-1)7,44,44
C....     END OF JOB
   44 WRITE OUTPUT TAPE 3,111
  111 FORMAT (16HCEND OF PROGRAM.)
      REWIND 6
      REWIND 7
      REWIND 15
      END FILE 8
      REWIND 8
  300 FORMAT (132HOID BAT CYC Y DAY S  A  B    C  P   TIME    DEL-TI AVEP
     1WR AVECUR PRESS  V 0   V 1   V 2   V 3   V 4   V 5   V 6   V 7   V
     2 8  V 9    )
  301 FORMAT (1H ,I2,I3,I5,I2,I4,I2,I3,I4,I5,F3.0,F7.3,F8.4,2F7.3,F6.2,
     110F6.3)
      NUBAT=0
  201 WRITE OUTPUT TAPE 3,303
  303 FORMAT (1H1,40X,41HBINARY TAPE CONVERTED TO BCD FOR OUTCARDS)
  202 READ TAPE 8,IND,ICNT,(IMOD(I),I=1,9),((FMOD(I,J),I=1,16),J=1,ICNT)
      K=1
      IF (NUBAT-IMOD(2))203,204,203
  203 NUBAT=IMOD(2)
      WRITE OUTPUT TAPE 3,303
  204 IF (FMOD(1,1))205,205,208
  205 WRITE OUTPUT TAPE 3,300
      DO 207 I=1,ICNT
      K=I
      IF (FMOD(1,I))206,206,208
  206 WRITE OUTPUT TAPE 3,301,(IMOD(N),N=1,9),(FMOD(J,I),J=1,16)
  207 CONTINUE
      GO TO 212
  208 WRITE OUTPUT TAPE 3,300
      DO 210 I=K,ICNT
  209 WRITE OUTPUT TAPE 3,301,(IMOD(N),N=1,9),(FMOD(J,I),J=1,16)
  210 CONTINUE
  212 IF (IND)202,202,213
  213 REWIND 8
  214 CALL EXIT
C.... LAST RECORD OF LAST TAPE HAS BEEN ENCOUNTERED
   41 IND=1
      GO TO 33
      END


*     OUTCARD BINARY MERGE                              STEP G-1
*     PAUSE
*     XEQ
*     CARDS COLUMN
C.... PLACE BINARY TAPE TO BE MERGED ON A5
C.... PLACE OLD DATA TAPE FOR BAT 0 TO 7 ON A6
C.... PLACE OLD DATA TAPE FOR BAT 8 TO 11 ON A7
C.... PLACE BLANK TAPE ON B6 FOR MERGED BAT 0 TO 7
C.... PLACE BLANK TAPE ON B7 FOR MERGED BAT 8 TO 11
C.... INITIALIZATION
   10 ISW=0
      KSW=0
C.... II IS BINARY TAPE TO BE MERGED
      II=5
      REWIND II
C.... J1 IS BAT 0 TO 7 OLD DATA,J2 IS BAT 8 TO 11 OLD DATA
      J1=6
      J2=7
      REWIND J1
      REWIND J2
C.... M1 IS BAT 0 TO 7 MERGED NEW DATA,M2 IS BAT 8 TO 11 MERGED NEW DATA
      M1=16
      M2=17
      REWIND M1
      REWIND M2
      GO TO 11
C.... READ TAPE ROUTINES ***
C.... READ ROUTINE FOR TAPE TO BE MERGED
      DIMENSION IA(9),AA(16,400),IB(9),BB(16,400)
    1 READ TAPE II,INB,ICB,(IB(N),N=1,9),((BB(N,M),N=1,16),M=1,ICB)
      GO TO (12,25,33,45),L1
C
C.... READ ROUTINE FOR OLD DATA
    2 READ TAPE J1,INA,ICA,(IA(N),N=1,9),((AA(N,M),N=1,16),M=1,ICA)
      GO TO (13,20),L2
C
C.... WRITE TAPE ROUTINES ***
C.... WRITE ROUTINE FROM AA ARRAY
    3 WRITE TAPE M1,INA,ICA,(IA(N),N=1,9),((AA(N,M),N=1,16),M=1,ICA)
```

```
      GO TO (19,30,35,39,48),L3
C
C.... WRITE ROUTINE FROM BB ARRAY
    4 WRITE TAPE M1,INB,ICB,(IB(N),N=1,9),((BB(N,M),N=1,16),M=1,ICB)
      GO TO (23,31),L4
C
C.... MAINLINE PROGRAM
   11 L1=1
      GO TO 1
   12 L2=1
      GO TO 2
   13 IABAT=IA(2)
   14 IF (ISW)15,15,27
   15 IF (INA)16,16,36
   16 IF (KSW)17,17,18
   17 IF (IABAT-IA(2))21,18,49
   18 L3=1
      GO TO 3
   19 L2=2
      GO TO 2
   20 GO TO 14
C.... NEXT BATTERY ENCOUNTERED ON J1,CHECK FOR SAME BAT ON I1
   21 IF (IABAT-IB(2))13,22,50
   22 L4=1
      GO TO 4
   23 IF (INB)24,24,26
   24 L1=2
      GO TO 1
   25 GO TO 21
   26 KSW=1
      GO TO 13
C.... EXIT ROUTINE
   27 IF (INA)16,16,28
   28 IF (KSW)29,29,34
   29 INA=0
      L3=2
      GO TO 3
   30 L4=2
      GO TO 4
   31 IF (INB)32,32,35
   32 L1=3
      GO TO 1
   33 GO TO 30
   34 L3=3
      GO TO 3
   35 PRINT 100
  100 FORMAT (12H0 END OF JOB)
      END FILE M1
      REWIND M1
      REWIND I1
      REWIND J1
      CALL EXIT
C.... END BAT 7 ROUTINE
   36 IF (KSW)37,37,47
   37 IF (IB(2)-7)51,38,47
   38 INA=0
      L3=4
      GO TO 3
   39 INA=INB
      ICA=ICB
      DO 40 N=1,9
      IA(N)=IB(N)
   40 CONTINUE
      DO 42 N=1,ICB
      DO 41 M=1,16
      AA(M,N)=BB(M,N)
   41 CONTINUE
   42 CONTINUE
   43 IF (INB)44,44,46
   44 L1=4
      GO TO 1
   45 GO TO 37
   46 KSW=1
   47 INA=1
      L3=5
      GO TO 3
   48 ISW=1
      END FILE M1
      REWIND M1
      REWIND J1
      J1=J2
      M1=M2
      GO TO 12
C.... PROGRAM ERROR STATEMENTS
   49 PRINT 101
  101 FORMAT (40HO ERROR IN STATEMENT 17,IMPROPER RESULTS)
      GO TO 35
   50 PRINT 102
  102 FORMAT (40HO ERROR IN STATEMENT 21,IMPROPER RESULTS)
      GO TO 35
   51 PRINT 103
  103 FORMAT (40HO ERROR IN STATEMENT 37,IMPROPER RESULTS)
      GO TO 35
      END
```

```
$IBSYS
$ID P13P                              STEP B-2
$PAUSE
$EXECUTE         SORT
         FILE,INPUT/2,BLO/14,REE/1,MOD/D,DENSITY/H
         FILE,OUT,BLO/14,MOD/D,DEN/H
         RECORD,TYPE/F,LEN/(14),FIELD/(6,69,1)
         SORT,FILE/2,ORDER/2,FIE/(1A,3A),SEQ/S
         CHANNELS,INP/A,MERGE/(A,B)
         OPT,TAP
         REMARK ENDCARD 1 DATA SORT
         END
*        END TAPE
```

```
$IBSYS
$ID P13P                             STEP B-3
$PAUSE
$EXECUTE         SORT
         FILE,INPUT/4,BLO/14,REE/1,MOD/D,DEN/H
         FILE,OUT,BLO/14,MOD/D,DEN/H
         RECORD,TYPE/F,LEN/(14),FIELD/(6,69,1)
         SORT,FILE/4,ORDER/2,FIE/(1A,3A),SEQ/S
         CHANNELS,INP/A,MERGE/(A,B)
         OPT,TAP
         REMARK ENDCARD 2 DATA SORT
         END
```

```
*        ALPHA CHECK PROGRAM FOR ENDCARDS 1                  STEP C-2
*        PAUSE
*        XEQ
*        CARDS COLUMN
*        FORMAP
*        FAP
 INIT     ETTA
          NOP
          AXT       14,1
          STZ       WKAR+14,1
          TIX       *-1,1,1
 READ     RTDA      5
          RCHA      COMM1
          TCOA      *
          TRCA      RDERR
          TEFA      END5
          TRA       VALCK
 RDERR    AXT       9,2
 RDBSP    BSRA      5
          RTDA      5
          RCHA      COMM1
          TCOA      *
          TRCA      *+2
          TRA       RDERR-2
          TIX       RDBSP,2,1
          TSX       $(SPH),4
          PZE       MESSA,,1
          TSX       $(FIL),4
          CALL      $EXIT
 VALCK    AXT       14,1
          AXT       6,2
          CAL       WKAR+14,1
          SLW       TEMP
```

```
CK1    ANA      MASK
       LAS      SIXTY
       TRA      ERROR
       TRA      CK2
       LAS      NINE
       TRA      ERROR
       TRA      CK2
CK2    CAL      TEMP
       ARS      6
       SLW      TEMP
       TIX      CK1,2,1
       TIX      VALCK+1,1,1
       TRA      WRITE
ERROR  CAL      WKAR
       SLW      MESSG+2
       CAL      TEMP
       SLW      MESSG+5
       TSX      $(SPH),4
       PZE      MESSG,,1
       TSX      $(FIL),4
       TSX      $(SPH),4
       PZE      MESSB,,1
       TSX      $(FIL),4
       TRA      INIT
WRITE  WTDA     6
       RCHA     COMM1
       TCOA     *
       ETTA
       TRA      END6
       TRCA     WIERR
       TRA      INIT
END5   LXA      SAM,1
       TIX      *+2,1,1
       TRA      LAST
       SXA      SAM,1
       RUNA     5
       TSX      $(SPH),4
       PZE      MESSD,,1
       TSX      $(FIL),4
       HTR      READ
LAST   RUNA     5
       TSX      $(SPH),4
       PZE      MESSE,,1
       TSX      $(FIL),4
       AXT      1,1
       SXA      SAM,1
END6   CAL      CODE
       SLW      WKAR
       WTDA     6
       RCHA     COMM1
       TCOA     *
       WEFA     6
       RUNA     6
       CAL      WKAR
       SLW      MESSG+2
       TSX      $(SPH),4
       PZE      MESSG,,1
       TSX      $(FIL),4
       TSX      $(SPH),4
       PZE      MESSF,,1
       TSX      $(FIL),4
       HTR      INIT
```

```
 WIERR AXT     9,2
 WIBSP BSRA    6
       WTDA    6
       RCHA    COMM1
       TCOA    *
       TRCA    *+2
       TRA     INIT
       BSRA    6
       WTDA    6
       TCOA    *
       TRCA    WZ
       TIX     WIBSP+1,2,1
WZ     TSX     $(SPH),4
       PZE     MESSC,,1
       TSX     $(FIL),4
       CALL    $EXIT
 MESSA BCI     7,(35H READ ERROR ENCOUNTERED, CALL EXIT.)
 MESSB BCI     7,(35H INVALID CHARACTER IN RECORD ABOVE.)
 MESSC BCI     7,(36H WRITE ERROR ENCOUNTERED, CALL EXIT.)
 MESSD BCI     7,(29H PREPARE NEW A5. PRESS START.)
 MESSE BCI     7,(29H LAST A5.PEOCESSING COMPLETE.)
 MESSF BCI     7,(37H PREPARE A6 IF PRCCESSING INCOMPLETE.)
 MESSG BCI     7,(35H                                  )
 WKAR  BSS     14
 SIXTY OCT     60
 CODE  OCT     111111111111
 MASK  OCT     77
 NINE  OCT     11
 TEMP  PZE
 SAM   PZE     1
 COMM1 IOCD    WKAR,,14
       END


*      ALPHA CHECK PROGRAM FOR ENDCARDS 2              STEP C-3
*      PAUSE
*      XEQ
*      CARDS COLUMN
*      FORMAP
*      FAP
 INIT  ETTA
       NOP
       AXT     14,1
       STZ     WKAR+14,1
       TIX     *-1,1,1
 READ  RTDA    5
       RCHA    COMM1
       TCOA    *
       TRCA    RDERR
       TEFA    END5
       TRA     VALCK
 RDERR AXT     9,2
 RDBSP BSRA    5
       RTDA    5
       RCHA    COMM1
       TCOA    *
       TRCA    *+2
       TRA     RDERR-2
       TIX     RDBSP,2,1
       TSX     $(SPH),4
       PZE     MESSA,,1
       TSX     $(FIL),4
```

```
        CALL    $EXIT
VALCK   AXT     14,1
        AXT     6,2
        CAL     WKAR+14,1
        SLW     TEMP
CK1     ANA     MASK
        LAS     SIXTY
        TRA     ERROR
        TRA     CK2
        LAS     NINE
        TRA     ERROR
        TRA     CK2
CK2     CAL     TEMP
        ARS     6
        SLW     TEMP
        TIX     CK1,2,1
        TIX     VALCK+1,1,1
        TRA     WRITE
ERROR   CAL     WKAR
        SLW     MESSG+2
        CAL     TEMP
        SLW     MESSG+5
        TSX     $(SPH),4
        PZE     MESSG,,1
        TSX     $(FIL),4
        TSX     $(SPH),4
        PZE     MESSB,,1
        TSX     $(FIL),4
        TRA     INIT
WRITE   WTDA    6
        RCHA    COMM1
        TCOA    *
        ETTA
        TRA     END6
        TRCA    WIERR
        TRA     INIT
END5    LXA     SAM,1
        TIX     *+2,1,1
        TRA     LAST
        SXA     SAM,1
        RUNA    5
        TSX     $(SPH),4
        PZE     MESSD,,1
        TSX     $(FIL),4
        HTR     READ
LAST    RUNA    5
        TSX     $(SPH),4
        PZE     MESSE,,1
        TSX     $(FIL),4
        AXT     1,1
        SXA     SAM,1
END6    CAL     CODE
        SLW     WKAR
        WTDA    6
        RCHA    COMM1
        TCOA    *
        WEFA    6
        RUNA    6
        CAL     WKAR
        SLW     MESSG+2
        TSX     $(SPH),4
        PZE     MESSG,,1
```

```
       TSX     $(FIL),4
       TSX     $(SPH),4
       PZE     MESSF,,1
       TSX     $(FIL),4
       HTR     INIT
 WIERR AXT     9,2
 WIBSP BSRA    6
       WTDA    6
       RCHA    COMM1
       TCOA    *
       TRCA    *+2
       TRA     INIT
       BSRA    6
       WTDA    6
       TCOA    *
       TRCA    WZ
       TIX     WIBSP+1,2,1
WZ     TSX     $(SPH),4
       PZE     MESSC,,1
       TSX     $(FIL),4
       CALL    $EXIT
 MESSA BCI     7,(35H READ ERROR ENCOUNTERED, CALL EXIT.)
 MESSB BCI     7,(35H INVALID CHARACTER IN RECORD ABOVE.)
 MESSC BCI     7,(36H WRITE ERROR ENCOUNTERED, CALL EXIT.)
 MESSD BCI     7,(29H PREPARE NEW A5. PRESS START.)
 MESSE BCI     7,(29H LAST A5.PEOCESSING COMPLETE.)
 MESSF BCI     7,(37H PREPARE A6 IF PROCESSING INCOMPLETE.)
 MESSG BCI     7,(35H                                   )
 WKAR  BSS     14
 SIXTY OCT     60
 CODE  OCT     111111111111
 MASK  OCT     77
 NINE  OCT     11
 TEMP  PZE
 SAM   PZE     1
 COMM1 IOCD    WKAR,,14
       END

*     PRELIMINARY ANALYSIS FOR ENDCARDS 1               STEP D-2
*     PAUSE
*     XEQ
C**** ENDCARD 1 DATA VALIDITY ANALYSIS   *REA*
      DIMENSION ICHTI(13),NCHTI(13),IDSTI(13),NDSTI(13),IEE(13),NEE(13)
      DIMENSION ICE(13),NCE(13),ICHWH(13),NCHWH(13)
      DIMENSION IDSWH(13),NDSWH(13),IUPSI(13)
C**** BATTERY  0 TEST PARAMETERS
      ICHTI(1)=32500
      NCHTI(1)=500
      IDSTI(1)=5000
      NDSTI(1)=150
      IEE(1)=700
      NEE(1)=30
      ICE(1)=770
      NCE(1)=30
      ICHWH(1)=2840
      NCHWH(1)=50
      IDSWH(1)=1960
      NDSWH(1)=40
      IUPSI(1)=0
C***** BATTERY  1 TEST PARAMETERS
      ICHTI(2)=32500
      NCHTI(2)=500
      IDSTI(2)=20000
      NDSTI(2)=300
      IEE(2)=720
      NEE(2)=25
      ICE(2)=770
      NCE(2)=25
      ICHWH(2)=2820
      NCHWH(2)=50
      IDSWH(2)=2020
      NDSWH(2)=40
      IUPSI(2)=0
C**** BATTERY  2 TEST PARAMETERS
      ICHTI(3)=32500
      NCHTI(3)=500
      IDSTI(3)=1250
      NDSTI(3)=25
      IEE(3)=675
      NEE(3)=25
      ICE(3)=770
      NCE(3)=25
      ICHWH(3)=2790
      NCHWH(3)=50
      IDSWH(3)=1870
      NDSWH(3)=40
      IUPSI(3)=0
C**** BATTERY  3 TEST PARAMETERS
      ICHTI(4)=12900
      NCHTI(4)=600
      IDSTI(4)=5000
      NDSTI(4)=150
      IEE(4)=915
      NEE(4)=25
      ICE(4)=975
      NCE(4)=25
      ICHWH(4)=2220
      NCHWH(4)=150
      IDSWH(4)=2050
      NDSWH(4)=100
      IUPSI(4)=0
C**** BATTERY  4 TEST PARAMETERS
      ICHTI(5)=17000
      NCHTI(5)=4000
      IDSTI(5)=5000
      NDSTI(5)=75
      IEE(5)=800
      NEE(5)=100
      ICE(5)=900
      NCE(5)=100
      ICHWH(5)=2600
      NCHWH(5)=200
      IDSWH(5)=1985
      NDSWH(5)=40
      IUPSI(5)=0
C**** BATTERY  5 TEST PARAMETERS
      ICHTI(6)=5150
      NCHTI(6)=200
      IDSTI(6)=5000
      NDSTI(6)=75
      IEE(6)=915
      NEE(6)=25
      ICE(6)=985
      NCE(6)=25
      ICHWH(6)=2225
      NCHWH(6)=100
      IDSWH(6)=2030
      NDSWH(6)=50
      IUPSI(6)=0
C**** BATTERY  6 TEST PARAMETERS
      ICHTI(7)=19400
      NCHTI(7)=500
      IDSTI(7)=5000
      NDSTI(7)=75
      IEE(7)=895
      NEE(7)=30
      ICE(7)=955
      NCE(7)=30
      ICHWH(7)=2270
      NCHWH(7)=100
      IDSWH(7)=2030
      NDSWH(7)=50
      IUPSI(7)=0
C**** BATTERY  7 TEST PARAMETERS
      ICHTI(8)=65000
      NCHTI(8)=1000
      IDSTI(8)=5000
      NDSTI(8)=75
      IEE(8)=350
      NEE(8)=25
      ICE(8)=385
      NCE(8)=25
      ICHWH(8)=5725
      NCHWH(8)=50
      IDSWH(8)=1975
      NDSWH(8)=40
      IUPSI(8)=0
C**** BATTERY  8 TEST PARAMETERS
      ICHTI(9)=10000
      NCHTI(9)=2000
      IDSTI(9)=5000
      NDSTI(9)=75
      IEE(9)=800
      NEE(9)=90
      ICE(9)=870
      NCE(9)=90
      ICHWH(9)=2300
      NCHWH(9)=175
      IDSWH(9)=1990
      NDSWH(9)=40
      IUPSI(9)=0
C**** BATTERY  9 TEST PARAMETERS
      ICHTI(10)=28000
      NCHTI(10)=3000
      IDSTI(10)=5000
      NDSTI(10)=75
      IEE(10)=760
      NEE(10)=50
      ICE(10)=830
      NCE(10)=50
      ICHWH(10)=2600
      NCHWH(10)=250
      IDSWH(10)=1985
      NDSWH(10)=40
      IUPSI(10)=0
```

```
C**** BATTERY 10 TEST PARAMETERS
      ICHTI(11)=25000
      NCHTI(11)=375
      IDSTI(11)=8667
      NDSTI(11)=1000
      IEE(11)=800
      NEE(11)=100
      ICE(11)=900
      NCE(11)=100
      ICHWH(11)=4400
      NCHWH(11)=200
      IDSWH(11)=3450
      NDSWH(11)=200
      IUPSI(11)=1
C**** BATTERY 11 TEST PARAMETERS
      ICHTI(12)=9150
      NCHTI(12)=300
      IDSTI(12)=9050
      NDSTI(12)=500
      IEE(12)=865
      NEE(12)=100
      ICE(12)=985
      NCE(12)=100
      ICHWH(12)=4075
      NCHWH(12)=100
      IDSWH(12)=3520
      NDSWH(12)=200
      IUPSI(12)=1
      KBAT=88888
      KCYC=88888
      K=2
      JJ=0
      KK=1
C**** PARAMETER DATA LISTING
      WRITE OUTPUT TAPE 3,200
      DO 300 NN=1,12
      NNN=NN-1
      WRITE OUTPUT TAPE 3,201,NNN,ICHTI(NN),NCHTI(NN),IDSTI(NN),
     1NDSTI(NN),IEE(NN),NEE(NN),ICE(NN),NCE(NN),ICHWH(NN),NCHWH(NN),
     2IDSWH(NN),NDSWH(NN),IUPSI(NN)
  300 CONTINUE
      WRITE OUTPUT TAPE 3,202
   98 FORMAT (81H0     BAT    CYCLE      YEAR   DAY   TIME       EEF   WAT
     1T-HR   CEF     CD NO     P)
   99 FORMAT (1H ,10I8)
  100 FORMAT (I2,I4,I1,I3,1X,I5,12X,I3,5X,I5,30X,I3,I1,I1)
  101 FORMAT (40H0 ILLEGAL CARD NUMBER FOR FOLLOWING CARD)
  102 FORMAT (40H0 ILLEGAL YEAR OR DAY FOR FOLLOWING CARD)
  103 FORMAT (34H0 ILLEGAL PHASE FOR FOLLOWING CARD)
  104 FORMAT (32H0 TIME DURATION OUT OF LIMITS BY,I6,15H  ON CARD BELOW)
  105 FORMAT (29H0 WATT-HOURS OUT OF LIMITS BY,I6,15H  ON CARD BELOW)
  106 FORMAT (54H0 ENERGY OR CURRENT EFF. SHOULD BE ZERO FOR CARD BELOW)
  107 FORMAT (30H0 ENERGY EFF. OUT OF LIMITS BY,I6,15H  ON CARD BELOW)
  108 FORMAT (31H0 CURRENT EFF. OUT OF LIMITS BY,I6,15H  ON CARD BELOW)
  110 FORMAT (34H0 INCORRECT NUMBER OF CARDS ON BAT,I4,6H CYCLE,I6)
  111 FORMAT (38H0 BOTH CARDS ARE CHARGE OR DIS. ON BAT,I4,6H CYCLE,I6)
  112 FORMAT (32H0 ALL DATA PROCESSED,EXIT CALLED)
  200 FORMAT(117H1       BAT  CHG-TI   LIMIT  DIS-TI   LIMIT    EEF    LI
     1MIT    CEF    LIMIT CHG-WH   LIMIT  DIS-WH   LIMIT   CYC-TYPE)
  201 FORMAT(1H0,14I8)
  202 FORMAT (1H1,40X,41HOUT OF LIMITS DATA FOR ENDCARDS 1   *REA*)
C**** READ ENDCARD 1 INTO STORAGE
```

```
     1 READ INPUT TAPE 5,100,NBAT,NCYC,NY,NDAY,NTIME,NEEFE,NWH,NCEFE,
    1NC,NPHAS
       IF(NBAT-99)2,33,33
     2 IF (NC-1)3,5,3
     3 ISTNO=1
       WRITE OUTPUT TAPE 3,101
     4 WRITE OUTPUT TAPE 3,98
       WRITE OUTPUT TAPE 3,99 ,NBAT,NCYC,NY,NDAY,NTIME,NEEFE,NWH,NCEFE,
    1NC,NPHAS
       GU TO (25,33),ISTNO
     5 IF (NY-5)7,7,6
     6 ISTNO=1
       WRITE OUTPUT TAPE 3,102
       GO TO 4
     7 IF (NDAY-366)8,8,6
     8 IF (NPHAS-1)22,10,9
     9 ISTNO=1
       WRITE OUTPUT TAPE 3,103
       GO TO 4
C**** DISCHARGE PHASE ANALYSIS
    10 N=NBAT+1
       IDEV =IDSTI(N)-NTIME
       IABDEV=XABSF(IDEV)
       IF (NDSTI(N)-IABDEV)11,12,12
    11 ISTNO=1
       WRITE OUTPUT TAPE 3,104,IDEV
       GO TO 4
    12 IDEV=IDSWH(N)-NWH
       IABDEV=XABSF(IDEV)
       IF (NDSWH(N)-IABDEV)13,14,14
    13 ISTNO=1
       WRITE OUTPUT TAPE 3,105,IDEV
       GO TO 4
    14 IF (IUPSI(N))15,15,18
    15 IF (NEEFE)17,16,17
    16 IF (NCEFE)17,25,17
    17 ISTNO=1
       WRITE OUTPUT TAPE 3,106
       GO TO 4
    18 IDEV=IEE(N)-NEEFE
       IABDEV=XABSF(IDEV)
       IF (NEE(N)-IABDEV)19,20,20
    19 ISTNO=1
       WRITE OUTPUT TAPE 3,107,IDEV
       GO TO 4
    20 IDEV=ICE(N)-NCEFE
       IABDEV=XABSF(IDEV)
       IF (NCE(N)-IABDEV)21,25,25
    21 ISTNO=1
       WRITE OUTPUT TAPE 3,1C8,IDEV
       GO TO 4
C****CHARGE PHASE ANALYSIS
    22 N=NBAT+1
       IDEV=ICHTI(N)-NTIME
       IABDEV=XABSF(IDEV)
       IF (NCHTI(N)-IABDEV)11,23,23
    23 IDEV=ICHWH(N)-NWH
       IABDEV=XABSF(IDEV)
       IF (NCHWH(N)-IABDEV)13,24,24
    24 IF (IUPSI(N))18,18,15
C**** CHECK FOR FULL CYCLE
    25 IF (NBAT-KBAT)28,26,28
```

```
   26 IF (NCYC-KCYC)28,27,28
   27 K=K+1
      JJ=NPHAS
      GO TO 1
   28 IF (K-2)29,30,29
   29 WRITE OUTPUT TAPE 3,11C,KBAT,KCYC
      GO TO 32
   30 IF (JJ-KK)32,31,32
   31 WRITE OUTPUT TAPE 3,111,KBAT,KCYC
   32 K=1
      KK=NPHAS
      KBAT=NBAT
      KCYC=NCYC
      GO TO 1
   33 WRITE OUTPUT TAPE 3,112
      REWIND 5
      CALL EXIT
      END


*        ENDCARD DELETE AND DUPLICATE RECORD ELIMINATION      STEP E-2
*       PAUSE
*       XEQ
*       CARDS COLUMN
C..... ENDCARD 1DELETE AND DUPLICATE RECORD ELIMINATION PROGRAM
C.... BELOW IS FORMAT EXPLANATION FOR RECORDS TO BE DELETED
C *1* COL 1 CONTAINS A 0 IF ENTIRE CYCLE IS TO BE DELETED
C *2* COL 1 CONTAINS A 1 IF PHASE OF CYCLE IS TO BE DELETED ONLY (OUTCARDS
C *3* COL 5 AND 6 CONTAINS BATTERY NUMBER
C *4* COL 8 TO 11 CONTAINS CYCLE NUMBER
C *5* COL 14 CONTAINS PHASE (IF COL 1 CONTAINS 1),0=CHG,1=DISCHARGE
C *6* LAST CARD SIGNAL CONTAINS 99 IN COL 5 AND 6 (THIS ENDS ALL DELETIONS)
C *7* THIS DATA WILL BE PLACED BEHIND DELETE PROGRAM
C.... INITIALIZATION
      DIMENSION NN(20),MM(20)
    2 READ INPUT TAPE 2,100,ID,IBAT,ICYC,IPHAS
    1 REWIND 5
      REWIND 6
      WRITE OUTPUT TAPE 3,103
      ISW=0
C.... FORMAT STATEMENTS
  100 FORMAT (I1,I5,I5,I3)
  101 FORMAT (I2,I4,I1,I3,1X,I5,1X,I5,1X,I5,I3,8I5,I3,2I1,I4)
  102 FORMAT (27HODUPLICATE RECORD DELETED  ,I2,I4,I1,I3,1X,I5,1X,I5,1X,
     1I5,I3,8I5,I3,2I1,I4)
  1C3 FORMAT (1H1,45X,32HD E L E T I O N    P R O G R A M)
  104 FORMAT (13H0  END OF JOB)
  1C5 FORMAT (27HCREQUESTED RECORD DELETED  ,I2,I4,I1,I3,1X,I5,1X,I5,1X,
     1I5,I3,8I5,I3,2I1,I4)
  1C6 FORMAT (36HCREQUEST FOR DELETION OUT OF ORDER  ,4I6)
  107 FORMAT (47HCCOULD NOT LOCATE RECORD FOR DELETION REQUEST  ,2I6)
    3 READ INPUT TAPE 5,1C1,(NN(I),I=1,20)
    4 READ INPUT TAPE- 5,1C1,(MM(I),I=1,20)
      DO 5 I=1,20
      IF (NN(I)-MM(I))9,5,9
    5 CONTINUE
      WRITE OUTPUT TAPE 3,102,(NN(I),I=1,20)
    6 DO 7 I=1,20
      NN(I)=MM(I)
    7 CONTINUE
      IF (MM(1)-99)4,8,8
    8 WRITE OUTPUT TAPE 6,101,(MM(I),I=1,20)
```

```
      WRITE OUTPUT TAPE 3,104
      GO TO 22
    9 IF (IBAT-99)11,10,10
   10 WRITE OUTPUT TAPE 6,101,(NN(I),I=1,20)
      GO TO 6
   11 IF (IBAT-NN(1))16,12,10
   12 IF (ICYC-NN(2))16,13,10
   13 IF (ID)15,15,14
   14 IF (IPHAS-NN(19))10,15,10
   15 WRITE OUTPUT TAPE 3,105,(NN(I),I=1,20)
      ISW=1
      GO TO 6
   16 IF (ISW)21,21,17
   17 READ INPUT TAPE 2,100,ID,IIBAT,IICYC,IPHAS
      ISW=0
      IF (IBAT-IIBAT)19,18,20
   18 IF (ICYC-IICYC)19,20,20
   19 IBAT=IIBAT
      ICYC=IICYC
      GO TO 9
   20 WRITE OUTPUT TAPE 3,106,IBAT,IIBAT,ICYC,IICYC
      GO TO 22
   21 WRITE OUTPUT TAPE 3,107,IBAT,ICYC
      GO TO 17
   22 REWIND 5
      END FILE 6
      REWIND 6
      CALL EXIT
      END
*     DATA
   99


*    NASA BATTERY ENDCARD BCD TO BINARY ROUTINE    STEP F-2, F-3
*     PAUSE
*     XEQ
*     CARDS COLUMN
C     ENDCARD 1 AND 2 MERGE , VALIDITY AND BINARY CONVERSION PROGRAM
C
C.... INPUT TAPE FOR ENDCARD CODES IS A6
C.... INPUT TAPE FOR ENDCARD 1 IMAGES IS A8
C.... INPUT TAPE FOR ENDCARD 2 IMAGES IS B5
C.... OUTPUT BINARY TAPE IS A7
C.... A3 WILL CONTAIN LISTING OF DATA ON BINARY TAPE A7 IN BCD FORM
C....           AND REJECTED DATA MESSAGES
      REWIND 6
      REWIND 7
      REWIND 8
      REWIND 15
      DIMENSION IA(25),IB(25),IOA(25),IOB(25),IOC(25),IOD(25)
      DIMENSION IMOD(15,4),FMOD(65,4)
      NUBAT=0
C
C     INSURE THAT IOA,IOB,IOC,IOD, HAVE UNEQUAL CYCLE TO START
C
 1000 IOA(2)=99999
      IOB(2)=88888
      IOC(2)=77777
      IOD(2)=66666
      ICNT=0
      IND=0
C
```

```
C       FCOUNT IS THE TO TOTAL NUMBER OF CYCLES WRITTEN ON OUTAPE
C
        FCOUNT=0.0
C
C       HEADING AND MESSAGE INFORMATION ON TAPE 3
C
    1 WRITE OUTPUT TAPE 3,100
  100 FORMAT (49H1ENDCARD 1 AND 2 MERGE AND VALIDITY CHECK PROGRAM)
      WRITE OUTPUT TAPE 3,101
  101 FORMAT (49H0LRL-MARTIN COMPANY NASA BATTERY TEST PROGRAM-REA)
C
C       TAPE 6 CONTAINS CODING INFORMATION
C
    2 READ INPUT TAPE 6,102,KODBAT,KODCYC,KOD,KODTIM
  102 FORMAT (I6,I6,I3,I3)
C
C      TAPE 8 CONTAINS ENDCARD 1 RECORDS
C
    3 READ INPUT TAPE 8,950,(IA(I),I=1,19)
  950 FORMAT (I2,I4,I1,I3,3I6,I3,8I5,I3,I2,I1)
  103 FORMAT (1H ,I4,I6,I3,I5,3I8,I5,8I7,I5,I4,I3)
      IF(IA(1)-99)5,46,46
C
C       CHECK TO SEE IF A NEXT BATTERY HAS BEEN ENCOUNTERED
C
    5 IF (NUBAT-IA(1))6,7,6
    6 WRITE OUTPUT TAPE 3,107,NUBAT
  107 FORMAT (8H0BATTERY,I4,2X,9HCOMPLETE.)
      NUBAT=IA(1)
C
C       CHECK FOR CARD AND PHASE
C
    7 IF (IA(18)-10)14,9,8
    8 IF (IA(18)-11)14,11,14
C
C       PLACE ENDCARD 1 IN PROPER ARRAY
C
    9 DO 10 N=1,19
      IOA(N)=IA(N)
   10 CONTINUE
      GO TO 13
   11 DO 12 N=1,19
      IOC(N)=IA(N)
   12 CONTINUE
C
C       COMPARE CYCLE NUMBERS IN IOA AND IOC ARRAYS
C
   13 IF (IOA(2)-IOC(2))3,15,3
C
C       ERROR ROUTINE FOR ILLEGAL PHASE CODE FOR ENDCARD 1
C
   14 WRITE OUTPUT TAPE 3,109
  109 FORMAT (45H0CARD BELOW HAS ILLEGAL PHASE, CYCLE OMITTED.)
      WRITE OUTPUT TAPE 3,103,(IA(I),I=1,19)
      GO TO 31
C
C       TAPE 15 CONTAINS ENDCARD 2 RECORDS
C
   15 READ INPUT TAPE 15,951,(IB(I),I=1,21)
  951 FORMAT (I2,I4,I1,I3,11I4,4I5,I2,I1)
  110 FORMAT (1H ,I4,I6,I3,I5,11I6,4I7,I4,I3)
C
```

```
C     RUN OUT ENDCARD 2 FOR PREVIOUS BATTERY
C
      IF(IOA(1)-IB(1))200,200,600
  6C0 WRITE OUTPUT TAPE 3,601
  6C1 FORMAT (47HOENDCARD 2 BELOW NOT USED, IMPROPER BATTERY NO.)
      WRITE OUTPUT TAPE 3,11C,(IB(I),I=1,21)
      GO TO 15
C
C     COMPARE ENDCARD 1 CYCLE TO ENDCARD 2 CYCLE NUMBER
C
  200 IF(IB(1)-99)203,46,46
  203 IF (IOA(2)-IB(2))17,18,16
   16 WRITE OUTPUT TAPE 3,111
  111 FORMAT (47HCENDCARD 2 CYCLE LESS THAN ENDCARD 1,OMIT CARD.)
      WRITE OUTPUT TAPE 3,110,(IB(I),I=1,21)
      GO TO 15
   17 WRITE OUTPUT TAPE 3,112
  112 FORMAT (49HOONE OR BOTH ENDCD.2 MISSING,OMIT ENDCD. 1 BELOW.)
      WRITE OUTPUT TAPE 3,103,(IOA(I),I=1,19)
      WRITE OUTPUT TAPE 3,103,(IOC(I),I=1,19)
      BACKSPACE 15
      GO TO 31
C
C     PROPER ENDCARD 2 SELECTED
C
C
C     CHECK CARD NO. AND PHASE
C
   18 IF (IB(20)-20)25,20,19
   19 IF (IB(20)-21)25,22,25
C
C     PLACE ENDCARD 2 IMPROPER ARRAY IOB OR IOD
C
   20 DO 21 N=1,21
      IOB(N)=IB(N)
   21 CONTINUE
      GO TO 24
   22 DO 23 N=1,21
      IOD(N)=IB(N)
   23 CONTINUE
C
C     COMPARE CYCLE NUMBERS IN IOB AND IOD ARRAYS
C
   24 IF (IOB(2)-IOD(2))15,26,15
C
C     ERROR ROUTINE FOR ILLEGAL PHASE IN ENDCARD 2
C
   25 WRITE OUTPUT TAPE 3,113
  113 FORMAT (47HOENDCD.2 BELOW HAS ILLEGAL PHASE,CYCLE OMITTED.)
      WRITE OUTPUT TAPE 3,110,(IB(I),I=1,21)
      GO TO 31
C
C     IOA,IOB,IOC,AND IOD ARRAYS COMPOSE AN ENTIRE CYCLE, CHECK VALIDITY
C
   26 DO 28 N=1,3
      IF (IOA(N)-IOB(N))29,27,29
   27 IF (IOC(N)-IOD(N))29,28,29
   28 CONTINUE
      GO TO 300
   29 WRITE OUTPUT TAPE 3,114
  114 FORMAT (42HOBANO,CYNO,YEAR,OR DAY ILLEGAL,OMIT CYCLE.)
   3C WRITE OUTPUT TAPE 3,103,(IOA(I),I=1,19)
```

```
      WRITE OUTPUT TAPE 3,110,(IOB(I),I=1,21)
      WRITE OUTPUT TAPE 3,103,(IOC(I),I=1,19)
      WRITE OUTPUT TAPE 3,110,(IOD(I),I=1,21)
   31 IOA(2)=99999
      IOB(2)=88888
      IOC(2)=77777
      IOD(2)=66666
      GO TO 3
C
C     CHECK FOR ZERO EFFICIENCY
C
  300 IF (IOA(1)-10)33,35,35
   33 IF (IOA(8))34,37,34
   34 IF(IOA(17))301,37,301
   35 IF (IOC(8))36,37,36
   36 IF (IOC(17))301,37,301
C
C     CHECK FOR ACTIVE CELL BEING UNEQUAL FOR CHG. AND DIS.
C
  301 IF (IOB(17)-IOD(17))303,302,303
  302 IF (IOB(18)-IOD(18))303,38,303
  303 WRITE OUTPUT TAPE 3,304
  304 FORMAT (36HOACTIVE CELL ON HALF CYCLE DISAGREE.)
      GO TO 30
   37 WRITE OUTPUT TAPE 3,115
  115 FORMAT (33HOEFFICIENCYS ARE ZERO,OMIT CYCLE.)
      GO TO 30
C
C     TEST TO SEE IF A CODE SHOULD BE PLACED ON THIS CYCLE
C
   38 IF (KODBAT-99)39,42,42
   39 IF (KODBAT-IOA(1))44,40,42
   40 IF (KODCYC-IOA(2))44,41,42
   41 IOA(20)=KOD
      IOA(21)=KODTIM
      IOA(22)=0
      READ INPUT TAPE 6,102,KODBAT,KODCYC,KOD,KODTIM
      GO TO 43
   42 IOA(20)=0
      IOA(21)=0
      IOA(22)=0
C
C     THE IO ARRAYS ARE READY TO BE PLACED IN OUTPUT IMOD-FMOD ARRAYS
C
   43 IF (ICNT-4)50,66,66
C
C     ERROR ENCOUNTERED IN STATEMENT 39 OR 40
C
   44 WRITE OUTPUT TAPE 3,119
  119 FORMAT (34HODATA WAS REJECTED FOR CODE BELOW.)
      WRITE OUTPUT TAPE 3,102,KODBAT,KODCYC,KOD,KODTIM
      READ INPUT TAPE 6,102,KODBAT,KODCYC,KOD,KODTIM
      GO TO 38
   45 WRITE OUTPUT TAPE 3,120
  120 FORMAT (16HOEND OF PROGRAM.)
      REWIND 8
      REWIND 15
      REWIND 6
      END FILE 7
      REWIND 7
      WRITE OUTPUT TAPE 3,121,FCOUNT
  121 FORMAT (34H TOTAL NUMBER OF CYCLES PROCESSED=,F15.0)
```

```
C
C     ROUTINE TO CONVERT BINARY TO BCD FOR PROGRAM CHECK
      PRINT 121,FCOUNT
      NUBAT=0
  401 WRITE OUTPUT TAPE 3,404
  404 FORMAT (1H1,40X,43H BINARY TAPE CONVERTED TO BCD FOR ENDCARDS.)
  405 READ TAPE 7,IND,ICNT,((IMOD(I,J),I=1,11),(FMOD(I,J),I=12,63),
     1J=1,ICNT)
      DO 406 J=1,ICNT
      IF (IMOD(2,J)-NUBAT)700,700,800
  800 NUBAT=IMOD(2,J)
      WRITE OUTPUT TAPE 3,404
  700 WRITE OUTPUT TAPE 3,650
  650 FORMAT (1H0,131H         ID BAT CYC Y DAY S  A  B    C  0123456789
     1TI-DUR AMP-MI WAT-MI EEF  AMP-H WAT-H  AVCUR  MICUR  MXCUR AVPWR M
     2IPWR MXPWR CEF )
      WRITE OUTPUT TAPE 3,500, (IMOD(I,J),I=1,11),(FMOD(I,J),I=12,24)
      WRITE OUTPUT TAPE 3,501, (IMOD(I,J),I=1,11),(FMOD(I,J),I=38,50)
      WRITE OUTPUT TAPE 3,651
  651 FORMAT (1H0,131H         ID BAT CYC Y DAY S  A  B    C  0123456789
     1 E/I  CYC 0 CYC 1 CYC 2 CYC 3 CYC 4 CYC 5 CYC 6 CYC 7 CYC 8 CYC 9
     2MAXVAR TIME     )
      WRITE OUTPUT TAPE 3,502, (IMOD(I,J),I=1,11),(FMOD(I,J),I=25,37)
      WRITE OUTPUT TAPE 3,503, (IMOD(I,J),I=1,11),(FMOD(I,J),I=51,63)
  406 CONTINUE
      IF (IND)405,405,408
  500 FORMAT (1H ,8HCD 1 CHG,I2,I3,I5,I2,I4,I2,I3,I4,I5,I6,I5,F7.4,
     12F7.2,F5.1,2F6.3,3F7.3,3F6.3,F5.1)
  501 FORMAT (1H ,8HCD 1 DIS,I2,I3,I5,I2,I4,I2,I3,I4,I5,I6,I5,F7.4,
     12F7.2,F5.1,2F6.3,3F7.3,3F6.3,F5.1)
  502 FORMAT (1H ,8HCD 2 CHG,I2,I3,I5,I2,I4,I2,I3,I4,I5,I6,I5,F6.3,
     110F6.0,F6.3,F7.3)
  503 FORMAT (1H ,8HCD 2 DIS,I2,I3,I5,I2,I4,I2,I3,I4,I5,I6,I5,F6.3,
     110F6.0,F6.3,F7.3)
  408 REWIND 7
  402 CALL EXIT
C
C     LAST RECORD ON LAST TAPE ENCOUNTERED
C
   46 IND=1
      GO TO 66
C
C     IOA,IOB,IOC,AND IOD CONTENTS TO BE PACKED IN IMOD COMMON DATA
C
   50 ICNT=ICNT+1
      N=ICNT
   51 IMOD(1,N)=1                                                       ID
      IMOD(2,N)=IOA(1)                                                  BATT
      IMOD(3,N)=IOA(2)                                                  CYCL
      IMOD(4,N)=IOA(3)                                                  YEAR
      IMOD(5,N)=IOA(4)                                                  DAY
      IMOD(6,N)=IOA(19)                                                 SEQ
      IMOD(7,N)=IOA(20)                                                 CODE
      IMOD(8,N)=IOA(21)                                                 CODE
      IMOD(9,N)=IOA(22)                                                 CODE
      IMOD(10,N)=IOB(17)                                                ACT
      IMOD(11,N)=IOB(18)                                                ACT
C
C     REMAINDING CONTENTS OF ARRAYS TO BE PACKED INTO FMOD ARRAY
C                           START ENDCARD 1 CHARGE
   52 FMOD(12,N)=IOA(5)                                                 TIME
      FMOD(12,N)=FMOD(12,N)*.0001
      FMOD(13,N)=IOA(6)                                               AMP-
      FMOD(13,N)=FMOD(13,N)*.01
      FMOD(14,N)=IOA(7)                                               WATT
      FMOD(14,N)=FMOD(14,N)*.01
      FMOD(15,N)=IOA(8)                                               ENGY
      FMOD(15,N)=FMOD(15,N)*.1
   53 I=9                                                             *WAT
      DO 54 K=16,23                                                   *
      FMOD(K,N)=IOA(I)                                                * TH
      FMOD(K,N)=FMOD(K,N)*.001                                        *
      I=I+1                                                           * MA
   54 CONTINUE                                                        * PC
   55 FMOD(24,N)=IOA(17)                                              CUR-
      FMOD(24,N)=FMOD(24,N)*.1
C
C     START PACKING ENDCARD 2 CHARGE
C
   56 FMOD(25,N)=IOB(5)                                               EI R
      FMOD(25,N)=FMOD(25,N)*.001
      I=6                                                             * CE
      DO 57 K=26,35                                                   *
      FMOD(K,N)=IOB(I)                                                * CY
      I=I+1                                                           *
   57 CONTINUE                                                        * CC
      FMOD(36,N)=IOB(16)                                              MAX.
      FMOD(36,N)=FMOD(36,N)*.001
      FMOD(37,N)=IOB(19)                                              TIME
      FMOD(37,N)=FMOD(37,N)*.001
C
C     START PACKING ENDCARD 1 DISCHARGE
C
   58 FMOD(38,N)=IOC(5)                                               DEL
      FMOD(38,N)=FMOD(38,N)*.0001
      FMOD(39,N)=IOC(6)                                               AMP-
      FMOD(39,N)=FMOD(39,N)*.01
      FMOD(40,N)=IOC(7)                                               WATT
      FMOD(40,N)=FMOD(40,N)*.01
      FMOD(41,N)=IOC(8)                                               ENGY
      FMOD(41,N)=FMOD(41,N)*.1
   59 I=9                                                             *WAT
      DO 60 K=42,49                                                   *
      FMOD(K,N)=IOC(I)                                                * TH
      FMOD(K,N)=FMOD(K,N)*.001                                        *
      I=I+1                                                           * MA
   60 CONTINUE                                                        * PC
   61 FMOD(50,N)=IOC(17)                                              CUR-
      FMOD(50,N)=FMOD(50,N)*.1
C
C     START PACKING ENDCARD 2 DISCHARGE
C
   63 FMOD(51,N)=IOD(5)                                               EI F
      FMOD(51,N)=FMOD(51,N)*.001
      I=6                                                             * CE
      DO 64 K=52,61                                                   *
      FMOD(K,N)=IOD(I)                                                * CY
      I=I+1                                                           *
   64 CONTINUE                                                        * CC
      FMOD(62,N)=IOD(16)                                              MAX.
      FMOD(62,N)=FMOD(62,N)*.001
      FMOD(63,N)=IOD(19)                                              TIME
      FMOD(63,N)=FMOD(63,N)*.001
   65 GO TO 31
```

```
C     TAPE 7 IS USED FOR OUTPUT TAPE
   66 CNT=ICNT
      FCOUNT=FCOUNT+CNT
      WRITE TAPE 7,IND,ICNT,((IMOD(I,J),I=1,11),(FMOD(I,J),I=12,63),
     1J=1,ICNT)
      ICNT=0
      IF (IND-1)50,45,45
      END


*     NASA BATTERY REASEARCH ENDCARD BINARY MERGE          STEP G-2
*     PAUSE
*     XEQ
*     CARDS COLUMN
C.... NOTES
C.... PLACE BINARY TAPE TO BE MERGED ON B8
C.... PLACE OLD DATA TAPE FOR BAT 0 TO 7 ON A6
C.... PLACE OLD DATA TAPE FOR BAT 8 TO 11 ON A7
C.... PLACE BLANK TAPE ON B6 FOR MERGED BAT 0 TO 7
C.... PLACE BLANK TAPE ON B7 FOR MERGED BAT 8 TO 11
C.... PLACE BLANK SCRATCH TAPE ON A5 FOR SCRATCH ONLY
C.... INITIALIZATION
      DIMENSION IA(11,4),AA(52,4),IB(11,4),BB(52,4)
      II=5
      IN=18
      REWIND II
      REWIND IN
      IWS=0
  200 N=1
      READ TAPE IN,INA,ICA,((IA(L,K),L=1,11),(AA(M,K),M=1,52),K=1,ICA)
  300 IF (N-ICA)700,400,600
  400 WRITE TAPE II,INA,ICA,((IA(L,K),L=1,11),(AA(M,K),M=1,52),K=1,ICA)
  500 IF (INA)200,200,5000
  600 WRITE OUTPUT TAPE 3,601
  601 FORMAT (35HOERROR IN STATEMENT 400,EXIT CALLED)
      GO TO 2100
  700 IF (IA(2,N)-IA(2,N+1))1000,800,900
  800 N=N+1
      GO TO 300
  900 WRITE OUTPUT TAPE 3,901
  901 FORMAT (35HOERROR IN STATEMENT 700,EXIT CALLED)
      GO TO 2100
 1000 IF (INA)1200,1200,1100
 1100 INA=0
      ISW=1
 1200 WRITE TAPE II,INA,N,((IA(L,J),L=1,11),(AA(M,J),M=1,52),J=1,N)
      K=1
 1300 DO 1400 L=1,11
      IA(L,K)=IA(L,N+1)
 1400 CONTINUE
      DO 1500 L=1,52
      AA(L,K)=AA(L,N+1)
 1500 CONTINUE
      IF((N+1)-ICA)1600,1800,1700
 1600 K=K+1
      N=N+1
      GO TO 1300
 1700 WRITE OUTPUT TAPE 3,1701
 1701 FORMAT (53HOERROR IN STATEMENT BETWEEN 1500 AND 1600,EXIT CALLED)
      GO TO 2100
 1800 IF (ISW)2000,2000,1900
 1900 INA=1
 2000 WRITE TAPE II,INA,K,((IA(L,J),L=1,11),(AA(M,J),M=1,52),J=1,K)
      GO TO 500
 2100 REWIND IN
      REWIND II
      CALL EXIT
 5000 END FILE II
      REWIND II
      REWIND IN
   10 ISW=0
      KSW=0
C.... II IS BINARY TAPE TO BE MERGED
      II=5
      REWIND II
C.... J1 IS BAT 0 TO 7 OLD DATA,J2 IS BAT 8 TO 11 OLD DATA
      J1=6
      J2=7
      REWIND J1
      REWIND J2
C.... M1 IS BAT 0 TO 7 MERGED NEW DATA,M2 IS BAT 8 TO 11 MERGED NEW DATA
      M1=16
      M2=17
      REWIND M1
      REWIND M2
      GO TO 11
C.... READ TAPE ROUTINES ***
C.... READ ROUTINE FOR TAPE TO BE MERGED
    1 READ TAPE II,INB,ICB,((IB(N,M),N=1,11),(BB(N,M),N=1,52),M=1,ICB)
      GO TO (12,25,33,45),L1
C
C.... READ ROUTINE FOR OLD DATA
    2 READ TAPE J1,INA,ICA,((IA(N,M),N=1,11),(AA(N,M),N=1,52),M=1,ICA)
      GO TO (13,20),L2
C
C.... WRITE TAPE ROUTINES ***
C.... WRITE ROUTINE FROM AA ARRAY
    3 WRITE TAPE M1,INA,ICA,((IA(N,M),N=1,11),(AA(N,M),N=1,52),M=1,ICA)
      GO TO (19,30,35,39,48),L3
C
C.... WRITE ROUTINE FROM BB ARRAY
    4 WRITE TAPE M1,INB,ICB,((IB(N,M),N=1,11),(BB(N,M),N=1,52),M=1,ICB)
      GO TO (24,31,13),L4
C
C.... MAINLINE PROGRAM
   11 L1=1
      GO TO 1
   12 L2=1
      GO TO 2
   13 IABAT=IA(2,1)
   14 IF (ISW)15,15,27
   15 IF (INA)16,16,36
   16 IF (KSW)17,17,18
   17 IF (IABAT-IA(2,1))21,18,49
   18 L3=1
      GO TO 3
   19 L2=2
      GO TO 2
   20 GO TO 14
C.... NEXT BATTERY ENCOUNTERED ON J1,CHECK FOR SAME BAT ON II
   21 IF (IABAT-IB(2,1))13,22,50
   22 IF (INB)23,23,26
   23 L4=1
      GO TO 4
   24 L1=2
      GO TO 1
   25 GO TO 21
   26 KSW=1
      INB=0
      L4=3
      GO TO 4
C.... EXIT ROUTINE
   27 IF (INA)16,16,28
   28 IF (KSW)29,29,34
   29 INA=0
      L3=2
      GO TO 3
   30 L4=2
      GO TO 4
   31 IF (INB)32,32,35
   32 L1=3
      GO TO 1
   33 GO TO 30
   34 L3=3
      GO TO 3
   35 PRINT 100
  100 FORMAT (12HO END OF JOB)
      END FILE M1
      REWIND M1
      REWIND II
      REWIND J1
      CALL EXIT
C.... END BAT 7 ROUTINE
   36 IF (KSW)37,37,47
   37 IF (IB(2,1)-7)51,38,47
   38 INA=0
      L3=4
      GO TO 3
   39 INA=INB
      ICA=ICB
      DO 42 N=1,ICB
      DO 40 M=1,11
      IA(M,N)=IB(M,N)
   40 CONTINUE
      DO 41 M=1,52
      AA(M,N)=BB(M,N)
   41 CONTINUE
   42 CONTINUE
   43 IF (INB)44,44,46
   44 L1=4
      GO TO 1
   45 GO TO 37
   46 KSW=1
   47 INA=1
      L3=5
      GO TO 3
   48 ISW=1
      REWIND J1
      END FILE M1
      REWIND M1
      J1=J2
      M1=M2
      GO TO 12
C.... PROGRAM ERROR STATEMENTS
   49 PRINT 101
  101 FORMAT (40HO ERROR IN STATEMENT 17,IMPROPER RESULTS)
```

```
     GO TO 35
  50 PRINT 102
 102 FORMAT (40HO ERROR IN STATEMENT 21,IMPROPER RESULTS)
     GO TO 35
  51 PRINT 103
 103 FORMAT (40HO ERROR IN STATEMENT 37,IMPROPER RESULTS)
     GO TO 35
     END
```

APPENDIX C

FLOW CHARTS

AND

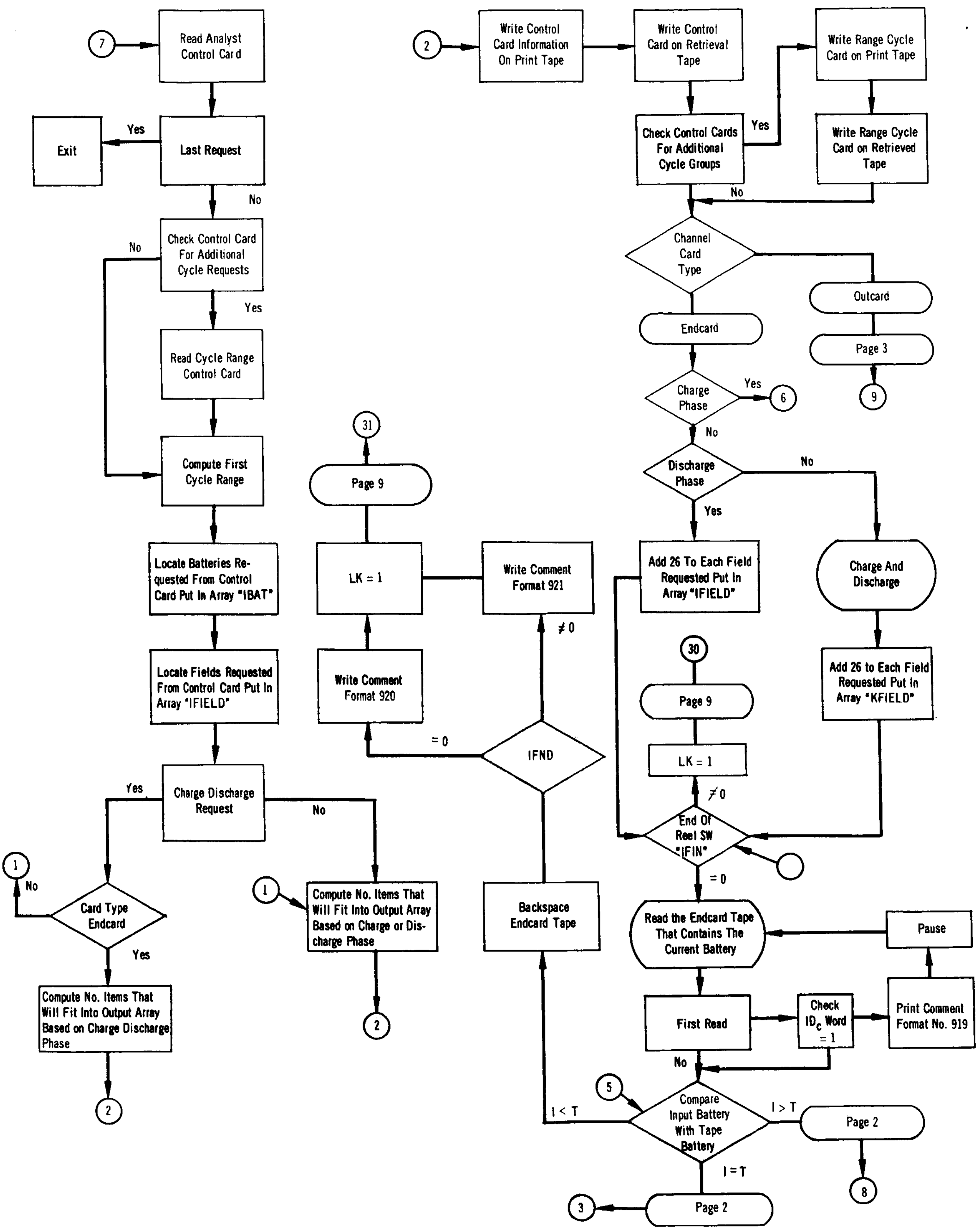
SOURCE DECK STATEMENTS

FOR THE SYSTEM OF BATTERY ANALYSIS
COMPUTER PROGRAMS

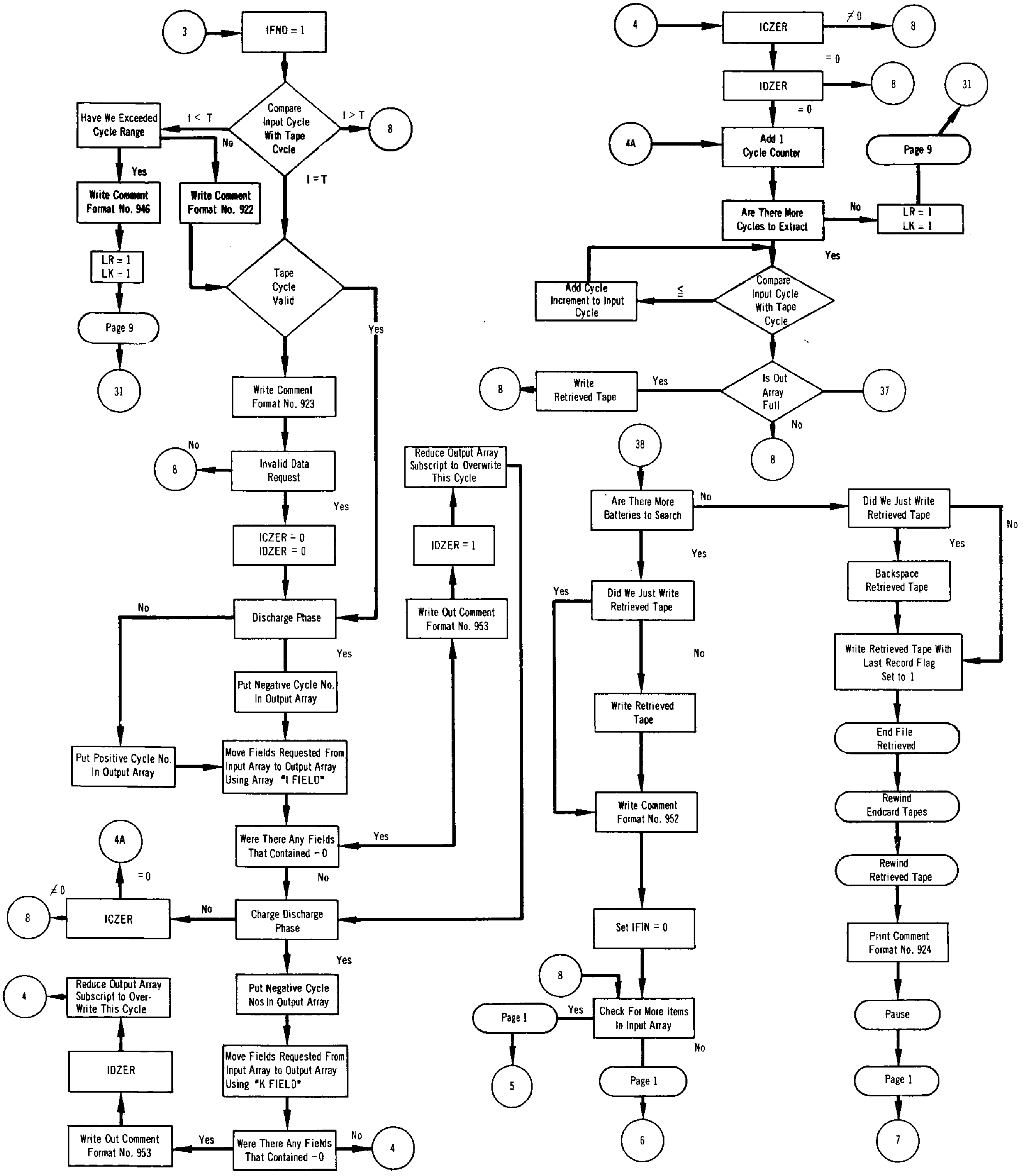
CONTENTS

1. Retrieve-Formatter Routine

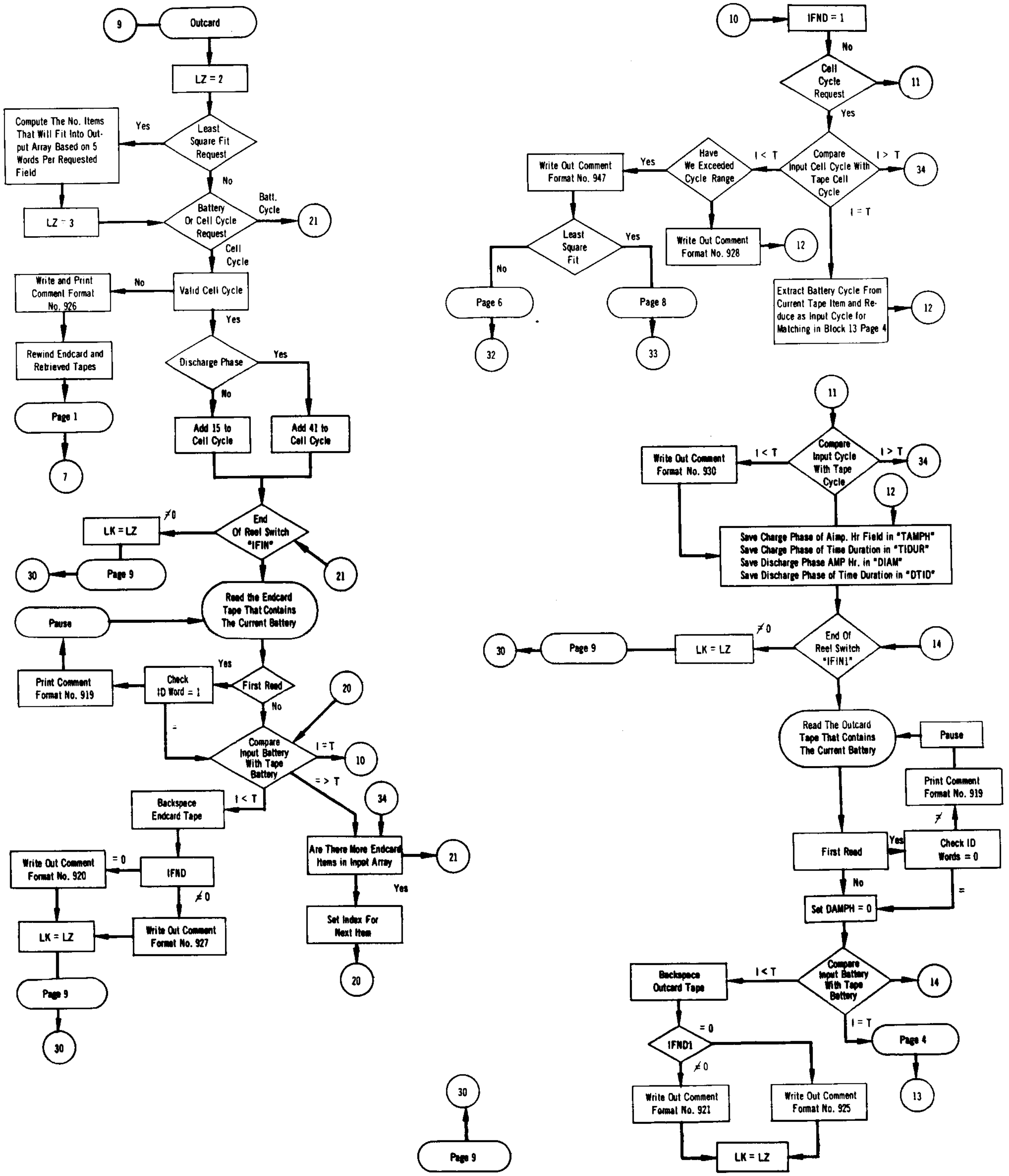
1a. Data Retrieve Formatter Routine Flow Charts


7
Read Analyst Control Card
Last Request
Yes
Exit
No
Check Control Card For Additional Cycle Requests
No
Yes
Read Cycle Range Control Card
Compute First Cycle Range
Locate Batteries Requested From Control Card Put In Array "IBAT"
Locate Fields Requested From Control Card Put In Array "IFIELD"
Charge Discharge Request
Yes
No
1
No
Card Type Endcard
Yes
Compute No. Items That Will Fit Into Output Array Based on Charge Discharge Phase
2
1
Compute No. Items That Will Fit Into Output Array Based on Charge or Discharge Phase
2
31
Page 9
LK = 1
Write Comment Format 921
≠ 0
Write Comment Format 920
= 0
IFND
Backspace Endcard Tape
2
Write Control Card Information On Print Tape
Write Control Card on Retrieval Tape
Check Control Cards For Additional Cycle Groups
Yes
Write Range Cycle Card on Print Tape
Write Range Cycle Card on Retrieved Tape
No
Channel Card Type
Outcard
Page 3
9
Endcard
Charge Phase
Yes
6
No
Discharge Phase
No
Yes
Add 26 To Each Field Requested Put In Array "IFIELD"
Charge And Discharge
Add 26 to Each Field Requested Put In Array "KFIELD"
30
Page 9
LK = 1
≠ 0
End Of Reel SW "IFIN"
= 0
Read the Endcard Tape That Contains The Current Battery
Pause
First Read
Check ID$_c$ Word = 1
Print Comment Format No. 919
No
5
Compare Input Battery With Tape Battery
I < T
I > T
Page 2
8
I = T
Page 2
3

1a. Data Retrieve Formatter Routine Flow Charts (Continued)

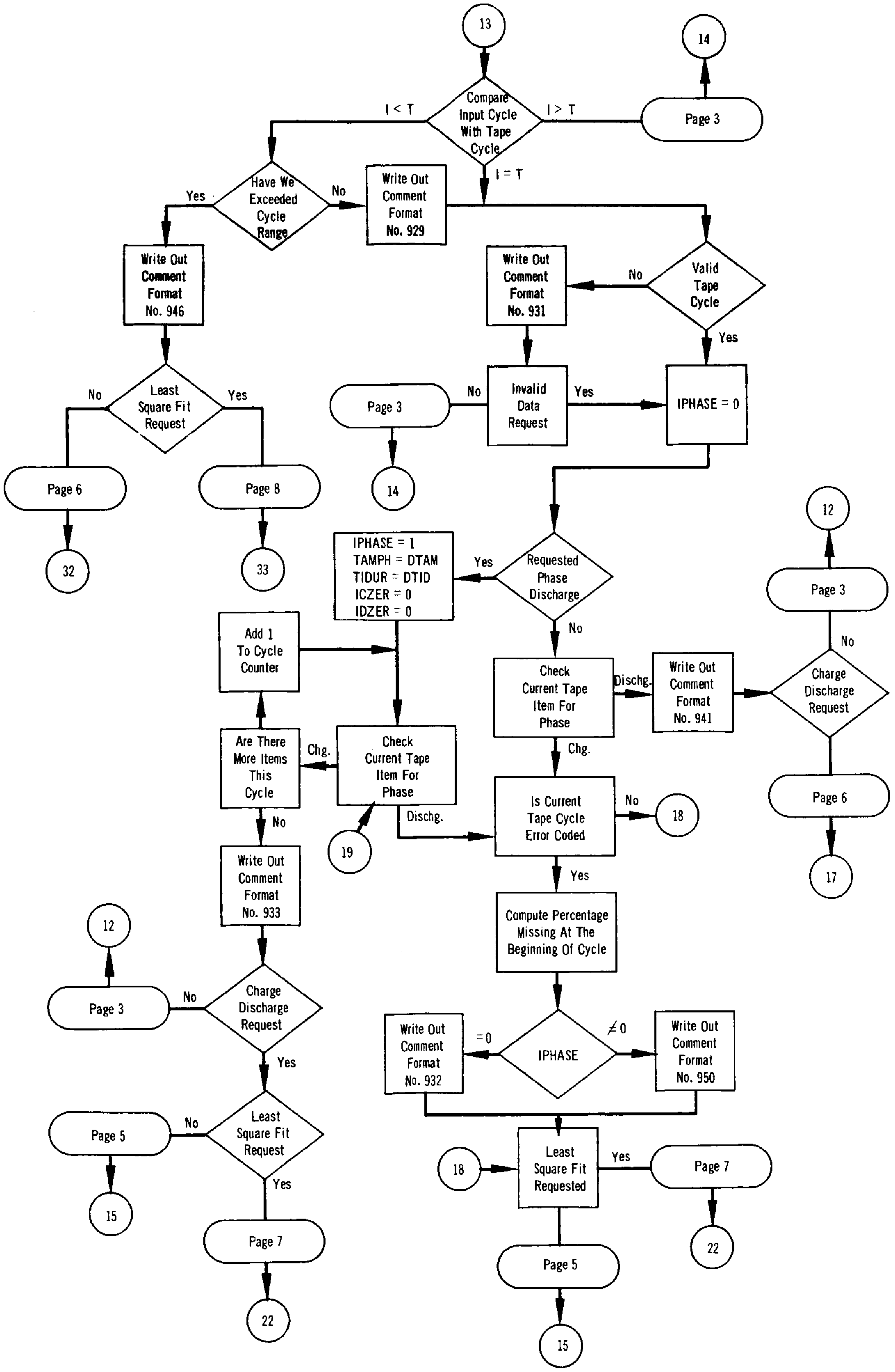

3
IFND = 1
Compare Input Cycle With Tape Cycle
I < T
I > T
I = T
8
No
Have We Exceeded Cycle Range
Yes
Write Comment Format No. 946
Write Comment Format No. 922
LR = 1
LK = 1
Page 9
31
Tape Cycle Valid
Yes
Write Comment Format No. 923
No
8
Invalid Data Request
Yes
ICZER = 0
IDZER = 0
Discharge Phase
No
Yes
Put Negative Cycle No. In Output Array
Put Positive Cycle No. In Output Array
Move Fields Requested From Input Array to Output Array Using Array "I FIELD"
Were There Any Fields That Contained - 0
Yes
No
Reduce Output Array Subscript to Overwrite This Cycle
IDZER = 1
Write Out Comment Format No. 953
4A
= 0
≠ 0
8
ICZER
No
Charge Discharge Phase
Yes
4
Reduce Output Array Subscript to Over-Write This Cycle
Put Negative Cycle Nos In Output Array
IDZER
Move Fields Requested From Input Array to Output Array Using "K FIELD"
Write Out Comment Format No. 953
Yes
Were There Any Fields That Contained - 0
No
4
4
ICZER
≠ 0
8
= 0
IDZER
8
= 0
31
4A
Add 1 Cycle Counter
Page 9
Are There More Cycles to Extract
No
LR = 1
LK = 1
Yes
Add Cycle Increment to Input Cycle
≤
Compare Input Cycle With Tape Cycle
8
Write Retrieved Tape
Yes
Is Out Array Full
37
No
8
38
Are There More Batteries to Search
No
Yes
Did We Just Write Retrieved Tape
Yes
No
Write Retrieved Tape
Write Comment Format No. 952
Set IFIN = 0
8
Page 1
Yes
Check For More Items In Input Array
No
5
Page 1
6
Did We Just Write Retrieved Tape
No
Yes
Backspace Retrieved Tape
Write Retrieved Tape With Last Record Flag Set to 1
End File Retrieved
Rewind Endcard Tapes
Rewind Retrieved Tape
Print Comment Format No. 924
Pause
Page 1
7

1a. Data Retrieve Formatter Flow Charts (Continued)

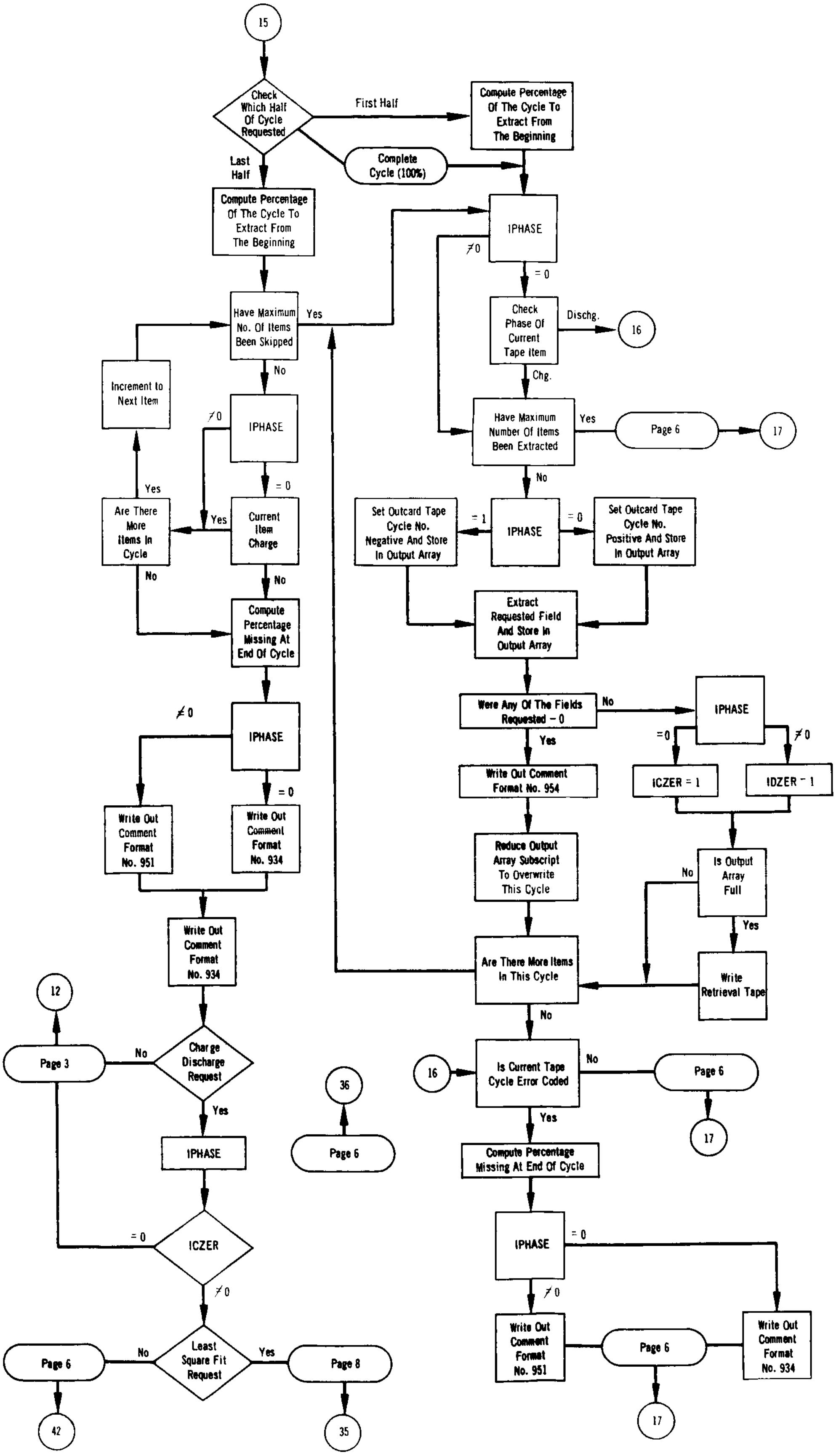


9
Outcard
LZ = 2
Least Square Fit Request
Yes
Compute The No. Items That Will Fit Into Output Array Based on 5 Words Per Requested Field
No
LZ = 3
Battery Or Cell Cycle Request
Batt. Cycle
21
Cell Cycle
Valid Cell Cycle
No
Write and Print Comment Format No. 926
Rewind Endcard and Retrieved Tapes
Page 1
7
Yes
Discharge Phase
Yes
No
Add 15 to Cell Cycle
Add 41 to Cell Cycle
End Of Reel Switch "IFIN"
≠0
LK = LZ
Page 9
30
21
Read the Endcard Tape That Contains The Current Battery
Pause
Yes
First Read
Check ID Word = 1
Print Comment Format No. 919
No
20
=
Compare Input Battery With Tape Battery
I = T
10
=> T
34
I < T
Backspace Endcard Tape
IFND
= 0
Write Out Comment Format No. 920
≠0
Write Out Comment Format No. 927
LK = LZ
Page 9
30
Are There More Endcard Items in Input Array
21
Yes
Set Index For Next Item
20
10
IFND = 1
No
Cell Cycle Request
11
Yes
Compare Input Cell Cycle With Tape Cell Cycle
I > T
34
I < T
Have We Exceeded Cycle Range
Yes
Write Out Comment Format No. 947
Write Out Comment Format No. 928
12
I = T
Least Square Fit
No
Yes
Page 6
Page 8
32
33
Extract Battery Cycle From Current Tape Item and Reduce as Input Cycle for Matching in Block 13 Page 4
12
11
Compare Input Cycle With Tape Cycle
I < T
I > T
34
Write Out Comment Format No. 930
12
Save Charge Phase of Aimp. Hr Field in "TAMPH"
Save Charge Phase of Time Duration in "TIDUR"
Save Discharge Phase AMP Hr. in "DIAM"
Save Discharge Phase of Time Duration in "DTID"
≠0
End Of Reel Switch "IFIN1"
14
LK = LZ
Page 9
30
Read The Outcard Tape That Contains The Current Battery
Pause
Print Comment Format No. 919
≠
Yes
First Read
Check ID Words = 0
No
=
Set DAMPH = 0
Compare Input Battery With Tape Battery
I < T
14
Backspace Outcard Tape
I = T
Page 4
13
= 0
IFND1
≠0
Write Out Comment Format No. 921
Write Out Comment Format No. 925
LK = LZ
30
Page 9

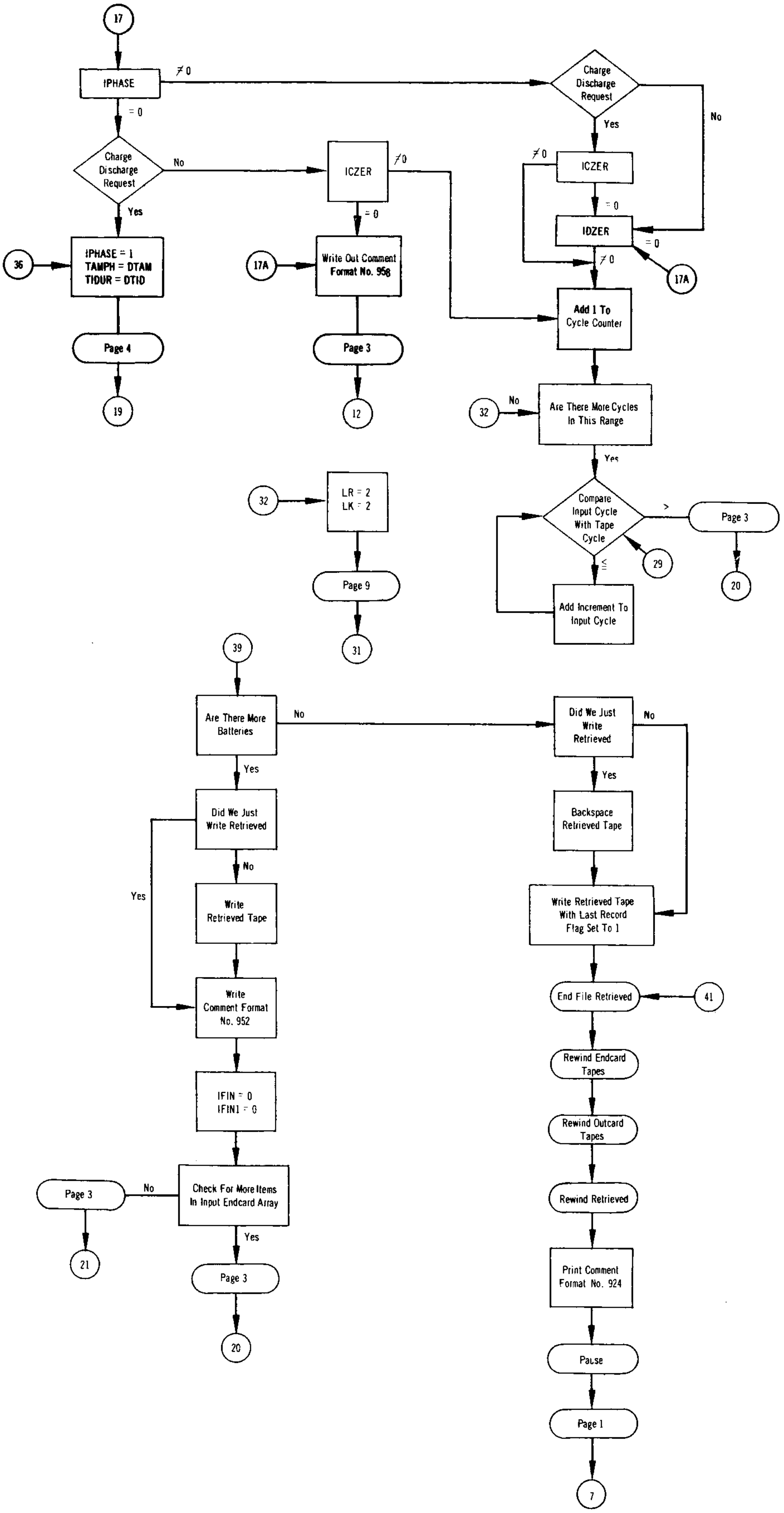
1a. Data Retrieve Formatter Routine Flow Charts (Continued)

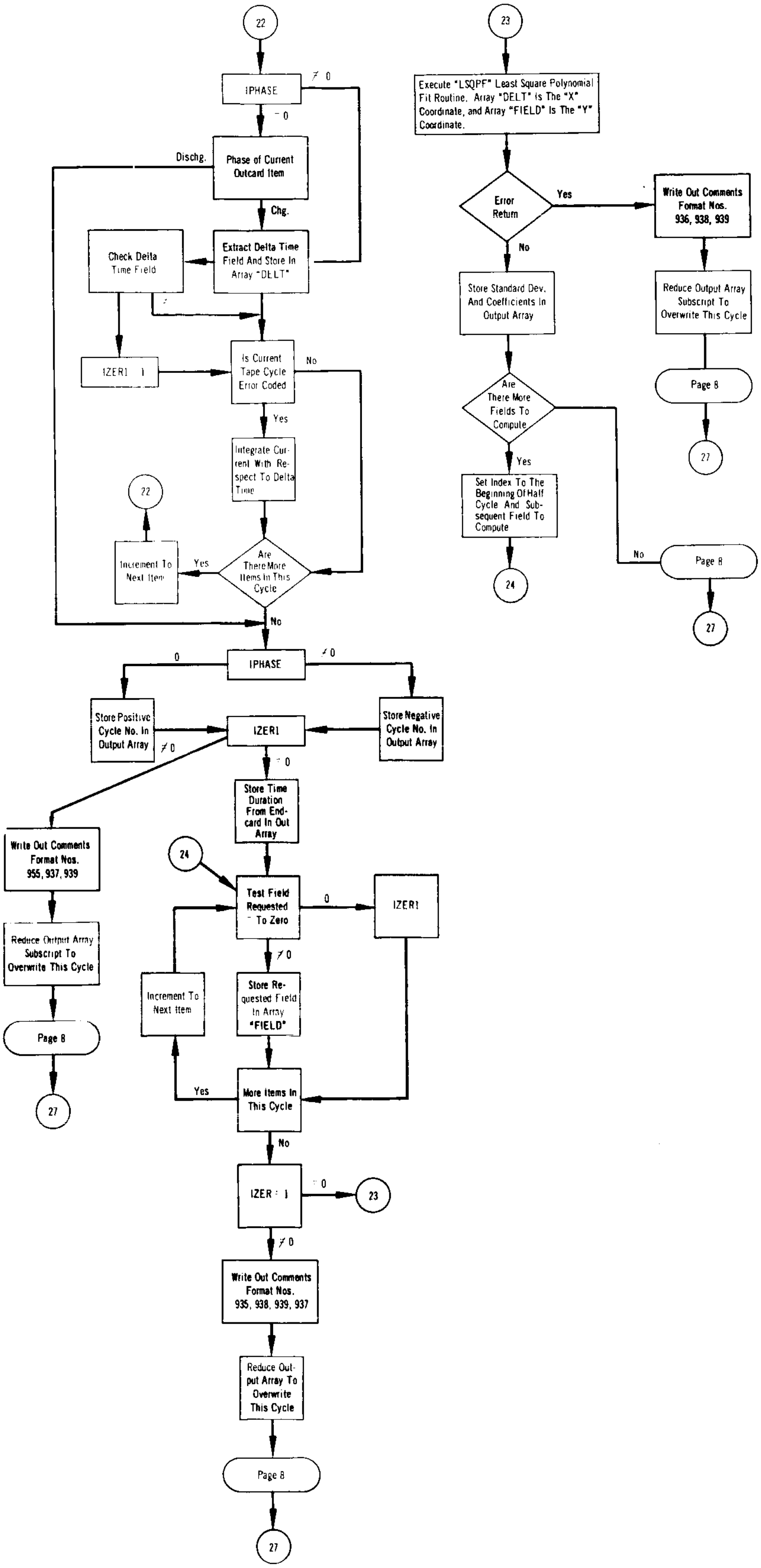

13
14
Compare Input Cycle With Tape Cycle
I < T
I > T
I = T
Page 3
Have We Exceeded Cycle Range
Yes
No
Write Out Comment Format No. 929
Write Out Comment Format No. 946
Valid Tape Cycle
No
Yes
Write Out Comment Format No. 931
Least Square Fit Request
No
Yes
Page 3
No
Invalid Data Request
Yes
IPHASE = 0
Page 6
Page 8
14
32
33
12
IPHASE = 1
TAMPH = DTAM
TIDUR = DTID
ICZER = 0
IDZER = 0
Yes
Requested Phase Discharge
No
Page 3
No
Add 1 To Cycle Counter
Check Current Tape Item For Phase
Dischg.
Write Out Comment Format No. 941
Charge Discharge Request
Are There More Items This Cycle
Chg.
Check Current Tape Item For Phase
Chg.
Page 6
No
Dischg.
Is Current Tape Cycle Error Coded
No
18
19
17
Write Out Comment Format No. 933
Yes
12
Compute Percentage Missing At The Beginning Of Cycle
Page 3
No
Charge Discharge Request
Yes
Write Out Comment Format No. 932
= 0
IPHASE
≠ 0
Write Out Comment Format No. 950
Page 5
No
Least Square Fit Request
Yes
18
Least Square Fit Requested
Yes
Page 7
15
Page 7
22
Page 5
22
15

1a. Data Retrieve Formatter Routine Flow Charts (Continued)

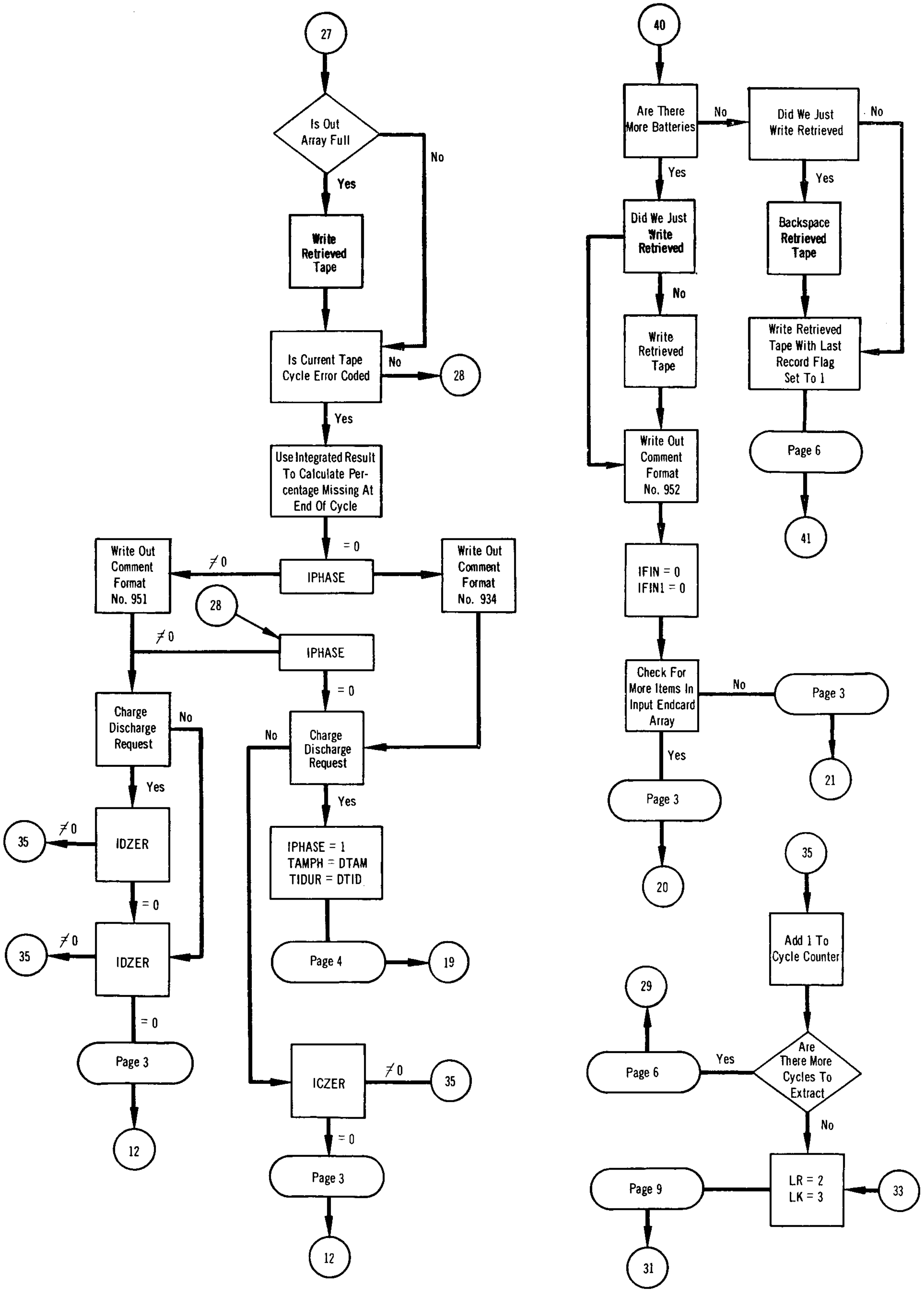

15
Check Which Half Of Cycle Requested
First Half
Compute Percentage Of The Cycle To Extract From The Beginning
Complete Cycle (100%)
Last Half
Compute Percentage Of The Cycle To Extract From The Beginning
IPHASE
≠0
= 0
Have Maximum No. Of Items Been Skipped
Yes
No
Check Phase Of Current Tape Item
Dischg.
16
Chg.
Increment to Next Item
IPHASE
≠0
Have Maximum Number Of Items Been Extracted
Yes
Page 6
17
No
= 0
Yes
Are There More Items In Cycle
Yes
Current Item Charge
Set Outcard Tape Cycle No. Negative And Store In Output Array
= 1
IPHASE
= 0
Set Outcard Tape Cycle No. Positive And Store In Output Array
No
No
Compute Percentage Missing At End Of Cycle
Extract Requested Field And Store In Output Array
Were Any Of The Fields Requested – 0
No
IPHASE
≠0
IPHASE
Yes
=0
≠0
= 0
Write Out Comment Format No. 954
ICZER = 1
IDZER = 1
Write Out Comment Format No. 951
Write Out Comment Format No. 934
Reduce Output Array Subscript To Overwrite This Cycle
No
Is Output Array Full
Yes
Write Out Comment Format No. 934
Are There More Items In This Cycle
Write Retrieval Tape
12
No
Page 3
No
Charge Discharge Request
36
16
Is Current Tape Cycle Error Coded
No
Page 6
Yes
Yes
17
IPHASE
Page 6
Compute Percentage Missing At End Of Cycle
= 0
ICZER
IPHASE
= 0
≠0
≠0
Write Out Comment Format No. 951
Page 6
Write Out Comment Format No. 934
Page 6
No
Least Square Fit Request
Yes
Page 8
17
42
35

1a. Data Retrieve Formatter Routine Flow Charts (Continued)

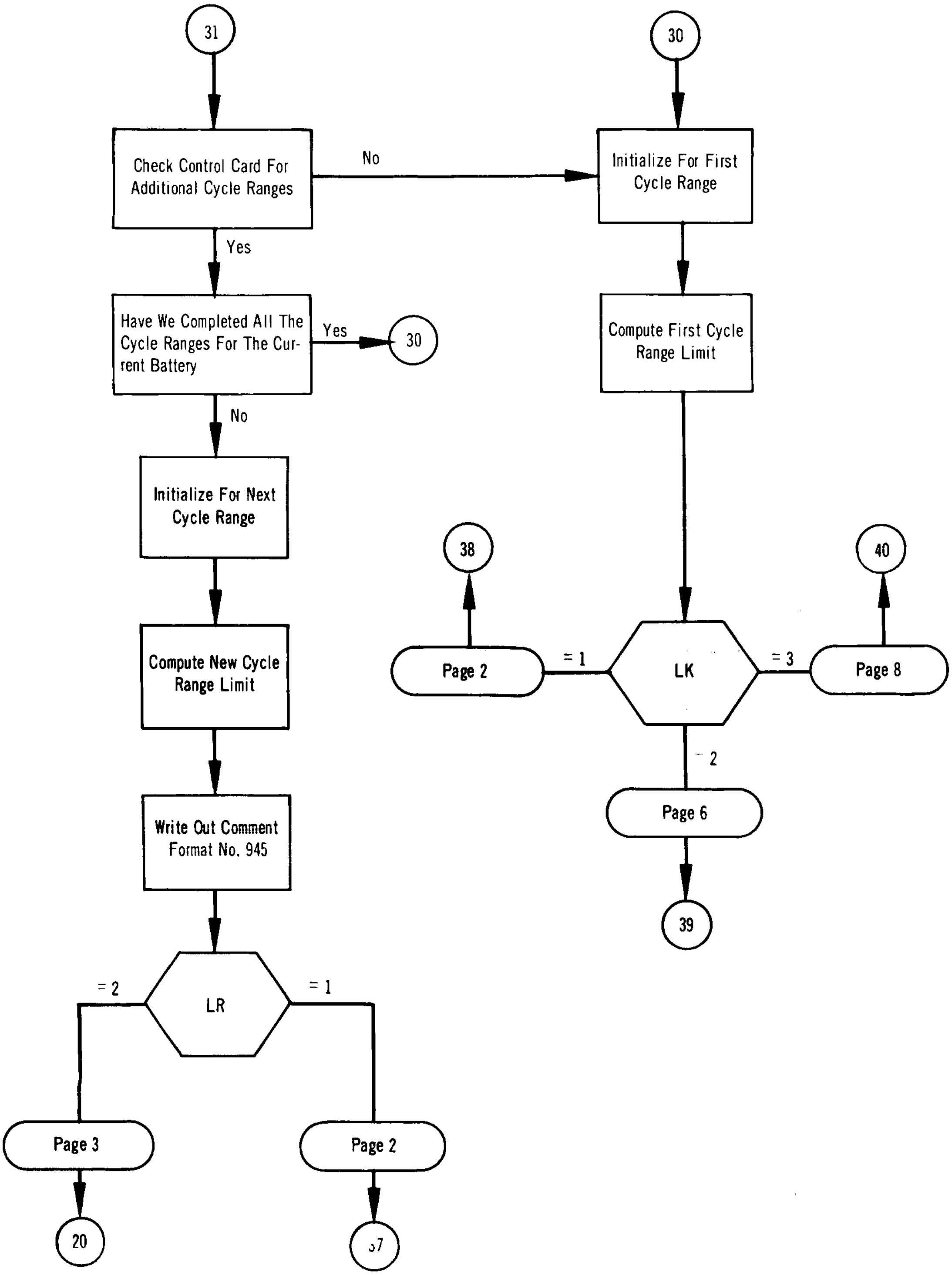



22
IPHASE
≠ 0
= 0
Dischg.
Phase of Current Outcard Item
Chg.
Check Delta Time Field
Extract Delta Time Field And Store In Array "DELT"
IZERI 1
Is Current Tape Cycle Error Coded
No
Yes
Integrate Current With Respect To Delta Time
22
Increment To Next Item
Yes
Are There More Items In This Cycle
No
IPHASE
0
≠ 0
Store Positive Cycle No. In Output Array
IZERI
Store Negative Cycle No. In Output Array
≠ 0
= 0
Write Out Comments Format Nos. 955, 937, 939
Reduce Output Array Subscript To Overwrite This Cycle
Page 8
27
Store Time Duration From Endcard In Out Array
24
Test Field Requested = To Zero
0
IZERI
≠ 0
Increment To Next Item
Store Requested Field In Array "FIELD"
Yes
More Items In This Cycle
No
IZER 1
= 0
23
≠ 0
Write Out Comments Format Nos. 935, 938, 939, 937
Reduce Output Array To Overwrite This Cycle
Page 8
27
23
Execute "LSQPF" Least Square Polynomial Fit Routine. Array "DELT" Is The "X" Coordinate, and Array "FIELD" Is The "Y" Coordinate.
Error Return
Yes
Write Out Comments Format Nos. 936, 938, 939
No
Store Standard Dev. And Coefficients In Output Array
Reduce Output Array Subscript To Overwrite This Cycle
Page 8
27
Are There More Fields To Compute
Yes
Set Index To The Beginning Of Half Cycle And Subsequent Field To Compute
24
No
Page 8
27

1a. Data Retrieve Formatter Routine Flow Charts (Continued)



1a. Data Retrieve Formatter Routine Flow Charts (Continued)


31
Check Control Card For Additional Cycle Ranges
No
Yes
Have We Completed All The Cycle Ranges For The Current Battery
Yes
30
No
Initialize For Next Cycle Range
Compute New Cycle Range Limit
Write Out Comment Format No. 945
LR
= 2
= 1
Page 3
20
Page 2
37
30
Initialize For First Cycle Range
Compute First Cycle Range Limit
38
40
Page 2
= 1
LK
= 3
Page 8
= 2
Page 6
39

1b. Data Retrieve Formatter Routine — Source Statements

```
$LOG     002101
*         BATTERY TEST RETRIEVAL PROGRAM
*        PAUSE
*        XEQ
*        BATTERY TEST DATA RETRIEVAL PROGRAM
*        PAUSE
*        SYMBOL TABLE
*        LIST8
*        CARDS COLUMN
*        FORMAP
*        LABEL
CDRP
C....         DATA  RETRIEVAL PROGRAM
         DIMENSION  IFIELD(30),FIELD(300),DELT(300),IAR(250),FAR(26,250),
        1IED(11,4),EED(52,4),IOD(9),FOD(16,520),COEF1(12),IBAT(15),PCH(50),
        1 ICD(43),FAQ(250),KFIELD(30),IRNG(19)
         AFLG = 6H$$$$$R
         A7 = 2HA7                                                                  1
         A8 = 2HA8                                                                  1
         B7 = 2HB7                                                                  1
         B8 = 2HB8                                                                  1
         IONE =1
         IZER = 0
         IN = 2                                                           A2
         IO = 3                                                           A3
         IOUT =18                                                         B8
         IOUT1 =17                                                        B7        1
         IEND =8                                                          A8
         ICC = 0
         IDD = 0
         IEND1 = 7                                                        A7        1
         IRETR = 19                                                       B9
         IREQ = 0
         IRNG(19)=0
         IEN=7
         IOU=7
C....     READ CONTROL CARD
6        READ INPUT TAPE IN,940,(ICD(I),I=1,43)
C....     CHECK LAST CARD
         IF(ICD(1)-9) 20,2,20
2        CALL EXIT
C        CHECK FOR ADDITIONAL CYCLE GROUPS
20       IF(ICD(40)) 501,500,501
501      READ INPUT TAPE IN,944,(IRNG(J8),J8=1,18)
         J8=1
500      K3=1
         K9 =0
         K10=0
         ILIM = ICD(14) + ICD(15) * (ICD(16)-1)
         ALIM=ILIM
         IFIN =0
         IFIN1 = 0                                                                  1
         ILST=0
         IF(ICD(37)) 22,7,22
7        READ INPUT TAPF IN,959,TAMPH,DIAM,TIDUR,DTID
22       IREQ = IREQ + 1
         ICHK = 0
         ICHK1 = 0                                                                  1
         ICHK2 = 0                                                                  1
         ICHK3 = 0                                                                  1
C....     WRITE CONTROL CARD OFF LINE AND ON  RETRIEVED DATA TAPE
C....     LOCATE BATTERYS
         K1= 0
         KCYC = 0
         DO 23 I= 2,13
         IF (ICD(I)) 24,23,24
 24      K1= K1 +1
         IBAT(K1)= I-2
 23      CONTINUE
C.....   LOCATE FIELDS REQUESTED
         K2 =0
         IF(ICD(42))3,4,3
 3       II1=21
         GO TO 5
 4       II1=20
 5       DO 33 I=II1,36
         IF(ICD(I)) 31,32,31
 31      K2 = K2 +1
         IFIELD(K2) = ICD(I)
 33      CONTINUE
 32      IF(K2)   40,38,40
C.....   ALL FIELDS ARE TO BE SELECTED
 38      IF(ICD(1)) 35,34,35
C.....   OUTCARD
 34      J=15
         GO TO 36
C.....   ENDCARD
 35      J = 26
 36      DO 37 K2 =1, J
         IFIELD(K2)= K2
 37      CONTINUE
         K2 = K2 -1
 40      IF(ICD(17) - 2)     85,78,85
 78      IF(ICD(1)) 1780,85,1780
 1780    INO=500/ (K2+K2+2)
         INO=INO+INO
         GO TO  86
 85      INO = 500 / (K2 + 1 )
 86      ICYC=ICD(14)
         WRITE OUTPUT TAPE  IO, 902, IREQ
         IF(ICD(1)) 42,41,42
 41      WRITE OUTPUT TAPE IO, 917
         GO TO 43
 42      WRITE OUTPUT TAPE IO, 916
 43      WRITE OUTPUT TAPE IO, 903,(IBAT(I),I=1,K1)
         WRITE OUTPUT TAPE IO, 904, ICYC
         WRITE OUTPUT TAPE IO, 905, ICD(15),ICD(16)
         IF(ICD(17))  45,44,45
 44      WRITE OUTPUT TAPE IO, 906
         GO TO 46
 45      IF(ICD(17)-2) 511,510,511
 510     WRITE OUTPUT TAPE IO,949
         GO TO 46
 511     WRITE OUTPUT TAPE IO,907
 46      IF (ICD(1))  54,47,54
 47      IF (ICD(38)-1)49,48,49
 48      WRITE OUTPUT TAPE IO, 908
         GO TO 54
 49      IF (ICD(18))   51,50,51
 50      WRITE OUTPUT TAPE IO, 911
         GO TO 54
 51      IF (ICD(18) - 2 )  53,52,53
 52      WRITE OUTPUT TAPE IO, 909, ICD(19)
         GO TO 54
 53      WRITE OUTPUT TAPE IO,910, ICD(19)
 54      IF(ICD(20))56,55,56
 55      WRITE OUTPUT TAPE IO, 918
         GO TO 57
 56      WRITE OUTPUT TAPE IO, 912,(IFIELD(I),I=1,K2)
```

1b. Data Retrieve Formatter Routine — Source Statements (Continued)

```
 57     IF(ICD(41)) 58,59,58
 58     WRITE OUTPUT TAPE IO, 913
 59     IF(ICD(42)) 60,61,60
 60     WRITE OUTPUT TAPE 10,914,ICD(20)
 61     WRITE OUTPUT TAPE 10,915,ICD(43
C.....  WRITE CARD IMAGE ON  RETRIEVED TAPE
        WRITE TAPE IRETR,AFLG,(ICD(I),I=1,43)
        IF(ICD(40)) 616,615,616
616     WRITE TAPE IRETR,(IRNG(I),I=1,18)
        I=1
        ICR = 2
617     WRITE OUTPUT TAPE IO, 956, ICR,IRNG(I),IRNG(I+1),IRNG(I+2)
        I=I+3
        ICR = ICR +1
        IF (IRNG(I)) 617,615,617
615     WRITE OUTPUT TAPE IO, 957
        IFND = 0
        IFND1 = 0
        IG1 = ICD(14)
        IG2 = ICD(15)
        IG3 = ICD(16)
C.....  CHANNEL CARD TYPE REQUESTS
        IF(ICD(1)) 10,200,10
C.....  ENDCARD
 10     IF(ICD(17))11,71,11
 11     IF(ICD(17)-1) 14,13,14
 13     DO 12 I=1,K2
        IFIELD(I) = IFIELD(I)+26
 12     CONTINUE
        GO TO 71
 14     DO 15 I=1,K2
        KFIELD(I) = IFIELD(I) + 26
 15     CONTINUE
 71     IF(IFIN) 561,542,561                                                   1
542     IF (IBAT(K3) -IEN) 75,75,543                                           1
 75     READ TAPE IEND, IFIN,ICNT,((IED(K,J),K=1,11),(FED(K,J),K=1,52),
       1 J=1, ICNT )
        IF(ICHK) 76,72,76                                                      1
543     READ TAPEIEND1,IFIN,ICNT,((IED(K,J),K=1,11),(FED(K,J),K=1,52),         1
       1J=1,ICNT)                                                              1
        IF(ICHK1)76,72,76                                                      1
 72     IF(IED(1)-1) 73,620,73                                                 1
 73     IF(IBAT(K3) - IEN) 579,579,580                                         1
579     PRINT 919,A8                                                           1
        GO TO 581                                                              1
580     PRINT 919,A7                                                           1
581     PAUSE                                                                  1
        GO TO  542                                                             1
620     IF(IBAT(K3) - IEN) 621,621,622                                         1
621     ICHK = 1                                                               1
        GO TO 76                                                               1
622     ICHK1 =1                                                               1
 76     DO 96 I=1, ICNT
        I = I
        IF(IBAT(K3) - IED(2,I)) 77,81,96
 77     IF(IBAT(K3) - IEN) 545,545,546                                         1
545     BACKSPACE IEND                                                         1
        GO TO 547                                                              1
546     BACKSPACE IEND1                                                        1
547     IF (IEND) 80,79,80                                                     1
 79     WRITE OUTPUT TAPE IO,920,IBAT(K3)
C.....  GO TO NEXT BATTERY LOGIC
        LK = 1
        GO TO 506
 80     WRITE OUTPUT TAPE IO,921,IED(2,I),ICYC,IBAT(K3)
        IFND = 0
```

```
C.... GO TO NEXT BATTERY LOGIC
561   LK =1
      GO TO 506
C.... BATT. LOCATED LETS FIND CYCLE
81    IFND = 1
      IF(ICYC - IED(3,I)) 82,92,96
C.... USE NEXT HIGHEST  CYCLE
82    IF(ILIM - IED(3,I)) 84,83,83
84    WRITE OUTPUT TAPE IO,946,KCYC
      IF(IBAT(K3) - IEN) 700,700,701
700   BACKSPACE IEND
      GO TO 111
701   BACKSPACE IEND1
      GO TO 111
83    WRITE OUTPUT TAPE IO,922,ICYC,IED(2,I),IED(3,I)
C.... CYCLE OR CLOSEST CYCLE FOUND
C....  CHECK VALIDITY OF CYCLE
92    IF (IED(7,I))    93, 94,93
C.... INVALID DATA  PRINT ITEM
93    WRITE OUTPUT TAPE IO, 923, IED(2,I)    ,IED(3,I)    ,IED(7,I)
     1 IED(8,I)
C.... SHOULD WE BYPASS INVALID DATA
      IF (ICD(41)) 94,96,94
C....SELECT REQUESTED FIELDS
94    K9 = K9 +1                                                    1
      IPH = 0                                                       1
      ICZER =0                                                      1
      IDZER =0
      IF(ICD(17) -1) 529,528,529
528   IAR(K9) = - IED(3,I)
      GO TO 530
529   IAR(K9) = IED(3,I)
C....  K2 = NO. OF FIELDS TO EXTRACT
530   DO 100 K12 =1,K2
      K8 =  IFIELD(K12)
      IF (FED(K8,I)) 583,582,583                                    1
582   IF(SIGNF(1.,FED(K8,I)))584,583,583
584   IF(ICZER) 681,751,681
751   ICZER =1
      IF(ICC-10) 680,681,680
680   WRITE OUTPUT TAPE IO, 953 , IAR(K9)                           1
      ICC =ICC+1
681   IF(ICD(39)) 682,583,682
682   K9 =K9-1                                                      1
      GO TO 585                                                     1
583   FAR(K12,K9) = FED(K8,I)
100   CONTINUE
      IF(ICZER) 585,683,585
683   ICC =0
C     CHECK CHARGE DISCHARGE REQUEST
585   IF(ICD(17)-2) 619,16,619                                      1
16    IF(K9 -INO) 586,98,586                                        1
586   K9 = K9+1                                                     1
      IAR(K9) = -IED(3,I)
      DO 17 K12 =1,K2
      K8 = KFIELD(K12)
      IF (FED(K8,I)) 589,588,589                                    1
588   IF(SIGNF(1.,FED(K8,I)))590,589,589
590   IF(IDZER) 686,750,686
750   IDZER =1
      IF(IDD-10) 685,686,685
685   WRITE OUTPUT TAPE IO, 953, IAR(K9)
      IDD =IDD+1
686   IF(ICD(39)) 687,589,687
687   K9 =K9-1
      GO TO 591                                                     1
```

1b. Data Retrieve Formatter Routine – Source Statements (Continued)

```
589    FAR(K12,K9) = FED(K8,I)
17     CONTINUE
       IF(IDZER) 591,689,591
689    IDD =0
       GO TO 591
C....   CHECK MORE CYCLES
619    IF(ICD(39)) 684,593,684
684    IF(ICZER) 96,593,96
591    IF(ICD(39)) 688,593,688
688    IF(ICZER) 592,593,592
592    IF(IDZER) 96,593,96                                                      1
593    KCYC = KCYC +1                                                           1
       IPH =1                                                                   1
       IF(KCYC - ICD(16)) 103,111,103
103    IF(ICYC - IED(3,I)) 116,116,1171
116    ICYC = ICYC + ICD(15)
       GO TO 103
111    LR = 1
       LK = 1
       GO TO 1
1171   IF(ICYC-ILIM) 117,117,1111
1111   WRITE OUTPUT TAPE 10,946,KCYC
       GO TO 111
117    IF(K9-INO) 96,98,96
98     K5 = K9 * (K2+1)
       WRITE TAPE IRETR,IZER,K5,IBAT(K3),(IAR(L2),(FAR(L1,L2)
      1, L1=1,K2),L2=1,K9)
       IF(IPH) 97,587,97                                                        1
587    K10 = K9                                                                 1
       K9 = 0                                                                   1
       GO TO 586                                                                1
C....  CHECK FOR MORE BATTERYS
104    KCYC = 0
       ICYC = ICD(14)
       IF(K3-K1) 95,106,95
95     K5 = K9 * (K2+1)
       IF(K9) 109,105,109
109    WRITE TAPE IRETR,IZER,K5,IBAT(K3),(IAR(L2),(FAR(L1,L2)
      1, L1=1,K2),L2=1,K9)
105    K3 = K3 + 1
       WRITE OUTPUT TAPE 10,952,IBAT(K3)
       IEND = 0
       ICC=0
       IDD=0
97     K10 = K9
       K9 =0
       IFIN=0
96     CONTINUE
       GO TO 71
C....   ALL DATA EXTRACTED FROM CURRENT CONTROL CARD
C....   WRITE LAST BLOCK.
106    IF(K9) 108,107,108
107    K9 = K10
       K3= K3-1
       BACKSPACE IRETR
108    K5 = K9 * (K2 +1)
       WRITE TAPE IRETR,IONE,K5,IBAT(K3),(IAR(L2),(FAR(L1,L2)
      1, L1=1,K2),L2=1,K9)
       END FILE IRETR
       REWIND IRETR
       REWIND IEND
       REWIND IEND1                                                             1
       PRINT 924,IREQ
       WRITE OUTPUT TAPE 10,943,IREQ
       PAUSE
C...   CHECK FOR ADDITIONAL CONTROL CARDS
```

```
      GO TO 6
C.... OUTCARD  REQUEST
200   FCYCLE = ICD(14)
      FINC   = ICD(15)
      PCENT  = ICD(19)
      LZ=2
      OCY = 0.
      PCENT = PCENT/100.
C.... CALCULATE THE NO. OF ITEMS THAT WILL FIT INTO A 500 WORD ARRAY
C.... LEAST SQ. FIT
      IF (ICD(38)-1)222,220,222
C.... YES
220   INO = 500/(K2*5+2)
      KWD = K2 * 5
      LC = XLOCF(COEF1(12))
      LZ=3
C.... BATTERY OR CELL CYCLE REQUEST
222   IF(ICD(42)) 190,206,190
C.... CELL CYCLE , LOCATE CELL FIELD
190   IF(ICD(20)) 191,193,191
191   IF(ICD(20) -10) 192,193,193
C.... CHARGE OR DISCHARGE
192   IF(ICD(17)-1) 239,197,239
239   ICELL = ICD(20)+15
      GO TO 206
197   ICELL = ICD(20)+41
      GO TO 206
193   WRITE OUTPUT TAPE IO,926
      PRINT 926
      REWIND IRETR
      REWIND IEND
      REWIND IEND1                                                    1
      GO TO 6
C.... CELL CYCLE
C.... SEARCH ENDCARD TAPE FOR BATTERY
206   IF(ICD(37)) 218,566,218
218   IF(IFIN) 562,549,562
549   IF(IBAT(K3) - IEN) 223,223,550                                  1
223   READ TAPE IEND,IFIN,ICNT,((IED(K,J),K=1,11),(FED(K,J),K=1,52),
     1 J=1, ICNT )
      IF(ICHK)566,551,566                                             1
550   READ TAPE IEND1,IFIN,ICNT,((IED(K,J),K=1,11),(FED(K,J),K=1,52),  1
     1 J=1, ICNT )                                                    1
      IF(ICHK1)566,551,566                                            1
551   IF(IED(1)-1) 568,623,568                                        1
568   IF(IBAT(K3)- IEN) 5760,5760,577
5760  PRINT 919,A8
      GO TO 578                                                       1
577   PRINT 919,A7                                                    1
578   PAUSE                                                           1
      GO TO 549                                                       1
623   IF(IBAT(K3) - IEN) 624,624,625                                  1
624   ICHK = 1                                                        1
      GO TO 566                                                       1
625   ICHK1 = 1                                                       1
566   DO 219 I=1,ICNT
      I=I
224   IF(ICD(37)) 704,195,704
704   IF(IBAT(K3)-IED(2,I)) 207,211,210
207   IF(IBAT(K3) - IEN) 555,555,556                                  1
555   BACKSPACE IEND                                                  1
      GO TO 557                                                       1
556   BACKSPACE IEND1                                                 1
557   IF(IFND) 208,209,208                                            1
208   IF(ICD(42)) 131,130,131
```

1b. Data Retrieve Formatter Routine – Source Statements (Continued)

```
 130   FCYCLE = ICYC
 131   WRITE OUTPUT TAPE IO,927,IED(2,I),FCYCLE,IBAT(K3)
 C.... GO FIND NEXT BATTERY
       IFND = 0
 562   LK = LZ                                                            1
       GO TO 506
 209   WRITE OUTPUT TAPE IO,920, IBAT(K3)
       LK=LZ
       GO TO 506
 211   IFND =1
       IF(ICD(42)) 214,215,214
 C.... BATTERY LOCATED.FIND CELL CYCLE
 214   IF(FCYCLE  -  FED(ICELL,I)) 212,213,210
 212   IF(ALIM-FED(ICELL,I))225,221,221
 225   WRITE OUTPUT TAPE IO,947
       IF (ICD(38)-1) 289,313,289
 221   WRITE OUTPUT TAPE IO, 928, FCYCLE , IBAT(K3),FED(ICELL,I)
 213   ICYC = IED(3,I)

 C.... LOCATE BATTERY CYCLE
 215   IF(ICYC - IED(3,I)) 216,217,210
 216   WRITE OUTPUT TAPE IO,930,ICYC,IBAT(K3),IED(3,I)
 C....CHARGE
 217   TAMPH = FED(5,I)
       TIDUR = FED(1,I)
 C....
       DIAM = FED(31,I)
       DTID = FED(27,I)
 C.... FIRST LOCATE BATTERY
 195   IF(IFIN1) 562,552,562                                              1
 552   IF(IBAT(K3) -IOU) 196,196,553                                      1
 196   READ TAPEIOUT,IFIN1,ICNT,(IOD(K),K=1,9),((FOD(K,J),K=1,16),J=1,
      1ICNT)
       IF(ICHK2) 576,554,576                                              1
 553   READ TAPEIOUT1,IFIN1,ICNT,(IOD(K),K=1,9),((FOD(K,J),K=1,16),J=1,   1
      1ICNT)                                                              1
       IF(ICHK3) 576,554,576                                              1
 554   IF(IOD(1)) 572,626,572                                             1
 572   IF(IBAT(K3) -IOU) 573,573,574                                      1
 573   PRINT 919,B8                                                       1
       GO TO 575                                                          1
 574   PRINT 919,B7                                                       1
 575   PAUSE                                                              1
       GO TO 552                                                          1
 626   IF(IBAT(K3) - IOU) 627,627,628                                     1
 627   ICHK2 = 1                                                          1
       GO TO 576                                                          1
 628   ICHK3 = 1                                                          1
 576   DAMPH = 0.                                                         1
       IF (IBAT(K3) -IOD(2 )) 202,205,195
 202   IF(IBAT(K3)- IOU) 558,558,559                                      1
 558   BACKSPACE IOUT                                                     1
       GO TO 560                                                          1
 559   BACKSPACE IOUT1                                                    1
 560   IF(IFND1) 204,203,204                                              1
 203   WRITE OUTPUT TAPE IO, 925, IBAT(K3)
 C.... GET NEXT BATTERY
       LK=LZ
```

1b. Data Retrieve Formatter Routine – Source Statements (Continued)

```
        GO TO 506
 204    WRITE OUTPUT TAPE IO,921,IOD(2),ICYC,IBAT(K3)
        LK=LZ
        GO TO 506
 205    IFND1 = 1
 C....  FIND BATTERY CYCLE
        IF (ICYC - IOD(3)) 230,231,195
 230    IF(ILIM-IOD(3)) 2510,2501,2501
 2510   WRITE OUTPUT TAPE IO,946,KCYC
        IF(IBAT(K3) - IOU) 702,702,703
 702    BACKSPACE IOUT
        IF (ICD(38)-1) 289,313,289
 703    BACKSPACE IOUT1
        IF (ICD(38)-1) 289,313,289
 2501   WRITE OUTPUT TAPE IO,929,ICYC,IBAT(K3),IOD(3)
        OCY=IOD(3)
 C....  CHECK VALIDITY OF CYCLE
 231    IDIF = IOD(3) - IOD(9)
        IF(XABSF(IDIF) -2) 235,235,537
 C....  INVALID CYCLE
 235    WRITE OUTPUT TAPE IO,931, IBAT(K3),IOD(3)
 C....  CHECK INVALID DATA REQUEST
        IF(ICD(41)) 537,195,537
 C....  LOCATE HALF CYCLE
 537    IPHASE = 0
        IF(ICD(17)-1) 520,519,520
 519    IPHASE = 1
        TAMPH = DTAM
        TIDUR = DTID
 520    L = 1
        J7 = 1
        ICZER = 0                                                          1
        IDZER = 0                                                          1
        IF(IPHASE) 232,250,232
 C....  DISCHARGE. LOCATE CYCLE
 232    DO 234 L = J7,ICNT
        L =L
        IF(FOD(1,L)) 236,234,236
 234    CONTINUE
        WRITE OUTPUT TAPE IO,933,IOD(3)
        IF(ICD(17)-2) 217,541,217
 541    IF(ICD(38)-1) 601,611,601
 C....  CHARGE PHASE. MAKE SURE FIRST ITEM IS CHARGE
 250    IF(FOD(1,L)) 251,236,251
 251    WRITE OUTPUT TAPE IO,941,IOD(3)
        IF(ICD(17)-2)217,523,217
 C....  CHECK FOR CODE
 236    IF(IOD(7)) 237,241,237
 237    PER = FOD(3,L) * FOD(5,L) * 100. / TAMPH
        IF(IPHASF) 513,512,513
 512    WRITE OUTPUT TAPE IO,932,PER,IOD(3),IBAT(K3),IOD(7)
        GO TO 241
 513    WRITE OUTPUT TAPF IO,950,PER,IOD(3),IBAT(K3),IOD(7)
 C....  LFAST SQUARE FIT
 241    IF(ICD(38)-1) 245,298,245
 298    IF(ICD(18)-1) 245,299,245
 C....  CHFCK FIRST HALF OF CYCLE
 245    LL = L
```

1b. Data Retrieve Formatter Routine — Source Statements (Continued)

```
      IF(ICD(18) -2) 247,246,247
246   TAMPH = TAMPH * PCENT
      GO TO 260
C.... CHECK LAST HALF OF CYCLE
247   IF(ICD(18) - 3)   260,248,260
248   BAMPH = TAMPH - TAMPH * PCENT
      ILST=1
C.... SKIP TO LAST PORTION OF CYCLE REQUESTED
249   DO 255  LL = L , ICNT
      LL = LL
      IF(DAMPH) 401,400,401
400   DAMPH = FOD(3,LL) * FOD(5,LL)
      GO TO 402
401   DAMPH = DAMPH +(FOD(3,LL) - FOD(3,LL-1))* (FOD(5,LL)+FOD(5,LL-1))
     1 / 2.
402   IF(BAMPH-DAMPH) 260,260,256
C.... CHG.OR DISCHARGE
256   IF(IPHASE) 255,257,255
257   IF(FOD(1,LL)) 258,255,258
258   J7 =LL-1
      GO TO 259
255   CONTINUE
C....  PERCENT OF CYCLE REQUESTED NOT ON TAPE
259   PER = 100. -DAMPH*100./TAMPH
      IF(IPHASE)515,514,515
514   WRITE OUTPUT TAPE IO,934,PER,IOD(3),IBAT(K3),IOD(7)
516   WRITE OUTPUT TAPE IO,942
      IF(ICD(17)-2) 217,524,217
524   IF(IPHASE)5420,523,5420
5420  IF(ICZFR) 541,217,541
515   WRITE OUTPUT TAPE IO,951,PER,IOD(3),IBAT(K3),IOD(7)
      GO TO 516
C.....SELECT REQUESTED FIELDS
260   IF(ICD(38)-1) 261,299,261
261   DO 275 L=LL,ICNT
      L   =L
C.... CHARGE CYCLE
      IF(IPHASE) 263,262,263
C.... MAKE SURE WERE NOT IN DISCHARGE PHASE.
262   IF(FOD(1,L ))  268,263,268
268   J7 = L-1
      GO TO 276
263   IF(ILST) 413,414,413
413   ILST=0
      GO TO 412
414   IF(DAMPH) 411,410,411
410   DAMPH = FOD(3,L) * FOD(5,L)
      GO TO 412
411   DAMPH = DAMPH+(FOD(3,L)-FOD(3,L-1))*(FOD(5,L)+FOD(5,L-1))/2.
412   IF(TAMPH- DAMPH) 280,264,264
264   K9 = K9 +1
      IF(IPHASE) 525,526,525
525   IAR(K9) = - IOD(3)
      GO TO 527
526   IAR(K9) = IOD(3)
527   ISK =0
C.... K2 EQUALS NO. OF FIELDS TO EXTRACT
      DO 265 K12 =1,K2
```

1b. Data Retrieve Formatter Routine — Source Statements (Continued)

```
      K8 = IFIELD(K12) +1
      IF (FOD(K8,L)) 596,594,596
594   IF(SIGNF(1.,FOD(K8,L)))595,596,596
595   IF(ISK) 691,752,691
752   ISK =1
      IF(IPHASE) 755,753,755
753   IF(ICC-10) 690,691,690
690   ICC =ICC+1
      GO TO 754
755   IF(IDD-10) 756,691,756
756   IDD =IDD+1
754   WRITE OUTPUT TAPE IO,954,IAR(K9),FOD(3,L)
691   IF(ICD(39)) 692,596,692
692   K9 =K9-1
      GO TO  275                                                 1
596   FAR(K12,K9) = FOD(K8,L)                                    1
265   CONTINUE
C..... CHECK TO SEE IF ARRAY IS READY TO WRITE
      IF(IPHASE) 598,597,598                                     1
597   ICZER = 1
266   IF(ISK) 599,757,599
757   ICC =0
      GO TO 599                                                  1
598   IDZER = 1                                                  1
      IF(ISK) 599,758,599
758   IDD =0
599   IF(INO -K9) 275,267,275                                    1
267   K5 = K9 * (K2+1)
      WRITE TAPE IRETR,IZER,K5,IBAT(K3),(IAR(L2),(FAR (L1,L2
     1),L1=1,K2),L2=1,K9)
      K10 = K9
      K9 = 0
275   CONTINUE
      J7 = L-1
C..... CHECK IF CYCLE IS CODED
276   IF(IOD(7))277,280,277
C.... COMPUTE PERCENTAGE MISSING AT END OF CYCLE
277   PER = 100. - DAMPH * 100. / TAMPH
      IF(IPHASE) 518,517,518
517   WRITE OUTPUT TAPE IO,934,PER,IOD(3),IBAT(K3),IOD(7)
      GO TO 280
518   WRITE OUTPUT TAPE IO,951,PER,IOD(3),IBAT(K3),IOD(7)
C.... CHECK FOR MORE CYCLES WITHIN THIS BATTERY
280   IF(IPHASE) 600,522,600                                     1
522   IF(ICD(17)-2) 521,523,521
523   IPHASE =1
      TAMPH = DTAM
      TIDUR = DTID
      DAMPH=0.
      GO TO 232
521   IF(ICZER) 603,629,603                                      1
629   IF(ICD(39)) 693,603,693
693   WRITE OUTPUT TAPE IO,958,IOD(3)
      GO TO 217                                                  1
600   IF(ICD(39)) 694,603,694
694   IF(ICD(17)-2) 602,601,602
601   IF(ICZER) 603,602,603                                      1
602   IF(IDZER) 603,629,603                                      1
```

1b. Data Retrieve Formatter Routine — Source Statements (Continued)

```
 603    KCYC = KCYC +1                                                       1
        IF(KCYC - ICD(16)) 281,289,281
 281    IF(ICD(42)) 120,123,120
 120    IF(OCY) 126,125,126
 126    IF(FCYCLE-OCY) 121,121,127
 125    IF(FCYCLE -FED(ICELL,I)) 121,121,1270
 121    FCYCLE = FCYCLE + FINC
        GO TO 120
 127    OCY=0.
 1270   IF(FCYCLE -ALIM) 224,224,1271
 1271   WRITE OUTPUT TAPE IO,946,KCYC
        GO TO 289
 123    IF(ICYC - IOD(3)) 124,124,1240
 124    ICYC = ICYC + ICD(15)
        GO TO 123
 1240   IF(ICYC- ILIM) 224,224,1271
 289    LR =2
        LK =2
        GO TO 1
 282    KCYC = 0
        FCYCLE = ICD(14)
        ICYC = ICD(14)
 C....  CHECK FOR NEXT BATTERY
        IF(K3 - K1) 286,283,286
 C....  DID WE JUST WRITE
 286    IF(K9) 287,288,287
 C....  NO.DUMP BUFFER
 287    K5 = K9 * (K2+1)
        WRITE TAPE IRETR,IZER,K5,IBAT(K3),(IAR(L2),(FAR (L1,L2)
       1,L1=1,K2),L2=1,K9)
        K10 = K9
        K9 = 0
 288    K3 = K3+1
        IFND = 0
        IFND1 = 0
        WRITE OUTPUT TAPE IO,952,IBAT(K3)
        ICC=0
        IDD=0
        IFIN = 0                                                             1
        IFIN1 = 0                                                            1
        GO TO 210
 283    IF(K9) 285,284,285
 284    BACKSPACE IRETR
        K3= K3-1
        K9 = K10
 285    K5 = K9 * (K2+1)
        WRITE TAPE IRETR,IONE,K5,IBAT(K3),(IAR(L2),(FAR (L1,
       1L2),L1=1,K2),L2=1,K9)
 564    END FILE IRETR                                                       1
        REWIND IRETR
        REWIND IEND
        REWIND IEND1                                                         1
        REWIND IOUT
        REWIND IOUT1                                                         1
        WRITE OUTPUT TAPE IO,943,IREQ
        PRINT 924,IREQ
        PAUSE
        GO TO 6
```

1b. Data Retrieve Formatter Routine — Source Statements (Continued)

```
 C...... LEAST SQUARE FIT ON FIELDS REQUESTED
 C...... STORE  DELTA TIME ARRAY
 299    L=LL
 300    K12 =0
        IZER1 = 0                                                        1
 C....   L =  ITEM WITHIN ARRAY
        DO 305 LL=L , ICNT
 C....  CHARGE OR DISCHARGE
        IF(IPHASE) 302,301,302
 C....  CHARGE . TEST IF WE ARE AT DISCHARGE
 301    IF(FOD(1,LL)) 531,302,531
 531    J7 = LL-1
        GO TO 303
 302    K12 = K12 +1
        DELT(K12) =  FOD(3,LL)
        IF(DELT(K12)) 650,604,650
 604    IZER1 = 1                                                        1
 650    IF(ILST) 651,420,651
 651    ILST=0
        GO TO 660
 420    IF(DAMPH) 422,421,422
 421    DAMPH = FOD(3,LL) * FOD(5,LL)
        GO TO 305
 422    DAMPH = DAMPH+(FOD(3,LL)-FOD(3,LL-1))*(FOD(5,LL)+FOD(5,LL-1))/2.
 660    IF(DAMPH-DAMPH) 342,305,305
 305    CONTINUE
 342    J7=LL-1
 C...... FIND FIELD 1 STORE IN ARRAY AND XEQ  LSQPF .THEN REPEAT FOR ALL
 C...... FIELDS
 303    K9 = K9+1
        JC=0
        IF (IPHASE) 532,538,532
 532    IAR(K9)= -IOD(3)
        GO TO 533
 538    IAR(K9)=IOD(3)
 533    IF(IZER1) 606,614,606                                            1
 606    WRITE OUTPUT TAPE 10,955, IAR(K9)                                1
        GO TO 331                                                        1
 614    FAQ(K9) = TIDUR                                                  1
        K13 = K12 + L -1
        LX = XLOCF(DELT(1)) - K12 + 1
        LY = XLOCF(FIELD(1)) - K12 +1
        DO 315 K14 =1, K2
        K15 = 0
        K8 =  IFIELD(K14) +1
        DO 325  LL =L , K13
        IF(FOD(K8,LL)) 332,306,332
 306    IZER1 = 1
 332    K15=K15 +1
        FIELD(K15) = FOD(K8,LL)
 325    CONTINUE
        IF(IZER1) 326,324,326
 324    TEST = LSQPF(LX,LY,K12,3,LC )
        IF(TEST)    329,307,329
 307    DO 308 J=1,5
        JC=JC+1
        FAR(JC,K9) = COEF1(J)
 308    CONTINUE
```

1b. Data Retrieve Formatter Routine — Source Statements (Continued)

```
 315  CONTINUE
      IF(IPHASE) 608,607,608                                                    1
C.... A FIELD CONTAINED ZERO IN SETTING UP ARRAY FOR LEAST SQ.FIT
 326  WRITE OUTPUT TAPE IO,935,IBAT(K3),IOD(3),IFIELD(K14)
      GO TO 630                                                                 1
 329  WRITE OUTPUT TAPE IO,936,IFIELD(K14),IBAT(K3),ICYCLE
 630  WRITE OUTPUT TAPE IO,938,IFIELD(K14)                                      1
      WRITE OUTPUT TAPE IO,939,(FIELD(J),J=1,K15 )
 331  WRITE OUTPUT TAPE IO,937
      WRITE OUTPUT TAPE IO, 939,(DELT (J),J=1,K12 )
      K9 = K9-1                                                                 1
      GO TO 609                                                                 1
 607  ICZER = 1                                                                 1
      GO TO 609                                                                 1
 608  IDZER = 1                                                                 1
C..... CHECK TO SEE IF BUFFER IS FULL
 609  IF(K9-INO) 320,309,320                                                    1
 309  K5 = K9 * (KWD + 2)
      WRITE TAPE IRETR,IZER,K5,IBAT(K3),(IAR(L2),FAQ(L2),(FAR(L1,L2),
     1 L1=1,KWD ),L2=1,K9)
      K10 = K9
      K9 =0
C.... IS CYCLE CODED
 320  IF(IOD(7)) 335,311,335
C.... COMPUTE PERCENT AT END OF CYCLE
 335  PER=100.-DAMPH*100. / TAMPH
      IF (IPHASE) 540,539,540
 539  WRITE OUTPUT TAPE IO ,934,PER,IOD(3),IBAT(K3),IOD(7)
      GO TO 534                                                                 1
 540  WRITE OUTPUT TAPE IO,951,PER,IOD(3),IBAT(K3),IOD(7)
      GO TO 610                                                                 1
C.... CHECK FOR WORE CYCLES WITHIN THIS BATTERY
 311  IF(IPHASE) 610,534,610                                                    1
 534  IF(ICD(17) -2) 536,535,536
 535  IPHASE =1
      TAMPH = DTAM
      TIDUR = DTID
      DAMPH=0.
      GO TO 232
 536  IF(ICZER) 613,217,613                                                     1
 610  IF(ICD(17)-2) 612,611,612                                                 1
 611  IF(ICZER) 613,612,613                                                     1
 612  IF(IDZER) 613,217,613                                                     1
 613  KCYC = KCYC+1                                                             1
      IF(KCYC - ICD(16)) 281,313,281
 313  LR = 2
      LK = 3
      GO TO 1
 322  KCYC = 0
      FCYCLE = ICD(14)
      ICYC = ICD(14)
C.... CHECK FOR NEXT BATTERY
      IF(K3-K1) 314,317,314
C.... DID WE JUST WRITE
 314  IF(K9)316,321,316
C.... NO DUMP BUFFER
 316  K5 = K9 * (KWD + 2)
      WRITE TAPE IRETR,IZER,K5,IBAT(K3),(IAR(L2),FAQ(L2),(FAR(L1,L2),
```

1b. Data Retrieve Formatter Routine – Source Statements (Continued)

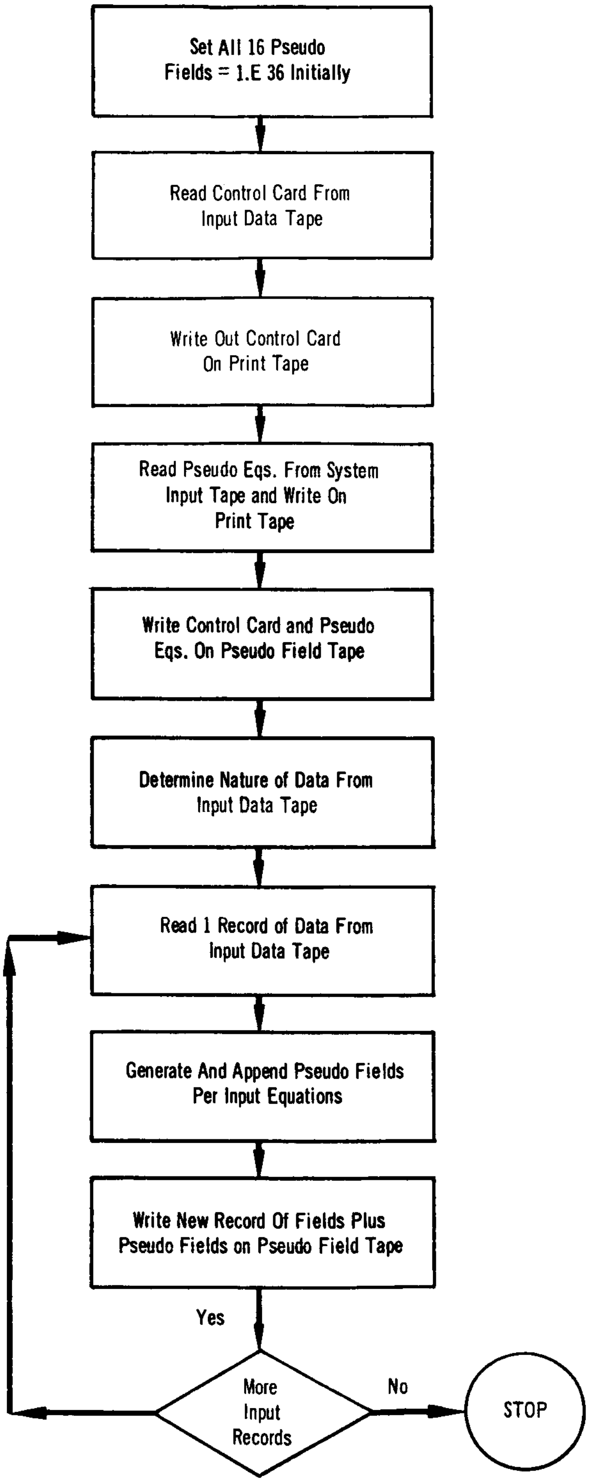
```
       1 L1 =1,KWD),L2 = 1,K9)
        K10 = K9
        K9 = 0
 321    K3=K3+1
        IFND = 0
        IFND1 = 0
        WRITE OUTPUT TAPE IO,952,IBAT(K3)
        IFIN = 0                                                    1
        IFIN1 = 0                                                   1
        GO TO 210
 317    IF(K9)  319,318,319
 318    BACKSPACE IRETR
        K9 = K10
 319    K5 = K9 * (KWD + 2)
        WRITE TAPE IRETR,IONE,K5,IBAT(K3),(IAR(L2),EAQ(L2),(EAR(L1,L
       12),L1=1,KWD),L2=1,K9)
        GO TO 564                                                   1
 1      IF(ICD(40))504,506,504
 504    IF (IRNG(J8)) 505,506,505
 505    ICD(14) = IRNG(J8)
        FCYCLE  = IRNG(J8)
        ICD(15) = IRNG(J8+1)
        FINC =  IRNG(J8+1)
        ICD(16) = IRNG(J8+2)
        J8 =J8+3
        KCYC = 0
        ICYC=ICD(14)
        WRITE OUTPUT TAPE IO, 945 , ICD(14),ICD(15),ICD(16)
        ILIM = ICD(14) + ICD(15) * (ICD(16)-1)
        ALIM=ILIM
        GO TO  ( 117,224),LR
 506    ICD(14) = IG1
        ICD(15) = IG2
        FINC = IG2
        ICD(16) = IG3
        ILIM = ICD(14) + ICD(15) * (ICD(16)-1)
        ALIM=ILIM
        J8 =1
        GO TO  ( 104,282,322),LK
 210    IF(ICD(37)) 219,195,219
 219    CONTINUE
        GO TO 206
 902    FORMAT  (1H120X,32HDATA RETRIEVAL RUN. REQUEST NO. I2)
 903    FORMAT  (1H05X,23HBATTERYS REQUESTED ARE 12I4)
 904    FORMAT (6X,21HSTARTING CYCLE NO IS I4)
 905    FORMAT (6X,13HINCREMENT IS I3,22H NUMBER OF INCREMENTS I4)
 906    FORMAT (6X,12HCHARGE PHASE)
 907    FORMAT (6X,15HDISCHARGE PHASE)
 908    FORMAT (6X,36HLEAST SQUARE FIT ON REQUESTED FIELDS)
 909    FORMAT  (6X,6HFIRST I2,27H PERCENT OF CYCLE REQUESTED)
 910    FORMAT (6X,5HLAST I2,27H PERCENT OF CYCLE REQUESTED)
 911    FORMAT (6X,31H100 PER CENT OF CYCLE REQUESTED)
 912    FORMAT (6X,16HFIELDS REQUESTED22I4)
 913    FORMAT (6X,22HINVALID DATA REQUESTED)
 914    FORMAT (6X,26HCELL CYCLE REQUEST ON CELLI5)
 915    FORMAT (6X,11HEXPIRED DAYI4)                                1
 916    FORMAT (6X,22HENDCARD DATA REQUESTED)
 917    FORMAT (6X,22HOUTCARD DATA REQUESTED)
```

1b. Data Retrieve Formatter Routine — Source Statements (Continued)

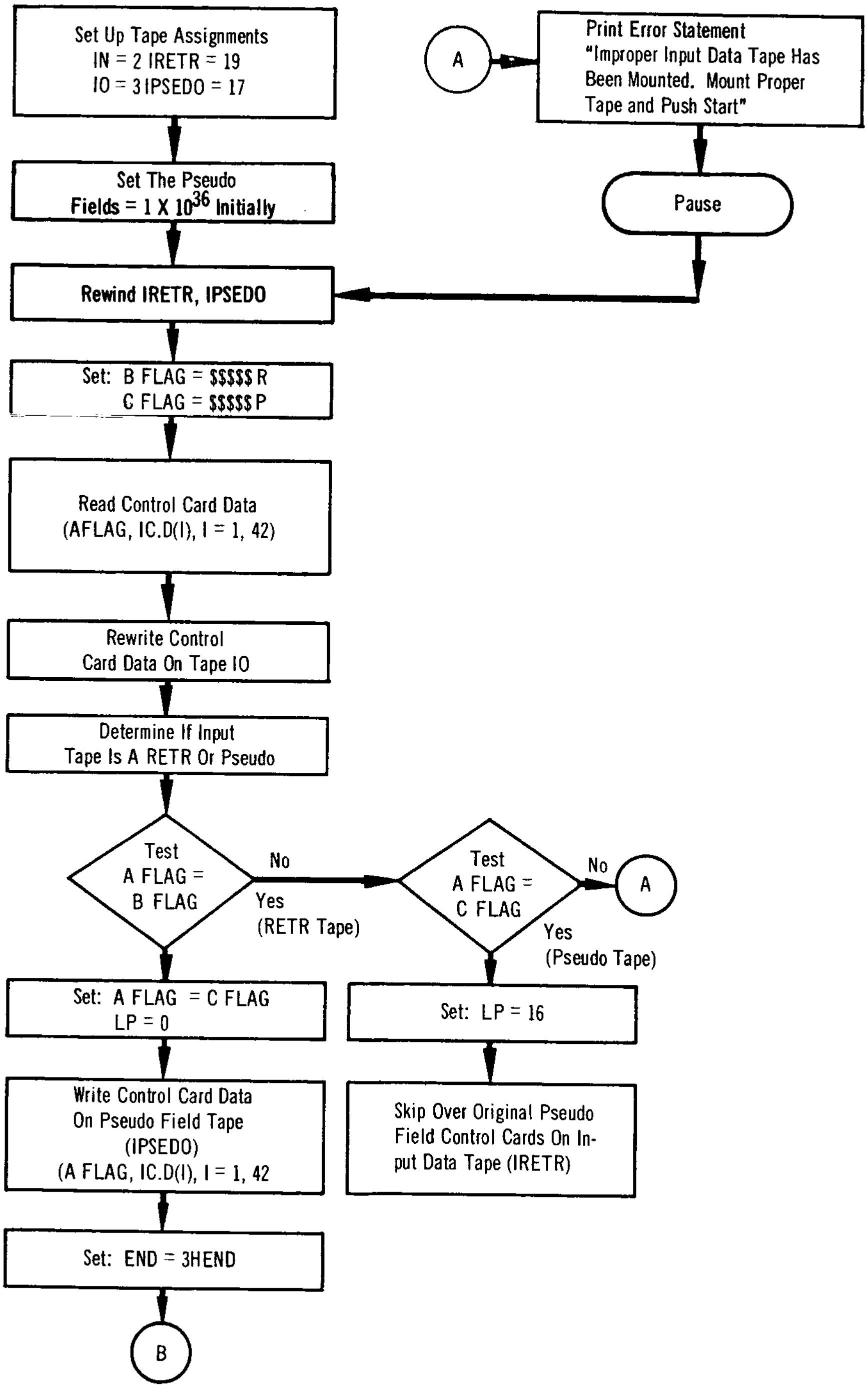
```
 918   FORMAT (6X,20HALL FIELDS REQUESTED)
 919   FORMAT (6X,18HINCORRECT TAPE ON A2,25H. MOUNT CORRECT TAPE AND PUS      1
      1H START)                                                                  1
 920   FORMAT (1H06X, 7HBATTERYI4,23H IS NOT ON ENDCARD TAPE)
 921   FORMAT (1H06X,21HWE ENCOUNTERD BATTERYI4,18H LOOKING FOR CYCLEI5
      1,11H OF BATTERYI4)
 922   FORMAT (1H06X,20HCOULD NOT FIND CYCLEI5,11H OF BATTERYI4,13H .USIN
      1G CYCLEI5)
 923   FORMAT (1H06X,21HINVALID DATA. BATTERYI4, 6H CYCLEI5. 5H CODEI4.
      1 5H TIMEI5)
 924   FORMAT (6X,76HPLEASE REMOVE B9 AND LABEL. MOUNT NEW B9 AND PUSH ST
      1ART TO CONTINUE. REQUESTI5,13H IS COMPLETED)
 925   FORMAT (1H06X, 7HBATTERYI4,23H IS NOT ON OUTCARD TAPE)
 926   FORMAT (1H16X,94HCELL CYCLE REQUEST BUT CANNOT LOCATE CELL CYCLE F
      1IELD ON CONTROL CARD. SKIPING TO NEXT REQUEST)
 927   FORMAT (1H06X,21HWE ENCOUNTERD BATTERYI4,23H LOOKING FOR CELL CYCL
      1EF6.0,11H OF BATTERYI4,16H OF ENDCARD TAPE)
 928   FORMAT (1H06X,25HCOULD NOT FIND CELL CYCLEF6.0,11H OF BATTERYI4,18
      1H .USING CELL CYCLEF6.0)
 929   FORMAT (1H06X,28HCOULD NOT FIND BATTERY CYCLEI5,  11H OF BATTERYI4
      1,29H OF OUTCARD TAPE. USING CYCLEI5)
 930   FORMAT (1H06X,28HCOULD NOT FIND BATTERY CYCLEI5,  11H OF BATTERYI4
      1,29H OF ENDCARD TAPE. USING CYCLEI5)
 931   FORMAT (1H06X, 7HBATTERYI4, 6H CYCLEI5,24H OF OUTCARD TAPE INVALID
      1)
 932   FORMAT (1H06X,F5.1,34H PERCENT AT THE BEGINNING OF CYCLEI5,21H IS
      1MISSING . BATTERYI4,14H ERROR CODE ISI4,13H.CHARGE PHASE)
 933   FORMAT (1H06X,43HCOULD NOT FIND THE DISCHARGE PHASE OF CYCLEI5)
 934   FORMAT (1H06X,F5.1,28H PERCENT AT THE END OF CYCLEI5,20H IS MISSIN
      1G. BATTERYI4,14H ERROR CODE ISI4,13H.CHARGE PHASE)
 935   FORMAT (1H06X, 7HBATTERYI4,4H CYCI5    ,23H OF OUTCARD TAPE. FIELDI
      14,39H CONTAINS ZERO. DID NOT COMPUTE L.S.FIT)
 936   FORMAT (1H06X,52HTHE L.S.FIT ROUTINE CAN'T FIT A POLYNOMIAL FOR FI
      1ELDI4, 8H BATTERYI4, 6H CYCLEI5)
 937   FORMAT (1H06X,24HDELTA TIME VALUES FOLLOW)
 938   FORMAT (1H06X, 5HFIELDI4,14H VALUES FOLLOW)
 939   FORMAT (3X,8E15.6)
 940   FORMAT  (13I1,I4,I3,I4,I1,I1,I2,18I2,2I1,3I1,I3)
 941   FORMAT (1H06X,40HCOULD NOT FIND THE CHARGE PHASE OF CYCLEI5)
 942   FORMAT (1H06X,60HPERCENT MISSING GREATER THAN PERCENT REQUESTED.CY
      1CLE SKIPPED)
 943   FORMAT (28H0   ....... RETRIEVAL REQUESTI3,18H COMPLETED .......)
 944   FORMAT   (6(I4,I3,I4))
 945   FORMAT (/////7X,36H**** NEW CYCLE GROUP. STARTING CYCLEI5,10H INCR
      1EMENTI5,21H NUMBER OF INCREMENTSI5)
 946   FORMAT (1H06X,27H**** CYCLE RANGE EXCEEDED. I5,17H CYCLES EXTRACTE
      1D)
 947   FORMAT (6X,50HCYCLE RANGE EXCEEDED SEARCHING ENDCARD CELL CYCLE.I5
      1,18H CYCLES EXTRACTED.)
 949   FORMAT (6X,26HCHARGE AND DISCHARGE PHASE)
 950   FORMAT (1H06X,F5.1,34H PERCENT AT THE BEGINNING OF CYCLEI5,20H IS
      1MISSING. BATTERYI4,14H ERROR CODE ISI4,17H. DISCHARGE PHASE)
 951   FORMAT (1H06X,F5.1,28H PERCENT AT THE END OF CYCLEI5,20H IS MISSIN
      1G. BATTERYI4,14H ERROR CODE ISI4,17H. DISCHARGE PHASE)
 952   FORMAT(/////30X24H******* STARTING BATTERYI4, 8H *******/////)
 953   FORMAT (1H06X,13HENDCARD CYCLEI5,40H          A FIELD REQUESTED CO      1
      1NTAINED -0)                                                               1
 954   FORMAT (1H06X,13HOUTCARD CYCLEI5,11H DELTA TIMEF7.4,47H  A FIELD R      1
      1EQUESTED CONTAINED -0                    )
 955   FORMAT (1H06X,13HOUTCARD CYCLEI5,40H SKIPPED. DELTA TIME FIELD CON          1
      1TAINS ZERO)                                                                 1
 956   FORMAT (1H06X,11HCYCLE RANGEI2,16H  STARTING CYCLEI5,11H  INCREMEN
      1TI5,22H  NUMBER OF INCREMENTSI5)
 957   FORMAT (1H117X)
 958   FORMAT (1H06X,13HOUTCARD CYCLEI5,72H SKIPPED. EACH CARD OF THE CYC          1
      1LE HAD A REQUESTED FIELD THAT CONTAINED -0)                                 1
 959   FORMAT (4F10.4)
       END
 )1 'XZ  X   )(P                                                          LSQP0000
 *J (((P9XH  G4'E 5'  '' 54'('  '9  '4'X'  91  '9  '4'(7  8'  '9  '74'      5 'G  LSQP0001
 *J7Z(P$VGP  G4'549   G '4  '5  7   94  '7  85 7G4(449'  X'  '9  (G5(448'  W75  '  'LSQP0002
 *J7(((5,,,3 ' (7   ' 9' (4  X1  W'  ''  ''  ''  7G6(4  (1  w'  ''  '7  =5  '45 7  9-7'LSQP0003
 *J771X$,VGT 5 =75 '  4  '5 '9 X3  (4  (1  X9  ('  '9  X'  '1  (9  (7  (2  ('  9E  (4  (LSQP0004
 *J7(,*7GGGG ' '9  (8  X9  (4  (1  ('  7'  ''  ''  ='  ''  ''  71  W'  ''  ''  '4  (1  (1  (LSQP0005
 *J7('DPGG,W ' 74  ('  91  (1  ('  ''  '4  (1  (9 ,7 ,2  ('  9E ,4 ,1  ('  ''  7'  ''  'LSQP0006
 *J7(P(****W ' 4'  7'  ''  ''  Z'  X'  ('  U'  ('  ('  '75W'  '45 '  XE5 -7''  (75(4  (LSQP0007
 *J7((G7GGGG 95 -774  (8 X9  (1  (9  X8  ('  '4  T1  ('  ''  '8  X'  71  ('  ''  ''  ''  7LSQP0008
 *J7$X(****V 1 X'  7'  7'  '4  '1  ('  '8  X'  74  (9  (75W4  '45 9  (65 9  (76(76 7  (LSQP0009
 *J7P,(3D=,0 =6 96 -6'45 96 77XG5X75(76 74 65 95 -7=5 '-4774(-77G5X-7'G6(LSQP0010
 *J6M=*PD=,T 77(46 3 (9 (4 (' '-6'46 95 7 87 (E6 -7'06'' ((X74 'D (9 U75ULSQP0011
 *J7()(7GGGD 75 74 65 95 -77D U8 X' U9 '' '' '-4'D 71 (' 7D (8 X9 (9 7' 'LSQP0012
 *J7**P**=,W ' '(X675,067G6(G4(D X' '=4 95 -4'D '8 (' '-78-67G6(-7'D '' (LSQP0013
 *J7'(P*7GGF 77,74(G4'-777  ' 'D '9 UD (8 X9 (9 U' '' UD (8 (9 (7  ' '74(LSQP0014
 *J7*)'7GGGD 75 74 66 96 07'04'047G47D '1 (' '067-5'D (1 X9 (A '' '04'D 'LSQP0015
 *J77X(****6 1 (' 'D '1 (' '76(-770677G(4 (96,-6'76(4 X9 ,7 X76 96 -6'4 (LSQP0016
 *J4*(**7(*N 1 (' '4 (G4(448G 77 W2 (' '' (' '1 4' (4 (9 '5 '4 X9 '4 W1 (LSQP0017
 *J7(X(***=W 6 (3 (9 (76,4 (96(-6'4 774(76,G5XG4X46 95 -4'-77-64G6(-65G4XLSQP0018
 *J7)$,NGGGG 04'G6(74,55 5 '5 '5 '5 '5 '5 95 776'(X'76(5 '76'(P'76V5 '76'LSQP0019
 *J7*(,***** (M'76(5 '76'(U976'5 '767' '(D'7675 '767' =(U=76'5 '(X'76X5 'LSQP0020
 *J5*X(**=,W 76,4 (96(-6'74(76,G5XG4(46 95 -4'-77(X7G6(-67G4(04'G6(76WG4(LSQP0021
 *J7*(P*7AGD 4 (9 ,75(7 ,76 65X9 ,07705'4 ,=6 94 7 96 ,9 ,-47-67G4(4 (' 7LSQP0022
 *J7)$X*NG)W 4 ,' (' (4 77 (G6(-6'96 4 (4 (9 ,5 'G4(G5(G6(5477 (7 ,9 X7 (LSQP0023
 *J7M((***** 7 X9 (7 (7 (9 (7 (7 (9 (7 X7 (9 S7 X7 X9 Z7 X7 (9 Z7 (7 (9 ZLSQP0024
 *J7*(*****P 7 (7 (9 X7 (7 X9 (7 (7 (9 (7 (7 (9 (7 (7 (9 W7 X7 X9 ,5694 XLSQP0025
 *J7(*XP.,,N ' (E4 ' (0474 S' VE4 ' (04'4 '4 Z= (4 (' Y' (E4 ' (04(4 (4 (LSQP0026
 *JWX)**7G.W = Z4 (' ,' (E4 ' (04(4 V4 (= X6 (4 (' (' (E4 ' (04(4 (4 ,= (LSQP0027
 *JX,)**7G)W 6 Z4 (' (' XE4 ' (04(4 X4 (= W6 (= (4 (' (' ,E4 ' (04(4 (4 (LSQP0028
 *JX,*(NC81   = X6 (= X94 4 (7 =5 (569                  9  9 3X  8  (        LSQP0029
 *JXXJ(1        '  7      '     6   P  *   *   *   P   *   P+ P   *   *   P   P   *  LSQP0030
 *JX(*X 'N    P  P     '94 '   P   *'*        4 (5 X= (G4(04'                LSQP0031
 *     DATA
 0111000000000000010500050002250607080910                               1
 2.000      1.500      3.250      0.500
 '                                                                       EOF E SY
```

2. Pseudo generator routine

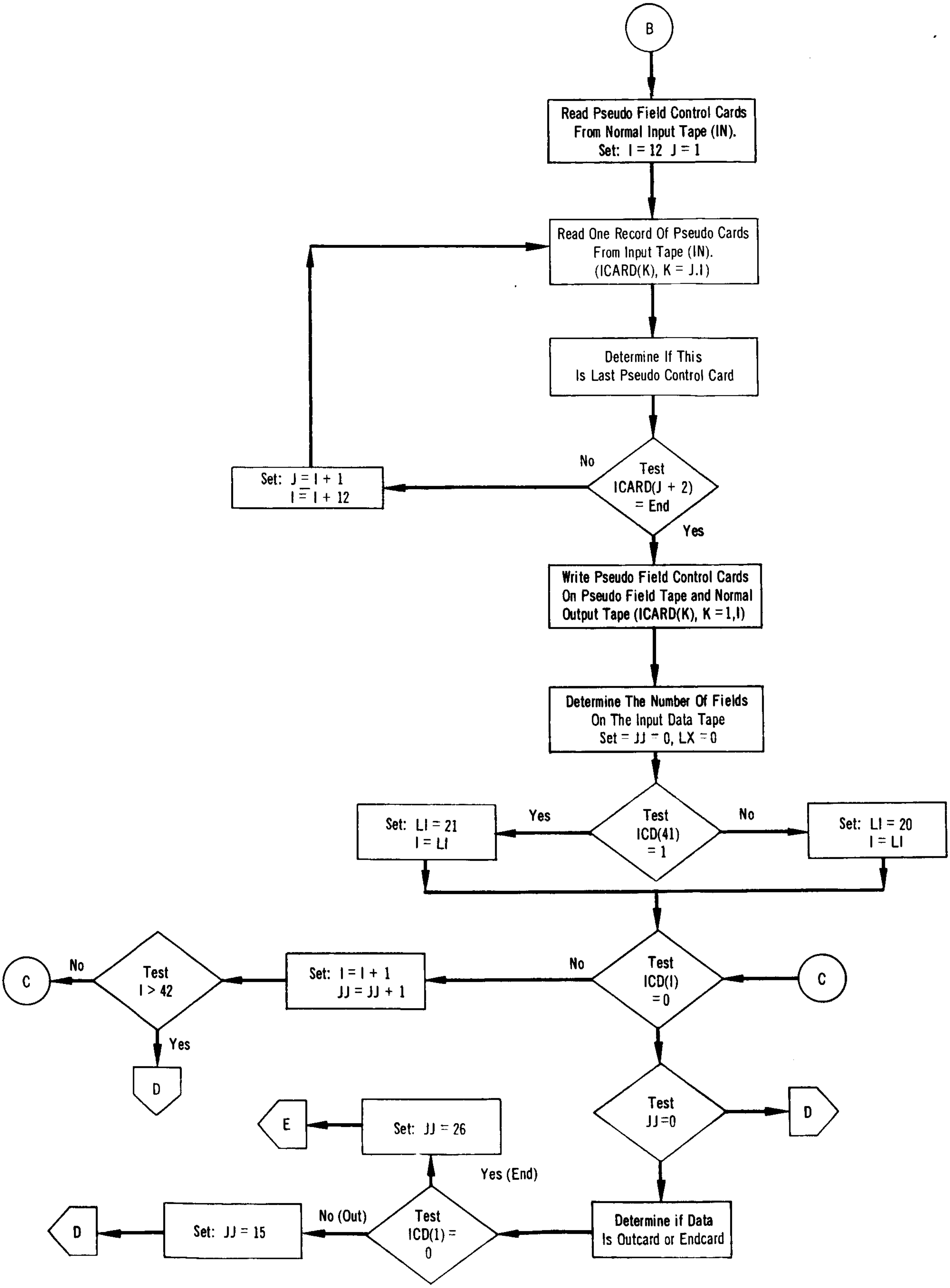
2a. Pseudo Field Generator Program -- Detailed Flow Chart


Set All 16 Pseudo Fields = 1.E 36 Initially
Read Control Card From Input Data Tape
Write Out Control Card On Print Tape
Read Pseudo Eqs. From System Input Tape and Write On Print Tape
Write Control Card and Pseudo Eqs. On Pseudo Field Tape
Determine Nature of Data From Input Data Tape
Read 1 Record of Data From Input Data Tape
Generate And Append Pseudo Fields Per Input Equations
Write New Record Of Fields Plus Pseudo Fields on Pseudo Field Tape
Yes
More Input Records
No
STOP

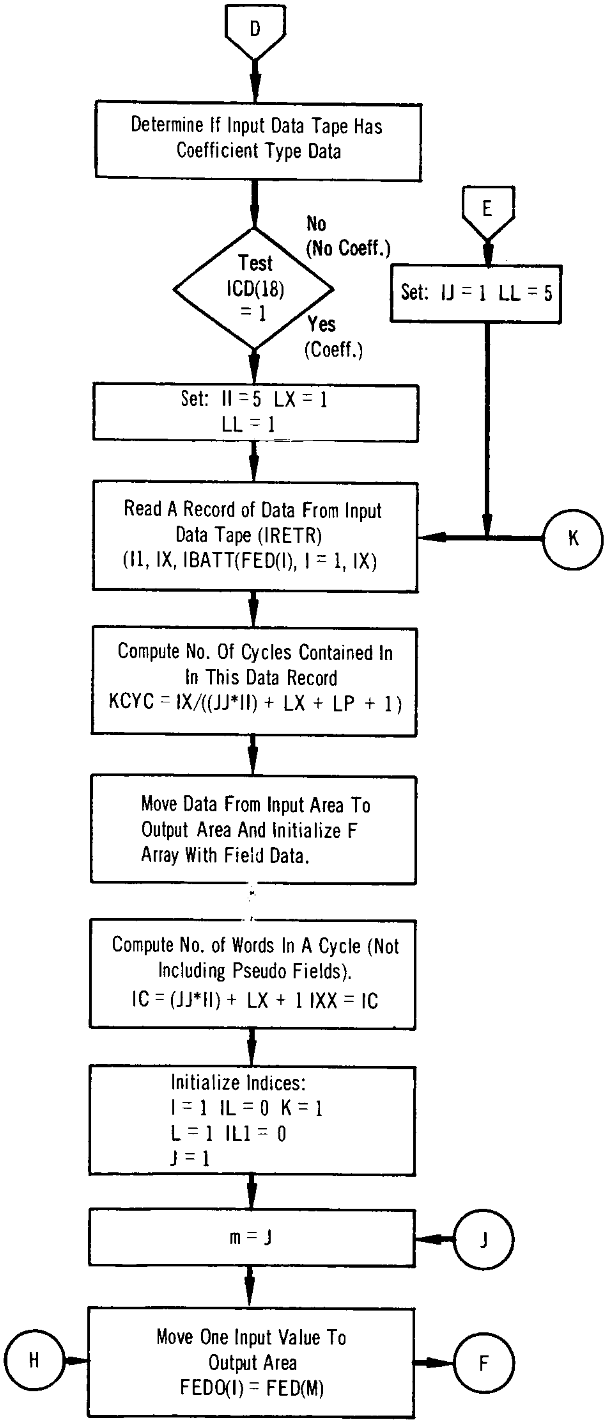
2a. Pseudo Field Generator Program —Detailed Flow Chart (Continued)



2a. Pseudo Field Generator Program — Detailed Flow Chart (Continued)

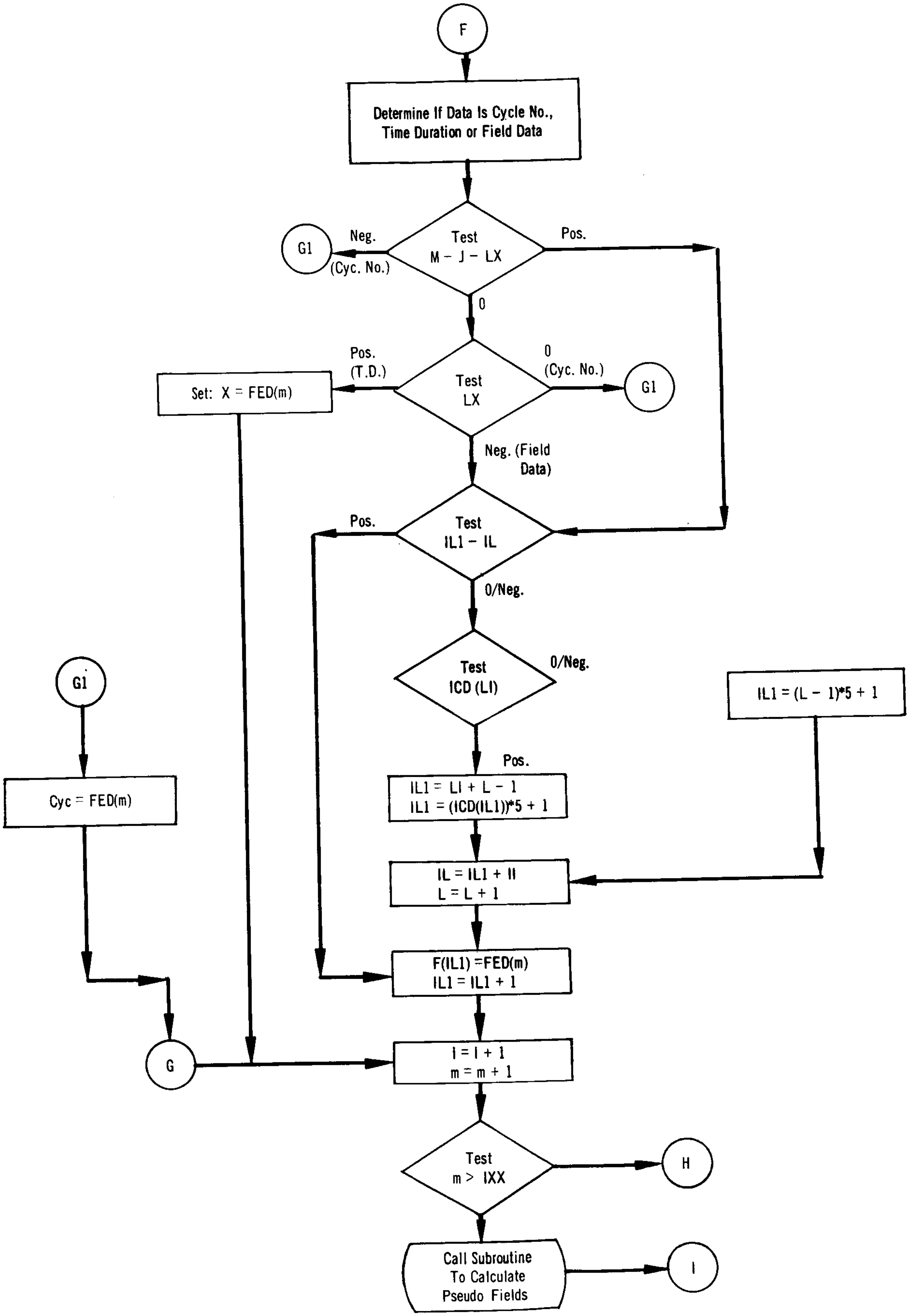

B
Read Pseudo Field Control Cards From Normal Input Tape (IN). Set: I = 12 J = 1
Read One Record Of Pseudo Cards From Input Tape (IN). (ICARD(K), K = J.I)
Determine If This Is Last Pseudo Control Card
Test ICARD(J + 2) = End
No
Yes
Set: J = I + 1 I = I + 12
Write Pseudo Field Control Cards On Pseudo Field Tape and Normal Output Tape (ICARD(K), K = 1,I)
Determine The Number Of Fields On The Input Data Tape Set = JJ = 0, LX = 0
Test ICD(41) = 1
Yes
No
Set: LI = 21 I = LI
Set: LI = 20 I = LI
Test ICD(I) = 0
No
C
Set: I = I + 1 JJ = JJ + 1
Test I > 42
No
C
Yes
D
Test JJ=0
D
Determine if Data Is Outcard or Endcard
Test ICD(1) = 0
Yes (End)
Set: JJ = 26
E
No (Out)
Set: JJ = 15
D

2a. Pseudo Field Generator Program – Detailed Flow Chart (Continued)

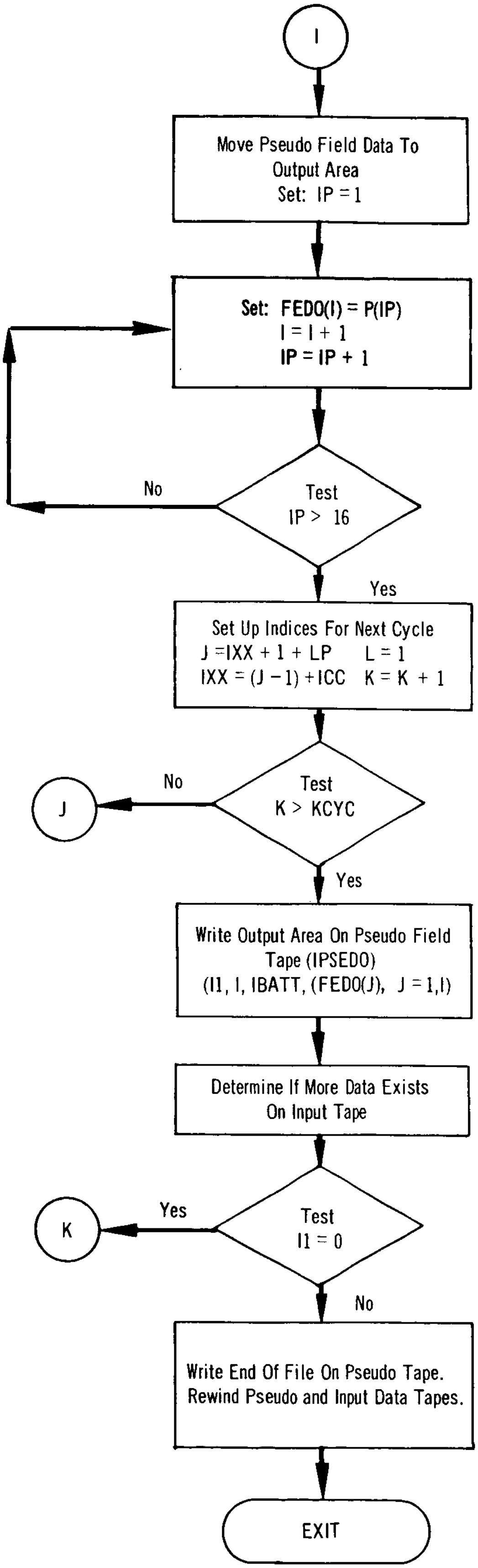


D
Determine If Input Data Tape Has Coefficient Type Data
Test ICD(18) = 1
No (No Coeff.)
Yes (Coeff.)
E
Set: IJ = 1 LL = 5
Set: II = 5 LX = 1 LL = 1
Read A Record of Data From Input Data Tape (IRETR) (I1, IX, IBATT(FED(I), I = 1, IX)
K
Compute No. Of Cycles Contained In In This Data Record KCYC = IX/((JJ*II) + LX + LP + 1)
Move Data From Input Area To Output Area And Initialize F Array With Field Data.
Compute No. of Words In A Cycle (Not Including Pseudo Fields). IC = (JJ*II) + LX + 1 IXX = IC
Initialize Indices: I = 1 IL = 0 K = 1 L = 1 IL1 = 0 J = 1
m = J
J
H
Move One Input Value To Output Area FEDO(I) = FED(M)
F

2a. Pseudo Field Generator Program — Detaiiea r'iow Chart (Continued)



2a. Pseudo Field Generator Program – Detailed Flow Chart (Continued)


I
Move Pseudo Field Data To Output Area
Set: IP = 1
Set: FEDO(I) = P(IP)
I = I + 1
IP = IP + 1
No
Test
IP > 16
Yes
Set Up Indices For Next Cycle
J = IXX + 1 + LP   L = 1
IXX = (J − 1) + ICC   K = K + 1
J
No
Test
K > KCYC
Yes
Write Output Area On Pseudo Field Tape (IPSEDO)
(I1, I, IBATT, (FEDO(J), J = 1,I)
Determine If More Data Exists On Input Tape
K
Yes
Test
I1 = 0
No
Write End Of File On Pseudo Tape.
Rewind Pseudo and Input Data Tapes.
EXIT

2b. Pseudo Field Generator Routine — Source Statements

```
*       CARDS COLUMN
*       LIST
*       SYMBOL TABLE
C PSEUDO FIELD GENERATOR PROGRAM.
        COMMON FEDO,FED,F,P,X,CYC
        DIMENSION FEDO(4500),FED(4500),F(130),P(16),ICD(42),ICARD(1200),
       1F1(5), F2(5), F3(5), F4(5), F5(5), F6(5), F7(5), F8(5), F9(5)  ,
       2F10(5),F11(5),F12(5),F13(5),F14(5),F15(5),F16(5),F17(5),F18(5) ,
       3F19(5),F20(5),F21(5),F22(5),F23(5),F24(5),F25(5),F26(5)
        EQUIVALENCE (P(1),P1),(P(2),P2),(P(3),P3),(P(4),P4),(P(5),P5),
       1(P(6),P6),(P(7),P7),(P(8),P8),(P(9),P9),(P(10),P10),(P(11),P11),
       2(P(12),P12),(P(13),P13),(P(14),P14),(P(15),P15),(P(16),P16)
        EQUIVALENCE (F(1),F1),(F(6),F2),(F(11),F3),(F(16),F4),(F(21),F5),
       1(F(26),F6),(F(31),F7),(F(36),F8),(F(41),F9),
       1(F(46),F10),(F(51),F11)
       2,(F(56),F12),(F(61),F13),(F(66),F14),(F(71),F15),(F(76),F16),
       3(F(81),F17),(F(86),F18),(F(91),F19),(F(96),F20),(F(101),F21),
       4(F(106),F22),(F(111),F23),(F(116),F24),(F(121),F25),(F(126),F26)
        EQUIVALENCE (BFLAG,IBFLAG),(AFLAG,IAFLAG),(END,IEND),
       1 (CFLAG,ICFLAG)
C.....SET UP TAPE ASSIGNMENTS.
 10     IN = 2
 20     IO = 3
 30     IRETR = 19
 40     IPSEDO = 17
C.....SET PSEUDO FIELDS = 1.E36 INITIALLY.
 100    DO 110 I=1,16
 110    P(I) = 1.E36
 120    REWIND IRETR
 130    REWIND IPSEDO
C.....READ CONTROL CARD FROM INPUT DATA TAPE.
 200    BFLAG = 6H$$$$$R
 210    CFLAG = 6H$$$$$P
 220    READ TAPE IRETR, AFLAG,(ICD(I),I=1,42)
C.....TEST IF 2ND CONTROL CARD EXISTS.
        IF (ICD(39)-1) 222,221,222
 221    READ TAPE IRETR , (FED(I),I=1,18)
        FED(19) = 0.
C.....EJECT PAGE BEFORE WRITING CONTROL CARD.
 222    WRITE OUTPUT TAPE IO,230
 230    FORMAT(18H1                 )
C.....WRITE OUT CONTROL CARD.
        CALL CCARD (ICD,FED)
C.....TEST IF INPUT DATA TAPE IS A RETRIEVED OR PSEUDO FIELD DATA TAPE.
C.....$$$$$R = RETRIEVED , $$$$$P = PSEUDO FIELD .
        IF (IBFLAG-IAFLAG) 241,240,241
C.....SET AFLAG = $$$$$P
 240    AFLAG = CFLAG
        LP = 0
        GO TO 250
 241    IF (ICFLAG-IAFLAG) 900,242,900
 242    LP = 16
C.....SKIP OVER PSEUDO EQUATION CARDS ON INPUT PSEUDO FIELD TAPE.
        READ TAPE IRETR, ICARD(1)
C.....WRITE CONTROL RECORD ON PSEUDO FIELD DATA TAPE.
 250    WRITE TAPE IPSEDO, AFLAG,(ICD(I),I=1,42)
        IF (ICD(39)-1) 251,2500,251
 2500   WRITE TAPE IPSEDO, (FED(I),I=1,18)
```

2b. Pseudo Field Generator Routine — Source Statements (Continued)

```
C.....TEMPORARY PROCEDURE.....READ PSEUDO EQUATION CARDS FROM INPUT TAPE
 251    END = 3HEND
 252    I = 12
 253    J = 1
 254    DO 259 JJ=1,100
 255    READ INPUT TAPE IN,256, (ICARD(K),K=J,I)
 256    FORMAT(12A6)
C.....TEST FOR END OF PSEUDO EQUATION CARDS.
 257    IF (ICARD(J+1)-IEND) 258,261,258
 258    J = I+1
 259    I = I + 12
C.....WRITE PSEUDO EQUATION CARDS ON PSEUDO FIELD DATA TAPE.
 261    WRITE TAPE IPSEDO, I, (ICARD(K),K=1,I)
C.....WRITE OUT PSEUDO EQUATION CARDS.
        WRITE OUTPUT TAPE IO,262, (ICARD(K),K=1,I)
 262    FORMAT(25H0  PSEUDO EQUATION CARDS //(2X,12A6))
C.....DETERMINE THE NUMBER OF FIELDS ON INPUT DATA TAPE.
 300    JJ = 0
        LX = 0
        IF(ICD(41)-1) 305,306,305
 305    LI = 20
        GO TO 310
 306    LI = 21
 310    DO 330  I = LI,38
        IF (ICD(I)) 320,340,320
 320    JJ = JJ + 1
 330    CONTINUE
        GO TO 360
C.....TEST TO DETERMINE IF ALL FIELDS ARE ON THE INPUT DATA TAPE.
 340    IF (JJ) 360,350,360
C.....TEST TO DETERMINE IF DATA IS OUTCARD OR ENDCARD DATA.
 350    IF(ICD(1)) 351,352,351
 351    JJ = 26
        GO TO 370
 352    JJ = 15
C.....DETERMINE IF COEFFICIENTS EXIST ON INPUT DATA TAPE.
 360    IF(ICD(18)-1) 370,380,370
 370    II = 1
        LL = 5
 371    GO TO 400
 380    II = 5
        LL = 1
 381    LX  = 1
C.....READ COMPLETE RECORD OF INPUT DATA TAPE.
 400    READ TAPE IRETR, I1,IX,IBATT,(FED(I),I=1,IX)
C.....COMPUTE NO. OF CYCLES IN THIS RECORD.
        KCYC = IX/((JJ*II)+LX+LP+1)
C.....MOVE DATA FROM INPUT BUFFER TO OUTPUT BUFFER.
C.....INITIALIZE F BLOCK WITH DATA FROM INPUT DATA TAPE FOR 1 CYCLE.
        IC = (JJ*II)+LX+1
        IXX = IC
 510    I    =  1
 520    L    =  1
 530    J    =  1
        IL = 0
        IL1 = 0
 540    DO 740 K = 1,KCYC
 550    DO 592 M =J,IXX
```

2b. Pseudo Field Generator Routine — Source Statements (Continued)

```
 560   FEDO(I) = FED(M)
 570   IF (M-J-LX) 5800,580,590
 580   IF(LX) 590,5800,581
 5800  END = FED(M)
       CYC = IEND
       GO TO 592
 581   X = FED(M)
       GO TO 592
 590   IF (IL1-IL) 591,5900,5900
 5900  IF (ICD(LI)) 5901,5901,5902
 5901  IL1 = (L-1)*5+1
       GO TO 5903
 5902  IL1 = LI+L-1
       IL1 = (ICD(IL1)-1)*5+1
 5903  IL = IL1 + II
       L = L+1
 591   F(IL1) = FED(M)
       IL1 = IL1+1
 592   I = I + 1
C.....CALL SUBROUTINE TO CALCULATE PSEUDO FIELDS.
       CALL EQUAI(F1,F2,F3,F4,F5,F6,F7,F8,F9,F10,F11,F12,F13,F14,F15,F16,
      1          F17,F18,F19,F20,F21,F22,F23,F24,F25,F26,X,P1,P2,P3,P4,
      2          P5,P6,P7,P8,P9,P10,P11,P12,P13,P14,P15,P16,CYC )
C.....MOVE PSEUDO FIELD DATA INTO OUTPUT BUFFER.
 600   DO 620 IP = 1,16
 610   FEDO(I) = P(IP)
 620   I = I + 1
C.....SET UP INDICIES FOR NEXT CYCLE.
 700   J = IXX + 1 + LP
       IXX = (J-1)+IC
 740   L = 1
C.....WRITE OUTPUT BUFFER ON PSEUDO FIELD DATA TAPE.
 800   I = I - 1
 810   WRITE TAPE IPSEDO, I1,I,IBATT,(FEDO(J),J=1,I)
C.....TEST IF MORE DATA EXISTS ON INPUT DATA TAPE.
 820   IF (I1) 830,400,830
 830   END FILE IPSEDO
       REWIND IRETR
 840   REWIND IPSEDO
       CALL EXIT
C.....IMPROPER INPUT DATA TAPE HAS BEEN MOUNTED. WRITE ERROR STATEMENT.
 900   PRINT 910
 910   FORMAT(80H1 IMPROPER INPUT DATA TAPE HAS BEEN MOUNTED. MOUNT PROPE
      1R TAPE AND PUSH START.
 920   PAUSE
 930   GO TO 120
       END
```

2c. Control Card Flow Subroutine Used in Pseudo Generator Routine Data Listing Routine, Statistical Analysis Routine, and Printed Plot Routine

Flow Charts and Source Statements

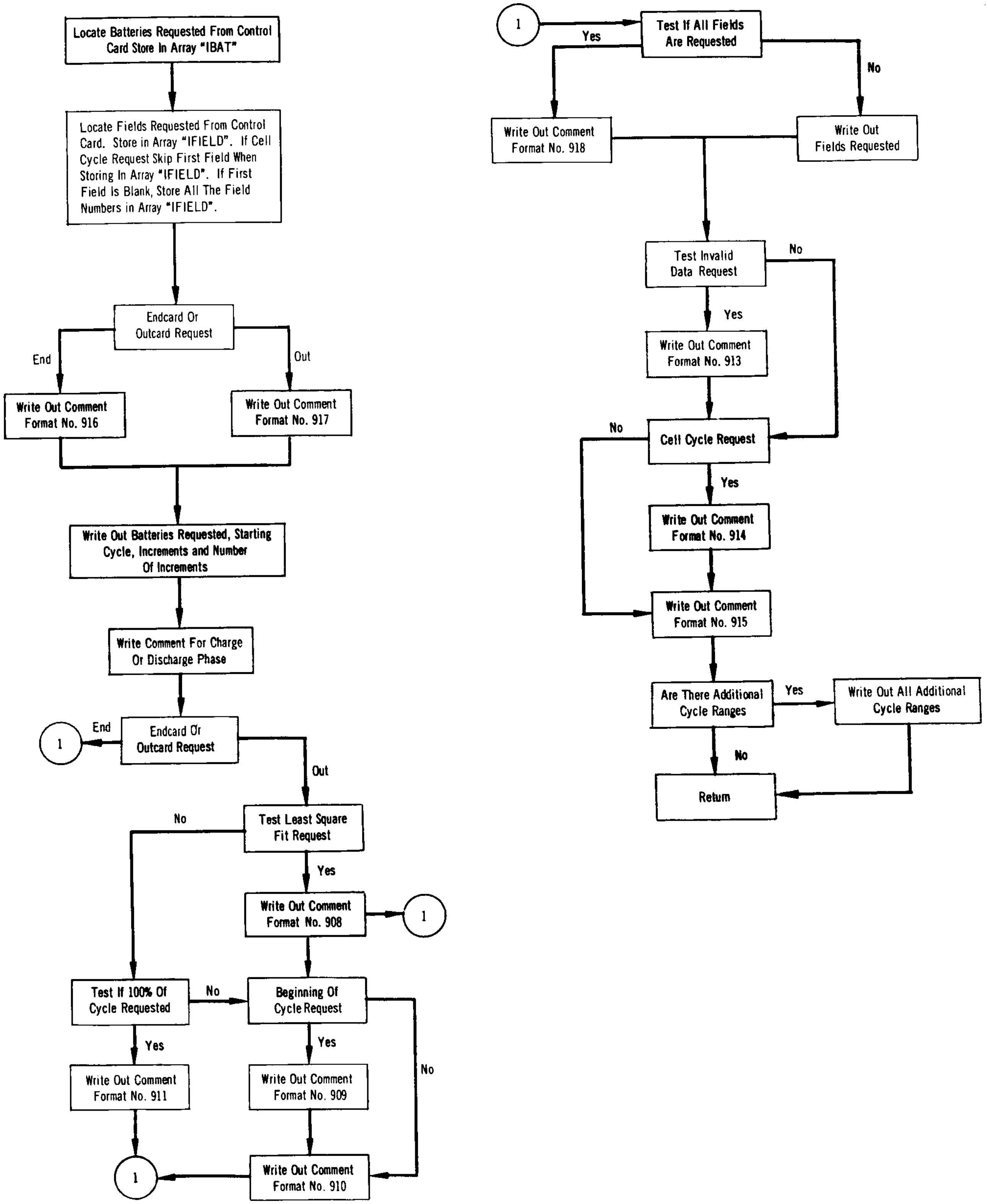
2c. Control Card Flow Subroutine Used in Pseudo Generator Routine Data Listing Routine, Statistical Analysis Routine, and Printed Plot Routine Flow Charts and Source Statements (Continued)

```
CCARD

         SUBROUTINE CCARD(ICD.IRNG)
         DIMENSION ICD(43).IBAT(15).IFIELD(30).IRNG(19)
         IO = 3
         IRNG(19) =0
         K1= 0
         DO 23 I= 2.13
         IF (ICD(I)) 24.23.24
   24    K1= K1 +1
         IBAT(K1)= I-2
   23    CONTINUE
C.....   LOCATE FIELDS REQUESTED
         K2 =0
         IF(ICD(42))3.4.3
    3    II1=21
         GO TO 5
    4    II1=20
    5    DO 33 I=II1.36
         IF(ICD(I)) 31.32.31
   31    K2 = K2 +1
         IFIELD(K2) = ICD(I)
   33    CONTINUE
   32    IF(K2)   40.38.40
C.....   ALL FIELDS ARE TO BE SELECTED
   38    IF(ICD(1)) 35.34.35
C.....   OUTCARD
   34    J=15
         GO TO 36
C.....   ENDCARD
   35    J = 26
   36    DO 37 K2 =1. J
         IFIELD(K2)= K2
   37    CONTINUE
   40    IGYG = ICD(14)
         IF(ICD(1)) 42.41.42
   41    WRITE OUTPUT TAPE IO. 917
         GO TO 43
   42    WRITE OUTPUT TAPE IO. 916
   43    WRITE OUTPUT TAPE IO. 903.(IBAT(I).I=1.K1)
         WRITE OUTPUT TAPE IO. 904. IGYG
         WRITE OUTPUT TAPE IO. 905. ICD(15).ICD(16)
         IF(ICD(17))  45.44.45
   44    WRITE OUTPUT TAPE IO. 906
         GO TO 46
   45    IF(ICD(17)-2) 511.510.511
  510    WRITE OUTPUT TAPE IO.949
         GO TO 46
  511    WRITE OUTPUT TAPE IO.907
   46    IF (ICD(1))  54.47.54
   47    IF (ICD(30) - 1 ) 49 .48. 49
   48    WRITE OUTPUT TAPE IO. 908
         GO TO 54
   49    IF (ICD(18))   51.50.51
   50    WRITE OUTPUT TAPE IO. 911
         GO TO 54
   51    IF (ICD(18) - 2 )  53.52.53
```

2c. Control Card Flow Subroutine Used in Pseudo Generator Routine Data Listing Routine, Statistical Analysis Routine, and Printed Plot Routine Flow Charts and Source Statements (Continued)

```
~~CCARD~~

~~52      WRITE OUTPUT TAPE IO, 909, ICD(19)~~
        GO TO 54
~~53      WRITE OUTPUT TAPE IO,910, ICD(19)~~
54      IF(ICD(20))56.55.56
~~55      WRITE OUTPUT TAPE IO, 918~~
        GO TO 57
~~56      WRITE OUTPUT TAPE IO, 912,(IFIELD(I),I=1,K2)~~
57      IF(ICD(41)) 58.59.58
~~58      WRITE OUTPUT TAPE IO, 913~~
59      IF(ICD(42)) 60.61.60
~~60      WRITE OUTPUT TAPE IO.914.ICD(20)~~
61      WRITF OUTPUT TAPF IO.915.ICD(42)
~~        IF (ICD(40)) 63,62.63~~
63      I =1                                                          1
~~        ICR = 2                                                       1~~
64      WRITE OUTPUT TAPE IO. 955.ICR.IRNG(I).IRNG(I+1).IRNG(I+2)     1
~~        I = I+3                                                       1~~
        ICR = ICR +1                                                  1
~~        IF(IRNG(I)) 64,62.64~~
62      RETURN                                                        1
~~903     FORMAT  (1H05X,23HBATTERYS REQUESTED ARE 12I4)~~
904     FORMAT (6X.21HSTARTING CYCLE NO IS I4)
~~905     FORMAT (6X,13HINCREMENT IS I3,22H NJMBER OF INCREMENTS I4)~~
906     FORMAT (6X.12HCHARGE PHASE)
~~907     FORMAT (6X.15HDISCHARGE PHASE)~~
908     FORMAT (6X.36HLEAST SQUARE FIT ON REQUESTED FIELDS)
~~909     FORMAT  (6X,6HFIRST I2,27H PERCENT OF CYCLE REQUESTED)~~
910     FORMAT (6X.5HLAST I2.27H PERCENT OF CYCLE REQUESTED)
~~911     FORMAT (6X,31H100 PER CENT OF CYCLE REQUESTED)~~
912     FORMAT (6X.16HFIELDS REQUESTED22I4)
~~913     FORMAT (6X,22HINVALID DATA REQUESTED)~~
914     FORMAT(6X.35HCELL CYCLE REQUEST ON OUTCARD FIELDI5)
~~915     FORMAT (6X,11HEXPIRED DAYI4)~~
916     FORMAT (6X.22HENDCARD DATA REQUESTED)
~~917     FORMAT (6X,22HOUTCARD DATA REQUESTED)~~
918     FORMAT (6X.20HALL FIELDS REQUESTED)
~~949     FORMAT (6X,26HCHARGE AND DISCHARGE PHASE)~~
955     FORMAT (1H06X.11HCYCLE RANGEI2.16H  STARTING CYCLEI5,11H  INCREMEN   1
~~       1TI5,22H  NUMBER OF INCREMENTSI5)                                 1~~
        END(1.0.0.0.0.0.1.0.0.1.0.0.0.0.0)
```

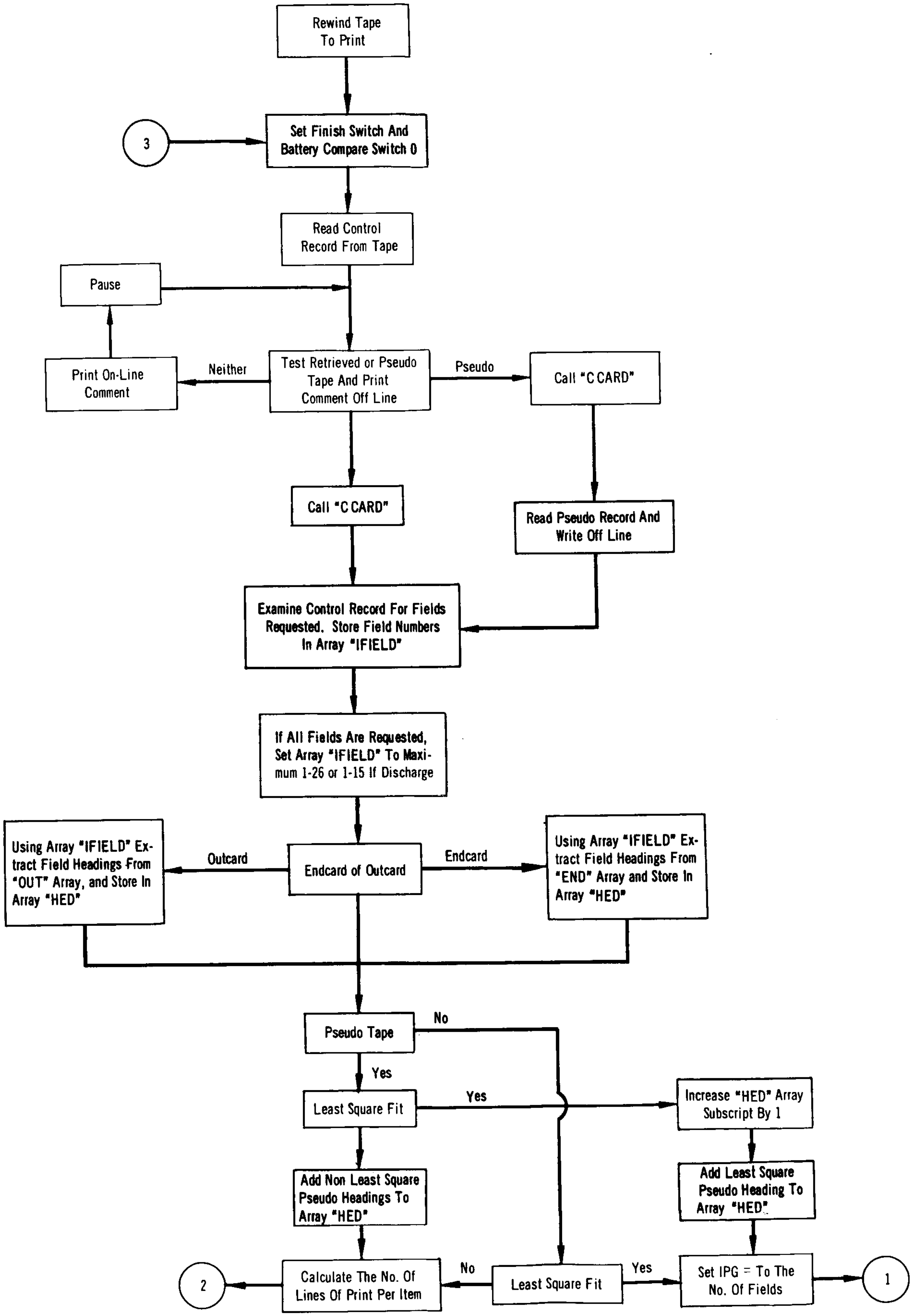
2c. Control Card Flow Subroutine Used in Pseudo Generator Routine, Data Listing Routine, Statistical Analysis Routine, and Printed Plot Routine Flow Charts and Source Statements



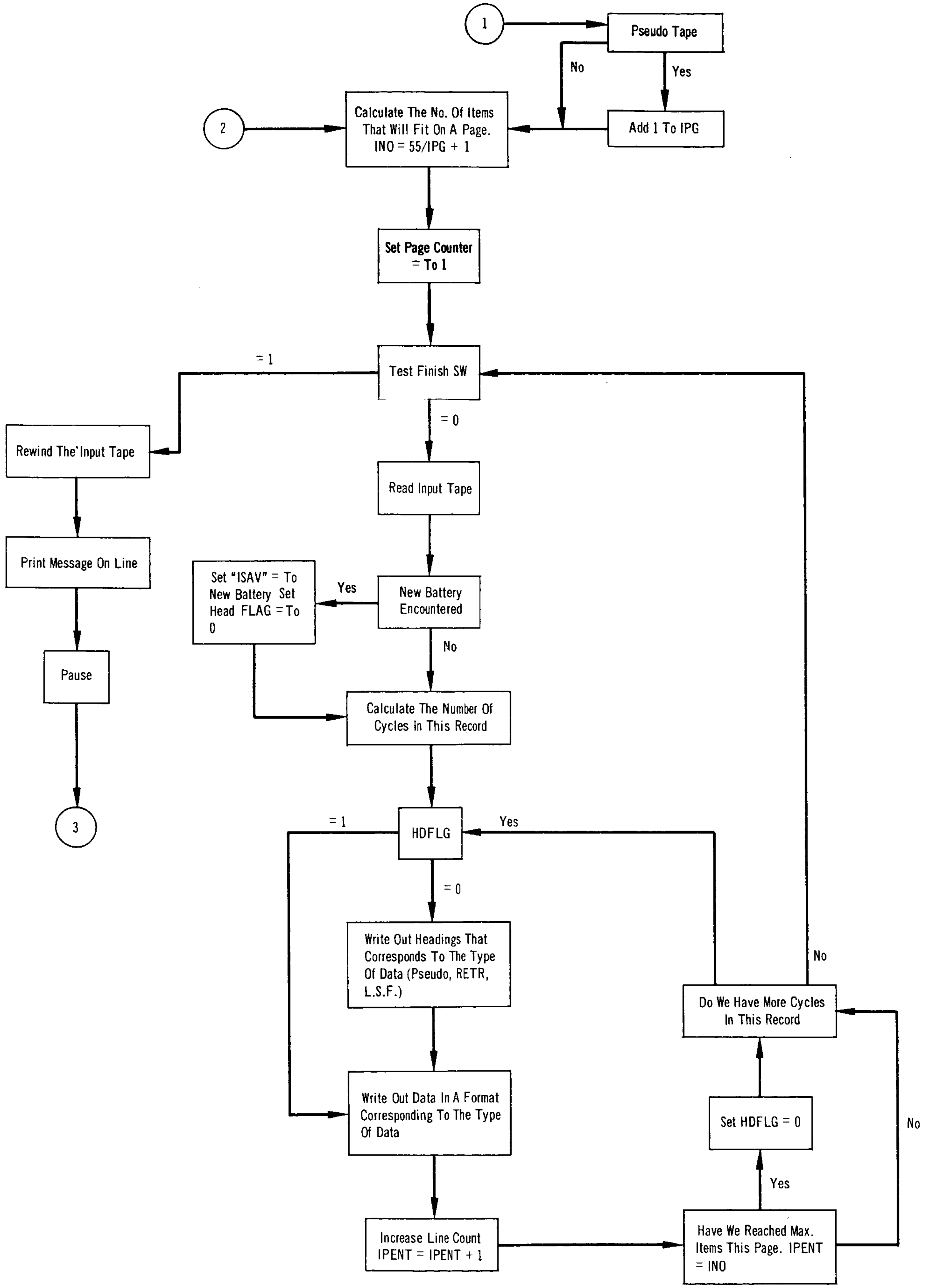
Locate Batteries Requested From Control Card Store In Array "IBAT"
Locate Fields Requested From Control Card. Store in Array "IFIELD". If Cell Cycle Request Skip First Field When Storing In Array "IFIELD". If First Field Is Blank, Store All The Field Numbers in Array "IFIELD".
Endcard Or Outcard Request
End
Out
Write Out Comment Format No. 916
Write Out Comment Format No. 917
Write Out Batteries Requested, Starting Cycle, Increments and Number Of Increments
Write Comment For Charge Or Discharge Phase
1
End
Endcard Or Outcard Request
Out
No
Test Least Square Fit Request
Yes
Write Out Comment Format No. 908
1
Test If 100% Of Cycle Requested
No
Beginning Of Cycle Request
Yes
Yes
No
Write Out Comment Format No. 911
Write Out Comment Format No. 909
1
Write Out Comment Format No. 910
1
Yes
Test If All Fields Are Requested
No
Write Out Comment Format No. 918
Write Out Fields Requested
Test Invalid Data Request
No
Yes
Write Out Comment Format No. 913
No
Cell Cycle Request
Yes
Write Out Comment Format No. 914
Write Out Comment Format No. 915
Are There Additional Cycle Ranges
Yes
Write Out All Additional Cycle Ranges
No
Return

3. Data List Routine Flow Charts

3a. Data List Routine Flow Charts


Rewind Tape To Print
3
Set Finish Switch And Battery Compare Switch 0
Read Control Record From Tape
Pause
Print On-Line Comment
Neither
Test Retrieved or Pseudo Tape And Print Comment Off Line
Pseudo
Call "C CARD"
Call "C CARD"
Read Pseudo Record And Write Off Line
Examine Control Record For Fields Requested. Store Field Numbers In Array "IFIELD"
If All Fields Are Requested, Set Array "IFIELD" To Maximum 1-26 or 1-15 If Discharge
Using Array "IFIELD" Extract Field Headings From "OUT" Array, and Store In Array "HED"
Outcard
Endcard of Outcard
Endcard
Using Array "IFIELD" Extract Field Headings From "END" Array and Store In Array "HED"
Pseudo Tape
No
Yes
Least Square Fit
Yes
Increase "HED" Array Subscript By 1
Add Non Least Square Pseudo Headings To Array "HED"
Add Least Square Pseudo Heading To Array "HED"
2
Calculate The No. Of Lines Of Print Per Item
No
Least Square Fit
Yes
Set IPG = To The No. Of Fields
1

3a. Data List Routine Flow Charts (Continued)


1
Pseudo Tape
No
Yes
Add 1 To IPG
2
Calculate The No. Of Items That Will Fit On A Page. INO = 55/IPG + 1
Set Page Counter = To 1
= 1
Test Finish SW
= 0
Rewind The' Input Tape
Read Input Tape
Print Message On Line
Set "ISAV" = To New Battery Set Head FLAG = To 0
Yes
New Battery Encountered
No
Pause
Calculate The Number Of Cycles In This Record
3
= 1
HDFLG
Yes
= 0
Write Out Headings That Corresponds To The Type Of Data (Pseudo, RETR, L.S.F.)
No
Do We Have More Cycles In This Record
Write Out Data In A Format Corresponding To The Type Of Data
Set HDFLG = 0
No
Yes
Increase Line Count IPENT = IPENT + 1
Have We Reached Max. Items This Page. IPENT = INO

3b. Data List Routine Source Statements

```
*       LIST8
*       CARDS COLUMN
*       SYMBOL TABLE
*       FORMAP
*       LABEL
CPRINT
C       GENERALIZED PRINT PROGRAM WHICH WILL PRINT RETRIEVED OR PSEUDO
C       GENERATED TAPES
        DIMENSION  END(25),OUT(15),PPS(1200),PP(16),IFIELD(26),DATA(4500),       1
       1 HED(42),ICD(43),IRNG(19)                                                1
        EQUIVALENCE (FR,IR),(FP,IP)
        IREQ = 0
        IO = 3
        IPRT = 19
C       END CARD HEADINGS
        END(1) =6HTI DUR
        END(2) =6HAMP MI
        END(3) =6HWAT MI
        END(4) =6H E EFF
        END(5) =6H AMP H
        END(6) =6H WAT H
        END(7) =6HAV CUR
        END(8) =6HMI CUR
        END(9) =6HMX CUR
        END(10)=6HAV PWR
        END(11)=6HMI PWR
        END(12)=6HMX PWR
        END(13)=6H C EFF
        END(14)=6HE/I  R
        END(15)=6HCYC  0
        END(16)=6HCYC  1
        END(17)=6HCYC  2
        END(18)=6HCYC  3
        END(19)=6HCYC  4
        END(20)=6HCYC  5
        END(21)=6HCYC  6
        END(22)=6HCYC  7
        END(23)=6HCYC  8
        END(24)=6HCYC  9
        END(25)=6HMX VAR
        END(26)=6HR TIME
C       OUTCARD HEADINGS
        OUT(1) =6H TIME
        OUT(2) =6HD TIME
        OUT(3) =6HAV PWR
        OUT(4) =6H  CUR
        OUT(5) =6HPRESS
        OUT(6) =6H V 0
        OUT(7) =6H V 1
        OUT(8) =6H V 2
        OUT(9) =6H V 3
        OUT(10)=6H V 4
        OUT(11)=6H V 5
        OUT(12)=6H V 6
        OUT(13)=6H V 7
        OUT(14)=6H V 8
        OUT(15)=6H V 9
        FR =6H$$$$$R
```

```
         FP  =6H$$$$$P
         PPP=6HP1-8
         PPPP   =6HP9-16
         PP(1)   =6H   P1
         PP(2)   =6H   P2
         PP(3)   =6H   P3
         PP(4)   =6H   P4
         PP(5)   =6H   P5
         PP(6)   =6H   P6
         PP(7)   =6H   P7
         PP(8)   =6H   P8
         PP(9)   =6H   P9                                                            1
         PP(10) =6H  P10                                                             1
         PP(11) =6H  P11                                                             1
         PP(12) =6H  P12                                                             1
         PP(13) =6H  P13                                                             1
         PP(14) =6H  P14                                                             1
         PP(15) =6H  P15                                                             1
         PP(16) =6H  P16                                                             1
 C       READ FIRST RECORD  CONTROL INFORMATION
 1       REWIND IPRT
         IFIN = 0
         ISAV = 99
         READ TAPE IPRT,IFLG, (ICD(I),I=1,43)
         IREQ  = IREQ+1
         IF(ICD(40)) 34,33,34
 34      READ TAPE IPRT,(IRNG(J8),J8=1,18)                                           1
 C       RETRIEVED OR PSEUDO
 33      IF (IFLG - IR) 31,30,31
 30      WRITE OUTPUT TAPE IO, 900, IREQ
         CALL CCARD (ICD,IRNG)                                                       1
         GO TO 41
 31      IF (IFLG - IP) 32,40,32
 C       INCORRECT TAPE
 32      PRINT  901
         PAUSE
         GO TO  33
 40      WRITE OUTPUT TAPE IO,902, IREQ
         CALL CCARD (ICD,IRNG)                                                       1
 C       READ PSEUDO CARDS  AND WRITE OFF LINE
         READ TAPE IPRT,INO, (PPS(I),I=1,INO)
         WRITE OUTPUT TAPE IO, 903
         WRITE OUTPUT TAPE IO, 904, (PPS(I), I=1,INO)
 C       DETERMINE WHICH FIELDS ARE USED IN ORDER TO SET UP FIELD HEADINGS
 41      WRITE OUTPUT TAPE IO,915
         K2 =0
         IF(ICD(42)) 3,4,3
 3       II1=21
         GO TO 5
 4       II1=20
 5       DO 45 I=II1,36
         IF(ICD(I)) 46,47,46
 46      K2 = K2+1
         IFIELD(K2) = ICD(I)
 45      CONTINUE
 47      IF(K2)   55,48,55
 C       ALL FIELDS ARE TO BE SELECTED
 48      IF(ICD(1))  50,49,50
```

3b. Data List Routine Source Statements (Continued)

```
 C      OUTCARD
 49     J = 15
        GO TO 51
 C      ENDCARD
 50     J = 26
 51     DO 52 K2=1, J
        IFIELD(K2) = K2
 52     CONTINUE
        K2=K2-1
 C      SETUP FORMATS AND CALCULATE THE NO. OF ITEMS PER PAGE
 55     DO 65  I =1 , K2
        I = I
        IA = IFIELD(I)
        IF(ICD(1))61,62,61
 61     HED(I) = END(IA)
        GO TO 65
 62     HED(I) = OUT(IA)
 65     CONTINUE
        I=I-1
        IPG = 0
 C      TEST PSEUDO AND NOT L.S.F.
        IF(IFLG - IP)   77,70,77
 70     IF(ICD(38) - 1 ) 71,76,71
 C      YES .  ADD P  HEADINGS TO HED(I)
 71     DO 75 J=1,16                                                   1
 72     I = I+1
        HED(I) = PP(J)
 75     CONTINUE
 C      CALCULATE ITEMS PER PAGE
 78     I1 = I
 80     IF(I1 -8) 81,81,79
 79     IPG = IPG +1
        I1 = I1 -8
        GO TO 80
 81     IPG = IPG +1
        GO TO 83
 C      TEST  L.S.F.
 77     IF(ICD(38) - 1) 78,74,78
 76     HED(I+1) = PPP                                                 1
        HED(I+2) = PPPP                                                1
 74     IPG = K2
        IF(IFLG - IP )  83,82,83
 82     IPG = IPG + 2                                                  1
 C      CALC. NO OF ITEMS PER PAGE. NON L.S.F. COUNT INCLUDES HEADING
 83     INO =  55 /(IPG + 1)
 C      READ DATA
 88     IPAG = 1
 89     IF(IFIN)140,87,140
 87     READ TAPE IPRT,IFIN,ICNT,IBAT, (DATA(L1),L1=1,ICNT )
        IF(IBAT-ISAV) 84,85,84
 84     ISAV=IBAT
        HDFLG=0
 85     LP=0
        LX = 0
        II = 1
        IF (IFLG - IP ) 91,90,91
 90     LP = 16                                                        1
 91     IF (ICD(38) - 1 )  93,92,93
```

3b. Data List Routine Source Statements (Continued)

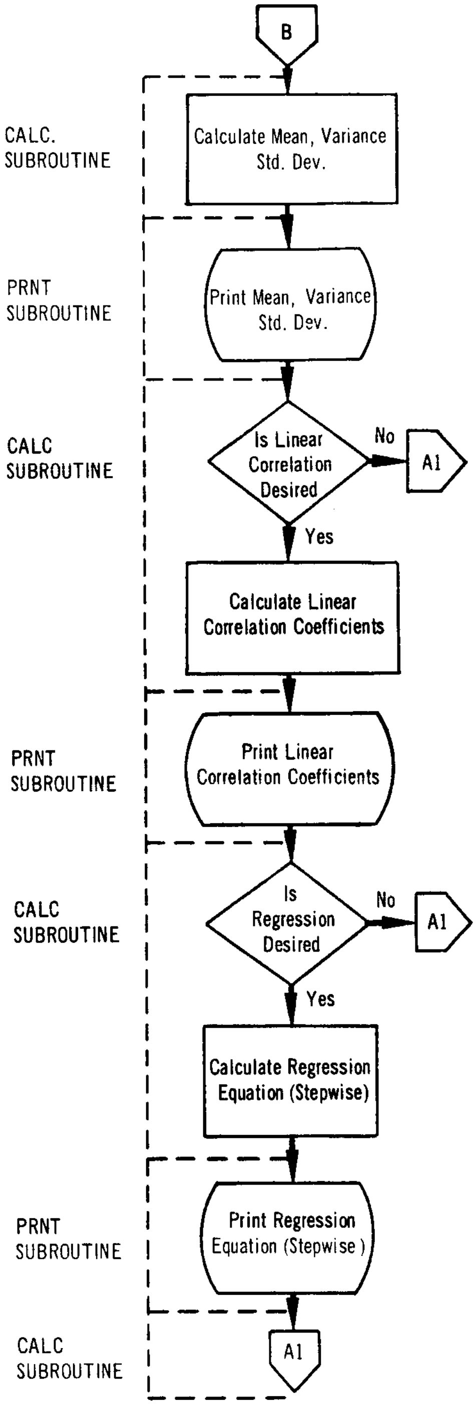
```
 92     LX =1
        II =5
        J5 = K2 * II + 1 + LX + LP
 93     KCY = ICNT / ((K2 * II ) +1 +LX + LP)
        IF (ICD(38) -1) 94,110,94
 94     J1 = 1
        J3 =2
        J8 = K2 + LP + 1
        J4 = J8
        DO  100     KK =1 , KCY
        IF (HDFLG)  96,95,96
 95     WRITE OUTPUT TAPE IO,905, IPAG
        WRITE OUTPUT TAPE IO,906, (HED(K),K=1,I )
        WRITE OUTPUT TAPE IO,913
        IPAG = IPAG +1
        IPCNT = 1
 C      WRITE DATA
        HDFLG =1
 96     WRITE OUTPUT TAPE IO,908,IBAT,DATA(J1),(DATA(J2),J2=J3,J4)
        WRITE OUTPUT TAPE IO,913
        J1 = J1+J8
        J3=J3+J8
        J4=J4+J8
        IPCNT = IPCNT + 1
        IF (IPCNT - INO)   100,97,100
 97     HDFLG = 0
 100    CONTINUE
        GO TO 89
 C      SETUP  LEAST SQUARE HEADINGS .
 110    J1 = 1
        J2 = 2
        J4 = 3
        J6 = 7
        DO     130     KK =1 , KCY
        IF   (HDFLG)   112,111,112
 111    WRITE OUTPUT TAPE IO,905, IPAG
        WRITE OUTPUT TAPE IO,909
        IPAG = IPAG + 1
        HDFLG  = 1
        IPCNT = 0
 C      WRITE DATA
 112    J7 =1
        WRITE OUTPUT TAPE IO,910,IBAT,DATA(J1),HED(1),(DATA(J3),J3=J4,J6),
       1DATA(J2)
 114    IF(J7 - K2) 113,117,113
 113    J7 = J7+1
        J6 = J6 +5
        J4 = J4 +5
        WRITE OUTPUT TAPE IO,911,  HED(J7), (DATA(J3),J3=J4,J6)
        GO TO 114
 C      FIELDS COMPLETED TEST  PSEUDO
 117    IF( IFLG - IP )    122,118,122
 118    J6 = J6 + 8
        J7 = J7 + 1
        J4 = J4 + 5
        ICT = 0                                                            1
 102    WRITE OUTPUT TAPE IO,912,HED(J7),(DATA(J3),J3=J4,J6)               1
        IF(ICT)  103,101,103                                               1
```

3b. Data List Routine Source Statements (Continued)

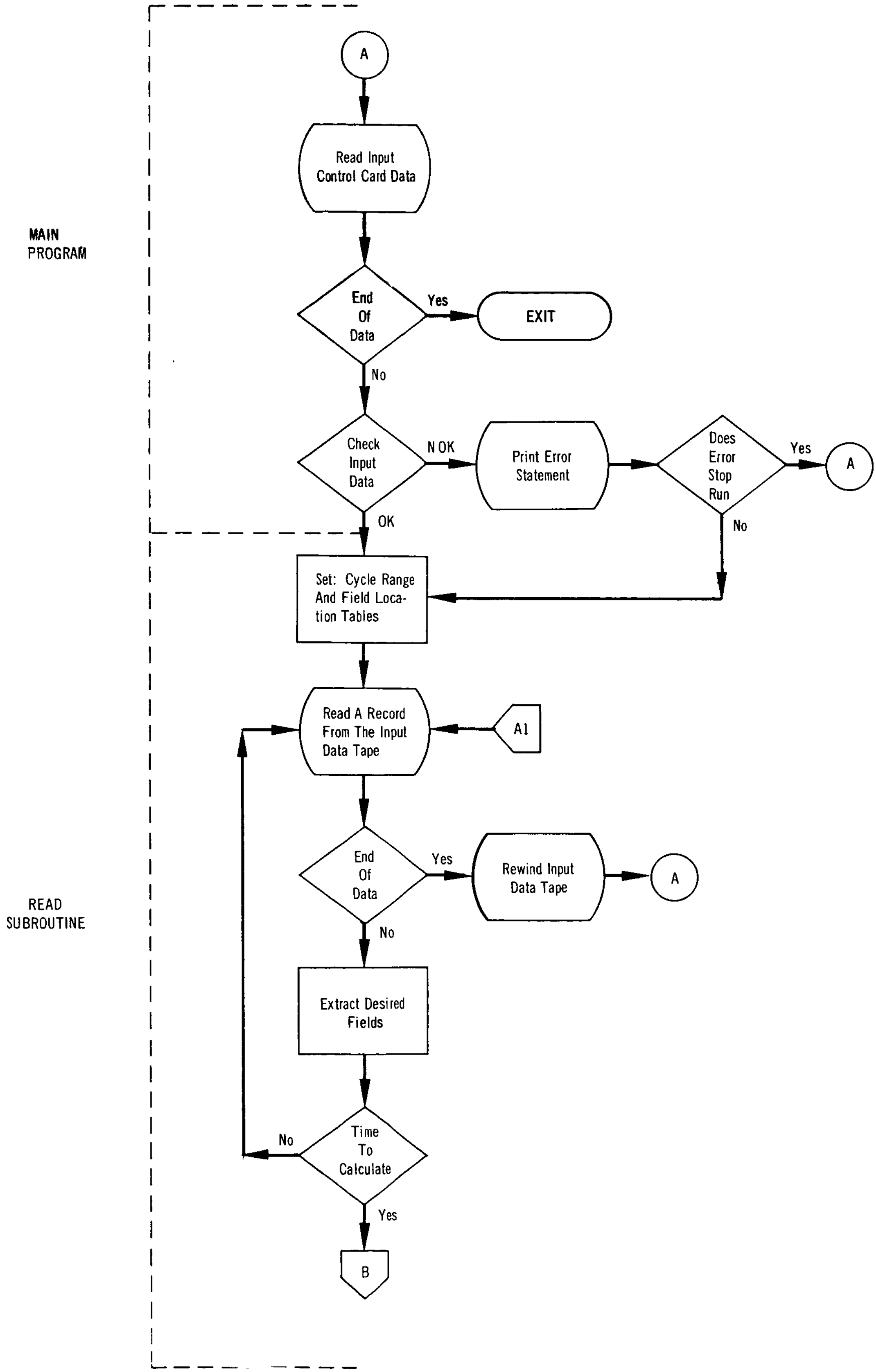
```
 101   ICT = 1                                                               1
       J6 = J6 + 8                                                           1
       J4 = J4 + 8                                                           1
       J7 = J7 + 1                                                           1
       GO TO 102                                                             1
 103   J4 = J4 + 10                                                          1
       GO TO 123
 122   J4 = J4 + 7
 123   J6 = J6 + 7
       J1 = J1 + J5
       J2 = J2 + J5
       WRITE OUTPUT TAPE IO, 913
       IPCNT = IPCNT +1
       IF(IPCNT - INO )  130,124,130
 124   HDFLG = 0
 130   CONTINUE
       GO TO 89
 140   REWIND IPRT
       WRITE OUTPUT TAPE IO,914
       PRINT 920
       PAUSE
       GO TO 1
 900   FORMAT (1H120X,11HREQUEST NO.I4,36H. PRINT OUT OF A RETRIEVED DATA
      1 TAPE///)
 901   FORMAT (6X55HINCORRECT INPUT TAPE. MOUNT CORRECT TAPE AND PUSH STA
      1RT)
 902   FORMAT (1H120X,11HREQUEST NO.I4,43H. PRINT OUT OF A PSEUDO GENERAT
      1ED DATA TAPE///)
 903   FORMAT (1H020X,31HPSEUDO SOURCE STATEMENTS FOLLOW///)
 904   FORMAT (10X,12A6)
 905   FORMAT (1H125X,13HB A T T E R Y11X, 7HT E S T11X, 7HD A T A34X,
      1 4HPAGEI5//)
 906   FORMAT (12H BATT CYC   1H14X,8(A6,8X),1H1/ 12X,1H24X,8(A6,8X),1H2/
      1 12X,1H34X,8(A6,8X),1H3/12X,1H44X,8(A6,8X),1H4/12X,1H54X,8(A6,8X),
      1 1H5/12X,1H64X,2(A6,8X))
 908   FORMAT (2X,I2,    I5,3X1H1 1P8E14.5,4X,1H1/12X,1H28E14.5,4X,1H2/
      1 12X,1H38E14.5,4X,1H3/12X,1H48E14.5,4X,1H4/12X,1H58E14.5,4X,1H5/
      1 12X,1H62E14.5)
 909   FORMAT (84H BATT    CYC    FLD NA     STD. DEV.           X3
      1X2              X                    C24X,10HTIME  DUR.//)
 910   FORMAT (2X,I2,3X,I5,2X,A6, 1P5E14.5,14X,E14.5 )
 911   FORMAT (14X,A6,1P5E14.5)
 912   FORMAT (14X,A6,1P8E14.5)
 913   FORMAT (18X)
 914   FORMAT (1H120X,47H............PRINT REQUEST COMPLETED............)
 915   FORMAT (1H020X,50HFIELDS NOT USED WILL HAVE THE VALUE 10 TO THE 36
      1TH)
 920   FORMAT (5X,45HPUSH START IF ANOTHER TAPE IS TO BE PROCESSED)
       END
```

4. Statistical Analysis Routine

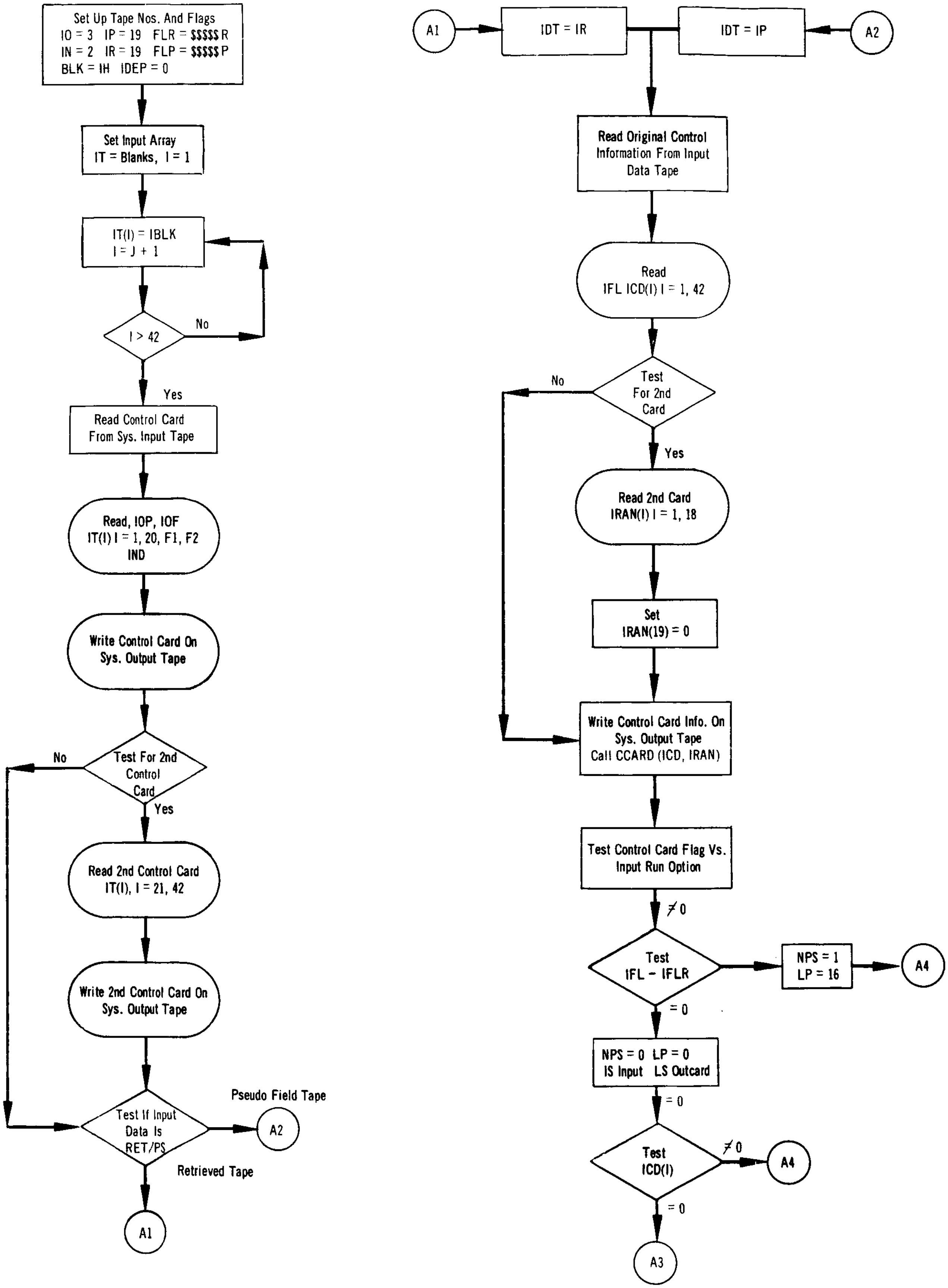
4. General Flow Charts



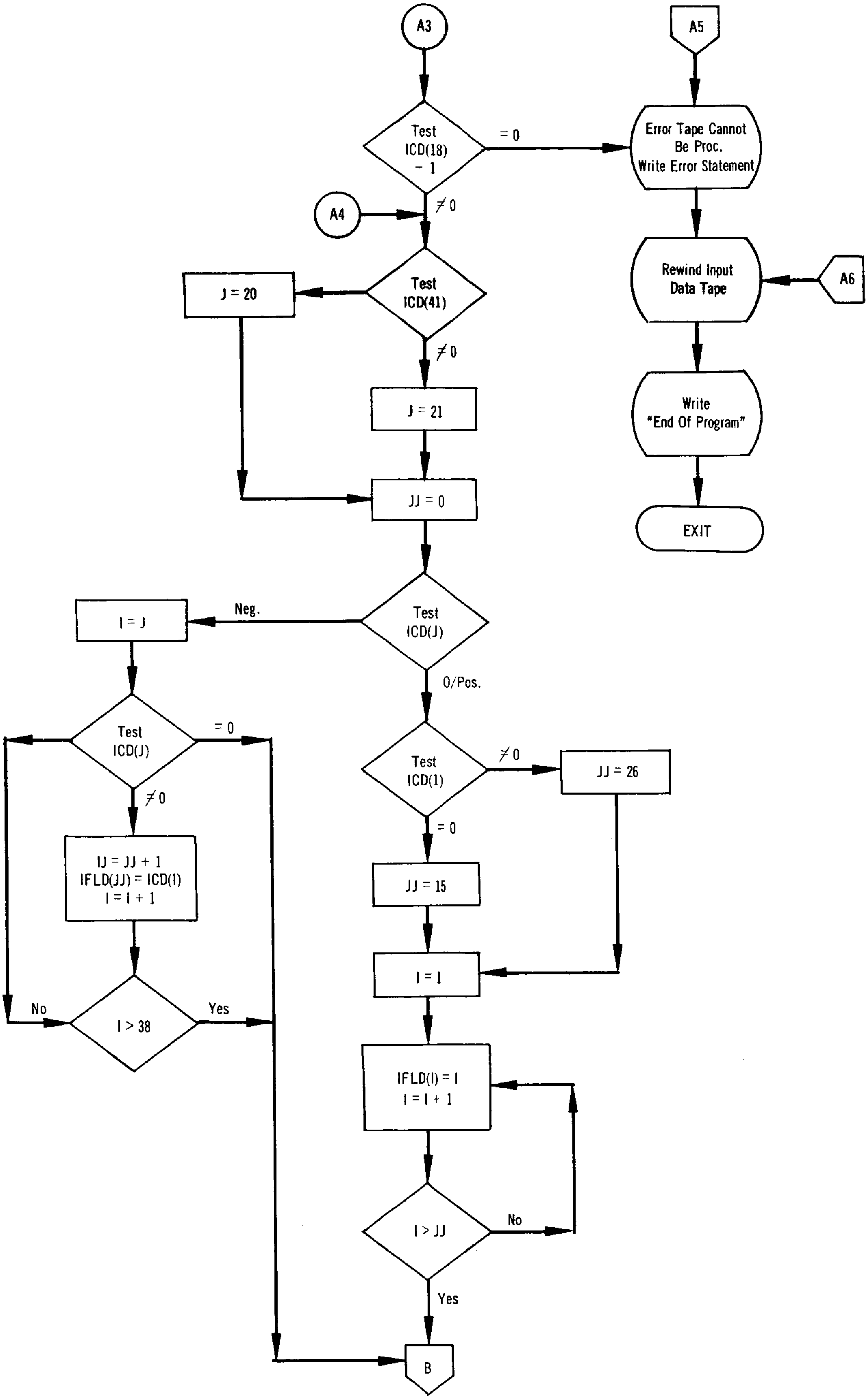
4. General Flow Charts (Continued)



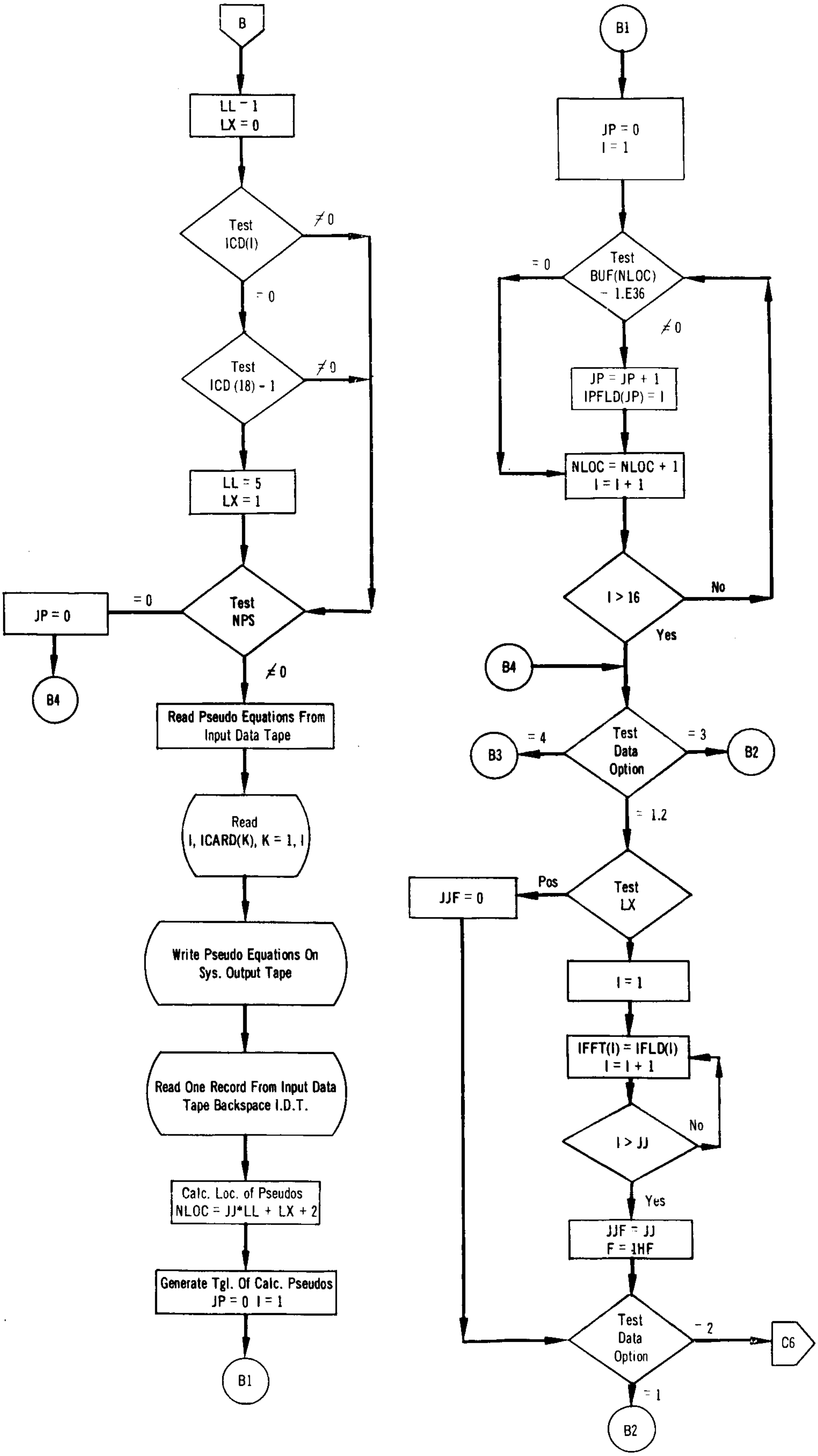
4a. Statistical Analysis Routine Flow Charts – Main Program



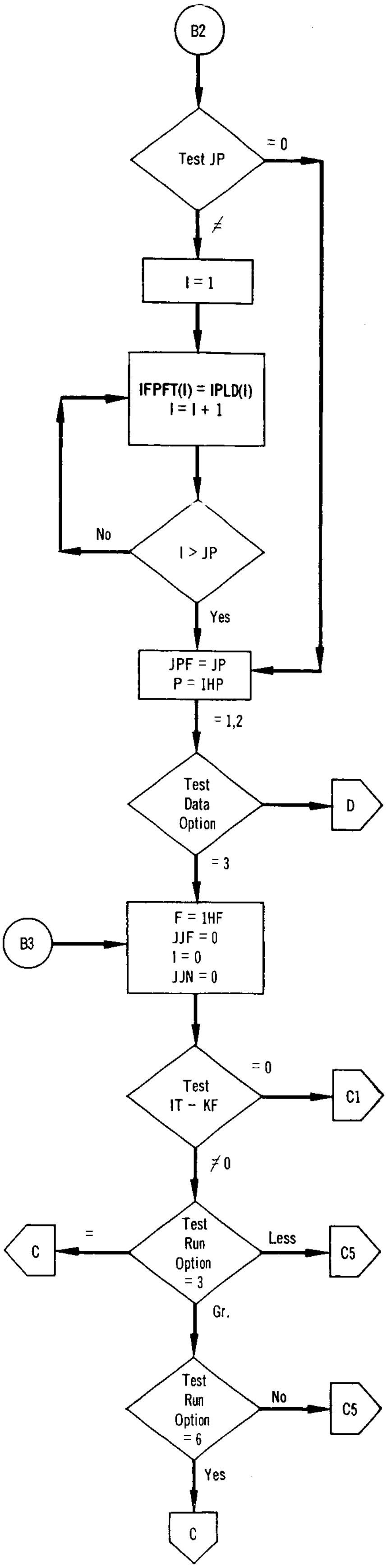
4a. Statistical Analysis Routine Flow Charts – Main Program (Continued)



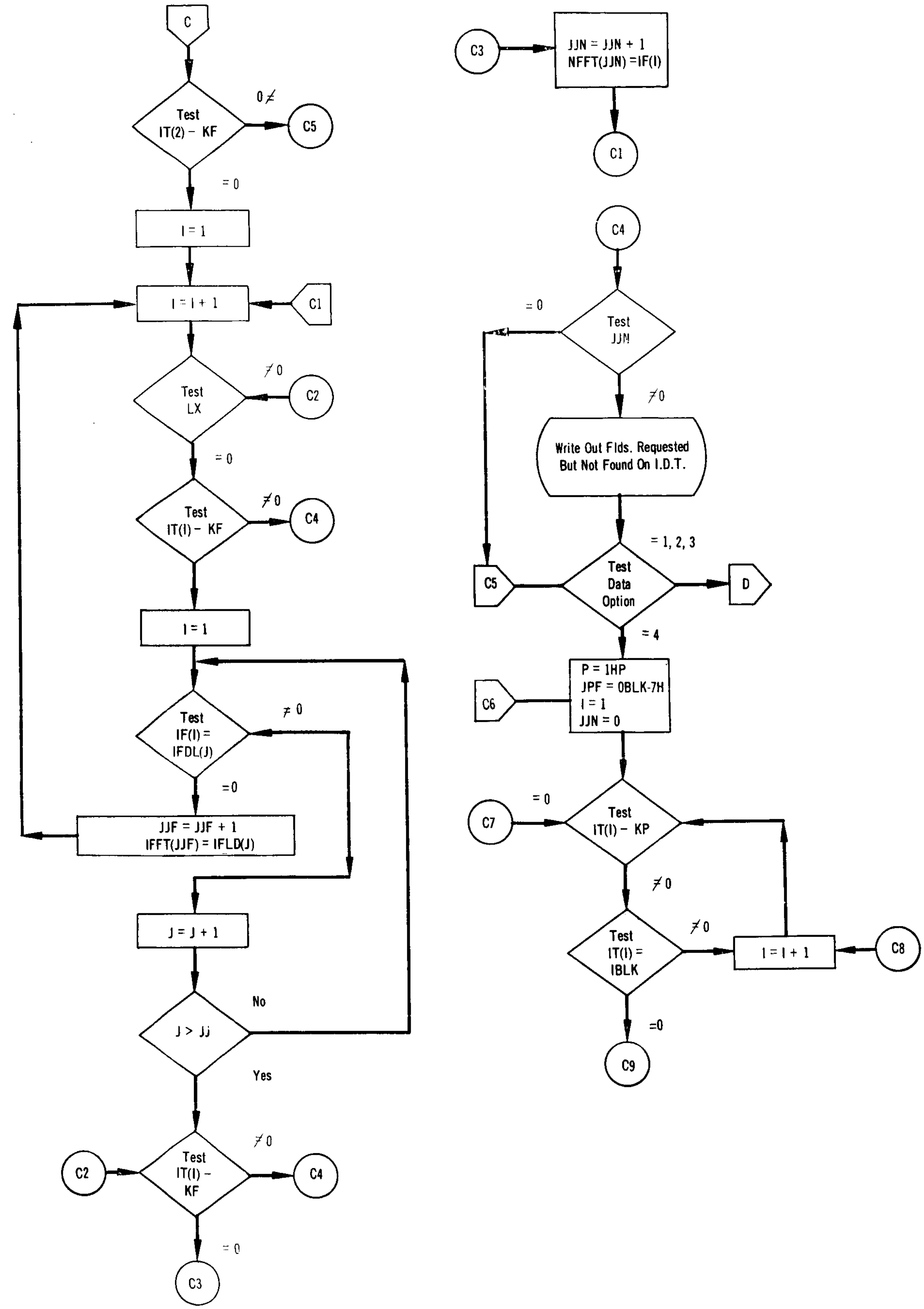
4a. Statistical Analysis Routine Flow Charts – Main Program (Continued)



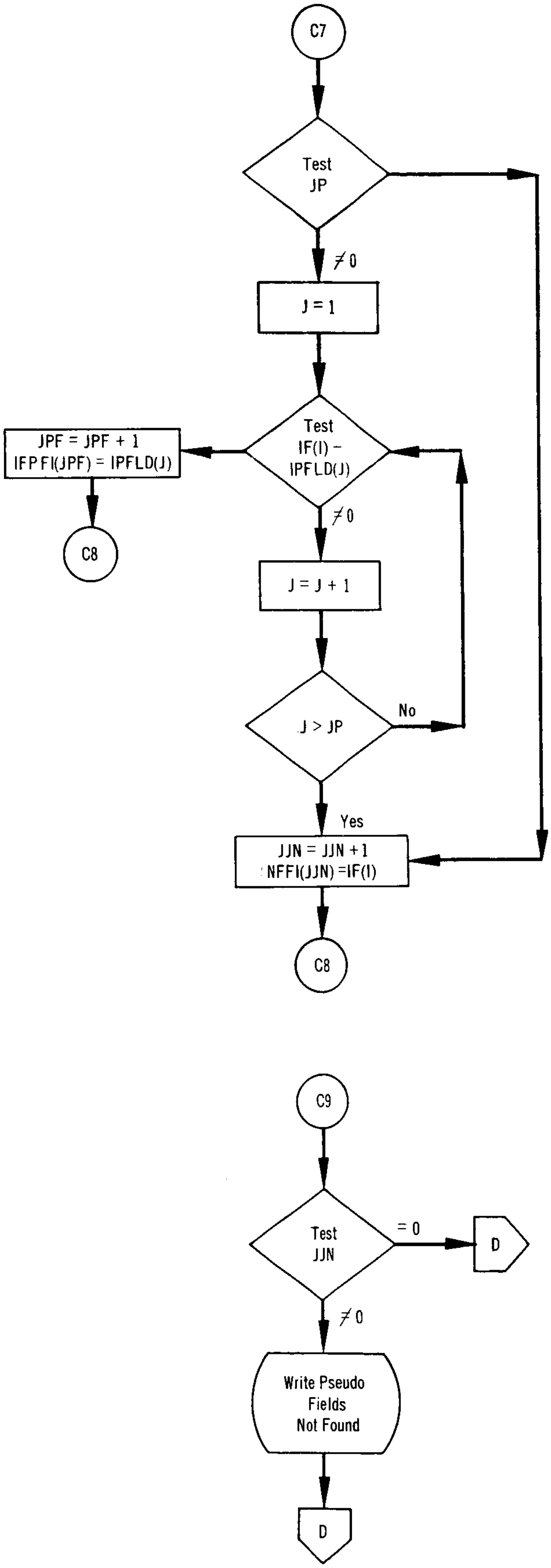
4a. Statistical Analysis Routine Flow Charts – Main Program (Continued)



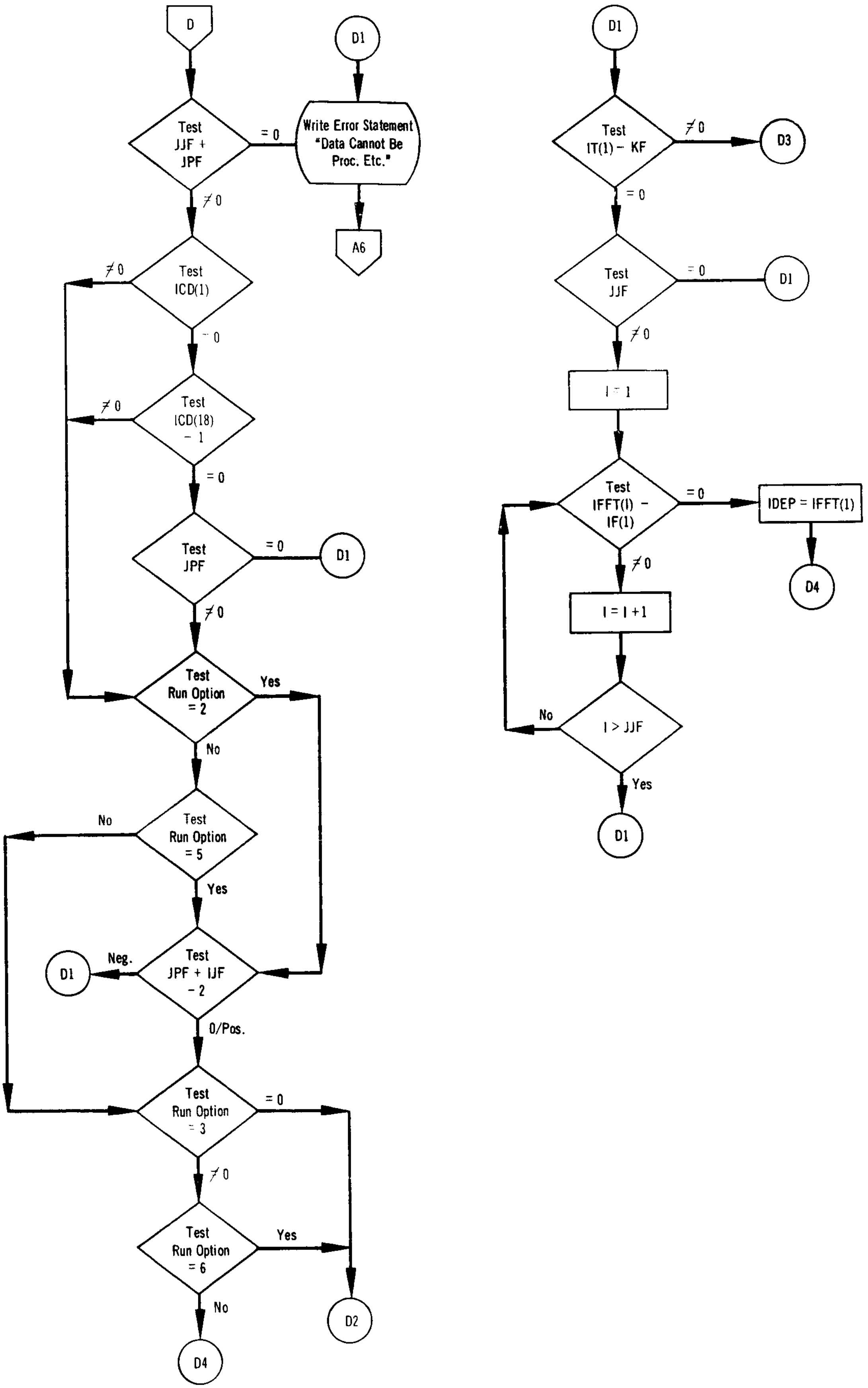
4a. Statistical Analysis Routine Flow Charts – Main Program (Continued)



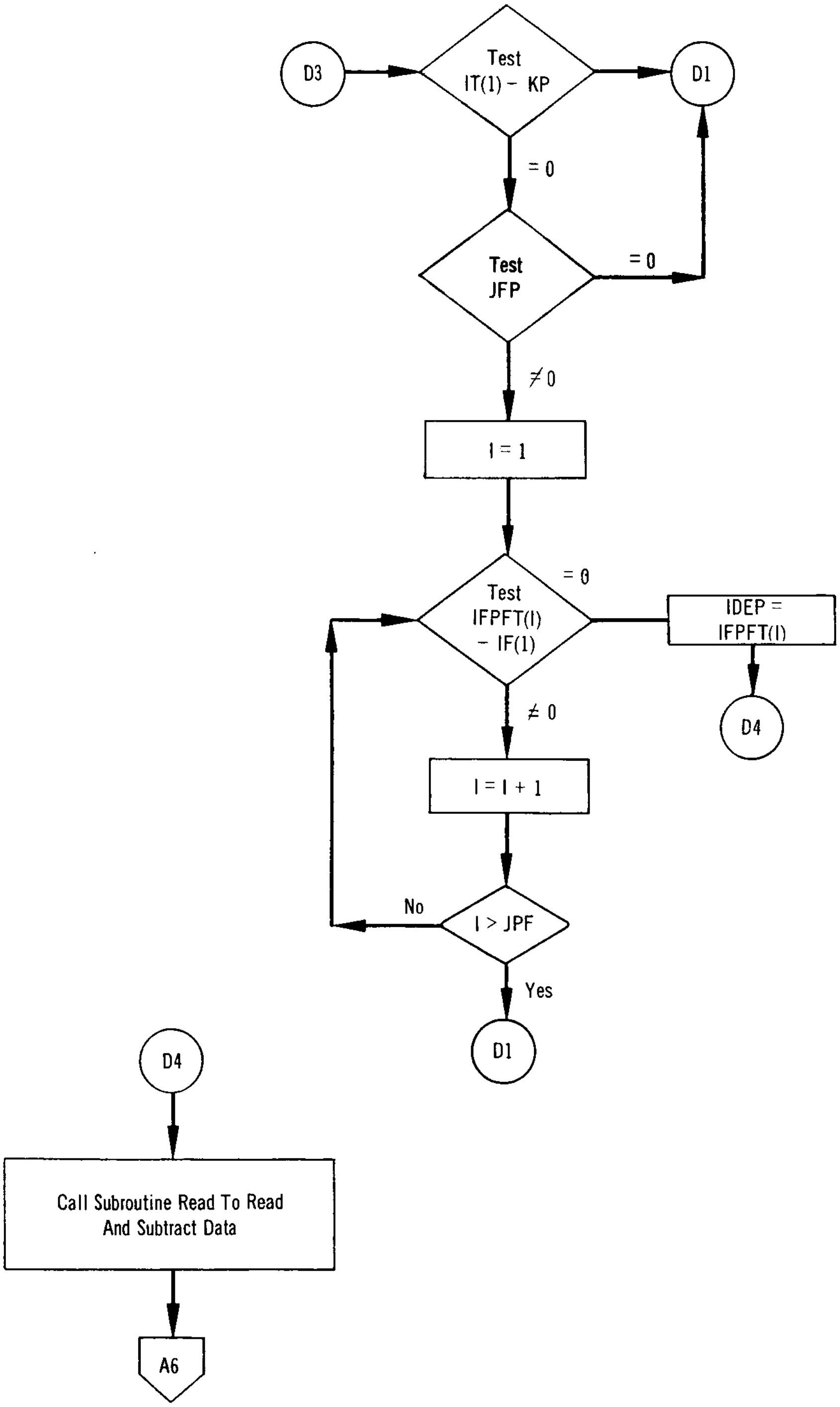
4a. Statistical Analysis Routine Flow Charts – Main Program (Continued)



4a. Statistical Analysis Routine Flow Charts – Main Program (Continued)



4a. Statistical Analysis Routine Flow Charts – Main Program (Continued)



4a. Statistical Analysis Routine Source Statements – Main Program

```
*       CARDS COLUMN
*       SYMBOL TABLE
*       LIST8
*       LABEL
C MAIN PROGRAM FOR FINAL STAT. ANALYSIS ROUTINE.
        COMMON BUF,IT,IF,ICD,ICARD,IFLD,IPFLD,IFFT,IFPFT,NFFT,END,P,F,BLK,
       1 FLP,FLR,JJ,JP,JJF,JPF,IDEP,IOP,IOF,F1,F2,IND,LL,LX,LP,IBATT,I1,
       2 ICNT,KCYC,IRAN
        COMMON JJL,ICYC,R,BETA,SWIX,LCYC,MCYC,L,I,J,NRAN,IBAT,NBAT,L1,L2,
       1 L3,K,KK,K2,K3,NOVAR,NOBS,NCYC,X,XBAR,SIGMA,SB,V,NM1,EMSQ,M,SUMWT,
       2 TOL,SWICH,SITCH,PHI,ISIP,SY,VMIN,VMAX,NMAX,NMIN,BETAZ,KMUL,JUMP,
       3 FINK,FF,IPHI,II,JJN
        COMMON LSCYC,YSY,ANN,TEMP,IEM,IEM2,IDT,IFL,IN,IO,IR,NLOC,NPS,
       1 RHED,HED,RBETA,RSB,END,OUT,PP,JA,IP,LC
        DIMENSION BUF(4500),IT(42),IF(42),ICD(42),ICARD(1200),IFLD(42),
       1 IPFLD(16),IFFT(42),IFPFT(42),NFFT(42),IRAN(19)
        DIMENSION JJL(42),ICYC(20),R(50,50),S(50,50),A(50,50),B(50,50),
       1C(50,50),X(50),XBAR(50),SB(50),SWTX(50),IBU(4500),BETA(50)
        DIMENSION V(50),SIGMA(50),FF(50),HED(50),RHED(50),RBETA(50),
       1 RSB(50),END(26),OUT(15),PP(16)
        EQUIVALENCE  (END,IEND),(FLP,IFLP),(FLR,IFLR),(F,KF),(P,KP),
       1 (BLK,IBLK)
        EQUIVALENCE (BUF,IBU), (R,S,A,B,C) , (X,XBAR), (SWTX,SB)
C.....SET UP TAPE NOS.
        IO = 3
        IN = 2
        IP = 17
        IR = 19
        FLR = 6H$$$$$R
        FLP = 6H$$$$$P
        BLK = 1H
C.....SET INPUT ARRAY IT = BLANKS.
1       DO 2 L =1,42
2       IT(I) = IBLK
        IDEP = 0
C.....READ CONTROL CARD FROM SYSTEM INPUT TAPE (IN).
        READ INPUT TAPE IN,1000,IOP,IOF,(IT(I),IF(I),I=1,20),F1,F2,IND
1000    FORMAT (2I1,1X,20(A1,I2),2F4.0,I1)
C.....WRITE FIRST CONTROL CARD ON SYS. OUTPUT TAPE (IO).
        WRITE OUTPUT TAPE IO,1001, IOP,IOF,(IT(I),IF(I),I=1,20),F1,F2,IND
1001    FORMAT(20HIRUN TYPE  DATA OPT.,24X,34HSPECIFIED FIELDS AND PSEUDO
       1FIELDS,26X, 2HF1,4X,2HF2,6X,3HIND //4X,I1,9X,I1,7X,20(A1,I2,1X),
       21X,F4.0,2X,F4.0,6X,I1 )
        IF (IOP-9) 5,4,5
4       CALL FINI
C.....TEST FOR 2ND. CONTROL CARD.
5       IF (IND) 110,110,100
100     READ INPUT TAPE IN,1002,(IT(I),IF(I),I=21,42)
1002    FORMAT(22(A1,I2))
C.....WRITE 2ND CONTROL CARD ON SYS. OUTPUT TAPE (IO).
        WRITE OUTPUT TAPE IO,1003, (IT(I),IF(I),I= 21,42)
1003    FORMAT(//44X42HSPECIFIED FIELDS AND PSEUDO FIELDS (CONT.)//
       1 22X,22(A1,I2,1X) )
C.....DETERMINE IF INPUT DATA NO. IS 19 OR 17.
110     IF (IOP-3) 120,120,130
120     IDT = IR
        GO TO 140
130     IDT = IP
```

4a. Statistical Analysis Routine Source Statements — Main Program (Continued)

```
C.....READ ORIGINAL CONTROL INFORMATION FROM INPUT DATA TAPE (IDT).
140   READ TAPE IDT, IFL,ICD
C.....TEST IF MORE CONTROL INFORMATION EXISTS.
      IF(ICD(39)-1) 160,150,160
150   READ TAPE IDT , (IRAN(I),I=1,18)
      IRAN(19) = 0
C.....WRITE CONTROL INFORMATION ON TAPE IO.
160   CALL CCARD (ICD,IRAN)
C.....TEST CONTROL INFO. FLAG VS. INPUT RUN OPTION
      IF (IFL-IFLR) 180,170,180
170   NPS = 0
      LP = 0
C.....TEST IF IDT IS OUTCARD L.S. TYPE.
      IF (ICD(1)) 190,175,190
175   IF (ICD(18)-1) 190,176,190
C.....ERROR TAPE CANNOT BE PROCESSED.
176   WRITE OUTPUT TAPE IO,1004
1004  FORMAT(90H0INPUT DATA TAPE IS OUTCARD LEAST SQ. WITHOUT PSEUDO FIE
     1LDS, HENCE NO PROCESSING POSSIBLE.)
177   REWIND IDT
      WRITE OUTPUT TAPE IO,1116
1116  FORMAT(1H0,32X,66H********** END OF STATISTICAL ANALYSIS PROGRAM O
     1UTPUT. **********  )
      PRINT 1117
1117  FORMAT (40H0  PUSH START TO CONTINUE WITH NEXT RUN.  )
      PAUSE
      GO TO 1
180   NPS = 1
      LP = 16
C.....DETERMINE WHAT INFO. IS PRESENT ON IDT.
190   IF (ICD(41)) 210,200,210
200   J = 20
      GO TO 220
210   J = 21
220   JJ = 0
      IF (ICD(J)) 260,260,230
230   DO 250 I=J,38
      IF(ICD(I)) 310,310,240
240   JJ = JJ +1
250   IFLD(JJ)= ICD(I)
      GO TO 310
C.....DETERMINE IF IDT IS ENDCARD OR OUTCARD.
260   IF (ICD(1)) 270,280,270
270   JJ = 26
      GO TO 290
280   JJ = 15
290   DO 300 I=1,JJ
300   IFLD(I) = I
C.....DETERMINE IF IDT IS L.S. TYPE.
310   LL = 1
      LX = 0
      IF (ICD(1)) 340,320,340
320   IF (ICD(18)-1) 340,330,340
330   LL = 5
      LX = 1
C.....DOES IDT CONTAIN PSEUDO FIELDS.
340   IF (NPS) 341,410,341
C.....READ PSEUDO EQUATIONS FROM IDT.
```

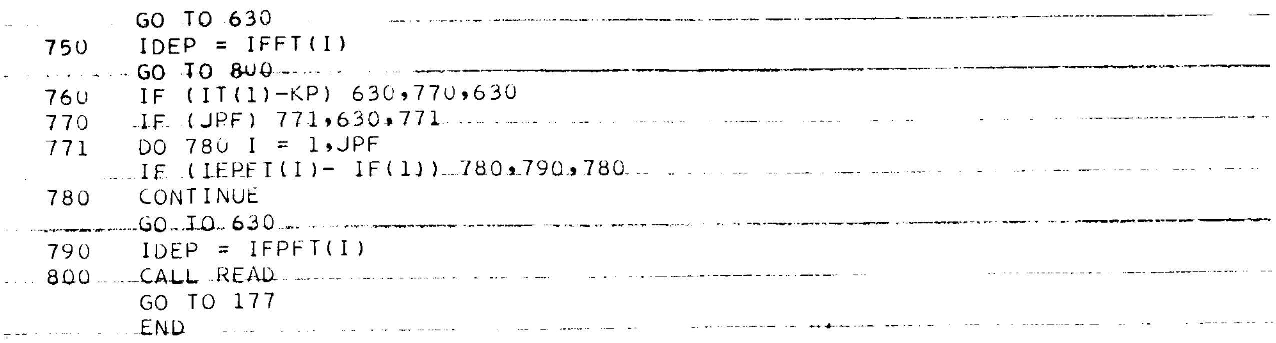
4a. Statistical Analysis Routine Source Statements – Main Program (Continued)

```
 341    READ TAPE IDT , I,(ICARD(K),K=1,I)
C.....WRITE PSEUDO EQUATIONS ON TAPE IO.
 370    WRITE OUTPUT TAPE IO,1005, (ICARD(K),K=1,I)
 1005   FORMAT(25H0   PSEUDO EQUATION CARDS //(1X,12A6))
C.....READ ONE RECORD FROM IDT AND SET UP PSEUDO FIELD TABLE (IPFLD)
        READ TAPE IDT,I1,ICNT,IBAT,(BUF(I),I=1,ICNT)
        BACKSPACE IDT
C.....CALCULATE LOCATION OF PSEUDOS.
        NLOC = (JJ*LL)+LX+2
C.....GENERATE TABLE OF CALCULATED PSEUDOS.
        JP = 0
        DO 400 I=1,16
        IF (BUF(NLOC)-1.E36) 380,390,380
 380    JP = JP+1
        IPFLD(JP) = I
 390    NLOC = NLOC +1
 400    CONTINUE
        GO TO 420
 410    JP = 0
C.....CHECK REQUESTED FIELDS VS. FIELDS ON IDT.
 420    GO TO  (421,421,440,460),IOF
C.....MOVE ALL FIELDS INTO FINAL FIELD TABLE (IFFT).
 421    IF (LX) 423,423,422
 422    JJF = 0
        GO TO 432
 423    DO 430 I = 1,JJ
 430    IFFT(I)= IFLD(I)
 431    JJF = JJ
        E = 1HF
 432    GO TO (440,540),IOF
C.....MOVE ALL PSEUDO FIELDS INTO FINAL PSEUDO FIELD TABLE (IFPFT).
 440    IF (JP) 441,451,441
 441    DO 450 I =1,JP
 450    IFPFT(I)= IPFLD(I)
 451    JPF = JP
        P = 1HP
        GO TO (620,620,460),IOF
C.....CHECK SPECIFIED FIELDS. MOVE SPECIFIED FIELDS INTO FINAL FIELD TB.
C.....PRINT LIST OF FIELDS REQUESTED BUT NOT FOUND ON IDT.
 460    F = 1HF
        JJF = 0
        I = 0
        JJN = 0
        IF (IT-KF) 461,470,461
C.....IS RUN A MULTIPLE REGRESSION TYPE.
 461    IF (IOP-3) 530,463,462
 462    IF (IOP-6) 530,463,530
 463    IF (IT(2)-KF) 530,464,530
 464    I = 1
 470    I = I + 1
        IF (LX) 471,471,501
 471    IF (IT(I)-KF) 510,480,510
 480    DO 500 J=1,JJ
        IF (IF(I)-IFLD(J)) 500,490,500
 490    JJF = JJF +1
        IFFT(JJF)= IFLD(J)
        GO TO 470
 500    CONTINUE
```

4a. Statistical Analysis Routine Source Statements – Main Program (Continued)

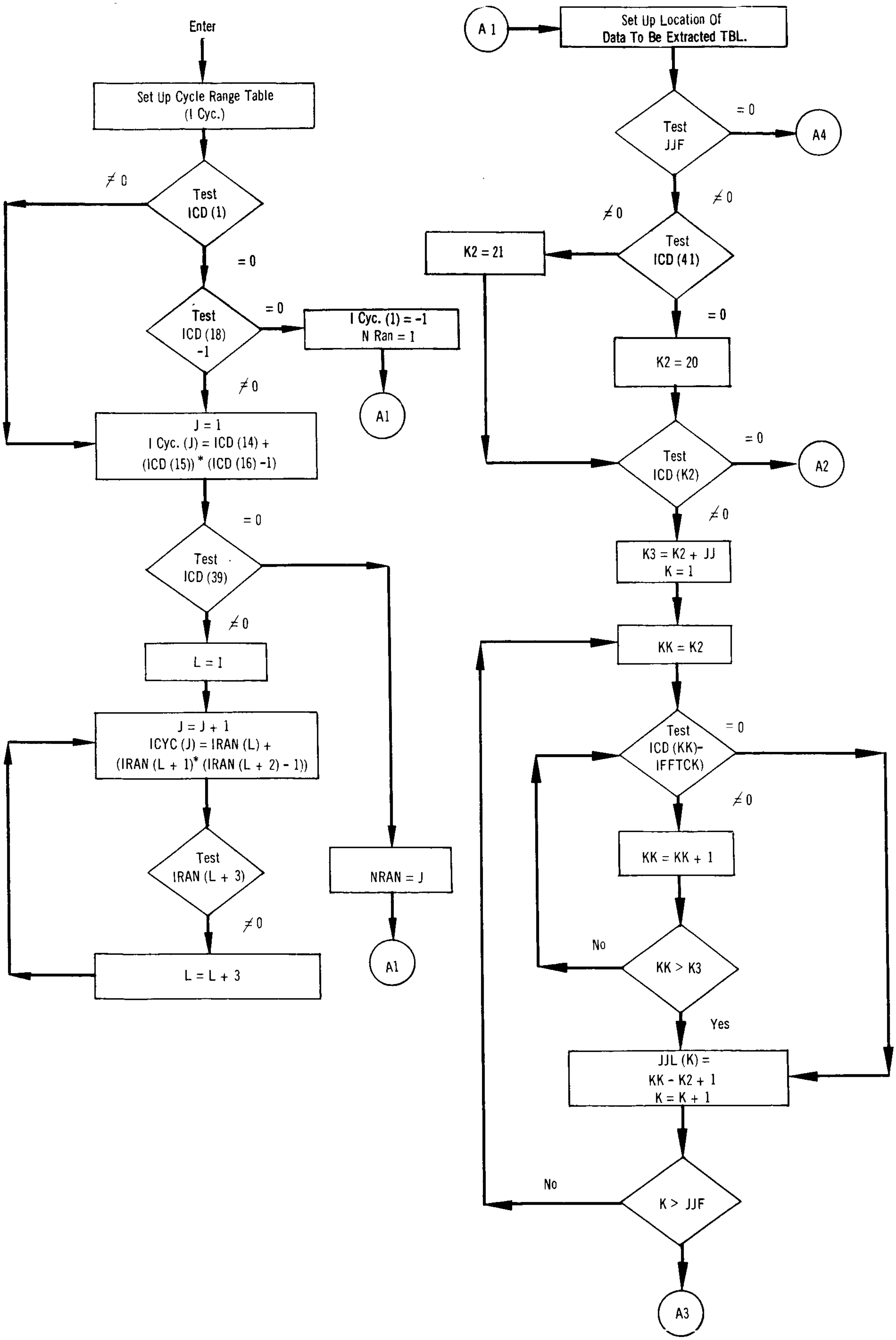
```
 501    IF (IT(I) - KF) 510,502,510
 502    JJN = JJN + 1
        NFFT(JJN)= IF(I)
        GO TO 470
 C.....LIST THOSE FIELDS NOT FOUND.
 510    IF (JJN) 520,530,520
 520    WRITE OUTPUT TAPE IO,1006, (NFFT(I),I=1,JJN)
 1006   FORMAT (63H0 THE FOLLOWING FIELDS WERE NOT FOUND ON THE INPUT DATA
       1 TAPE.../ 25I4 )
 530    GO TO (620,620,620,540),IOF
 C.....CHECK SPEC. PSEUDO FIELDS. MOVE SPEC. PSEUDOS INTO FINAL TABLE.
 C.....PRINT LIST OF FIELDS REQUESTED BUT NOT FOUND ON IDT.
 540    P = 1HP
        JPF = 0
        I = 1
        JJN = 0
        BLK = 1H
 541    IF (IT(I)-KP) 550,570,550
 550    IF( IT(I)-IBLK) 560,600,560
 560    I = I +1
        GO TO 541
 570    IF (JP) 571,591,571
 571    DO 590 J =1,JP
        IF (IF(I)- IPFLD(J)) 590,580,590
 580    JPF = JPF+1
        IFPFT(JPF)= IPFLD(J)
        GO TO 560
 590    CONTINUE
 591    JJN = JJN + 1
        NFFT(JJN) = IF(I)
        GO TO 560
 C.....LIST THOSE PSEUDOS NOT FOUND.
 600    IF (JJN) 610,620,610
 610    WRITE OUTPUT TAPE IO,1007, (NFFT(I),I=1,JJN)
 1007   FORMAT (70H0 THE FOLLOWING PSEUDO FIELDS WERE NOT FOUND ON THE INP
       1UT DATA TAPE.../ 25I4)
 C.....CAN ANY DATA BE PROCESSED.
 620    IF ( JJF + JPF ) 640,630,640
 C.....NO DATA CAN BE PROCESSED. WRITE ERROR STATEMENT.
 630    WRITE OUTPUT TAPE IO,1008
 1008   FORMAT(43H0 DATA CANNOT BE PROCESSED. RUN TERMINATED.)
        GO TO 177
 C.....TEST IF ANY PSEUDOS ON OUTCARD L.S. TAPE.
 640    IF (ICD(1)) 670,650,670
 650    IF (ICD(18)-1) 670,660,670
 660    IF (JPF ) 670,630,670
 C.....CAN LINEAR CORRELATION BE RUN.
 670    IF (IOP-2) 800,690,680
 680    IF (IOP-4) 690,800,690
 690    IF (JPF+JJF-2) 630,700,700
 C.....CAN MULTIPLE REGRESSION BE RUN.
 700    IF (IOP-3) 710,720,710
 710    IF (IOP-6) 800,720,800
 720    IF (IT(1)-KF) 760,730,760
 730    IF (JJF) 731,630,731
 731    DO 740 I = 1,JJF
        IF (IFFT(I)-IF(1)) 740,750,740
 740    CONTINUE
```

4a. Statistical Analysis Routine Source Statements – Main Program (Continued)

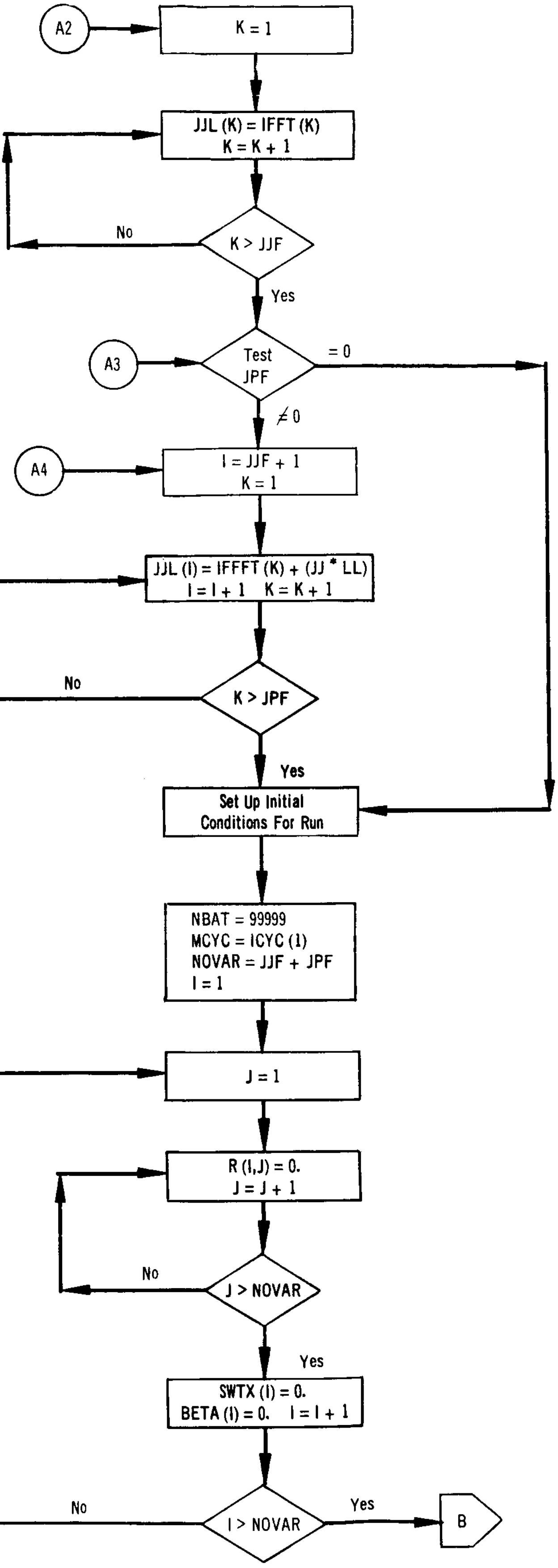


```
        GO TO 630
 750    IDEP = IFFT(I)
        GO TO 800
 760    IF (IT(1)-KP) 630,770,630
 770    IF (JPF) 771,630,771
 771    DO 780 I = 1,JPF
        IF (IEPFT(I)- IF(1)) 780,790,780
 780    CONTINUE
        GO TO 630
 790    IDEP = IFPFT(I)
 800    CALL READ
        GO TO 177
        END
```

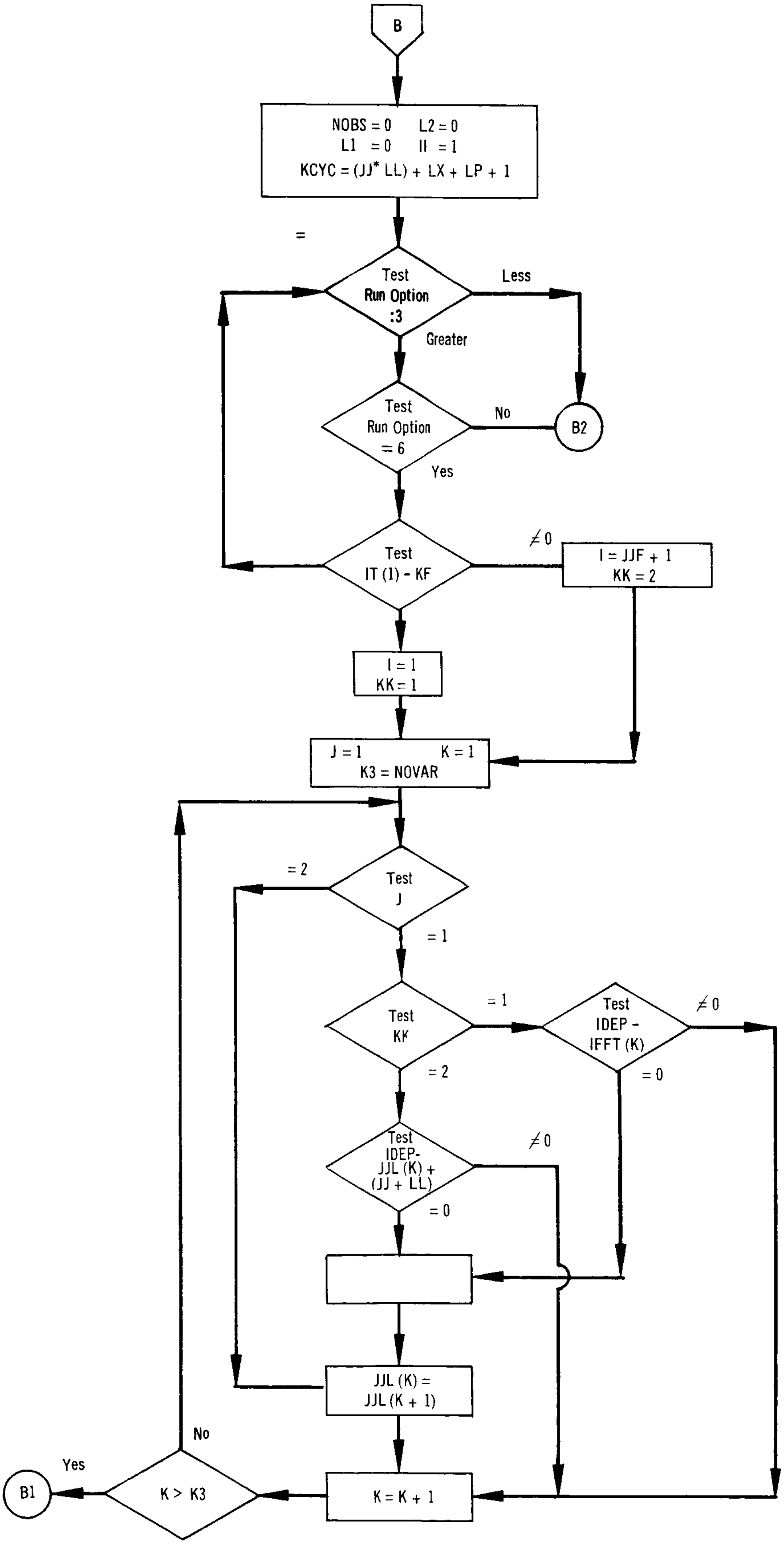
4b. Statistical Analysis Routine Flow Chart – Subroutine Read



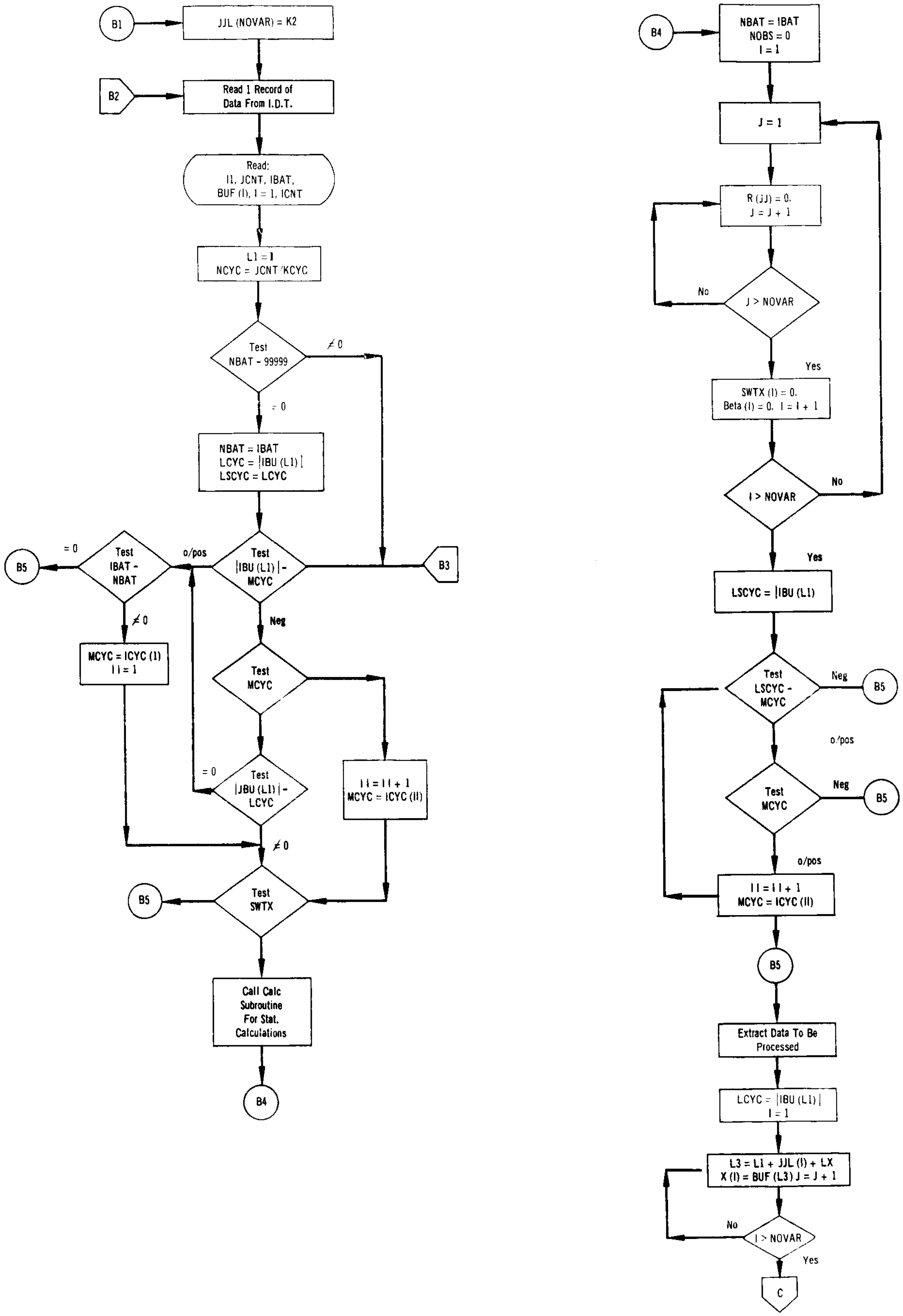
4b. Statistical Analysis Program Flow Chart – Subroutine Read (Continued)



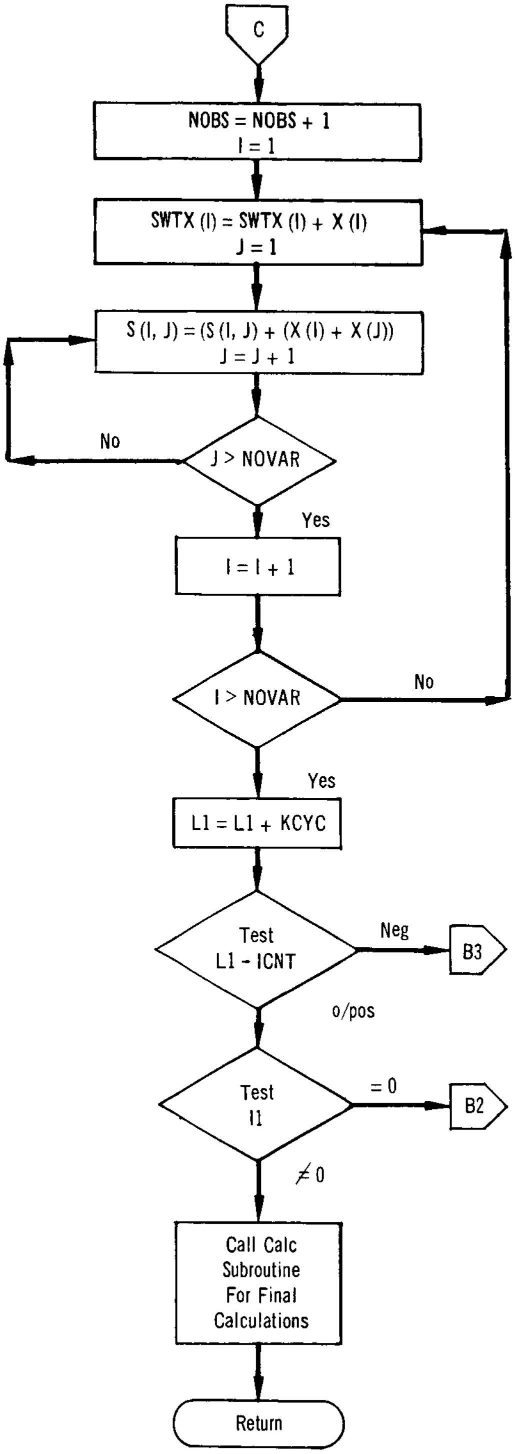
4b. Statistical Analysis Program Flow Chart – Subroutine Read (Continued)



4b. Statistical Analysis Program Flow Chart – Subroutine Read (Continued)



4b. Statistical Analysis Program Flow Chart – Subroutine Read (Continued)



4b. Statistical Analysis Routine Source Statements – Subroutine Read

```
*       CARDS COLUMN
*       LIST8
*       SYMBOL TABLE
*       LABEL
CREAD   SUBROUTINE FOR STAT. ANALYSIS PROGRAM.
        SUBROUTINE READ
        COMMON BUF,IT,IF,ICD,ICARD,IFLD,IPFLD,IFFT,IFPFT,NFFT,END,P,F,BLK,
      1 FLP,FLR,JJ,JP,JJF,JPF,IDEP,IOP,IOF,F1,F2,IND,LL,LX,LP,IBATT,I1,
      2 ICNT,KCYC,IRAN
        COMMON JJL,ICYC,R,BETA,SWTX,LCYC,MCYC,L,I,J,NRAN,IBAT,NBAT,L1,L2,
      1 L3,K,KK,K2,K3,NOVAR,NOBS,NCYC,X,XBAR,SIGMA,SD,V,NM1,EMSQ,M,SUMWT,
      2 TOL,SWTCH,SITCH,PHI,ISTP,SY,VMIN,VMAX,NMAX,NMIN,BETAZ,RMUL,JUMP,
      3 FINK,FF,IPHI,II,JJN
        COMMON LSCYC,YSY,ANN,TEMP,TEM,TEM2,IDT,IFL,IN,IO,IR,NLOC,NPS,
      1 RHED,HED,RBETA,RSD,END,OUT,PP,JA,IB,LC
        DIMENSION BUF(4500),IT(42),IF(42),ICD(42),ICARD(1200),IFLD(42),
      1 IPFLD(16),IFFT(42),IFPFT(42),NFFT(42),IRAN(19)
        DIMENSION JJL(42),ICYC(20),R(50,50),S(50,50),A(50,50),B(50,50),
      1C(50,50),X(50),XBAR(50),SB(50),SWTX(50),IBU(4500),BETA(50)
        DIMENSION V(50),SIGMA(50),FF(50),HED(50),RHED(50),RBETA(50),
      1 RSB(50),END(26),OUT(15),PP(16)
        EQUIVALENCE  (END,IEND),(FLP,IFLP),(FLR,IFLR),(F,KF),(P,KP),
      1 (BLK,IBLK)
        EQUIVALENCE (BUF,IBU), (R,S,A,B,C) , (X,XBAR), (SWTX,SB)
C.....IS THIS A CHARGE-DISCHARGL RUN.
        IF (ICD(17)-2) 111,110,111
 110    IPH = 1
        M1 = 0
        GO TO 112
 111    IPH = 0
C.....SET UP CYCLE RANGL TABLE(ICYC).
 112    IF (ICD(1)) 130,120,130
 120    IF (ICD(18)-1) 145,130,145
 130    J = 1
        ICYC(J) = ICD(14) + (ICD(15)*(ICD(16)-1))
        IF (ICD(39)) 140,146,140
 140    L = 1
 141    J = J + 1
        ICYC(J) = IRAN(L) + (IRAN(L+1)*(IRAN(L+2)-1))
        IF (IRAN(L+3)) 142,143,142
 142    L = L +3
        GO TO 141
 143    NRAN = J
        GO TO 150
 145    ICYC(1)= -1
 146    NRAN = 1
C.....SET LOCATION OF DATA TO BE EXTRACTED TABLE. (JJL).
 150    IF (JJF) 151,260,151
 151    IF (ICD(41)) 160,170,160
 160    K2 = 21
        GO TO 180
 170    K2 = 20
 180    IF (ICD(K2)) 190,230,190
 190    K3 = K2 + JJ
        DO 220 K = 1,JJF
        DO 200 KK=K2,K3
        IF (ICD(KK)-IFFT(K)) 200,210,200
 200    CONTINUE
```

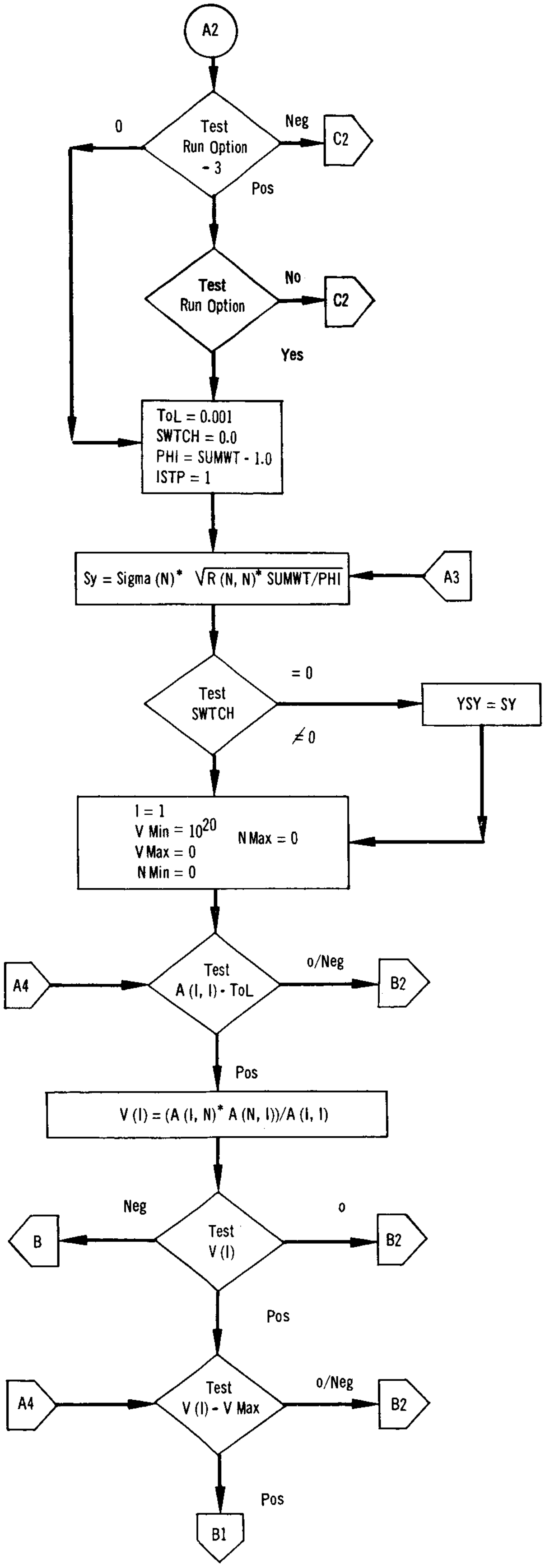
4b. Statistical Analysis Routine Source Statements – Subroutine Read (Continued)

```
210     JJL(K) = KK-K2 +1
220     CONTINUE
        GO TO 250
230     DO 240 K = 1,JJF
240     JJL(K) = IFFT(K)
250     IF (JPF) 260,280,260
260     I = JJF +1
        DO 270 K= 1,JPF
        JJL(I) = IFPFI(K) + (JJ*LL)
270     I = I +1
C.....SET UP INITIAL CONDITIONS FOR RUN.
280     NBAT = 99999
        MCYC = ICYC(1)
        NOVAR = JJF + JPF
281     DO 300 I = 1,NOVAR
        DO 290 J = 1,NOVAR
290     R(I,J) = 0.
        SWTX(I)= 0.
300     BETA(I)= 0.
        NOBS = 0
        L1 = 1
        II = 1
        IF (IPH)  310,3000,3000
C.....COMPUTE NO. OF WORDS IN ONE CYCLE.
3000    KCYC = (JJ*LL) + LX + LP + 1
        L2 = 0
C.....IS THIS M.R. RUN.
        IF (IOP-3) 310,302,301
301     IF (IOP-6) 310,302,310
C.....SET UP DEPENDENT VARIABLE TO BE EXTRACTED LAST.
C.....IS DEP. VAR. A FIELD OR PSEUDO FIELD.
302     IF (IT(1)-KF) 303,304,303
303     I = JJF + 1
        KK = 2
        GO TO 305
304     I = 1
        KK = 1
305     J = 1
        K3 = NOVAR
        DO 3091 K= I,K3
        GO TO (306,3090),J
306     GO TO (307,308),KK
307     IF (IDEP - IFFT(K)) 3091,309,3091
308     IF (IDEP - JJL(K) + (JJ*LL)) 3091,309,3091
309     J = 2
        K2 = JJL(K)
3090    JJL(K) = JJL(K+1)
3091    CONTINUE
        JJL(NOVAR) = K2
C.....READ ONE RECORD OF DATA FROM THE IDT.
310     READ TAPE IDT , I1,ICNT,IBAT,(BUF(I),I=1,ICNT)
C.....COMPUTE NO. OF CYCLES IN THIS RECORD.
        L1 = 1
        NCYC = ICNT/KCYC
C.....TEST IF THIS INITIAL TIME THRU THIS ROUTINE.
320     IF (NBAT-99999) 330,321,330
321     NBAT = IBAT
        LCYC = XABSF(IBU(L1))
```

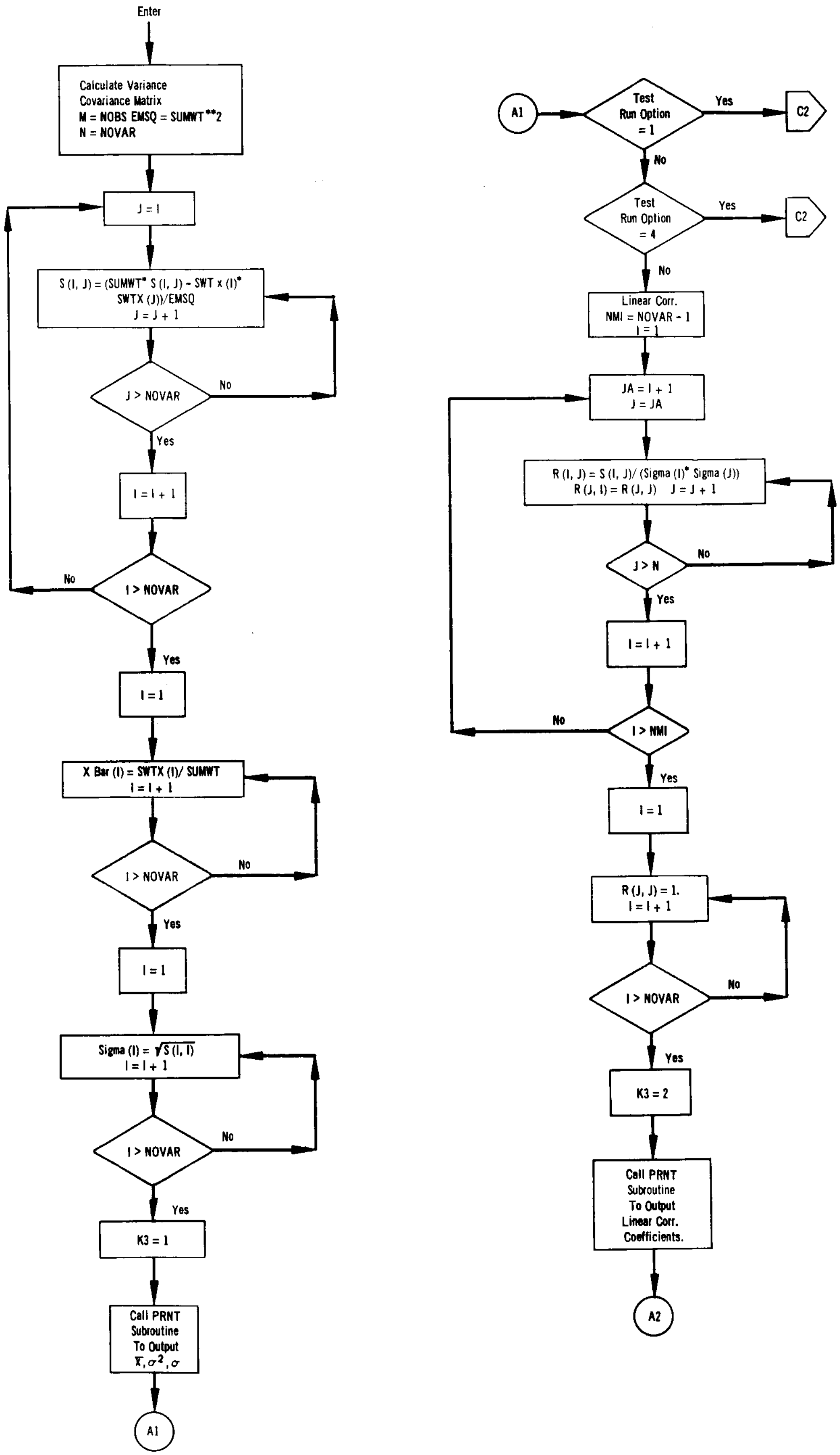
4b. Statistical Analvsis Routine Source Statements – Subroutine Read (Continued)

```
      LSCYC = LCYC
      GO TO 431
330   IF (XABSF(IBU(L1))-MCYC) 380,380,360
360   IF (MCYC) 361,370,370
361   IF (XABSF(IBU(L1))-LCYC) 400,380,400
370   II = II +1
      MCYC = ICYC(II)
      GO TO 400
380   IF (IBAT - NBAT) 390,440,390
390   MCYC = ICYC(1)
      II = 1
C.....TEST IF ANY DATA APPEARED FOR THIS CYCLE RANGE.
400   IF (NOBS) 410,4300,410
C.....CALL CALC SUBROUTINE TO PERFORM CALCULATIONS AND OUTPUT DATA.
410   CALL CALC
C.....ZERO OUT BUFFERS.
      NOBS = 0
      DO 430 I = 1,NOVAR
      DO 420 J = 1,NOVAR
420   R(I,J) = 0.
      SWTX(I)= 0.
430   BETA(I)= 0.
4300  NBAT = IBAT
      LSCYC = XABSF(IBU(L1))
431   IF (LSCYC-MCYC) 440,432,432
432   IF (MCYC) 440,433,433
433   II = II + 1
      MCYC = ICYC(II)
      GO TO 431
C.....EXTRACT DATA TO BE PROCESSED.
440   LCYC = XABSF(IBU(L1))
      IF (IPH) 441,444,442
441   IF (IBU(L1)) 444,471,471
442   IF (IBU(L1)) 443,471,444
443   M1 = 1
      GO TO 471
444   DO 450 I = 1,NOVAR
      L3 = L1 + JJL(I) + LX
450   X(I) =  BUF(L3)
      NOBS = NOBS +1
      DO 470 I = 1,NOVAR
      SWTX(I) = SWTX(I) + X(I)
      DO 460 J = 1,NOVAR
460   S(I,J) = S(I,J) + (X(I) * X(J))
470   CONTINUE
471   L1 = L1 + KCYC
      IF (L1-ICNT) 330,480,480
C.....TEST IF MORE DATA EXISTS ON IDT.
480   IF (I1) 490,310,490
C.....CALCULATE FINAL RESULTS.
490   IF (NOBS) 500,510,500
500   CALL CALC
510   IF (IPH) 560,560,520
520   IF (M1) 560,560,525
525   REWIND IDT
      IPH = -1
      READ TAPE IDT ,DUM
      IF (ICD(39)) 530,540,530
530   READ TAPE IDT ,DUM
540   IF (LP) 550,280,550
550   READ TAPE IDT ,DUM
      GO TO 280
560   RETURN
      END
```

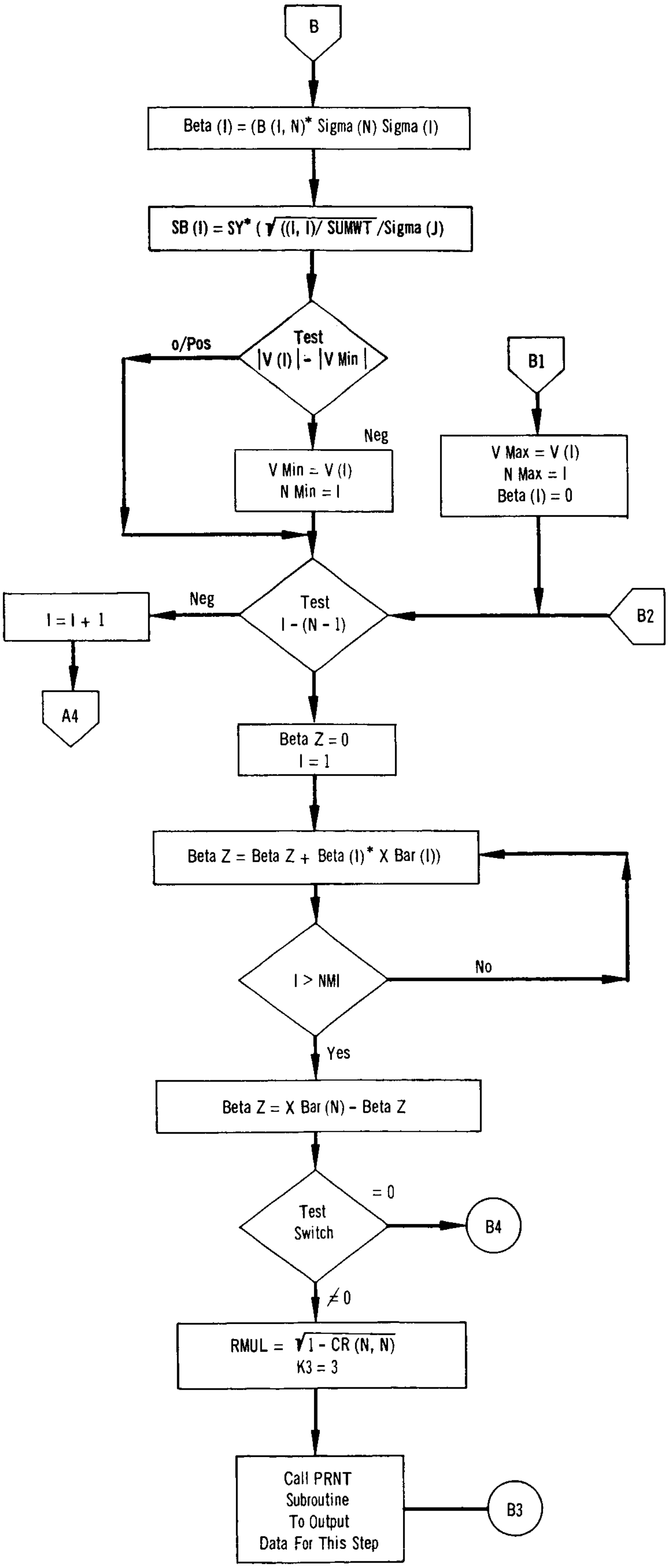
4c. Statistical Analysis Routine Flow Chart - Subroutine Calc



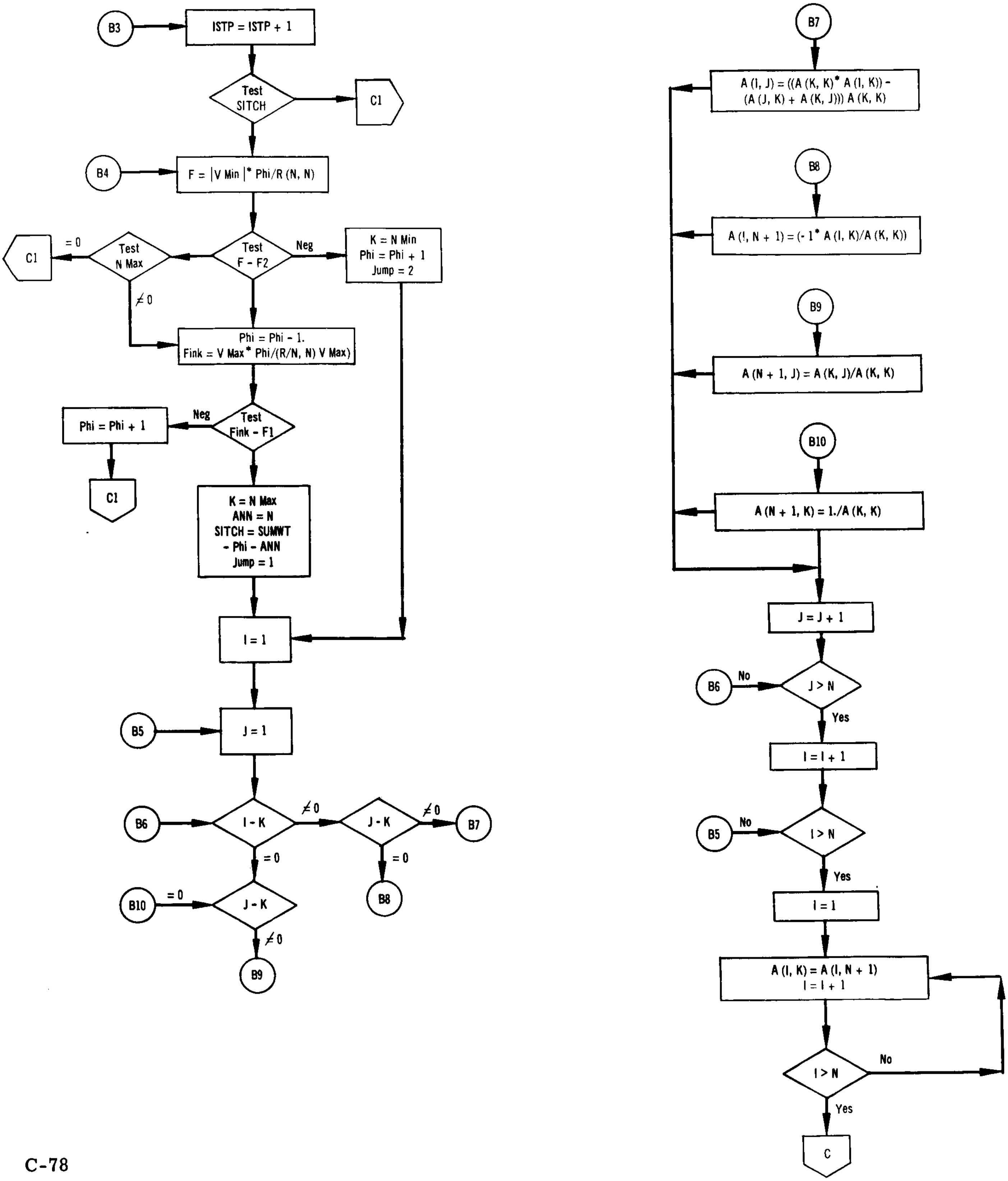
4c. Statistical Analysis Program Flow Chart – Subroutine Calc (Continued)



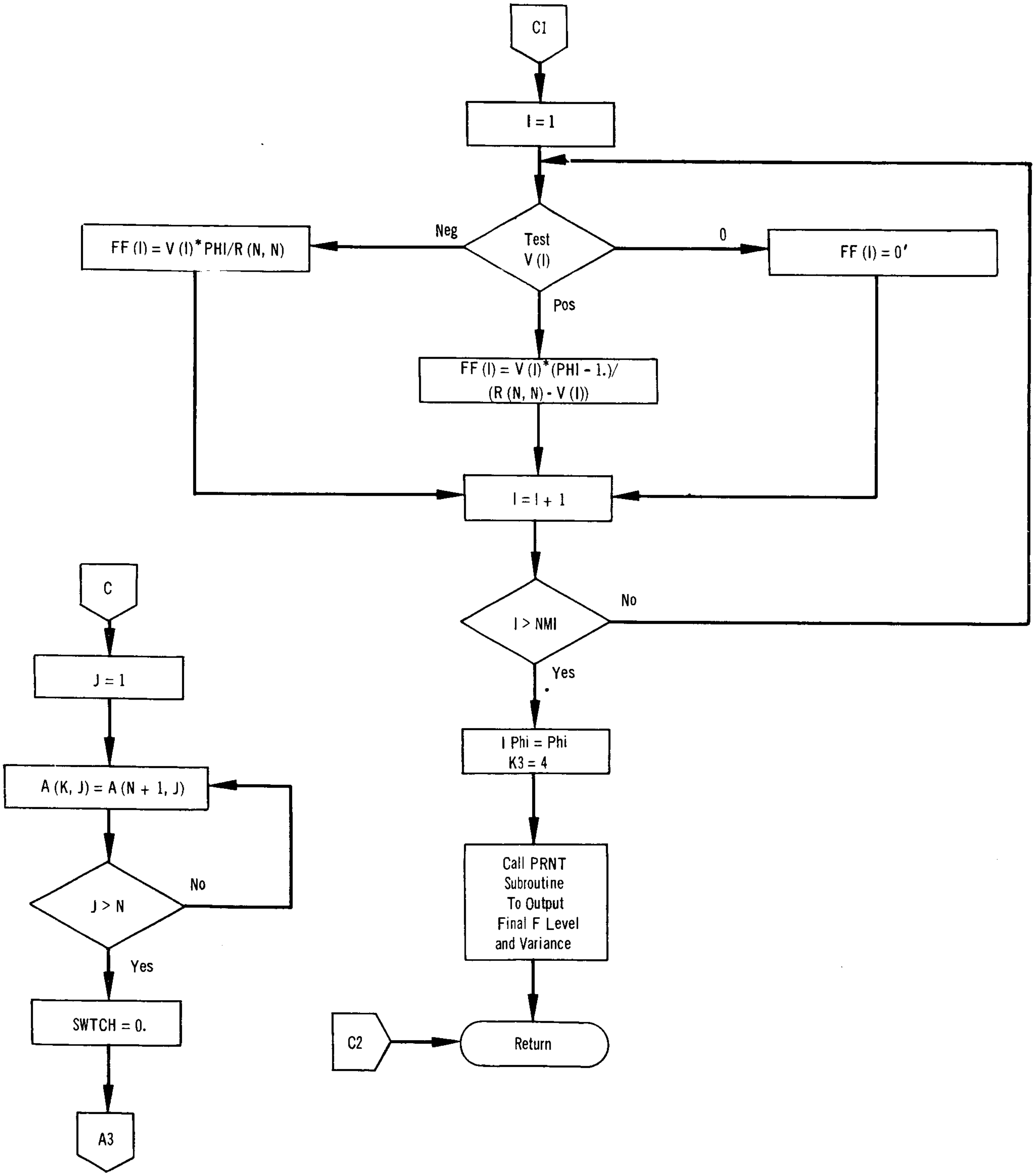
4c. Statistical Analysis Program Flow Chart – Subroutine Calc (Continued)



4c. Statistical Analysis Program Flow Chart – Subroutine Calc (Continued)



4c. Statistical Analysis Program Flow Chart – Subroutine Calc (Continued)



4c. Statistical Analysis Routine Source Statements – Subroutine Calc

```
*     CARDS COLUMN
*     SYMBOL TABLE
*     LIST8
*     LABEL
CCALC SUBROUTINE.
      SUBROUTINE CALC
      COMMON BUF,IT,IF,ICD,ICARD,IFLD,IPFLD,IFET,IFPET,NFET,END,P,F,BLK,
     1 FLP,FLR,JJ,JP,JJF,JPF,IDEP,IOP,IOF,F1,F2,IND,LL,LX,LP,IBATT,I1,
     2 ICNT,KCYC,IRAN
      COMMON JJL,ICYC,R,BETA,SWTX,LCYC,MCYC,L,I,J,NRAN,IBAT,NBAT,L1,L2,
     1 L3,K,KK,K2,K3,NOVAR,NOBS,NCYC,X,XBAR,SIGMA,SB,V,NM1,EMSQ,M,SUMWT,
     2 TOL,SWTCH,SITCH,PHI,ISTP,SY,VMIN,VMAX,NMAX,NMIN,BETAZ,RMUL,JUMP,
     3 FINK,FF,IPHI,II,JJN
      COMMON LSCYC,YSY,ANN,TEMP,TEM,TEM2,IDT,IFL,IN,IO,IR,NLOC,NPS,
     1 RHED,HED,RBETA,RSB,END,OUT,PP,JA,IP,LC
      DIMENSION BUF(4500),IT(42),IF(42),ICD(42),ICARD(1200),IFLD(42),
     1 IPFLD(16),IFET(42),IFPET(42),NFET(42),IRAN(19)
      DIMENSION JJL(42),ICYC(20),R(50,50),S(50,50),A(50,50),B(50,50),
     1C(50,50),X(50),XBAR(50),SB(50),SWTX(50),IBU(4500),BETA(50)
      DIMENSION V(50),SIGMA(50),FF(50),HED(50),RHED(50),RBETA(50),
     1 RSB(50),END(26),OUT(15),PP(16)
      EQUIVALENCE  (END,IEND),(FLP,IFLP),(FLR,IFLR),(F,KF),(P,KP),
     1 (BLK,IBLK)
      EQUIVALENCE (BUF,IBU), (R,S,A,B,C) , (X,XBAR), (SWTX,SB)
C.....CALCULATE VARIANCE - COVARIANCE MATRIX.
100   M = NOBS
      SUMWT = M
      N = NOVAR
      EMSQ = SUMWT**2
      DO 110 I =1,NOVAR
      DO 110 J =I,NOVAR
110   S(I,J) = (SUMWT * S(I,J) - SWTX(I) * SWTX(J)) / EMSQ
C.....CALCULATE MEAN.
      DO 120  I =1,NOVAR
120   XBAR(I) =   SWTX(I)/SUMWT
C.....CALCULATE STANDARD DEVIATION.
      DO 130 I=1,NOVAR
130   SIGMA(I) = SQRTF(S(I,I))
C.....CALL PRINT ROUTINE TO OUTPUT MEAN,VARIANCE AND STANDARD DEVIATION.
      K3   = 1
      CALL PRNT
C.....IS LINEAR CORRELATION DESIRED.
      IF (IOP-1) 510,510,140
140   IF (IOP-4) 145,510,145
C.....CALCULATE LINEAR CORRELATION COEFFICIENTS.
145   NM1 = NOVAR - 1
      DO 150 I =1,NM1
      JA = I+1
      DO 150 J =JA,N
      R(I,J) = S(I,J)/(SIGMA(I)*SIGMA(J))
150   R(J,I) = R(I,J)
      DO 156  I =1,NOVAR
156   R(I,I) = 1.
C.....CALL PRINT ROUTINE TO OUTPUT LINEAR CORRELATION COEFFICIENTS.
      K3   = 2
      CALL PRNT
C.....IS STEPWISE MULTIPLE REGRESSION DESIRED.
      IF (IOP-3) 510,161,160
```

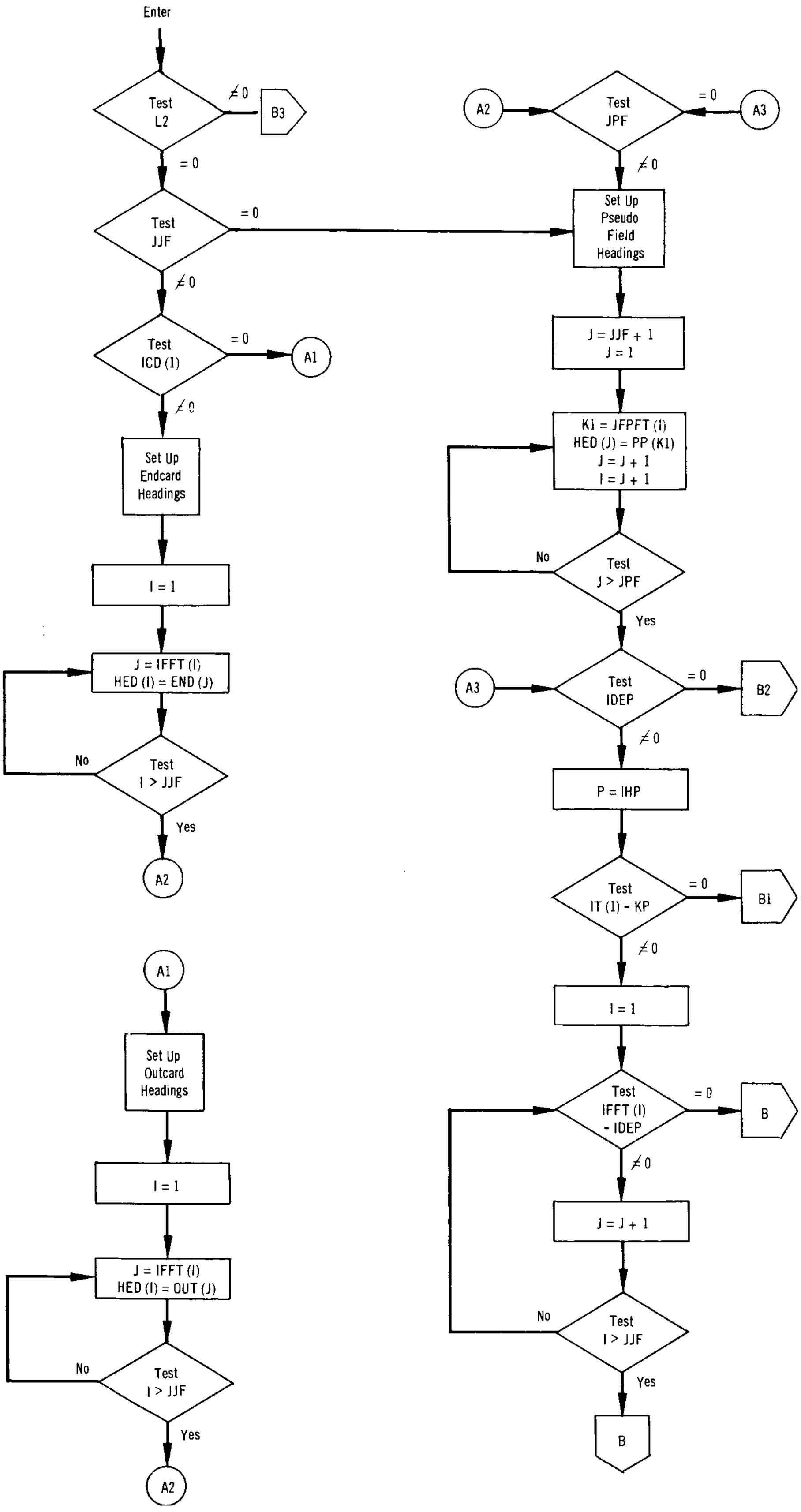
4c. Statistical Analysis Routine Source Statements – Subroutine Calc (Continued)

```
160   IF (IOP-6) 510,161,510
161   TOL = 0.001
      SWTCH = 0.0
170   SITCH = 1.0
      PHI = SUMWT - 1.0
      ISTP = 1
C.....STANDARD ERROR OF DEPENDENT VARIABLE.
180   SY = SIGMA(N) * SQRTF(R(N,N) * SUMWT / PHI)
      IF (SWTCH) 190,181,190
181   YSY = SY
190   I = 1
      VMIN =  10.0E20
      VMAX =  0.0
      NMIN = 0
      NMAX = 0
200   IF ( A(I,I) - TOL ) 260,260,210
210   V(I) = (A(I,N) * A(N,I)) /A(I,I)
      IF (V(I)) 240,260,220
220   IF (V(I)-VMAX) 260,260,230
230   VMAX = V(I)
      NMAX = I
      BETA(I) = 0.
      GO TO 260
C.....COEFFICIENTS OF INDEPENDENT VARIABLES.
240   BETA(I) = (B(I,N) * SIGMA(N)) / SIGMA(I)
C.....STD. ERR. OF COEFFICIENTS.
      SB(I) = SY * SQRTF( C(I,I) / SUMWT) / SIGMA(I)
      IF (ABSF (V(I)) - ABSF(VMIN))  250,260,260
250   VMIN = V(I)
      NMIN =  I
260   IF (I - (N-1)) 270,280,280
270   I = I+1
      GO TO 200
280   BETAZ = 0.
      DO 290 I = 1,NM1
290   BETAZ = BETAZ + (BETA(I)* XBAR(I))
C.....CONSTANT TERM.
      BETAZ = XBAR(N) - BETAZ
C.....DAVES INSERTION.
      IF (SWTCH) 300,310,300
C.....MULTIBLE R.
300   RMUL =  SQRTF (1. - R(N,N))
C.....CALL PRINT ROUTINE FOR STEPWISE REGRESSION OUTPUT.
      K3 = 3
      CALL PRNT
      ISTP = ISTP + 1
301   IF ( SITCH) 310,460,310
310   F = ABSF(VMIN) * PHI / R(N,N)
      IF (F-F2) 320,330,330
320   K = NMIN
      PHI = PHI +1.
C.....VARIABLE IS LEAVING.
      JUMP = 2
      GO TO 360
330   IF (NMAX) 332,331,332
331   NMAX = -1
      GO TO 460
332   PHI = PHI-1.
```

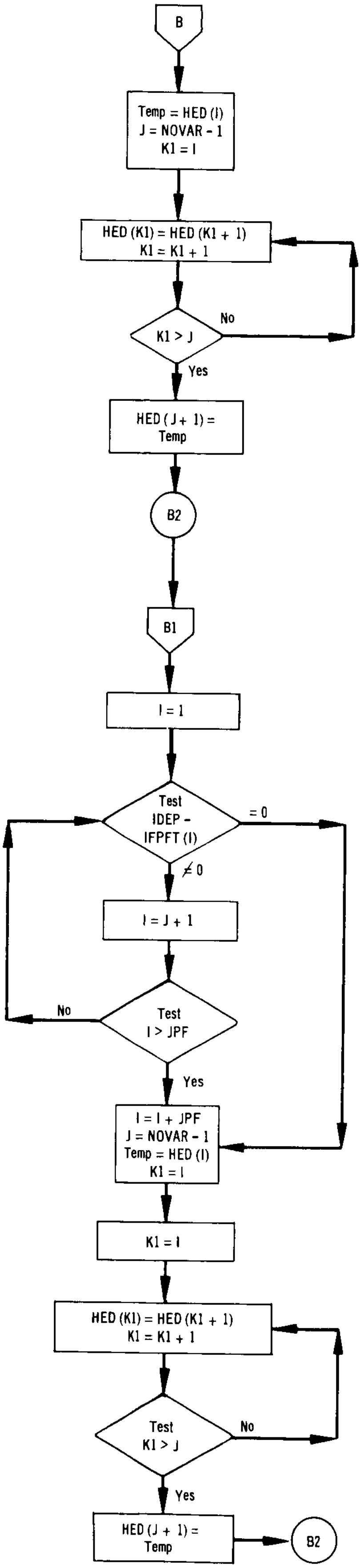
4c. Statistical Analysis Routine Source Statements – Subroutine Calc (Continued)

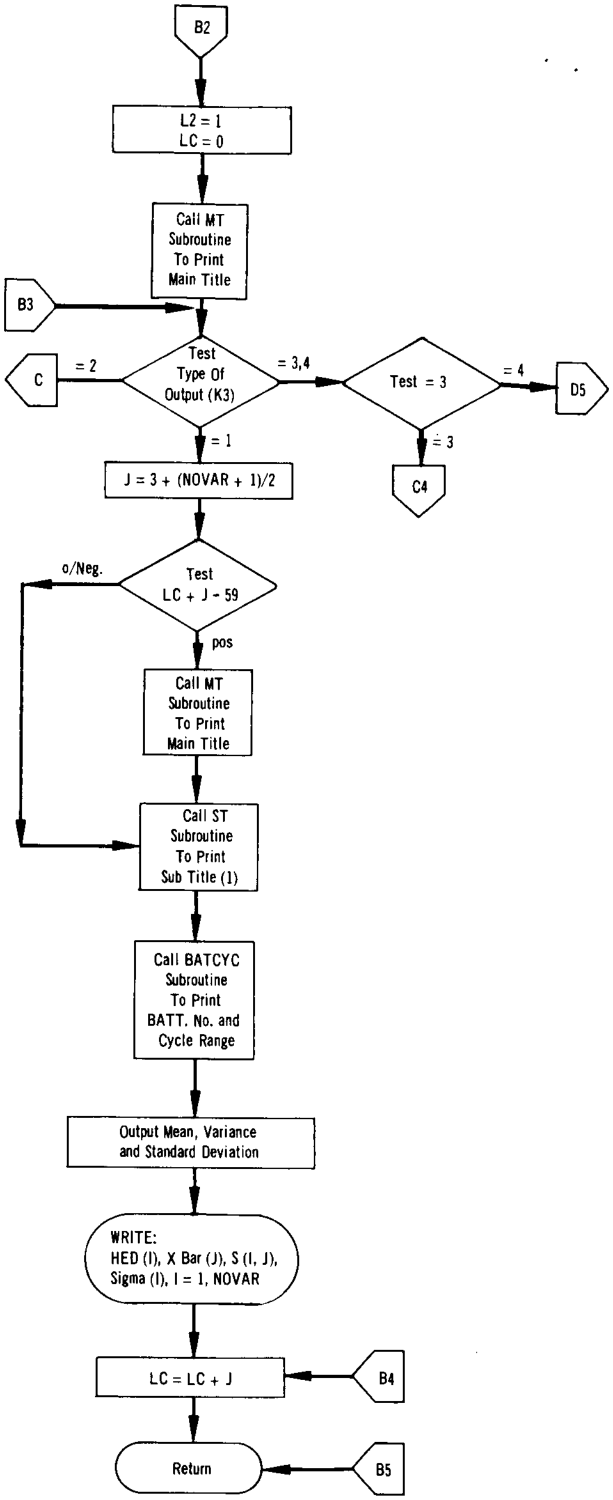
```
C.....F LEVEL
      FINK= VMAX*PHI/(R(N,N)-VMAX)
      IF (FINK - F1) 340,350,350
340   PHI = PHI + 1.
      GO TO 460
350   K = NMAX
      ANN = N
      SITCH  = SUMWT- PHI - ANN
C.....VARIABLE IS ENTERING.
      JUMP = 1
360   DO 430 I=1,N
      DO 430 J=1,N
      IF (I-K) 370,380,370
370   IF (J-K) 390,400,390
380   IF (J-K) 410,420,410
390   A(I,J) = ((A(K,K)*A(I,J))-(A(I,K)*A(K,J)))/A(K,K)
      GO TO 430
400   A(I,N+1)= (-1. *A(I,K)/A(K,K))
      GO TO 430
410   A(N+1,J) = A(K,J)/A(K,K)
      GO TO 430
420   A(N+1,K) = 1.0 /A(K,K)
430   CONTINUE
      DO 440  I=1,N
440   A(I,K) = A(I,N+1)
      DO 450 J=1,N
450   A(K,J) = A(N+1,J)
      SWTCH = 1.0
      GO TO 180
C.....FINAL VARIANCE CONTRIBUTIONS. FINAL F LEVELS.
460   DO 500 I = 1,NM1
      IF (V(I)) 470,480,490
470   FF(I) = V(I) * PHI/ R(N,N)
      GO TO 500
480   FF(I) = 0.
      GO TO 500
490   FF(I) = V(I) *(PHI-1.)/(R(N,N)-V(I))
500   CONTINUE
      IPHI = PHI
      K3 = 4
      CALL PRNT
510   RETURN
      END
```

4d. Statistical Analysis Routine – Subroutine PRNT

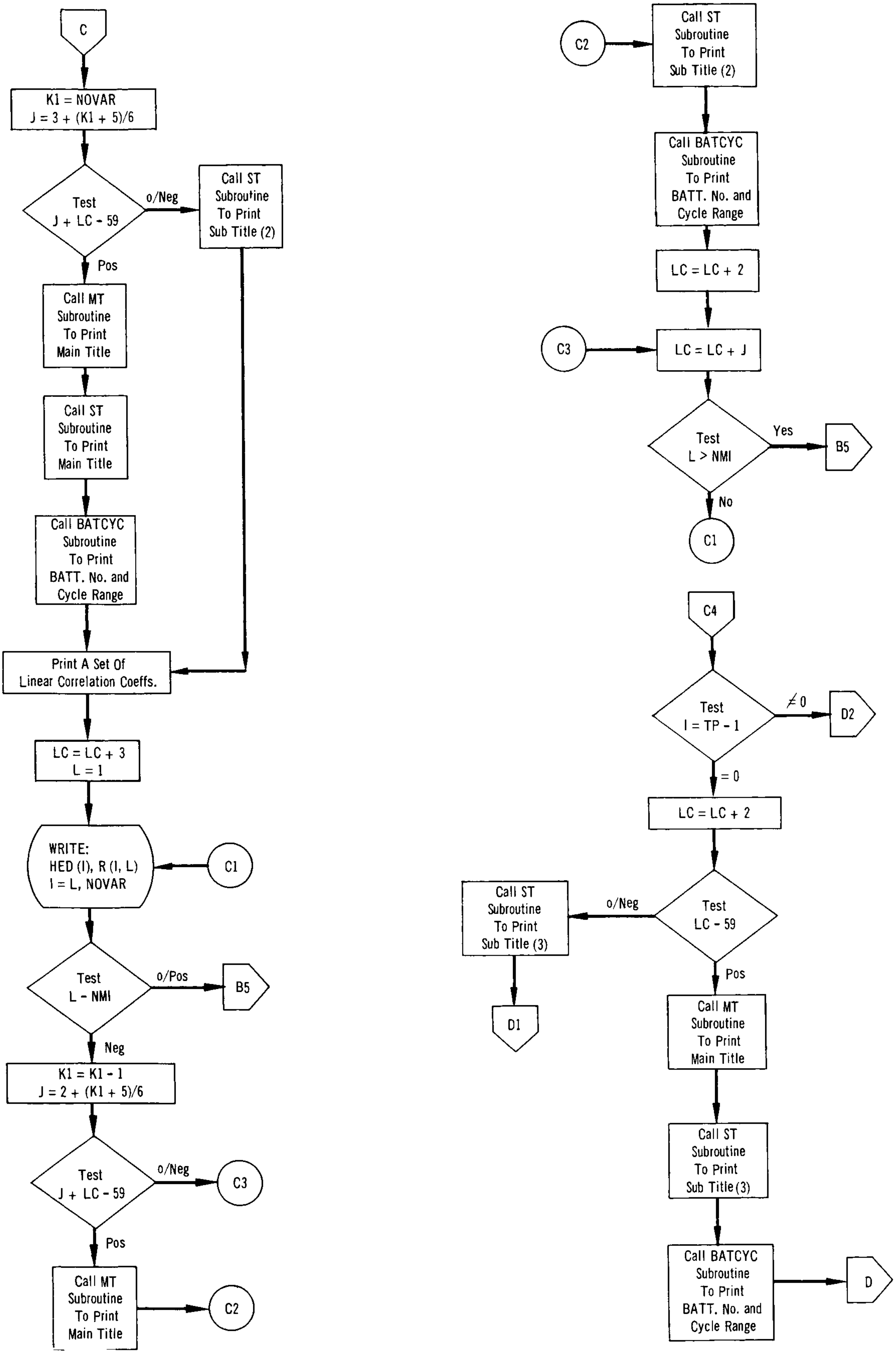


4d. Statistical Analysis Routine – PRNT Sub-Routine (Continued)

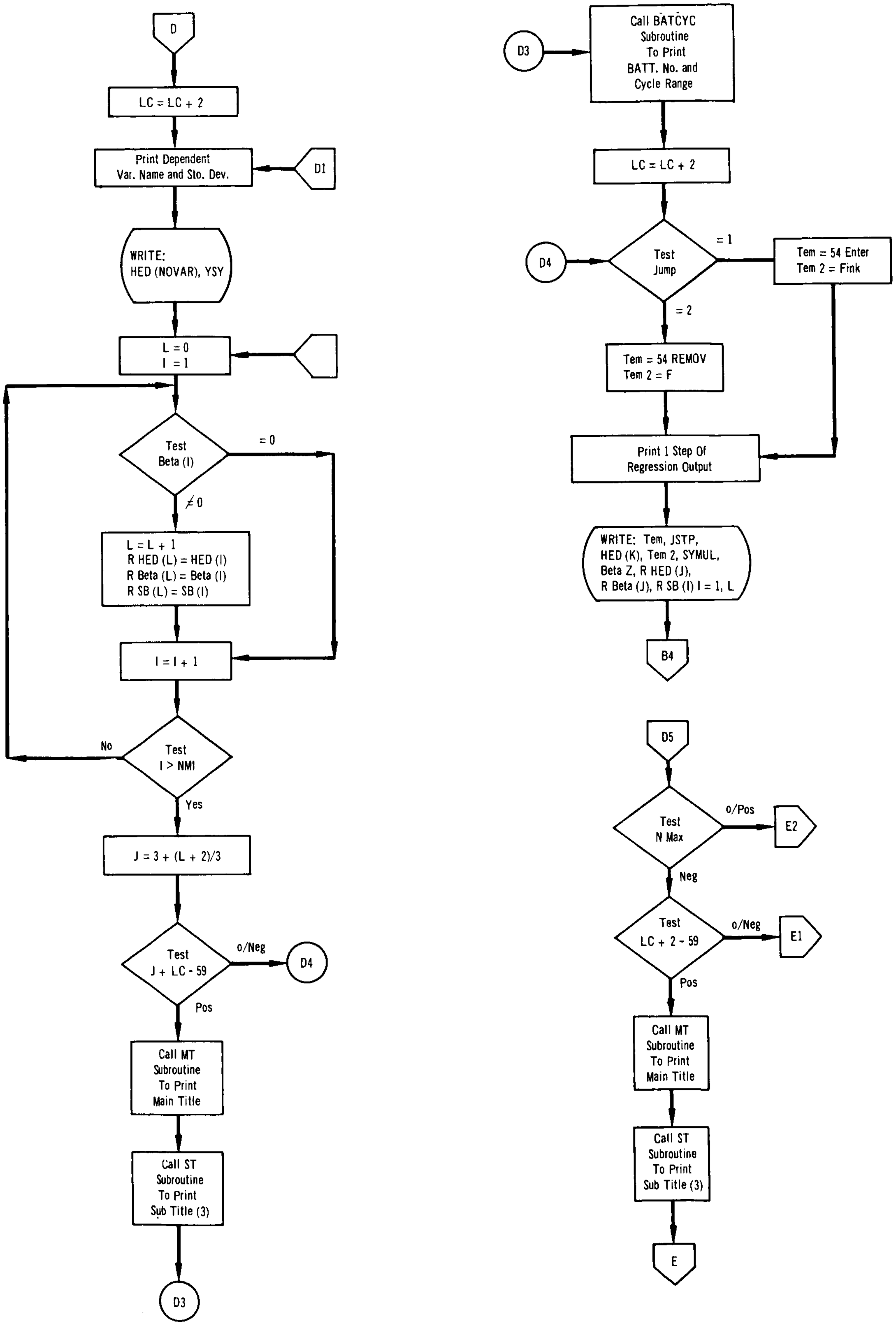




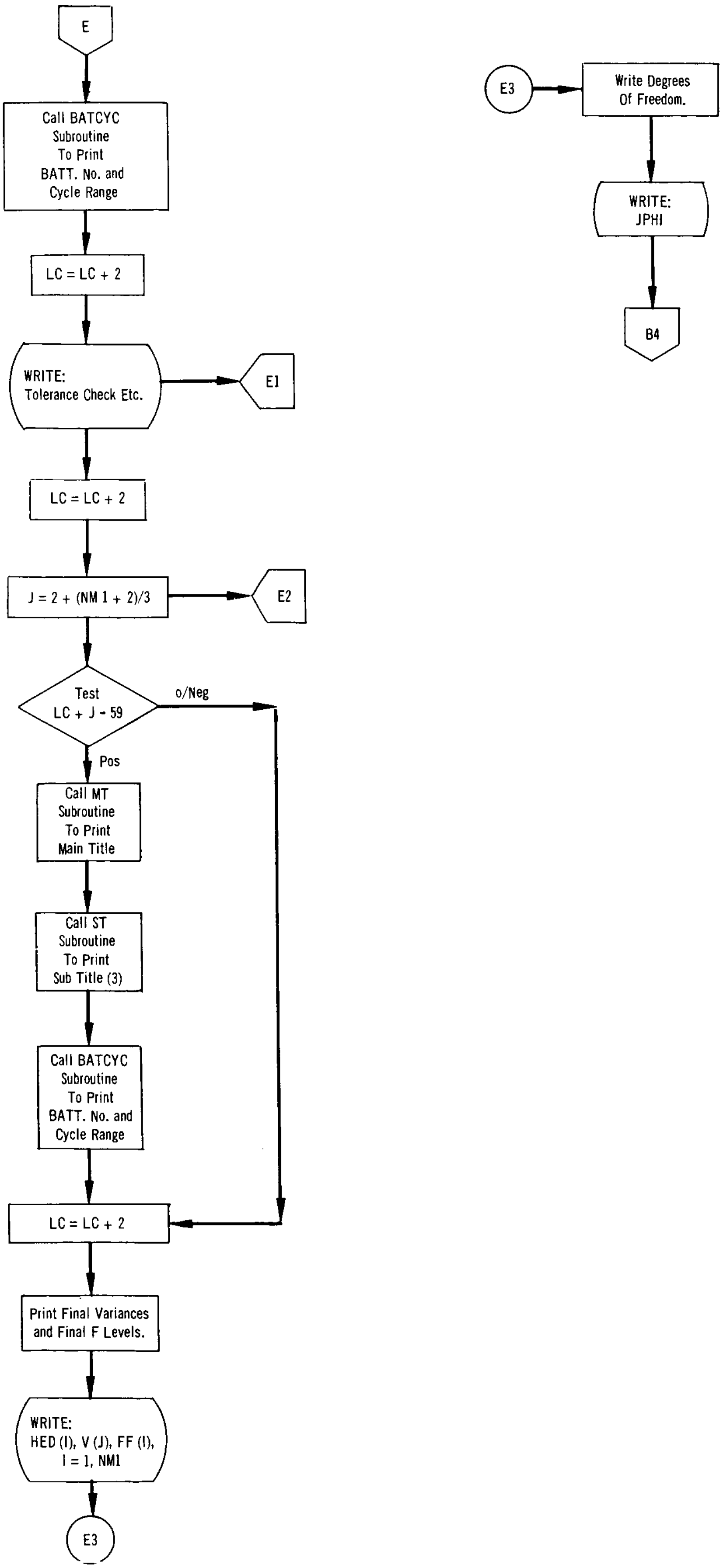
4d. Statistical Analysis Routine – Subroutine PRNT (Continued)



4d. Statistical Analysis Routine – Subroutine PRNT (Continued)


D
LC = LC + 2
Print Dependent Var. Name and Sto. Dev.
D1
WRITE: HED (NOVAR), YSY
L = 0
I = 1
Test Beta (I)
= 0
≠ 0
L = L + 1
R HED (L) = HED (I)
R Beta (L) = Beta (I)
R SB (L) = SB (I)
I = I + 1
Test I > NMI
No
Yes
J = 3 + (L + 2)/3
Test J + LC - 59
o/Neg
D4
Pos
Call MT Subroutine To Print Main Title
Call ST Subroutine To Print Sub Title (3)
D3
D3
Call BATCYC Subroutine To Print BATT. No. and Cycle Range
LC = LC + 2
D4
Test Jump
= 1
Tem = 54 Enter
Tem 2 = Fink
= 2
Tem = 54 REMOV
Tem 2 = F
Print 1 Step Of Regression Output
WRITE: Tem, JSTP, HED (K), Tem 2, SYMUL, Beta Z, R HED (J), R Beta (J), R SB (I) I = 1, L
B4
D5
Test N Max
o/Pos
E2
Neg
Test LC + 2 - 59
o/Neg
E1
Pos
Call MT Subroutine To Print Main Title
Call ST Subroutine To Print Sub Title (3)
E

4d. Statistical Analysis Routine – Subroutine PRNT (Continued)


E
Call BATCYC Subroutine To Print BATT. No. and Cycle Range
LC = LC + 2
WRITE: Tolerance Check Etc.
E1
LC = LC + 2
J = 2 + (NM 1 + 2)/3
E2
Test LC + J - 59
o/Neg
Pos
Call MT Subroutine To Print Main Title
Call ST Subroutine To Print Sub Title (3)
Call BATCYC Subroutine To Print BATT. No. and Cycle Range
LC = LC + 2
Print Final Variances and Final F Levels.
WRITE: HED (I), V (J), FF (I), I = 1, NM1
E3
E3
Write Degrees Of Freedom.
WRITE: JPHI
B4

4d. Statistical Analysis Routine – Source Statements

```
*       CARDS COLUMN
*       LIST8
*       SYMBOL TABLE
*       LABEL
CPRNT   -PRINT ROUTINE.
        SUBROUTINE PRNT
        COMMON BUF,IT,IF,ICD,ICARD,IFLD,IPFLD,IFFT,IFPFT,NFFT,END,P,F,BLK,
      1 FLP,FLR,JJ,JP,JJF,JPF,IDEP,IOP,IOF,F1,F2,IND,LL,LX,LP,IBATT,I1,
      2 ICNT,KCYC,IRAN
        COMMON JJL,ICYC,R,BETA,SWTX,LCYC,MCYC,L,I,J,NRAN,IBAT,NBAT,L1,L2,
      1 L3,K,KK,K2,K3,NOVAR,NOBS,NCYC,X,XBAR,SIGMA,SB,V,NM1,EMSQ,M,SUMWT,
      2 TOL,SWTCH,SITCH,PHI,ISTP,SY,VMIN,VMAX,NMAX,NMIN,BETAZ,RMUL,JUMP,
      3 FINK,FF,IPHI,II,JJN
        COMMON LSCYC,YSY,ANN,TEMP,TEM,TEM2,IDT,IFL,IN,IO,IR,NLOC,NPS,
      1 RHED,HED,RBETA,RSB,END,OUT,PP,JA,IP,LC
        DIMENSION BUF(4500),IT(42),IF(42),ICD(42),ICARD(1200),IFLD(42),
      1 IPFLD(16),IFFT(42),IFPFT(42),NFFT(42),IRAN(19)
        DIMENSION JJL(42),ICYC(20),R(50,50),S(50,50),A(50,50),B(50,50),
      1C(50,50),X(50),XBAR(50),SB(50),SWTX(50),IBU(4500),BETA(50)
        DIMENSION V(50),SIGMA(50),FF(50),HED(50),RHED(50),RBETA(50),
      1 RSB(50),END(26),OUT(15),PP(16)
        EQUIVALENCE  (END,IEND),(FLP,IFLP),(FLR,IFLR),(F,KF),(P,KP),
      1 (BLK,IBLK)
        EQUIVALENCE (BUF,IBU), (R,S,A,B,C) , (X,XBAR), (SWTX,SB)
C.....TEST IF THIS THE FIRST TIME THRU THE PRINT ROUTINE.
        IF (L2) 190,100,190
C.....SET UP FIELD AND PSEUDO FIELD HEADINGS.
100     IF (JJF) 110,141,110
110     IF (ICD(1)) 120,130,120
C.....ENDCARD HEADINGS.
120     END(1) = 6HTI DUR
        END(2) = 6HAMP MI
        END(3) = 6HWAT MI
        END(4) = 6H E EFF
        END(5) = 6H AMP H
        END(6) = 6H WAT H
        END(7) = 6HAV CUR
        END(8) = 6HMI CUR
        END(9) = 6HMX CUR
        END(10)= 6HAV PWR
        END(11)= 6HMI PWR
        END(12)= 6HMX PWR
        END(13)= 6H C EFF
        END(14)= 6HE/I  R
        END(15)= 6HCYC  0
        END(16)= 6HCYC  1
        END(17)= 6HCYC  2
        END(18)= 6HCYC  3
        END(19)= 6HCYC  4
        END(20)= 6HCYC  5
        END(21)= 6HCYC  6
        END(22)= 6HCYC  7
        END(23)= 6HCYC  8
        END(24)= 6HCYC  9
        END(25)= 6HMX VAR
        END(26)= 6HR TIME
C.....SET UP HEADINGS FOR REQUESTED FIELDS.
        DO 121 I =1,JJF
```

4d. Statistical Analysis Routine – Source Statements (Continued)

```
      J = IFFT(I)
121   HED(I) = END(J)
      GO TO 140
C.....OUTCARD HEADINGS.
130   OUT(1) = 6H TIME
      OUT(2) = 6HD TIME
      OUT(3) = 6HAV PWR
      OUT(4) = 6H  CUR
      OUT(5) = 6HPRESS
      OUT(6) = 6H V 0
      OUT(7) = 6H V 1
      OUT(8) = 6H V 2
      OUT(9) = 6H V 3
      OUT(10)= 6H V 4
      OUT(11)= 6H V 5
      OUT(12)= 6H V 6
      OUT(13)= 6H V 7
      OUT(14)= 6H V 8
      OUT(15)= 6H V 9
C.....SET UP HEADINGS FOR REQUESTED FIELDS.
      DO 131 I= 1,JJF
      J = IFFT(I)
131   HED(I) = OUT(J)
C.....WERE PSEUDO FIELDS REQUESTED.
140   IF (JPF) 141,150,141
C.....PSEUDO FIELD HEADINGS.
141   PP(1) = 4H  P1
      PP(2) = 4H  P2
      PP(3) = 4H  P3
      PP(4) = 4H  P4
      PP(5) = 4H  P5
      PP(6) = 4H  P6
      PP(7) = 4H  P7
      PP(8) = 4H  P8
      PP(9) = 4H  P9
      PP(10)= 4H P10
      PP(11)= 4H P11
      PP(12)= 4H P12
      PP(13)= 4H P13
      PP(14)= 4H P14
      PP(15)= 4H P15
      PP(16)= 4H P16
C.....SET UP HEADINGS FOR REQUESTED PSEUDO FIELDS.
      J = JJF + 1
      DO 142  I = 1,JPF
      K1 =  IFPFT(I)
      HED(J) = PP(K1)
142   J = J + 1
C.....IS THIS A MULTIPLE REGRESSION RUN.
150   IF(IDEP) 151,180,151
C.....YES - SET DEPENDENT VARIABLE LAST IN ARRAY OF HEADINGS.
151   P = 1HP
      IF ( IT(1) -KP) 160,170,160
160   DO 161 I =1,JJF
      IF (IFFT(I)-IDEP) 161,162,161
161   CONTINUE
162   TEMP = HED(I)
      J = NOVAR-1
      DO 163 K1=I,J
163   HED(K1)= HED(K1+1)
      HED(J+1)= TEMP
      GO TO 180
170   DO 171 I =1,JPF
      IF (IDEP- IFPFT(I)) 171,172,171
171   CONTINUE
172   I = I + JJF
      J = NOVAR-1
      TEMP = HED(I)
      DO 173 K1 = I,J
173   HED(K1) = HED(K1+1)
      HED(J+1) = TEMP
180   L2 = 1
C.....INITIALIZE PAGE COUNT.
      LC = 0
C.....PRINT MAIN TITLE.
      CALL MT (LC)
C.....TEST FOR TYPE OF OUTPUT
190   GO TO (191,240,310,420),K3
C.....MEAN,VARIANCE,STD. DEV.
191   J = 3 + (NOVAR+1)/2
C.....TEST LINE COUNT VS. LINES TO BE PRINTED.
      IF (LC + J - 59)  210,210,200
200   CALL MT (LC)
210   CALL ST(1)
C.....PRINT BATT AND CYCLE RANGE
      CALL BATCYC (NBAT,LSCYC,LCYC)
C.....PRINT MEAN VAR.,STD. DEV.
      WRITE OUTPUT TAPE IO,1009,(HED(I),XBAR(I),S(I,I),SIGMA(I),
     1 I = 1,NOVAR)
1009  FORMAT( 2(5X,5HFIELD,10X,4HMEAN,12X,8HVARIANCE,8X,9HSTD. DEV.,1X)/
     1 (5X,A6,5X,1PE12.5,5X,E12.5,5X,E12.5,5X,A6,5X,E12.5,5X,E12.5,5X,
     2 E12.5    ))
220   LC = LC + J
230   RETURN
C.....LINEAR CORRELATION COEFFICIENTS.
C.....
C.....CALC. NO. OF LINES TO BE PRINTED FOR A SET.
240   K1 = NOVAR
250   J =  3 + (K1+5)/6
      IF (J + LC - 59) 270,270,260
260   CALL MT (LC)
      CALL ST (2)
      CALL BATCYC (NBAT,LSCYC,LCYC)
      GO TO 280
270   CALL ST (2)
280   LC = LC + J
C.....PRINT A SET OF COEFFICIENTS.
      DO 300 L = 1,NM1
      WRITE OUTPUT TAPE IO,1010, (HED(I),R(I,L),I=L,NOVAR)
1010  FORMAT(/6(1X,5HFIELD,6X,6HCOEFF.,4X)/(1X,A6,2X,1PE12.5,2X,A6,2X
     1 E12.5,2X,A6,2X,E12.5,2X,A6,2X,E12.5,2X,A6,2X,E12.5,2X,A6,2X,E12.5
     2 ))
C.....DOES ANOTHR SET OF DATA EXIST.
      IF (L-NM1) 281,230,230
C.....TEST IF ANOTHER SET MAY BE PRINTED ON THIS PAGE.
281   K1 = K1-1
```

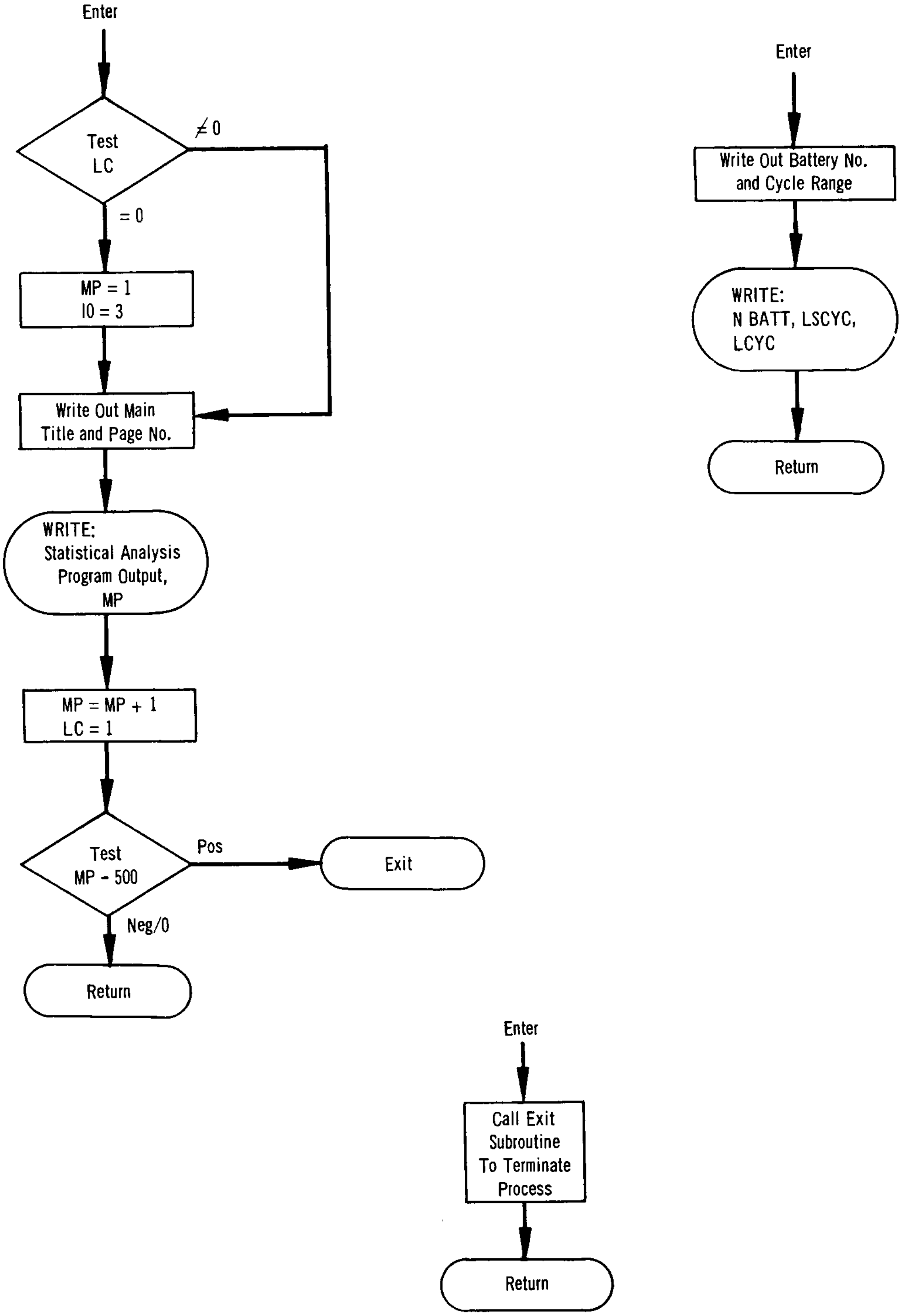
4d. Statistical Analysis Routine – Source Statements (Continued)

```
          J = 2 + (K1+5)/6
          IF (J +LC -59)  300,300,290
  290     CALL MT (LC)
          CALL ST(2)
          CALL BATCYC (NBAT,LSCYC,LCYC)
          LC = LC + 2
  300     LC = LC + J
          GO TO 230
C.....STEPWISE REGRESSION.
C.....
C.....CALC. NO OF LINES TO BE PRINTED.
  310     IF (ISTP-1) 350,320,350
  320     LC = LC + 2
          IF (LC - 59) 340,340,330
  330     CALL MT (LC)
          CALL ST (3)
          CALL BATCYC (NBAT,LSCYC,LCYC)
          LC = LC+2
          GO TO 350
  340     CALL ST (3)
          WRITE OUTPUT TAPE IO,1111,HED(NOVAR),YSY
  1111    FORMAT(41X,9HDEP. VAR.,2X,A6,12X,10HSTD. ERROR,2X,1PE12.5)
  350     L = 0
          DO 360  I = 1,NM1
          IF (BETA(I) ) 351,360,351
  351     L = L +1
          RHED(L) = HED(I)
          RBETA(L)= BETA(I)
          RSB(L)  = SB(I)
  360     CONTINUE
          J = 3 + (L+2)/3
          IF ( J + LC - 59) 380,380,370
  370     CALL MT (LC)
          CALL ST (3)
          CALL BATCYC (NBAT,LSCYC,LCYC)
          LC = LC + 2
  380     GO TO (390,400),JUMP
  390     TEM   = 5HENTER
          TEM2 = FINK
          GO TO 410
  400     TEM   = 5HREMOV
          TEM2 = F
  410     WRITE OUTPUT TAPE IO,1112, TEM,ISTP,HED(K),TEM2,SY,RMUL,BETAZ
  1112    FORMAT(12X,4HSTEP,6X,5HVAR. A5,2HED,10X,7HF LEVEL,8X,22HSTD. ERR.
         1OF DEP. VAR.,6X,10HMULTIPLE R,10X,13HCONSTANT TERM /12X,I4,10X,A6,
         24(10X,1PE12.5))
          WRITE OUTPUT TAPE IO,1113, (RHED(I),RBETA(I),RSB(I),I=1,L)
  1113    FORMAT(3(5X,5HFIELD,5X,11HCOEFFICIENT,6X,10HSTD. ERROR )/
         1 (5X,A6,4X,1PE12.5,4X,E12.5,4X,A6,4X,E12.5,4X,E12.5,4X,A6,4X,E12.5
         2,4X,E12.5))
          GO TO 220
C.....TOLERANCE CHECK.
  420     IF (NMAX) 421,424,424
  421     IF (LC+2-59) 423,423,422
  422     CALL MT (LC)
          CALL ST (3)
          CALL BATCYC (NBAT,LSCYC,LCYC)
          LC = LC + 2
```

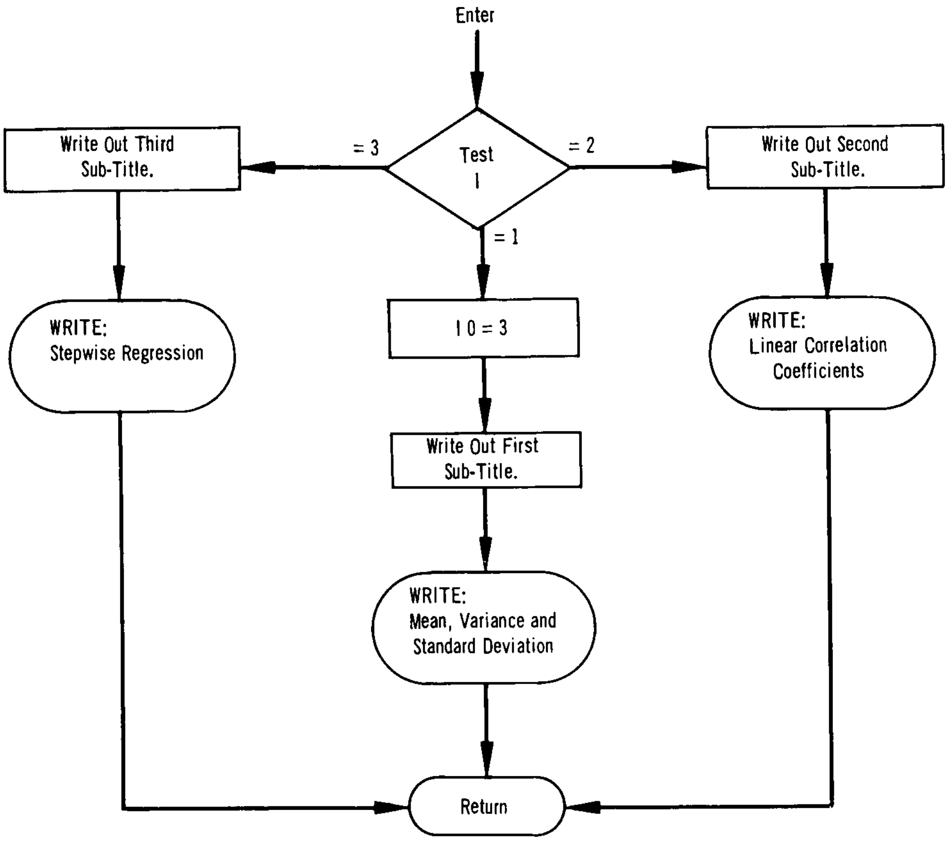
4d. Statistical Analysis Routine – Source Statements (Continued)

```
 423    WRITE OUTPUT TAPE IO,1116
 1116   FORMAT(111HOTOLERANCE CHECK....REMAINING IND. VARS. HAVE TOO HIGH
       1A CORRELATION WITH VARS. ALREADY IN REGRESSION EQUATION.  )
        LC = LC + 2
C.....FINAL VARIANCE CONTRIBUTION AND FINAL F LEVELS.
 424    J = 2 + (NM1+2)/3
        IF (LC + J - 59) 440,440,430
 430    CALL MT (LC)
        CALL ST (3)
        CALL BATCYC (NBAT,LSCYC,LCYC)
        LC = LC + 2
 440    WRITE OUTPUT TAPE IO,1114,(HED(I),V(I),FF(I),I=1,NM1)
 1114   FORMAT(1H0,3(4X,5HFIELD,4X,14HFINAL VARIANCE,2X,13HFINAL F LEVEL)/
       1 (5X,A6,4X,1PE12.5,4X,E12.5,4X,A6,4X,E12.5,4X,E12.5,4X,A6,4X,E12.5
       2,4X,E12.5 ))
C.....WRITE  DEGREES OF FREEDOM.
        WRITE OUTPUT TAPE IO,1115, IPHI
 1115   FORMAT (22H0 DEGREES OF FREEDOM = I5 )
        GO TO 220
        END
```

4e. Statistical Analysis Routine – Subroutines MT, BAT-CYC, ST, and FINI


Enter
Test LC
≠ 0
= 0
MP = 1
I0 = 3
Write Out Main Title and Page No.
WRITE: Statistical Analysis Program Output, MP
MP = MP + 1
LC = 1
Test MP - 500
Pos
Exit
Neg/0
Return
Enter
Write Out Battery No. and Cycle Range
WRITE: N BATT, LSCYC, LCYC
Return
Enter
Call Exit Subroutine To Terminate Process
Return

4e. Statistical Analysis Routine – Subroutine MT, BAT-CYC, ST, and FINI (Continued)

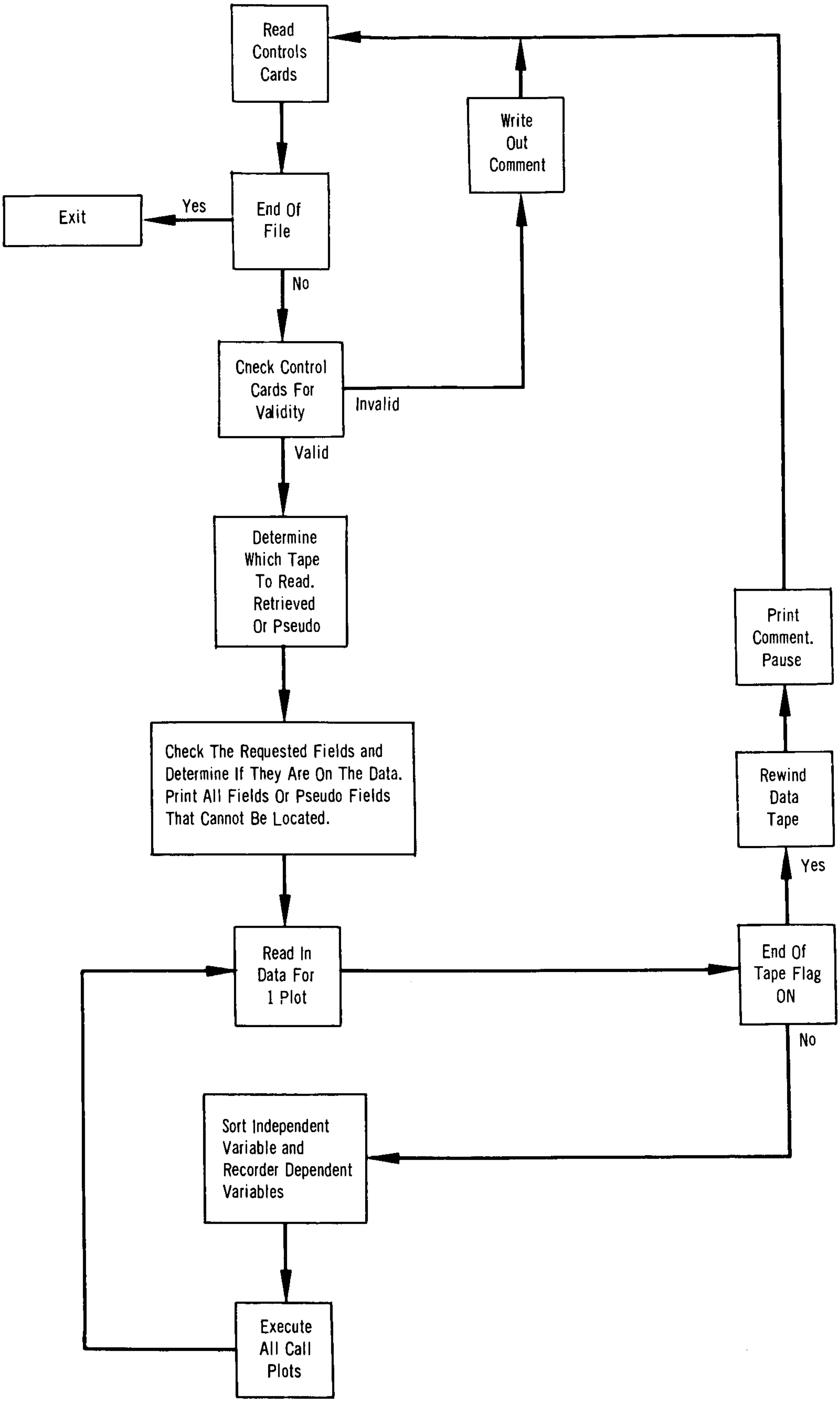

Enter
Test I
= 3
= 2
= 1
Write Out Third Sub-Title.
Write Out Second Sub-Title.
I 0 = 3
WRITE: Stepwise Regression
WRITE: Linear Correlation Coefficients
Write Out First Sub-Title.
WRITE: Mean, Variance and Standard Deviation
Return

4e. Statistical Analysis Routine – Subroutines MT, BAT-CYC, ST, and FINI (Continued)

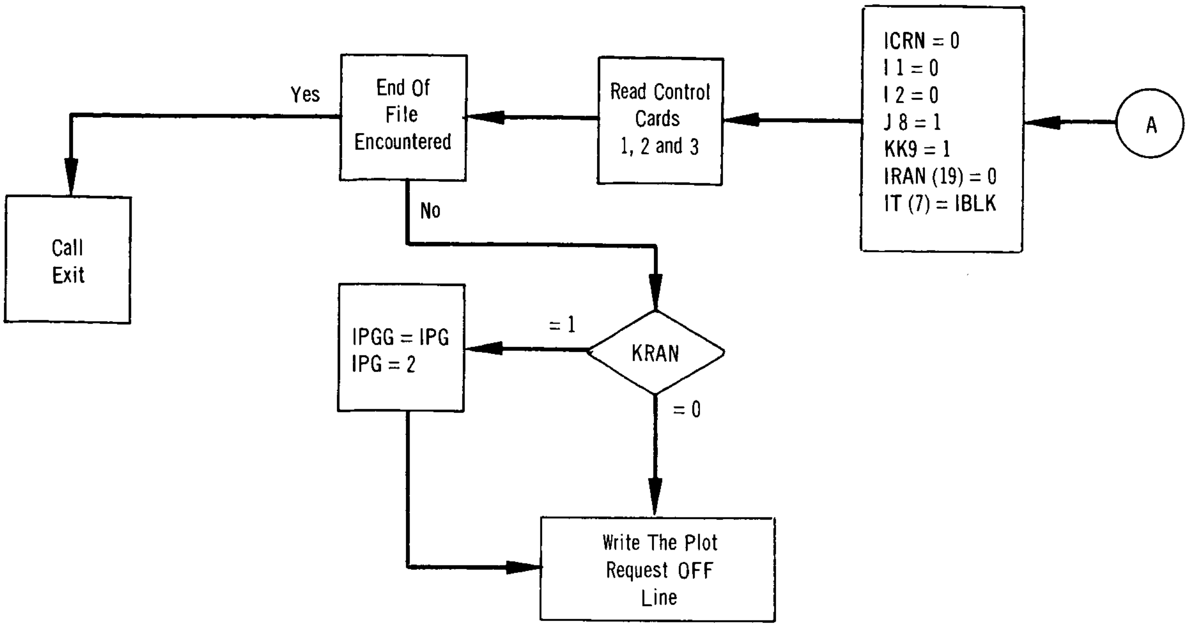
```
*       CARDS COLUMN
*       LIST8
*       LABEL
CMT      - MAIN TITLE ROUTINE.
        SUBROUTINE MT (LC)
        IF (LC) 2,1,2
1       MP = 1
        IO = 3
2       WRITE OUTPUT TAPE IO, 3, MP
3       FORMAT(1H1,27X,72HS T A T I S T I C A L    A N A L Y S I S    P R
       1O G R A M    O U T P U T ,21X,5HPAGE I5 )
        MP = MP + 1
        LC = 1
        IF (MP-500) 5,5,4
4       CALL FINI
5       RETURN
        END
*       CARDS COLUMN
*       LIST8
*       LABEL
CST      - SUB TITLE ROUTINE.
        SUBROUTINE ST (I)
        GO TO (1,3,5),I
1       IO = 3
        WRITE OUTPUT TAPE IO,2
2       FORMAT(25X,77HM E A N  ,   V A R I A N C E    A N D    S T A N D A
       1R D    D E V I A T I O N  )
        GO TO 7
3       WRITE OUTPUT TAPE IO,4
4       FORMAT(32X,63HL I N E A R    C O R R E L A T I O N    C O E F F I
       1C I E N T S )
        GO TO 7
5       WRITE OUTPUT TAPE IO,6
6       FORMAT(46X,38HS T E P W I S E    R E G R E S S I O N  )
7       RETURN
        END
*       CARDS COLUMN
*       LIST8
*       LABEL
CBATCYC- BATTERY NO. CYCLE RANGE ROUTINE.
        SUBROUTINE BATCYC (IBAT,LSCYC,LCYC)
        WRITE OUTPUT TAPE 3,1,IBAT,LSCYC,LCYC
1       FORMAT(45X,7HBATTERY I2,10X,11HCYCLE RANGE,I5,1H-,I4 )
        RETURN
        END
*       CARDS COLUMN
*       LIST8
*       LABEL
CFINI
        SUBROUTINE FINI
        CALL EXIT
        RETURN
        END
```

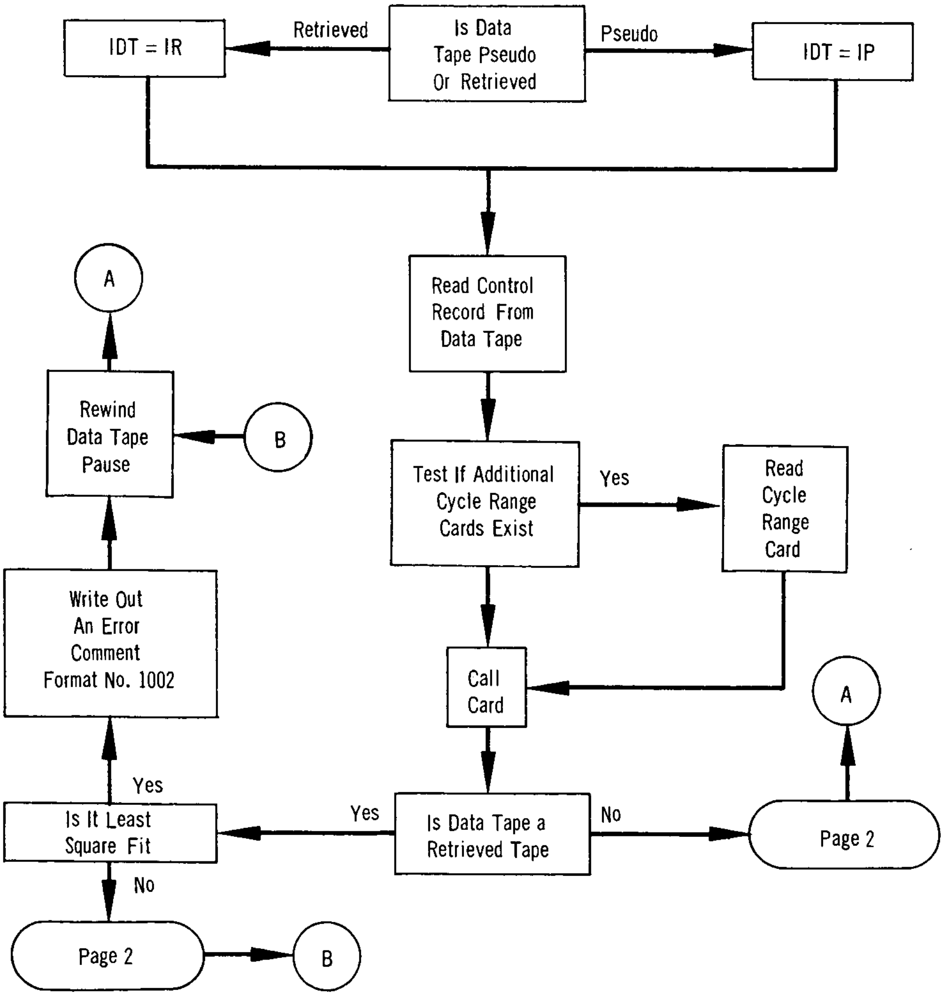
5. Printed Plot Routine

5. Printer Plot Routine - General Flow Charts

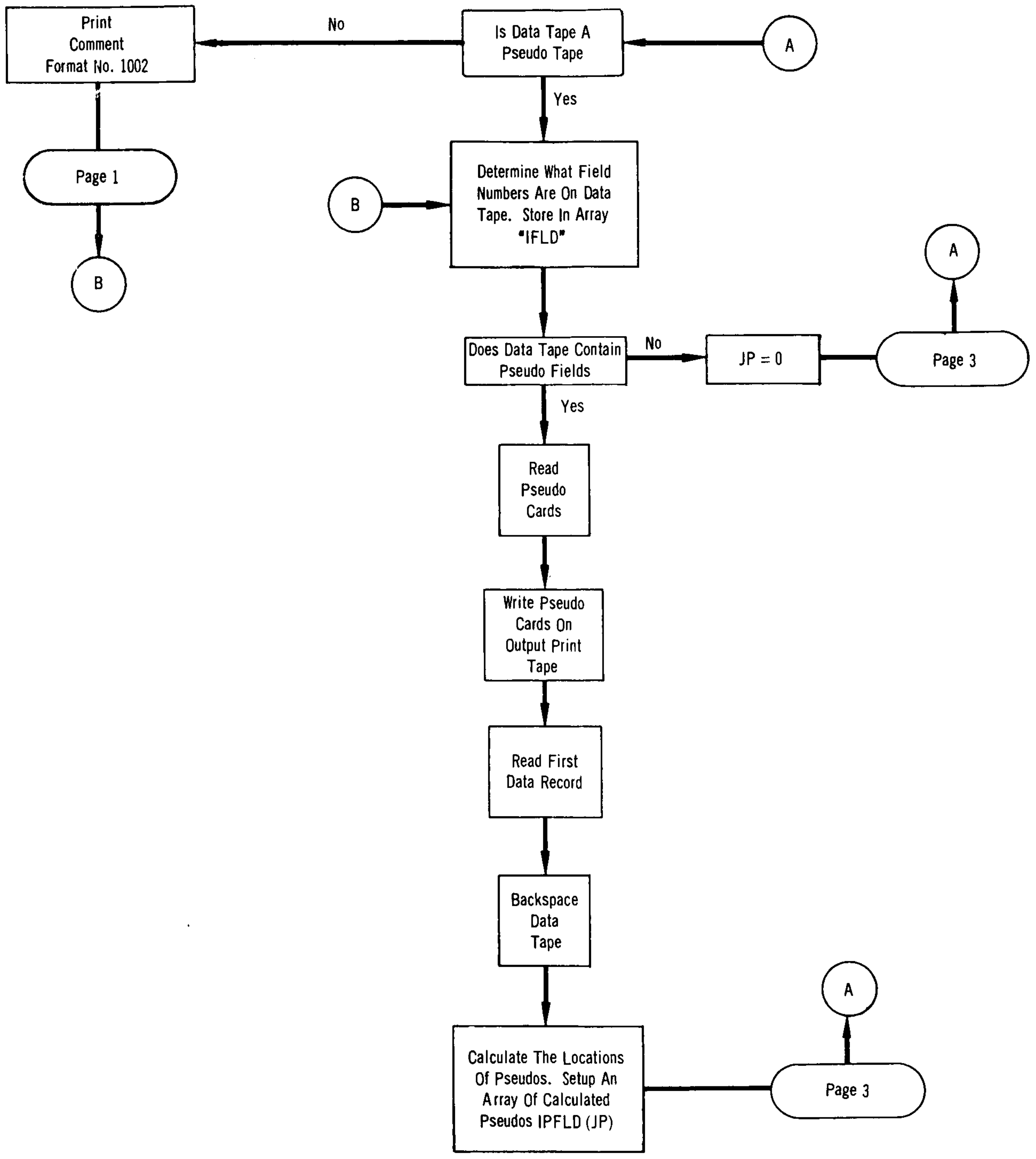

Read Controls Cards
Write Out Comment
Exit
Yes
End Of File
No
Cneck Control Cards For Validity
Invalid
Valid
Determine Which Tape To Read. Retrieved Or Pseudo
Print Comment. Pause
Check The Requested Fields and Determine If They Are On The Data. Print All Fields Or Pseudo Fields That Cannot Be Located.
Rewind Data Tape
Yes
Read In Data For 1 Plot
End Of Tape Flag ON
No
Sort Independent Variable and Recorder Dependent Variables
Execute All Call Plots

5a. Printer Plot Routine – Flow Charts



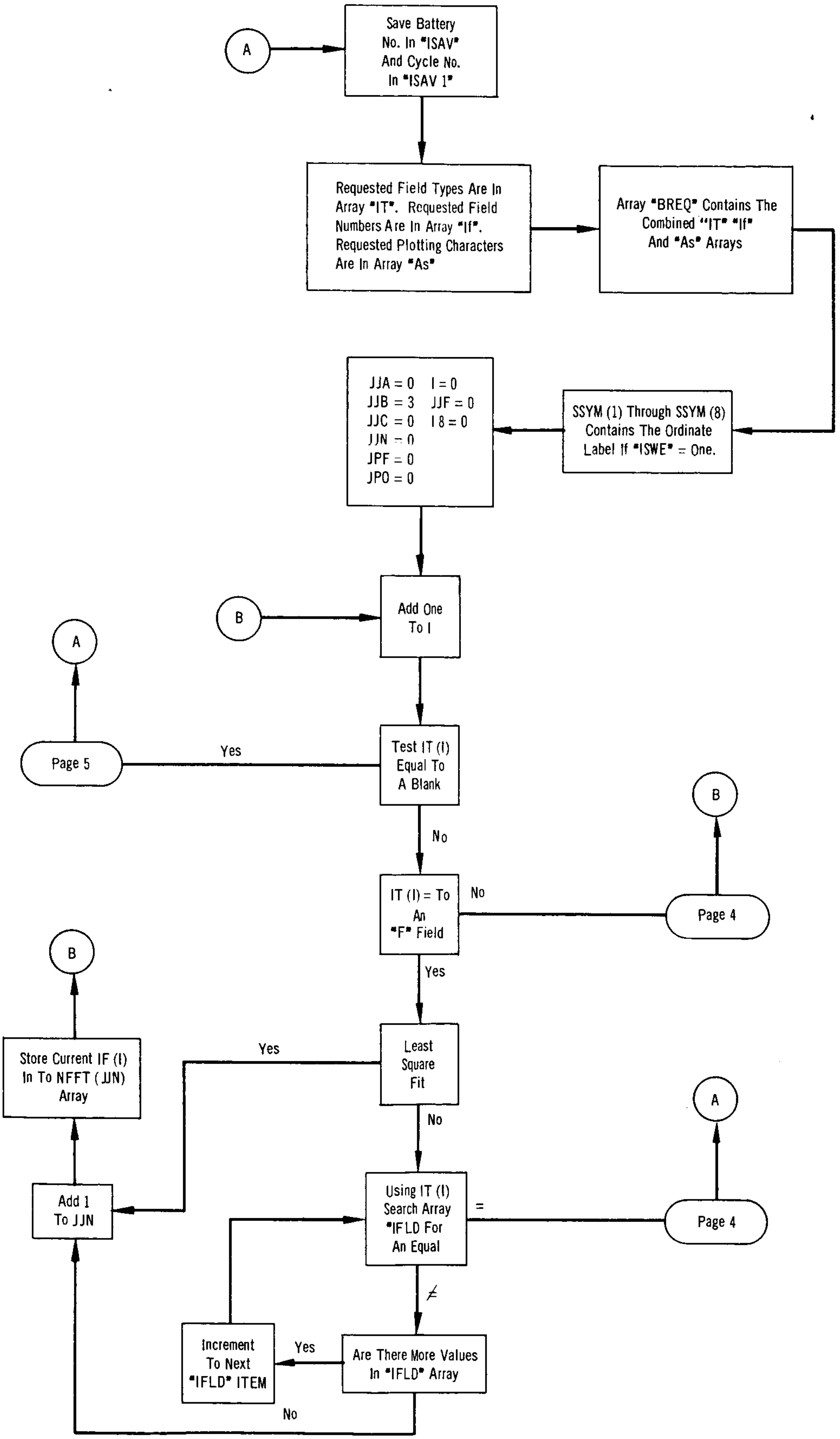

Is Data Tape Pseudo Or Retrieved
Retrieved
IDT = IR
Pseudo
IDT = IP
Read Control Record From Data Tape
Test If Additional Cycle Range Cards Exist
Yes
Read Cycle Range Card
Call Card
Is Data Tape a Retrieved Tape
No
Page 2
A
Yes
Is It Least Square Fit
Yes
Write Out An Error Comment Format No. 1002
Rewind Data Tape Pause
B
A
No
Page 2
B

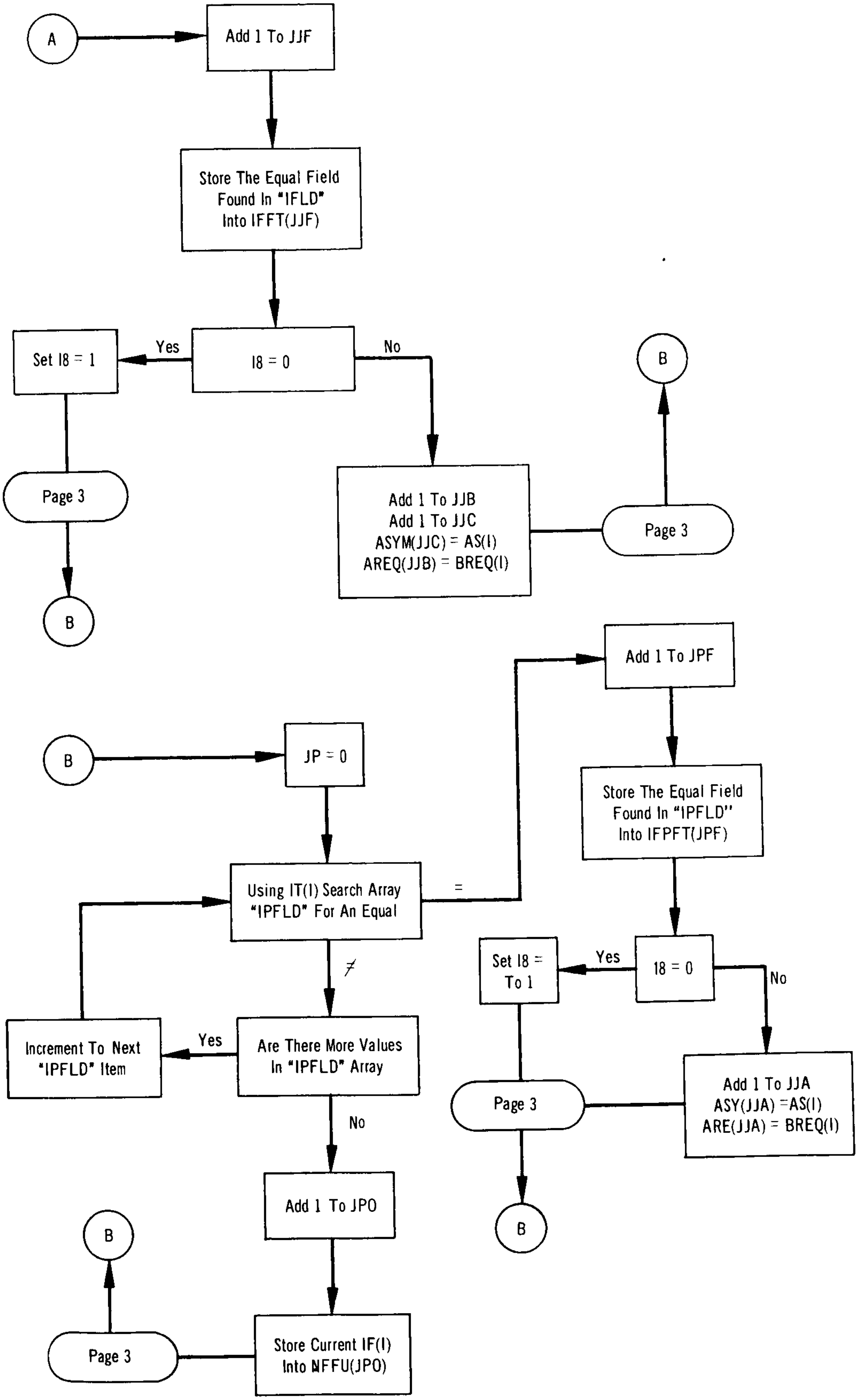
5a. Printer Plot Routine – Flow Charts (Continued)



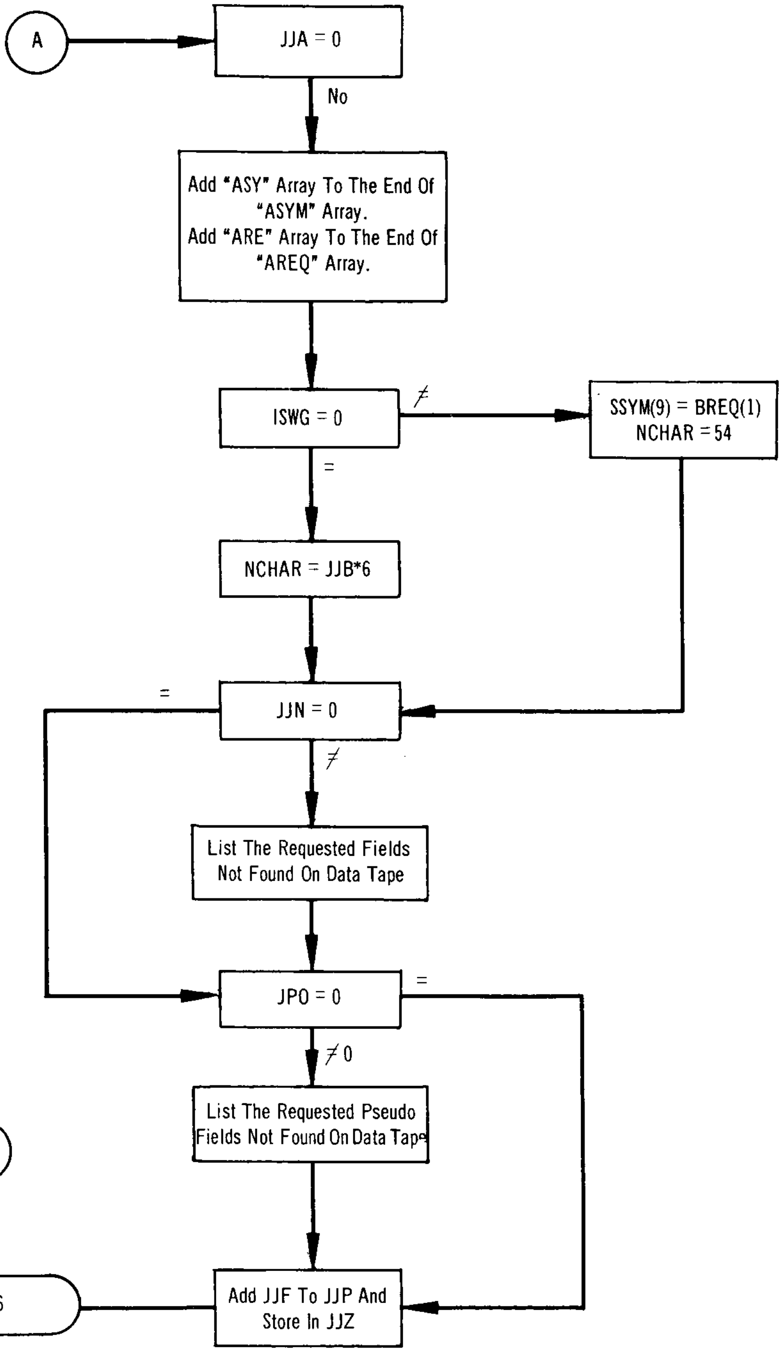
Print Comment Format No. 1002
No
Is Data Tape A Pseudo Tape
A
Yes
Page 1
B
Determine What Field Numbers Are On Data Tape. Store In Array "IFLD"
B
A
Does Data Tape Contain Pseudo Fields
No
JP = 0
Page 3
Yes
Read Pseudo Cards
Write Pseudo Cards On Output Print Tape
Read First Data Record
Backspace Data Tape
A
Calculate The Locations Of Pseudos. Setup An Array Of Calculated Pseudos IPFLD (JP)
Page 3

5a. Printer Plot Routine – Flow Charts (Continued)

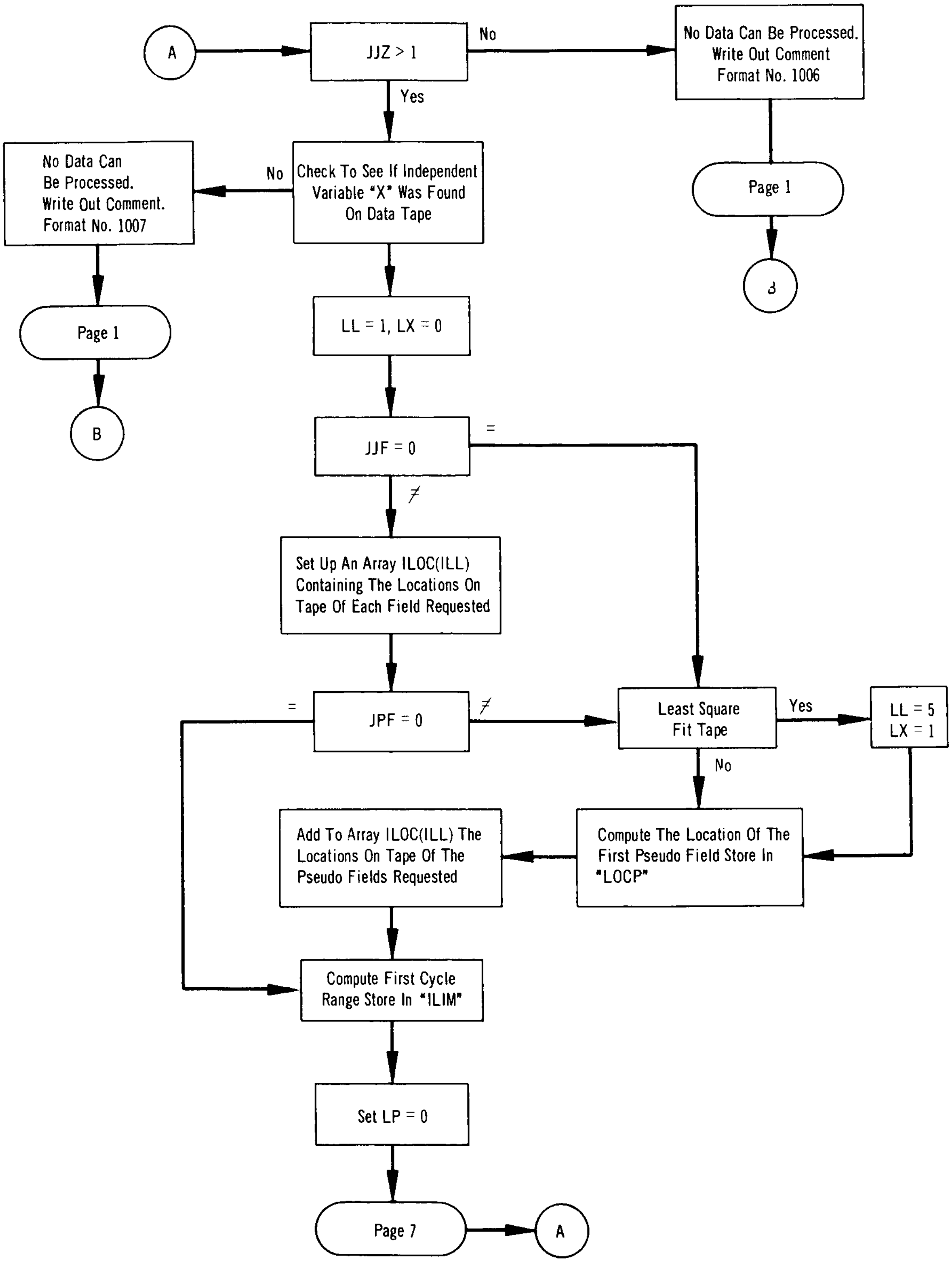

A
Save Battery No. In "ISAV" And Cycle No. In "ISAV 1"
Requested Field Types Are In Array "IT". Requested Field Numbers Are In Array "If". Requested Plotting Characters Are In Array "As"
Array "BREQ" Contains The Combined "IT" "If" And "As" Arrays
SSYM (1) Through SSYM (8) Contains The Ordinate Label If "ISWE" = One.
JJA = 0
I = 0
JJB = 3
JJF = 0
JJC = 0
I 8 = 0
JJN = 0
JPF = 0
JPO = 0
B
Add One To I
A
Page 5
Yes
Test IT (I) Equal To A Blank
No
B
IT (I) = To An "F" Field
No
Page 4
Yes
B
Store Current IF (I) In To NFFT (JJN) Array
Yes
Least Square Fit
No
A
Add 1 To JJN
Using IT (I) Search Array "IFLD For An Equal
=
Page 4
≠
Increment To Next "IFLD" ITEM
Yes
Are There More Values In "IFLD" Array
No


A
Add 1 To JJF
Store The Equal Field Found In "IFLD" Into IFFT(JJF)
Set I8 = 1
Yes
I8 = 0
No
B
Page 3
Add 1 To JJB
Add 1 To JJC
ASYM(JJC) = AS(I)
AREQ(JJB) = BREQ(I)
Page 3
B
Add 1 To JPF
B
JP = 0
Store The Equal Field Found In "IPFLD" Into IFPFT(JPF)
Using IT(I) Search Array "IPFLD" For An Equal
=
≠
Set I8 = To 1
Yes
I8 = 0
No
Increment To Next "IPFLD" Item
Yes
Are There More Values In "IPFLD" Array
Page 3
Add 1 To JJA
ASY(JJA) = AS(I)
ARE(JJA) = BREQ(I)
No
Add 1 To JPO
B
B
Page 3
Store Current IF(I) Into NFFU(JPO)

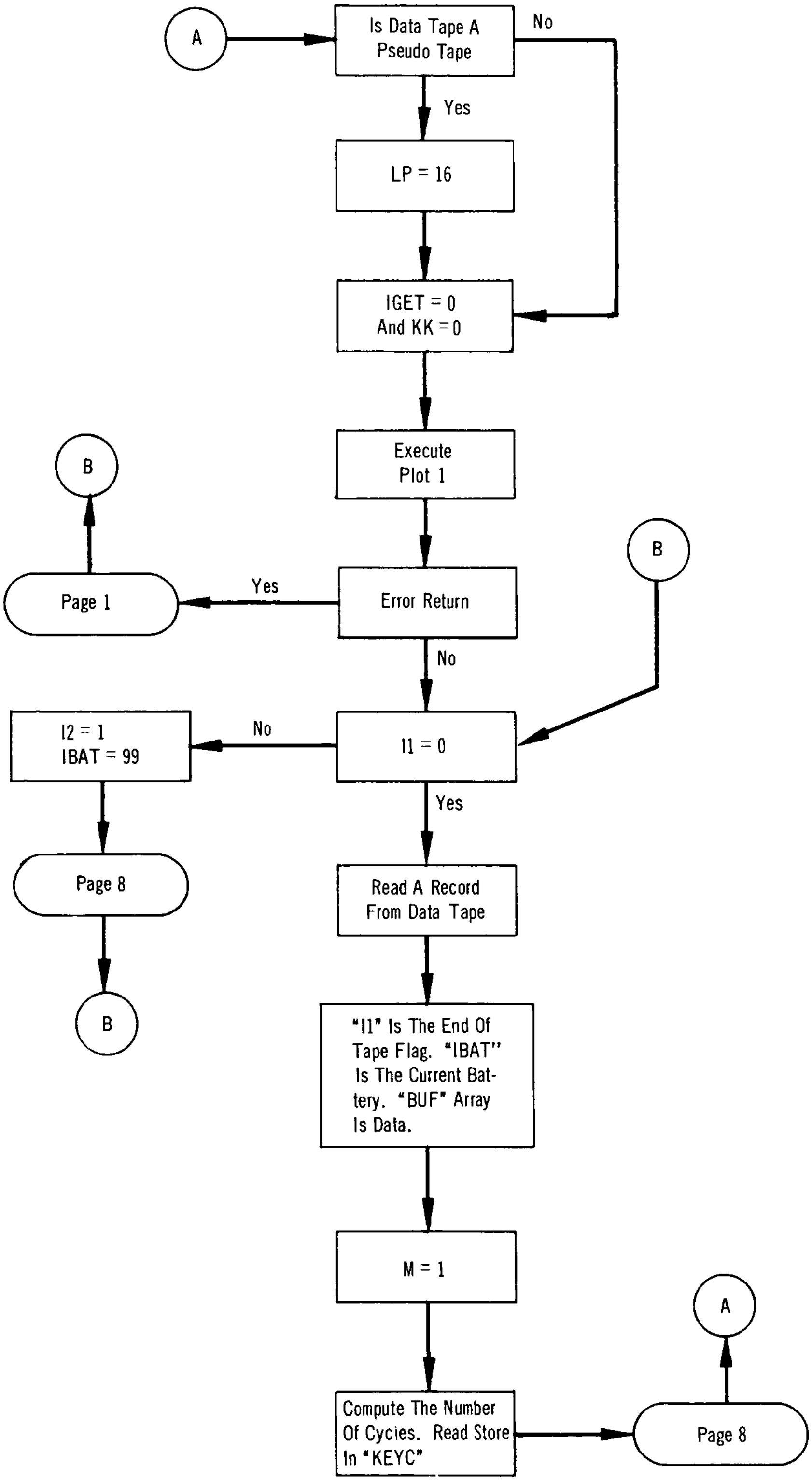
5a. Printer Plot Routine – Flow Charts (Continued)


A
JJA = 0
No
Add "ASY" Array To The End Of "ASYM" Array.
Add "ARE" Array To The End Of "AREQ" Array.
ISWG = 0
≠
SSYM(9) = BREQ(1)
NCHAR = 54
=
NCHAR = JJB*6
=
JJN = 0
≠
List The Requested Fields Not Found On Data Tape
JPO = 0
=
≠ 0
List The Requested Pseudo Fields Not Found On Data Tape
A
Page 6
Add JJF To JJP And Store In JJZ

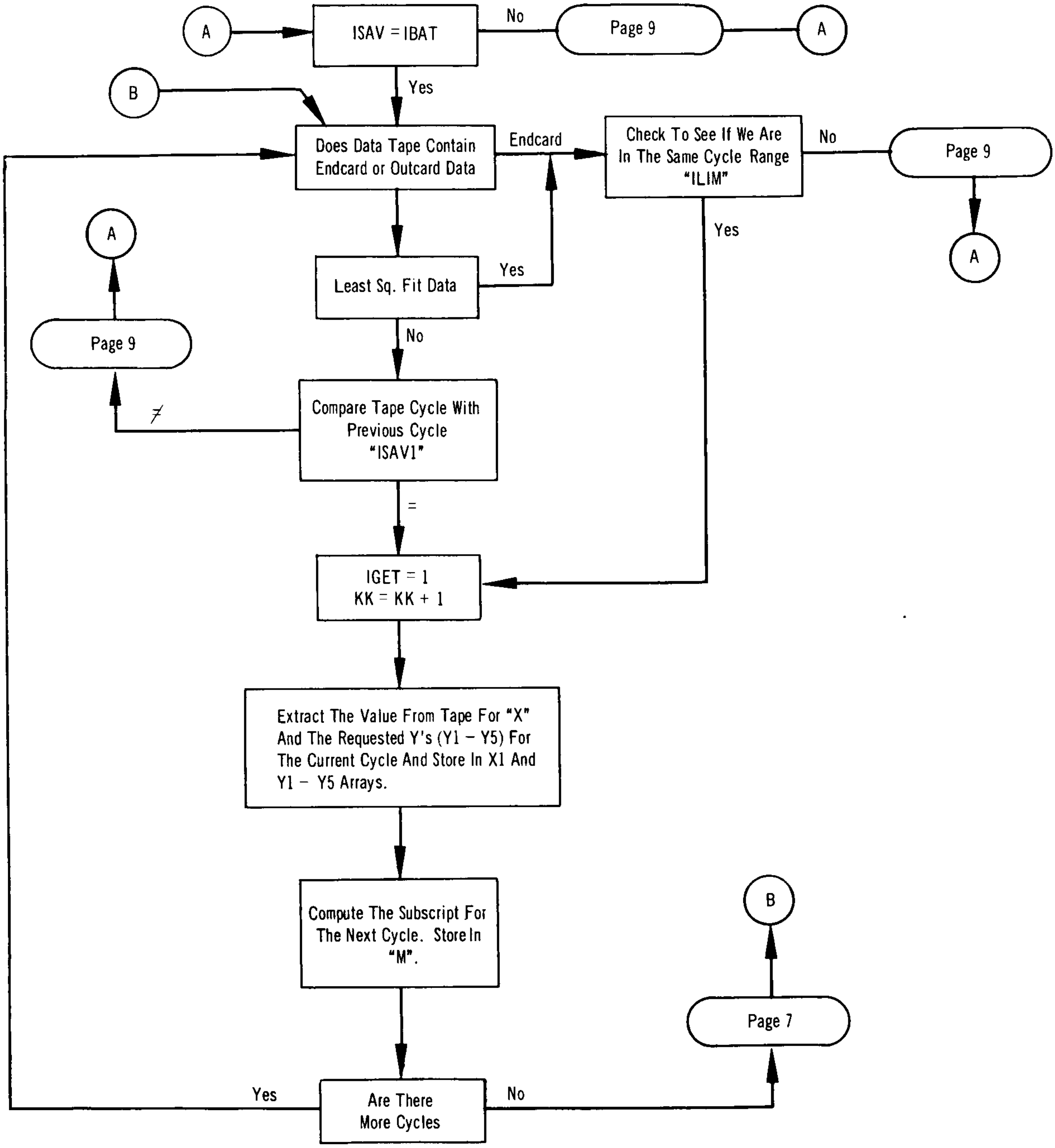
5a. Printer Plot Routine – Flow Charts (Continued)


A
JJZ > 1
No
No Data Can Be Processed. Write Out Comment Format No. 1006
Yes
Page 1
3
No Data Can Be Processed. Write Out Comment. Format No. 1007
No
Check To See If Independent Variable "X" Was Found On Data Tape
Page 1
B
LL = 1, LX = 0
JJF = 0
=
≠
Set Up An Array ILOC(ILL) Containing The Locations On Tape Of Each Field Requested
=
JPF = 0
≠
Least Square Fit Tape
Yes
LL = 5 LX = 1
No
Add To Array ILOC(ILL) The Locations On Tape Of The Pseudo Fields Requested
Compute The Location Of The First Pseudo Field Store In "LOCP"
Compute First Cycle Range Store In "ILIM"
Set LP = 0
Page 7
A

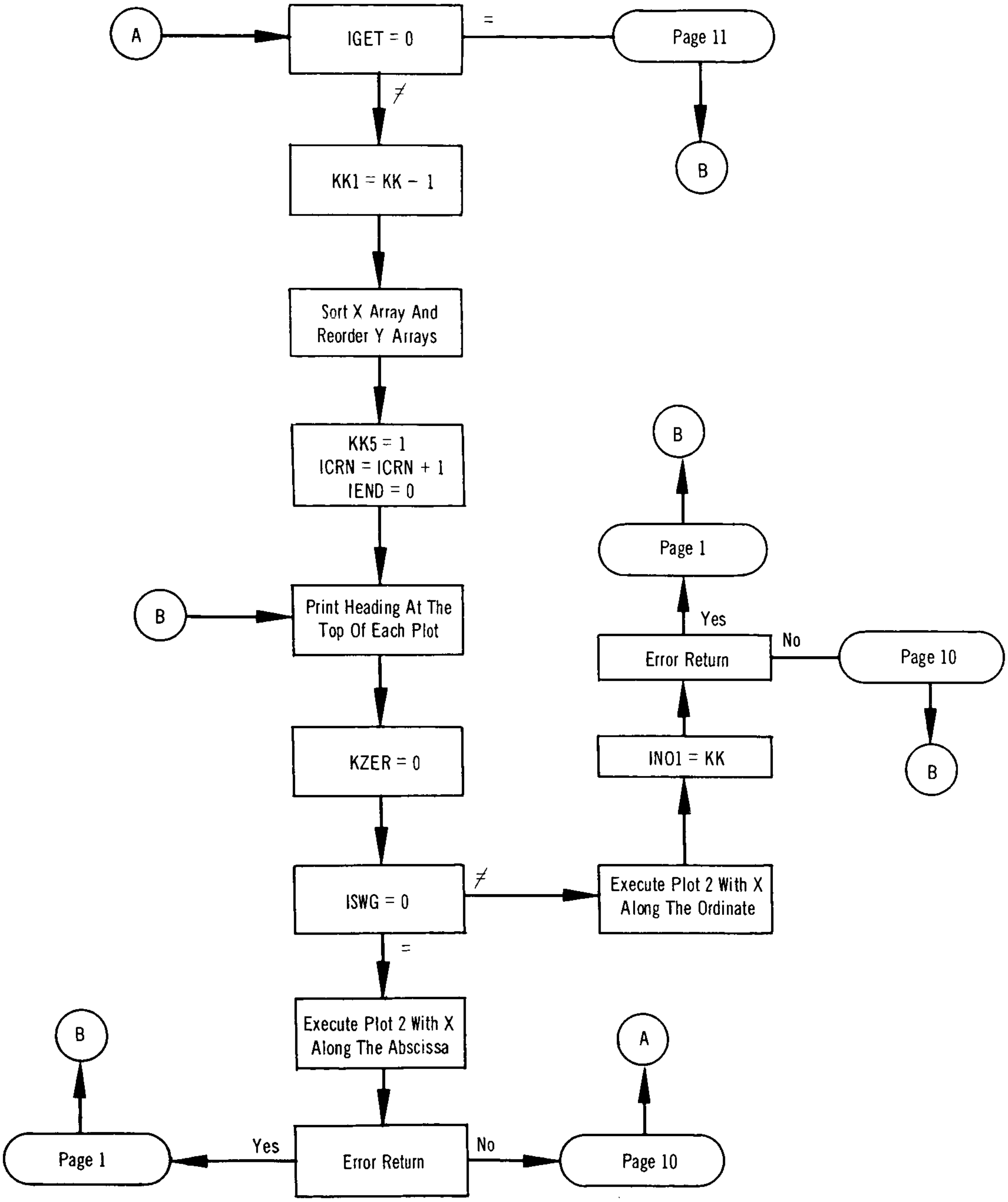
5a. Printer Plot Routine – Flow Charts (Continued)


A
Is Data Tape A Pseudo Tape
No
Yes
LP = 16
IGET = 0 And KK = 0
Execute Plot 1
B
B
Page 1
Yes
Error Return
No
I2 = 1 IBAT = 99
No
I1 = 0
Yes
Page 8
Read A Record From Data Tape
B
"I1" Is The End Of Tape Flag. "IBAT" Is The Current Battery. "BUF" Array Is Data.
M = 1
A
Compute The Number Of Cycles. Read Store In "KEYC"
Page 8

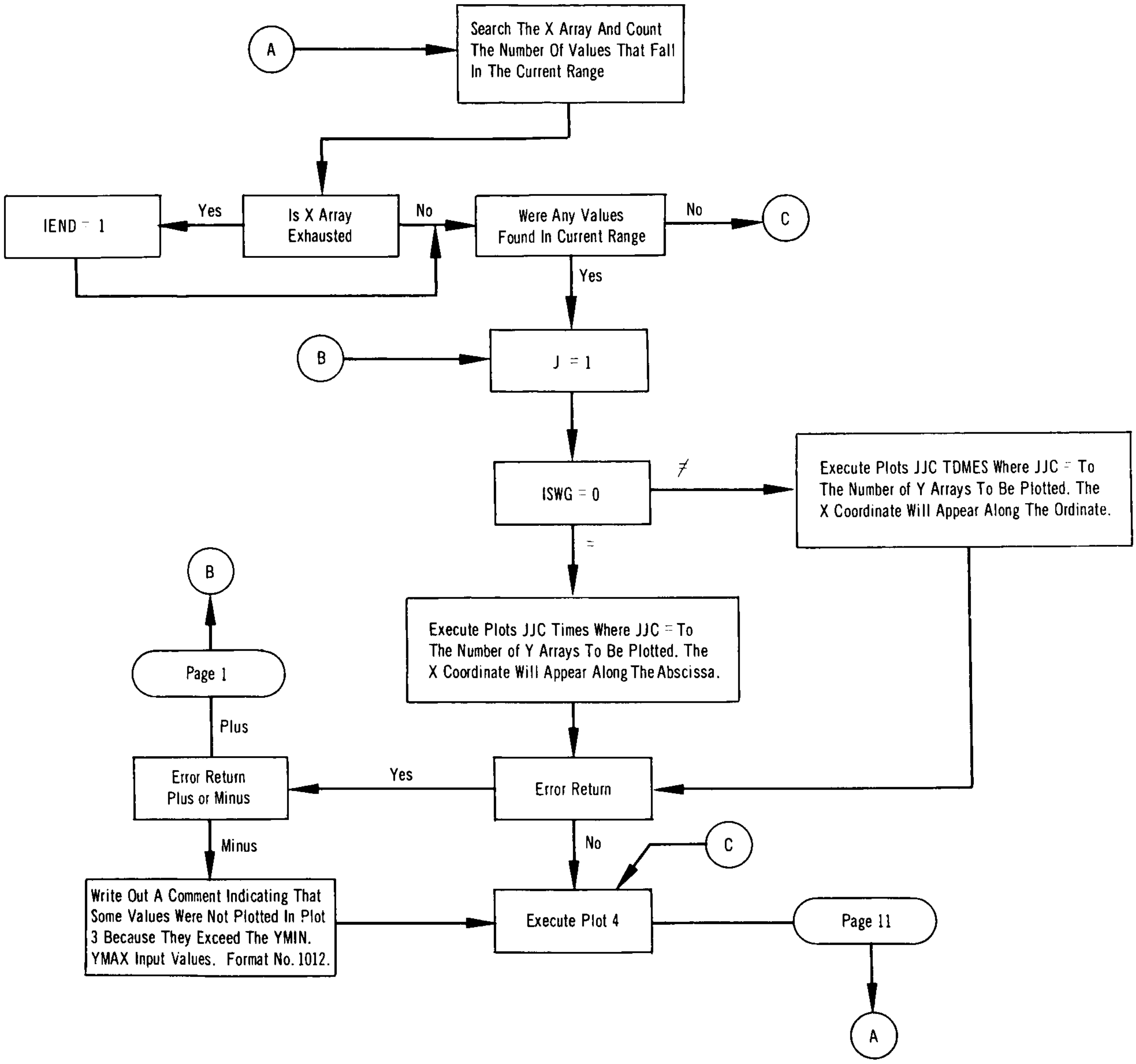
5a. Printer Plot Routine – Flow Charts (Continued)


A
ISAV = IBAT
No
Page 9
A
Yes
B
Does Data Tape Contain Endcard or Outcard Data
Endcard
Check To See If We Are In The Same Cycle Range "ILIM"
No
Page 9
A
A
Least Sq. Fit Data
Yes
Yes
Page 9
No
≠
Compare Tape Cycle With Previous Cycle "ISAV1"
=
IGET = 1
KK = KK + 1
Extract The Value From Tape For "X" And The Requested Y's (Y1 – Y5) For The Current Cycle And Store In X1 And Y1 – Y5 Arrays.
Compute The Subscript For The Next Cycle. Store In "M".
B
Page 7
Yes
Are There More Cycles
No

5a. Printer Plot Routine – Flow Charts (Continued)

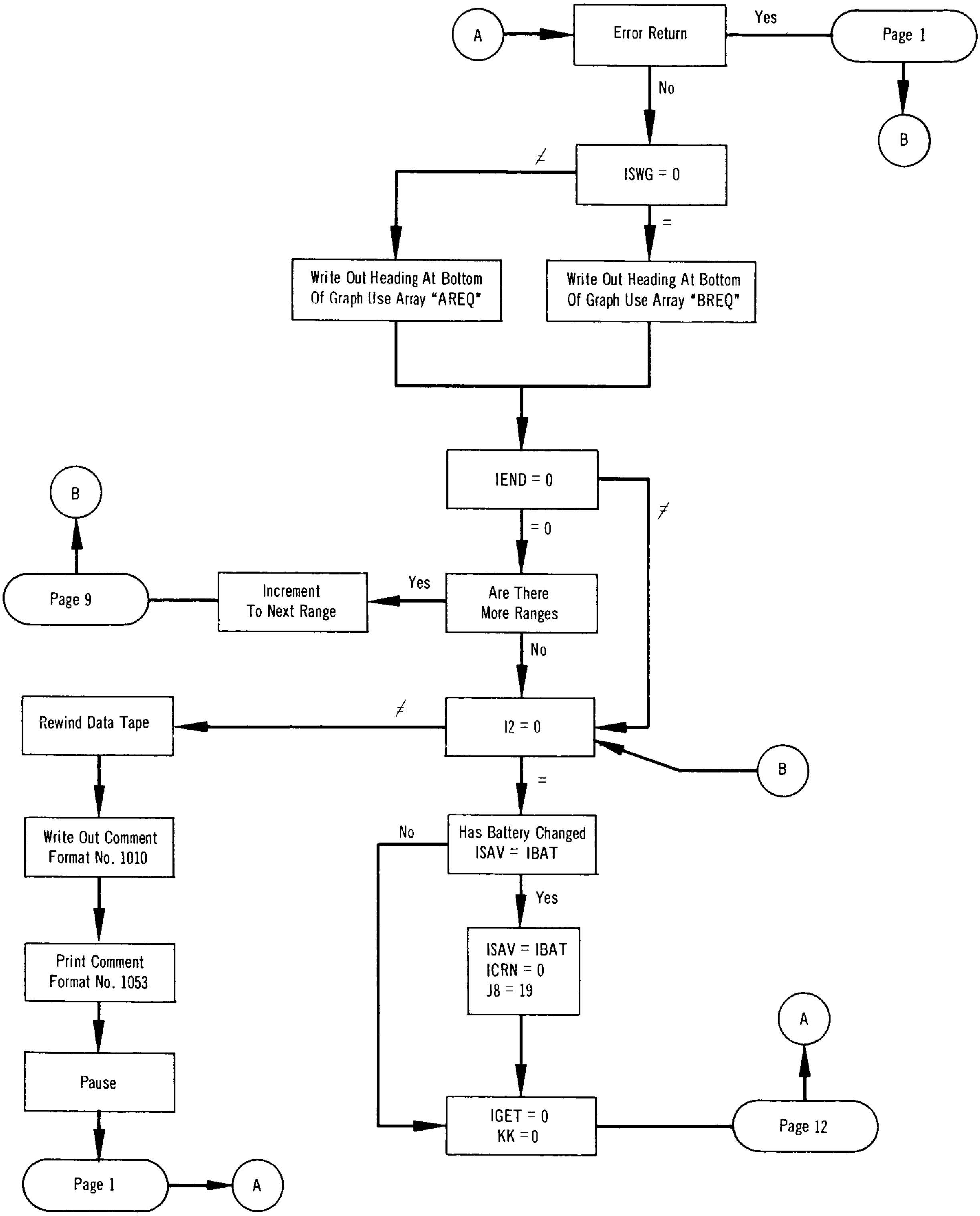


5a. Printer Plot Routine – Flow Charts (Continued)

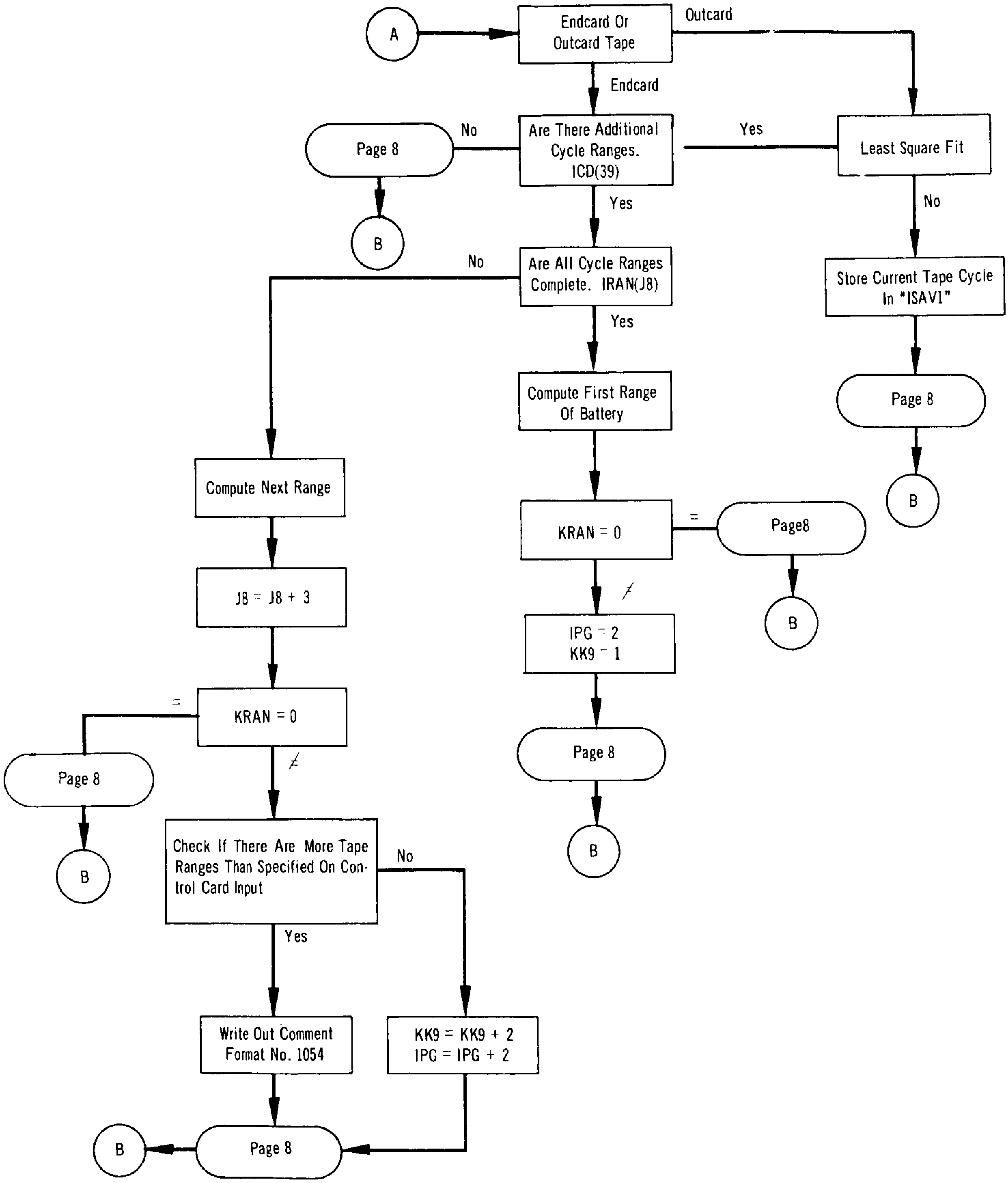


A
Search The X Array And Count The Number Of Values That Fall In The Current Range
IEND = 1
Yes
Is X Array Exhausted
No
Were Any Values Found In Current Range
No
C
Yes
B
J = 1
ISWG = 0
≠
Execute Plots JJC TDMES Where JJC = To The Number of Y Arrays To Be Plotted. The X Coordinate Will Appear Along The Ordinate.
=
B
Page 1
Execute Plots JJC Times Where JJC = To The Number of Y Arrays To Be Plotted. The X Coordinate Will Appear Along The Abscissa.
Plus
Error Return Plus or Minus
Yes
Error Return
Minus
No
C
Write Out A Comment Indicating That Some Values Were Not Plotted In Plot 3 Because They Exceed The YMIN. YMAX Input Values. Format No. 1012.
Execute Plot 4
Page 11
A

5a. Printer Plot Routine – Flow Charts (Continued)


A
Error Return
Yes
Page 1
B
No
ISWG = 0
≠
=
Write Out Heading At Bottom Of Graph Use Array "AREQ"
Write Out Heading At Bottom Of Graph Use Array "BREQ"
IEND = 0
≠
= 0
B
Page 9
Increment To Next Range
Yes
Are There More Ranges
No
Rewind Data Tape
≠
I2 = 0
B
=
Write Out Comment Format No. 1010
No
Has Battery Changed ISAV = IBAT
Yes
ISAV = IBAT
ICRN = 0
J8 = 19
Print Comment Format No. 1053
A
Pause
IGET = 0
KK = 0
Page 12
Page 1
A

5a. Printer Plot Routine – Flow Charts (Continued)



A
Endcard Or Outcard Tape
Outcard
Endcard
Page 8
No
Are There Additional Cycle Ranges. ICD(39)
Yes
Least Square Fit
B
Yes
No
No
Are All Cycle Ranges Complete. IRAN(J8)
Store Current Tape Cycle In "ISAV1"
Yes
Compute First Range Of Battery
Page 8
B
Compute Next Range
KRAN = 0
=
Page8
J8 = J8 + 3
≠
B
IPG = 2
KK9 = 1
=
KRAN = 0
Page 8
Page 8
≠
B
Check If There Are More Tape Ranges Than Specified On Control Card Input
No
B
Yes
Write Out Comment Format No. 1054
KK9 = KK9 + 2
IPG = IPG + 2
B
Page 8

5b. Printed Plot Routine – Source Statements

```
$LOG    000403
*       BATTERY TEST DATA PLOT ROUTINE       LRL P13P     (A. F.)
*       LRL (AL FLINN)    P13P      BLDG 1
*       PAUSE
*       SYMBOL TABLE
*       CARDS COLUMN
*       LIST8
*       FORMAP
*       LABEL
CBPP         BATTERY TEST PRINTER PLOT ROUTINE
        DIMENSION  IMAGE(7000),BUF(4500),ICARD(1200),ICD(42),IRAN(19),
       1 AS(6),IF(6),IT(7),NSCALE(5),ARE(5),BREQ(6),ASYM(5),NFFU(6),
       1 IEPFT(6),IPFLD(16),ASY(5),AREQ(8),ILOC(6),NFFT(6),IFFT(6),
       1 IFLD(26),X1(500),Y1(500),Y2(500),Y3(500),Y4(500),Y5(500),
       1 IBU(1),XGRD(50),SSYM(9)
        EQUIVALENCE  (F,KF),(P,KP),(BLK,IBLK),(BUF,IBU),(FLR,IFLR),
       1(FLP,IFLP)
C       SET UP TAPE NOS.
        IO = 3
        IN = 2
        IP = 17
        IR = 19
        FLR = 6H$$$$$R
        FLP = 6H$$$$$P
        F =1HF
        P =1HP
        BLK =1H
        ATT=6HTHE
        ATT1=6H  X  X
        ATT2=6H    VA
        ATT3=6HRIABLE
        ATT4=6H FIELD
        AT5=6H IS
        SSYM(1) = BLK
        SSYM(2) = BLK
        SSYM(3) = ATT
        SSYM(4) = ATT1
        SSYM(5) = ATT2
        SSYM(6) = ATT3
        SSYM(7) = ATT4
        SSYM(8) = AT5
10      ICRN = 0
        I1=0
        I2=0
        J8 = 1
        KK9 = 1
        IRAN(19) = 0
        IT(7)= IBLK
C       READ CONTROL CARD
        READ INPUT TAPE IN,1000,IOP,(NSCALE(I),I=1,5),NHL,NSBH,NVL,NSBV,
       1 IT(1),IF(1),(IT(J),IF(J),AS(J),J=2,6)
1000    FORMAT(6I1,4(1X,I2),1X,A1,I2,5(A1,I2,A1))
        BACKSPACE  IN
        READ INPUT TAPE IN,1013,(BREQ(I),I=1,6)
1013    FORMAT (19X,A3, 5A4 )
C....   READ IN  MIN.AND MAX. VALUES FOR Y
        READ INPUT TAPE IN,1050,YMIN,YMAX,IPG,ISWG,KRAN
1050    FORMAT ( 2E10.4,I2,2I1)
```

5b. Printed Plot Routine – Source Statements (Continued)

```
C.... READ IN GRID RANGES
      READ INPUT TAPE IN,1051,(XGRD(I),I=1,IPG)
 1051 FORMAT(6F10.4)
C     WRITE OUT CONTROL CARD
C     TEST IF IGRD ARRAY IS FOR COMPLETE RANGE OR IF EACH PAIR IS A RANGE
      IF(KRAN) 501,500,501
 501  IPGG = IPG
      IPG = 2
 500  WRITE OUTPUT TAPE IO,1001, IOP,II(1),IE(1), (II(J),IE(J),AS(J),J
     1 =2,6)
 1001 FORMAT(1H16X,25HDATA PLOT REQUEST. OPTION,I2,28H           X      VAR.
     1 FIELD IS ,A1,I2,56H.      Y      VARIABLE FIELDS WITH PLOTTING CHAR
     1ACTERS ARE,/7X,5(A1,I2,A1,3X))
      AREQ(1) =  BLK
      AREQ(2) =  BLK
      AREQ(3) =  BLK
C     DETERMINE IF DATA TAPE TO READ IS B9 OR B7
      IF( IOP - 1)  51,50,51
 50   IDT=IP
      GO TO 52
 51   IDT=IR
C     READ FIRST CONTROL RECORD FROM DATA TAPE
 52   READ TAPE IDT, IFL,ICD
C     TEST IF ADDITIONAL CYCLE RANGE CARD EXISTS
      IF(ICD(39))   53,54,53
 53   READ TAPE IDT, (IRAN(I),I=1,18)
C     WRITE CONTROL INFO. ON OUTPUT TAPE
 54   CALL CCARD(ICD,IRAN)
C     RETRIEVED TAPE WITH L.S.F.IS AN ERROR REQUEST
      IF (IFL - IFLR) 300,55,300
 55   IF(ICD(18)-1)57,56,57
C     CANNOT PROCESS THIS TAPE
 56   WRITE OUTPUT TAPE IO,1002
 1002 FORMAT (73H0INPUT DATA TAPE IS OUTCARD LEAST SQ. WITHOUT PSEUDO FI
     1ELDS. CANNOT PLOT.)
 68   REWIND IDT
      PAUSE
      GO TO 10
 300  IF( IFL - IFLP) 301,57,301
 301  PRINT 1020
 1020 FORMAT (57H  INPUT TAPE IS NOT A RETRIEVED OR PSEUDO TAPE. SCRAP R
     1UN)
      GO TO 68
C     DETERMINE WHAT INFORMATION IS AVAILABLE ON IDT
 57   IF(ICD(41))59,58,59
 58   J = 20
      GO TO 60
 59   J=21
 60   JJ =0
      IF (ICD(J)) 61,64,61
 61   DO  63  I=J,38
      IF(ICD(I)) 62,71,62
 62   JJ = JJ +1
 63   IFLD(JJ) = ICD(I)
      GO TO 71
C     DETERMINE IF ENDCARD OR OUTCARD
 64   IF(ICD(1)) 65,66,65
 65   JJ = 26
```

5b. Printed Plot Routine – Source Statements (Continued)

```
      GO TO  67
66    JJ = 15
67    DO 70 I=1,JJ
70    IFLD(I) =  I
C     DETERMINE IF IDT IS L.S TYPE
71    LL =1
      LX =0
      IF(ICD(18)-1) 73,72,73
72    LL= 5
      LX= 1
C     DOES DATA TAPE CONTAIN PSEUDO FIELDS
73    IF(IFL - IFLP) 83,75,83
C     READ PSEUDO EQUATIONS
75    READ TAPE IDT, I, (ICARD(K),K=1,I)
C     WRITE OUT PSEUDO EQUATION CARDS
      WRITE OUTPUT TAPE IO, 1003, (ICARD(K),K=1,I)
1003  FORMAT (25H0   PSEUDO EQUATION CARDS //(1X,12A6))
C     READ ONE RECORD FROM IDT AND SET UP PSEUDO FIELD TABLE (IPFLD)
      READ TAPE IDT, I1,ICNT,IDAT, (BUF(I),I=1,ICNT)
      ISAV = IDAT
      ISAV1= XABSF(IBU)
      BACKSPACE IDT
C     CALCULATE LOCATIONS OF PSEUDOS
      NLOC = (JJ * LL) + LX +2
C     GENERATE A TABLE OF CALCULATED PSEUDOS
      JP = 0
      DO 80 I=1,16
      IF(BUF(NLOC) - 1.E36)  81,82,81
81    JP = JP+1
      IPFLD(JP) = I
82    NLOC = NLOC +1
80    CONTINUE
      GO TO 84
83    JP =0
84    JJA= 0
      JJB = 3
      JJC = 0
      JJN = 0
      JPF = 0
      JPO = 0
      I=0
      JJF=0
      I8 = 0
94    I=I+1
      IF(IT(I) -IBLK)  90,101,90
90    IF(IT(I) -KF) 95,91,95
91    IF(ICD(18)-1) 85,96,85
85    DO 93 J=1,JJ
      IF (IF(I) - IFLD(J))93,92,93
92    JJF = JJF +1
      IFFI(JJF) = IFLD(J)
      IF(I8)   87 ,86, 87
86    I8 = 1
      GO TO  94
87    JJB  = JJB +1
      JJC = JJC +1
      ASYM(JJC)= AS(I)
      AREQ(JJB)= BREQ(I)
```

5b. Printed Plot Routine – Source Statements (Continued)

```
      GO TO  94
93    CONTINUE
96    JJN = JJN +1
      NFFT(JJN) = IF(I)
      GO TO 94
95    IF(JP) 98,100,98
98    DO 99   J=1, JP
      IF (IF(I) - IPFLD(J)) 99,97,99
97    JPF = JPF+1
      IFPFT(JPF) = IPFLD(J)
      IF(I8)  89,88,89
88    I8 =1
      GO TO 94
89    JJA = JJA+1
      ASY (JJA) = AS(I)
      ARE (JJA) = BREQ(I)
      GO TO 94
99    CONTINUE
100   JPO = JPO +1
      NFFU(JPO) = IF(I)
      GO TO 94
C     ADD ASY AND ARE  ARRAYS TO  ASYM AND AREO ARRAYS
101   IF(JJA) 103,108,103
103   DO 104   I =1, JJA
      JJB = JJB +1
      JJC = JJC +1
      ASYM(JJC) = ASY(I)
      AREQ(JJB) = ARE(I)
104   CONTINUE
C     ARE WE SWITCHING X,Y, COORIDINATES
C     CALCULATE THE NO. OF CHAR. FOR ORDINATE LABEL IN PLOT 4
108   IF(ISWG) 120,109,120
120   SSYM(9)=BREQ(1)
      NCHAR=54
      GO TO 102
109   NCHAR=JJB * 6
C     LIST THOSE FIELDS NOT FOUND
102   IF (JJN) 105,106,105
105   WRITE OUTPUT TAPE IO,1004, (NFFT(I),I=1,JJN)
1004  FORMAT (63H0 THE FOLLOWING FIELDS WERE NOT FOUND ON THE INPUT DATA
     1 TAPE.../ 6I4)
C     LIST THOSE PSEUDOS NOT FOUND
106   IF(JPO) 107,110,107
107   WRITE OUTPUT TAPE IO, 1005, (NFFU(I),I=1,JPO)
1005  FORMAT (70H0 THE FOLLOWING PSEUDO FIELDS WERE NOT FOUND ON THE INP
     1UT DATA TAPE.../ 6I4)
C     CAN ANY DATA BE PROCESSED
110   JJZ = JJF + JPF
      IF(JJZ-1)      111,111,112
C     NO DATA CAN BE PROCESSED
111   WRITE OUTPUT TAPE IO,1006
1006  FORMAT(43H0 DATA CANNOT BE PROCESSED. RUN TERMINATED.)
      GO TO 68
C     CHECK TO SEE IF INDEPENDENT VARIABLE X WAS FOUND ON  IDT
112   IF(IT(1) - KF)    115,113,115
113   IF (IF - IFFT) 114,116,114
114   WRITE OUTPUT TAPE IO, 1007
1007  FORMAT (56H0 INDEPENDENT VARIABLE CANNOT BE LOCATED. RUN TERMINATE
```

5b. Printed Plot Routine – Source Statements (Continued)

```
      1D)
      GO TO 68
115   IF (IF - IFPFT) 114,116,114
C     SETUP AN ARRAY CONTAINING THE LOCATIONS ON TAPE OF EACH FIELD
C     REQUESTED.
116   LL=1
      LX=0
      ILL =0
C     WERE FIELDS REQUESTED
      IF (JJF) 130,140,130
130   IF(IT - KF ) 135,132,135
135   ILL = 1
132   DO 134 I=1,JJF
      ICNT = 0
      DO 131  J=1,JJ
      ICNT = ICNT +1
      IF(IFFT(I) - IFLD(J)) 131,133,131
133   ILL = ILL +1
      ILOC(ILL) = ICNT
      GO TO 134
131   CONTINUE
134   CONTINUE
C     PSEUDO FIELDS REQUESTED
      IF(JPF)143,150,143
140   ILL=1
C     SETUP LOCATIONS FOR PSEUDO FIELDS IN ARRAY  ILOC
143   IF(ICD(18)-1) 142,141,142
141   LL = 5
      LX = 1
142   LOCP = JJ * LL + LX
      J=1
      IF(IT-KF)127,126,127
127   ILOC = IFPFT + LOCP
      IF(JPF -1) 150,150,128
128   J=2
126   DO 145 I=J,JPF
      ILL = ILL +1
      ILOC(ILL) =  IFPFT(I) + LOCP
145   CONTINUE
C     COMPUTE FIRST RANGE
150   ILIM = ICD(14) + ICD(15) * (ICD(16)-1)
C
146   LP = 0
      IF(IFL - IFLP) 129,147,129
147   LP = 16
129   IGET = 0
      KK = 0
      R = PLOT1(NSCALE,NHL,NSBH,NVL,NSBV)
      IF(R) 149,149,68
149   I1 = 0
148   IF(I1)215,151,215
215   I2=1
      IBAT=99
      GO TO 213
151   READ TAPE IDT,I1,ICNT,IBAT,(BUF(I),I=1,ICNT)
      M=1
C     COMPUTE THE NUMBER OF CYCLES READ
      KCYC = ICNT / (( JJ * LL) +1 + LX + LP)
```

5b. Printed Plot Routine – Source Statements (Continued)

```
C       EXTRACT DATA TO PLOT AND STORE SEPARATE ARRAYS.
C       HAVE WE CHANGED BATTERYS
213     DO 250 I=1,KCYC
        IF(ISAV-IBAT) 200,214,200
214     IF(ICD) 153,136,153
C       ARE WE IN THE SAME CYCLE RANGE(ENDCARD)
153     IF(XABSF(IBU(M))-ILIM) 154,154,200
C       ARE WE IN THE SAME CYCLE (OUTCARD)
C       CHECK LEAST SQ. FIT
136     IF(ICD(18)-1) 137,153,137
137     IF(XABSF(IBU(M))-ISAV1) 200,154,200
154     IGET = 1
        KK = KK +1
        J = 1
        IAD = ILOC(J) + M
        X1(KK) = BUF(IAD)
        J = J+1
158     IAD=ILOC(J) + M
        Y1(KK)=BUF(IAD)
163     IF (J - ILL) 164,190,164
164     J = J+1
        IAD=ILOC(J) + M
        Y2(KK) = BUF(IAD)
168     IF(J - ILL) 170,190,170
170     J = J+1
        IAD=ILOC(J) + M
        Y3(KK) = BUF(IAD)
174     IF(J - ILL) 175,190,175
175     J = J+1
        IAD=ILOC(J) + M
        Y4(KK) = BUF(IAD)
179     IF(J - ILL) 180,190,180
180     J=J+1
        IAD=ILOC(J) + M
        Y5(KK) = BUF(IAD)
C       COMPUTE SUBSCRIPT FOR NEXT CYCLE
190     M = M +(JJ * LL) +1 + LX + LP
        GO TO 250
C       X AND Y ARRAYS ARE SETUP FOR A COMPLETE RANGE IF ENDCARD OR A
C       COMPLETE CYCLE IF OUTCARD .
C       SORT X  ARRAY AND REORDER  Y  ARRAYS
C       CHECK TO SEE IF ANY CYCLES WERE EXTRACTED
200     IF(IGET)  202,240,202
202     KK1 = KK -1
        DO 205 L1 =1, KK1
        L4 = 0
        XX1 = X1(L1)
        L2 = L1 +1
        DO 206 L3= L2,KK
        IF (X1(L3) - XX1) 207,206,206
207     L4  = L3
        XX1 = X1(L3)
206     CONTINUE
        IF(L4) 201,205,201
201     J = 2
        X1(L4)  = X1(L1)
        X1(L1) = XX1
        XX1 = Y1(L4)
```

5b. Printed Plot Routine – Source Statements (Continued)

```
        Y1(L4) = Y1(L1)
        Y1(L1) = XX1
        IF( J - ILL) 208,205,208
 208    J = J+1
        XX1 = Y2(L4)
        Y2(L4) = Y2(L1)
        Y2(L1) = XX1
        IF (J - ILL) 209,205,209
 209    J = J+1
        XX1 = Y3(L4)
        Y3(L4) = Y3(L1)
        Y3(L1) = XX1
        IF (J - ILL) 210,205,210
 210    J = J+1
        XX1 = Y4(L4)
        Y4(L4) = Y4(L1)
        Y4(L1) = XX1
        IF (J - ILL) 211,205,211
 211    XX1 = Y5(L4)
        Y5(L4) = Y5(L1)
        Y5(L1) = XX1
 205    CONTINUE
 204    KK5=1
        ICRN=ICRN+1
        IEND = 0
        DO 254 K = KK9,IPG,2
C       PRINT HEADING AT THE TOP OF EACH PLOT
        IF(ICD(18)-1) 212,227,212
 212    IF(ICD) 227,228,227
 227    WRITE OUTPUT TAPE IO,1008,ISAV,ICRN
 1008   FORMAT (1H145X,7HBATTERY,I4,12H CYCLE RANGE,I3//)
        GO TO 405
 228    WRITE OUTPUT TAPE IO,1009,ISAV,ISAV1
 1009   FORMAT (1H145X,7HBATTERY,I4, 6H CYCLE,I5//)
 405    KZER =0
        IF(ISWG) 451,450,451
 450    R=PLOT2(IMAGE,XGRD(K+1),XGRD(K),YMAX,YMIN)
        IF(R) 406,406,68
 451    R=PLOT2(IMAGE,YMAX,YMIN,XGRD(K+1),XGRD(K))
        INO1=KK
 452    IF(R) 411,411,68
 406    DO 410 J=KK5,KK
        IF(X1(J) - XGRD(K+1)) 408,408,409
 409    INO1 = J - KK5
        IMAX1 = KK5
        KK5 = J
        GO TO 415
 408    KZER = 1
 410    CONTINUE
        IEND = 1
        INO1 = KK - KK5 +1
        IMAX1 = KK5
C       CHECK TO SEE IF ANY VALUES WERE FOUND IN CURRENT RANGE. CHARGE
 415    IF(KZER) 411,226,411
 411    J = 1
        IF(ISWG) 454,453,454
 453    R=PLOT3(ASYM(J),X1(IMAX1),Y1(IMAX1),INO1)
        GO TO 455
```

5b. Printed Plot Routine – Source Statements (Continued)

```
454    R=PLOT3(ASYM(J),Y1,X1,INO1)
455    IF(R) 232,233,68
232    L50 = 1
       GO TO 270
233    IF(JJC-J) 222,226,222
222    J = J+1
       IF(ISWG) 457,456,457
456    R=PLOT3(ASYM(J),X1(IMAX1),Y2(IMAX1),INO1)
       GO TO 458
457    R=PLOT3(ASYM(J),Y2,X1,INO1)
458    IF(R) 234,235,68
234    L50 = 2
       GO TO 270
235    IF(JJC -J) 223,226,223
223    J= J+1
       IF(ISWG) 460,459,460
459    R=PLOT3(ASYM(J),X1(IMAX1),Y3(IMAX1),INO1)
       GO TO 461
460    R=PLOT3(ASYM(J),Y3,X1,INO1)
461    IF(R) 236,237,68
236    L50 = 3
       GO TO 270
237    IF(JJC -J) 224,226,224
224    J = J+1
       IF(ISWG) 463,462,463
462    R=PLOT3(ASYM(J),X1(IMAX1),Y4(IMAX1),INO1)
       GO TO 464
463    R=PLOT3(ASYM(J),Y4,X1,INO1)
464    IF(R) 238,239,68
238    L50 = 4
       GO TO 270
239    IF(JJC -J) 225,226,225
225    J = J+1
       IF(ISWG) 466,465,466
465    R=PLOT3(ASYM(J),X1(IMAX1),Y5(IMAX1),INO1)
       GO TO 467
466    R=PLOT3(ASYM(J),Y5,X1,INO1)
467    IF(R) 251,226,68
251    L50 = 5
       GO TO 270
226    IF(ISWG) 469,468,469
468    R=PLOT4(NCHAR,AREQ)
       GO TO 470
469    R=PLOT4(NCHAR,SSYM)
470    IF(R) 253,253,68
C      WRITE OUT  FIELD HEADING FOR  INDEPENDENT VARIABLE
253    IF(ISWG) 471,472,471
472    WRITE OUTPUT TAPE IO,1011,BREQ(1)
1011   FORMAT(1H0,40X,34HTHE      X      VARIABLE FIELD IS ,A3)
       GO TO 473
471    WRITE OUTPUT TAPE IO,1052,(AREQ(J),J=1,JJB)
1052   FORMAT(1H0,25X,8A6)
473    IF(IEND) 240,254,240
270    WRITE OUTPUT TAPE IO,1012, ASYM(J)
1012   FORMAT(6X,44HPLOT3 FAILED TO PLOT ALL POINTS FOR SYMBOL  ,A1)
       GO TO (233,235,237,239,226),L50
254    CONTINUE
C      ALL PLOTS FOR A CYCLE OR RANGE ARE COMPLETED. SETUP FOR NEXT SET
```

5b. Printed Plot Routine – Source Statements (Continued)

```
C       HAS BATTERY CHANGED
240     IF(I2) 260,244,260
244     IF(ISAV-IBAT) 241,242,241
241     ISAV = IBAT
        ICRN=0
        J8 = 19
242     IGET =0
        KK = 0
C       ENDCARD OR OUTCARD
        IF(ICD) 243,248,243
C       ENDCARD
C       ARE THERE ADDITIONAL  RANGES
243     IF(ICD(39)) 245,214,245
C       ARE ALL RANGES COMPLETE
245     IF(IRAN(J8)) 246,247,246
C       NO. SETUP NEXT RANGE
246     ILIM = IRAN(J8) + IRAN(J8+1) * ( IRAN(J8+2) -1)
        J8 = J8+3
        IF(KRAN) 502,214,502
502     IF(IPG - IPGG) 505,504,504
504     WRITE OUTPUT TAPE IO,1054
1054    FORMAT (1H1,10X,66HRUN TERMINATED. TAPE RANGES EXCEED INPUT RANGES
       1 ON CONTROL CARD 3)
        GO TO 68
505     KK9 = KK9 +2
        IPG = IPG +2
        GO TO 214
C       SETUP UP FIRSTRANGE
247     ILIM = ICD(14) + ICD(15) * (ICD(16)-1)
        J8 = 1
        IF(KRAN) 503,214,503
503     IPG = 2
        KK9 = 1
        GO TO 214
248     IF(ICD(18)-1) 249,243,249
249     ISAV1= XABSF(IBU(M))
        GO TO 214
250     CONTINUE
        GO TO 148
C       TAPE COMPLETELY PROCESSED
260     REWIND IDT
        WRITE OUTPUT TAPE IO, 1010
1010    FORMAT (1H1,10X,22HPLOT REQUEST COMPLETED)
        PRINT 1053
1053    FORMAT (39H  PUSH START TO CONTINUE WITH NEXT RUN)
        PAUSE
        GO TO 10
        END
                                                                 EOF E SY
```

5b. Printed Plot Routine – Source Statements (Continued)

```
$LOG     002292
*        PAUSE
*        SYMBOL TABLE
*        LIST8
*        CARDS COLUMN
*        FORMAP
*        LABEL
CCCARD
         SUBROUTINE CCARD(ICD,IRNG)
         DIMENSION ICD(43),IBAT(15),IFIELD(30),IRNG(19)
         IO = 3                                                       A3
         IRNG(19) =0
         K1= 0
         DO 23 I= 2,13
         IF (ICD(I)) 24,23,24
24       K1= K1 +1
         IBAT(K1)= I-2
23       CONTINUE
C....    LOCATE FIELDS REQUESTED
         K2 =0
         IF(ICD(42))3,4,3
3        II1=21
         GO TO 5
4        II1=20
5        DO 33 I=II1,36
         IF(ICD(I)) 31,32,31
31       K2 = K2 +1
         IFIELD(K2) = ICD(I)
33       CONTINUE
32       IF(K2)    40,38,40
C....    ALL FIELDS ARE TO BE SELECTED
38       IF(ICD(1)) 35,34,35
C....    OUTCARD
34       J=15
         GO TO 36
C....    ENDCARD
35       J = 26
36       DO 37 K2 =1, J
         IFIELD(K2)= K2
37       CONTINUE
40       ICYC = ICD(14)
         IF(ICD(1)) 42,41,42
41       WRITE OUTPUT TAPE IO, 917
         GO TO 43
42       WRITE OUTPUT TAPE IO, 916
43       WRITE OUTPUT TAPE IO, 903,(IBAT(I),I=1,K1)
         WRITE OUTPUT TAPE IO, 904, ICYC
         WRITE OUTPUT TAPE IO, 905, ICD(15),ICD(16)
         IF(ICD(17)) 45,44,45
44       WRITE OUTPUT TAPE IO, 906
         GO TO 46
45       IF(ICD(17)-2) 511,510,511
510      WRITE OUTPUT TAPE IO,949
         GO TO 46
511      WRITE OUTPUT TAPE IO,907
46       IF (ICD(1))  54,47,54
47       IF (ICD(18) - 1 ) 49 ,48, 49
48       WRITE OUTPUT TAPE IO, 908
         GO TO 54
49       IF (ICD(18))   51,50,51
50       WRITE OUTPUT TAPE IO, 911
         GO TO 54
51       IF (ICD(18) - 2 )  53,52,53
```

5b. Printed Plot Routine – Source Statements (Continued)

```
52      WRITE OUTPUT TAPE IO, 909, ICD(19)
        GO TO 54
53      WRITE OUTPUT TAPF IO,910, ICD(19)
54      IF(ICD(20))56,55,56
55      WRITE OUTPUT TAPE IO, 918
        GO TO 57
56      WRITE OUTPUT TAPE IO, 912,(IFIFLD(I),I=1,K2)
57      IF(ICD(41)) 58,59,58
58      WRITE OUTPUT TAPE IO, 913
59      IF(ICD(42)) 60,61,60
60      WRITE OUTPUT TAPE IO,914,ICD(20)
61      WRITE OUTPUT TAPE IO,915,ICD(43)
        IF (ICD(40)) 63,62,63
63      I =1                                                          1
        ICR = 2                                                       1
64      WRITE OUTPUT TAPE IO, 955,ICR,IRNG(I),IRNG(I+1),IRNG(I+2)     1
        I = I+3                                                       1
        ICR = ICR +1                                                  1
        IF(IRNG(I)) 64,62,64
62      RETURN                                                        1
903     FORMAT  (1H05X,23HBATTERYS REQUESTED ARE 12I4)
904     FORMAT (6X,21HSTARTING CYCLE NO IS I4)
905     FORMAT (6X,13HINCREMENT IS I3,22H NUMBER OF INCREMENTS I4)
906     FORMAT (6X,12HCHARGE PHASE)
907     FORMAT (6X,15HDISCHARGE PHASE)
908     FORMAT (6X,36HLEAST SQUARE FIT ON REQUESTED FIELDS)
909     FORMAT  (6X,6HFIRST I2,27H PERCENT OF CYCLE REQUESTED)
910     FORMAT (6X,5HLAST I2,27H PERCENT OF CYCLE REQUESTED)
911     FORMAT (6X,31H100 PER CENT OF CYCLE REQUESTED)
912     FORMAT (6X,16HFIELDS REQUESTED22I4)
913     FORMAT (6X,22HINVALID DATA REQUESTED)
914     FORMAT(6X,35HCELL CYCLE REQUEST ON OUTCARD FIELDI5)
915     FORMAT (6X,11HEXPIRED DAYI4)
916     FORMAT (6X,22HENDCARD DATA REQUESTED)
917     FORMAT (6X,22HOUTCARD DATA REQUESTED)
918     FORMAT (6X,20HALL FIELDS REQUESTED)
949     FORMAT (6X,26HCHARGE AND DISCHARGE PHASE)
955     FORMAT (1H06X,11HCYCLE RANGEI2,16H  STARTING CYCLEI5,11H  INCREMEN   1
       1TI5,22H  NUMBER OF INCREMENTSI5)                                      1
        END
                                                                  EOF E SY
```

DISTRIBUTION LIST FOR FINAL REPORT

CONTRACT NAS3-5908

Addressee	No. of Copies
NASA-Lewis Research Center	
21000 Brookpark Road	
Cleveland, Ohio 44135	
Attn: C. C. Conger (M. S. 54-1)	1
J. R. Eski (M. S. 54-2)	1
H. R. Hunczak (M. S. 54-3)	9
E. W. Otto (M. S. 54-1)	1
Library (M. S. 60-3)	2
Reports Control Office (M. S. 5-5)	1
Technology Utilization Office (M. S. 3-19)	1
National Aeronautics and Space Administration	
Washington, D. C. 20546	
Attn: J. Lazar, RNT	1
J. Mullin, RNT	1
NASA Scientific and Technical Information Facility	
Box 33	
College Park, Maryland 20740	
Attn: NASA Representative RQT-2448	6
NASA Marshall Space Flight Center	
Huntsville, Alabama 35812	
Attn: K. Erich, R-RP-DIR	1
G. Heller, R-RP-DIR	1
Dr. E. Stuhlinger, R-RP-DIR	1
NASA Langley Research Center	
Hampton, Virginia 23365	
Attn: R. Hook, MS 316	1
M. Ellis, MS 160	1
NASA Goddard Space Flight Center	
Greenbelt, Maryland 20771	
Attn: Library	1
D. Suddeth, Code 460	1
Art White, Code 734	1

DISTRIBUTION (Cont'd)

Addressee	No. of Copies
Wright-Patterson Air Force Base	
Dayton, Ohio 45433	
Attn: AFAPL (API) G. Sherman	1
AFAPL (APIE) P. Lindquist	1
AFAPL (APIE) Capt. P. E. Peko	1
AFWL	
Kirtland AFB, New Mexico 87117	
Attn: WLPC/Capt. C. F. Ellis	1
National Security Agency	
Fort George G. Mead, Maryland 20755	
Attn: Lt. John D. Ryder, Code 188-7250	1
Electro-Optical Systems, Inc.	
300 North Halstead Street	
Pasadena, California 91107	
Attn: Dr. R. Buhler	1
Dr. G. Cann	1
R. C. Speiser	1
Giannini Scientific Corporation	
3839 South Main Street	
Santa Ana, California 92705	
Attn: A. Ducati	1
General Electric Company	
Electric Space Propulsion Projects	
Evendale, Ohio 45202	
Attn: Dr. M. L. Bromberg	1
General Electric Missiles and Space Division	
Valley Forge Technical Center	
P.O. Box 8555	
Philadelphia, Pennsylvania 19101	
Attn: Dr. Per Gloersen	1
Marquardt Corporation	
16555 Saticoy Street	
Van Nuys, California 91406	
Attn: R. Page	1
A. Thomas	1

DISTRIBUTION (Cont'd)

Addressee	No. of Copies
Jet Propulsion Laboratory California Institute of Technology 4800 Oak Grove Drive Pasadena, California 91103 Attn: John W. Stearns, Jr. MS 125-224	1
Rocketdyne 6633 Canoga Avenue Canoga Park, California 91303	1
Aerospace Corporation P.O. Box 95085 Los Angeles, California 90045 Attn: Library Technical Documents Group	1
Ling-Temco-Vought, Inc. Advanced Systems, Astro. Division Box 6267 Dallas, Texas 75221 Attn: F. T. Esenwein	1
Westinghouse Astronuclear Laboratories Electrical Propulsion Laboratory Pittsburgh, Pennsylvania 15234 Attn: H. W. Szymanowski, Manager	1
Philco Corporation Newport Beach, California 92660 Attn: R. Spongberg ATC	1
Garrett Corporation (AiResearch) 9851 Sepulveda Boulevard Los Angeles, California 90045 Attn: R. Fischer	1
Grumman Aircraft Eng. Co. Thermodynamics and Propulsion Section Plant No. 5 Bethpage, N. Y. 11714 Attn: I. Tobias	1

DISTRIBUTION (Concl'd)

Addressee	No. of Copies
Space Dynamics Corporation Plasmajet Systems Division 2215 Florence Avenue Cincinnati, Ohio 45206 Attn: Dr. M. Ghai	1
Douglas Aircraft Company Missile and Space Systems Santa Monica, California 90406 Attn: A. Pisciotta, Jr.	1
Massachusetts Institute of Technology Department of Mechnical Engineering Cambridge, Massachusetts 02138 Attn: Dr. J. Fay	1
Case Institute of Technology 10900 Euclid Avenue Cleveland, Ohio 44106 Attn: Dr. Eli Reshotko	1
TRW Inc. TRW Systems Group One Space Park Redondo Beach, California 90278 Attention: W. F. Krieve	1
Colorado State University Fort Collins, Colorado 80521 Attn: Prof. W. R. Mickelsen	2
Research Library - Wilmington (+1 reproducible)	3
Research Library - Lowell (+1 reproducible)	1
Reports Distribution Center Attn: Dr. R. R. John	50