MULTI-GROUP ANALYSIS OF NUCLEAR REACTORS IN THREE SPACE DIMENSIONS

R59FPD936

F.B. BANAN
O.S. CLENDINNING
D.B. VOLLENWEIDER
D.H. WATSON
C.H. WARLICK

1960

FLIGHT PROPULSION LABORATORY DEPARTMENT
KNOLLS ATOMIC POWER LABORATORY
GENERAL ELECTRIC

LEGAL NOTICE

Printed in USA. Price $3.00. Available from the Office of Technical Services, Department of Commerce Washington 25, D. C.

UNCLASSIFIED

R59FPD936
Physics and Math.
TID-4500 (15th Ed.)

PROGRAM

TRIXY

Multi-Group Analysis of Nuclear Reactors in Three Space Dimensions

(For the IBM Type 704 Computer)

by

F. B. Banan
O. S. Clendinning
D. B. Vollenweider
C. H. Warlick
D. H. Watson

Authorized Classifier

4/20/60
Date

The General Electric Company
Flight Propulsion Laboratory Department
Evendale, Ohio

Knolls Atomic Power Laboratory
Schenectady, N. Y.

UNCLASSIFIED

FOR USE OF G-E EMPLOYEES ONLY

GENERAL ELECTRIC

TECHNICAL INFORMATION SERIES

Title Page

AUTHOR	SUBJECT CLASSIFICATION	NO.
(see below)	Nuclear Reactor Analysis	R59FPD936
		DATE 4-20-60

TITLE

PROGRAM TRIXY--Multi-Group Analysis of Nuclear Reactors in Three Space Dimensions

Authors: F. B. Banan, O. S. Clendinning, D. B. Vollenweider, C. H. Warlick, D. H. Watson

Abstract:

TRIXY is a computer program that provides a multigroup analysis of nuclear reactors in three space dimensions. The program can be run on an IBM 704 computer that has at least 8192 words of memory and six tape units. The input data consist of cross-sections, geometrical configurations describing the location of material within the reactor, control rod positions, mesh spacing for the three axes, and boundary conditions. For each energy group, g, the program computes ϕ, the neutron flux, by a numerical approximation to the age-diffusion equation. When ϕ has been calculated for all groups, the fission source, S, is computed. The criticality and source error are calculated and printed. If the fission source has not converged to the degree of accuracy requested, S is recomputed. The results of the program are edited so that values of flux and power density for any section of the reactor can be printed.

G.E. CLASS	REPRODUCIBLE COPY FILED AT	NO. PAGES
I	FPD Technical Information Service - Evendale, Ohio	167
GOV. CLASS. Unclassified	Office of Tech. Services - Washington, D. C.	

By cutting out this rectangle and folding on the center line, the above information can be fitted into a standard card file.

For list of contents — drawings, photos, etc. and for distribution see next page (GT 2063-B1).

INFORMATION PREPARED FOR Knolls Atomic Power Laboratory

TESTS MADE BY

COUNTERSIGNED J C Richter OPERATION Evendale Computations

DIVISION Flight Propulsion Lab. LOCATION Evendale, Ohio

GT 2063-A1 (10-52)

ACKNOWLEDGEMENTS

The TRIXY program was written by personnel at the General Electric Company, Evendale Computations Operation, Flight Propulsion Division, Evendale, Ohio, and the Knolls Atomic Power Laboratory, Schenectady, New York. The specifications for the program were prepared by Dr. Paul Greebler, Dr. Richard Stark, and Dr. E. L. Wachspress of KAPL. Dr. Greebler and Dr. David Epstein worked with the Evendale personnel in co-ordinating the complete program.

The TRIXY project at Evendale was directed by Mr. Charles Warlick. The various sections of the program and corresponding parts of this report were written by the following people: Input Data Preparation and Master Tape Generation by Mr. Doneley Watson, Group Source by Mr. Fred Banan, Group Sweep by Mrs. Deborah Vollenweider, Fission Source by Mr. Banan and Mr. Watson, all of Evendale Computations. The Edit Routine was prepared by Mrs. Ouida Clendinning of KAPL. Sections of the KAPL 1549 report, written by Dr. Greebler, are included in this report.

The preliminary work on TRIXY was directed by Mr. Thomas Glahn. Mrs. Betty Storrs and Mr. W. H. Houssell programmed part of the Master Tape Generation. Mr. Warlick did the initial work on the Group Sweep. Miss Marcia Mooney programmed a large part of the Group Source aided by Miss Shyrl Emhoff, who also contributed to the preparation of this report.

The authors would like to thank Miss Lois Ross for typing the report.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

TRIXY is a computer program that provides a multigroup analysis of nuclear reactors in three space dimensions. The program can be run on an IBM 704 computer that has at least 8192 words of memory and six tape units. The input data consist of cross-sections, geometrical configurations describing the location of material within the reactor, control rod positions, mesh spacing for the three axes, and boundary conditions. For each energy group, g, the program computes φ, the neutron flux, by a numerical approximation to the age-diffusion equation. When φ has been calculated for all groups, the fission source, S, is computed. The criticality and source error are calculated and printed. If the fission source has not converged to the degree of accuracy requested, S is recomputed. The results of the program are edited so that values of flux and power density for any section of the reactor can be printed.

GENERAL FEATURES OF TRIXY

1. Three-dimensional Cartesian coordinates are used.
2. Variable mesh spacings may be chosen on all three axes.
3. There may be up to 72 mesh planes perpendicular to any of the three axes.

4. Up to six energy groups may be used.

5. Up to 975 different material compositions are permitted within the mesh (with 32,768 word memory).

6. The mesh is bounded by a rectangular parallelepiped. Boundary conditions are specified independently for each of the six faces, either zero flux, or zero or non-zero current.

7. The product of the number of mesh points and the number of energy groups is limited to 150,000.

8. The contribution to the neutron source due to slowing down of neutrons is restricted to degradation from the next higher energy group.

9. Fluxes in each energy group are calculated by an iterative technique (accelerated Liebmann).

10. The edit routine yields fluxes and power densities for mesh points in selected columns (a column is a mesh line parallel to the $X^{(3)}$ axis), the power integral and average power density for selected columns, flux integrals and average flux values for each composition region of the mesh, and the number of fissions per source neutron (proportional to criticality).

11. A power guess for the first run (a run is a power iteration) may be specified in a coarse mesh, or the results of an earlier calculation may

be used if the number of mesh planes in each direction is identical for the two problems.

12. Adjoint fluxes may be computed, provided the ratio F^g/F^G is independent of position throughout that portion of the mesh that contains fissionable material. F^g is the fission cross section for energy group g, and G is the lowest energy (thermal) group.

LIMITATIONS DETERMINED BY SIZE OF MEMORY

Although TRIXY was written primarily for use on a 32,768 word computer, it may also be run on a 8192 or 16,384 word computer with the following input restrictions:

Words of memory	8192	16,384	32,768
Product of the number of mesh planes perpendicular to the $X^{(2)}$ and $X^{(3)}$ axes.	1000	2000	4000
Approximate number of material compositions permitted within the file of cross sections records.	500	1500	3000
Number of material compositions permitted within the mesh.	200	475	975

If there is a change in mesh size or material composition along any column at only every fifth point, then the running times for a given problem will be approximately the same for any size machine.

II. DESCRIPTION OF THE MESH

GEOMETRIC PROPERTIES

Consider the Cartesian coordinate axes $X^{(1)}$, $X^{(2)}$, and $X^{(3)}$. Mesh planes are perpendicular to these axes and have coordinates $X_i^{(1)}$, $X_j^{(2)}$, $X_k^{(3)}$. The integers i, j, and k identify the planes and have the maximum values I, J, K = 72.

Mesh points, identified as (i, j, k), are located at the intersections of the mesh planes of corresponding indexes. Point (1, 1, 1) is located at the origin of the coordinate system.

The spacing (cm) between adjacent mesh planes need not be uniform and is defined by:

$$\Delta_i^{(1)} = X_{i+1}^{(1)} - X_i^{(1)} \quad \text{for } 1 < i < I - 1; \quad \Delta_0^{(1)} = \Delta_I^{(1)} = 0, \qquad \text{(II 1)}$$

$$\Delta_j^{(2)} = X_{j+1}^{(2)} - X_j^{(2)} \quad \text{for } 1 < j < J-1; \quad \Delta_0^{(2)} = \Delta_J^{(2)} = 0, \qquad \text{(II 2)}$$

$$\Delta_k^{(3)} = X_{k+1}^{(3)} - X_k^{(3)} \quad \text{for } 1 < k < K-1; \quad \Delta_0^{(3)} = \Delta_K^{(3)} = 0. \qquad \text{(II 3)}$$

The six mesh planes, i, i+1, j, j+1, k, and k+1 bound a rectangular parallelepiped of uniform composition. This parallelepiped is identified as "composition box" (i, j, k) and its volume is:

$$V_{i,j,k} = \Delta_i^{(1)} \quad \Delta_j^{(2)} \quad \Delta_k^{(3)} . \tag{II 4}$$

A composition box is divided into eight boxes of equal volume, identified as octants. The octant nearest mesh point (i, j, k) is identified as octant (i, j, k), and its volume is:

$$v_{i,j,k} = \Delta_i^{(1)} \quad \Delta_j^{(2)} \quad \Delta_k^{(3)}/8. \tag{II 5}$$

Each mesh point that does not lie on a boundary plane is surrounded by eight octants which may be of different composition. Boundary points touch one, two, or four octants, respectively, for points that lie on corners, edges, or faces of the mesh. The total region of all octants that touch a given mesh point (i, j, k) is defined as "flux box" (i, j, k), and this volume is:

$$V'_{i,j,k} = \sum_{\ell, m, n=0}^{1} V_{i-\ell, j-m, k-n}. \tag{II 6}$$

Some of the octants in flux box (i, j, k) may contain fissionable material. The beta function is defined as:

$$\begin{aligned} \beta_{i,j,k} &= 1 \text{ if } v_{i,j,k} \text{ contains fuel,} \\ \beta_{i,j,k} &= 0 \text{ if } v_{i,j,k} \text{ does not contain fuel.} \end{aligned} \tag{II 7}$$

Hence, the fissionable volume in a flux box (i, j, k) is:

$$V''_{i,j,k} = \sum_{\ell, m, n=0}^{1} v_{i-\ell, j-m, k-n} \beta_{i-\ell, j-m, k-n}. \quad \text{(II 8)}$$

and the total core volume is:

$$V_F = \sum_{i,j,k=1}^{I, J, K} V''_{i,j,k}. \quad \text{(II 9)}$$

CROSS SECTIONS

Each material composition in the mesh is identified by an integer l_n, where $l_1=1$, $l_2=2$, $l_3=3$, etc. To distinguish between compositions that contain fissionable material and those that do not, it is required that $1 \leq l_n \leq 999$ for compositions that contain fuel and $1001 \leq l_n \leq 1999$ for non-fissionable compositions. G sets of values D^g, A^g, M^g, and F^g are associated with each l_n in the fissionable range, where the superscript g denotes the energy group $1 \leq g \leq G$. D^g is the diffusion coefficient (cm), M^g is the slowing down cross section (cm^{-1}), and F^g is the fission cross section (cm^{-1}). A^g is the removal cross section (cm^{-1}) which is the sum of the absorption and slowing down cross sections. G sets of values D^g, A^g, and M^g are associated with each l_n in the non-fissionable range.

A library of diffusion coefficients and cross sections are maintained on tape. Each file in this library contains four separate records:

(1) The identification number of the file, the number of energy groups G, χ^g for each group (χ is the fraction of the fission spectrum that occurs in group g), and the number of words in each record in the file,

(2) Diffusion coefficients and cross sections for compositions containing fuel in the order of increasing l_n,

(3) Diffusion coefficients and cross sections for non-fissionable compositions in the order of increasing l_n, and

(4) Ratios of outgoing current to flux, Z^g, for various reflector compositions, each identified by an index l_n, where $2001 \leq l_n \leq 2999$.

G values of Z^g are listed for each l_n, and compositions are listed in the order of increasing l_n. The Z^g values are used to specify non-zero current boundary conditions.

The diffusion coefficients and cross sections of composition box (i, j, k) are identified by a composition index $l_{i,j,k}$, where $l_{i,j,k}$ is one of the integers in records two and three. Current boundary conditions for the boundary planes are identified by the composition indexes $l^{(1)}_{Z_0}$, $l^{(1)}_{Z_I}$, $l^{(2)}_{Z_0}$, $l^{(2)}_{Z_J}$ $l^{(3)}_{Z_0}$, and $l^{(3)}_{Z_K}$, respectively for the six planes, where $l^{(p)}_{Z_p}$, is identified with an l_n in record four. Composition indexes $l_{i,j,k}$ for each composition box of the mesh, combined with a file of diffusion coefficients and cross sections, provide a complete set of cross section data for the mesh. Values $D^g_{i,j,k}$,

$A^{g}_{i,j,k'}$ $M^{g}_{i,j,k'}$ and $F^{g}_{i,j,k}$ for all groups associated with index $l_{i,j,k}$ describe the nuclear properties of composition box (i, j, k).

SPECIFICATION OF CONFIGURATIONS OF MATERIALS

A configuration is an area of uniform material composition in a plane normal to the $X^{(3)}$ axis (a k-plane). List A and list B are used to project these plane figures into three-dimensional volumes. In constructing each k-plane, list A is examined, and the appropriate configurations are placed on the plane; then list B (control-rod list) is examined, and the control rod configurations are written over the list A configurations.

Each configuration is a record in a file of configurations preceding the files of Cross Section Data. The maximum number of configurations is limited only by the physical length of the tape.

III. INPUT DATA

CLASSIFICATION OF INPUT

There are two main divisions of input for a problem. The first set of input consists of a tape containing a file of configurations and several files of Cross Section data. It is expected that one tape can be used for a number of problems.

The second set of input consists of the following:

A. Input control data.

1. Miscellaneous data, such as I, J, K, G, $\delta\phi^{g}$ for each group, δg , V_F, g'_F ($F^g = 0$ everywhere for $g < g'$), six mesh planes (i^F_{min}, i^F_{max}, j^F_{min}, j^F_{max}, k^F_{min}, and k^F_{max}) that bound all fissionable material in the mesh, and $\ell^{(p)}_{Z_p}$, for each boundary plane of specified current-to-flux ratio. The machine calculates I, J, K, and V_F. These calculations are listed as input for the purpose of checking the consistency of the input data.
2. List A
3. List B

4. Information for calculation of mesh spacings. Input data list coordinates along each axis and the number of equally spaced intervals between adjacent coordinates

5. Input control data for the EDIT routine. Output data that may be requested are:

   a. The power density for each mesh point, either for the entire mesh or for selected columns,

   b. The average power density and the power integral for a column, either for all the columns of the mesh or for selected columns,

   c. The flux at each mesh point for selected energy groups, either for the entire mesh or for selected columns,

   d. The flux integral and average flux for each of the different composition regions of the mesh for each group, and

   e. The number of fissions per source neutron (f), and the contribution of each energy group to f.

B. A power guess.

If a power guess and flux guess are available from a previous problem, these may be used. If no guess is available, a rough power guess must be supplied in a coarse mesh.

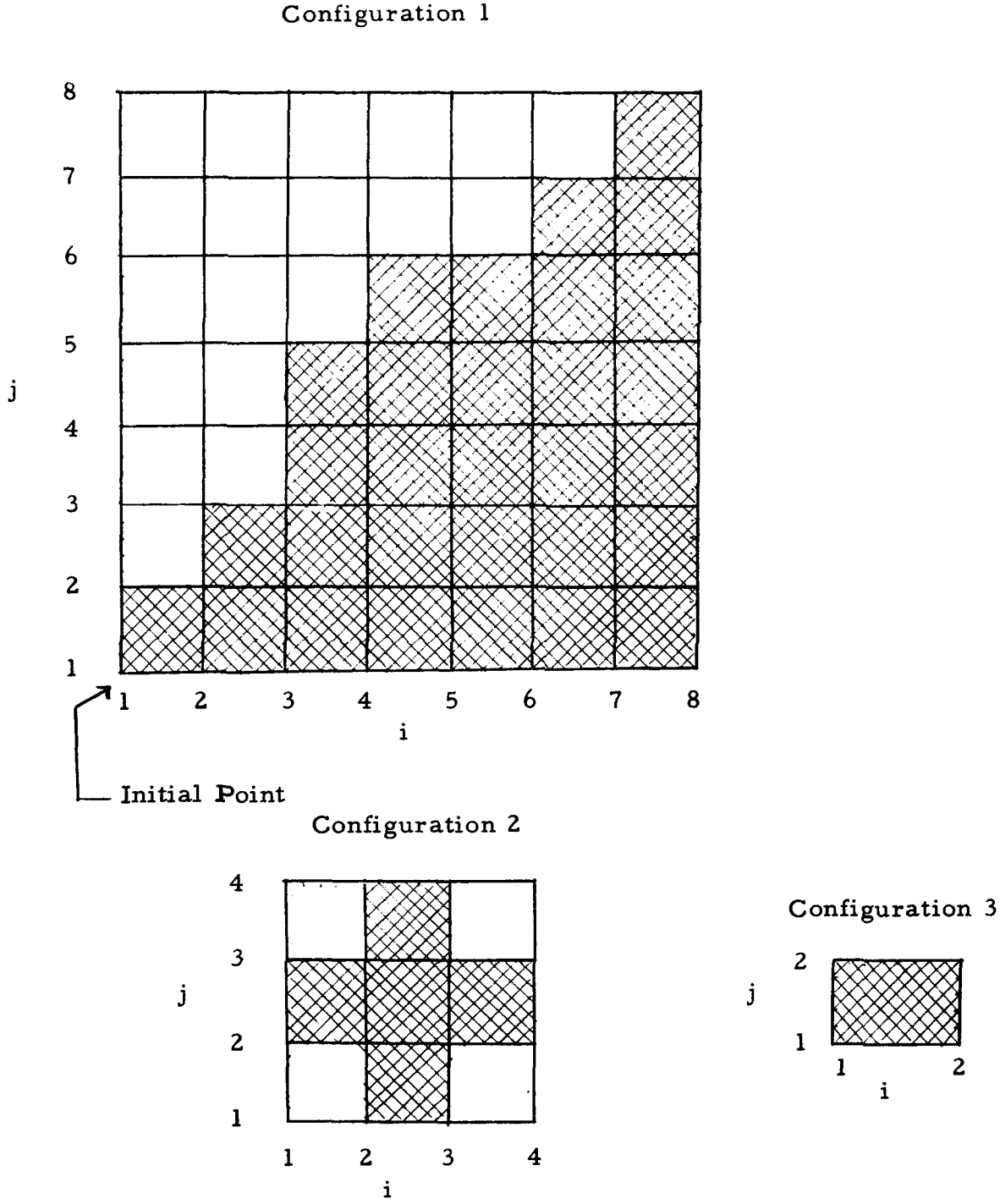
MESH SPACINGS AND COORDINATES

A series of numbers, $n_a^{(1)}$, $X_a^{(1)}$, $n_b^{(1)}$, $X_b^{(1)}$, ..., $X_I^{(1)}$, 0, $n_a^{(2)}$, $X_a^{(2)}$, $n_b^{(2)}$, $X_b^{(2)}$, ..., $X_J^{(2)}$, 0, $n_a^{(3)}$, $X_a^{(3)}$, $n_b^{(3)}$, $X_b^{(3)}$, ..., $X_K^{(3)}$, 0, is used to specify mesh increments. Along the $X^{(1)}$ axis, the interval between coordinate $X_a^{(1)}$ and the origin is divided by mesh planes into $n_a^{(1)}$ equal intervals. The interval between $X_b^{(1)}$ and $X_a^{(1)}$ is divided by mesh planes into $n_b^{(1)}$ equal intervals, until the first zero is reached. The zero indicates the end of the specifications of mesh spacings along the $X^{(1)}$ axis. A similar interpretation applies to the specification of the mesh spacings along $X^{(2)}$ and $X^{(3)}$ axes.

CONFIGURATIONS WITH LIST A AND LIST B

A hypothetical reactor is constructed using the configurations in Figure 1 to demonstrate the method of specifying configurations and the projection of these plane figures into three-dimensional volume.

Each of the configurations is enclosed by a rectangle. To describe configuration 1, the number of intervals along the j axis is seven. Reading up the first column, there is one block within the configuration. There are six



Two dimensional representations of hypothetical configurations.

FIGURE 1

blocks outside (-). Therefore, the first column is described by 1, -6, 100, where 100 denotes the end of a column. This process is repeated for the remaining columns, and 1000 is written at the end of the configuration.

Configuration 1 ($n \Delta_j = 7$) is completely described by the following series of numbers:

1, 7, 1, -6, 100, 2, -5, 100, 4, -3, 100, 5, -2, 100, 5, -2, 100, 6, -1, 100, 7, 1000.

Similarly for configuration 2:

2, 3, -1, 1, -1, 100, 3, 100, -1, 1, -1, 1000,

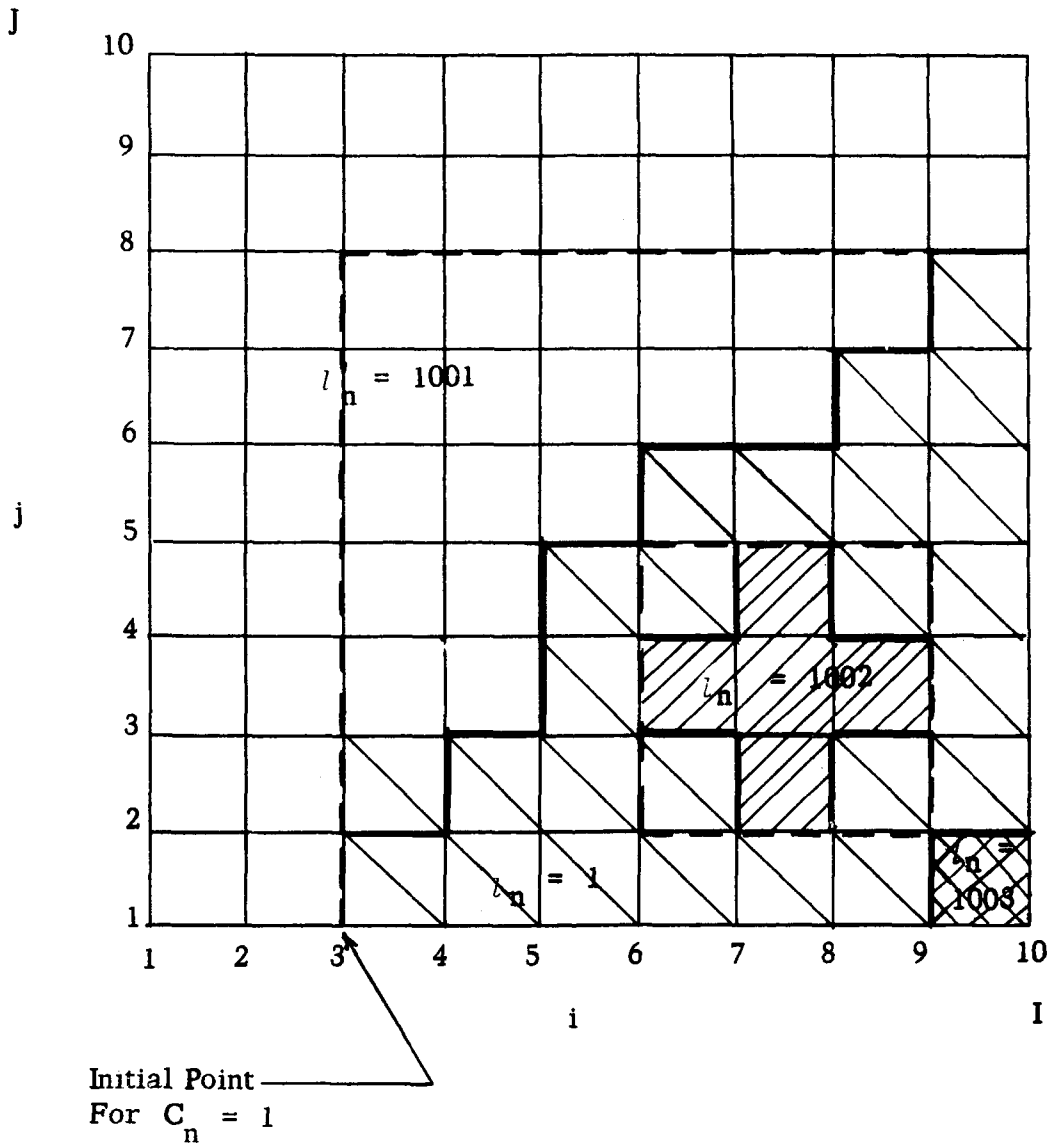
and configuration 3:

3, 1, 1, 1000.

In this hypothetical reactor (see Figure 2) let configuration 1 be the core, composition index $l_n = 1$, and extend from plane k=3 to k=5 where the configuration changes to composition index 2. This composition index extends to k=7. The core is surrounded by composition $l_n = 1001$ on all sides except i=I and j=1.

A control rod, configuration 2, composition index 1002, extends from k=4 to k=K. The volume under this control rod is composition $l_n = 1001$.

Another control rod, configuration 3, composition index 1003, extends from k=3 to k=K.



Plane k = 4 of a Hypothetical Reactor

Figure 2

List A	k	C_n	l_n	i	j
	1	0[1]	1001	1	1
	3	1	1	3	1
	5	1	2	3	1
	7	0	1001	1	1

List B	k	C_n	l_n	i	j
	3	3	1003	9	1
	3	2	1001	6	2
	4	2	1002	6	2

ROUGH POWER GUESS

Coordinates (cm) of a coarse mesh are listed as a series of numbers $X_1^{(1)}, X_2^{(1)}, \ldots, X_I^{(1)}, 0, X_1^{(2)}, \ldots, X_J^{(2)}, 0, X_1^{(3)}, \ldots, X_K^{(3)}, 0$. The values entered for $X_I^{(1)}, X_J^{(2)}, X_K^{(3)}$ must correspond to those entered for these quantities in the mesh spacings and coordinates. $X_1^{(1)}$ cannot be greater than the coordinate corresponding to i_{min}^F in the fine mesh. These coordinates are followed by values for the relative power densities ($p_{i,j,k}^{(0)}$) for each of the boxes enclosed by the coordinates of the coarse mesh. The values of the

---

[1] A configuration index $C_n = 0$ indicates that the entire plane is of uniform composition.

relative power density are entered in the following order:

$p^{(0)}_{0,0,0}$, $p^{(0)}_{0,0,1}$, . . , $p^{(0)}_{0,0,K-1}$, $p^{(0)}_{0,1,0}$, . . ., $p^{(0)}_{0,1,K-1}$, . . ., $p^{(0)}_{0,J-1,K-1}$,

$p^{(0)}_{1,0,0}$, . . ., $p^{(0)}_{I-1,J-1,K-1}$.

If $N^{(1)}$, $N^{(2)}$, and $N^{(3)}$ are the number of coordinates for the three axes then there are $N^{(1)}\,N^{(2)}\,N^{(3)}$ values of relative power density listed.

IV. GENERAL PROCEDURE

TRIXY can be divided into six sections

1. Master Tape Generation
2. Rough Source Guess
3. Group Source
4. Group Sweep
5. Fission Source
6. Edit Code

The relation between the various sections is illustrated by the overall TRIXY flow chart (see Figure 3).

OVERALL FLOW CHART

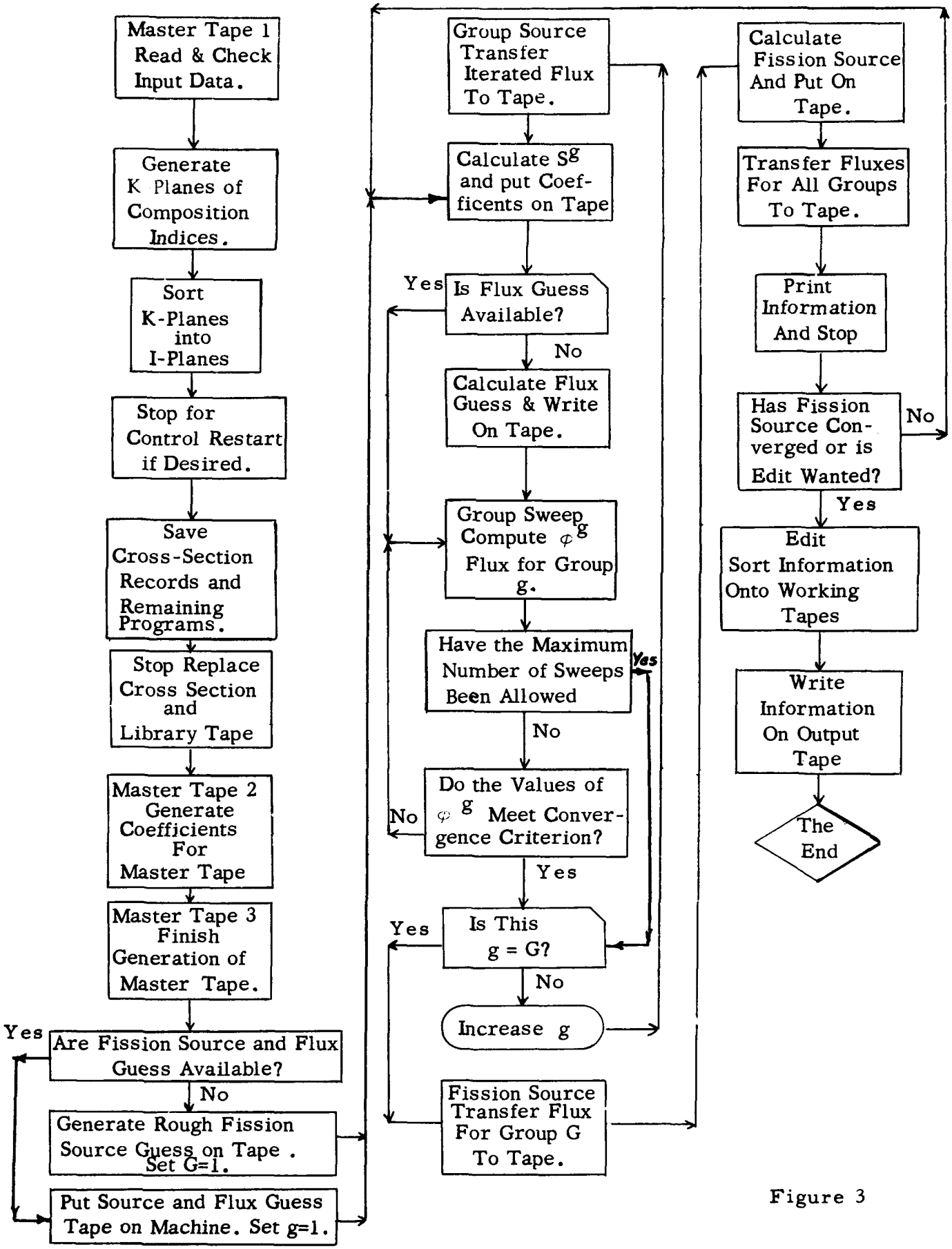


Figure 3

V. GENERATION OF THE MASTER TAPE AND THE ROUGH FISSION SOURCE GUESS

This section of TRIXY consists of three separate programs.

Master tape 1 processes and checks the input, generates the I planes of composition indices, and generates a working library tape.

Master tape 2 generates the master tape for all groups up to the Fission Source program.

Master tape 3 finishes the master tape and generates the Rough Fission Source Guess, if needed.

EQUATIONS - MASTER TAPE GENERATION

Backward Coefficients - Current Boundaries

$$C^{(1)g}_{0,j,k} = Z^{(1)g}_{0} \sum_{m,n=0}^{1} \Delta^{(2)}_{j-m} \Delta^{(3)}_{k-n} / 4 \qquad \text{(V 1)}$$

$$C^{(2)g}_{i,0,k} = Z^{(2)g}_{0} \sum_{\ell,n=0}^{1} \Delta^{(1)}_{i-\ell} \Delta^{(3)}_{k-n} / 4 \qquad \text{(V 2)}$$

$$C^{(3)g}_{i,j,0} = Z^{(3)g}_{0} \sum_{\ell,m=0}^{1} \Delta^{(1)}_{i-\ell} \Delta^{(2)}_{j-m} / 4 \qquad \text{(V 3)}$$

Forward Coefficients - Current Boundaries

$$C^{(1)g}_{I,j,k} = Z^{(1)g}_{I} \sum_{m,n=0}^{1} \Delta^{(2)}_{j-m} \Delta^{(3)}_{k-n} / 4 \qquad \text{(V 4)}$$

$$C^{(2)g}_{i,J,k} = Z^{(2)g}_{J} \sum_{\ell,\, n=0}^{1} \Delta^{(1)}_{i-\ell} \quad \Delta^{(3)}_{k-n} \Big/ 4 \qquad \text{(V 5)}$$

$$C^{(3)g}_{i,j,K} = Z^{(3)g}_{K} \sum_{\ell,\, m=0}^{1} \Delta^{(1)}_{i-\ell} \quad \Delta^{(2)}_{j-m} \Big/ 4 \qquad \text{(V 6)}$$

Backward Coefficients - Flux Boundaries. Substitute i, j, or k=1 in the appropriate formula.

$$C^{(1)g}_{i,j,k} = \sum_{m,\, n=0}^{1} D^{g}_{i,\, j-m,\, k-n} \quad \Delta^{(2)}_{j-m} \quad \Delta^{(3)}_{k-n} \Big/ 4 \quad \Delta^{(1)}_{i} \qquad \text{(V 7)}$$

$$C^{(2)g}_{i,j,k} = \sum_{\ell,\, n=0}^{1} D^{g}_{i-\ell,\, j,\, k-n} \quad \Delta^{(1)}_{i-\ell} \quad \Delta^{(3)}_{k-n} \Big/ 4 \quad \Delta^{(2)}_{j} \qquad \text{(V 8)}$$

$$C^{(3)g}_{i,j,k} = \sum_{\ell,\, m=0}^{1} D^{g}_{i-\ell,\, j-m,\, k} \quad \Delta^{(1)}_{i-\ell} \quad \Delta^{(2)}_{j-m} \Big/ 4 \quad \Delta^{(3)}_{k} \qquad \text{(V 9)}$$

Additional Coefficients

$$v_{i,j,k} = \Delta^{(1)}_{i} \quad \Delta^{(2)}_{j} \quad \Delta^{(3)}_{k} \Big/ 8. \qquad \text{(V 10)}$$

$$r^{(1)g}_{i,j,k} = \sum_{m=0}^{1} D^{g}_{i,\, j-m,\, k} \quad \Delta^{(2)}_{j-m} \quad \Delta^{(3)}_{k} \Big/ 4 \quad \Delta^{(1)}_{i} \qquad \text{(V 11)}$$

$$r^{(2)g}_{i,j,k} = \sum_{\ell=0}^{1} D^{g}_{i-\ell,\, j,\, k} \quad \Delta^{(1)}_{i-\ell} \quad \Delta^{(3)}_{k} \Big/ 4 \quad \Delta^{(2)}_{j} \qquad \text{(V 12)}$$

$$E^{g}_{i,j,k} = \sum_{\ell,m,n=0}^{1} A^{g}_{i-\ell,j-m,k-n}\, v_{i-\ell,j-m,k-n} \qquad (V\ 13)$$

$$H^{g}_{i,j,k} = \sum_{\ell,m,n=0}^{1} M^{g}_{i-\ell,j-m,k-n}\, v_{i-\ell,j-m,k-n} \qquad (V\ 14)$$

$$F'^{g}_{i,j,k} = \sum_{\ell,m,n=0}^{1} F^{g}_{i-\ell,j-m,k-n}\, v_{i-\ell,k-m,k-n} \qquad (V\ 15)$$

$$V''_{i,j,k} = \sum_{\ell,m,n=0}^{1} v_{i-\ell,j-m,k-n}\ \beta_{i-\ell,j-m,k-n} \qquad (V\ 16)$$

$$V_F = \sum_{i,j,k=1}^{I,\ J,\ K} V''_{i,j,k} \qquad (V\ 17)$$

EQUATIONS - ROUGH FISSION SOURCE GUESS

Rough Power Guess

$$P^{(0)}_{i,j,k}$$

Unnormalized Fission Source

$$\text{ß}^{(0)+}_{i,j,k} = P^{(0)}_{i,j,k}\, v_{i,j,k}. \qquad (V\ 18)$$

Normalized Fission Source

$$\text{ß}^{(0)}_{i,j,k} = (\ \text{ß}^{(0)+}_{i,j,k}\ ) \left(V_F \Big/ \sum_{i,j,k=1}^{I,\ J,\ K} \text{ß}^{(0)+}_{i,j,k} \right). \qquad (V\ 19)$$

MASTER TAPE 1 (see Figure 4)

Input is read into the machine. The program is removed from the machine and the input corrected if any of the following errors occur:

1. The Mesh Spacings and Coordinates do not agree with I, J, or K.
2. The value of G given in the input differs from that in the file selected from the cross section tape.
3. The product I x J x K x G is greater than 150,000.
4. List A or B is out of order.
5. The configuration is outside the I or J boundary.
6. The product J x K is greater than 1000 for 8000 word memory, 2000 for 16,000 word memory, or 4000 for 32,000 word memory.
7. Elastic Scattering was requested (this is not coded at present).
8. There are too many words of input in the Geometrical and Physical Data, List A, List B, Mesh Spacings and Coordinates, Edit Input, or the Rough Power Guess.

The maximum number of words allowed in the various sections of input is as follows:

Input	Maximum No. of Words
Geometrical and Physical Data	50
List A	1500
List B	1000

Input	Maximum No. of Words
Mesh spacings and Coordinates	250
Edit Input	550
Rough Power Guess	500

After the input has been checked, this section generates k-planes of composition indices and writes them on tape three. Each i-column is written as a record and each k-plane as a file. After all k-planes have been generated, the i-planes are generated by sorting the k-planes and are written one record per plane on tape 2. It is possible to remove the program from the machine at this time so that control rod positioning may be checked.

The program constructs a working library of the remaining programs on tape 5 and writes the Group Source program on tape 3. The cross section records are read from tape 4, and Master Tape 2 of the Master Tape code is read from Tape 1. Tapes 1 and 4 are then replaced by blank tapes.

MASTER TAPE 2 (see Figure 5)

This section writes part one of the Group Sweep restart procedure on the master tape (or tapes). This record is followed by the file containing the Group Source program.

Three passes over all planes in the mesh are made, unless the plane is a flux boundary. The first pass computes the sentinals and $C^{(2)}$ backward

coefficients which are written on the master tape as one record.

The second pass computes the E, $r^{(1)}$, $r^{(2)}$, $C^{(3)}$ coefficients. These coefficients are written as records, which vary in length depending on memory size but are always an integral number of columns. Each column is preceded by a block of $r^{(1)}$, $r^{(2)}$, $C^{(3)}$ backward coefficients. The coefficients E, $r^{(1)}$ $r^{(2)}$, $C^{(3)}$, and backward coefficients $r^{(1)}$, $r^{(2)}$, $C^{(3)}$ are omitted if the point lies on a flux boundary. If the last i-plane is a current boundary, the $r^{(1)}_{I,j,k}$ are replaced by $C^{(1)}_{I,j,k}$. Also, if k=K is a current boundary, then $C^{(3)}_{i,j,K}$ are computed by the formula given for boundary points. If the plane j=J is a current boundary, $r^{(2)}_{i,J,k}$ are replaced by $C^{(2)}_{i,J,k}$.

The third pass is made to compute $H^{(g)}$, $F'^{(g)}$, and V''. $H^{(g)}$ are computed for all planes unless the plane is a zero flux boundary. $H^{(g)}$ are written as one record (JK words) per plane on tape 1. The $F'^{(g)}$ are written on tape 4 as one record per plane. Only the planes between $i^F_{min.}$ and $i^F_{max.}$ inclusive are written, and on those planes only the points between $j^F_{min.}$, $j^F_{max.}$ and $k^F_{min.}$, $k^F_{max.}$ inclusive are written. $F'^{(g)}$ are computed only for groups $g \geq g'$. V'' are computed only during the mesh pass when g=G. $H^{(g)}$ are not computed for this group.

At the end of a group $g < G$, the Group Sweep code is transferred from tape 5 to the master tape following a file of E, $r^{(1)}$, $r^{(2)}$, $C^{(3)}$. The Group

Source code is written after the file containing the Group Sweep code. The $H^{(g)}$ are transferred to the master tape from tape 1. The mesh passes are then repeated for the next higher group. At the end of group g=G, the Group Sweep code is transferred to the master tape and the code for Master Tape 3 is read into memory.

MASTER TAPE 3 (see Figure 6)

This section transfers the Fission Source code from tape 5 to the (Figure 7) master tape. The following files are written on the master tape: (1) the $F'^{(g)}$, one group per file, (2) the Edit code, (3) Edit input, (4) V'', (5) mesh spacings, and (6) the I-planes of composition indices. The Source Flux Guess Word (SFGW) is inspected to see if a flux guess tape is supplied from a previous problem. If it is supplied, the machine stops and the flux guess tape is put on tape 5. If a rough power guess is supplied, it is expanded. The unnormalized source guess is computed and written on tape 4. This guess is then normalized and written on tape 5. The block of input needed by the Group Source and Group Sweep programs is set up. The Group Source code is read in from tape 3.

It is now possible to provide a duplicate of the master tape if one was not written simultaneously with the first copy. After duplication of the master tape (if performed), control is transferred to the Group Source program.

TAPE UNIT OPERATION

Every record that is written on tape is backspaced and read to check for errors. If there is a tape check or a check sum error, the record is rewritten and reread. If there is still an error, the machine stops. When reading a record, the check sum and tape check are tested; if either is in error the record is backspaced and reread. If there is still an error, the machine stops. If a tape error occurs in master tape 1, the program must be reloaded and started again. If the error occurs in master tape 2 and master tape 3, it is possible to try again to rewrite or reread the record. If the error repeats, the entire program must be restarted.

The records on tapes 2, 3, 4, and 5, are numbered. The program checks to see that it has the correct record. If the tape is mispositioned, the program backspaces or skips over the proper number of records and checks again to see if it has the correct record. If the correct record is not found, the machine stops. In master tape 2 and master tape 3 it is possible to try to read the record once more. If the correct record cannot be found, the entire program must be restarted.

MASTER TAPE PROGRAM - 1

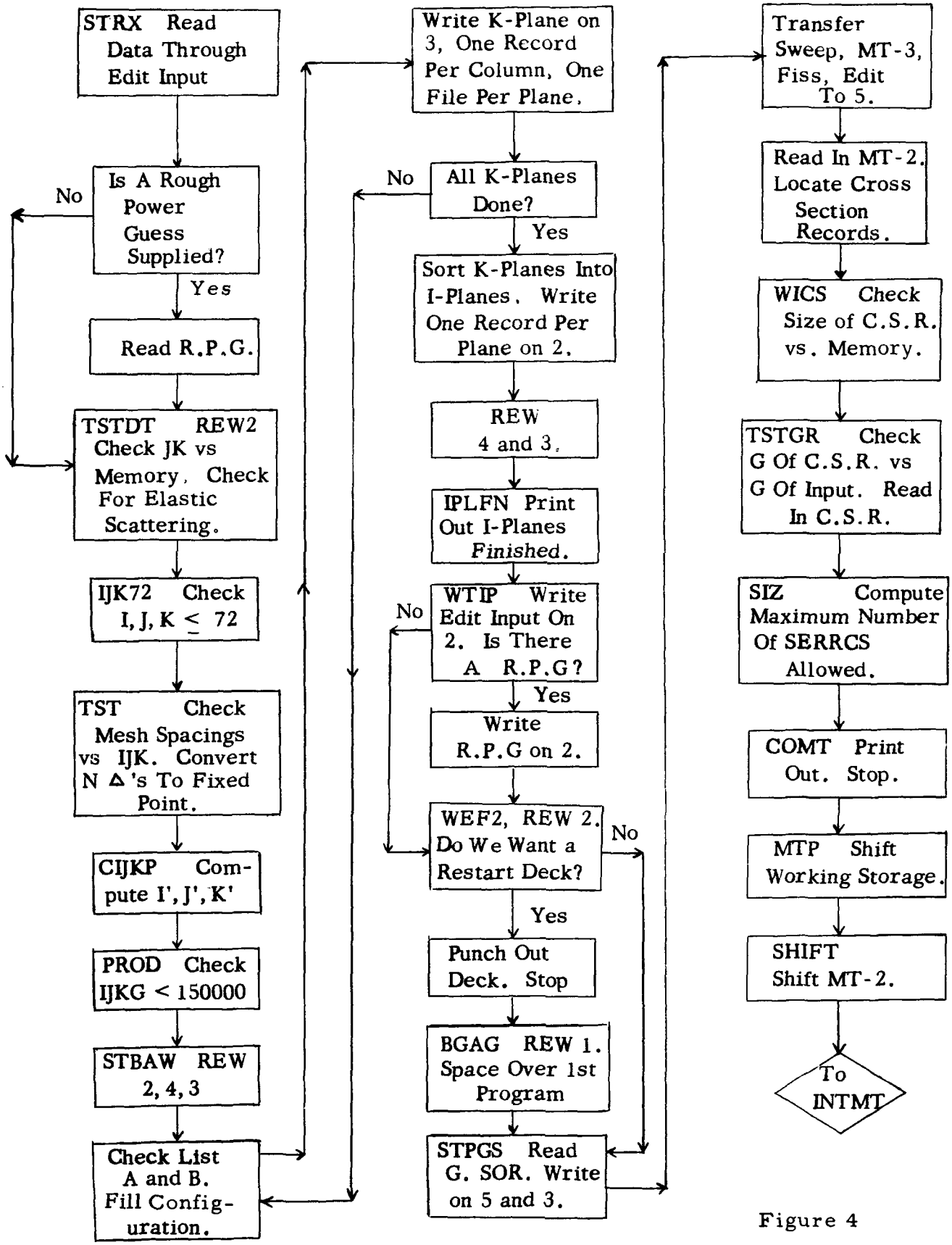


Figure 4

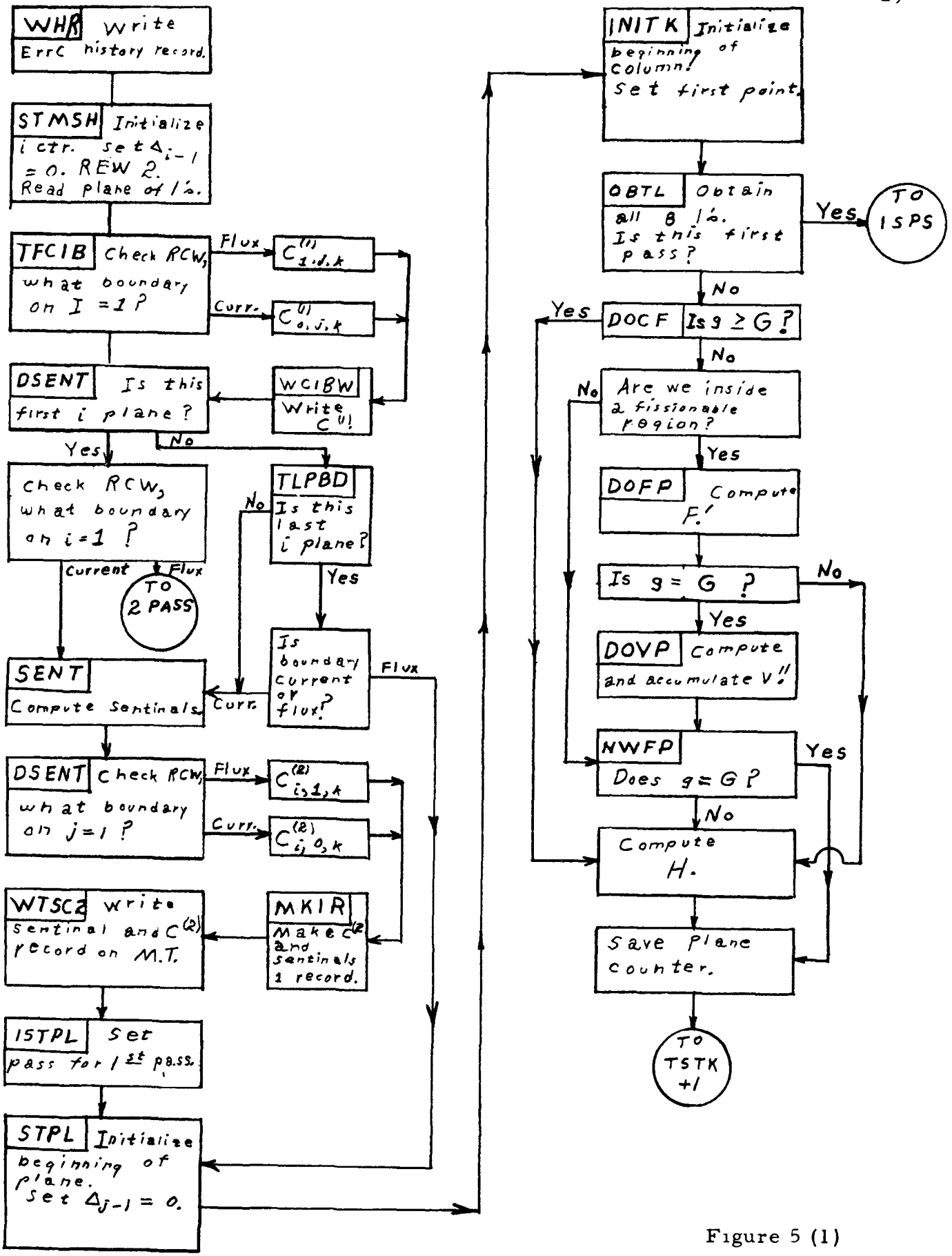

WHR Write Err. history record.
STMSH Initialize i ctr. set Δi-1 = 0. REW 2. Read plane of I's.
TFCIB Check RCW, what boundary on I = 1 ?
Flux
Curr.
DSENT Is this first i plane ?
WCIBW Write C(1)
Yes
No
Check RCW, what boundary on i = 1 ?
current
Flux
TO 2 PASS
TLPBD
No
Is this last i plane?
Yes
Is boundary current or flux?
Flux
Curr.
SENT
Compute Sentinals
DSENT Check RCW, what boundary on j = 1 ?
Flux
Curr.
WTSC2 write sentinal and C(2) record on M.T.
MKIR Make C(2) and sentinals 1 record.
ISTPL Set pass for 1st pass.
STPL Initialize beginning of plane. Set Δj-1 = 0.
INITK Initialize beginning of column. Set first point.
OBTL Obtain all B I's. Is this first pass?
Yes
TO ISPS
No
DOCF Is g ≥ G?
Yes
No
Are we inside a fissionable region?
No
Yes
DOFP Compute F.
Is g = G ?
No
Yes
DOVP Compute and accumulate V.
NWFP Does g = G?
Yes
No
Compute H.
Save plane counter.
TO TSTK +1

Figure 5 (1)

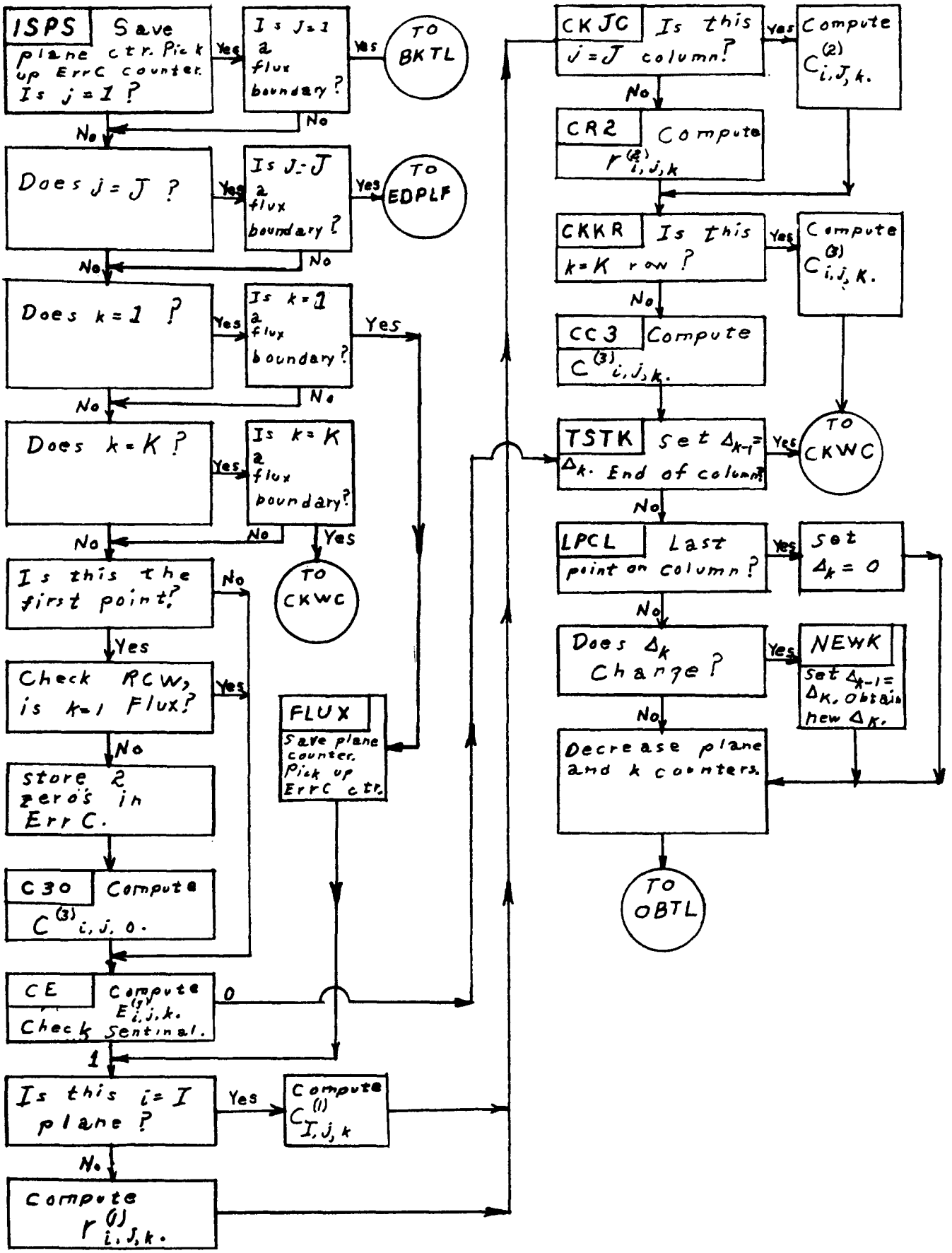

ISPS Save plane ctr. Pick up ErrC counter. Is j = 1 ?
Is J = 1 a flux boundary?
Yes
TO BKTL
No
Does j = J ?
Is J = J a flux boundary?
Yes
TO EDPLF
No
Does k = 1 ?
Is k = 1 a flux boundary?
Yes
No
Does k = K ?
Is k = K a flux boundary?
Yes
No
TO CKWC
Is this the first point?
No
Yes
Check RCW, is k=1 Flux?
Yes
No
store 2 zero's in ErrC.
C30 Compute C(3) i, j, 0.
FLUX Save plane counter. Pick up ErrC ctr.
CE Compute E(3) i, j, k. Check Sentinal.
0
1
Is this i = I plane ?
Yes
Compute C(1) I, j, k
No
Compute r(0) i, J, k.
CKJC Is this j = J column?
Yes
Compute C(2) i, J, k.
No
CR2 Compute r(2) i, j, k
CKKR Is this k = K row ?
Yes
Compute C(3) i, j, K.
No
CC3 Compute C(3) i, j, k.
TSTK Set Δk-1 = Δk. End of column?
Yes
TO CKWC
No
LPCL Last point on column?
Yes
Set Δk = 0
No
Does Δk Change?
Yes
NEWK Set Δk-1 = Δk. Obtain new Δk.
No
Decrease plane and k counters.
TO OBTL

Figure 5 (2)

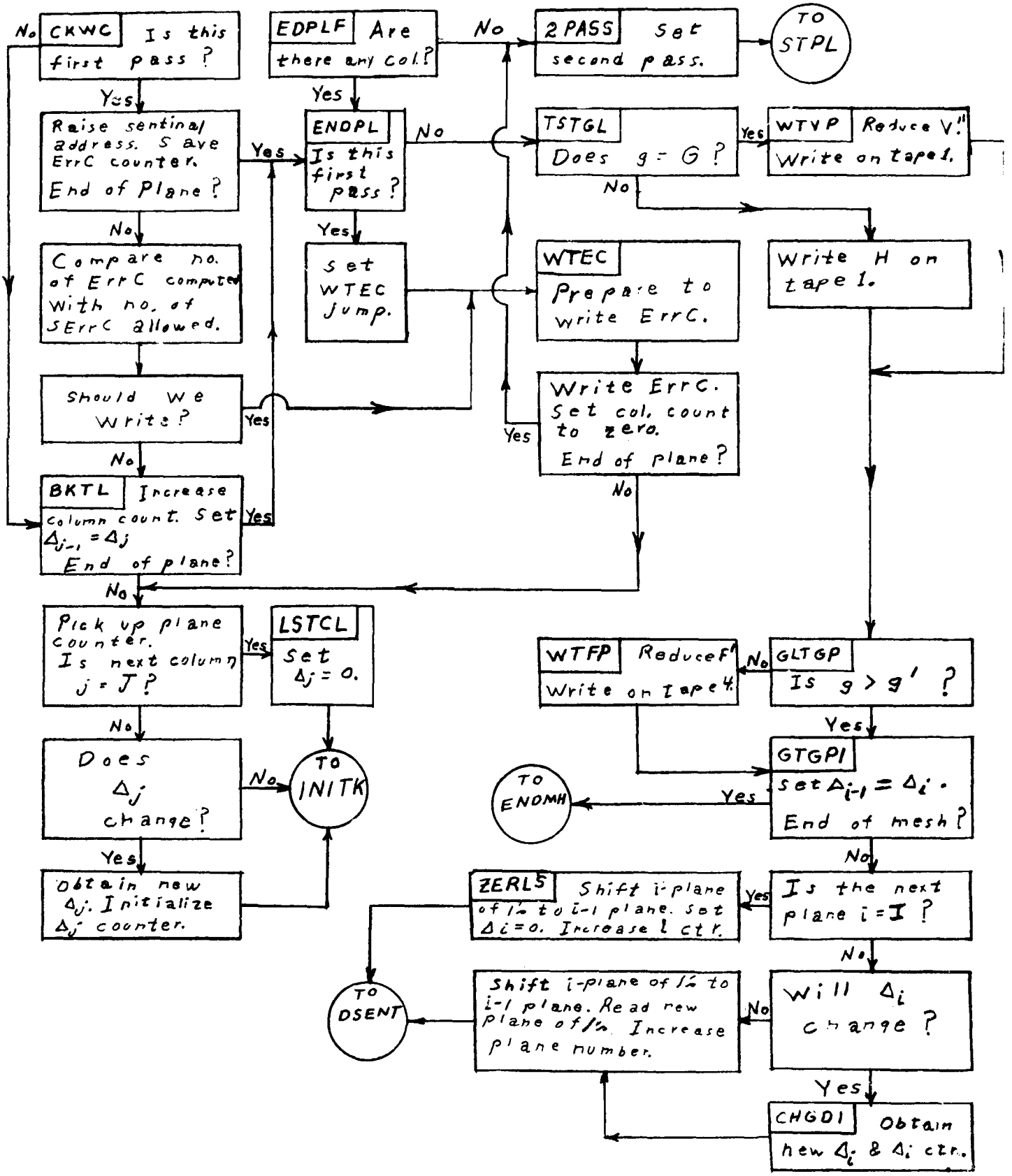

CKWC Is this first pass?
No
Yes
Raise sentinal address. Save ErrC counter. End of Plane?
No
Compare no. of ErrC computed with no. of SErrC allowed.
Should We Write?
Yes
No
BKTL Increase column count. Set Δj-1 = Δj End of plane?
Yes
No
Pick up plane counter. Is next column j = J?
Yes
No
Does Δj change?
No
Yes
Obtain new Δj. Initialize Δj counter.
LSTCL Set Δj = 0.
TO INITK
EDPLF Are there any col.?
Yes
ENDPL Is this first pass?
Yes
Set WTEC jump.
No
2PASS Set second pass.
TO STPL
TSTGL Does g = G?
Yes
WTVP Reduce V' Write on tape 1.
No
Write H on tape 1.
WTEC Prepare to write ErrC.
Write ErrC. Set col. count to zero. End of plane?
Yes
No
WTFP Reduce F' Write on tape 4.
GLTGP Is g > g' ?
No
Yes
GTGPI Set Δi-1 = Δi. End of mesh?
Yes
TO ENDMH
No
Is the next plane i = I?
Yes
No
ZERL5 Shift i-plane of 1/2 to i-1 plane. Set Δi = 0. Increase l ctr.
Will Δi change?
No
Shift i-plane of 1/2 to i-1 plane. Read new plane of 1/2. Increase plane number.
TO DSENT
Yes
CHGDI Obtain new Δi & Δi ctr.

Figure 5 (3)

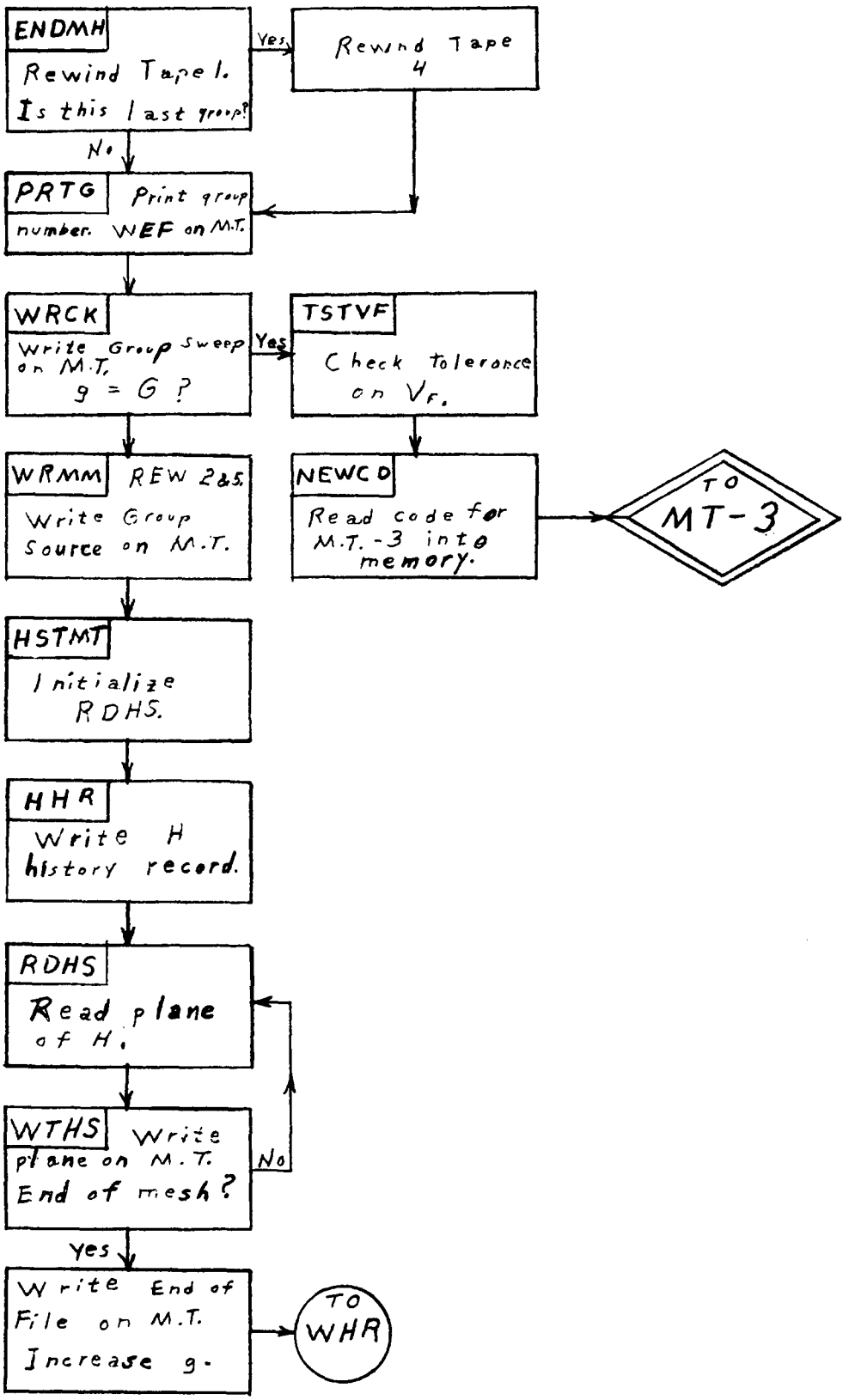

ENDMH
Rewind Tape 1.
Is this last group?
Yes
Rewind Tape 4
No
PRTG
Print group number. WEF on M.T.
WRCK
Write Group Sweep on M.T.
g = G ?
Yes
TSTVF
Check Tolerance on V_F.
WRMM
REW 2 as.
Write Group Source on M.T.
NEWCD
Read code for M.T. -3 into memory.
TO MT-3
HSTMT
Initialize RDHS.
HHR
Write H history record.
RDHS
Read plane of H.
WTHS
Write plane on M.T.
End of mesh?
No
Yes
Write End of File on M.T.
Increase g.
TO WHR

Figure 5 (4)

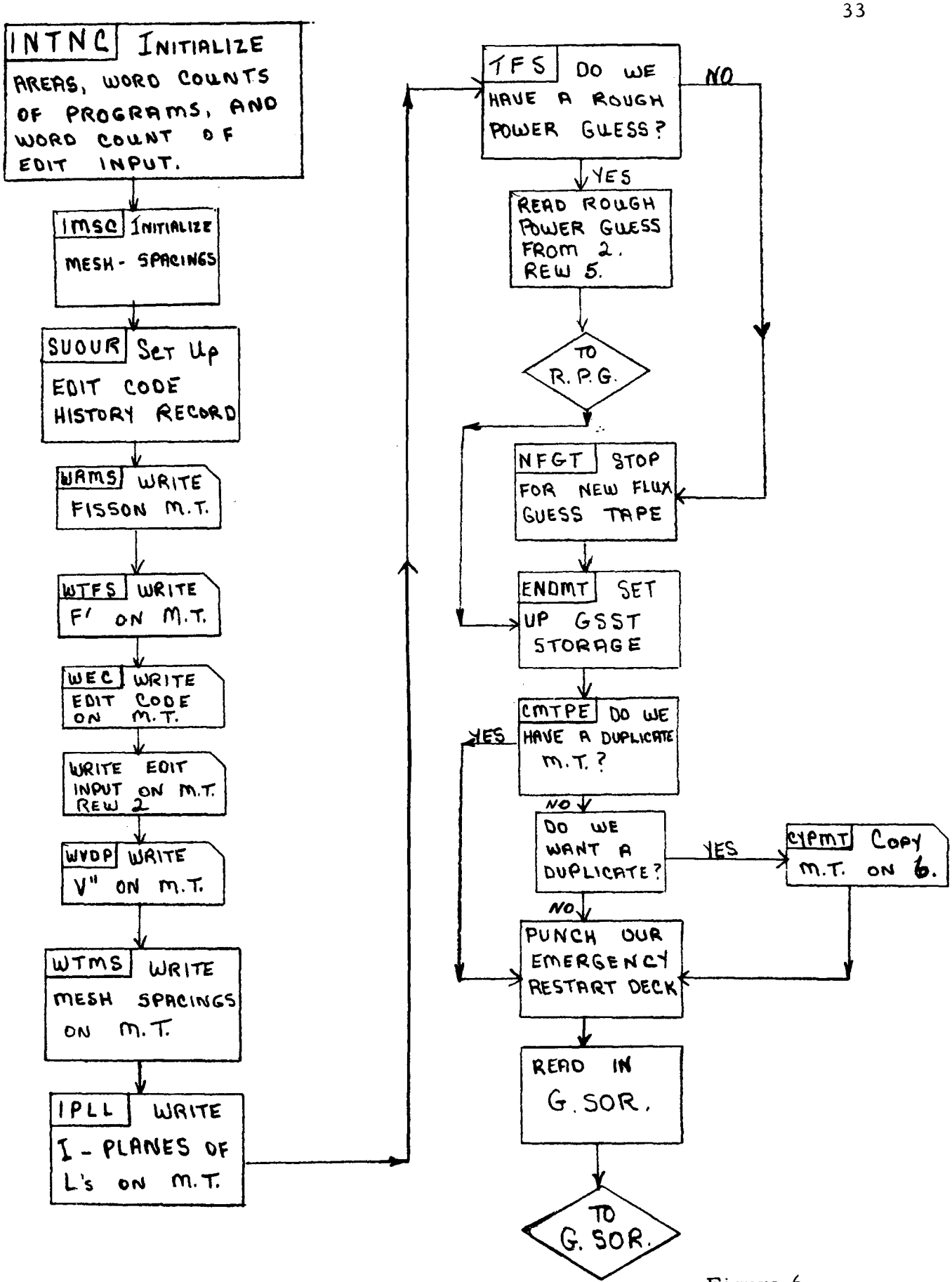

INTNC Initialize areas, word counts of programs, and word count of edit input.
IMSC Initialize mesh-spacings
SUOUR Set up edit code history record
WAMS Write fisson M.T.
WTFS Write F' on M.T.
WEC Write edit code on M.T.
Write edit input on M.T. Rew 2
WVDP Write V" on M.T.
WTMS Write mesh spacings on M.T.
IPLL Write I-planes of L's on M.T.
TFS Do we have a rough power guess?
NO
YES
Read rough power guess from 2. Rew 5.
To R.P.G.
NFGT Stop for new flux guess tape
ENDMT Set up GSST storage
CMTPE Do we have a duplicate m.t.?
YES
NO
Do we want a duplicate?
YES
CYPMT Copy m.t. on 6.
NO
Punch our emergency restart deck
Read in G.SOR.
To G.SOR.

Figure 6

ROUGH FISSION SOURCE GUESS

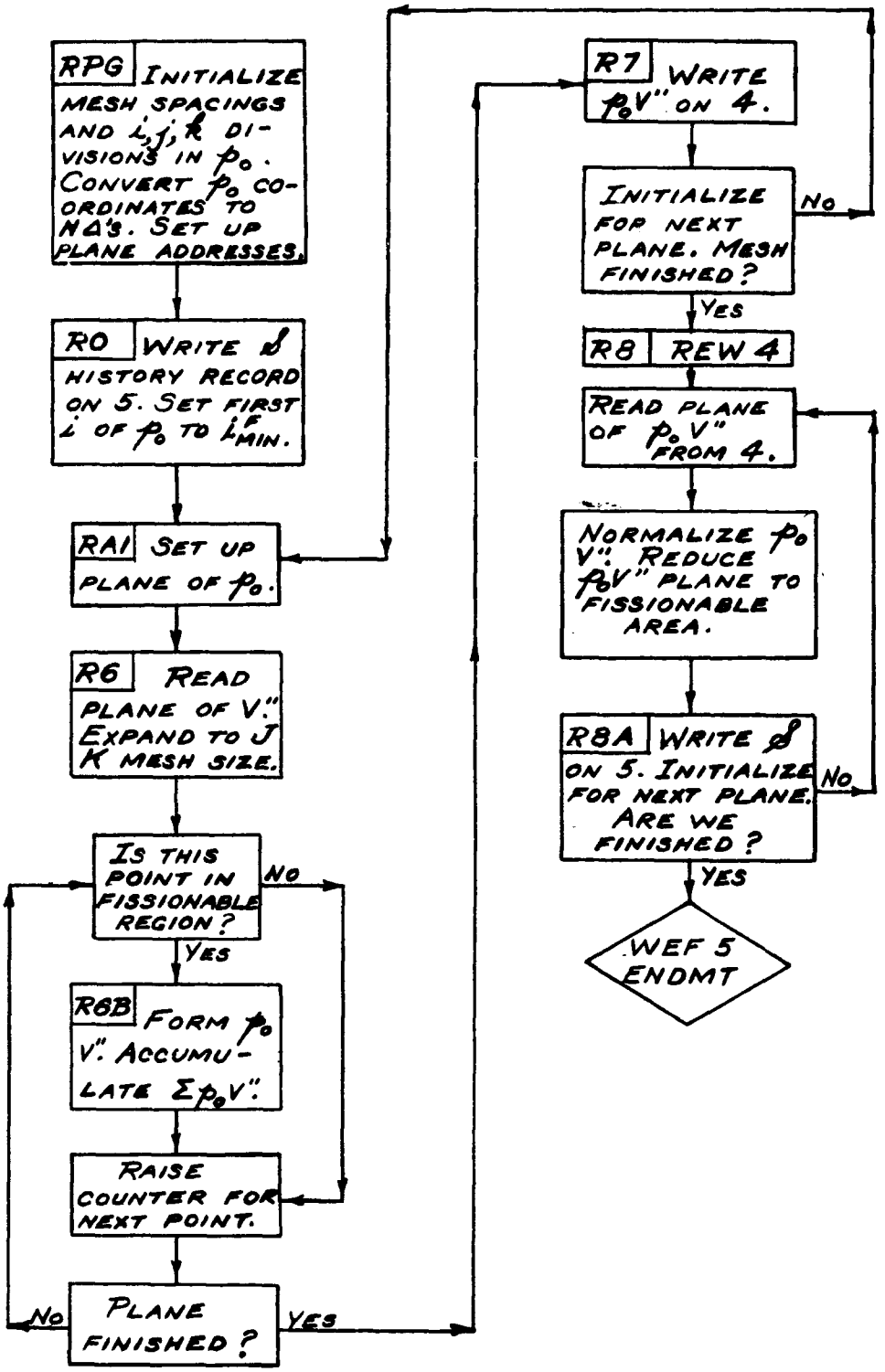


Figure 7

VI. GROUP SOURCE PROGRAM

PURPOSE

The purpose of the TRIXY Group Source Program (GSP) is to prepare certain data for use by the Group Sweep Program. These data consist of a file of SErrC and associated coefficients and a file of ϕ (flux) values over which iteration is to be performed. In addition, the Group Source Program saves on tape the results of all previous flux iterations so that they are readily available either to the Editing Program (in case of convergence) or for a subsequent run (in case of non-convergence). (See Figures 8 and 9.)

INPUT

To allow these purposes to be accomplished, the following information is written on tape by previous programs: (See Figures 10, 11, and 12).

1. Tape A:
   a. Files of ErrC, associated coefficients, and sentinels for each group. The history (identification) record of each ErrC file contains value of χ (fraction of the fission spectrum occurring in each group) used by the Group Source Program in calculating S (group source). It also contains other values which are transferred to the history records of the ϕ files for use by the Group Sweep Program.

b. Files of H (scattering coefficients) for each group except the last.

2. ,pe B:

a. A file of iterated ϕ for the previous group (not applicable when the Group Source Program is in group 1).

3. Tape E:

a. A file of χ (fission source).

b. A file or files of ϕ guess. (These files are not present at the start of the first run if a ϕ guess is made by the Group Source Program).

NOTE: There are files of information present on the tapes other than those enumerated above. However, these files are not used by the Group Source Program.

OPERATION

The operation of the Group Source Program can be described by the following chronological steps. It is assumed that each file written on tape includes a history record.

1. (Not applicable to group 1). A plane of H for the previous group is read from tape A. The corresponding plane of iterated φ is read from tape B. This latter plane is transferred to tape E (for ultimate

use by the Fission Source Program), following the last file previously written on tape E. Points of H are multiplied by corresponding points of ϕ, and a plane of $H\phi$ is written on tape C. This entire step is repeated for all planes, and an end of file is written on tapes C and E. Tapes B, C, and E are rewound.

2. The values of χ pertinent to the group under consideration are obtained from the ErrC history record on tape A and stored for future use.

3. A plane of $H\phi$ is read from tape C. If a record of backward coefficients is present, it is read from tape A. A record of sentinels is read from tape A and transferred to tape D. These sentinels are also retained in memory for future use. A plane of S is read from tape E and value of S at each point in the plane is calculated from the formula

$$S = \chi S + H\phi$$

Appropriate short-cuts are taken in this calculation in case (a) $\chi = 0$ or 1, (b) S is not given, as for points outside the fissionable region, or (c) $H\phi$ is not present, as in the case of group 1.

4. A block of ErrC coefficients is read from tape A. Under control of the sentinels, the ErrC coefficients are written on tape D so that each value of E is immediately preceded by the corresponding value of S. The records thus produced are known as SErrC records. At this point, if

a ϕ guess for the current group is not present on tape E, the program proceeds to step 5. If a ϕ guess is present, a test is made to see if the current SErrC plane is completed. If it is not completed, step 4 is repeated. If the plane is completed, the program proceeds to step 6.

5. A flux guess is made for each point in the current block according to the formula

$$\phi = S/E.$$

A test is made to see if the plane is completed. If the plane is not completed, step 4 is repeated; otherwise, the program proceeds to step 6.

6. A test is made to see whether all planes are processed. If they are not processed, the procedure is repeated from step 3. After all planes are completed, an end of file is written on tapes B and D, and on tape E, if a ϕ guess was made by the Group Source Program. Tapes B, C, and D are rewound. The ϕ guess tape (tape B if a ϕ guess was made, tape E if it was given) is positioned to the beginning of the ϕ file over which the Group Sweep Program is to iterate. The latter program is read into memory and control is transferred to it.

FINAL RESULTS

After all groups are processed for a run, the information written on

tapes B, C, and D by the Group Source Program is erased by either the Group Sweep Program or the Fission Source Program. However, the pertinent data is written on tape E for use by the Editing Program in either of two orders.

1. First run with ϕ guess not given:

a)	$\S$	
b)	ϕ guess for group 1	written by G.S.P. during group 1
c)	Iterated ϕ for group 1	written by G.S.P. during group 2
d)	ϕ guess for group 2	written by G.S.P. during group 2
e)	Iterated ϕ for group 2	written by G.S.P. during group 3
f)	ϕ guess for group 3	written by G.S.P. during group 3
g)	Iterated ϕ for group 3	written subsequently by Fission Source Program

2. First run with ϕ guess given, and all subsequent runs:

a)	$\S$	
b)	f (Absent during first run)	
c)	ϕ guess for group 1	
d)	ϕ guess for group 2	
e)	ϕ guess for group 3	
f)	Iterated ϕ for group 1	written by G.S.P. during group 2
g)	Iterated ϕ for group 2	written by G.S.P. during group 3
h)	Iterated ϕ for group 3	written subsequently by Fission Source Program

GROUP SOURCE PROGRAM

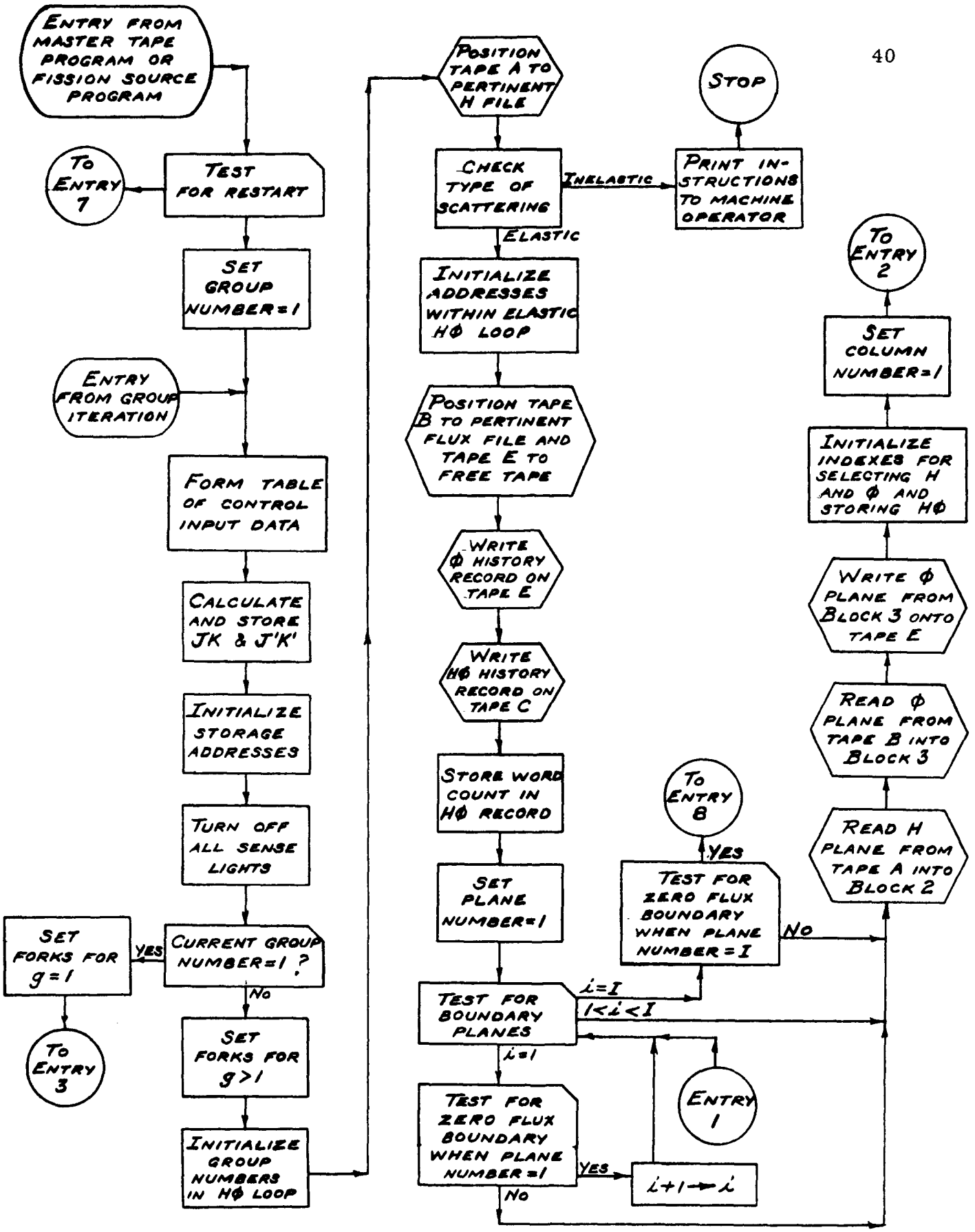


Figure 8 (1)

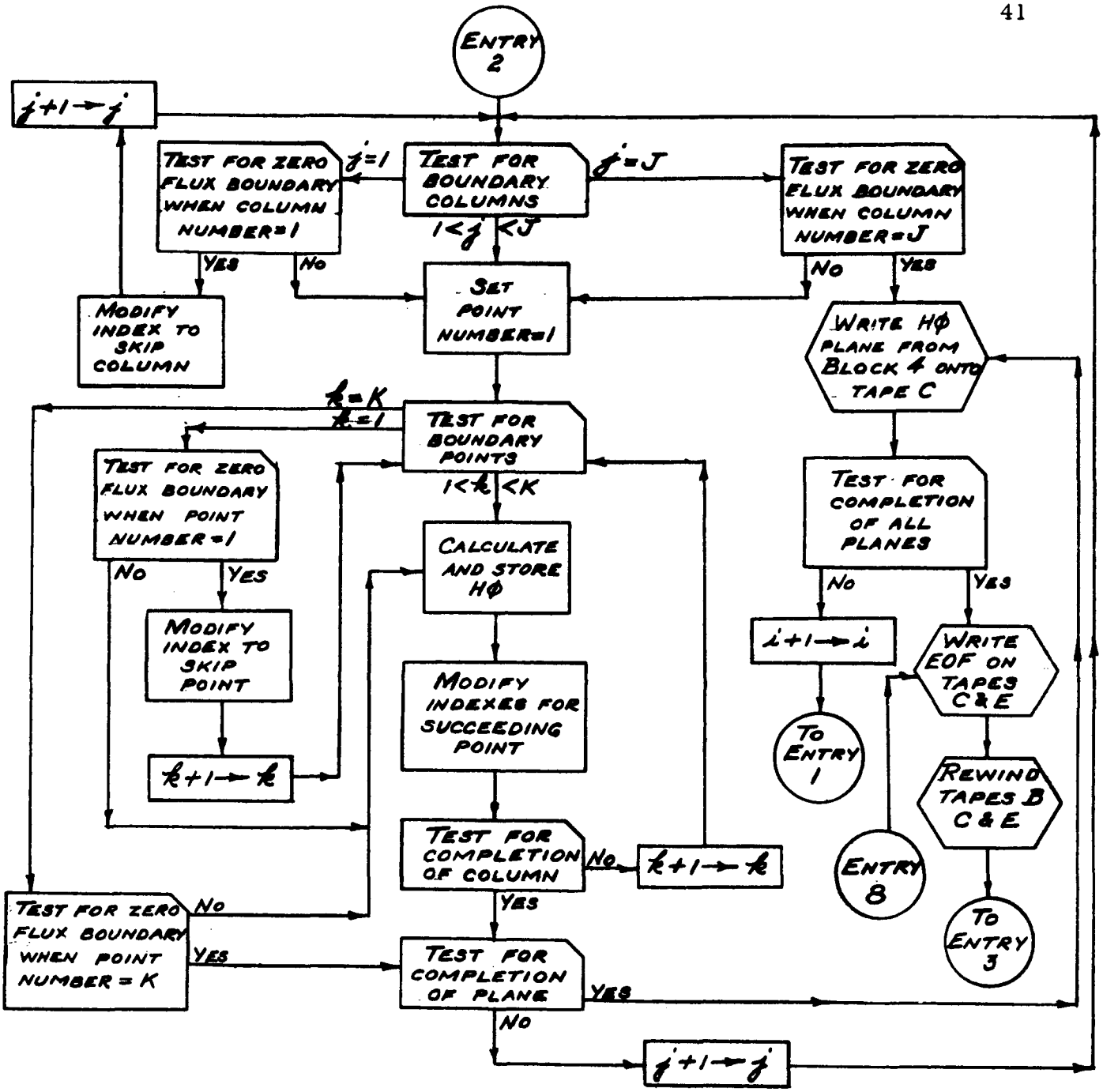

ENTRY 2
j+1→j
TEST FOR ZERO FLUX BOUNDARY WHEN COLUMN NUMBER=1
j=1
TEST FOR BOUNDARY COLUMNS
j=J
TEST FOR ZERO FLUX BOUNDARY WHEN COLUMN NUMBER=J
1<j<J
YES
NO
NO
YES
MODIFY INDEX TO SKIP COLUMN
SET POINT NUMBER=1
WRITE HΦ PLANE FROM BLOCK 4 ONTO TAPE C
k=K
k=1
TEST FOR BOUNDARY POINTS
TEST FOR ZERO FLUX BOUNDARY WHEN POINT NUMBER=1
1<k<K
TEST FOR COMPLETION OF ALL PLANES
CALCULATE AND STORE HΦ
NO
YES
NO
YES
MODIFY INDEX TO SKIP POINT
i+1→i
WRITE EOF ON TAPES C & E
MODIFY INDEXES FOR SUCCEEDING POINT
TO ENTRY 1
k+1→k
REWIND TAPES B C & E
TEST FOR COMPLETION OF COLUMN
NO
k+1→k
ENTRY 8
TEST FOR ZERO FLUX BOUNDARY WHEN POINT NUMBER=K
NO
YES
TO ENTRY 3
YES
TEST FOR COMPLETION OF PLANE
YES
NO
j+1→j

Figure 8 (2)

GROUP SOURCE PROGRAM

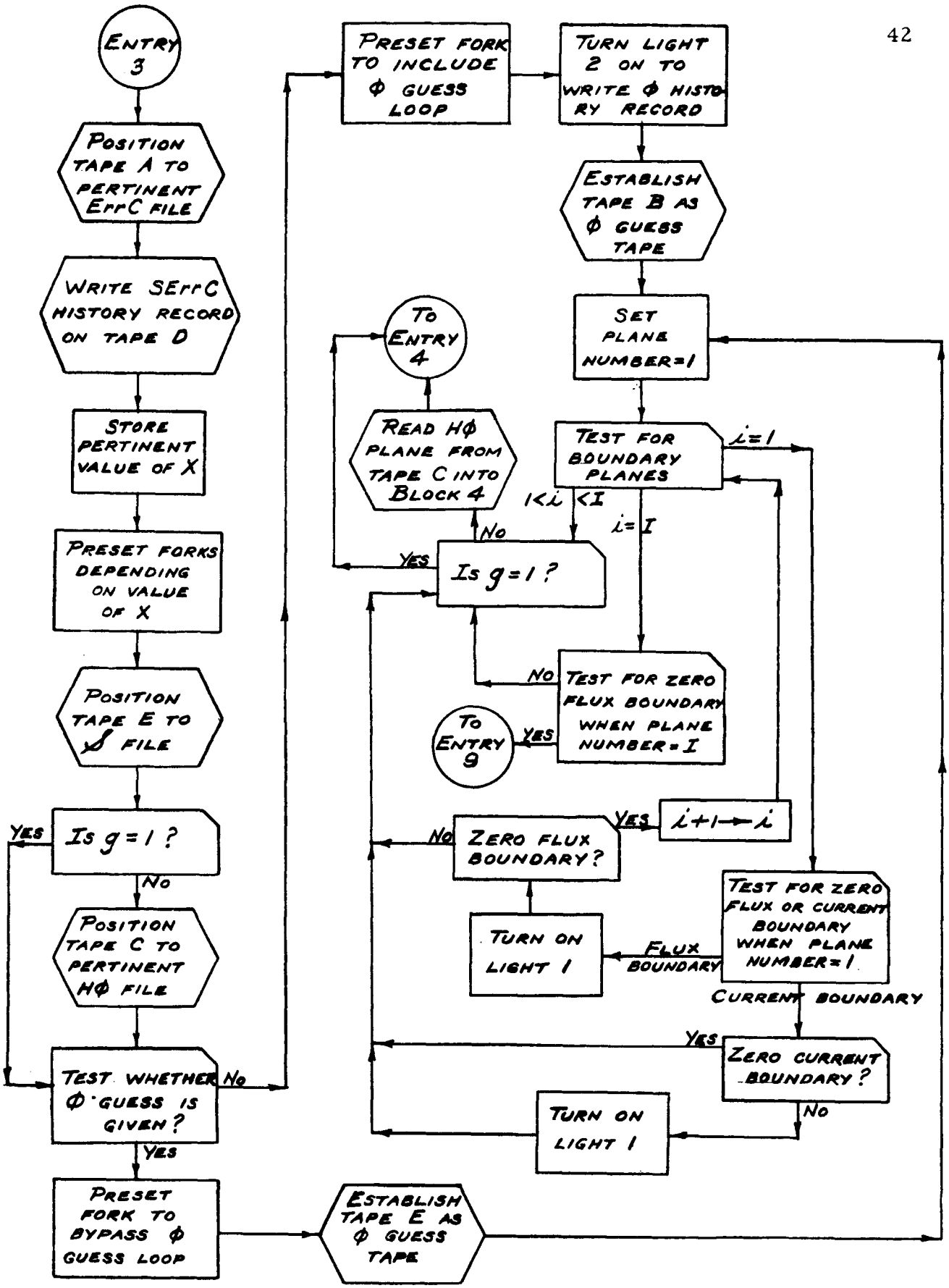


Figure 8 (3)

GROUP SOURCE PROGRAM

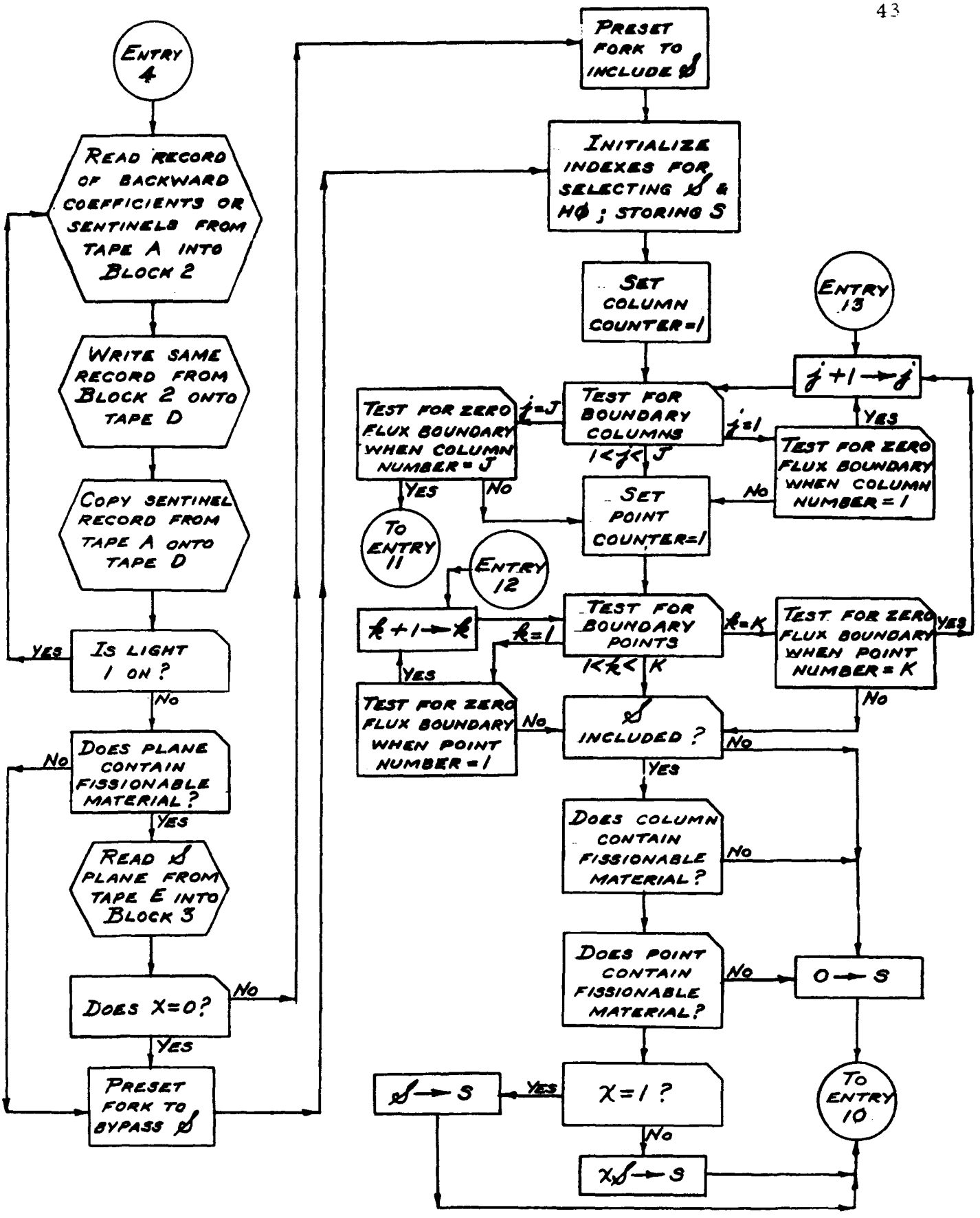


Figure 8 (4)

GROUP SOURCE PROGRAM

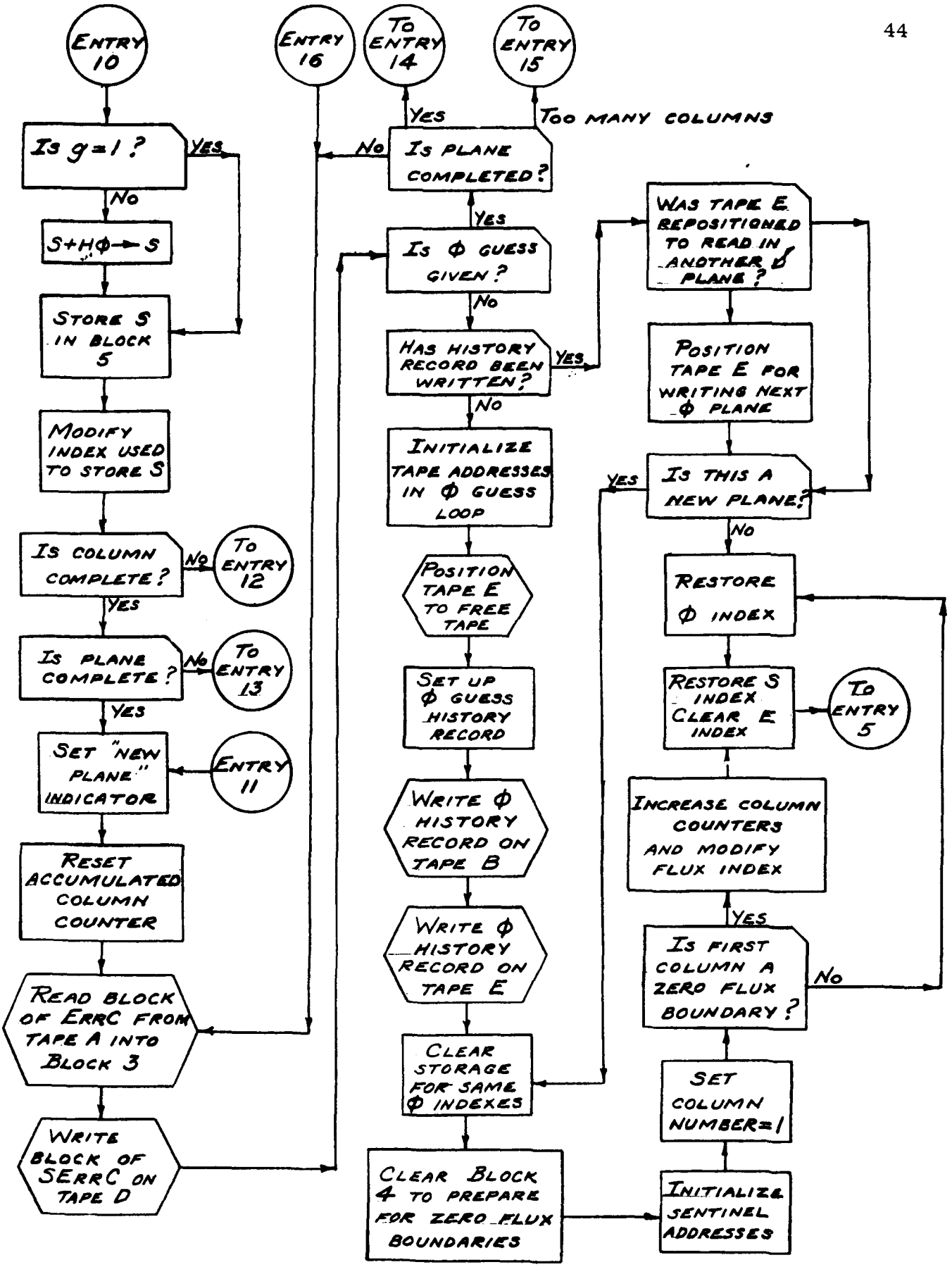


Figure 8 (5)

GROUP SOURCE PROGRAM

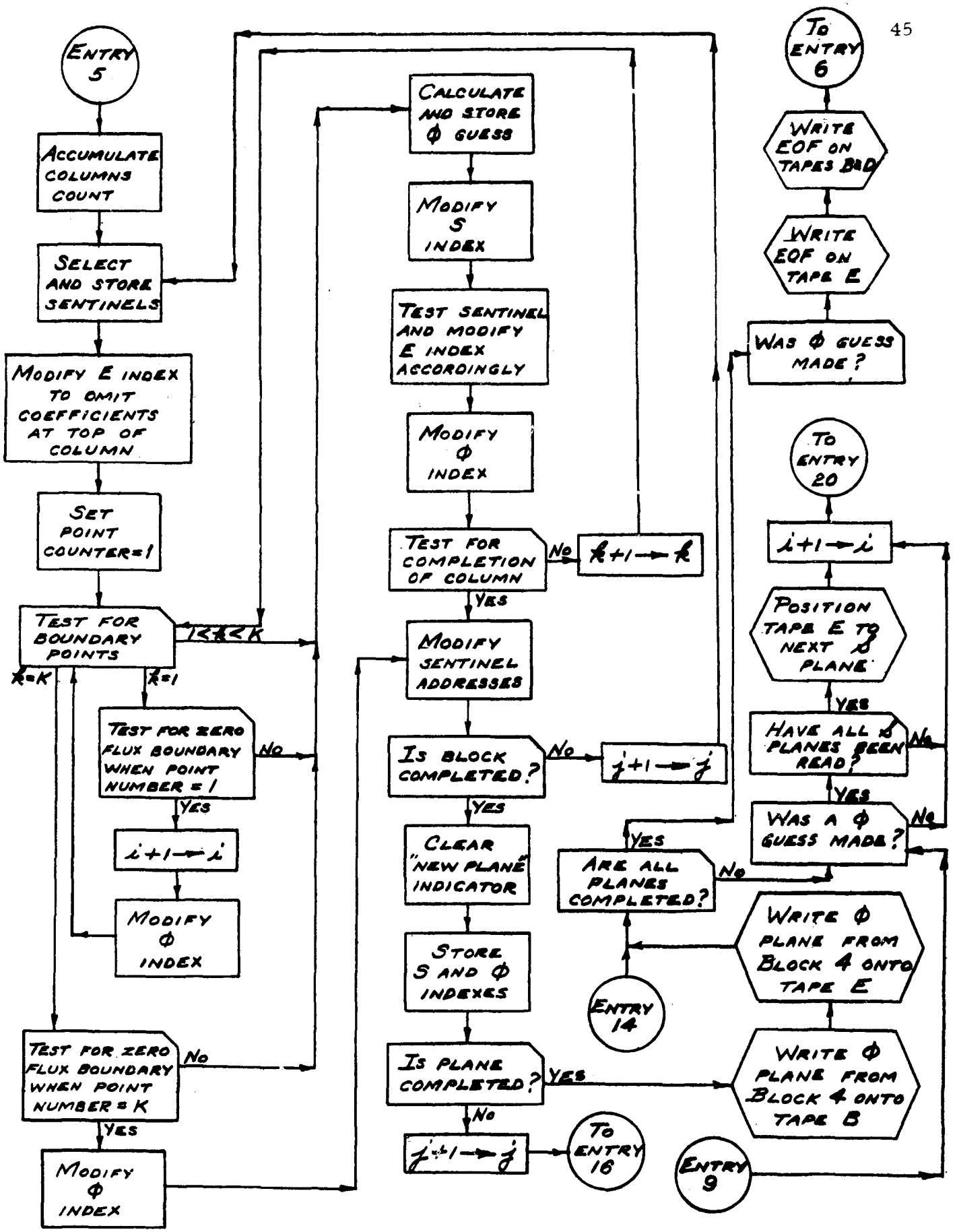


Figure 8 (6)

SERRC SUBROUTINE

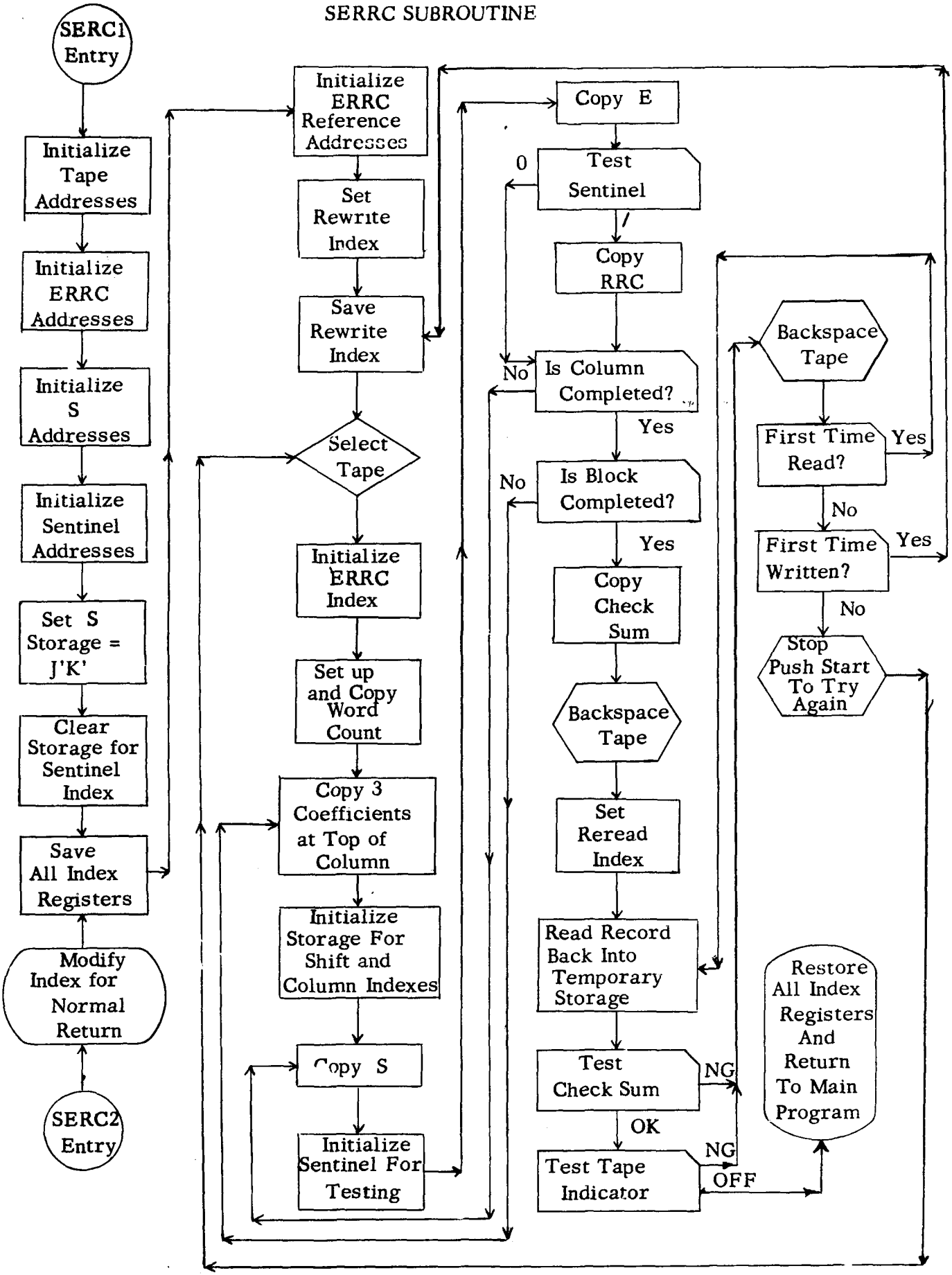


Figure 9

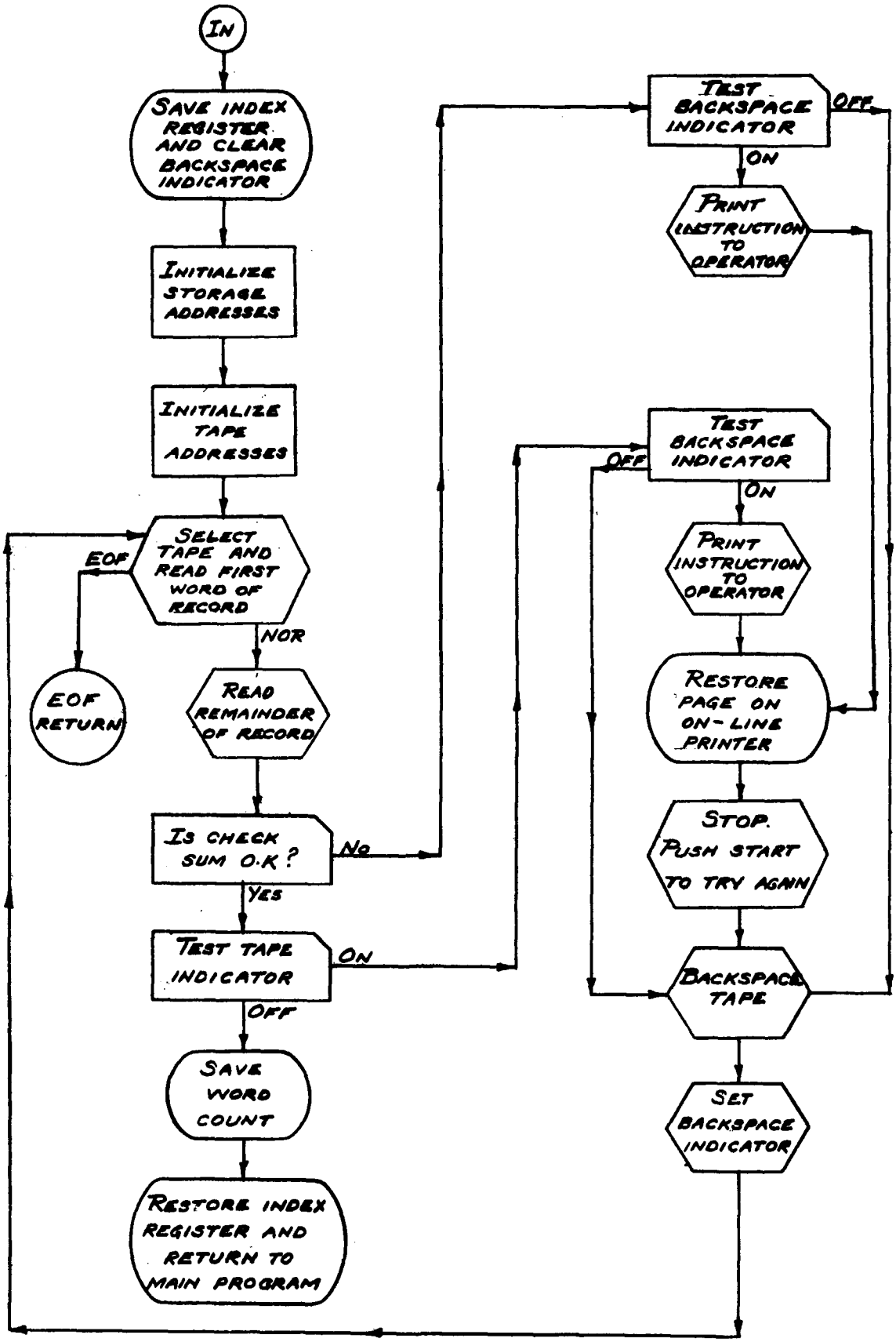

IN
SAVE INDEX REGISTER AND CLEAR BACKSPACE INDICATOR
INITIALIZE STORAGE ADDRESSES
INITIALIZE TAPE ADDRESSES
SELECT TAPE AND READ FIRST WORD OF RECORD
EOF
NOR
EOF RETURN
READ REMAINDER OF RECORD
IS CHECK SUM O.K ?
No
Yes
TEST TAPE INDICATOR
ON
OFF
SAVE WORD COUNT
RESTORE INDEX REGISTER AND RETURN TO MAIN PROGRAM
TEST BACKSPACE INDICATOR
OFF
ON
PRINT INSTRUCTION TO OPERATOR
TEST BACKSPACE INDICATOR
OFF
ON
PRINT INSTRUCTION TO OPERATOR
RESTORE PAGE ON ON-LINE PRINTER
STOP. PUSH START TO TRY AGAIN
BACKSPACE TAPE
SET BACKSPACE INDICATOR

Figure 10

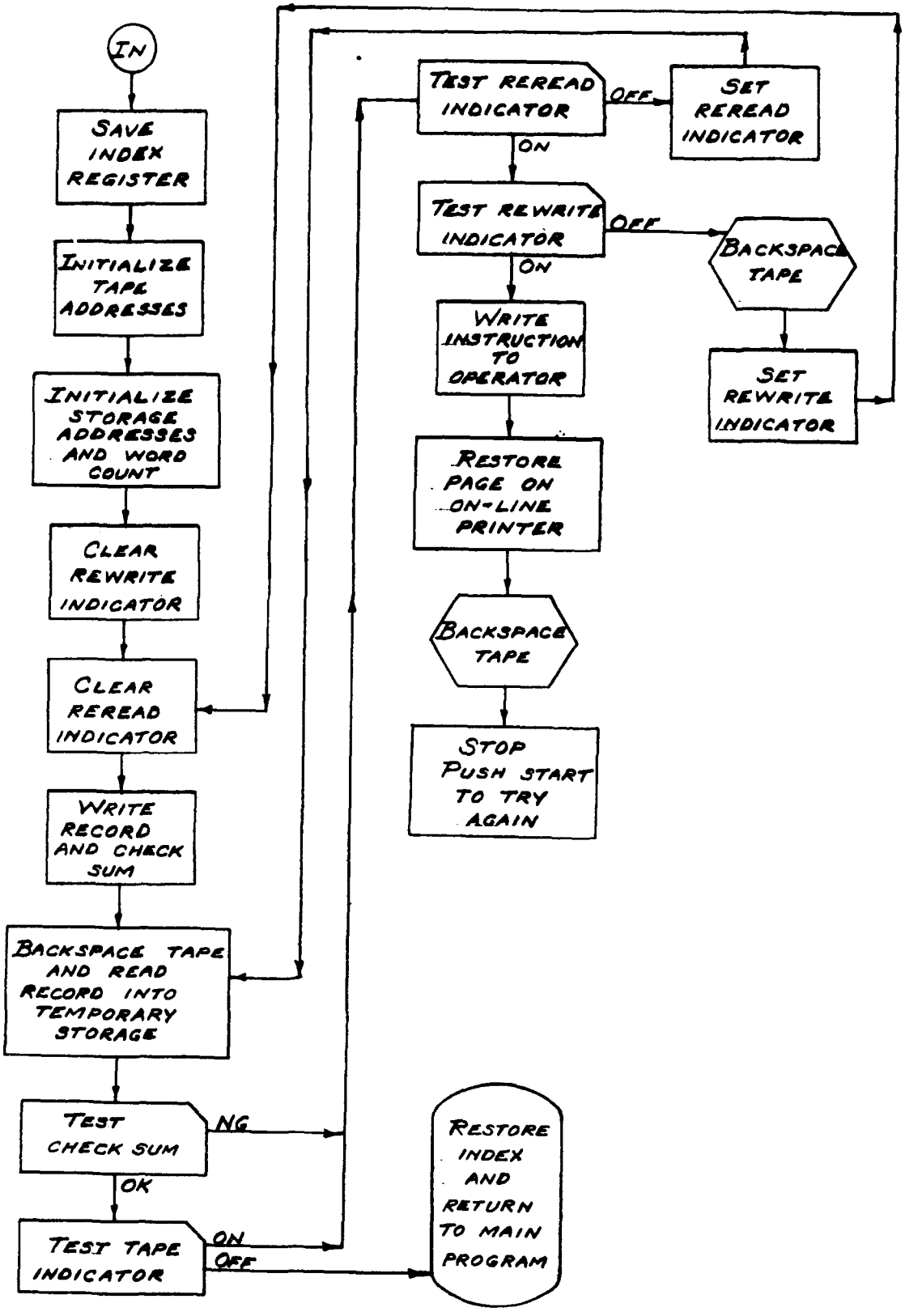

IN
SAVE INDEX REGISTER
INITIALIZE TAPE ADDRESSES
INITIALIZE STORAGE ADDRESSES AND WORD COUNT
CLEAR REWRITE INDICATOR
CLEAR REREAD INDICATOR
WRITE RECORD AND CHECK SUM
BACKSPACE TAPE AND READ RECORD INTO TEMPORARY STORAGE
TEST CHECK SUM
NG
OK
TEST TAPE INDICATOR
ON
OFF
RESTORE INDEX AND RETURN TO MAIN PROGRAM
TEST REREAD INDICATOR
OFF
SET REREAD INDICATOR
ON
TEST REWRITE INDICATOR
OFF
BACKSPACE TAPE
SET REWRITE INDICATOR
ON
WRITE INSTRUCTION TO OPERATOR
RESTORE PAGE ON ON-LINE PRINTER
BACKSPACE TAPE
STOP PUSH START TO TRY AGAIN

Figure 11

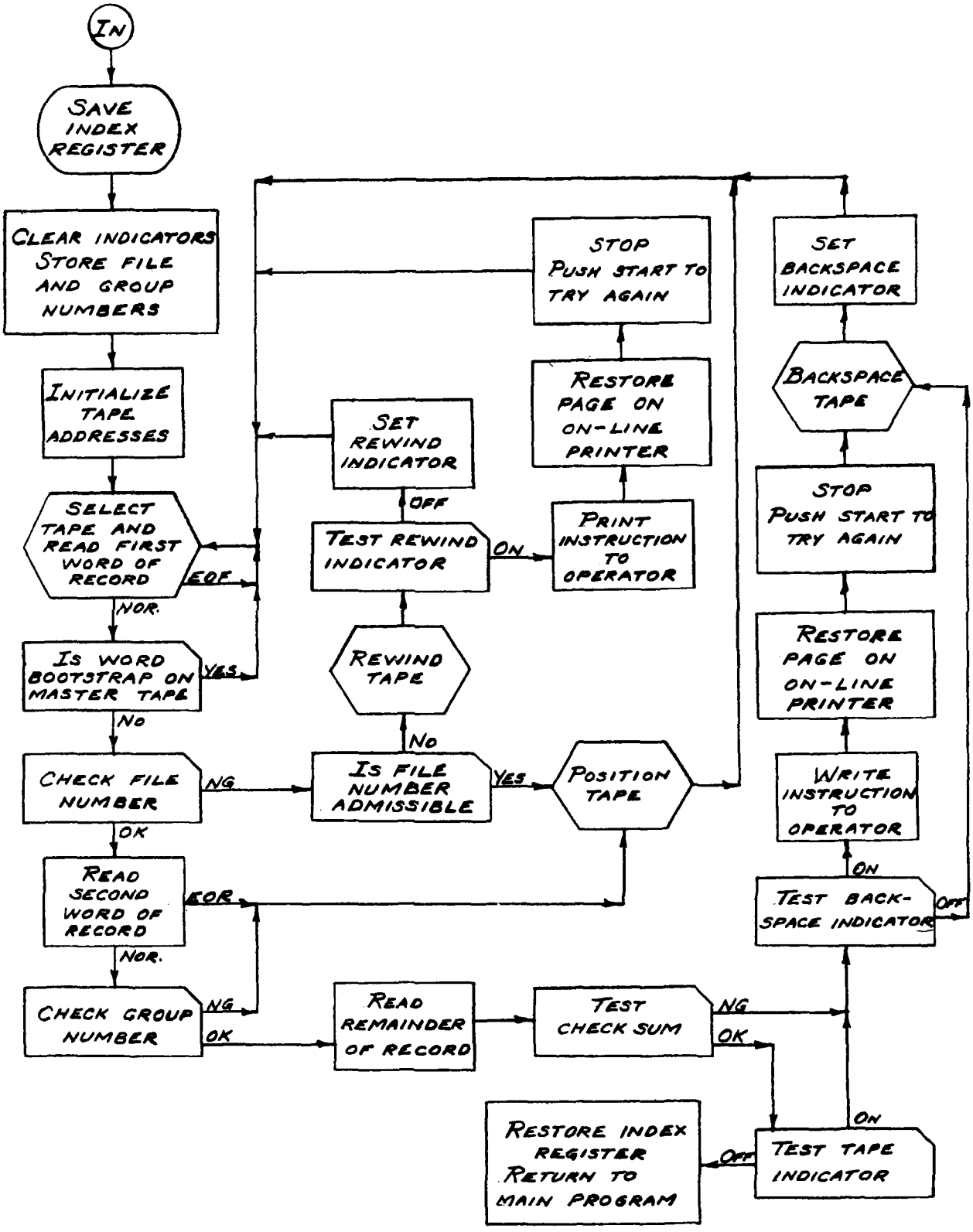

IN
SAVE INDEX REGISTER
CLEAR INDICATORS STORE FILE AND GROUP NUMBERS
INITIALIZE TAPE ADDRESSES
SELECT TAPE AND READ FIRST WORD OF RECORD
EOF
NOR.
IS WORD BOOTSTRAP ON MASTER TAPE
YES
No
CHECK FILE NUMBER
NG
OK
READ SECOND WORD OF RECORD
EOR
NOR.
CHECK GROUP NUMBER
NG
OK
SET REWIND INDICATOR
OFF
TEST REWIND INDICATOR
ON
REWIND TAPE
No
IS FILE NUMBER ADMISSIBLE
YES
READ REMAINDER OF RECORD
STOP PUSH START TO TRY AGAIN
RESTORE PAGE ON ON-LINE PRINTER
PRINT INSTRUCTION TO OPERATOR
POSITION TAPE
TEST CHECK SUM
NG
OK
SET BACKSPACE INDICATOR
BACKSPACE TAPE
STOP PUSH START TO TRY AGAIN
RESTORE PAGE ON ON-LINE PRINTER
WRITE INSTRUCTION TO OPERATOR
ON
TEST BACK-SPACE INDICATOR
OFF
ON
TEST TAPE INDICATOR
OFF
RESTORE INDEX REGISTER RETURN TO MAIN PROGRAM

Figure 12

VII. GROUP SWEEP PROGRAM *

The TRIXY Group Sweep Program improves the fluxes of each energy group (g) by applying the "accelerated Liebmann technique." The flux estimates are given at mesh points i, j, k and are improved as follows. Each point along a line in the $X^{(3)}$ (or K) direction (i, j constant) is improved in successive order from k=1 to k=K; this process is called a "line sweep." In a given plane (i constant), the line sweeps are made in successive order from j=1 to j=J, and this process is referred to as a "plane sweep." Similarly, a "mesh sweep" is defined as the process of making plane sweeps in successive order from i=1 to i=I. In theory, a group iteration consists of as many sweeps as are necessary to satisfy the following predetermined condition:

$$\epsilon_{\phi}^{(t)} < \delta_{\phi}^{g}, \qquad \text{(VII 1)}$$

where δ_{ϕ}^{g} is the primary convergence criterion for the fluxes. However, two other means for terminating the group iteration are used:

$$\epsilon_{\phi}^{(t)} < C_1\% \text{ of } \epsilon_{\phi}^{(1)} \qquad \text{(VII 2)}$$

where $C_1\%$ is the percentage of initial total improvement that serves as a secondary convergence criterion, and

$$t \geq t_{max}, \qquad \text{(VII 3)}$$

where t_{max}, the upper limit on the number of sweeps allowable, is the

* Group Sweep Flow Charts (Figure 13)

tertiary "convergence criterion." For the point iteration to improve $\varphi^{g(t-1)}_{i,j,k}$, the following fluxes at the six neighboring points are used: $\varphi^{g(t)}_{i-1,j,k'}$ $\varphi^{g(t)}_{i,j-1,k'}$ $\varphi^{g(t)}_{i,j,k-1'}$ $\varphi^{g(t-1)}_{i+1,j,k'}$ $\varphi^{g(t-1)}_{i,j+1,k'}$ and $\varphi^{g(t-1)}_{i,j,k+1}$. After each point iteration, the improved value is used for the remainder of the mesh sweep. The "Liebmann Acceleration Factor" used in the point iteration equation is constant throughout a group iteration.

An additional factor that must be considered is the neutron balance at the end of each mesh sweep. It is taken into account by a renormalization factor, $R^{g(t-1)}$. Moreover, the flux change between successive iterations, $\epsilon^{(t)}_{\varphi}$, is calculated at each mesh point, and the neutron leakage, $L^{(1)}_{1}$, $L^{(1)}_{I}$, etc., is computed for each mesh point on a boundary plane.

ASSUMPTIONS

The Group Sweep Program is stored on a master tape, read into memory, and given control by the Group Source Program. The following quantities are available in a block of core storage (see "Glossary of Symbols"):

Location (Decimal)	Decrement	Tag	Address
29	α (in floating point form)		
30	g'	0	G
31	g^{i}	C	RCW
32	--	0	I'

Location (Decimal)	Decrement	Tag	Address
33	--	0	J'
34	--	0	K'
35	--	0	I
36	--	0	J
37	--	0	K
38	tape A	0	tape B
39	tape C	0	tape D
40	tape E	0	flux guess tape
41	run number	0	tape F
42	0	4	1 (HTJ 1, 4)

Tapes are supplied as follows:

1. Tape A: master tape (positioned to read Group Source Calculation Routine).
2. Tape B: available for working storage (rewound) or the flux guess tape (see 5. below).
3. Tape C: available for working storage (rewound).
4. Tape D: coefficient tape containing the following file (rewound):

Record Number	Contents
0	History record
1	Plane (J'K' values) of $C^{(1)}_{i-1,j,k}$ coefficients.[1]
2	Word count (in decrement); J' (or 2J') sentinels; K' values of $C^{(2)}_{i,j-1,k}$ coefficients
3	Column count (in decrement) and word count (in address); $r^{(1)}_{i-1,j,k'}$ $r^{(2)}_{i,j-1,k'}$ $C^{(3)}_{i,j,k-1}$; SE $r^{(1)}_{i,j,k'}$ $r^{(2)}_{i,j,k'}$ $C^{(3)}_{i,j,k}$ for an integral number of columns
4 . . . N	Successive records of 3 until all values for plane i have been recorded
N+1	Record number 2 for plane i+1
N+2	Repeat of record number 3 for the i+1 plane

This pattern is repeated until all I' planes have been recorded. The last record is followed by an end of file gap.

5. Flux guess tape: tape E or B depending on whether or not a flux guess is available from a previous problem, positioned to the file containing the following:

[1] This record is not present if plane i=1 is a zero flux plane.

Record Number	Contents
0	History record
1	Plane (JK values) of flux estimates for plane i=1[1]
2	Successive records of
.	i-planes until all
.	the flux guess have
N	been recorded

In general, N=I'. However, if either planes i=1 or i=I are non zero flux planes, then N=I.

6. Tape F: available for working storage (rewound).

PROGRAMMING METHOD

Point Iteration

For the iteration at the point i, j, k, three coefficients are available from previous point iterations. (Boundary points are exceptions and are discussed later.) The coefficient $C^{(1)}_{i-1,j,k}$ is held in memory from an iteration in the previous plane. $C^{(2)}_{i,j-1,k}$ is held in memory from an iteration in the previous column (a "column" is a line of mesh points along the $X^{(3)}$ direction). $C^{(3)}_{i,j,k-1}$ is retained from the previous point iteration.

During a plane sweep of the group iteration, fluxes for the planes i-1, i, and i+1 are held in memory; this requires 3JK words. Values of

---

[1] This record is not present if plane i=1 is a zero flux plane.

$C^{(1)}_{i-1,j,k}$ held in memory require JK words, while retention of $C^{(2)}_{i,j-1,k}$ requires K additional words. Sentinels that are described below require J words if $K \leq 36$ or 2J words if $K > 36$.

Sources and Coefficients

Sources and coefficients are read into their reserved memory locations from tape D, one record at a time. Each record of sources and coefficients contains values of "SErrC" that have been calculated prior to the group iteration. For each mesh point, the record contains either the set of two values, $S_{i,j,k}$ and $E_{i,j,k}$, or the set of five values, $S_{i,j,k}$, $E_{i,j,k}$, $r^{(1)}_{i,j,k}$, $r^{(2)}_{i,j,k}$ and $C^{(3)}_{i,j,k}$. The sentinels are used to indicate which of these conditions holds. If the sentinel bit for a given k is "0", S and E are given; if the bit is "1" the set of five values is supplied. The sources and coefficients are stored in mesh sweep order; the three "backward" coefficients, $r^{(1)}_{i-1,j,k}$, $r^{(2)}_{i,j-1,k}$ and $C^{(3)}_{i,j,k-1}$, are stored at the beginning of each "column" of SErrC information. After the Group Sweep Routine and the quantities mentioned in the preceding paragraph have been stored, the memory space available determines the number of words in each SErrC record for a particular case. If possible, the record contains the sources and coefficients for an entire plane; if not, the record contains SErrC information for an integral number of columns.[1]

[1] Each record contains a word count and a column count.

Boundary Conditions

Boundary planes i=1, j=1, and k=1 are exceptions to the procedure described above, since the coefficients, $C^{(1)}_{i-1,j,k}$, $C^{(2)}_{i,j-1,k}$, and $C^{(3)}_{i,j,k-1}$ are not available from previous points. This problem is resolved by calculating the missing coefficients prior to the group iteration and storing them on tape D. The first record after the history record on tape D is a plane of $C^{(1)}_{i-1,j,k}$ values; however, if i=1 is a zero current boundary, then the plane of $C^{(1)}_{i-1,j,k}$ values is set to zero by the Group Sweep Routine. There is also a column of $C^{(2)}_{i,j-1,k}$ coefficients and the sentinels as a record preceding the SErrC information for each plane. The $C^{(3)}_{i,j,k-1}$ coefficient is supplied with $r^{(1)}_{i-1,j,k}$ and $r^{(2)}_{i,j,k-1}$ as the first set of coefficients for each column in the SErrC record(s) for a plane.

Iterations are skipped for mesh points in a specified flux boundary plane. Thus, if plane i=1 is a flux plane, flux iteration begins on plane i=2. The first plane of flux guesses supplied is $\phi_{i-1,j,k} = \phi_{1,j,k}$, and the plane of "backward" coefficients is $C^{(1)}_{i,j,k}$, etc. This problem of logic is handled by examining the boundary conditions (as described in the Reactor Control Word) at the beginning of a given group sweep and presetting several forks throughout the routine (see "Flow Diagram" for a detailed description of this procedure).

Note that for any current plane, the iteration equation contains fluxes for points that are outside the mesh. To save space, the memory locations for these fluxes are not carried as zeros. Instead, certain instructions used in computing the flux iteration equation are changed to "no operation" instructions and then are restored to their original form.

Use of Tapes

For the first mesh sweep, flux estimates are read from tape B (or tape E) and are written on tape C one plane at a time. At the end of the first sweep, the tape designations are interchanged as follows: the flux guess tape (tape B or tape E) is designated as tape F, the tape saved for the Repeat and Restart Procedures (see "Repeat and Restart Procedures"), tape C is designated tape B; and tape F is designated as tape C. During the second mesh sweep, and for all succeeding sweeps, fluxes are read from the tape designated as B. After the second sweep, the tape designations are interchanged as follows: tape C is designated as tape B; tape B is designated as tape F; and the original tape B is designated as tape C. After the first sweep, the flux guess tape is available for restart procedures. After the second sweep, nothing is written on the flux guess tape. Consequently, if tape E is the flux guess tape, it is positioned at the end of the file of flux guess planes and is not rewound or used for writing. After the third sweep, and for all succeeding sweeps, the tape

designations are interchanged cyclically; that is, the tape just read becomes the restart tape, the tape just written becomes the tape to read, and the tape previously saved for restart is now available for writing. The use of relative tape designations permits this procedure. The converged flux planes always appear on the tape designated as tape B. The flux planes are written on the tape designated tape C, but when convergence is reached, tape B and tape C designations are interchanged

Procedure After Convergence

The group number (g^i) of the flux estimates to be improved is supplied in the history records of tapes D and flux guess tape and in a memory location (see "Programming Conventions"). This group number is rewritten on the history record of tape B to indicate the group number of the converged fluxes contained on the tape. After convergence, the group number (g^p) stored in memory is compared with G, which is also stored in memory. If $g^p < G$, g^p is increased by one (so that $g^p = g^i + 1$), and the Group Source Program is read into memory. If $g^p = G$, the run number is increased by one, and the Fission Source Program is read into memory. In either case, tapes B, C, D, and F are rewound and control is transferred to the beginning of the program read into storage.

Storage Allocations

At the beginning of the Group Sweep Program the following list of addresses is computed from control data:

Master List of Initial Addresses

Location	Address	Description
ML	MS	Sentinels
ML+1	$MS + J' + 1 = (MS)_1$ or $MS + 2J' + 1 = (MS)_1$	$C^{(2)}_{i,j-1,k}$ column
ML+2	$(MS)_1 + K' + 1 = (MS)_2$	$C^{(1)}_{i-1,j,k}$ plane
ML+3	$(MS)_2 + JK + 1 = (MS)_3$	BLOCK 1 of flux
ML+4	$(MS)_3 + JK + 1 = (MS)_4$	BLOCK 2 of flux
ML+5	$(MS)_4 + JK + 1 = (MS)_5$	BLOCK 3 of flux
ML+6	$(MS)_5 + JK = (MS)_6$	SErrC

Here "MS" is the address of the first word of memory available for storage. (The Group Sweep Routine, associated subroutines, temporary storage, and control data are stored in the lower part of memory and MS is the location of the first word after this block.)

The maximum number of SErrC that can be retained in memory at one time by the Group Sweep Routine can be computed by the following formula:

$$N_{SErrC} = N_{Machine} - (MS + 5 + J^* + K' + 4JK), \qquad \text{(VII 4)}$$

where $N_{Machine}$ is the number of words of core storage in the machine, $J^* = J'$ if $K \leq 36$, or $J^* = 2J'$ if $K > 36$.

This formula is used in the master tape routine to determine the maximum number of ErrC columns that can be written in one record. If the Group Sweep Routine is reassembled, the absolute location for MS may change. In this case, the constant value of MS used in the master tape routine also must be changed.

Since storage locations are assigned within the program, increased memory capacity is utilized without changes in programming. The limits imposed on input quantities I, J, and K control storage allocations.

Tape Subroutines

All reading or writing on tapes B, C, D, E, or F is done with a tape read-write subroutine. When writing, the subroutine computes a complemented logical check sum and stores it as the last word of the record, but the record is not backspaced after writing to test the check sum. However, when reading, the check sum is tested as well as the tape check indicator, and if either test fails the subroutine returns to the error return. If an end of file is encountered, return is made to the error return.

The file identification records of programs on the master tape are read by means of the Read File Identification Subroutine. This subroutine reads the file identification, the first word of the program history record, and compares

it with the file identification given in the calling sequence. If the file identification checks, the subroutine reads the next record (the program) into temporary storage to check the check sum and the status of the tape check indicator. If these tests are all right, the master tape is backspaced one record, the Read Program Record Routine (RPR) is read into 0 through 21 (decimal), and the subroutine transfers control to the RPR subroutine (see below). If the file identification does not check, the subroutine positions the master tape to the beginning of the next file, checking to be certain that the end of the master tape has not been reached, and reads the next file identification. If the proper file is not found, this process continues to the end of the master tape which is marked by a double end of file. The tape is then rewound and the search is repeated. If the correct file cannot be found after the second pass, the subroutine prints the error indication instructions, calls for the extra master tape, and stops. The program is prepared to repeat the entire search when the start button is pressed. If either the check sum or tape check test fails, the subroutine repeats reading the record once before giving the error indication as above. This procedure for reading the file identification is an attempt to assure that the proper program is found and read into memory, thereby avoiding a stop in the operation of TRIXY.

The RPR subroutine is a tape read routine that reads the program into

core storage, beginning at location $(43)_{10}$, and transfers control to $(48)_{10}$. The routine reads the record twice before giving the error stop at $(42)_{10}$. This routine must be relocated to 0 through 21 (decima is used.

Entering New Data

During the operation of the Group Sweep Program it may be desirable to enter a new Liebmann acceleration factor (α). At the beginning of a mesh sweep, it is possible to load decimal data from the card reader by putting the deck for the data load program MDL9C and decimal data in the card reader. If the card reader is readied and sense switch 4 depressed, the Group Sweep Program loads the cards. MDL9C is read into upper memory by a special load routine (MBILD)* and destroyed later by reading in flux planes. If an error is found in the data, a comment is printed and the program stops at $(42)_{10}$. If an end of file card is encountered, the program stops at $(42)_{10}$. If the normal end of record punch is given, return is made to the Group Sweep Program. The Group Sweep Program prints instructions and stops at $(42)_{10}$ to allow these instructions to be followed.

Itermediate Printing

The Group Sweep Program uses the TRIXY Print Subroutine (PR) to print certain information during a group iteration.

*MBILD is loaded into locations 0 - 24 and does not destroy the control words stored in 29 - 42.

At the beginning of the first mesh sweep the following information is printed:

(1) Run and Group Number

(2) Convergence Code and Sweep Number

(3) Liebmann Acceleration Factor (α)

(4) $\delta \phi^{g}$ (primary convergence criterion)

(5) C_1 (secondary convergence criterion)

(6) t_{max} (tertiary convergence criterion)

At the end of each mesh sweep, the sweep number (t) is compared with a print sweep counter. The print sweep regulator (C_2) is stored in this counter on the first sweep, and C_2 is added to the counter each time it is equal to the sweep number. If the sweep number is greater than or equal to the print sweep counter and if sense switch 3 is depressed, the following quantities are printed:

(1) Run and Group numbers

(2) Convergence Code and Sweep Numbers

(3) Liebmann Acceleration factor (α)

(4) Flux Change (ϵ_{ϕ}^{t})

(5) Error ratio ($\epsilon_{\phi}^{t} / \epsilon_{\phi}^{(t-C_2)}$)

(6) C_1% of ϵ_{ϕ}^{t}

When convergence is reached this information is printed regardless of the setting of sense switch 3.

Repeat Procedure

Since the three tapes used for reading and writing flux planes are interchanged cyclically between mesh sweeps (see "Use of Tapes"), during any sweep (t) the flux planes from the previous sweep (t-1) are available. However, there are no flux planes available from the previous sweep when t=1, and the repeat procedure described below is not applicable during the first sweep. Also, an exception must be made when t=2. If the flux guess is supplied on tape E, during the second sweep tape E is the tape theoretically available for restart. Tape E would have to be rewound to use it again but the Group Source Program assumes that tape E is not rewound. The special procedure followed when t=1 or t=2 (and the flux guess tape is tape E) is described later.

If an error is encountered in reading flux planes during a sweep, the record is reread. If the error occurs again, the program transfers to the Repeat Procedure. The Repeat Procedure tests the sweep counter. If $t \neq 1$ and $t \neq 2$ (or if when t=2 the flux guess tape was tape B), the Repeat Procedure prints that repeat and restart procedures are available, prints instructions, and stops. When the start button is pressed, the routine transfers to Prepare Restart Procedure (PRP). If sense switch 5 is not down, PRP returns control to the Repeat Procedure. At this point, one is subtracted from the sweep counter, $R^{(t-1)}$ is stored in place of $R^{(t)}$, the designations of tape F (which contains the

flux planes for sweep t-1) and tape B are interchanged, tapes B, C, D, and F are rewound, and control is transferred to the beginning of the mesh sweep. In other words, if during sweep t' an error is encountered in reading from tape B, the Repeat Procedure begins the iteration on sweep t' - 1. Since all conditions for the iteration on sweep t' - 1 have been restored, using the Repeat Procedure produces no change in the results of a mesh sweep.

NOTE: If the error in reading flux planes on sweep t' occurs a second time, it would be desirable to remove TRIXY from the machine. (Some malfunction of the machine with regard to tape reading or writing is indicated, and this error prevents TRIXY from continuing). In this case, when the stop occurs which signifies that a repeat procedure is necessary, sense switch 5 may be depressed before the start button is pressed. As described above, a transfer to Prepare Restart Procedure (PRP) is executed (see below). The Repeat Procedure is designed to compensate for a single isolated tape error, while the Restart Procedure is used in event of a persistent error.

If t=1 or t=2 and the flux guess tape is tape E, the Repeat Procedure prints that repeat and restart procedures are not available and stops at the special location to indicate error.

Prepare Restart Procedure

In addition to the Repeat Procedure described above a method is

provided for interrupting TRIXY in the middle of the Group Sweep Program, removing TRIXY from the machine, and restarting the program after the interruption. The Restart Procedure is designed for either voluntary interruption, (e.g., when the machine is requested for the operation of other programs) or mandatory interruption (in the event of a persistent machine error that demands correction). At the beginning and the end of a mesh sweep, the Group Sweep Program transfers to the PRP subroutine. If sense switch 5 is up, PRP immediately returns control to the program. If sense switch 5 is down, the routine checks the sweep counter. If $t=1$ or $t=2$ and the flux guess tape is tape E, the routine returns control to the main program.

If $t\neq1$ and $t\neq2$ (or if $t=2$ and the flux guess tape is tape B), PRP prints instructions and prepares the restart procedure. The designations of tapes B and F are interchanged, and the flux guess tape is designated as tape B. The subroutine tests to see if tape A is designated as logical tape 1 or 3. If tape A is logical tape 1, PRP does not modify the tape designations, as tape A may be loaded using the load tape button. If tape A is not logical tape 1 but logical tape 3, PRP sets tape A to logical tape 1. The subroutine finds the tape formerly designated as logical tape 1 and sets this tape to logical tape 3, thus preventing two tapes from being designated as logical tape 1. PRP punches out the contents of locations 30 through 42 and stops at a special location to

indicate that TRIXY may be removed from the machine.

To restart TRIXY only the following need be done:

(1) The tapes must be replaced exactly as they were. If this is the first time TRIXY has been restarted, the master tape (formerly logical tape 3) must be set to tape 1, and the tape that was 1 must be set to tape 3.

(2) The card punched out by PRP, followed by the deck for the data load routine (MDL9C) and a data card giving the proper Liebmann factor (α), must be placed in the card reader.

(3) Sense switch 4 must be depressed and the other sense switches set as before.

(4) The load tape button must be pressed.

(More detailed instructions for restarting TRIXY are given in the "TRIXY Operating Manual", Appendix E.)

When the load tape button is pressed, the self-load loop on the master tape is read into memory. The data contained on the card punched out by PRP is read into memory locations 30 through 42, thereby re-storing the quantities that various TRIXY routines assume are in memory. The Group Source Program is read into memory, and a sense light is turned on to indicate to the Group Source Program that is is a restart. Control is transferred to the Group Source Program which places tape E in the required position. (If a flux guess

was available from a previous problem, tape E will be positioned as if the flux guess planes had been read by the Group Sweep Program.) The Group Source Program reads the Group Sweep Program into memory and transfers control to the Group Sweep Program. At this point, all the conditions of core storage, tape positions, and control at the time TRIXY was interrupted have been duplicated, except that the Liebmann factor (α) is not available. Consequently, α must be read into memory by the Group Sweep Program. (See instructions in (2) and (3) above.)

If TRIXY was interrupted in the middle of sweep t', the Restart Procedure starts the iteration on sweep t' - 1. There may be a slight difference in the results on sweep t' after TRIXY has been restarted, because the renormalization factor (R (t)) is set equal to one by the Group Sweep Routine. However, the effect of this change is negligible, and all tests of the restart procedure indicate that virtually no change in final results is noticable after a restart.

EQUATIONS

1. Point Iteration Equation:

$$\phi^{(t)}_{i,j,k} = \left[R^{(t-1)}(1-\alpha)\right] \varphi^{(t-1)}_{i,j,k} + \frac{\alpha}{B_{i,j,k}}\left[S_{i,j,k} + C^{(1)}_{i-1,j,k}\,\phi^{(t)}_{i-1,j,k} + C^{(2)}_{i,j-1,k}\,\varphi^{(t)}_{i,j-1,k} + C^{(3)}_{i,j,k-1}\,\varphi^{(t)}_{i,j,k-1} + R^{(t-1)}\left(C^{(1)}_{i,j,k}\,\phi^{(t-1)}_{i+1,j,k} + C^{(2)}_{i,j,k}\,\phi^{(t-1)}_{i,j+1,k} + C^{(3)}_{i,j,k}\,\phi^{(t-1)}_{i,j,k+1}\right)\right], \quad \text{(VII 5)}$$

where $\phi^{(t)}_{i,j,k}$ is the improved flux calculated from the flux estimate $\phi^{(t-1)}_{i,j,k}$, and α is the Liebmann Acceleration Factor.

2. Renormalization factor $(R^{(t-1)})$:[1] (VII 6)

$$R^{(t-1)} = \sum_{i=1,j=1,k=1}^{I,J,K} S_{i,j,k} \Bigg/ \left(L_B^{t-1} + \sum_{i=1,j=1,k=1}^{I,J,K} E_{i,j,k}\,\varphi^{(t-1)}_{i,j,k}\right),$$

where the neutron leakage from the mesh is

$$L_B^{(t-1)} = L_1^{(1)} + L_I^{(1)} + L_1^{(2)} + L_J^{(2)} + L_1^{(3)} + L_K^{(3)}. \quad \text{(VII 7)}$$

For specified current condition

$$L_1^{(1)} = \sum_{i=1,k=1}^{J,K} C^{(1)}_{0,j,k}\,\phi^{(t-1)}_{1,j,k}, \quad \text{(VII 8)}$$

$$L_I^{(1)} = \sum_{j=1,k=1}^{J,K} C^{(1)}_{I,j,k}\,\phi^{(t-1)}_{I,j,k} \text{ etc.} \quad \text{(VII 9)}$$

[1] $R^{(0)} = 1$

For specified flux condition

$$L_1^{(1)} = \sum_{j=1,\,k=1}^{J,\,K} C_{1,j,k}^{(1)} \left(\phi_{2,j,k}^{(t-1)} - \phi_{1,j,k}^{(t-1)} \right), \qquad \text{(VII 10)}$$

$$L_I^{(1)} = \sum_{j=1,\,k=1}^{J,\,K} C_{I-1,j,k}^{(1)} \left(\phi_{I-1,j,k}^{(t-1)} - \phi_{I,j,k}^{(t-1)} \right), \text{ etc.} \qquad \text{(VII 11)}$$

3. Neutron conductance coefficients (C):

a) $$C_{i,j,k}^{(1)} = r_{i,j,k}^{(1)} + r_{i,j,k-1}^{(1)} \qquad \text{(VII 12)}$$

b) $$C_{i,j,k}^{(2)} = r_{i,j,k}^{(2)} + r_{i,j,k-1}^{(2)}. \qquad \text{(VII 13)}$$

4. Neutron depletion coefficient (B):

$$B_{i,j,k} = E_{i,j,k} + C_{i,j,k}^{(1)} + C_{i,j,k}^{(2)} + C_{i,j,k}^{(3)} + C_{i-1,j,k}^{(1)} + C_{i,j-1,k}^{(2)} + C_{i,j,k-1}^{(3)}. \qquad \text{(VII 14)}$$

5. Flux Change ($\epsilon_{\phi}^{(t)}$):

$$\epsilon_{\phi}^{(t)} = \sum_{i=1,\,j=1,\,k=1}^{I,\,J,\,K} \left(\phi_{i,j,k}^{t} - \phi_{i,j,k}^{t-1} \right)^2. \qquad \text{(VII 15)}$$

6. Error Ratio:

$$\epsilon_{\phi}^{(t)} \Big/ \epsilon_{\varphi}^{(t-C_2)}, \qquad \text{(VII 16)}$$

where C_2 is the frequency print regulator.

7. Convergence criterion:

a) Primary

$$\epsilon_{\phi}^{(t)} < \delta_{p}^{(g)}, \qquad \text{(VII 1)}$$

where δ_{p}^{g} is predetermined convergence criterion.

b) Secondary

$$\epsilon_{p}^{(t)} < C_1\% \text{ of } \epsilon_{\phi}^{(1)}, \qquad \text{(VII 2)}$$

where C_1 is predetermined percentage (expressed as a decimal).

c) Tertiary

$$t \geq t_{max}, \qquad \text{(VII 3)}$$

where t_{max} is the upper limit on the number of allowable sweeps.

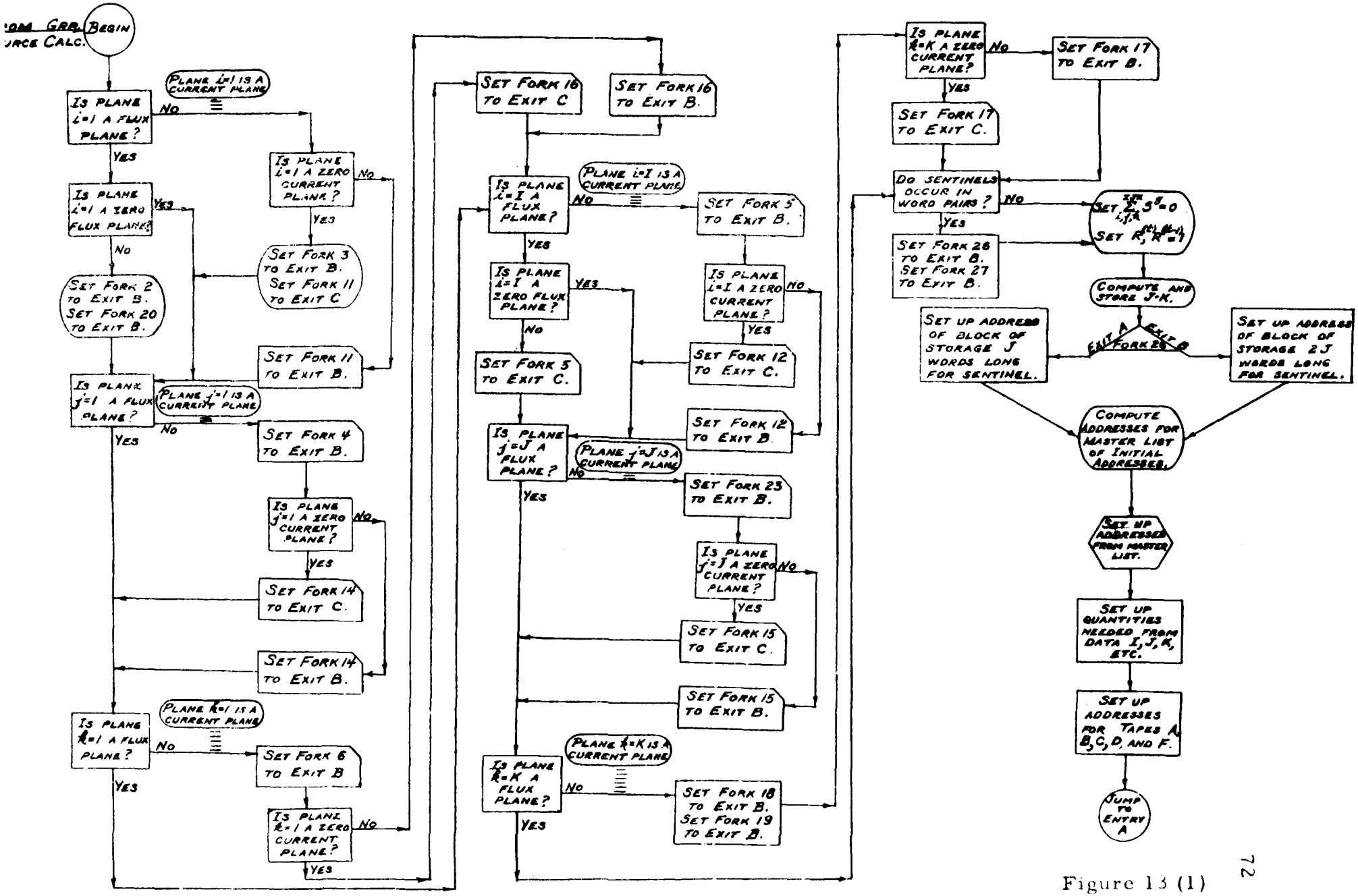

FROM GRR. SOURCE CALC.
BEGIN
PLANE i=I IS A CURRENT PLANE
IS PLANE i=I A FLUX PLANE?
NO
YES
IS PLANE i=I A ZERO CURRENT PLANE?
NO
YES
IS PLANE i=I A ZERO FLUX PLANE?
YES
NO
SET FORK 3 TO EXIT B. SET FORK 11 TO EXIT C
SET FORK 2 TO EXIT B. SET FORK 20 TO EXIT B.
SET FORK 11 TO EXIT B.
IS PLANE j=1 A FLUX PLANE?
PLANE j=1 IS A CURRENT PLANE
NO
YES
SET FORK 4 TO EXIT B.
IS PLANE j=1 A ZERO CURRENT PLANE?
NO
YES
SET FORK 14 TO EXIT C.
SET FORK 14 TO EXIT B.
PLANE k=1 IS A CURRENT PLANE
IS PLANE k=1 A FLUX PLANE?
NO
YES
SET FORK 6 TO EXIT B
IS PLANE k=1 A ZERO CURRENT PLANE?
NO
YES
SET FORK 16 TO EXIT C
SET FORK 16 TO EXIT B.
PLANE i=I IS A CURRENT PLANE
IS PLANE i=I A FLUX PLANE?
NO
YES
SET FORK 5 TO EXIT B.
IS PLANE i=I A ZERO FLUX PLANE?
YES
NO
IS PLANE i=I A ZERO CURRENT PLANE?
NO
YES
SET FORK 5 TO EXIT C.
SET FORK 12 TO EXIT C.
SET FORK 12 TO EXIT B.
IS PLANE j=J A FLUX PLANE?
PLANE j=J IS A CURRENT PLANE
NO
YES
SET FORK 23 TO EXIT B.
IS PLANE j=J A ZERO CURRENT PLANE?
NO
YES
SET FORK 15 TO EXIT C.
SET FORK 15 TO EXIT B.
PLANE k=K IS A CURRENT PLANE
IS PLANE k=K A FLUX PLANE?
NO
YES
SET FORK 18 TO EXIT B. SET FORK 19 TO EXIT B.
IS PLANE k=K A ZERO CURRENT PLANE?
NO
YES
SET FORK 17 TO EXIT B.
SET FORK 17 TO EXIT C.
DO SENTINELS OCCUR IN WORD PAIRS?
NO
YES
SET FORK 26 TO EXIT B. SET FORK 27 TO EXIT B.
SET
SET
COMPUTE AND STORE J·K.
EXIT A
EXIT B
FORK 26
SET UP ADDRESS OF BLOCK OF STORAGE J WORDS LONG FOR SENTINEL.
SET UP ADDRESS OF BLOCK OF STORAGE 2J WORDS LONG FOR SENTINEL.
COMPUTE ADDRESSES FOR MASTER LIST OF INITIAL ADDRESSES.
SET UP ADDRESSES FROM MASTER LIST.
SET UP QUANTITIES NEEDED FROM DATA I, J, K, ETC.
SET UP ADDRESSES FOR TAPES A, B, C, D, AND F.
JUMP TO ENTRY A

Figure 13 (1)

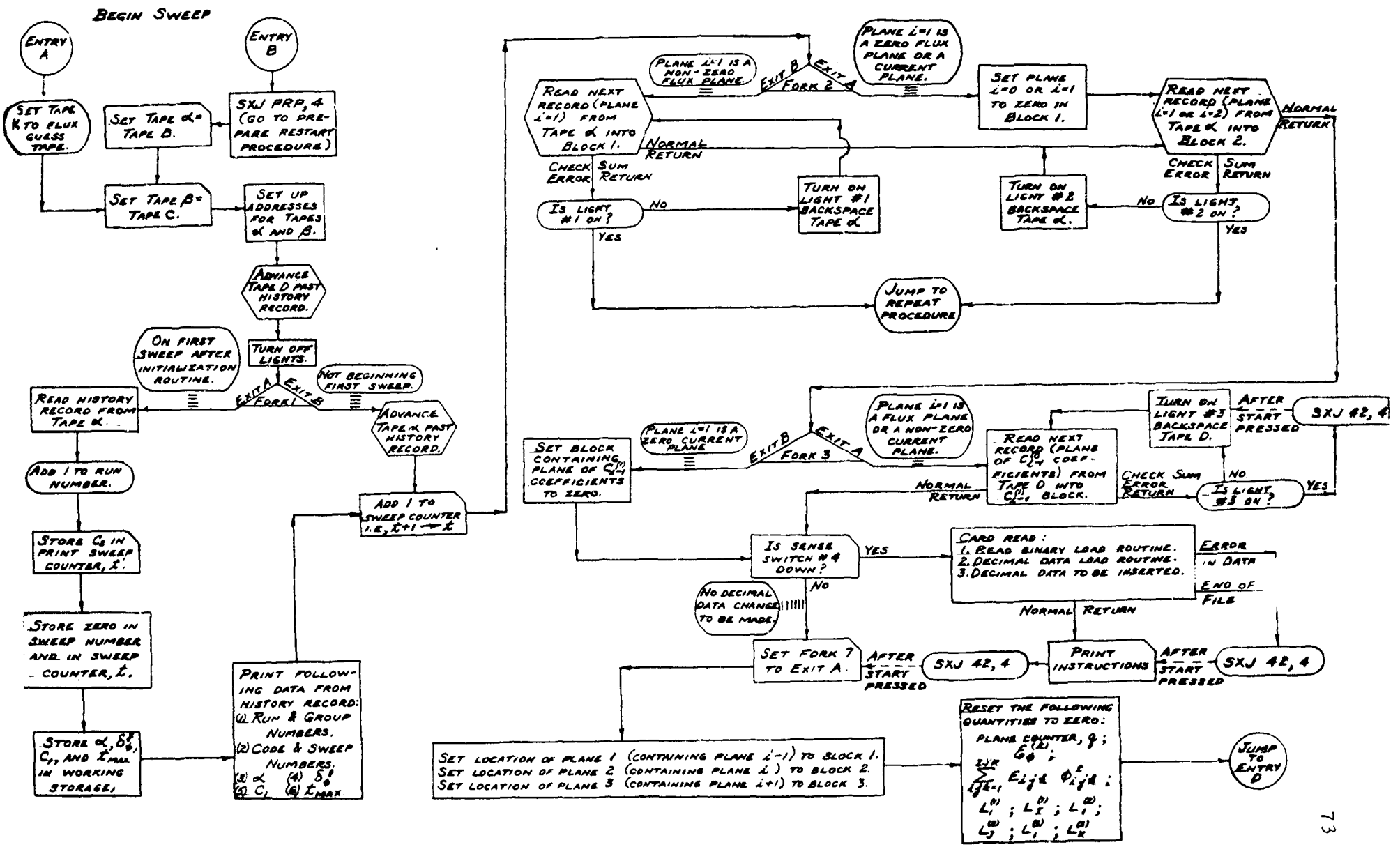

BEGIN SWEEP
GROUP SWEEP
ENTRY A
ENTRY B
SET TAPE K TO FLUX GUESS TAPE.
SET TAPE α = TAPE B.
SXJ PRP, 4 (GO TO PREPARE RESTART PROCEDURE)
SET TAPE β = TAPE C.
SET UP ADDRESSES FOR TAPES α AND β.
ADVANCE TAPE D PAST HISTORY RECORD.
TURN OFF LIGHTS.
ON FIRST SWEEP AFTER INITIALIZATION ROUTINE.
NOT BEGINNING FIRST SWEEP.
EXIT A
EXIT B
FORK 1
READ HISTORY RECORD FROM TAPE α.
ADVANCE TAPE α PAST HISTORY RECORD.
ADD 1 TO RUN NUMBER.
ADD 1 TO SWEEP COUNTER I.E., i+1 → i
STORE C_0 IN PRINT SWEEP COUNTER, i'.
STORE ZERO IN SWEEP NUMBER AND IN SWEEP COUNTER, i.
STORE α, δ_ϕ^i, $C_{i'}$, AND i_{MAX} IN WORKING STORAGE.
PRINT FOLLOWING DATA FROM HISTORY RECORD:
(1) RUN & GROUP NUMBERS.
(2) CODE & SWEEP NUMBERS.
(3) α (4) δ_ϕ^i
(5) $C_{i'}$ (6) i_{MAX}.
PLANE i=1 IS A NON-ZERO FLUX PLANE.
PLANE i=1 IS A ZERO FLUX PLANE OR A CURRENT PLANE.
EXIT B
EXIT A
FORK 2
READ NEXT RECORD (PLANE i=1) FROM TAPE α INTO BLOCK 1.
SET PLANE i=0 OR i=1 TO ZERO IN BLOCK 1.
READ NEXT RECORD (PLANE i=1 OR i=2) FROM TAPE α INTO BLOCK 2.
NORMAL RETURN
CHECK ERROR
SUM RETURN
IS LIGHT #1 ON?
NO
YES
TURN ON LIGHT #1 BACKSPACE TAPE α.
TURN ON LIGHT #2 BACKSPACE TAPE α.
IS LIGHT #2 ON?
JUMP TO REPEAT PROCEDURE
PLANE i=1 IS A FLUX PLANE OR A NON-ZERO CURRENT PLANE.
PLANE i=1 IS A ZERO CURRENT PLANE.
EXIT B
EXIT A
FORK 3
SET BLOCK CONTAINING PLANE OF $C_{i-1}^{(n)}$ COEFFICIENTS TO ZERO.
READ NEXT RECORD (PLANE OF $C_{i-1}^{(n)}$ COEFFICIENTS) FROM TAPE D INTO $C_{i-1}^{(n)}$ BLOCK.
TURN ON LIGHT #3 BACKSPACE TAPE D.
AFTER START PRESSED
SXJ 42, 4
CHECK SUM ERROR RETURN
NO.
YES
IS LIGHT #3 ON?
NORMAL RETURN
IS SENSE SWITCH #4 DOWN?
YES
NO
NO DECIMAL DATA CHANGE TO BE MADE.
CARD READ:
1. READ BINARY LOAD ROUTINE.
2. DECIMAL DATA LOAD ROUTINE.
3. DECIMAL DATA TO BE INSERTED.
ERROR IN DATA
END OF FILE
NORMAL RETURN
SET FORK 7 TO EXIT A.
AFTER START PRESSED
SXJ 42, 4
PRINT INSTRUCTIONS
AFTER START PRESSED
SXJ 42, 4
SET LOCATION OF PLANE 1 (CONTAINING PLANE i'-1) TO BLOCK 1.
SET LOCATION OF PLANE 2 (CONTAINING PLANE i') TO BLOCK 2.
SET LOCATION OF PLANE 3 (CONTAINING PLANE i'+1) TO BLOCK 3.
RESET THE FOLLOWING QUANTITIES TO ZERO:
PLANE COUNTER, g;
$\mathcal{E}^{(2)}$;
$\sum_{j,k=1}^{I,K} E_{ijk}\,\phi_{ijk}^{\mathcal{E}}$;
$L_i^{(n)}$; $L_I^{(n)}$; $L_i^{(2)}$;
$L_J^{(2)}$; $L_i^{(3)}$; $L_K^{(3)}$
JUMP TO ENTRY D

Figure 13 (2)

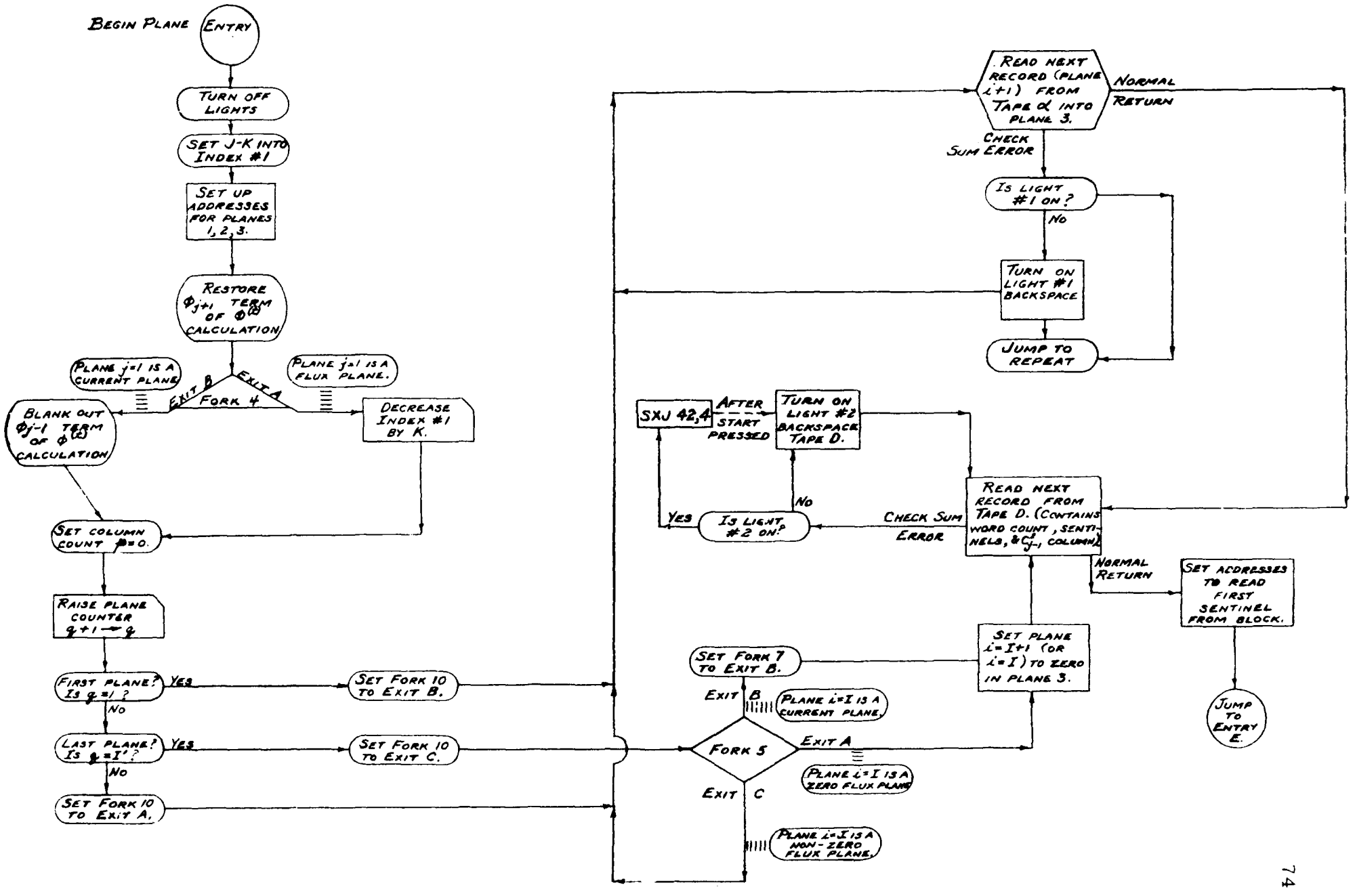

BEGIN PLANE ENTRY
TURN OFF LIGHTS
SET J-K INTO INDEX #1
SET UP ADDRESSES FOR PLANES 1, 2, 3.
RESTORE Φ_{j+1} TERM OF $\Phi^{(2)}$ CALCULATION
PLANE j=1 IS A CURRENT PLANE
PLANE j=1 IS A FLUX PLANE.
EXIT B
EXIT A
FORK 4
BLANK OUT Φ_{j-1} TERM OF $\Phi^{(2)}$ CALCULATION
DECREASE INDEX #1 BY K.
SET COLUMN COUNT #=0.
RAISE PLANE COUNTER g+1 → g
FIRST PLANE? Is g=1 ?
YES
NO
SET FORK 10 TO EXIT B.
LAST PLANE? Is g=I' ?
YES
NO
SET FORK 10 TO EXIT C.
SET FORK 10 TO EXIT A.
READ NEXT RECORD (PLANE i+1) FROM TAPE α INTO PLANE 3.
NORMAL RETURN
CHECK SUM ERROR
IS LIGHT #1 ON?
NO
TURN ON LIGHT #1 BACKSPACE
JUMP TO REPEAT
SXJ 42,4
AFTER START PRESSED
TURN ON LIGHT #2 BACKSPACE TAPE D.
YES
IS LIGHT #2 ON?
NO
CHECK SUM ERROR
READ NEXT RECORD FROM TAPE D. (CONTAINS WORD COUNT, SENTINELS, & C^{j}_{j-1} COLUMN)
NORMAL RETURN
SET ADDRESSES TO READ FIRST SENTINEL FROM BLOCK.
JUMP TO ENTRY E.
SET PLANE i=I+1 (OR i=I) TO ZERO IN PLANE 3.
SET FORK 7 TO EXIT B.
EXIT B
PLANE i=I IS A CURRENT PLANE.
FORK 5
EXIT A
PLANE i=I IS A ZERO FLUX PLANE
EXIT C
PLANE i=I IS A NON-ZERO FLUX PLANE.

Figure 13 (3)

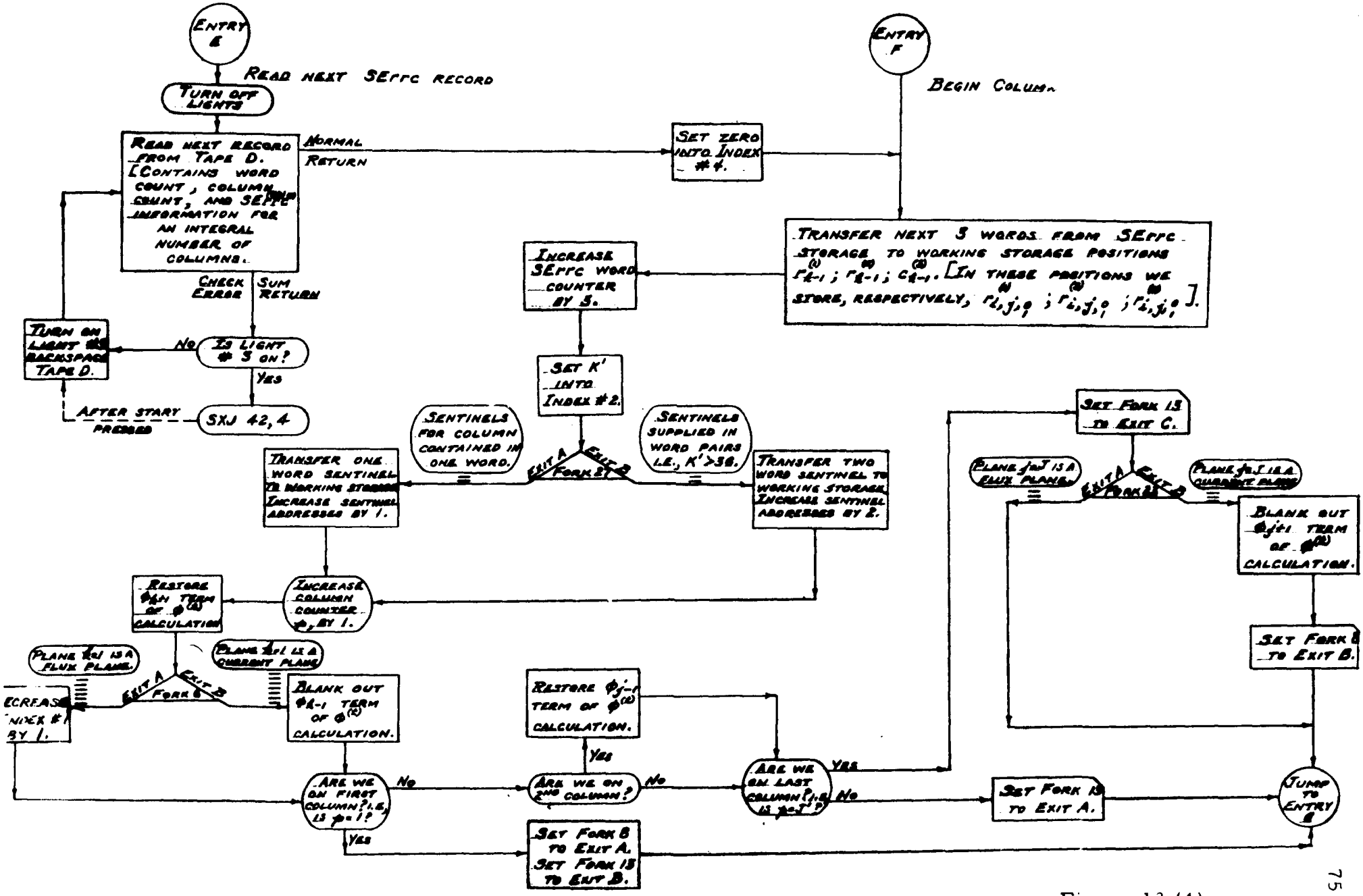

GROUP SWEEP
ENTRY E
READ NEXT SEffc RECORD
TURN OFF LIGHTS
READ NEXT RECORD FROM TAPE D. [CONTAINS WORD COUNT, COLUMN COUNT, AND SEffc INFORMATION FOR AN INTEGRAL NUMBER OF COLUMNS.
NORMAL RETURN
CHECK ERROR
SUM RETURN
TURN ON LIGHT #3 BACKSPACE TAPE D.
NO
IS LIGHT #3 ON?
YES
AFTER START PRESSED
SXJ 42,4
SET ZERO INTO INDEX #4.
ENTRY F
BEGIN COLUMN
TRANSFER NEXT 3 WORDS FROM SEffc STORAGE TO WORKING STORAGE POSITIONS
INCREASE SEffc WORD COUNTER BY 3.
SET K' INTO INDEX #2.
SENTINELS FOR COLUMN CONTAINED IN ONE WORD.
SENTINELS SUPPLIED IN WORD PAIRS I.E., K'>38.
EXIT A
EXIT B
FORK 21
TRANSFER ONE WORD SENTINEL TO WORKING STORAGE INCREASE SENTINEL ADDRESSES BY 1.
TRANSFER TWO WORD SENTINEL TO WORKING STORAGE INCREASE SENTINEL ADDRESSES BY 2.
SET FORK 13 TO EXIT C.
PLANE j OF J IS A FLUX PLANE.
PLANE j OF J IS A CURRENT PLANE
BLANK OUT φj+1 TERM OF φ(2) CALCULATION.
SET FORK 8 TO EXIT B.
RESTORE φi TERM OF φ(2) CALCULATION
INCREASE COLUMN COUNTER φi BY 1.
PLANE i OF I IS A FLUX PLANE.
PLANE i OF I IS A CURRENT PLANE
DECREASE INDEX #1 BY 1.
BLANK OUT φi-1 TERM OF φ(2) CALCULATION.
RESTORE φj-1 TERM OF φ(2) CALCULATION.
ARE WE ON FIRST COLUMN? I.E., IS φi=1?
ARE WE ON 2ND COLUMN?
ARE WE ON LAST COLUMN? I.E., IS φi=I?
SET FORK 13 TO EXIT A.
JUMP TO ENTRY E
SET FORK 8 TO EXIT A. SET FORK 13 TO EXIT B.

Figure 13 (4)

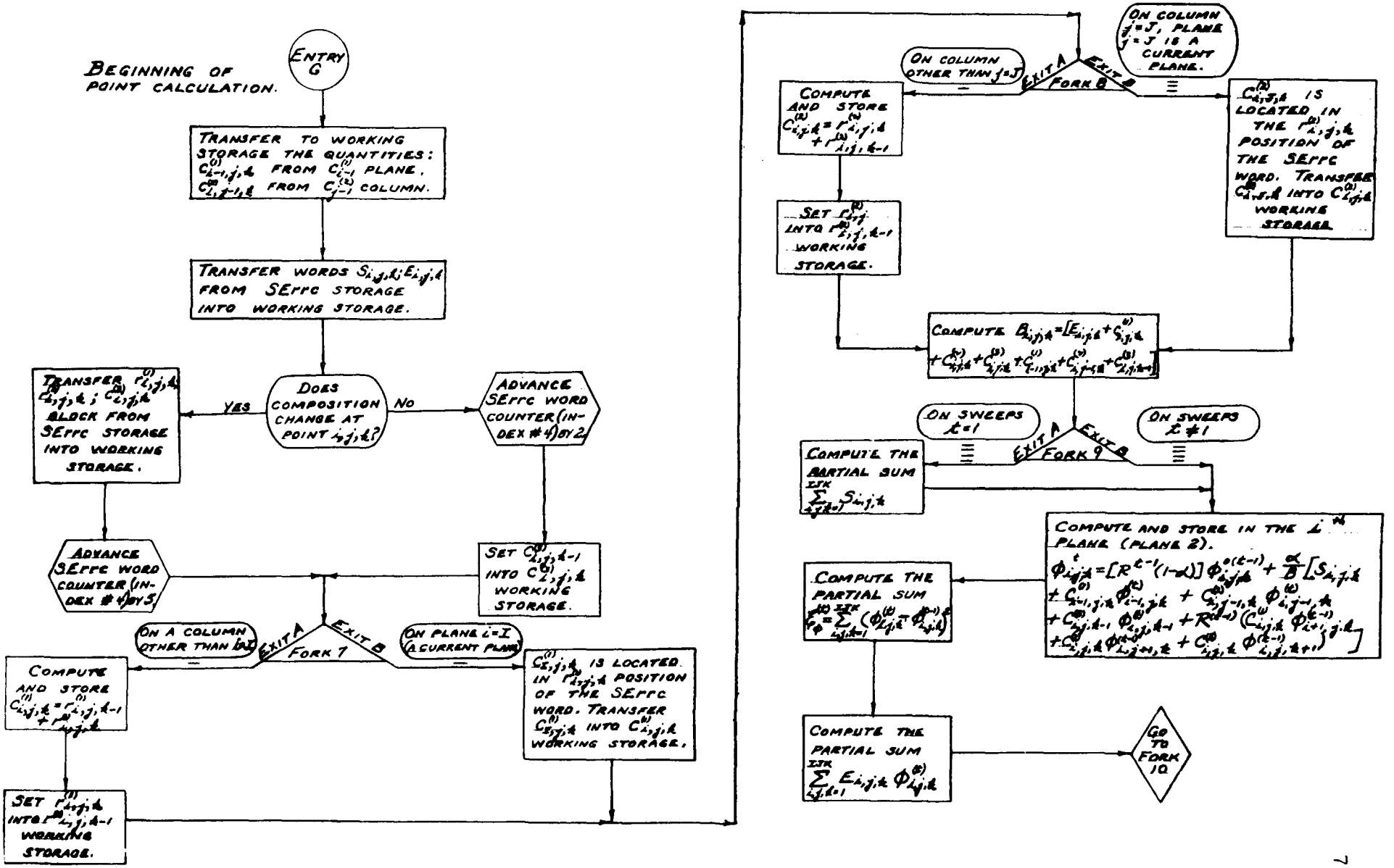

BEGINNING OF POINT CALCULATION.
ENTRY G
TRANSFER TO WORKING STORAGE THE QUANTITIES: $C^{(1)}_{i-1,j,k}$ FROM $C^{(1)}_{i-1}$ PLANE. $C^{(2)}_{i,j-1,k}$ FROM $C^{(2)}_{j-1}$ COLUMN.
TRANSFER WORDS $S_{i,j,k}; E_{i,j,k}$ FROM SErrc STORAGE INTO WORKING STORAGE.
DOES COMPOSITION CHANGE AT POINT i,j,k?
YES
NO
TRANSFER $r^{(1)}_{i,j,k}$, $C^{(2)}_{i,j,k}$; $C^{(3)}_{i,j,k}$ BLOCK FROM SErrc STORAGE INTO WORKING STORAGE.
ADVANCE SErrc WORD COUNTER (INDEX #4) BY 2
ADVANCE SErrc WORD COUNTER (INDEX #4) BY 5
SET $C^{(1)}_{i,j,k-1}$ INTO $C^{(1)}_{i,j,k}$ WORKING STORAGE.
FORK 7
EXIT A
EXIT B
ON A COLUMN OTHER THAN $j=J$
ON PLANE $i=I$ (A CURRENT PLANE)
COMPUTE AND STORE $C^{(1)}_{i,j,k} = r^{(1)}_{i,j,k-1} + r^{(1)}_{i,j,k}$
SET $r^{(1)}_{i,j,k}$ INTO $r^{(1)}_{i,j,k-1}$ WORKING STORAGE.
$C^{(1)}_{E,j,k}$ IS LOCATED IN $r^{(1)}_{i,j,k}$ POSITION OF THE SErrc WORD. TRANSFER $C^{(1)}_{E,j,k}$ INTO $C^{(1)}_{i,j,k}$ WORKING STORAGE.
ON COLUMN OTHER THAN $j=J$
ON COLUMN $j=J$, PLANE $j=J$ IS A CURRENT PLANE.
FORK 8
EXIT A
EXIT B
COMPUTE AND STORE $C^{(2)}_{i,j,k} = r^{(2)}_{i,j,k} + r^{(2)}_{i,j,k-1}$
SET $r^{(2)}_{i,j,k}$ INTO $r^{(2)}_{i,j,k-1}$ WORKING STORAGE.
$C^{(2)}_{i,J,k}$ IS LOCATED IN THE $r^{(2)}_{i,j,k}$ POSITION OF THE SErrc WORD. TRANSFER $C^{(2)}_{i,J,k}$ INTO $C^{(2)}_{i,j,k}$ WORKING STORAGE
COMPUTE $B_{i,j,k} = [E_{i,j,k} + C^{(1)}_{i,j,k} + C^{(2)}_{i,j,k} + C^{(3)}_{i,j,k} + C^{(1)}_{i-1,j,k} + C^{(2)}_{i,j-1,k} + C^{(3)}_{i,j,k-1}]$
ON SWEEPS $k=1$
ON SWEEPS $k \neq 1$
FORK 9
EXIT A
EXIT B
COMPUTE THE PARTIAL SUM $\sum_{i,j,k=1}^{IJK} S_{i,j,k}$
COMPUTE AND STORE IN THE L^{th} PLANE (PLANE 2).
$\phi^{t}_{i,j,k} = [R^{t-1}(1-\alpha)]\,\phi^{o(t-1)}_{i,j,k} + \frac{\alpha}{B}[S_{i,j,k} + C^{(1)}_{i-1,j,k}\phi^{(1)}_{i-1,j,k} + C^{(2)}_{i,j-1,k}\phi^{(2)}_{i,j-1,k} + C^{(3)}_{i,j,k-1}\phi^{(3)}_{i,j,k-1} + R^{t(t-1)}(C^{(1)}_{i,j,k}\phi^{(t-1)}_{i+1,j,k} + C^{(2)}_{i,j,k}\phi^{(t-1)}_{i,j+1,k} + C^{(3)}_{i,j,k}\phi^{(t-1)}_{i,j,k+1})]$
COMPUTE THE PARTIAL SUM $\phi^{(t)} = \sum_{i,j,k=1}^{IJK} (\phi^{(t)}_{i,j,k} - \phi^{(t-1)}_{i,j,k})$
COMPUTE THE PARTIAL SUM $\sum_{i,j,k=1}^{IJK} E_{i,j,k}\,\phi^{(t)}_{i,j,k}$
GO TO FORK 10

Figure 13 (5)

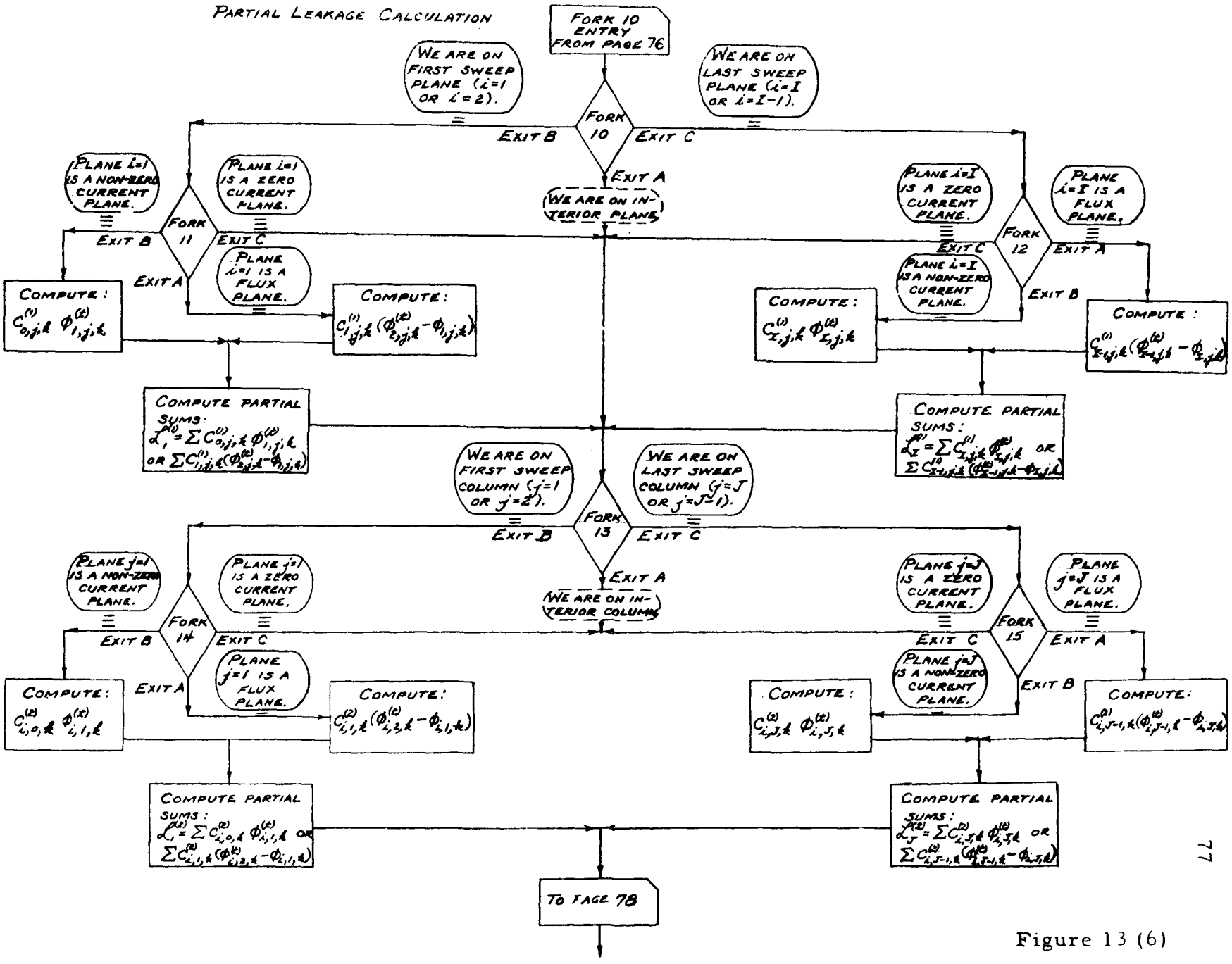

PARTIAL LEAKAGE CALCULATION
FORK 10 ENTRY FROM PAGE 76
FORK 10
EXIT A
EXIT B
EXIT C
WE ARE ON FIRST SWEEP PLANE (i=1 OR i=2).
WE ARE ON LAST SWEEP PLANE (i=I OR i=I-1).
WE ARE ON INTERIOR PLANE
FORK 11
PLANE i=1 IS A NON-ZERO CURRENT PLANE.
PLANE i=1 IS A ZERO CURRENT PLANE.
PLANE i=1 IS A FLUX PLANE.
COMPUTE: $C^{(1)}_{0,j,k}\,\Phi^{(2)}_{1,j,k}$
COMPUTE: $C^{(1)}_{1,j,k}(\Phi^{(2)}_{2,j,k}-\Phi_{1,j,k})$
FORK 12
PLANE i=I IS A ZERO CURRENT PLANE.
PLANE i=I IS A FLUX PLANE.
PLANE i=I IS A NON-ZERO CURRENT PLANE.
COMPUTE: $C^{(1)}_{I,j,k}\,\Phi^{(2)}_{I,j,k}$
COMPUTE: $C^{(1)}_{I-1,j,k}(\Phi^{(2)}_{I-1,j,k}-\Phi_{I,j,k})$
COMPUTE PARTIAL SUMS: $\mathcal{L}^{(1)}_{1}=\sum C^{(1)}_{0,j,k}\,\Phi^{(2)}_{1,j,k}$ OR $\sum C^{(1)}_{1,j,k}(\Phi^{(2)}_{2,j,k}-\Phi_{1,j,k})$
COMPUTE PARTIAL SUMS: $\mathcal{L}^{(1)}_{I}=\sum C^{(1)}_{I,j,k}\,\Phi^{(2)}_{I,j,k}$ OR $\sum C^{(1)}_{I-1,j,k}(\Phi^{(2)}_{I-1,j,k}-\Phi_{I,j,k})$
WE ARE ON FIRST SWEEP COLUMN (j=1 OR j=2).
WE ARE ON LAST SWEEP COLUMN (j=J OR j=J-1).
FORK 13
EXIT A
EXIT B
EXIT C
WE ARE ON INTERIOR COLUMN
FORK 14
PLANE j=1 IS A NON-ZERO CURRENT PLANE.
PLANE j=1 IS A ZERO CURRENT PLANE.
PLANE j=1 IS A FLUX PLANE.
COMPUTE: $C^{(2)}_{i,0,k}\,\Phi^{(2)}_{i,1,k}$
COMPUTE: $C^{(2)}_{i,1,k}(\Phi^{(2)}_{i,2,k}-\Phi_{i,1,k})$
FORK 15
PLANE j=J IS A ZERO CURRENT PLANE.
PLANE j=J IS A FLUX PLANE.
PLANE j=J IS A NON-ZERO CURRENT PLANE.
COMPUTE: $C^{(2)}_{i,J,k}\,\Phi^{(2)}_{i,J,k}$
COMPUTE: $C^{(2)}_{i,J-1,k}(\Phi^{(2)}_{i,J-1,k}-\Phi_{i,J,k})$
COMPUTE PARTIAL SUMS: $\mathcal{L}^{(2)}_{1}=\sum C^{(2)}_{i,0,k}\,\Phi^{(2)}_{i,1,k}$ OR $\sum C^{(2)}_{i,1,k}(\Phi^{(2)}_{i,2,k}-\Phi_{i,1,k})$
COMPUTE PARTIAL SUMS: $\mathcal{L}^{(2)}_{J}=\sum C^{(2)}_{i,J,k}\,\Phi^{(2)}_{i,J,k}$ OR $\sum C^{(2)}_{i,J-1,k}(\Phi^{(2)}_{i,J-1,k}-\Phi_{i,J,k})$
TO PAGE 78

Figure 13 (6)

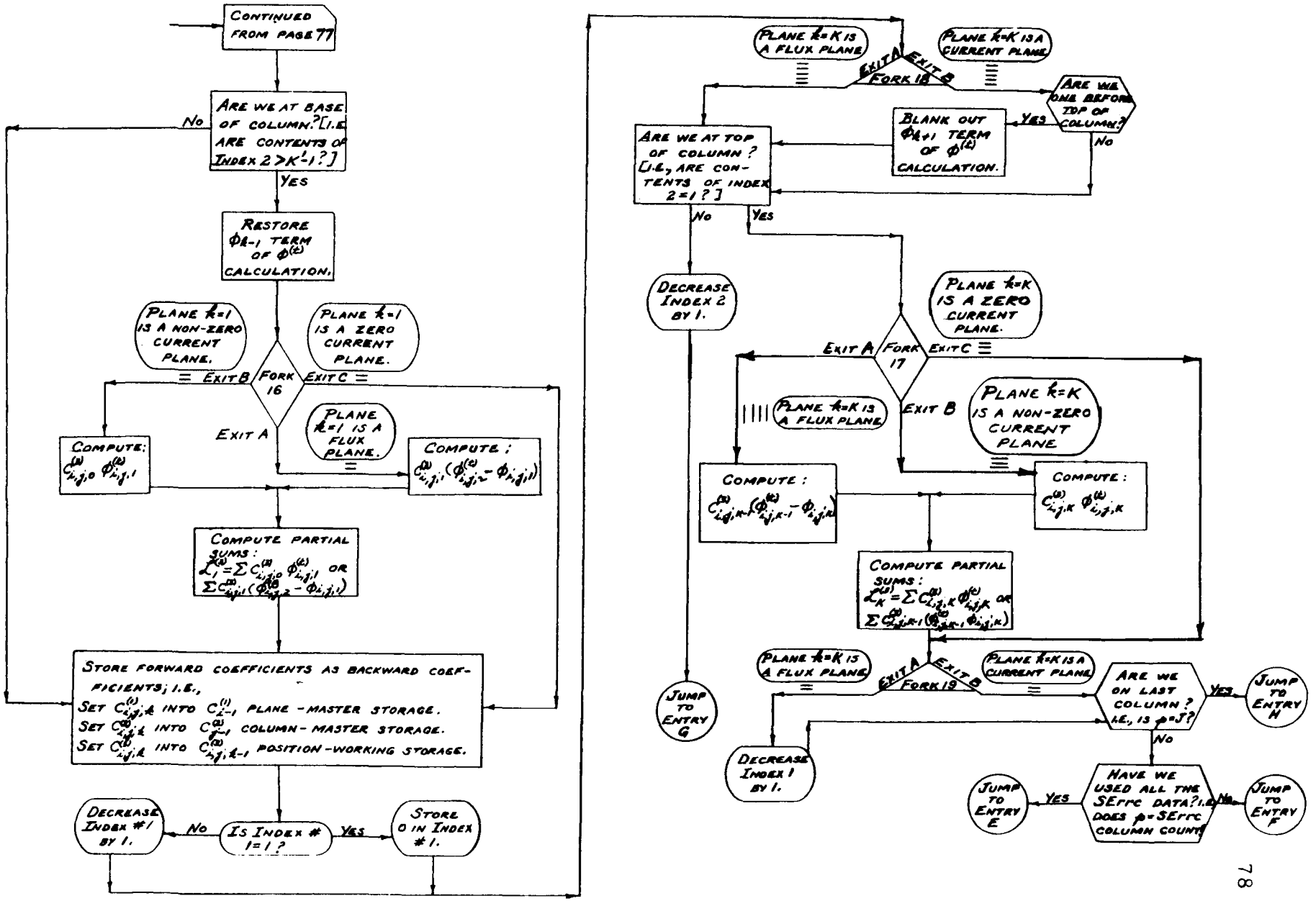

CONTINUED FROM PAGE 77
ARE WE AT BASE OF COLUMN? [I.E. ARE CONTENTS OF INDEX 2 > K'-1?]
No
YES
RESTORE Φ_{k-1} TERM OF $\Phi^{(z)}$ CALCULATION.
PLANE k=1 IS A NON-ZERO CURRENT PLANE.
PLANE k=1 IS A ZERO CURRENT PLANE.
FORK 16
EXIT B
EXIT C
EXIT A
PLANE k=1 IS A FLUX PLANE.
COMPUTE: $C^{(z)}_{i,j,0}\,\Phi^{(z)}_{i,j,1}$
COMPUTE: $C^{(z)}_{i,j,1}(\Phi^{(z)}_{i,j,2}-\Phi_{i,j,1})$
COMPUTE PARTIAL SUMS: $\mathcal{L}^{(z)}_1=\Sigma C^{(z)}_{i,j,0}\,\Phi^{(z)}_{i,j,1}$ OR $\Sigma C^{(z)}_{i,j,1}(\Phi^{(z)}_{i,j,2}-\Phi_{i,j,1})$
STORE FORWARD COEFFICIENTS AS BACKWARD COEFFICIENTS; I.E.,
SET $C^{(1)}_{i,j,k}$ INTO $C^{(1)}_{i,j,-1}$, PLANE-MASTER STORAGE.
SET $C^{(2)}_{i,j,k}$ INTO $C^{(2)}_{i,j,-1}$, COLUMN-MASTER STORAGE.
SET $C^{(2)}_{i,j,k}$ INTO $C^{(2)}_{i,j,k-1}$ POSITION-WORKING STORAGE.
DECREASE INDEX #1 BY 1.
No
IS INDEX # 1=1?
YES
STORE 0 IN INDEX #1.
PLANE k=K IS A FLUX PLANE
PLANE k=K IS A CURRENT PLANE
EXIT A
EXIT B
FORK 18
ARE WE ONE BEFORE TOP OF COLUMN?
YES
No
BLANK OUT Φ_{k+1} TERM OF $\Phi^{(z)}$ CALCULATION.
ARE WE AT TOP OF COLUMN? [I.E., ARE CONTENTS OF INDEX 2=1?]
No
YES
DECREASE INDEX 2 BY 1.
PLANE k=K IS A ZERO CURRENT PLANE.
EXIT A
FORK 17
EXIT C
EXIT B
PLANE k=K IS A FLUX PLANE
PLANE k=K IS A NON-ZERO CURRENT PLANE
COMPUTE: $C^{(z)}_{i,j,k-1}(\Phi^{(z)}_{i,j,k-1}-\Phi_{i,j,k})$
COMPUTE: $C^{(z)}_{i,j,k}\,\Phi^{(z)}_{i,j,k}$
COMPUTE PARTIAL SUMS: $\mathcal{L}^{(z)}_K=\Sigma C^{(z)}_{i,j,k}\,\Phi^{(z)}_{i,j,k}$ OR $\Sigma C^{(z)}_{i,j,k-1}(\Phi^{(z)}_{i,j,k-1}-\Phi_{i,j,k})$
PLANE k=K IS A FLUX PLANE
PLANE k=K IS A CURRENT PLANE
EXIT A
EXIT B
FORK 19
ARE WE ON LAST COLUMN? I.E., IS ϕ=J?
YES
JUMP TO ENTRY H
No
HAVE WE USED ALL THE SErrc DATA?, I.E., DOES ϕ=SErrc COLUMN COUNT?
No
JUMP TO ENTRY F
YES
JUMP TO ENTRY E
DECREASE INDEX 1 BY 1.
JUMP TO ENTRY G

Figure 13 (7)

GROUP SWEEP

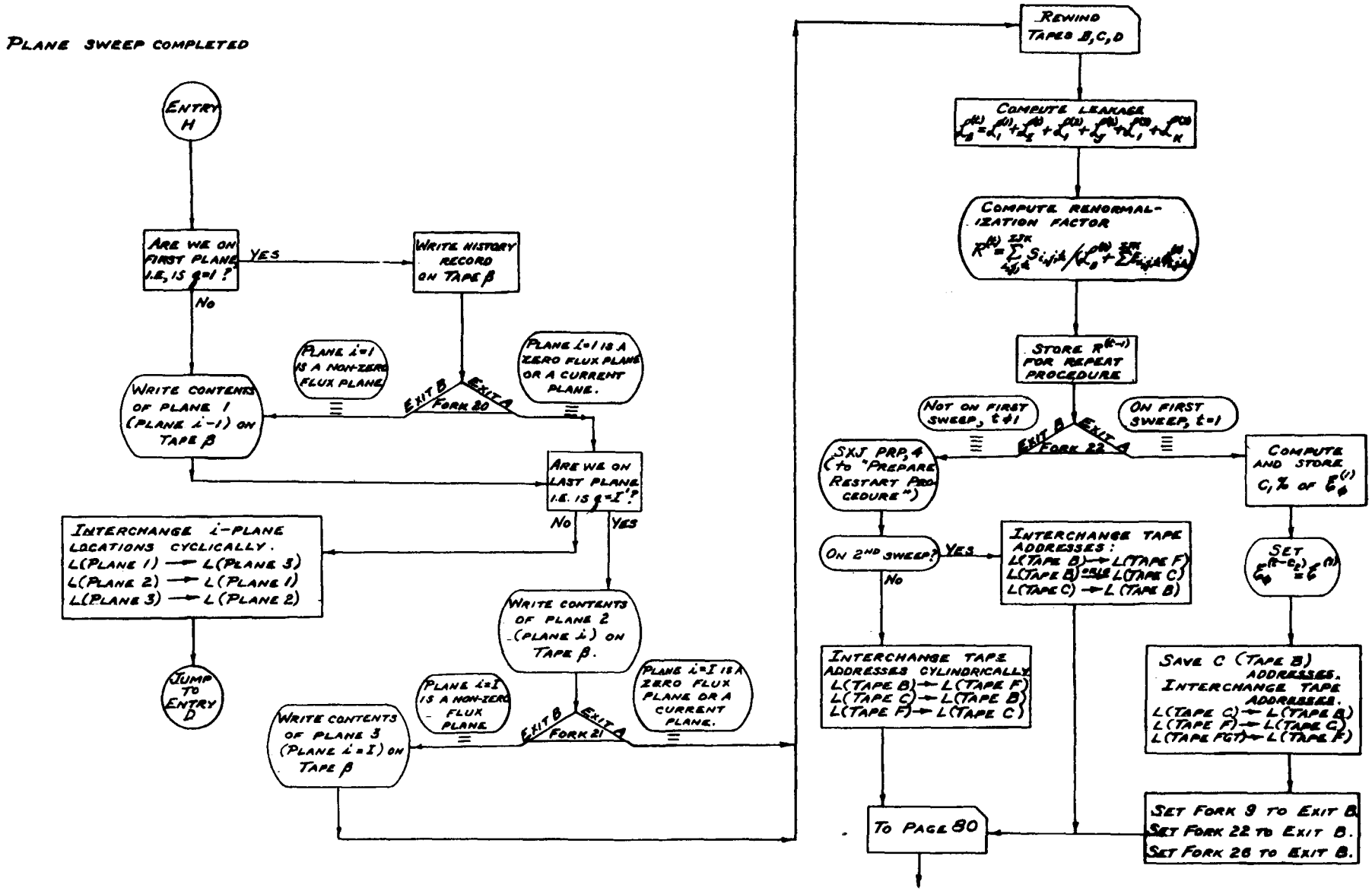


Figure 13 (8)

GROUP SWEEP

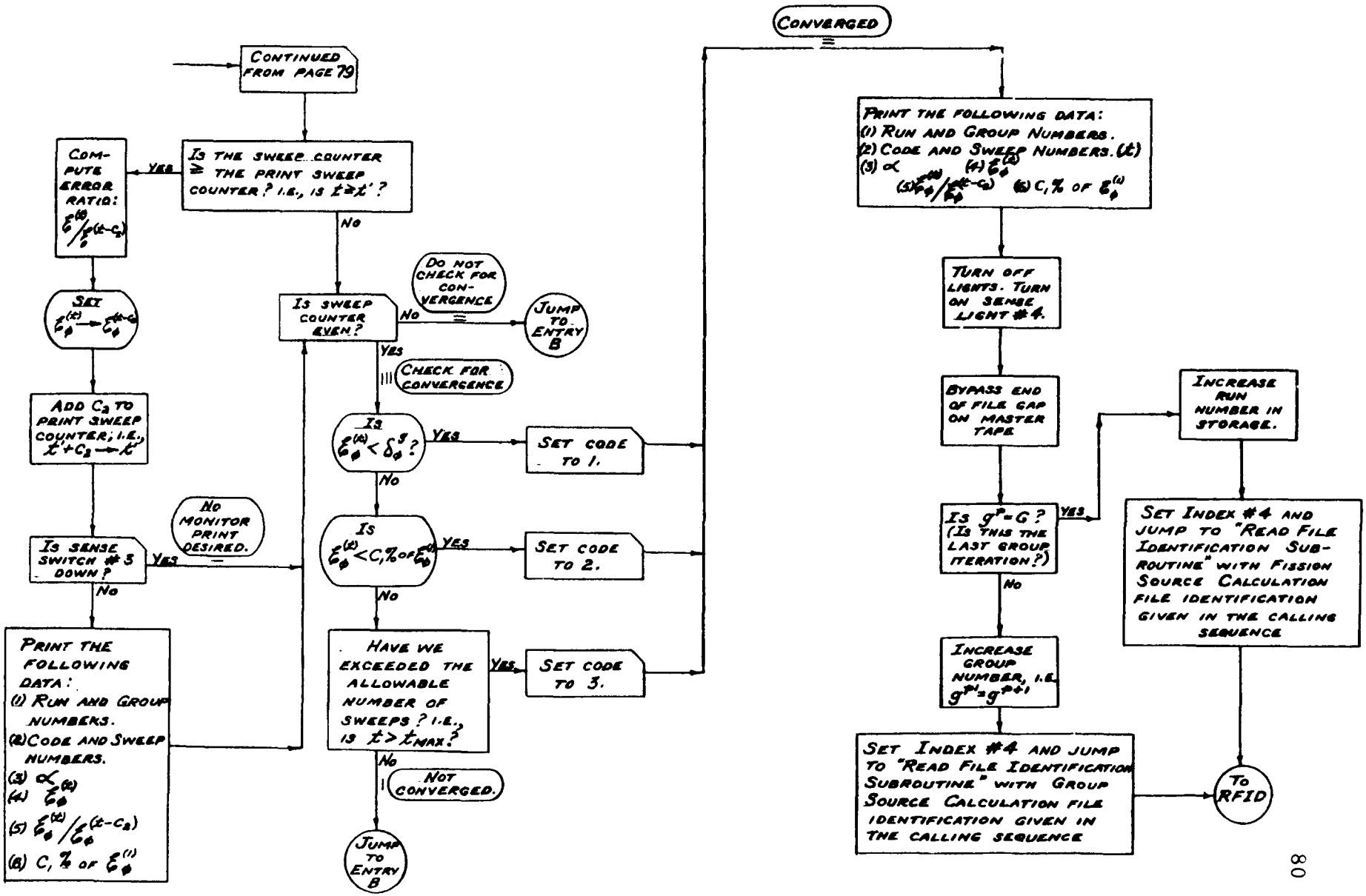


Figure 13 (9)

GROUP SWEEP

READ FILE IDENTIFICATION SUBROUTINE

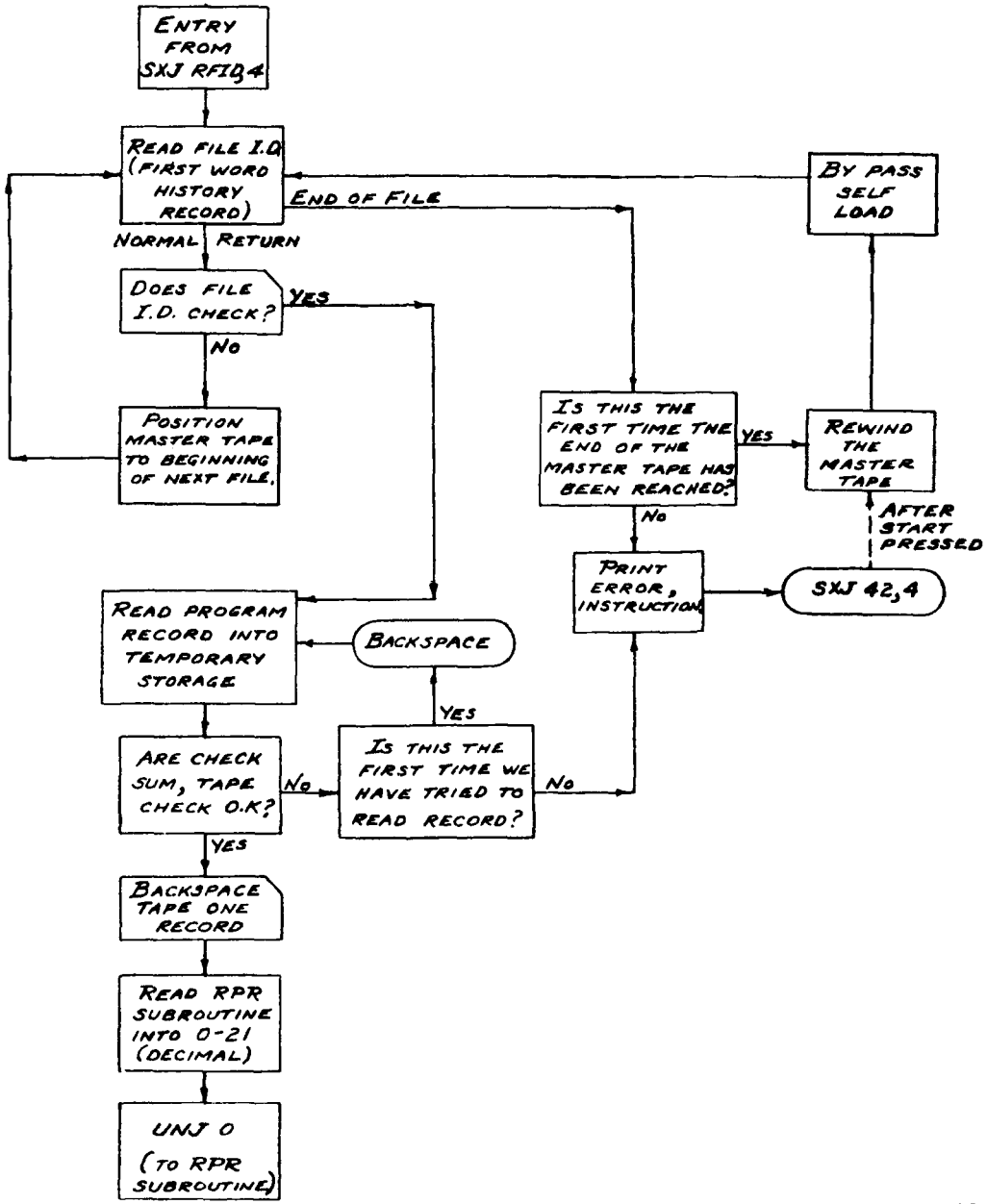


Figure 13 (10)

GROUP SWEEP

READ PROGRAM RECORD ROUTINE

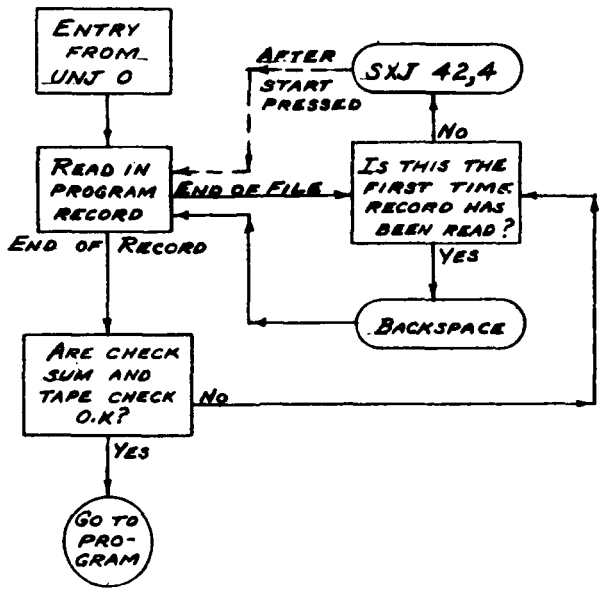


Figure 13 (11)

GROUP SWEEP

REPEAT PROCEDURE

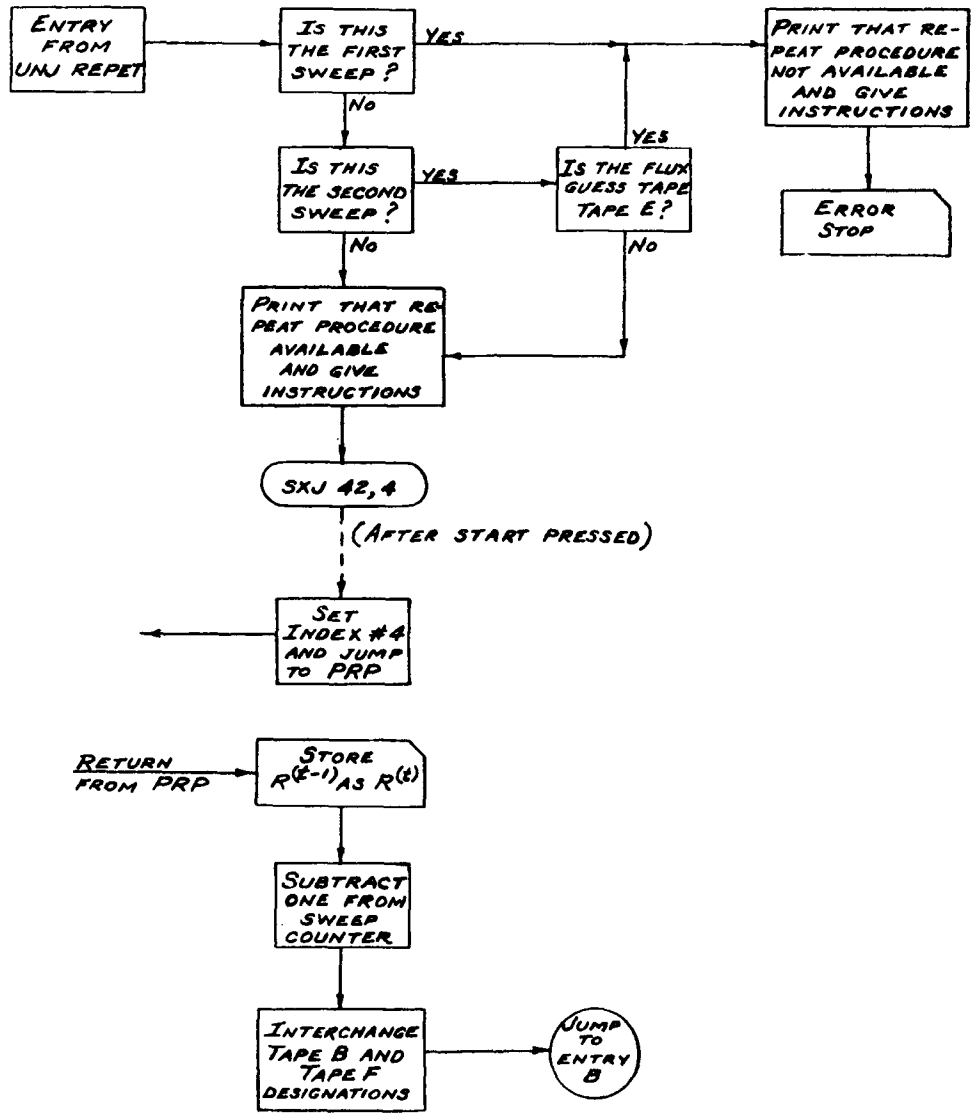


Figure 13 (12)

GROUP SWEEP

PREPARE RESTART PROCEDURE

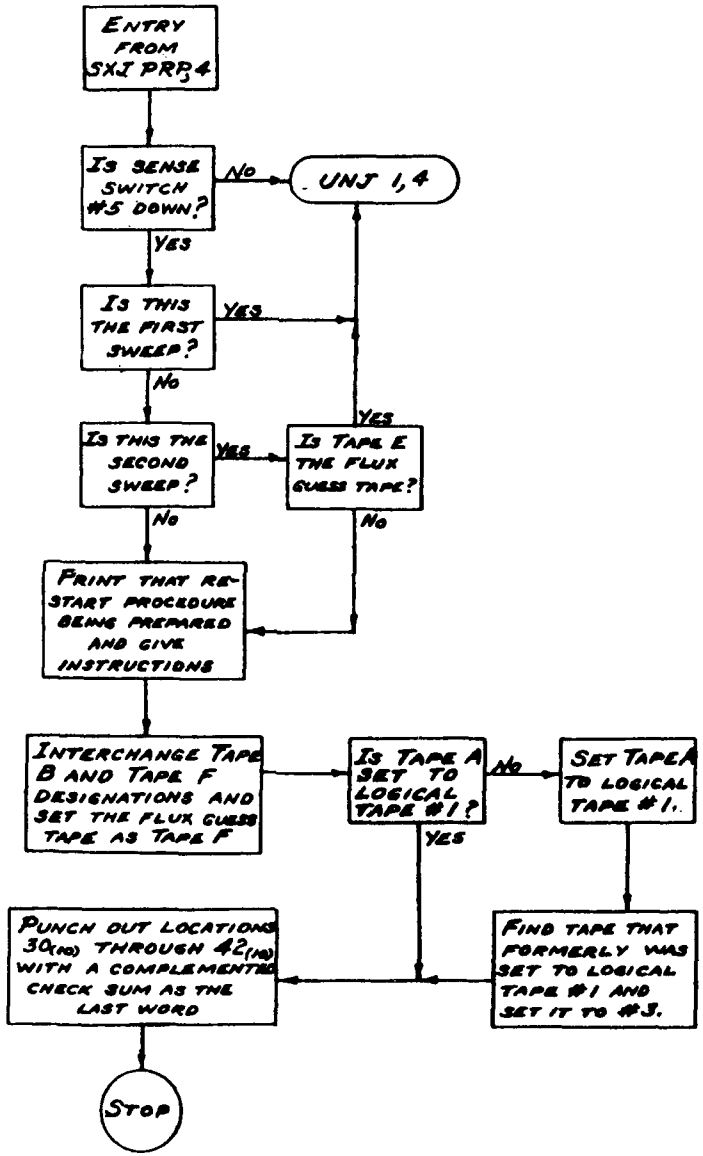


Figure 13 (13)

GROUP SWEEP

TAPE READ-WRITE SUBROUTINE

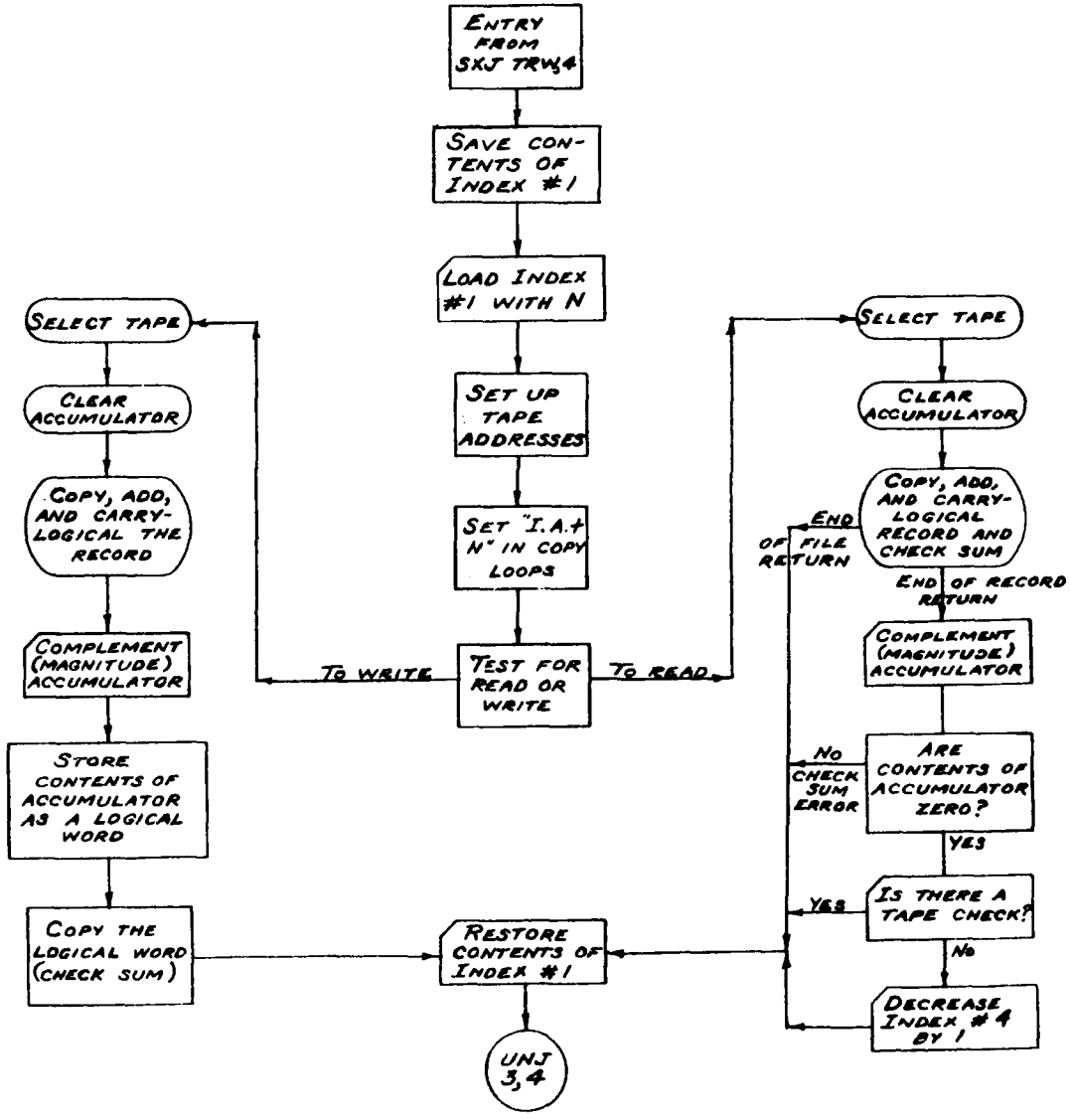


Figure 13 (14)

VIII. FISSION SOURCE PROGRAM *

DESCRIPTION OF PROGRAM

The TRIXY Fission Source Program computes a new fission source by an iterative procedure. If the fission source has not converged, tape E containing the iterated fission source and the flux iterations from the last run are prepared and turned over to the Group Source program, unless it is decided to stop the fission source iterations.

After tape E has been prepared, the program prints out several lines of information which include the criticality, maximum delta δ and its location within the mesh, the total fission source error, and the value of ω used for the next iteration. Convergence or non-convergence is indicated. The program stops, and the operator can go directly to the Edit program if instructed.

The Fission Source Program copies the iterated fluxes for group G onto the tape containing the iterated fluxes for the previous groups and the fission source iteration from the last run. Using $F'^{g}_{i,j,k}$ from the master tape and the iterated $\varphi^{g}_{i,j,k}$ from the last run, the following values are computed and written on a working tape

$$\delta^{g**}_{i,j,k} = F'^{g}_{i,j,k} \; \varphi^{g}_{i,j,k} \qquad \text{(VIII 1)}$$

$\delta^{g**}_{i,j,k}$ for the next higher group are computed, and $\delta^{(g-1)**}_{i,j,k}$ from the

*Fission Source Flow Charts (Figure 15)

previous run are read into the memory and added to these results. The sum is written on tape and the process repeated until the sum $\S^{G^{**}}_{i,j,k}$ has been formed. For each group, the program calculates

$$f^g = \sum^{F} \S^{g^{**}}_{i,j,k} \; / \; V_F \qquad \text{(VIII 2)}$$

and the sum

$$f = \sum_{g=g'}^{G} f^g. \qquad \text{(VIII 3)}$$

When all groups are finished, the normalized fission source $\left(\S^{(r-1)}_{i,j,k}\right)$ for the last run and the unnormalized fission source $\left(\S^{G^{**}}_{i,j,k}\right)$ for the present run are read into memory. The normalized source is computed:

$$\S^{(r)}_{i,j,k} = \S^{(r-1)}_{i,j,k} + \omega \left(\frac{\S^{G^{**}}_{i,j,k}}{f} - \S^{(r-1)}_{i,j,k} \right) \qquad \text{(VIII 4)}$$

The value for ω is read in from the card reader. As the new $\S$ is computed, the maximum delta $\S$ is found, the total fission source error is accumulated, and the convergence is checked. The new $\S$ is written on the new tape E, and the iterated fluxes from the previous run are transferred to this tape. The program prints the information mentioned earlier and stops. (See Figure 14).

```
END OF RUN  1          NOT CONVERGED
TAPE 2 IS THE NEW TAPE E.
DEPRESS SWITCH 6 IF EDIT IS DESIRED
PUSH START BUTTON TO CONTINUE


CRITICALITY                  1.1186704

TOTAL SOURCE ERROR           1121.6346

MAX DELTA S DURING RUN       134.94965
PLANE  3. COLUMN  2. POINT  2.

OMEGA                        1.0000000
```

Example of Printing by
FISSION SOURCE PROGRAM

Figure 14

FISSION SOURCE

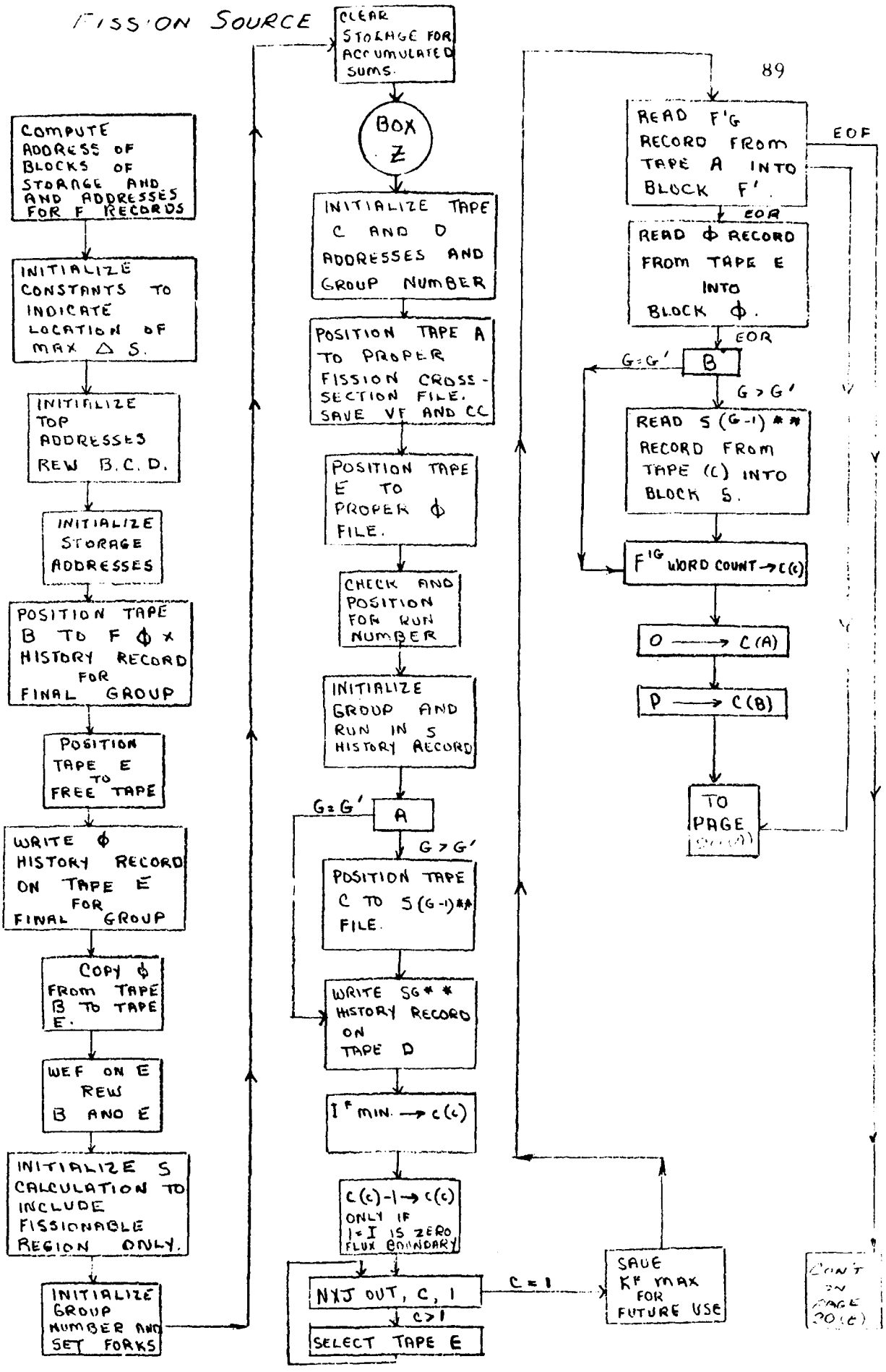


Figure 15 (1)

FISSION SOURCE

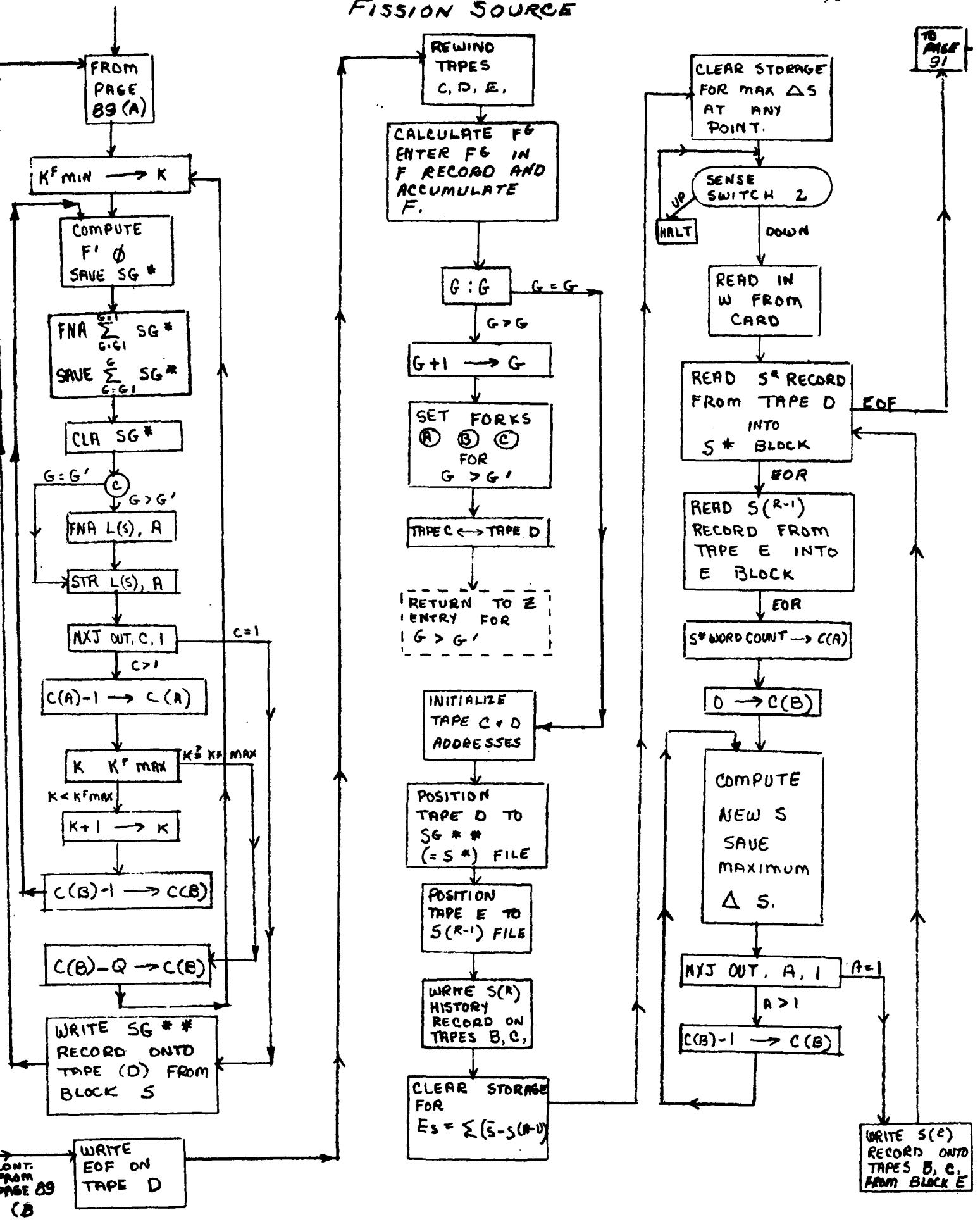


Figure 15 (2)

FISSION SOURCE

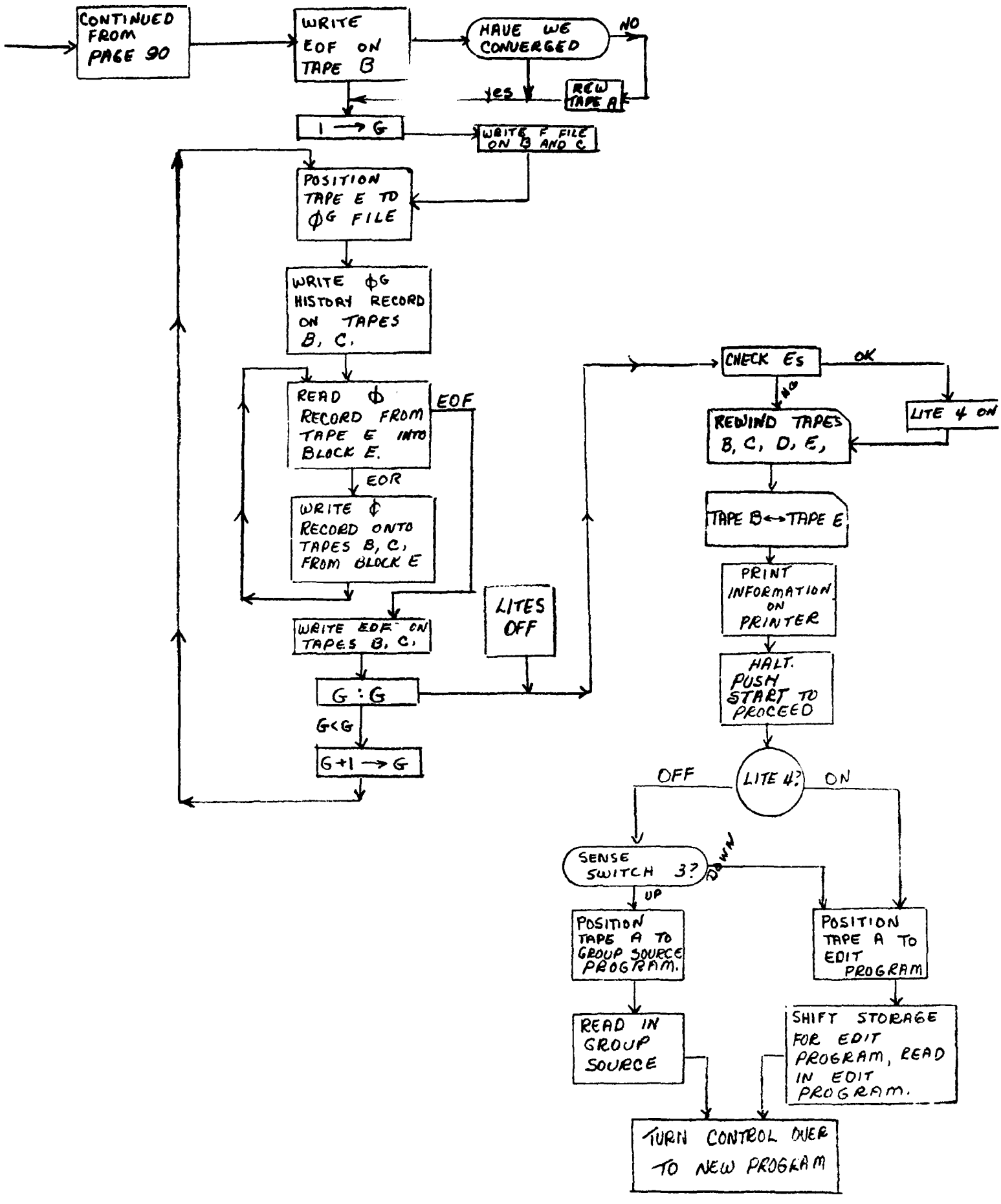


Figure 15 (3)

IX. EDIT PROGRAM*

PURPOSE

The Edit Program selects, sorts, and, where necessary, computes requested output. The following equations are used:

Power density at point i, j, k

$$P_{i,j,k} = \mathcal{S}_{i,j,k} / V''_{i,j,k} . \qquad \text{(IX 1)}$$

Power integral for column i, j

$$\mathcal{S}_{i,j} = \sum_{k=k^F_{min}}^{k^F_{max}} \mathcal{S}_{i,j,k} . \qquad \text{(IX 2)}$$

Power density for column i, j

$$P_{i,j} = \mathcal{S}_{i,j} \Big/ \sum_{k=k^F_{min}}^{k^F_{max}} V''_{i,j,k} . \qquad \text{(IX 3)}$$

Flux integral for composition region ℓ_n

$$\varphi^g_{\ell_n} = \sum_{i,j,k=1}^{I,J,K} \varphi^g_{i,j,k} \sum_{\ell,m,n=0}^{1} v_{i-\ell,\,j-m,\,k-n}\, \delta_{\ell_n,\,\ell_{i-\ell,\,j-m,\,k-n}} \qquad \text{(IX 4)}$$

*Edit Flow Charts (Figure 18)

where

$$\delta_{\ell_n, \ell_{i,j,k}} = 1 \text{ if } \ell_n = \ell_{i,j,k} \qquad \text{(IX 5)}$$

$$\delta_{\ell_n, \ell_{i,j,k}} = 0 \text{ if } \ell_n \neq \ell_{i,j,k}. \qquad \text{(IX 6)}$$

Average flux for composition region ℓ_n is

$$\bar{\phi}^g_{\ell_n} = \phi^g_{\ell_n} \Big/ \sum_{i,j,k=1}^{I,J,K} \sum_{\ell,m,n=0} v_{i-\ell, j-m, k-n}$$

$$\delta_{\ell_n, \ell_{i-\ell, j-m, k-n}}. \qquad \text{(IX 7)}$$

Output is divided into seven categories. (See Figure 16).

Data 1: $P_{i,j,k}$ for each point of column i, j, in order of increasing k, followed by $\delta_{i,j}$ and $P_{i,j}$.

Data 2: $\delta_{i,j}$ for each column of plane i, in order of increasing j.

Data 3: $P_{i,j}$ for each column of plane i, in order of increasing j.

Data 4: Number of fissions per source neutron (f^g) for all groups, followed by the total number of fission per source neutron.

Data 5: $\Phi^g_{\ell_n}$ for group g for each composition, in order of increasing index number ℓ_n.

Data 6: $\bar{\phi}^g_{\ell_n}$ for group g for each composition, in order of increasing index number ℓ_n.

Data 7: Flux, $\varphi^{g}_{i,j,k}$ for group g, for each point of column i, j, in order of increasing k.

The general flow of the Edit Program is as follows. A plane of $\mathcal{S}_{i,j,k}$ values and a plane of $V''_{i,j,k}$ values are read from tapes E and A respectively. Data 1, 2, and 3 are found for a column. Data 1 are written on tape B, Data 2 on tape C, and Data 3 on tape D. When the above procedure has been repeated for all columns and for all planes, the information on tapes C and D is transferred to tape B. Data 4 are then transferred from tape E to tape B. Mesh spacings and material indices for a plane are read from tape A, and the corresponding values of flux are read from tape E. The computation of $\Phi^{g}_{l_n}$ and of $\bar{\phi}^{g}_{l_n}$ is begun, and Data 7 are found for one column at a time and written on tape D. When Data 5 are completed they are written on tape B, and Data 6 are written on tape C. Data 5, 6, 7 are found for the remaining groups, and the information on tape C and D is transferred to tape B. The information on tape B is read, converted to BCD, and written on tape 10. Unrequested data is not computed, and when no sorting is necessary data are written directly on tape B. Each section of output has a two line heading containing the data type (q), the plane (i), the column (j), the group (g), the run (r), the problem number and the cross section data file number.

The output routine GL OUT2 (SHARE distribution number 84) is used.

It has been modified to set up a 72 character line and to use the SHARE standard printer board number 2. The output routine writes numbers in the format -YY XXXXXXXX. The decimal point is assumed to be to the left of the mantissa. Positive signs are indicated by blanks, and leading zeros in the characteristic are not printed. If the mantissa is zero, no characteristic is printed.

RE-EDIT

If the master tapes A and E are saved, a re-edit for information not requested during the edit may be performed.

Operating instructions:

Load and ready tape A on unit 1.

Load and ready tape E on unit 5.

Load and ready working tapes 2, 3, and 4.

Set up output tape 10.

Place deck in read hopper containing

RE-EDIT DECK (6 cards)

Edit Input Request cards

Do not put blank cards on the back of the deck.

CLEAR - LOAD CARDS.

Error stops:

$(04631)_8$ Illegal double punch on input cards. Correct card and start over.

$(11646)_8$ Tape error in reading history of edit code. Depress START to try again.

$(04563)_8$ ERROR IN EDIT CODE printed on-line. Depress START to try again.

EDIT REQUEST

There are seven different items of output which may be requested. These items are combinations of the following data, where the output data are given in mesh sweep order.

Data 1: for all i planes.

Power density ($P_{i,j,k}$) for each point, followed by the power integral ($\S_{i,j}$) and the power density ($P_{i,j}$) of the column.

Data 1a: for selected columns.

Same as Data 1.

Data 2: for all i planes.

Power integral ($\S_{i,j}$) for every column.

Data 3: for all i planes.

Power density for every column.

Data 4: Number of fissions per source neutron (f^g), for all groups followed by the total number of fissions per source neutron.

Data 5: for all groups.

Flux integral ($\phi^g_{l_n}$) for each composition, in order of increasing l_n.

Data 6: for all groups.

Average flux for each composition in order of increasing l_n.

Data 7: for group g for all i planes.

Flux ($\varphi^g_{i,j,k}$) for each point.

Data 7a: for group g for selected columns.

Same as data 7.

The following table indicates which data are included under the various items of output. Improper combinations of items are also indicated.

	DATA									Excludes Items						
Item	1	1A	2	3	4	5	6	7	7A	A1	A2	A3	A4	A(4+g)	B	C_g
A1	X		X	X							X				X	
A2			X	X						X					X	
A3					X											
A4						X	X									
A(4+g)								X								X
B(i, j)		X								X	X					
C_g(i, j)									X					X		

Figure 16

Requests for items A1 through A(4+g) are submitted in the form of an octal word, WORD I[1]. WORD I is first represented as a binary number by placing a one in the box under each requested item and a zero in the box under each item not needed. All the boxes must be filled regardless of the number of groups in the problem. Each set of three binary digits represents one octal as shown below.

First 3 binary digits	Subsequent octal digits
000 = +0	0
001 = +1	1
010 = +2	2
011 = +3	3
100 = -0	4
101 = -1	5
110 = -2	6
111 = -3	7

If none of the items [A1 through A(4+g)] is requested, +000000000000 must be submitted for WORD I.

If item Cg(i, j) is requested, the group to which the request applies must be submitted in the form of an octal word, WORD II. WORD II, like WORD I.

[1] See Edit Input sheet

is first represented as a binary number and then converted to octal. If item Cg(i, j) is not requested, +000000000000 must be submitted for WORD II.

Item B(i, j) is requested by specifying the plane and columns for which the data are desired. Input data consist of sets of three consecutive words (i, j', j"). Each set selects all the columns in plane i between columns j' and j" inclusive. If only one column is desired, j" = j' must still be specified. If two or more sets are desired for a single plane, the sets must be listed in order of increasing j and may not overlap. Sets must be listed in order of increasing plane i. The last set must be followed by a zero. If item B(i, j) is not requested, a zero must be supplied as input.

Item Cg(i, j) is requested in the same manner as item B(i, j). The sets of (i, j', j") follow the zero after the B(i, j) sets. The last set must be followed by a zero. If item Cg(i, j) is not requested, a zero must be supplied as input. For example, if neither item B(i, j) nor item Cg(i, j) is requested, the third edit request input card contains (.3252, 0, 0,). If the selected columns for fluxes under item Cg(i, j) are the same as those selected under item B(i, j), a one may be put in the box under SEN in WORD II, and a zero substituted for the sets of (i, j', j')'s for item Cg(i, j).

```
TRIXY
DATA 1 PLANE    1 COLUMN    1 GROUP      RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
         K     POWER DENSITY
         1          00000000
         2          00000000
         3          00000000
POWER INTEGRAL        00000000 POWER DENSITY        00000000


DATA 1 PLANE    1 COLUMN    2 GROUP      RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
         K     POWER DENSITY
         1          28726732
         2          23390913
         3          11649406
POWER INTEGRAL      3 20427648 POWER DENSITY        21789491


DATA 1 PLANE    1 COLUMN    3 GROUP      RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
         K     POWER DENSITY
         1        1 16321599
         2        1 13479241
         3        1 10534715
POWER INTEGRAL      4 12612843 POWER DENSITY      1 13453699


DATA 1 PLANE    2 COLUMN    1 GROUP      RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
         K     POWER DENSITY
         1          28726732
         2          23390912
         3          11649405
POWER INTEGRAL      3 20427647 POWER DENSITY        21789490


DATA 1 PLANE    2 COLUMN    2 GROUP      RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
         K     POWER DENSITY
         1          43367021
         2          35583599
         3          22139617
POWER INTEGRAL      4 22423051 POWER DENSITY        34168459


DATA 1 PLANE    3 COLUMN    3 GROUP      RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
         K     POWER DENSITY
         1        1 16299149
         2        1 13569056
         3        1 10191957
POWER INTEGRAL      4 62846738 POWER DENSITY      1 13407304


DATA 4 PLANE       COLUMN       GROUP      RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
  G     NO FISSIONS PER SOURCE NEUTRON
  1                00000000
  2                13335234
  3                32139173
                   45474407
```

Figure 17 (1)

```
DATA 5 PLANE      COLUMN      GROUP   1 RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
  INDEX       FLUX INTEGRAL
      1          6 23154808
   1001          6 11387804
   1002          4 45632656

DATA 5 PLANE      COLUMN      GROUP   2 RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
  INDEX       FLUX INTEGRAL
      1          6 17502494
   1001          5 93046648
   1002          4 32566869

DATA 5 PLANE      COLUMN      GROUP   3 RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
  INDEX       FLUX INTEGRAL
      1          5 42182665
   1001          5 31505944
   1002          3 11430565

DATA 6 PLANE      COLUMN      GROUP   1 RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
  INDEX       AVERAGE FLUX
      1          1 88208790
   1001          1 30987223
   1002          1 45632656

DATA 6 PLANE      COLUMN      GROUP   2 RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
  INDEX       AVERAGE FLUX
      1          1 66676165
   1001          1 25318815
   1002          1 32566869

DATA 6 PLANE      COLUMN      GROUP   3 RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
  INDEX       AVERAGE FLUX
      1          1 16069587
   1001            85730459
   1002            11430565

DATA 7 PLANE   1 COLUMN   2 GROUP   1 RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
         K        FLUX
         1       1 79071158
         2       1 63674468
         3       1 22435395
         4         00000000

DATA 7 PLANE   1 COLUMN   3 GROUP   1 RUN  1
PROBLEM NUMBER     10 CROSS SECTION DATA FILE NUMBER      1
         K        FLUX
         1       2 15247520
         2       2 12542932
         3       1 50514745
         4         00000000
```

Figure 17 (2)

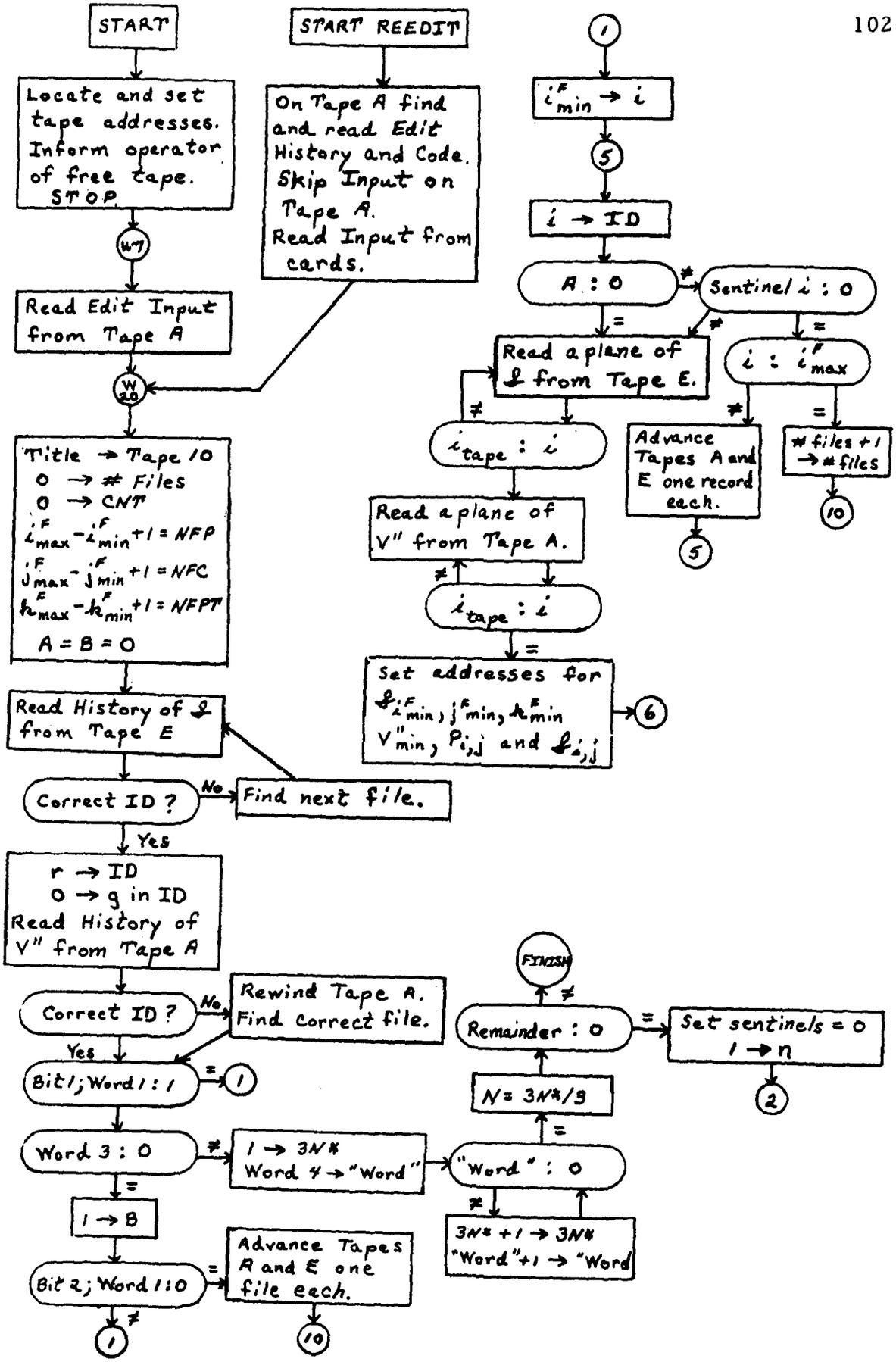

START
START REEDIT
Locate and set tape addresses. Inform operator of free tape. STOP
On Tape A find and read Edit History and Code. Skip Input on Tape A. Read Input from cards.
W7
Read Edit Input from Tape A
W 20
Title → Tape 10
0 → # Files
0 → CNT
$i^F_{max} - i^F_{min} + 1 = NFP$
$j^F_{max} - j^F_{min} + 1 = NFC$
$k^F_{max} - k^F_{min} + 1 = NFPT$
A = B = 0
Read History of ℓ from Tape E
Correct ID ?
No
Find next file.
Yes
r → ID
0 → q in ID
Read History of V'' from Tape A
Correct ID ?
No
Rewind Tape A. Find Correct file.
Yes
Bit 1; Word 1 : 1
1
Word 3 : 0
≠
1 → 3N*
Word 4 → "Word"
=
1 → B
Bit 2; Word 1 : 0
=
Advance Tapes A and E one file each.
10
1
≠
1
$i^F_{min} \rightarrow i$
5
i → ID
A : 0
≠
Sentinel i : 0
=
Read a plane of ℓ from Tape E.
≠
$i_{tape} : i$
≠
Read a plane of V'' from Tape A.
≠
$i_{tape} : i$
=
Set addresses for $\ell_{i^F_{min}, j^F_{min}, k^F_{min}}$ V''_{min}, $P_{i,j}$ and $\ell_{i,j}$
6
=
Advance Tapes A and E one record each.
5
$i : i^F_{max}$
≠
=
files + 1 → # files
10
FINISH
≠
Remainder : 0
=
Set sentinels = 0
1 → n
2
N = 3N*/3
=
"Word" : 0
≠
3N* + 1 → 3N*
"Word" + 1 → "Word"

Figure 18 (1)

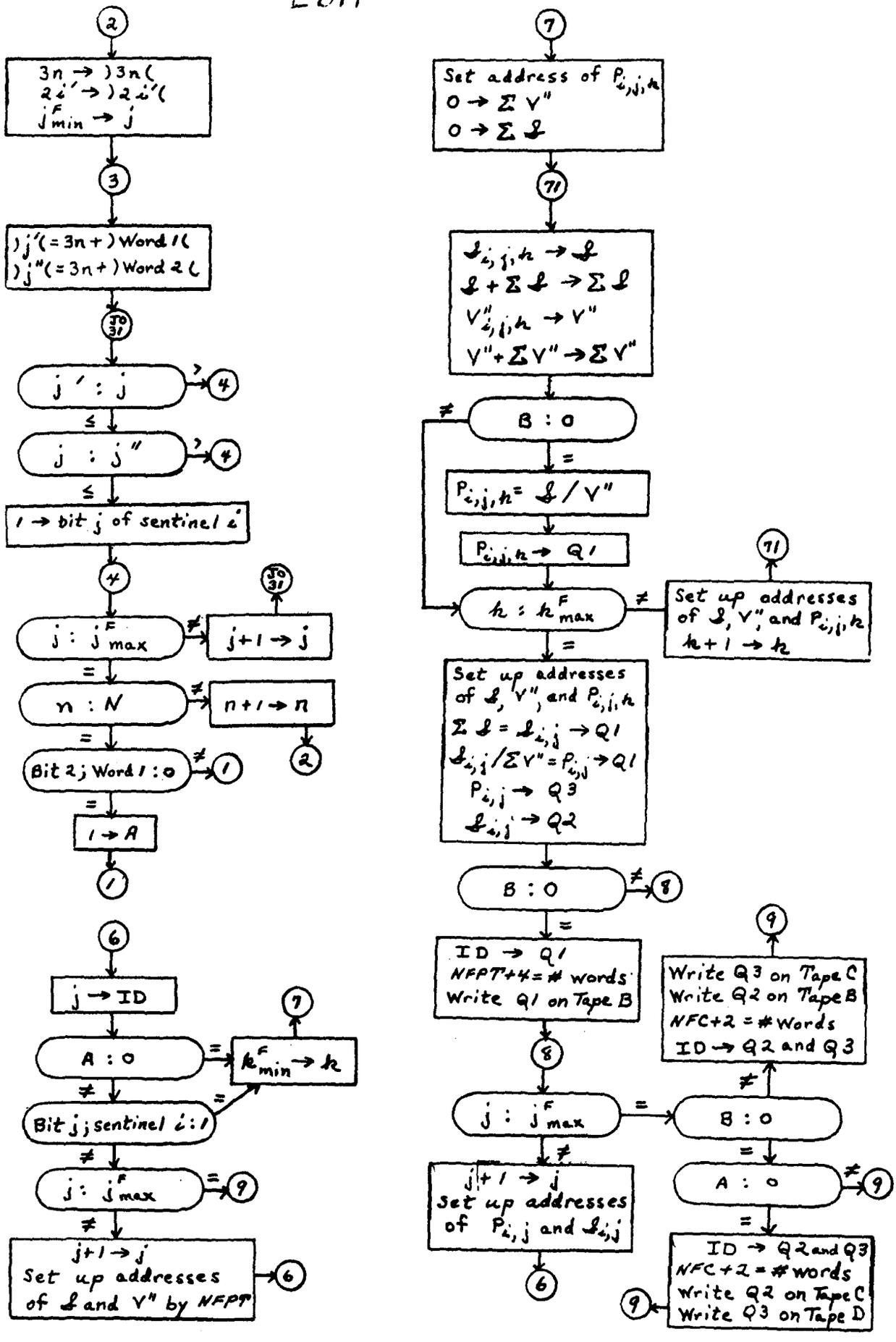

2
3n →)3n(
2i' →)2i'(
j^F_min → j
3
)j'(= 3n +) Word 1 (
)j''(= 3n +) Word 2 (
TO 31
j' : j
>
4
≤
j : j''
>
4
≤
1 → bit j of sentinel i'
4
j : j^F_max
≠
j+1 → j
TO 31
=
n : N
≠
n+1 → n
2
=
Bit 2j Word 1 : 0
≠
1
=
1 → A
1
6
j → ID
A : 0
=
k^F_min → k
7
≠
Bit j; sentinel i : 1
=
≠
j : j^F_max
=
9
≠
j+1 → j
Set up addresses
of ℓ and V'' by NFPP
6
7
Set address of P_i,j,k
0 → Σ V''
0 → Σ ℓ
71
ℓ_i,j,k → ℓ
ℓ + Σ ℓ → Σ ℓ
V''_i,j,k → V''
V'' + Σ V'' → Σ V''
≠
B : 0
=
P_i,j,k = ℓ / V''
P_i,j,k → Q1
k : k^F_max
≠
Set up addresses
of ℓ, V'', and P_i,j,k
k+1 → k
71
=
Set up addresses
of ℓ, V'', and P_i,j,k
Σ ℓ = ℓ_i,j → Q1
ℓ_i,j / Σ V'' = P_i,j → Q1
P_i,j → Q3
ℓ_i,j → Q2
B : 0
≠
8
=
ID → Q1
NFPT+4 = # words
Write Q1 on Tape B
8
j : j^F_max
=
B : 0
≠
j+1 → j
Set up addresses
of P_i,j and ℓ_i,j
6
=
A : 0
≠
9
=
ID → Q2 and Q3
NFC+2 = # words
Write Q2 on Tape C
Write Q3 on Tape D
9
9
Write Q3 on Tape C
Write Q2 on Tape B
NFC+2 = # Words
ID → Q2 and Q3

Figure 18 (2)

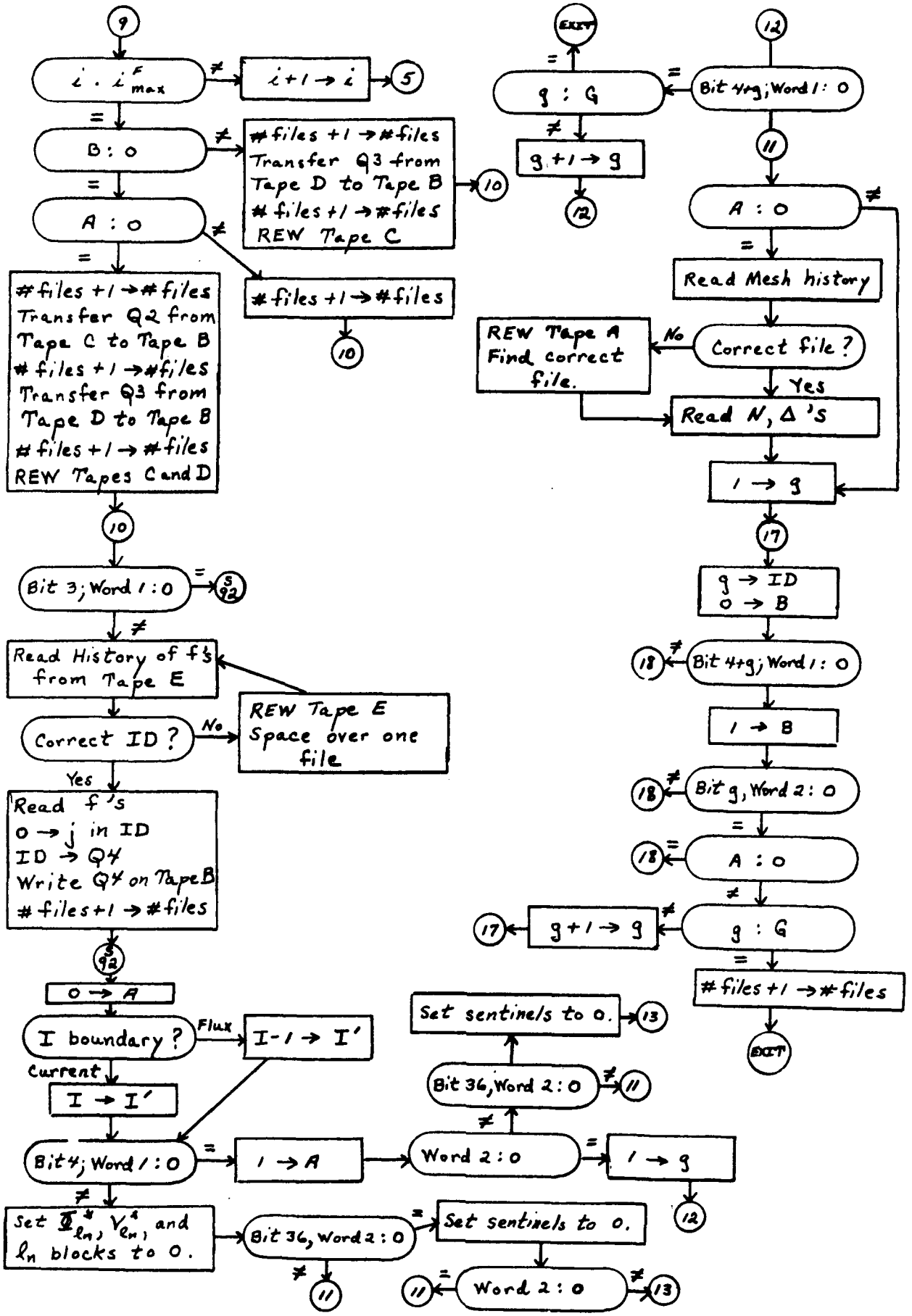

9
i . i^F_max
≠
i+1 → i
5
=
B : 0
≠
#files +1 → #files
Transfer Q3 from
Tape D to Tape B
#files +1 → #files
REW Tape C
10
=
A : 0
≠
=
#files +1 → #files
Transfer Q2 from
Tape C to Tape B
#files +1 → #files
Transfer Q3 from
Tape D to Tape B
#files +1 → #files
REW Tapes C and D
#files +1 → #files
10
10
Bit 3; Word 1 : 0
=
S
92
≠
Read History of f's
from Tape E
Correct ID ?
No
REW Tape E
Space over one
file
Yes
Read f's
0 → j in ID
ID → Q4
Write Q4 on Tape B
#files+1 → #files
S
92
0 → A
I boundary ?
Flux
I-1 → I'
Current
I → I'
Bit 4; Word 1 : 0
=
1 → A
≠
Set I^4_ln, V^4_ln, and
l_n blocks to 0.
Bit 36, Word 2 : 0
=
Set sentinels to 0.
≠
11
11
=
Word 2 : 0
≠
13
Set sentinels to 0.
13
Bit 36, Word 2 : 0
≠
11
≠
Word 2 : 0
=
1 → g
12
EXIT
=
g : G
≠
g+1 → g
12
12
Bit 4+g; Word 1 : 0
=
11
A : 0
≠
=
Read Mesh history
REW Tape A
Find correct
file.
No
Correct file ?
Yes
Read N, Δ's
1 → g
17
g → ID
0 → B
18
≠
Bit 4+g; Word 1 : 0
1 → B
18
≠
Bit g, Word 2 : 0
=
18
=
A : 0
≠
17
g+1 → g
≠
g : G
=
#files +1 → #files
EXIT

Figure 18 (3)

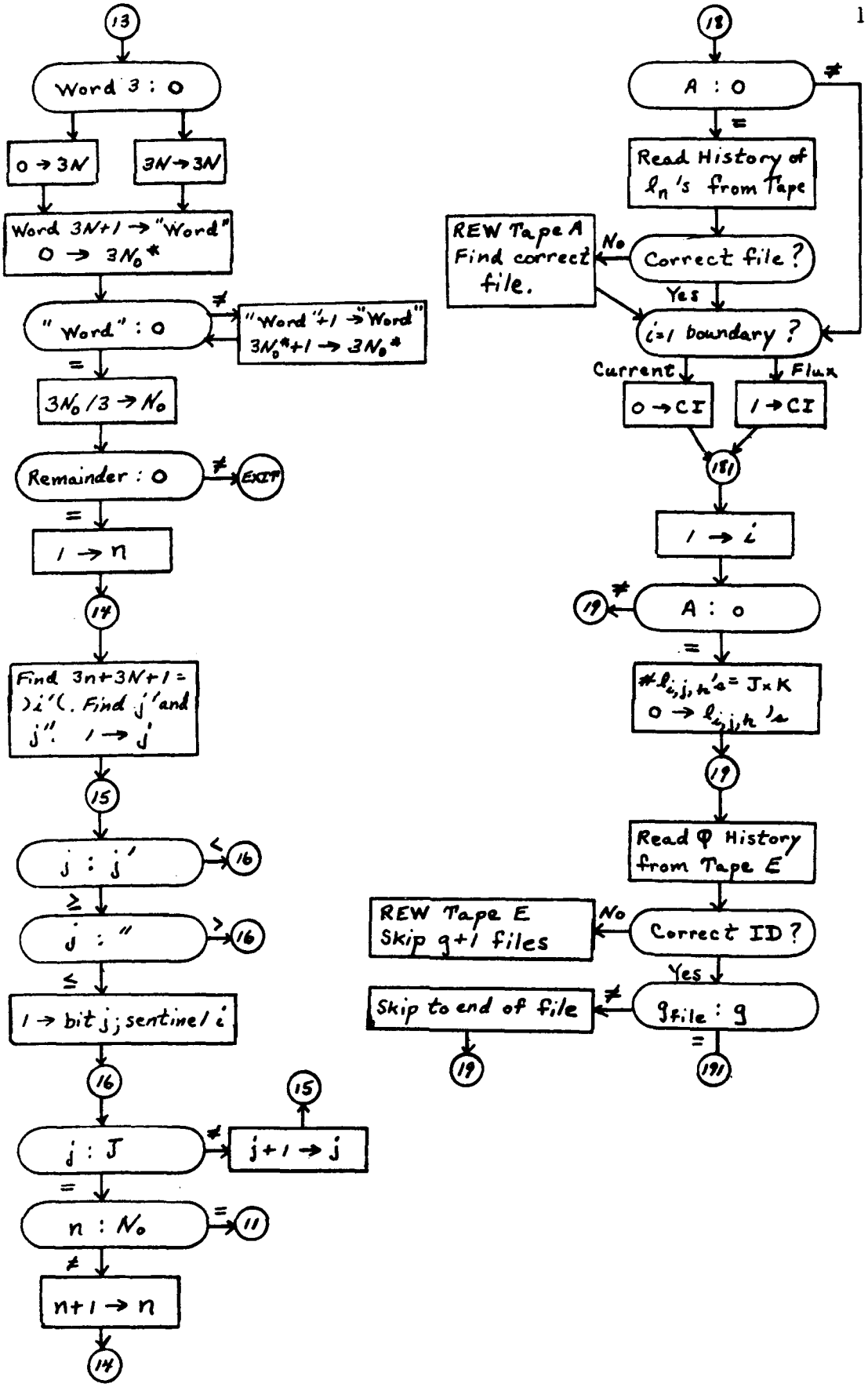

13
Word 3 : 0
0 → 3N
3N→3N
Word 3N+1 → "Word"
0 → 3N0*
" Word" : 0
≠
"Word"+1 → "Word"
3N0*+1 → 3N0*
=
3N0/3 → N0
Remainder : 0
≠
EXIT
=
1 → n
14
Find 3n+3N+1 =)i'(. Find j' and j'': 1 → j'
15
j : j'
<
16
≥
j : ''
>
16
≤
1 → bit j; sentinel i
16
15
j : J
≠
j+1 → j
=
n : N0
=
11
≠
n+1 → n
14
18
A : 0
≠
=
Read History of ln's from Tape
REW Tape A
Find correct file.
No
Correct file ?
Yes
i=1 boundary ?
Current
Flux
0 → CI
1 → CI
181
1 → i
19
≠
A : 0
=
#li,j,h's = J×K
0 → li,j,h's
19
Read φ History from Tape E
REW Tape E
Skip q+1 files
No
Correct ID ?
Yes
Skip to end of file
≠
qfile : q
=
19
191

Figure 18 (4)

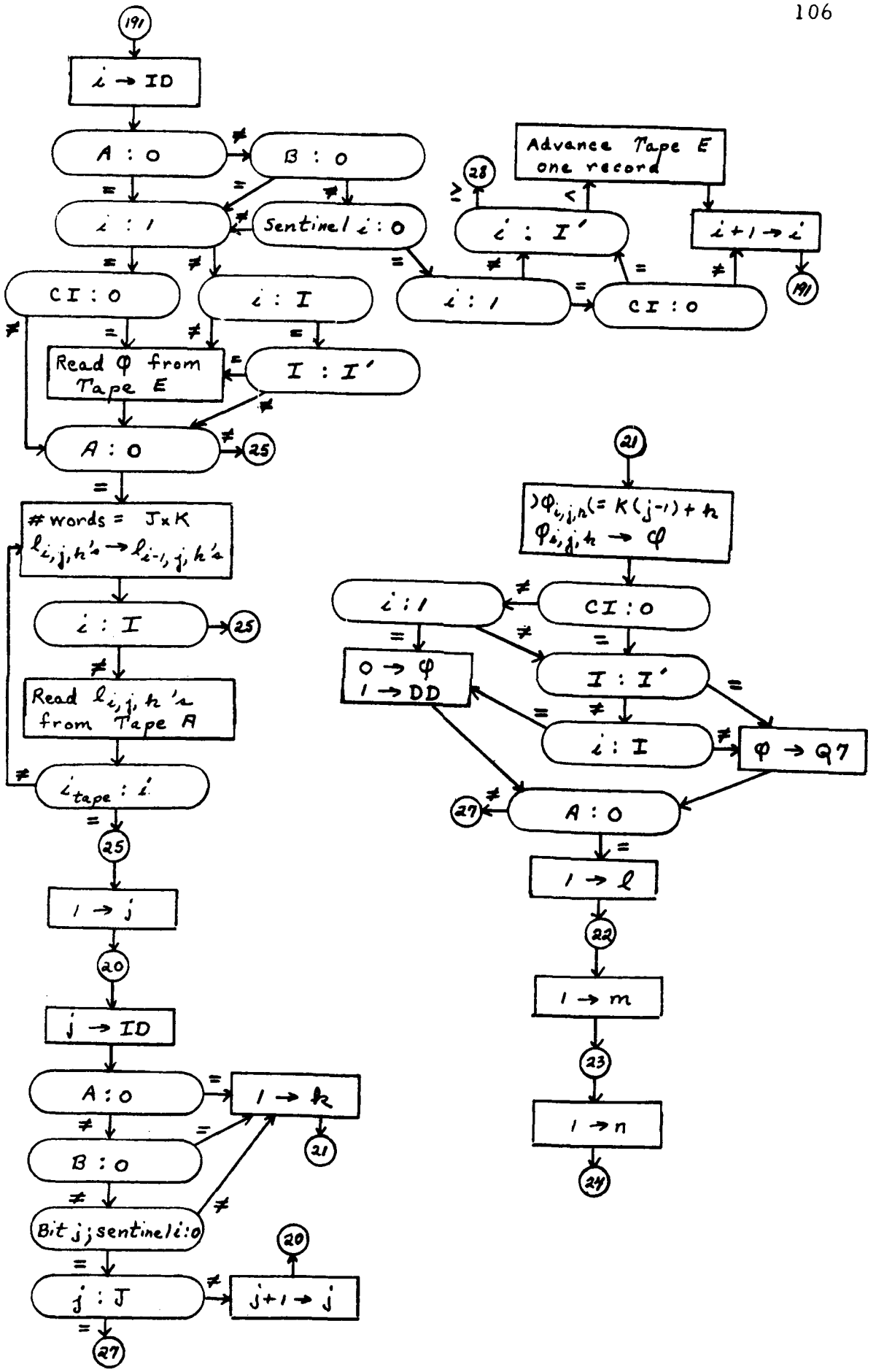

191
i → ID
A : 0
B : 0
i : 1
Sentinel i : 0
CI : 0
i : I
Read φ from Tape E
I : I'
A : 0
25
words = J x K
l i, j, h's → l i-1, j, h's
i : I
25
Read l i, j, h's from Tape A
i tape : i
25
1 → j
20
j → ID
A : 0
1 → k
B : 0
21
Bit j; sentinel i : 0
20
j : J
j+1 → j
27
28
Advance Tape E one record
i : I'
i+1 → i
191
i : 1
CI : 0
21
) φ i, j, h (= K(j-1) + h
φ i, j, h → φ
i : 1
CI : 0
0 → φ
1 → DD
I : I'
i : I
φ → Q7
27
A : 0
1 → l
22
1 → m
23
1 → n
24

Figure 18 (5)

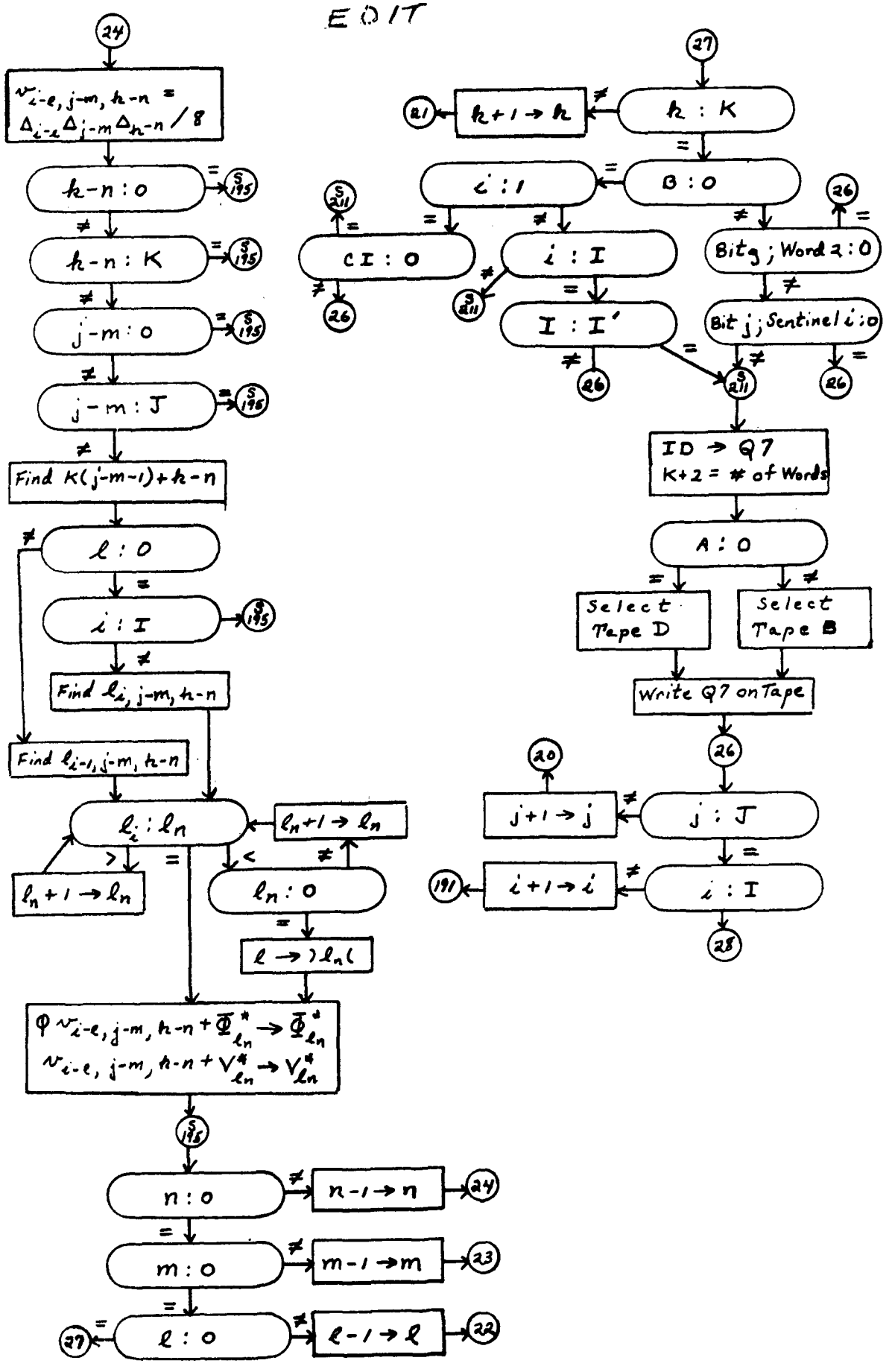

EDIT
24
$\nu_{i-\ell, j-m, h-n} = \Delta_{i-\ell}\Delta_{j-m}\Delta_{h-n}/8$
h-n : 0
h-n : K
j-m : 0
j-m : J
Find K(j-m-1)+h-n
ℓ : 0
i : I
Find ℓ_i, j-m, h-n
Find ℓ_{i-1}, j-m, h-n
$\ell_i : \ell_n$
$\ell_n + 1 \to \ell_n$
$\ell_n : 0$
$\ell \to)\ell_n($
$\Phi\, \nu_{i-\ell, j-m, h-n} + \Phi^*_{\ell_n} \to \Phi^*_{\ell_n}$
$\nu_{i-\ell, j-m, h-n} + V^*_{\ell_n} \to V^*_{\ell_n}$
n : 0
n-1 → n
m : 0
m-1 → m
ℓ : 0
ℓ-1 → ℓ
27
h : K
h+1 → h
i : I
B : 0
CI : 0
i : I
I : I'
Bit g ; Word 2 : 0
Bit j ; Sentinel i : 0
ID → Q7
K+2 = # of Words
A : 0
Select Tape D
Select Tape B
Write Q7 on Tape
j : J
j+1 → j
i : I
i+1 → i
S 195
S 211
21
26
20
191
28

Figure 18 (6)

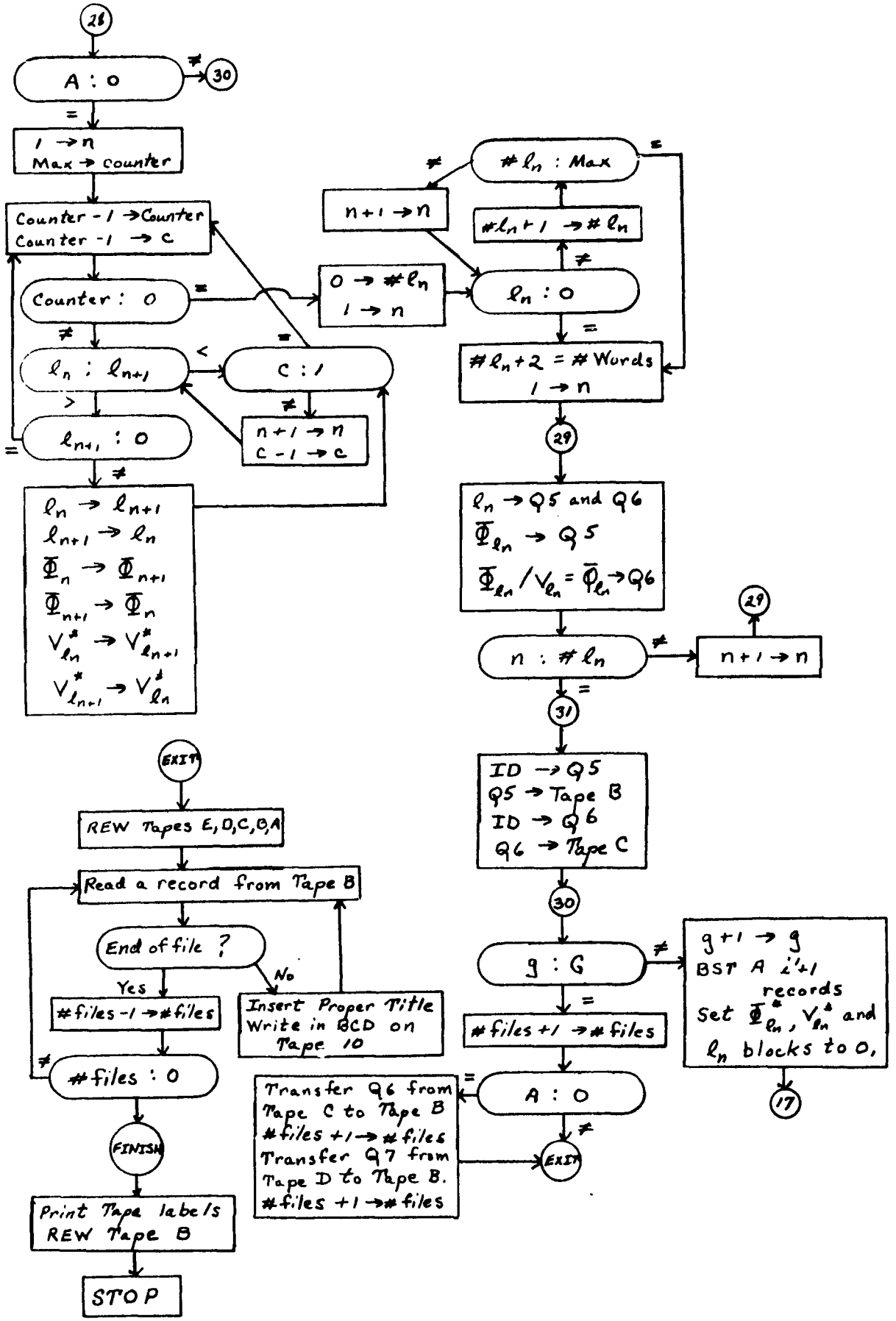

28
A : 0
≠
30
=
1 → n
Max → counter
Counter - 1 → Counter
Counter - 1 → c
Counter : 0
=
≠
ℓn : ℓn+1
<
=
>
ℓn+1 : 0
=
≠
ℓn → ℓn+1
ℓn+1 → ℓn
Φn → Φn+1
Φn+1 → Φn
V*ℓn → V*ℓn+1
V*ℓn+1 → V*ℓn
c : 1
≠
n+1 → n
c - 1 → c
0 → #ℓn
1 → n
≠
n+1 → n
#ℓn : Max
=
≠
#ℓn + 1 → #ℓn
ℓn : 0
=
#ℓn + 2 = # Words
1 → n
29
ℓn → Q5 and Q6
Φℓn → Q5
Φℓn / Vℓn = Φ̄ℓn → Q6
n : #ℓn
≠
n+1 → n
29
=
31
ID → Q5
Q5 → Tape B
ID → Q6
Q6 → Tape C
30
q : G
≠
q+1 → q
BST A i'+1
records
Set Φ*ℓn, V*ℓn and
ℓn blocks to 0,
17
=
#files + 1 → #files
A : 0
=
≠
EXIT
Transfer Q6 from
Tape C to Tape B
#files + 1 → #files
Transfer Q7 from
Tape D to Tape B.
#files + 1 → #files
EXIT
REW Tapes E,D,C,B,A
Read a record from Tape B
End of file ?
Yes
No
#files - 1 → #files
Insert Proper Title
Write in BCD on
Tape 10
≠
#files : 0
FINISH
Print Tape labels
REW Tape B
STOP

Figure 18 (7)

X. GLOSSARY OF SYMBOLS

Symbol	Unit	Possible Range	Description
Mesh			
i	integer	1 to I	Mesh plane perpendicular to $X^{(1)}$
j	integer	1 to J	Mesh plane perpendicular to $X^{(2)}$
k	integer	1 to K	Mesh plane perpendicular to $X^{(3)}$
i, j, k	integers		Location of mesh point at intersection of three planes. (Also used to identify composition and flux boxes, as described below).
I	integer		Maximum i
J	integer		Maximum j
K	integer		Maximum k
$X^{(1)}$	cm		Cartesian coordinates
$X^{(2)}$	cm		Cartesian coordinates
$X^{(3)}$	cm		Cartesian coordinates

Symbol	Unit	Possible Range	Description
Mesh			
$X_i^{(1)}$	cm		Coordinate of mesh plane i
$X_j^{(2)}$	cm		Coordinate of mesh plane j
$X_k^{(3)}$	cm		Coordinate of mesh plane k
$X_1^{(1)}$, $X_1^{(2)}$, $X_1^{(3)}$	cm		Coordinates of boundary planes
$X_I^{(1)}$, $X_J^{(2)}$, $X_K^{(3)}$	cm		Coordinates of boundary planes
$\Delta_i^{(1)}$	cm		Spacing between mesh planes i and i+1
$\Delta_j^{(2)}$	cm		Spacing between mesh planes j and j+1
$\Delta_k^{(3)}$	cm		Spacing between mesh planes k and k+1. Outside the mesh $\Delta_{m'}^{(l')} = 0$, where $l' = 1$, 2, or 3, and m'=0, I, J, or K.
Geometric			
$V_{i,j,k}$	cm^3		Volume of a box whose corners are (i, j, k), (i, j, k+1), (i, j+1, k), (i, j+1, k+1), (i+1, j, k), (i+1, j, k+1),

Symbol	Unit	Possible Range	Description
Geometric			
			(i+1, j+1, k), (i+1, j+1, k+1). This box must be of uniform composition, and is referred to as "composition box" (i, j, k).
$v_{i,j,k}$	cm^3		$v_{i,j,k} = \frac{1}{8} V_{i,j,k}$ is the octant of the composition box (i, j, k) nearest to mesh point (i, j, k). This elementary volume, obtained by dividing each composition box in eight equal parts, is referred to as an "octant".
			$(v_{0,j,k} = v_{I,j,k} = v_{i,0,k} = v_{i,J,k} = v_{i,j,0} = v_{i,j,K} = 0)$.
$V'_{i,j,k}$	cm^3		$V'_{i,j,k} = \sum_{\ell,m,n=0}^{1} v_{i-\ell,j-m,k-n}$ is the total volume of the eight octants adjacent to mesh point

Symbol	Unit	Possible Range	Description
Geometric			
			(i, j, k). This volume is referred to as "flux box" (i, j, k).
$\alpha^{(1)}_{j,k}$	cm^2		Area of the projection $v_{i,j,k}$ perpendicular to $X^{(1)}$, $\alpha^{(1)}_{j,k} = \frac{1}{4} \Delta^{(2)}_j \Delta^{(3)}_k$
$\alpha^{(2)}_{i,k}$	cm^2		Projection of $v_{i,j,k}$ $X^{(2)}$, $\alpha^{(2)}_{i,k} = \frac{1}{4} \Delta^{(1)}_i \Delta^{(3)}_k$
$\alpha^{(3)}_{i,j}$	cm^2		Projection of $v_{i,j,k}$ $X^{(3)}$, $\alpha^{(3)}_{i,j} = \frac{1}{4} \Delta^{(1)}_i \Delta^{(2)}_j$ Each of these areas is referred to as a "quadrant", since four quadrants comprise one face of a flux box. $\alpha^{(l')}_{m',n'} = 0$, where m' or n' = 0, I, J, K.
V_F	cm^3		Total volume of regions containing fissionable material

Symbol	Unit	Possible Range	Description
Geometric			
			$V_F = \sum_{i,j,k=i}^{I,J,K} V''_{i,j,k}$
V''			Fissionable volume in flux box (i, j, k), $V'' = \sum_{,m,n=0,1} {}^{\nu}{}_{i-,j-m,k-n}\ {}^{\beta}{}_{i-,j-m,k-n}$
$\sum^{F}$			Summation to be carried out only over that portion of the mesh containing fissionable material. $\sum^{F} = \sum_{i=i^F_{min}}^{i^F_{max}} \sum_{j=j^F_{min}}^{j^F_{max}} \sum_{k=k^F_{min}}^{k^F_{max}}$
Indexes			
g	integer	1 to G	R^g refers R to energy group g.
g'	integer	1 to G	In all cases, fission cross section is zero for $g < g'$.

Symbol	Unit	Possible Range	Description
Indexes			
G	integer	1 to 15	Maximum g.
(t)	integer	0 to 63	$R^{(t)}$ refers R to the t^{th} mesh sweep during a group iteration.
(T)	integer	0 to 63	Maximum t (the last mesh sweep of a group iteration).
(r)	integer	0 to 63	$R^{(r)}$ refers R to run r.
$l_{i,j,k}$	integer	0 to 63	Composition index (see symbols under "Composition").
q	integer	0 to 7	Identification of an octant in a flux box. The number q is the decimal equivalent of the binary number (l, m, n).

Symbol	Unit	Possible Range	Description
Indexes			
ℓ	integer	0 to 1	The ℓ, m, or n=0 denotes the positive direction along $X^{(1)}$, $X^{(2)}$ or $X^{(3)}$ respectively, and
m	integer	0 to 1	ℓ, m, or n=1 denotes the negative direction. Positive and negative, in this case, are referred to an
n	integer	0 to 1	origin at mesh point (i, j, k).
Composition			
$M^{g}_{i,j,k}$	cm^{-1}		Moderating (slowing-down) cross section.
$F^{g}_{i,j,k}$	cm^{-1}		Fission cross section.
$D^{g}_{i,j,k}$	cm		Diffusion coefficient.
$A^{g}_{i,j,k}$	cm^{-1}		Removal cross section = M + absorption cross section.
$l_{i,j,k}$	integer	0 to 63	Composition index of composition box (i, j, k). For each index, G sets of (M^{g}, F^{g}, D^{g}, and A^{g})

Symbol	Unit	Possible Range	Description
Composition			
			must be specified. If composition box (i, j, k) falls outside of the mesh (i, j, k < 1 or > I-1, J-1, K-1), then each M, F, D, A is zero.
$\beta_{i,j,k}$	integer	0 or 1	If composition box (i, j, k) contains fissionable material use 1; otherwise use 0.
$Z_0^{(1)g}$			Ratio of outward current to flux at a boundary plane (applicable when a current boundary condition is specified). $Z^g(\frac{D \nabla \varphi}{\varphi})^g$ is a property of the medium outside the boundary plane, which is assumed constant over the entire plane.
$Z_I^{(1)g}$			
$Z_0^{(2)g}$			
$Z_J^{(2)g}$			
$Z_0^{(3)g}$			
$Z_K^{(3)g}$			

Symbol	Unit	Description
Coefficients		
$C^{(1)g}_{i,j,k}$	cm^2	$C^{(1)g}_{i,j,k} = \sum_{m,n=0}^{1} D^{g}_{i,j-m,k-n}\ \alpha^{(1)}_{j-m,k-n} \Big/ \Delta^{(1)}_{i}$
$C^{(2)g}_{i,j,k}$	cm^2	$C^{(2)g}_{i,j,k} = \sum_{\ell,n=0}^{1} D^{g}_{i-\ell,j,k-n}\ \alpha^{(2)}_{i-\ell,k-n} \Big/ \Delta^{(2)}_{j}$
$C^{(3)g}_{i,j,k}$	cm^2	$C^{(3)g}_{i,j,k} = \sum_{\ell,m=0}^{1} D^{g}_{i-\ell,j-m,k}\ \alpha^{(3)}_{i-\ell,j-m} \Big/ \Delta^{(3)}_{k}$
$E^{g}_{i,j,k}$	cm^2	$E^{g}_{i,j,k} = \sum_{\ell,m,n=0}^{1} A^{g}_{i-\ell,j-m,k-n}\ v_{i-\ell,j-m,k-n}$
$B^{g}_{i,j,k}$	cm^2	$B^{g}_{i,j,k} = (C^{(1)}_{i,j,k} + C^{(1)}_{i-1,j,k} + C^{(2)}_{i,j,k} + C^{(2)}_{i,j-1,k} + C^{(3)}_{i,j,k} + C^{(3)}_{i,j,k-1} + E_{i,j,k})^{g}$
$H^{g}_{i,j,k}$	cm^2	$H^{g}_{i,j,k} = \sum_{\ell,m,n=0}^{1} M^{g}_{i-\ell,j-m,k-n}\ v_{i-\ell,j-m,k-n}$
$F'^{g}_{i,j,k}$	cm^2	$F'^{g}_{i,j,k} = \sum_{\ell,m,n=0}^{1} F^{g}_{i-\ell,j-m,k-n}\ v_{i-\ell,j-m,k-n}$
Boundary Coefficients		
$C^{(1)g}_{i',j,k}$	cm^2	$C^{(1)g}_{i',j,k} = Z^{(1)g}_{i'} \sum_{m,n=0}^{1} \alpha^{(1)}_{j-m,k-n}$ $i'=0$ or I

Symbol	Unit	Description
Boundary Coefficients		
$C^{(2)g}_{i,j',k}$	cm^2	$C^{(2)g}_{i,j',k} = Z^{(2)g}_{j'} \sum_{\ell,n=0}^{1} \alpha^{(2)}_{i-\ell,k-n}$ $\quad j' = 0 \text{ or } J$
$C^{(3)g}_{i,j,k'}$	cm^2	$C^{(3)g}_{i,j,k'} = Z^{(3)g}_{k'} \sum_{\ell,m=0}^{1} \alpha^{(3)}_{i-\ell,j-m}$ $\quad k' = 0 \text{ or } K$
Components of Coefficients		
$d^{(1)g}_{i,j,k}$	cm^2	$d^{(1)g}_{i,j,k} = D^{g}_{i,j,k}\ \Delta^{(2)}_{j}\ \Delta^{(3)}_{k} \Big/ 4\ \Delta^{(1)}_{i}$
$d^{(2)g}_{i,j,k}$	cm^2	$d^{(2)g}_{i,j,k} = D^{g}_{i,j,k}\ \Delta^{(1)}_{i}\ \Delta^{(3)}_{k} \Big/ 4\ \Delta^{(2)}_{j}$
$d^{(3)g}_{i,j,k}$	cm^2	$d^{(3)g}_{i,j,k} = D^{g}_{i,j,k}\ \Delta^{(1)}_{i}\ \Delta^{(2)}_{j} \Big/ 4\ \Delta^{(3)}_{k}$
$a^{g}_{i,j,k}$	cm^2	$a^{g}_{i,j,k} = A^{g}_{i,j,k}\ \Delta^{(1)}_{i}\ \Delta^{(2)}_{j}\ \Delta^{(3)}_{k} \Big/ 8$
$m^{g}_{i,j,k}$	cm^2	$m^{g}_{i,j,k} = M^{g}_{i,j,k}\ \Delta^{(1)}_{i}\ \Delta^{(2)}_{j}\ \Delta^{(3)}_{k} \Big/ 8$
$r^{(1)g}_{i,j,k}$	cm^2	$r^{(1)g}_{i,j,k} = d^{(1)g}_{i,j,k} + d^{(1)g}_{i,j-1,k}$
$r^{(2)g}_{i,j,k}$	cm^2	$r^{(2)g}_{i,j,k} = d^{(2)g}_{i,j,k} + d^{(2)g}_{i-1,j,k}$
$e^{g}_{i,j,k}$	cm^2	$e^{g}_{i,j,k} = \sum_{\ell,m=0}^{1} a^{g}_{i-\ell,j-m,k}$

Symbol	Unit	Description
Components of Coefficients		
$h^{g}_{i,j,k}$	cm^2	$h^{g}_{i,j,k} = \sum_{\ell,m=0}^{l} m^{g}_{i-\ell,j-m,k}$
$b^{g}_{i,j,k}$	cm^2	$b^{g}_{i,j,k} = r^{(1)g}_{i,j,k} + r^{(2)g}_{i,j,k} + C^{(3)g}_{i,j,k} + e^{g}_{i,j,k}$
Flux		
$\phi^{g(t)}_{i,j,k}$	$\frac{\text{neutrons}}{\text{sec-cm}^2}$	Flux in group g at mesh point i, j, k after it has been improved t times during the group iteration. $\phi^{g(0)}_{i,j,k}$ is the flux estimate at the beginning of the group iteration. $\phi^{g}_{0,j,k} = \phi^{g}_{I+1,j,k} = \phi^{g}_{i,0,k} = \phi^{g}_{i,J+1,k} = \phi^{g}_{i,j,0} = \phi^{g}_{i,j,K+1} = 0.$
Source and Loss		
$\mathcal{S}_{i,j,k}$	$\frac{\text{neutrons}}{\text{sec}}$	Source of fission neutrons in flux box (i, j, k). $\mathcal{S}_{i,j,k}$ is normalized to an average of one neutron/sec over all fissionable regions. $\mathcal{S}_{i,j,k} = \sum_{q=0}^{7} P_{(i,j,k)q}$

Symbol	Unit	Description
Source and Loss		
χ^g		Fraction of the fission neutrons that are born in group g. $0 \leq \chi^g \leq 1$.
$\S^*_{i,j,k}$	$\frac{\text{neutrons}}{\text{sec}}$	Fission source at point (i, j, k), not normalized.
$\S^{g*}_{i,j,k}$	$\frac{\text{neutrons}}{\text{sec}}$	Contribution of group g to $\S^*_{i,j,k}$. $\S^{g*}_{i,j,k} = F'^g_{i,j,k}\, \phi^g_{i,j,k}$.
$\S^{g**}_{i,j,k}$	$\frac{\text{neutrons}}{\text{sec}}$	$\S^{g_1**}_{i,j,k} = \sum_{g=g_1}^{g} \S^{g*}_{i,j,k}$
$S^g_{i,j,k}$	$\frac{\text{neutrons}}{\text{sec}}$	Neutron source in flux box (i, j, k), in group g, due to both fission and degradation. $S^g_{i,j,k} = \chi^g \S_{i,j,k} + H^{g-1}_{i,j,k}\, \phi^{(g-1)(t)}_{i,j,k}$
$\overline{\S}_{i,j,k}$	$\frac{\text{neutrons}}{\text{sec}}$	$\overline{\S}_{i,j,k} = \S^{**}_{i,j,k} \Big/ f$
$\S^{(r)}_{i,j,k}$	$\frac{\text{neutrons}}{\text{sec}}$	Estimates of the above quantities after completion of run
$S^{g(r)}_{i,j,k}$	$\frac{\text{neutrons}}{\text{sec}}$	Estimates of the above quantities after completion of run

Symbol	Unit	Description
Source and Loss		
$L^{g}_{i,j,k}$		Neutron removal and leakage from flux box (i,j,k), in group g. $L^{g}_{i,j,k}(\phi) = B^{g}_{i,j,k}\ \phi^{g}_{i,j,k} - (C^{(1)g}_{i,j,k}\ \phi^{g}_{i+1,j,k} + C^{(1)g}_{i-1,j,k}\ \phi^{g}_{i-1,j,k} + C^{(2)g}_{i,j,k}\ \phi^{g}_{i,j+1,k} + C^{(2)g}_{i,j-1,k}\ \phi^{g}_{i,j-1,k} + C^{(3)g}_{i,j,k}\ \phi^{g}_{i,j,k+1} + C^{(3)g}_{i,j,k-1}\ \phi^{g}_{i,j,k-1})$. $L^{g(t)}_{i,j,k}(\phi)$ refers $L^{g}_{i,j,k}$ to mesh sweep t during a group iterations.
$\Omega^{g}_{i,j,k}(\phi^{g}, S^{g})$	$\frac{\text{neutrons}}{\text{sec}}$	$\Omega^{g}_{i,j,k}(\phi^{g}, S^{g}) = L^{g}_{i,j,k}(\phi) - S^{g}_{i,j,k}$
		Neutron leakage through boundary plane
$L^{(1)g}_{1}$, $L^{(1)g}_{I}$		For specified current condition: $L^{(1)}_{1} = \sum_{j,k=1}^{J,K} C^{(1)g}_{0,j,k}\ \phi^{g}_{1,j,k}$
$L^{(2)g}_{1}$, $L^{(2)g}_{J}$		$L^{(1)g}_{I} = \sum_{j,k=1}^{J,K} C^{(1)g}_{I,j,k}\ \phi^{g}_{I,j,k,}$ etc.
$L^{(3)g}_{1}$, $L^{(3)g}_{K}$		

Symbol	Unit	Description
Source and Loss		
		Neutron leakage through boundary plane For specified flux condition: $L_I^{(1)g} = \sum_{j,k=1}^{J,K} C^{(1)g}_{1,j,k} (\phi^g_{2,j,k} - \phi^g_{1,j,k})$ $L_I^{(1)g} = \sum_{j,k=1}^{J,K} C^{(1)g}_{I-1,j,k} (\phi^g_{I-1,j,k} - \phi^g_{I,j,k})$, etc.
L_B^g	$\frac{\text{neutrons}}{\text{sec}}$	Neutron leakage from the mesh in group g. $L_B^g = (L_1^{(1)} + L_I^{(1)} + L_1^{(2)} + L_J^{(2)} + L_1^{(3)} + L_K^{(3)})^g$
Power		
$p^g_{(i,j,k)q}$	$\frac{\text{fissions}}{\text{sec-cm}^3}$	Power density in quadrant q of flux box (i, j, k) due to fissions induced by group g neutrons. $p^g_{(i,j,k)q} = F^g_{i-1,j-m,k-n} \phi^g_{i,j,k}$
$p_{(i,j,k)q}$	$\frac{\text{fissions}}{\text{sec-cm}^3}$	Power density $p_{(i,j,k)q} = \sum_{g=1}^{G} p^g_{(i,j,k)q}$
ν	$\frac{\text{neutrons}}{\text{fission}}$	

Symbol	Unit	Description
Miscellaneous Factors		
δ_{ϕ}^{g}		Flux change criterion for terminating flux iteration $$\sum_{i,j,k=1}^{I,J,K} \left(\phi_{i,j,k}^{g(t)} - \phi_{i,j,k}^{g(t-1)} \right)^2 < \delta_{\phi}^{g}$$
δ_p		Power change criterion for final run $$\sum_{i,j,k=1}^{I,J,K} \left(\sum_{q=0}^{7} (P_{(i,j,k)q}^{(r)} - P_{(i,j,k)q}^{(r-1)})^2 \right) < \delta_p$$
δ_f		Criticality error criterion for final run. $$f^{(r)} - f^{(r-1)} < \delta_f$$
$R^{g(t-1)}$		Renormalization factor after $(t-1)^{th}$ mesh sweep. $$R^{g(t-1)} = \frac{\sum_{i,j,k=1}^{I,J,K} S_{i,j,k}^{g}}{\left(L_B^{g(t-1)} + \sum_{i,j,k=1}^{I,J,K} E_{i,j,k}^{g} \phi_{i,j,k}^{g(t-1)} \right)}$$
α		Liebmann Acceleration Factor

Symbol	Unit	Description

Miscellaneous Factors

$\epsilon_{\varphi}^{(t)}$

Flux change,

$$\epsilon_{\varphi}^{(t)} = \sum_{i,j,k=1}^{I,\ J,\ K} \left(\varphi_{i,j,k}^{g(t)} - \varphi_{i,j,k}^{g(t-1)} \right)^2 .$$

$\epsilon_{p}^{(r)}$

Power change,

$$\epsilon_{p}^{(r)} = \sum_{i,j,k=1}^{I,\ J,\ K} \sum_{q=0}^{7} (P_{(i,j,k)q}^{(r)} - P_{(i,j,k)q}^{(r-1)})^2 .$$

$\epsilon_{s}^{(r)}$

Source change between successive runs,

$$\epsilon_{s}^{(r)} = \sum^{F} \left| s_{i,j,k}^{(r)} - s_{i,j,k}^{(r-1)} \right| .$$

δ_{s}

Convergence criterion for terminating the runs,

$$\epsilon_{s}^{(r)} \leq \delta_{s} .$$

C_1

Percentage of initial total improvement to serve as a secondary convergence criterion for terminating the flux iterations.

t_{max}

Maximum number of sweeps allowable in the flux iterations (tertiary convergence criterion).

Symbol	Unit	Description
Miscellaneous Factors		
C_2		Frequency print regulator and frequency regulator for computing error ratio, $\epsilon^{(t)} / \epsilon^{(t-C_2)}$ in flux iterations.
$\%V_F$		Percentage of the fissionable volume by which the computed volume and the estimated volume may differ. If the difference is greater than this the machine stops.
Programming Conventions		
I'	integer	Number of planes on which flux is computed.
J'	integer	Number of columns per plane on which flux is computed.
K'	integer	Number of points per column on which flux is computed.
g^i	integer	Group number of the flux guesses to be improved by the Group Sweep Program.
g^p	integer	$g^p = g^i + 1$.

Symbol	Unit	Description
Programming Conventions		
C	integer	0 if $K \leq 36$ 1 if $K > 36$ (C indicates whether the sentinels for each column are carried as one or two words.)
RCW	binary word	Reactor Control Word: a 12 bit representation of boundary conditions on the six faces of the reactor. odd bits denote 0 - flux boundary 1 - current boundary even bits denote 0 - zero boundary 1 - non-zero boundary Bits 35, 34 refer to i = 1 33, 32 j = 1 31, 30 k = 1 29, 28 i = I 27, 26 j = J 25, 24 k = K

Symbol	Unit	Description
Programming Conventions		
SFGW	binary word	Source Flux Guess Word: It is zero if a rough power guess is supplied, or non-zero if the results of a previous run are to be used.
CC	integer	Convergence Code for Group Sweep Program. CC = 1 - primary convergence test met; 2 - secondary convergence test met; 3 - tertiary convergence test met

A block of 14 words of control input are stored in memory and preserved there by all succeeding programs. This block contains:

Decimal Location	Decrement	Tag	Address
29	α (floating point form)		
30	g'	0	G
31	g^{p}[1]	4	RCW
32	i^{F}_{max}	0	I'
33	i^{F}_{min}	0	J'
34	j^{F}_{max}	0	K'

[1] $g^{p} = 1$ at the beginning of each run. g^{p} is increased by 1 at the end of a group iteration by the Group Sweep Program before control is transferred to the Group Source Program.

Decimal Location	Decrement	Tag	Address
35	j^{F}_{min}	0	I
36	k^{F}_{max}	0	J
37	k^{F}_{min}	0	K
38	Tape A[1]	0	Tape B
39	Tape C	0	Tape D
40	Tape E	0	Flux Guess Tape[2]
41			Tape F
42	0	4	1 (HTJ 1, 4)

[1] Master tape

[2] The Flux Guess Tape is either tape B (and the sign of 40 is -) or tape E (and the sign of 40 is +).

APPENDIX A: INPUT

INSTRUCTIONS FOR FILLING OUT INPUT FORMS

General Instructions

The lines on most of the input sheets are divided into two parts. The upper half of a line contains identification and headings which serve as guides to the person preparing the input. The lower half contains the numerical data which are punched by the keypunch operator. If no letter or number is given in column one, that column is to be left blank unless this is the last card of a record. An eight in column one indicates this is the last card of a record. This card may also have other punches in column one and must contain data. There can be no blanks within a record. If a value is zero, enter the number 0. Except for RPG (Rough Power Guess), no records can be omitted. The maximun number of digits is eight plus exponent. Commas cannot be used to separate the digits in a number. Floating point numbers can be entered in several ways. For example, π can be entered as:

3.1415927
+31416/-4
.03142/2

Fixed Point Geometrical and Physical Data

Memory Size - Must be given to the nearest thousands, i.e., 8000, 16,000, 32,000.

CSFN - Cross Section File Number. The file of cross section records to be selected from the configuration and cross section tape.

RCW - Reactor Control Word. This word is computed by defining the boundary conditions which are:

0 - zero flux

1 - zero current

2 - non zero flux (not coded at present)

3 - non zero current

The appropriate values are multiplied by:

1 for i = 1

4 for j = 1

16 for k = 1

64 for i = I

256 for j = J

1024 for k = K

The sum of these products is the RCW.

SFGW - Source Flux Guess Word. Positive for elastic scattering and negative for inelastic scattering (not coded at present). The value entered is zero if a flux guess is to be supplied from a previous problem. It is one if a rough power guess is supplied

Floating Point Input

V_F - Fissionable volume. If an estimate is not made this should be 0.

$\%V_F$ - This is the permissible error in V_F and is a decimal fraction.

List A and List B

Both of these lists must always be supplied. If one of the lists is to be disregarded, the number 100 must be written as the first word of the record. Then, column 1 has an 8 punch in addition to the letter D. The column headed with an asterisk is to be used only if 100 is written in that space.

Coarse Mesh and Rough Power Guess

The coarse mesh supplied for the rough power guess must not be larger than the actual mesh. The first coordinates of the rough power guess cannot be greater than i^F_{min}. If there are less than six values given for any of the coordinates, the remaining blanks on that line should be disregarded by both the person preparing the input and the keypunch operator.

Configurations

These must be listed in order of increasing C_n starting with $C_n = 1$.

ORDER OF INPUT

First Record[1]: Fixed Point Geometrical and Physical Data
Current Boundary Conditions
Floating Point Input

[1] If a value other than 2.46 is desired for ν, it can be entered with the first record, e.g., if $\nu = 1.5$ the card would be punched C1, 1.5,

Second Record: List A

Third Record: List B

Fourth Record: Mesh Spacings and Coordinates

Fifth Record: Edit Input

Sixth Record: Rough Power Guess (if supplied)

OMEGA (one card for each run - in card reader)

Two blank cards

INSTRUCTIONS TO MACHINE OPERATORS

It is necessary that the operator know:

1. If input is on cards or on tape,
2. If a control restart deck is to be punched,
3. If the Master Tape is to be duplicated, and
4. If Monitor printing is wanted during the Group Sweep. At present, Omega must be read in from cards. Also, if V_F is supplied, a tolerance must be given. If the V_F computed is outside this tolerance, the machine stops and this difference is printed. If this difference is small and it is decided to continue, this must also be indicated.

TRIXY INPUT DATA

Fixed Point Geometrical and Physical Data

col. 1	ADDRESS				MEM.SIZE		I		J		K		i^F_{min}		i^F_{max}	
D	2314	,	35	,		,		,		,		,		,		,

	ADDRESS				j^F_{min}		j^F_{max}		k^F_{min}		k^F_{max}		g'		G	
D		,	35	,		,		,		,		,		,		,

Current Boundary Conditions

col. 1	ADDRESS				$L_z{}^1{}_0$		$L_z{}^1{}_I$		$L_z{}^2{}_0$		$L_z{}^2{}_J$		$L_z{}^3{}_0$		$L_z{}^3{}_K$	
D		,	35	,		,		,		,		,		,		,

	ADDRESS				PROB. NO.		CSFN		RCW		SFGW		t_{max}		C_2	
D	2332	,	35	,		,		,		,		,		,		,

Floating Point Input

col. 1	ADDRESS		V_F		$\% V_F$		α		C_1		δ_f		δ_δ	
C	2339	,		,		,		,		,		,		,

	ADDRESS		$\delta_\phi^{(1)}$		$\delta_\phi^{(2)}$		$\delta_\phi^{(3)}$		$\delta_\phi^{(4)}$		$\delta_\phi^{(5)}$		$\delta_\phi^{(6)}$	
.	2345	,		,		,		,		,		,		,

Page 1 of ___A

TRIXY INPUT DATA

Compositions - List A

Col 1			k	C_n	l_n	i	j	
D,	4300,	35,	___,	___,	___,	___,	___,	(FIRST CARD ONLY)

Col 1	k	C_n	l_n	i	j	k	C_n	l_n	i	j	*
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,

(100,) IMMEDIATELY FOLLOWS LAST j, REGARDLESS OF COLUMN NOTATION, e.g. j k or j *

1001, 100, 1001, 100,

TRIXY INPUT DATA

Compositions - List A

Col 1			k	C_n	l_n	i	j	
____,	____,	____,	____,	____,	____,	____,	____,	(FIRST CARD ONLY)

Col 1	k	C_n	l_n	i	j	k	C_n	l_n	i	j	*
____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,
____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,
____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,
____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,
____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,
____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,
____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,
____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,
____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,
____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,	____,

(100,) IMMEDIATELY FOLLOWS LAST j, REGARDLESS OF COLUMN NOTATION, e.g. j k or j *
1001, 100, 1001, 100,

TRIXY INPUT DATA

Compositions - List B

Col 1			k	C_n	l_n	i	j	
D	5800	35	___	___	___	___	___	(FIRST CARD ONLY)

Col 1	k	C_n	l_n	i	j	k	C_n	l_n	i	j	*

(100,) IMMEDIATELY FOLLOWS LAST j, REGARDLESS OF COLUMN NOTATION, e.g. j k or j *

1001, 100, 1001, 100,

Page ___ of ___ B

TRIXY INPUT DATA

Compositions - List B

Col 1	k	C_n	l_n	i	j	k	C_n	l_n	i	j	*
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,
___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,	___,

(100,) IMMEDIATELY FOLLOWS LAST j, REGARDLESS OF COLUMN NOTATION, e.g.

j	k		j	*
1001,	100	or	1001,	100,

TRIXY INPUT DATA

Mesh Spacings and Coordinates

(Not to exceed 250 words)

col 1	ADDRESS		N_{il}		$X_{i1}^{(1)}$		N_{i2}		$X_{i2}^{(1)}$					
C		,		,		,		,		,		,		,
		,		,		,		,		,		,		,
		,		,		,		,		,		,		,
		,		,		,		,		,		,		,
		,		,		,		,		,		,		,
		,		,		,		,		,		,		,

AN 8 MUST BE IN COLUMN 1 OF LAST DATA CARD OF THIS INPUT.

TRIXY INPUT DATA

Mesh Spacings and Coordinates

(Not to exceed 250 words)

col. 1														
		,		,		,		,		,		,		,
		,		,		,		,		,		,		,
		,		,		,		,		,		,		,
		,		,		,		,		,		,		,
		,		,		,		,		,		,		,
		,		,		,		,		,		,		,
		,		,		,		,		,		,		,

AN 8 MUST BE IN COLUMN 1 OF LAST DATA CARD OF THIS INPUT.

TRIXY INPUT DATA

EDIT RECORD

A 1	A 2	A 3	A 4	Item A(4+g); All fluxes for group g equal: 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	WORD I

Item Cg(i, j); Selected fluxes for group g equal: 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 34 35	S E N	WORDII

If selected columns for fluxes under Item Cg(i, j) are the same as those selected for power under Item B(i, j), SEN = 1. If selected columns differ, SEN = 0.

col 1		col 7		col 18	OCTAL WORD I	col 31
0		06262				

col 1		col 7		col 18	OCTAL WORD II	col 31
0		06263				

D , 3252 , 35 , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ ,
___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ ,
___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ ,

Column 1 of the first two cards contains a zero.

There must be an 8 in column 1 of the last card of this record.

TRIXY INPUT DATA

Coarse Mesh and Rough Power Guess

col. 1	ADDRESS		I COORDINATES									
		,	,	,	,	,	,	,	,	,	,	,

col. 1	J COORDINATES											
		,	,	,	,	,	,	,	,	,	,	,

col. 1	K COORDINATES											
		,	,	,	,	,	,	,	,	,	,	,

col. 1	ROUGH POWER GUESS											
		,	,	,	,	,	,	,	,	,	,	,
		,	,	,	,	,	,	,	,	,	,	,
		,	,	,	,	,	,	,	,	,	,	,
		,	,	,	,	,	,	,	,	,	,	,
		,	,	,	,	,	,	,	,	,	,	,
		,	,	,	,	,	,	,	,	,	,	,

AN 8 MUST BE IN COLUMN 1 OF LAST DATA CARD OF THIS INPUT.

TRIXY INPUT DATA

Configurations

Col.
1

NOTE: Pattern of a configuration is D7000, 35, C_n, J of enclosing rectangle, 1st word, 2nd word, etc.
A configuration may be continued on more than one line but must not have more than 72 characters per line, or exceed a total of 199 words.
Column 1 will be left blank on all continued lines EXCEPT the last line of a configuration, then it will have an 8 written in it.
The first word of the configuration can be either a fill (no sign) or a skip (- sign).
100 must be written at the end of each column and 1000 at the end of each configuration.

TRIXY INPUT DATA

Cross Section Data - Record 1

col 1	ADDRESS				Ident.		G		Word R-1		Word R-2		Word R-3		Word R-4	
D	3000	,	35	,		,		,		,		,		,		,

col 1	CHI-1		CHI-2		CHI-3		CHI-4		CHI-5		CHI-6	
=		,		,		,		,		,		,

Cross Section Data-Record 2

col 1	ADDRESS		D_1^1		A_1^1		M_1^1		F_1^1	

FISSIONABLE

Example:

$D_{l=1}^{g=1}$

col 1	D		A		M		F		D		A		M		F	
		,		,		,		,		,		,		,		,

	D		A		M		F		D		A		M		F	
		,		,		,		,		,		,		,		,

	D		A		M		F		D		A		M		F	
		,		,		,		,		,		,		,		,

AN 8 MUST BE IN COLUMN 1 OF LAST DATA CARD OF EACH RECORD.

TRIXY INPUT DATA

Cross Section Data - Record 2 FISSIONABLE (con't.)

col. 1	D		A		M		F		D		A		M		F	
		,		,		,		,		,		,		,		,
	D		A		M		F		D		A		M		F	
		,		,		,		,		,		,		,		,
	D		A		M		F		D		A		M		F	
		,		,		,		,		,		,		,		,
	D		A		M		F		D		A		M		F	
		,		,		,		,		,		,		,		,
	D		A		M		F		D		A		M		F	
		,		,		,		,		,		,		,		,
	D		A		M		F		D		A		M		F	
		,		,		,		,		,		,		,		,

AN 8 MUST BE IN COLUMN 1 of LAST DATA CARD OF EACH RECORD.

Page 1 of ____ NF

TRIXY INPUT DATA

NON-FISSIONABLE

Cross Section Data - Record 3

col. 1	ADDRESS		D^1_{1001}		A^1_{1001}		M^1_{1001}		D		A		M	
C	3100	,		,		,		,		,		,		,

	D		A		M		D		A		M	
		,		,		,		,		,		,

	D		A		M		D		A		M	
		,		,		,		,		,		,

	D		A		M		D		A		M	
		,		,		,		,		,		,

	D		A		M		D		A		M	
		,		,		,		,		,		,

	D		A		M		D		A		M	
		,		,		,		,		,		,

Example:

$D^{g=1}_{l=1001}$

AN 8 MUST BE IN COLUMN 1 OF LAST DATA CARD OF EACH RECORD.

TRIXY INPUT DATA

NON-FISSIONABLE
(Con't)

Cross Section Data - Record 3

col. 1	D		A		M		D		A		M	
		,		,		,		,		,		,
	D		A		M		D		A		M	
		,		,		,		,		,		,
	D		A		M		D		A		M	
		,		,		,		,		,		,
	D		A		M		D		A		M	
		,		,		,		,		,		,
	D		A		M		D		A		M	
		,		,		,		,		,		,
	D		A		M		D		A		M	
		,		,		,		,		,		,

AN 8 MUST BE IN COLUMN 1 OF LAST DATA CARD OF EACH RECORD.

TRIXY INPUT DATA

CURRENT BOUNDARY CONDITIONS

Cross Section Data - Record 4

col. 1	ADDRESS		Z^1_{2001}		Z		Z		Z		Z		Z	
C	3100	,		,		,		,		,		,		,
	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,
	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,
	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,
	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,
	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,

Example:

$Z^{g-1}_{l} = 2001$

AN 8 MUST BE IN COLUMN 1 OF LAST DATA CARD OF EACH RECORD

TRIXY INPUT DATA

CURRENT BOUNDARY CONDITIONS

Cross Section Data - Record 4

col 1	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,
	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,
	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,
	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,
	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,
	Z		Z		Z		Z		Z		Z		Z	
		,		,		,		,		,		,		,

AN 8 MUST BE IN COLUMN 1 OF LAST DATA CARD OF EACH RECORD.

APPENDIX B· GENERATION OF LIBRARY TAPE

DECK MAKE-UP

GE BILD 1		
GE BILD 2		
PK1	(75 cards)	
LTW	(12 cards)	
TRXSLOAD	(1 card)	
PK1	(75 cards)	
MT-1	(71 cards)	(3000 words)
	Blank card	
G.SOR	(86 cards)	(2878 words)
	Blank card	
SWEEP	(92 cards)	(1825 words)
	Blank card	
MT-3	(67 cards)	(1559 words)
FISS	(73 cards)	(1605 words)
	Blank card	
EDIT	(139 cards)	(3079 words)
	Blank card	
MT-2	(110 cards)	(2728 words)
	4 blank cards	

OPERATING INSTRUCTIONS

Place blank tape on unit 3. Sense switches 2 and 3 must be down. CLEAR and LOAD CARDS; final stop $(3316)_8$.

DESCRIPTION

Errors in reading cards or writing tape are indicated on the printer. As each program is written on tape, the number of words in the program is printed out. If this does not check with the word count given above, an error in deck make-up is indicated. This program loads both SAP and CAGE binary cards. It does not continue if a card with an incorrect check sum is discovered.

APPENDIX C: GENERATION OF CONFIGURATION AND CROSS SECTION DATA TAPE

DECK MAKE-UP

Binary Cards

1. BILD I.
2. BILD II.
3. GEPRO (71 cards). This is a special copy of GEPRO.
4. Tape 4 (21 cards).
5. Transfer card $(3173)_8$.

Decimal Cards

1. Configuration data cards. One configuration per record.
2. End of File card (a blank card with an 8 in column 1).
3. Cross Section Data cards for File I.
4. End of File card.
5. Subsequent Cross Section Data cards for respective files. Each file followed by an EOF card.
6. Two End of File cards follow the last Cross Section file.

OPERATING INSTRUCTIONS

If the data cards are put on BCD tape, the tape must be put on unit 2. Regardless of whether tapes or cards are used, the output will be on tape unit 4.

Sense switch 2 must be UP if data are on tape, DOWN if on cards. Sense switch 3 must be DOWN.

The following errors are detected by this Deck:

1. Incorrect number of words in a cross section record.
2. Incorrect order of cross section files.
3. Cross section A equals zero.
4. Incorrect placement or omission of End of File card.
5. The number of words in record 2 is not a multiple of 4G.
6. The number of words in record 3 is not a multiple of 3G.
7. The number of words in record 4 is not a multiple of G.
8. Failure of tape 4 to write correctly.
9. Any mispunched data cards.

If any of these errors occur, corrections must be made and then the program re-loaded. Final stop $(3446)_8$.

APPENDIX D: PREPARATION OF A NEW FLUX GUESS TAPE

DESCRIPTION

This program to prepare a new flux guess tape has as input tape E from a previous run and record 1 of the input for a new case. It generates a new flux guess tape with the history records corrected to correspond to the new input. The f file is omitted from the new tape. Only G files of flux are copied onto the new tape. Since the files containing the rough source guess and flux are copied directly, the following words of input for the two cases must be the same:

$$I,\ J,\ K,\ i^{F}_{min},\ i^{F}_{max},\ j^{F}_{min},\ j^{F}_{max},\ k^{F}_{min},\ k^{F}_{max},\ \text{and RCW.}$$

The G for the new case cannot be greater than the G for a previous case.

DECK MAKE-UP

Binary cards:

BILD I

BILD II

PK1 (75 cards).

Tape 5 (12 cards).

Decimal cards:

Record one of the input for the new case (16 cards)

OPERATING INSTRUCTIONS

The results from the previous case must be placed on tape 6. The new tape is generated on tape 5. Sense switches 2 and 3 must be down. Any stop except the final stop $(3334)_8$, indicates either machine error or errors in the decimal cards. There should be a print-out which indicates the type of error.

APPENDIX E OPERATING MANUAL

INTRODUCTION

The purpose of this manual is to provide operating instructions for the TRIXY program. It is suggested that this operating manual be kept in a loose leaf notebook for convenient reference while operating the program.

The operation of TRIXY differs from standard operating procedures used by most installations. Hence, it is recommended that all persons concerned with programming, preparation of input, and machine operation of TRIXY read this description.

GENERAL DESCRIPTION

TRIXY is designed to run for several hours. Therefore, it is important that the operating instructions and the instructions from the customers be followed carefully. All tape units must be cleaned in preparation for running, and only full reels of tape should be used. Comments are printed to advise the operator of the progress of the program and to provide instructions whenever intervention by the operator is required.

The TRIXY Library Tape should be placed on unit 1 and the configuration and Cross Section Tape on unit 4. If the input is on (BCD) tape, this should be placed on unit 5. Blank tapes should be placed on units 2, 3 and 6. If the input is on cards, the deck should be placed in the card reader and sense switch 2

depressed. "OMEGA" data should be placed in the card reader behind the input. If input is on tape, the OMEGA cards should be placed directly in the card reader. If the customer desires a Control Restart deck, depress sense switch 4. The program stops after the deck is punched to allow the control rod positioning to be checked. This deck can also be used to restart the master tape generation. If the deck is to be used for restart only, then the program should be continued by pushing START.

After the input has been read, the program reads from tape 4 and writes on tape 3. Several passes are made to sort from tape 3 to tape 2. This operation may take more than an hour. When this section is finished, the control restart deck is punched if requested. Information is transferred from tape 1 to 5. The machine stops and tapes 1 and 4 must be replaced as indicated by instructions on the printer. At this point, specific sense switch settings must be made to control the operation of the program. It is recommended the sense switches be set so that a duplicate of the master tape is generated and monitor printing is provided during the Group Sweep. (Group Sweep printing does not take extra time, except for extremely small cases (IJK < 1000). Printing is done while a tape is rewound.)

The generation of the master tape is performed Data is computed and written on tapes 1, 3, 4 and 6 (if a duplicate of the master tape is being written.) The group number is printed and data transferred from tapes 1 and 5 to 3 (and

possibly 6). This process is repeated for each group. At the end of the master tape generation, data are transferred from tapes 1, 2, 4 and 5 to 3 (possibly 6).

If a rough source guess is required, it is computed, and tapes 4 and 5 are used. This computation is performed while rewinding the master tape (s). The machine stops. If only one master tape has been generated, it is possible to provide a copy of the master tape by pushing START. If a copy is not wanted, put sense switch 1 up and push START.

The Group Source Program is read from the master tape and control is transferred to this program. In case a restart procedure is being executed, the Group Source Program positions the tapes properly, reads in the Group Sweep Program, and transfers control. If a restart procedure is not being executed, values of the scattering coefficient (H) for one plane are read into memory from the master tape (tape A). A plane of iterated flux (ϕ) is read from tape B, and the values of ϕ are transferred to tape E for the Group Sweep Program. $H\phi$ is calculated and stored on tape C. This process is repeated for all planes. (This first step is omitted for the first group.)

A plane of fission source (δ) is read from tape E. The values are multiplied by χ^g and, for groups other than the first, are added to the corresponding value of $H\phi$ from tape C to determine the group source (S). Blocks of ErrC coefficients are read in from tape A, combined with values of S, and written on tape D as blocks of SErrC's. If a flux guess is not available

for the succeeding group, ϕ is calculated from S and E and written on tapes B and E. This process is repeated for all planes.

The Group Sweep Program is read from the master tape and control is transferred to it.

At the beginning of the Group Sweep Program, one line is printed. This line is printed at the end of each sweep and again when convergence is reached. During each sweep, values for the flux (ϕ) are read from either tape B, C, or F and written on one of these tapes. Values of SErrC are read from tape D. At the end of each sweep, tapes B, C, D, and F are rewound while the line is printed. When convergence is reached, the Group Source Program is read from the master tape.

Control alternates between Group Source and Group Sweep until the Fission Source Program is reached. Fission Source reads data from the master tape, tape E (the tape containing the flux and fission source guesses), and a flux tape prepared by Group Sweep. One "OMEGA" card is read in from the card reader. A new tape E is prepared. Information concerning the convergence and the tape number of the new tape E is printed, and the machine stops. At this point, control is turned over to Group Source, unless the fission source has converged or an Edit is requested by depressing switch 6.

If control is turned over to the Edit Program the tape to be used for

output is indicated on the printer. This program first sorts the computed results from tape E and the master tape onto the working tapes; then it writes the results on the output tape. This output may take more than one reel. If more than one reel is needed, the machine should be put on manual before the end of the tape is reached. This causes the machine to stop (not a programmed stop). An end of file should be written on the output tape, another output tape turned on, and the machine started again. The end of TRIXY is indicated on the printer.

NORMAL OPERATING PROCEDURE

1. Arrange the tapes as follows:

Unit	Description
1[1]	TRIXY Library Tape
2	blank
3	blank (labeled master tape)
4[1]	Configuration and Cross Section Tape
5	(BCD) input or blank
6	blank (duplicate of master tape)

2. Set sense switches as follows:

1	up

[1]This tape must have the file protect ring removed.

(Set sense switches continued)

2	up	input on tape 5
	down	input in card reader
3	up	
4	up	no restart deck
	down	a Control restart deck is requested
5	up	
6	up	

3. Place the input in the card reader (if it is not on tape).

4. Place the OMEGA deck in the card reader.

5 Push CLEAR and LOAD TAPE

Following is a listing of the printouts and stops which occur during a normal run of TRIXY.

Master Tape Generation

Stop	Printout
None	"The I Planes of 'Composition Indices are Finished'."
161	"The Tapes have been set up ---."
	At this point switches 2 and 3 only should be depressed unless otherwise instructed.
None	The group number is printed out as each group is finished.

Stop	Printout
161	"A copy of the master tape is on unit 6. Replace and save tape 6." It is also recommended that the file protect ring be removed from tapes 3 and 6.
None	"Remove Deck from Punch. -------"

Group Source Program

No stops or printouts.

Group Sweep Program

A line of information is printed at the beginning of the Group Sweep Program. At the end of each sweep, another line is printed. When the group has been finished, two lines are printed in rapid succession.

Fission Source Program

Stop	Printout
52	"End of Run --------". Several lines of information are printed. Unless Edit is requested by depressing switch 6 or the iterations have converged the program returns to the Group Source Program

Edit Program

Stop	Printout
6253	"Tape _____ is free. Set a tape 10 for output. Depress start."
4323	"End --- Save and label tape _____ = A and Tape _____ = E."

Other Stops

Stop	Description
161	Master Tape Generation. If V_F is outside tolerance, follow instructions from customer. In all other cases, follow instructions on the printer.
52	Group Source or Group Sweep. Follow instructions on the printer. If there is no printout, record index register C and push start. If the stop occurs again, display index register C and, use Emergency Restart.
2106	Group Sweep. Use Emergency Restart.
2315	Group Sweep. See Restart procedure.

Stop	Description
4323	Edit. The end. Follow instructions on the printer.
6253	Edit. Follow instructions on the printer.
11646	Re-edit. Restart Re-edit.
any others	Machine error. If a stop occurs before the Emergency Restart deck has been punched, use the Control Restart, if available; otherwise, start TRIXY over from the beginning. If this stop is not in Edit, use the Emergency Restart. If this stop is in Edit, save and label the master tape and tape E for Re-edit. List the output tape (tape 10).

UNUSUAL OPERATING PROCEDURE

Loading New Liebmann Factor

1. Punch card for new α as follows:

   .29, α , (Control punch must be a "." - end of record, floating point. Do not use an end of file card.)

2. Put deck for MDL9C followed by data card for α in card reader. Do

not put BILD on front of MDL9C deck. The MDL9C deck contains its own load routine, MBILD, as the first two cards of the deck. Deck is numbered MDL9CN01 through MDL9CN44.

Caution: There are three different pre-assembled decks for MBILD for the "8000", "16,000" and "32,000" word memory. Be sure to use the proper deck. The identification "N" punched in column 78 of the MDL9C deck is N=1 for "8000", N=2 for "16,000", and N=3 for "32,000.

3. Ready card reader.

4. Put sense switch 4 down.

5. After data have been loaded, the program stops at $(52)_8$. Flow instructions on printer. Put sense switch #4 up before pressing start.

RESTART PROCEDURES

A. Restart Procedure (for Group Sweep).

To interrupt TRIXY in the middle of Group Sweep:

1. Put sense switch 5 down.

2. When stop occurs at $(2315)_8$, follow instructions on printer for removing TRIXY from machine.

NOTE - "Marking each tape reel by number" means that the tape reel on unit 1 should be marked 1, 2, 3, etc.

To Restart TRIXY:

1. Replace the tape reels removed from the machine and set the numbers on the units to those indicated by the markings on the tape reel (see above).
2. Check to be sure step 1 has been done properly.
3. If this is the first time TRIXY has been restarted, set the tape unit that is now tape 1 to tape 3, and set the tape unit that was tape 3 to tape 1.

   CAUTION - If this is not the first time TRIXY has been restarted, omit this step and leave the tape numbers as they are.
4. Place the card punched out when TRIXY was interrupted, the MDL9C deck, and the proper Liebmann factor (α) in the card reader. The value for α must be read from the printer output of the Group Sweep Routine (first floating point number printed on a line) and punched on a special card.
5. Ready the card reader.
6. Set the sense switches for normal operation of TRIXY (see "Normal Operating Procedure" p. 159).
7. Put sense switch 4 down (α must be loaded).
8. Press "CLEAR" and "LOAD TAPE".

B. Emergency Restart (for Group Source or Sweep)

This restart requires that the master tape and tape E be saved. The master tape is on unit 3 unless the Restart described in Restart Procedure, page 164, has been used. In this case the master tape is on unit 1. Check the printer to see if a run has been completed. If a run has been completed, the number of the new tape E is indicated in the printout from the end of the last run. If a run has not been completed, tape E is on unit 5.

1. Set the master tape to 3 and tape E to 5.
2. Remove the "OMEGA" deck from the card reader. Take one card for each run which has been completed from the front of the deck.
3. Place the remainder of the "OMEGA" deck in back of the emergency restart deck.
4. This combined deck goes into the card reader.
5. Push CLEAR and LOAD CARDS.

If for any reason it is impossible to continue, save and label the master tape and tape E; save the Emergency Restart and "OMEGA" decks. To restart, place the master tape and tape E on the machine and follow instructions 1. ---- 5. under Emergency Restart.

C. Control Restart (for Master Tape Generation)

This restart requires that tapes on units 1 and 2 be saved.

1. Place tapes from units 1 and 2 on units 1 and 2 respectively.

2. Provide blanks on units 3, 4, 5, and 6.

3. Place the Control Restart deck in the card reader.

4. Push CLEAR and LOAD CARDS.