

UNCLASSIFIED

AD NUMBER
AD858689
NEW LIMITATION CHANGE
TO Approved for public release, distribution unlimited
FROM Distribution authorized to U.S. Gov't. agencies only; Administrative/Operational Use; NOV 1960. Other requests shall be referred to Air Force Rome Air Development Center, Attn: EMAEI, Griffiss AFB, NY.
AUTHORITY
radc, usaf ltr, 8 jan 1976

THIS PAGE IS UNCLASSIFIED

THIS REPORT HAS BEEN DELIMITED
AND CLEARED FOR PUBLIC RELEASE
UNDER DOD DIRECTIVE 5200.20 AND
NO RESTRICTIONS ARE IMPOSED UPON
ITS USE AND DISCLOSURE.

DISTRIBUTION STATEMENT A

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED.

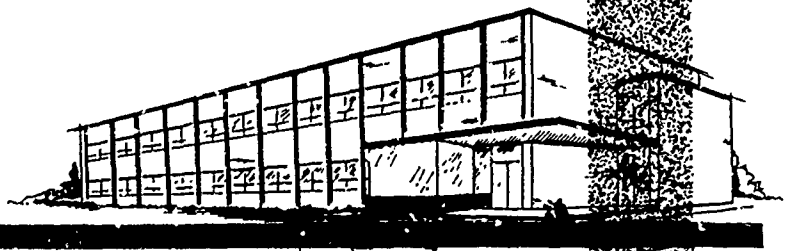
RADC-TR-59-167C, SUPP 1

BSC 33119-

Copy No. _____



AD858689



THE *Bendix* CORPORATION

BENDIX SYSTEMS DIVISION • ANN ARBOR, MICHIGAN

SEP 17 1969

60-S-13908A

249

AD858689

UNCLASSIFIED SUPPLEMENT
TO
COMPACT FINAL REPORT
ON PHASE II

BSR-108

VOLUME III SIMULATION STUDIES
November 1960

Approved by: Ernest S. VanValkenburg
Ernest S. VanValkenburg
Project Supervisor

BENDIX SYSTEMS DIVISION
OF
THE BENDIX CORPORATION
Ann Arbor, Michigan ✓

AF 30(602) -1796 ✓

Prepared for
Rome Air Development Center
Air Force Systems Command
United States Air Force
Griffiss Air Force Base
New York

attw:EMAE I

Each transmittal of this
U.S. Government must have

STANDARD FORM

SEP 1 1960

TABLE OF CONTENTS

	<u>Page</u>
1. STROBE TRIANGULATION FORTRAN PROGRAM	1-1
1.1 SECTION I - ENVIRONMENTAL PROGRAM	1-1
1.1.1 Input/Output	1-1
1.1.2 Glossary	1-6
1.1.3 FORTRAN Listing	1-12
1.2 SECTION II - TRIANGULATION PROGRAM	1-19
1.2.1 Input/Output	1-19
1.2.2 Glossary	1-21
1.2.3 FORTRAN Listing	1-27
1.3 SECTION III - DATA REDUCTION	1-39
1.3.1 Input/Output	1-39
1.3.2 Glossary	1-44
1.3.3 FORTRAN Listing	1-50
2. FORTRAN PROGRAM FOR SIMULATION OF THE COR- RELATION SYSTEM.	2-1
2.1 ORIGINAL PROGRAM	2-5
2.1.1 Glossary	2-5
2.1.2 FORTRAN Listing	2-16
2.2 ENVIRONMENT GENERATOR MODIFICATION FOR MANEUVERING TARGETS	2-49
2.2.1 Glossary	2-49
2.2.2 FORTRAN Listing	2-51
2.3 MODIFICATION TO ACCEPT NEW ENVIRONMENT GENERATOR	2-65
2.3.1 FORTRAN Listing	2-65

TABLE OF CONTENTS (CONT.)

	<u>Page</u>
2.4 MODIFICATION TO OBTAIN MISS-DISTANCE	2-97
2.4.1 Glossary	2-98
2.4.2 FORTRAN Listing	2-99
2.5 RAID SIZE ESTIMATION PROGRAM	2-123
2.5.1 Glossary	2-124
2.5.2 FORTRAN Listing	2-125
2.6 HISTORICAL DATA PROGRAM	2-161
2.6.1 Glossary	2-161
2.6.2 FORTRAN Listing	2-166

LIST OF ILLUSTRATIONS

<u>Figure</u>	<u>Title</u>	<u>Page</u>
1-1	Triangulation Section II General Flow Diagram	1-37/1-38
2-1	Original Program Flow Diagram	2-29 to 2-48
2-2	Environment Generator Modification Program Flow Diagram	2-55 to 2-64
2-3	Accept New Environment Generator Program Flow Diagram	2-77 to 2-96
2-4	Flow Diagram Error Measures	2-107 to 2-122
2-5	Modifications to Basic Program for Raid Size Estimation	2-139 to 2-160
2-6	General Flow Charts	2-173 to 2-174
2-7	Historical Data Analysis Flow Charting	2-175 to 2-184

SECTION 1 STROBE TRIANGULATION FORTRAN PROGRAM

The strobe triangulation simulator is programmed in three sections. Section 1 provides simulation of the target signals as they would appear to any azimuth-scan radar. Section 2 simulates the triangulation processes, and Section 3 provides data reduction on the output of Section 2.

This portion of the report presents the input/output format, glossary of terms, FORTRAN listing, and flow charts by Program Section

1.1 SECTION 1 - ENVIRONMENT PROGRAM

1.1.1 Input/Output

Input parameters necessary for each computer run are listed and described below:

- | | |
|------------|---|
| <u>NDS</u> | Number of data sets where a data set consists of all the parameters which follow. NDS is an integer and is punched in Columns 1 through 5. |
| <u>IM1</u> | Intermediate Output One. If IM1 is zero, run, clock, unsorted azimuths, associated range functions and radar number will be printed out. If IM1 is one, the output just described is bypassed. |
| <u>IM2</u> | Intermediate Output Two. The output data is the same as for IM1 except the azimuths have been sorted in ascending order. Output control is also the same. |
| <u>IM3</u> | Intermediate Output Three. This output provides data regarding detection criterion. NOL (see Glossary of Terms), T (I), S, AI (I), BI(I), SUMJ, SUMK, SNR, SR (K), PD (K), RN, ST (L), K and L are printed out, if IM3 = 0, and bypassed, if IM3 = 1. |

- IM4 Intermediate Output Four. The output consists of a table of detected azimuths, if IM4 = 0, and the output is bypassed, if IM4 = 1.
- ID Run number, an integer from one to one thousand.
- NO Number of targets, an integer from one to one hundred.
- NR Number of radars, an integer from one to five.
- IFL Number of flight legs. Flight legs are counted as follows: Assume a flight of four targets identified as numbers one through four. Assume that the four fly parallel courses initially. Suppose that at frame 10 (clock 10) target one starts a diversionary maneuver but targets two, three and four continue on their original courses. Suppose at frame 25, targets three and four break off from their original course and each assume new courses which are not parallel. For this example IFL = 3. Thus flight legs are counted on a group basis. Even though targets three and four took up new courses at clock 25 this maneuver counts as only one flight leg as far as determining IFL is concerned. IFL is an integer from one to two hundred.
- MAXLIM Maximum number of frames in any one run. MAXLIM is an integer from one to one hundred.
- SCALE Scale factor for the off-line display. The scale factor is in nautical miles.
- RFL Range function limit. RFL is computed as follows:
- $$RFL = \frac{1}{1.44h}$$
- where h is the altitude in feet of the attacking targets.
- AG Antenna gain.

X(I) Initial x coordinate (frame 0) of target I, in nautical miles.

Y(I) Initial y coordinate (frame) of target I, in nautical miles. Note: Only the x, y coordinates of one target are placed on a card. Furthermore, the targets will be identified by number starting with one in the order in which the cards are read.

RX(I) X coordinate of radar I, in nautical miles.

RY(I) Y coordinate of radar I, in nautical miles. Radars are also identified by number starting with one in the order in which the radar cards (one radar coordinate set per card) are read.

PD(I) Probability of detection.

SR(I) Signal Ratio.

ET Error term in radians. This is the "peak to peak" range throughout which a true azimuth may be jittered to simulate random azimuth error.

S One half of the passive main beam width, in radians.

Q Main beam width, in radians, ET, S and Q are read in from one card.

A Number of sequential frames for which computations are to be done. If A is negative then B frames are skipped before A frames are computed. A is an integer.

B Number of frames that are to be skipped.

LIMIT Frame number at which a new flight leg occurs, or at which it is desirable to change A and/or B. LIMIT is an integer from one to one hundred. A, B and LIMIT are read from one card.

DX (I) X increment for target I motion each frame in nautical miles.

DY (I) Y increment for target I motion each frame, in nautical miles. DX (I) and DY (I) for target I are read from one card. Note: The card order is very important since the increments are ordered numerically as read in starting with one. Thus, for example, the increment card for target 3 must be the third increment card so that it will be associated properly.

There are three general types of output; these are intermediate data, display data, and environment data.

1. 1. 1. 1 Intermediate Data

Intermediate data is primarily for check-out purposes and is put on Tape Unit 6 for tabulating. There are four sets of intermediate data each associated with an IM variable which is an input parameter. Run, clock, unsorted azimuths, range functions and radar number are associated with IM1. If IM1 is zero, output is obtained; if IM1 is a non-zero integer from one to 9999, no IM1 output is obtained.

IM2 output is comprised of the same variables as IM1 except that the azimuths have been ordered. The column titles are not printed as they are in IM1, but IM2 output is identified with the title "SORTED AZIMUTHS". The control for IM2 output is the same as for IM1 output and other IM outputs, which will be described in the paragraphs to follow.

IM3 Output provides information about strobe detection. The following variables comprise IM3: NOL, T(I), AI(I) (trailing edge of main beam), BI(I) (leading edge of main beam), SUM J, SUM K, SNR, SR(K), PD(K), RN, ST(L), K and L.

There are several idiosyncrasies about IM3 output which should be explained in order to avoid confusion in interpreting the data. First of all, note that AI(I), BI(I), SNR, and RN are never cleared; they are simply replaced as the occasion demands. Thus, whenever T(I) is negative (either -1. or -2.) AI(I), BI(I), SNR and RN are not computed so that whatever values are stored in AI(I), BI(I), SNR and RN remain there and are outputted with the associated value of negative T(I). Therefore, disregard the values of AI(I), BI(I), SNR and RN whenever T(I) is negative. Second, even though SUM J and SUM K are cleared, this clearing is done just before a new SUM J and SUM K are to be computed. Thus, when T(I) is negative SUM J and SUM K are not cleared and the latent values will be outputted. Therefore, disregard the values of SUM J and SUM K whenever T(I) is negative. Third, if SUM K is 0. the computation of SNR is bypassed so that latent value will be printed as well as that of RN. Fourth, the variables PD and SR are indexed by K and ST by L. L is not incremented unless an azimuth (strobe) is detected so the latent value of ST(L) will be printed. Therefore ST(L) is valid only when L changes and is not zero. K is incremented only for finite SNR. Thus the values of PD(K) and SR(K) are valid only when K= 1, 2, 3 or 4.

IM4 Output is a list of detected strobcs. Since the ST(L) table is placed into the first L positions in the T(I) table after the detection process is complete and since detected strobcs are printed out in groups of four only the first L entries in the 4n of the printed table will be valid (n is the number of lines in the table). For example, suppose there nine targets and only seven strobcs detected. Then two lines would be printed but only the first seven entries would be valid. The eighth azimuth would be an undetected strobe or possibly a detected one which had been "moved up". That is, perhaps strobcs 1, 2, 3, 5, 6, 7 and 8 were detected, then strobcs 1, 2, 3, 5 would be printed in the first line and 6, 7, 8, 8 in the second line.

1.1.1.2 Display Data

Three types of output are provided for display purposes. These are radar positions for each run (type 16) true target positions for each

frame (type 11) and azimuth in degrees and the direction points SINM COSM for each detected strobe for each radar for each frame (type 12). All display data is put on Tape Unit 5 and decimal cards are punched from the tape.

1. 1. 1. 3 Environment Data

There are eight types of environment data all of which are put on Tape Unit 4. Control output (no type number) occurs only once per run and consists of the run number, number of targets, number of radars and maximum limit of the clock. No titles are used. Radar position output (type 16) is identical in every respect to that used for display purposes. Similarly for true target position output (type 11). Index output (type 14) occurs once per radar per frame if the given radar detected stobes. The index is a count of the number of stobes detected by that radar. Sector data output occurs once per radar per frame if the given radar detected any stobes. Sector data consists of the number of targets in a sector and the leading and trailing edges of the sector. Run length and index output (type 17) is redundant in that it combines type 13 and 14 in slightly different form. Azimuth output (type 18) is a list of the center azimuth for each strobe for a given radar for a given frame. NRD output (type 19) has the same format as index output but the index is replaced with a count of the number of radars which have made at least one strobe detection in that frame.

1. 1. 2 Glossary

<u>A</u>	Number of frames to be computed. (If A is negative then B frames are skipped before A are computed.)
<u>AG</u>	Antenna gain expressed as a dimensionless ratio.
<u>AI (I)</u>	Trailing edge of main beam when main beam is centered on T (I).
<u>ALPHA (I) or ALPHA J</u>	Azimuth of the leading edge of a target where I or J is the running index on the variable.
<u>B</u>	Number of frames to be skipped.

<u>BETA (I) or BETA (J)</u>	Azimuth of the trailing edge of a target where I or J is the running index on the variable.
<u>BI (I)</u>	Leading edge of main beam when main beam is centered on T (I).
<u>C (I)</u>	W (I) converted to degrees for use with off-line Bendix display facility.
<u>CES (I)</u>	A variable generated for use by the off-line Bendix display facility. The variable is indexed by I. See FORTRAN statement following number 122 for a definition of CES (I).
<u>CLK</u>	Title for the frame number.
<u>DX (I)</u>	X increment by which target I is to be moved each frame where I is the running index on the variable.
<u>DY (I)</u>	Y increment by which target I is to be moved each frame where I is the running index on the variable.
<u>ET</u>	Error term. This is the total width in radians of the azimuth interval throughout which the observed azimuth may fall. The error term azimuth interval is centered about the true azimuth.
<u>I</u>	Index
<u>IC, IE</u>	Indices used for azimuths and corresponding range functions in the azimuth sorting routine IE is also set equal to the number of sectors less one and used as the maximum index value in a do loop.
<u>ID</u>	Run number.
<u>IFL</u>	Number of flight legs.
<u>IG</u>	Temporary storage for a particular value of I, an index in the merging routine.

<u>IJ</u>	Index equal to I plus one. IJ is used in the merging routine.
<u>IM1, IM2, etc.</u>	Intermediate Output 1, Intermediate Output 2, etc. If IM1 = 0, then the output identified with IM1 will be printed on Tape Unit 6. If IM1 = 1 then the output identified with IM1 will be bypassed. The primary purpose is to provide output for checkout purposes.
<u>J</u>	Index
<u>JLB</u>	Variable of an assigned "GO TO" statement.
<u>K (I) or K (J)</u>	Number of targets in a sector.
<u>KING</u>	Variable of an assigned "GO TO" statement.
<u>KLOCK</u>	Frame number.
<u>KPLB</u>	Klock plus LB.
<u>KTS</u>	Variable of an assigned "GO TO" statement.
<u>L</u>	Count of number of detections before merging.
<u>LA</u>	Integer variable corresponding to the floating variable A.
<u>LAP</u>	LA or some decremented value of LA.
<u>LB</u>	Integer variable corresponding to the floating variable B.
<u>LIMIT</u>	Either the number of a frame where a new flight leg is to be followed or the maximum number of frames in a given run.
<u>MAR</u>	Variable for output data type code. It takes the value 16 and indicates radar position output.
<u>MARG</u>	A fixed point variable for identification.

MAXLIM Maximum number of frames in a given run.

MT Variable for output data type code. The only value MT assumes is 18 which is the numerical type designation for azimuth output.

MX Variable for output data type code. The values and their significance are listed below:

- MX = 11 Target output
- = 12 T (I) in degrees and direction point output for off-line Bendix display
- = 13 Sector leading and trailing edge and number of targets in sector output
- = 14 Radar number and index output
- = 17 Azimuth run length and number of targets in sector output
- = 19 Number of radars making detections in given frame

MY Variable for type of target output. If MY = 6 the target output is true targets. Since only true targets are produced by Section I, MY will assume no other value in Section I.

NDS The number of data sets (runs) plus one. For example, suppose it is desired to make runs 4, 5 and 6 at one time. Then NDS = 4. If run 7 only is to be made, then NDS = 2.

NIL Nil - nothing.

NO Number of targets.

NOL A variable generated in the detection routine and used as the starting index on the SUMJ and SUMK do loop within the detection routine.

NR Number of radars in a given run.

NRD Number of radars making detections in a given frame.

NX Variable of an assigned "go to" statement.

PD (I) or PD(K) Probability of detection where I or K is the running index on the variable.

PI π to eight decimal places.

Q Beamwidth in radians.

RF (I) Range function. The range function is defined as the reciprocal of the square of the range of target I from radar IR.

RFL Range function limit.

RN A random number generated by routine RAM 2BF (0).

RX (I) or RX (IR) X coordinate of radar number I or IR where I and IR are running indexes on the variable.

RY (I) or RY (IR) Y coordinate of radar number I or IR where I and IR are running indexes on the variable.

S One half of the beamwidth, Q

SCALE Scale factor. This scale factor is used for an off-line Bendix display.

SEN (I) A variable generated for use by the off-line Bendix display facility. The variable is indexed by I.

SNR Signal to noise ratio. SNR is defined as follows:

$$\text{SNR} = \frac{\text{AG (SUMJ)}}{\text{SUMK}}$$

SR (I) or SR(K) Signal ratio where I or K is the running index on the variable.

<u>ST (L) or ST (I)</u>	Temporary storage for azimuths which have been detected but not merged where L or I are the running indices.
<u>SUMJ</u>	The sum of the range functions of the targets within the main beam when looking at target I as represented by jittered azimuth T (I).
<u>SUMK</u>	The sum of the range functions of the k targets in the sidelobes when the main beam is looking at target I as represented by the jittered azimuth T (I).
<u>T (I) or T (J)</u>	The azimuth of target I from radar IR. The azimuth is measured with the positive y axis as reference and with clockwise rotation as the positive sense. J is also a running index on the variable.
<u>TC</u>	Temporary storage for C (I).
<u>TEMPR</u>	Temporary storage for the range function corresponding to the azimuth in TEMPT.
<u>TEMPT</u>	Temporary storage for one azimuth in the azimuth sorting routine.
<u>TPI</u>	2π to eight decimal places.
<u>TRG</u>	Title for target number.
<u>TT or T</u>	Title for Output Data Type Code.
<u>TW</u>	Temporary storage for W (I).
<u>V (I)</u>	Target run length or sector width, i. e., the azimuth difference of the trailing edge less the leading edge.
<u>W (I)</u>	Center azimuth of a sector with the leading edge, ALPHA (I), and trailing edge, BETA (I). I is the running index on the variable.

X (I) X coordinate of target number I where I is a running index on the variable.

Y (I) Y coordinate of target number I where I is a running index on the variable.

1.1.3 FORTRAN Listing

```
DIMENSION X(100),Y(100),PD(4),SR(4),RX(8),RY(8),DX(100),DY(100),
XSEN(100),CES(100),T(100),K(100),ALPHA(100),BETA(100),W(100),
XAI(100),BI(100),ST(100),V(100),C(100),RF(100),ET(5),S(5),Q(5)
REWIND 4
READ                               600,NDS
600 FORMAT(I5)
    PI=3.14159265
    TPI= 6.28318530
    IFL=0
    NO=100
    J=100
    LIMIT=0
    KLOCK=1
    DO 137 I=1,8
    RX(I)=0.
137 RY(I)=0.
    DO 144 I=1,100
    X(I)=0.
144 Y(I)=0.
    IF DIVIDE CHECK 670,670
670 IF QUOTIENT OVERFLOW 138,138
69 NDS=NDS-1
    IF(NDS)601,601,602
601 END FILE 4
    REWIND 4
    PAUSE 209
    PAUSE
    PAUSE1
    PAUSE
    PAUSE 100
602 READ                               650,IM1,IM2,IM3,IM4
650 FORMAT(4I4)
    READ 1,ID,NO,NR,IFL,MAXLIM,SCALE,RFL,AG
    1 FORMAT(5I5,F7.2,F10.8,F8.2)
    WRITE OUTPUT TAPE 4,55,ID,NO,NR,MAXLIM
55 FORMAT (4I15)
C   TARGET INPUTS
    READ                               2,(X(I),Y(I),I=1,NO)
    2 FORMAT(2F7.2)
```

```

C   DETECTOR LOCUS
    READ                2,(RX(I),RY(I),I=1,NR)
    NIL=0
    KLOCK=0
    MAR=16
    DO 1000 I = 1,NR,4
    WRITE OUTPUT TAPE 4,26,(ID,KLOCK,MAR,NIL,RX(I),RY(I),RX(I+1),
    IRY(I+1),RX(I+2),RY(I+2),RX(I+3),RY(I+3))
    IF(SENSE SWITCH 1)93,1000
    93 PUNCH                26,(ID,KLOCK,MAR,NIL,RX(I),RY(I),RX(I+1),
    IRY(I+1),RX(I+2),RY(I+2),RX(I+3),RY(I+3))
    1000 CONTINUE
C   PROBABILITY AND SR TABLES
    READ 3,                (PD(I),I=1,4)
    3 FORMAT(4F5.2)
    READ                4,(SR(I),I=1,4)
    4 FORMAT(4F5.0)
C   PROGRAM CONSTANTS
    DO 350 I = 1,NR
    350 READ 5,(ET(I),S(I),Q(I))
    5 FORMAT(3F6.3)
    GO TO 70
    7 IFL=IFL-1
    IF(IFL) 69,69,70
C   VARIABLE INPUT
    70 READ                6,(A,B,LIMIT)
    6 FORMAT(2F4.0,I5)
    READ                2,(DX(I),DY(I),I=1,NO)
    LA=A
    LB=B
C   CHECK MODE CONTROL
    IF(A)8,12,12
    8 KPLB=KLOCK+LB
    IF(LIMIT-KPLB)151,152,152
    151 B=FLOATF(LIMIT-KLOCK)
    KLOCK=LIMIT
    GO TO 153
    152 KLOCK=KLOCK+LB
    153 DO 9 I=1,NO
    X(I)=X(I)+B*DX(I)
    9 Y(I)=Y(I)+B*DY(I)
    LA=ABSF(A)
    12 LAP=LA
    65 KLOCK=KLOCK+1
    IF(LIMIT-KLOCK)150,99,99
    150 KLOCK=KLOCK-1
    GO TO 7
    99 ASSIGN 28 TO JLB

```

```

NRD=0
IR=1
DO 13 I=1,NO
X(I)=X(I)+DX(I)
13 Y(I)= Y(I)+DY(I)
C CALCULATION OF RANGE FUNCTION AND AZIMUTH
14 IRP=IR
DO 15 I=1,NO
RF(I)=1./((X(I)-RX(IRP))**2.+(Y(I)-RY(IRP))**2.)
IF DIVIDE CHECK 23,115
115 IF(RF(I)-RFL)624,835,835
835 T(I)=ATN1F((RX(IR)-X(I)),(Y(I)-RY(IR)))
IF(T(I)) 21,21,123
123 T(I)=TPI-T(I)
21 RN=RAM2BF(0)
T(I) = T(I)+ET(IRP)*(RN-.5)
IF(T(I)-TPI)18,19,19
19 T(I)=T(I)-TPI
GO TO 15
18 IF(T(I))20,15,15
20 T(I)=T(I)+TPI
GO TO 15
23 T(I)=-1.
GO TO 15
624 T(I)=-2.
15 CONTINUE
IF(IM1)500,501,500
501 WRITE OUTPUT TAPE 6,502
502 FORMAT(38H1 RUN CLK T(I) RF(I) R)
DO 503 I=1,NO
503 WRITE OUTPUT TAPE 6, 513,ID,KLOCK,T(I),RF(I),IR
513 FORMAT(15,I4,F8.3,F18.8,I3)
500 GO TO JLB,(28,27)
C TARGET OUTPUT
28 I=1
MY=6
MX=11
MARG=-3
DO 25 I=1,NO,4
MARG=MARG+4
IF (SENSE SWITCH 1) 1001,25
1001 PUNCH 62,ID,KLOCK,MX,MY,MARG,X(I),Y(I),X(I+1),Y(I+1),
XX(I+2),Y(I+2),X(I+3),Y(I+3)
62 FORMAT (15,I4,I3,2I2,8F6.1)
25 WRITE OUTPUTTAPE 4,26,(ID,KLOCK,MX,MARG,X(I),Y(I),X(I+1),Y(I+1),
XX(I+2),Y(I+2),X(I+3),Y(I+3))
26 FORMAT( 15,I4,I3,I4,8F8.2)
ASSIGN 27 TO JLB

```

```

C   ORDERING AZIMUTHS (AND ASSOCIATED RANGES)
27  IC=1
34  IE=IC+1
35  IF(T(IC)-T(IE))36,36,39
36  IE=IE+1
    IF(IE-NO)35,35,37
37  IC=IC+1
    IF(IC-NO)34,699,699
39  TEMPT=T(IE)
    TEMPR=RF(IE)
    T(IF)=T(IC)
    RF(IF)=RF(IC)
    T(IC)=TEMPT
    RF(IC)=TEMPR
    GO TO 36
699 IF(IM2)504,505,504
505 WRITE OUTPUT TAPE 6,506
506 FORMAT(17H SORTED AZIMUTHS)
    DO 507 I=1,NO
507 WRITE OUTPUT TAPE 6,513, ID,KLOCK,T(I),RF(I),IR
C   DETECTION
504 NOL=1
    L=0
    ASSIGN 660 TO KTS
    DO 700 I=1,NO
    ASSIGN 710 TO KING
    IF(T(I))697,698,698
697 NOL=NOL+1
    GO TO 799
698 AI(I) = T(I)-S(IRP)
    IF(AI(I))701,702,702
701 AI(I)=AI(I)+TPI
702 BI(I)=T(I)+S(IRP)
    IF(BI(I)-TPI)703,704,704
704 BI(I)=BI(I)-TPI
703 SUM J=0.
    SUM K=0.
    IF(BI(I)-AI(I))706,706,705
706 ASSIGN 711 TO KING
705 DO 720 J=NOL,NO
725 GO TO KING,(710,711)
710 IF(AI(I)-T(J))707,708,709
707 IF(BI(I)-T(J))709,708,708
711 IF(BI(I)-T(J))712,708,708
712 IF(AI(I)-T(J))708,708,709
708 SUM J=SUM J + RF(J)
    GO TO 720
709 SUM K= SUM K + RF(J)

```

```

720 CONTINUE
    IF(SUMK) 718,718,759
718 L=L+1
    ST(L)=T(I)
    GO TO 799
759 SNR= AG*SUMJ/SUMK
    IF QUOTIENT OVERFLOW 761,758
761 SNR= 0.
758 DO 717 K=1,4
    IF(SNR-SR(K)) 717,716,716
717 CONTINUE
    GO TO 799
716 RN=RAM2BF(0)
    IF(PD(K)-RN) 799,714,714
714 L=L+1
    ST(L)=T(I)
799 IF(IM3) 700,715,700
715 GO TO KTS,(660,665)
660 WRITE OUTPUT TAPE 6,765
765 FORMAT(111H NOL T(I) S AI(I) BI(I) SUMJ SUMK
X SNR SR(K) PD(K) RN ST(L) K L)
    ASSIGN 665 TO KTS
665 WRITE OUTPUT TAPE 6,767,(NOL,T(I),S(IRP),AI(I),BI(I),SUMJ,SUMK,SNR,
XSR(K),PD(K),RN,ST(L)),K,L
767 FORMAT(I5,F6.3,F5.3,2F6.3,3F16.6,2F7.2,F5.3,F6.3,2I5)
700 CONTINUE
    IF(L) 300,300,96
96 DO 97 I=1,L
97 T(I)=ST(I)
    IF(IM4)42,762,42
762 DO 763 I=1,L,4
763 WRITE OUTPUT TAPE 6,764,T(I),T(I+1),T(I+2),T(I+3)
764 FORMAT( 6H T(I)= 4F6.3)
C MERGING
42 J=1
    I=1
98 ALPHA(J)=T(I)-S(IRP)
    IG=I
    ASSIGN 57 TO NX
    K(1)=1
    IF(ALPHA(J))44,45,45
44 ALPHA(J)=ALPHA(J)+TPI
45 BETA(J)=T(I)+Q(IRP)
    IF(BETA(J)-TPI)46,51,51
46 IJ=I+1
    IF(I-L)48,52,52
48 IF(BETA(J)-T(IJ))49,50,50
49 BETA(J)=T(I)+S(IRP)

```

```

    J=J+1
56 I=I+1
    ALPHA(J)=T(I)-S(IRP)
    K(J)=K(J)+1
    GO TO 45
50 K(J)=K(J)+1
    I=I+1
    GO TO 45
51 BETA(J)=BETA(J)-TPI
    IF(BETA(J)-T(IG))249,250,250
250 GO TO NX, (57,58)
249 BETA(J)=T(I)+S(IRP)
    IF(BETA(J)-TPI)54,251,251
251 BETA(J)=BETA(J)-TPI
    GO TO 54
57 ALPHA(1)=ALPHA(J)
    ASSIGN 58 TO NX
58 K(1)=K(J)+K(1)
    K(J)=0
54 IF(I-L)56,53,53
52 BETA(J)=T(I)+S(IRP)
53 IF(K(J))85,60,85
60 ALPHA(J)=0.00
    BETA(J)=0.00
    J=J-1
    IF(J)300,300,85
85 DO 61 I=1,J
    IF(BETA(I)-ALPHA(I))119,120,120
119 ALPHA(I)=ALPHA(I)-TPI
120 W(I)=(ALPHA(I)+BETA(I))/2.
    IF(W(I))121,122,122
121 W(I)=W(I)+TPI
122 SEN(I)= .2*RX(IR)/SCALE+ .4*SINF(W(I))+ .4
    CES(I)= .2*RY(IR)/SCALE+ .4*COSF(W(I))+ .4
    C(I)=W(I)*180./PI
    V(I)=BETA(I)-ALPHA(I)
    IF(ALPHA(I))43,61,61
43 ALPHA(I)=ALPHA(I)+TPI
61 CONTINUE
    IF(J-2)255,257,257
257 IF(W(1)-W(2))255,255,256
256 IE=J-1
    TW=W(1)
    TC=C(1)
    DO 258 I=1,IE
    W(I)=W(I+1)
258 C(I)=C(I+1)
    W(J)=TW

```

```

C(J)=TC
255 MX=12
DO 63 I=1,J,2
  IF (SENSE SWITCH 1) 63,1002
63 PUNCH          64, ID,KLOCK,MX,IR,C(I),C(I),SEN(I),CES(I),
XC(I+1),C(I+1),SEN(I+1),CES(I+1)
64 FORMAT (I5,I4,I3,I2,F6.3,F7.0,F5.3,2F6.3,F7.0,F5.3,F6.3)
1002 MX=14
WRITE OUTPUT TAPE 4,76,(ID,KLOCK,MX,IR,J)
MX=13
NRD=NRD+1
DO 66 I=1,J,4
66 WRITE OUTPUT TAPE 4,67,(ID,KLOCK,MX,IR,K(I),ALPHA(I),BETA(I),
XK(I+1),ALPHA(I+1),BETA(I+1),K(I+2),ALPHA(I+2),BETA(I+2),K(I+3),
YALPHA(I+3),BETA(I+3))
67 FORMAT(I5,I4,I3,I2,I4,2F6.3,I4,2F6.3,I4,2F6.3,I4,2F6.3)
MX=17
DO 102 I=1,J,4
102 WRITE OUTPUT TAPE 4,103,(ID,KLOCK,MX,IR,J,V(I),K(I),V(I+1),K(I+
X,V(I+2),K(I+2),V(I+3),K(I+3))
103 FORMAT(I5,I4,I3,I2,I3,F6.3,I4,F6.3,I4,F6.3,I4,F6.3,I4)
MT=18
DO 106 I=1,J,4
106 WRITE OUTPUT TAPE 4,118, ID,KLOCK,MT,IR,W(I),W(I+1),W(I+2),W(I+3)
118 FORMAT(I5,I4,I3,I2,4F10.3)
138 DO 100 I=1,J
  V(I)=0.
  K(I)=0.
  W(I)=0.
  C(I)=0.
  SEN(I)=0.
  CES(I)=0.
  ALPHA(I)=0.
100 BETA(I)=0.
300 DO 125 I=1,NO
125 T(I)=0.
  IF(LIMIT-KLOCK) 7,68,68
68 IF(IR-NR)80,81,81
80 IR=IR+1
  GO TO 14
81 LAP=LAP-1
  MX=19
  WRITE OUTPUT TAPE 4,76,(ID,KLOCK,MX,IR,NRD)
76 FORMAT (I5,I4,I3,I2,I60)
  IF(LAP)8,8,65
  END(C,1,1,0,1)

```

1.2 SECTION II - TRIANGULATION PROGRAM

1.2.1 Input/Output

There are four input cards prepared for Section II, they are:

- Card 1: Number of data sets being processed.
- Card 2: IM switches.
- Card 3: Radar identifications.
- Card 4: Azimuth limit angle, range of radars.

Card 1: The number of data sets is punched at a fixed point variable in the first five columns of the card.

NODSTS
1-5

Card 2: There are seven IM switches considered "on" if a zero is punched or "off" if a non-zero is punched in their field. The field for each switch in four columns designated as follows.

IM(1)	IM(2)	IM(3)	IM(4)	IM(5)	IM(6)	IM(7)
1-4	5-8	9-12	13-16	17-20	21-24	25-28

The function of each switch is as follows:

- IM(1): Range from primary radar 2, point check.
- IM(2): Output azimuth limits.
- IM(3): List rotated azimuths after sorting.
- IM(4): Intersection feasibility rejection.
- IM(5): Azimuth limit and range limit rejections.
- IM(6): Not used.
- IM(7): Required azimuth point check and no sector overlap rejection.

Refer to output for detail definitions.

Card 3: There is provision for up to five radars in a single run. The first two are designated as the base radars and the remaining three as the secondary radars. The field for

each radar identification is four columns designated as follows:

BI	B2	S1	S2	S3
1-4	5-8	9-12	13-16	17-20

The radars are numbered in order in Section I from 1 through 5. These identification numbers are punched in the fields of Card 2. Any two radars may be designated as the base radars and any or all of the remaining radars may be secondary. The ID is punched as a single digit fixed point variable.

Card 4: The azimuth limit angle, R_1 , is designated as a floating point number in the first seven columns of the card in the form xxx.xxx. The angle is in radians. The range limit, RANGLT, and twice the range limit, TRANGL, are designated in Columns 8-15, and 16-23, respectively.

R	RANGLT	TRANGL
1-7	8-15	16-23

The range limits are designated in miles in the form xxxxx.xx.

There are three types of output from Section II; apparent targets, display data, and intermediate check data.

Apparent targets are outputted on tape unit 3 in the form of x and y coordinates. Along with the coordinates a listing is presented giving the strobcs from each radar which yield an apparent target. The x and y coordinates are given in miles from the origin. The azimuths are given in radians.

Display data are outputted on punched cards under the control of sense switch 1. If the switch is down the coordinates of the apparent target positions are punched, four to a card. These cards are then processed for use in the display equipment.

Intermediate data are outputted on tape unit 6 under the control of the IM "switches." (See input data) This output is used specifically for test purposes and describes the several types of rejections. With this data all the strobcs from each of the radars can be accounted for and hand-checked to assure the program is functioning properly.

1.2.2 Glossary

<u>A</u>	The lower limit of a jammed sector calculated in Section I.
<u>B</u>	The upper limit of a jammed sector calculated in Section I.
<u>CG</u>	Temporary storage locations for the azimuth of an apparent target to the radar before output.
<u>I</u>	General indexing constant.
<u>IALRLK</u>	A sum of the number of azimuth limit rejections per frame. The last two letters indicate the radars whose limits are being tested.
<u>IC</u>	An index used in sorting rotated azimuths.
<u>ICBM</u>	An assigned GO TO constant for selecting Right or Left Plane computation.
<u>ICONL</u>	Number of radar combinations attempting triangulation.
<u>ID</u>	Run Number.
<u>IE</u>	An index used in sorting rotated azimuths.
<u>IFC</u>	Count of the number of combinations of the strobes from the primary radars for which there is no intersection feasible.
<u>IL</u>	An index used to keep track of the secondary radars used in Triangulation attempts.
<u>IM1 to IM7</u>	Intermediate output printed as specified by input card. See input-output for details.
<u>IP</u>	A count of the number of apparent targets found in a given frame.
<u>IR</u>	Radar identification number.

IX Temporary input storage for the number of sectors detected from a given radar.

J Identification number of a radar.

JD A temporary storage location for the radar identification number prior to a rejection output.

JF A temporary storage location for the radar identification number prior to a rejection output.

JJ An index for J radar strobes.

JJS Temporary output storage location for the identification number of the azimuth from radar J involved in a triangulation.

JOHNST An assigned GO TO Constant used as a conditional continue switch on intersection feasibility rejection.

JR Identification number of one of the prime radars.

JS Temporary output storage location for identifying a radar involved in a triangulation.

K Identification Number of a radar.

KK An index identifying a strobe from the Kth radar.

KKS Temporary output storage location for the identification number of the azimuth from radar K involved in a triangulation.

KLM A test constant used to indicate Right or Left Plane calculation.

KLOCK Frame number.

KM A dump storage for unused data taped in Section I.

KN A dump storage for unused data taped in Section I.

KO A dump storage for unused data taped in Section I.

KR Identification number of one of the prime radars.

KRJ A count of the number of range limit rejections from radar J.

KRK A count of the number of range limit rejections from radar K.

KRL A count of the number of range limit rejections from radar L.

KRQ A switching constant used to determine whether both the J and K azimuths have been rotated, sorted and outputted.

KSLBD An assigned GO TO Constant used to indicate the end of a frame.

KV Temporary output storage location for identifying a radar involved in a triangulation.

L Identification number of a radar.

LKLP An identification number indicating the last K azimuth in the left plane.

LKRP An identification number indicating the last K azimuth in the right plane.

LKSRP An index representing the identification number of the last strobe from radar KR in the right plane.

LL An index for L radar strobes.

LLS Temporary output storage location for the identification number of the azimuth from Radar L involved in a triangulation.

LR Identification numbers of secondary radars.

LS Temporary output storage location for identifying a radar involved in a triangulation.

LSK An index which keeps track of the first strobe in the left plane.

LT An assigned GO TO Constant used to indicate that all secondary radars have been tested.

M A temporary index used in calculating azimuth limits.

MA A temporary output constant.

MAR A temporary output index.

MARG A temporary output index.

MAXLIM The maximum number of frames in the run. Inputted on tape 4 from Section I.

MD A temporary output index.

ME A temporary output index.

MND An assigned GO TO constant used to indicate return point after a rejection output.

MNO An assigned GO TO constant used to indicate type of rejection output.

MO A temporary output index.

MOP An assigned GO TO constant used to indicate type of rejection output.

MX A temporary output constant.

MY A temporary output constant.

MZ A dump storage for unused data taped in Section I.

N A temporary index used in calculating azimuth limits.

NA, NB, NC, ND An input constant indicating the number of apparent targets in a strobe sector.

NIL An input constant indicating the identification number of the target positions.

NJ The number of strobcs calculated by a given radar.

NO An input constant indicating the number of targets.

NODSTS Input data designating the number of cases being calculated.

NOR Number of radars in the run.

NR An input constant indicating the number of radars.

NSL A temporary counter which keeps track of the number of strobcs from a secondary radar which have been tested.

NSTR An output index which keeps track of the strobe number being operated with after a range rejection has been encountered.

PI The mathematical constant π .

QA The width of a jammed sector in radians. Calculated in Section I.

QLIM Left limit of radars detecting sector.

R The geometric cutoff angle of the base radars.

RA The azimuth of one radar from another. For the base radars this is also the angle of rotation.

RANGLT An input constant representing the range of a radar.

RF Temporary data; the distance of an intersection from a base radar.

RLIM Right limit of a radar's detecting sector.

RLP Temporary data; the distance of an intersection from a secondary radar.

RQDA Temporary data; the azimuth of an intersection from a secondary radar.

RQLIM Rotated left limit of a radars detecting sector.

RRLIM Rotated right limit of a radars detecting sector.

RSV A temporary storage of RLP after a secondary radar range rejection.

RT Rotated bisectors of the jammed sectors calculated in Section I.

RTJ, RTK Rotated azimuths of the primary radars.

RX The x coordinate of a radar position.

RY The y coordinate of a radar position.

STAFJ The azimuth from the J radar involved in a triangulation.

STAFK The azimuth from the K radar involved in a triangulation.

ST AFL The azimuth from the L radar involved in a triangulation.

T Bisectors of the jammed sectors calculated in Section I.

TE1, Te2,
TE3 Temporary rejection output constants indicating RQDA, and right and left azimuth limits.

TEMP Temporary storage of rotated azimuth during sorting procedure.

TPI The mathematical value 2π .

TRANGL Input data equal to twice the range limit.

X The x coordinate of a true target.

XA The x coordinate of an apparent target.

XP The x coordinate of an intersection point.

XPT X coordinate of the midpoint of the line joining the two base radars.

Y The Y coordinate of a true target.

YA The Y coordinate of an apparent target.

YP Y coordinate of an intersection point.

YPT Y coordinate of the midpoint of the line joining the two radars.

1. 2. 3 FORTRAN Listing

```

DIMENSION RX(8),RY(8),X(100),Y(100),A(5,100),B(5,100),NJ(5),
XLLS(25),NSLC(25),STAFJ(25),STAFK(25),JS(25),STAFI(25),JJS(25),
XKV(25),KKS(25),LS(25),CG(25),RT(5,100),XA(25),YA(25),QA(100),
XRA(5,5),QLIM(5,5),RLIM(5,5),RQLIM(5,5),RRLIM(5,5),NOR(5),T(5,100),
XLR(5)
DISF(C,D,E,F)=SQRTF((C-D)**2+(E-F)**2)
RJPFC(C,D,E,F,P,Q)=(DISF(C,D,E,F)*SINF(Q))/(SINF(P-Q))
RKPC(C,D,E,F,P,Q)=(DISF(C,D,E,F)*SINF(P))/(SINF(P-Q))
PI=3.14159265
TPI=6.28318530
REWIND 3
REWIND 4
READ 1107,NODSTS
1107 FORMAT( 15)
456 IF(NODSTS) 1108, 1108,1109
1108 WRITE OUTPUT TAPE 3,1023
K = 0
A = 0
WRITE OUTPUT TAPE 3,493,K,K,K,K,A,A,A,A,A,A,A,A
REWIND 4
END FILE 3
REWIND 3
PAUSE 209
PAUSE
PAUSE 1
PAUSE
PAUSE 100

```

```

1109 NODSTS=NODSTS-1
      READ          446:IM1,IM2,IM3,IM4,IM5,IM6,IM7
446  FORMAT(7I4)
      READ          446,JR,KR,LR(1),LR(2),LR(3),LR(4),LR(5)
      READ          490,R,RANGLT,TRANGL
490  FORMAT(F7,4,2F8.2)
      READ INPUT TAPE 4,491,ID,NO,NR,MAXLIM
491  FORMAT(4I15)
      DO 492 I=1,NR,4
492  READ INPUT TAPE 4,493,ID,KLOCK,MA,NIL,RX(I),RY(I),RX(I+1),RY(I+1),
      XRX(I+2),RY(I+2),RX(I+3),RY(I+3)
493  FORMAT(I5,I4,I3,I4,8F8.2)
C    CALCULATE RA AND SECTOR LIMITS
      M = KR
      N = JR
      IF(DISF(RX(JR),RX(KR),RY(JR),RY(KR)) - TRANGL) 673,672,67.
673  XPT = .5*(RX(JR) + RX(KR))
      YPT = .5*(RY(JR) + RY(KR))
668  RA(M,N) = ATN1F((RX(M)-RX(N)), (RY(N)-RY(M)))
      IF (RA(M,N)) 190,199,190
190  RA(M,N) = TPI - RA(M,N)
199  RA(N,M) = RA(M,N) + PI
      IF (RA(N,M) - TPI) 194,195,195
195  RA(N,M) = RA(N,M) - TPI
194  QLIM(M,N) = RA(M,N) - R
      IF(QLIM(M,N)) 477,478,478
477  QLIM (M,N) = QLIM(M,N) + TPI
478  RLIM (M,N) = RA(M,N) + R
      IF (RLIM(M,N) - TPI) 479,480,480
480  RLIM(M,N) = RLIM(M,N) - TPI
479  IF (RX(N) - RX(M)) 305,304,11
      11 RA(M,N) = - RA(M,N)
      GO TO 304
305  RA(M,N) = TPI - RA(M,N)
304  QLIM(N,M) = QLIM(M,N) + PI
      IF (QLIM(N,M) - TPI) 177,178,178
178  QLIM(N,M) = QLIM(N,M) - TPI
177  RLIM(N,M) = RLIM(M,N) + PI
      IF(RLIM(N,M) - TPI) 179,180,180
180  RLIM(N,M) = RLIM(N,M) - TPI
179  RQLIM(M,N) = QLIM(M,N) + RA(KR,JR)
      IF (RQLIM(M,N)) 170,172,171
170  RQLIM(M,N) = RQLIM(M,N) + TPI
171  IF (RQLIM(M,N) -TPI) 172,173,173
173  RQLIM(M,N) = RQLIM(M,N) - TPI
172  RRLIM(M,N) = RLIM(M,N) + RA(KR,JR)
      IF (RRLIM(M,N)) 156,158,157
156  RRLIM(M,N) = RRLIM(M,N) + TPI
157  IF (RRLIM(M,N) - TPI) 158,159,159
159  RRLIM(M,N) = RRLIM(M,N) - TPI

```

```

156 IF(M - L) 154,162,154
154 RQLIM(N,M) = RQLIM(M,N) + PI
    IF(RQLIM(N,M) - TPI)160,161,161
161 RQLIM(N,M) = RQLIM(N,M) - TPI
160 RRLIM(N,M) = RRLIM(M,N) + PI
    IF (RRLIM(N,M) - TPI) 162,174,174
174 RRLIM(N,M) = RRLIM(N,M) - TPI
162 IF(IM2) 489,205,489
205 WRITE OUTPUT TAPE 6,206,N,M,RA(N,M),QLIM(N,M),RLIM(N,M),RQLIM(N,M)
    X,RRLIM(N,M)
    WRITE OUTPUT TAPE 6,206,M,N,RA(M,N),QLIM(M,N),RLIM(M,N),RQLIM(M,N)
    X,RRLIM(M,N)
206 FORMAT (2I3,5F8.2)
C CLEARING
672 CONTINUE
489 DO 487 I=1,5
    NJ(I)=0
487 NOR(I)=0
    DO 300 IZ=1,5
    DO 300 I= 1,100
    RT(IZ,I)=0.
300 T(IZ,I)= 0.
    ICONL=0
    KLM = 0
C READ ENVIRONMENT DATA
DO 494 I=1,NO,4
494 READ INPUT TAPE 4,493, ID,KLOCK,MA,NIL,X(I),Y(I),X(I+1),Y(I+1),
    XX(I+2),Y(I+2),X(I+3),Y(I+3)
488 READ INPUT TAPE 4,495, ID,KLOCK,MX,IR,IX
495 FORMAT ( I5,I4,I3,I2,I60)
    IF(MX-14)496,496,497
497 IF(IX-3) 455,505,505
455 IF(MAXLIM-KLOCK)456,456,489
496 DO 498 I=1,IX,4
498 READ INPUTTAPE4,499, ID,KO,MZ,IR,NA,A(IR,I),B(IR,I),NB,A(IR,I+1),
    XB(IR,I+1),NC,A(IR,I+2),B(IR,I+2),ND,A(IR,I+3),B(IR,I+3)
499 FORMAT(I5,I4,I3,I2,I4,2F6.3,I4,2F6.3,I4,2F6.3,I4,2F6.3)
    NOR(IR)=IR
    NJ(IR)= IX
    DO 502 I=1,IX,4
502 READ INPUT TAPE 4,500, ID,KO,MZ,IR,IX,QA(I),NA,QA(I+1),NB,QA(I+2),
    XNC,QA(I+3),ND
500 FORMAT(I5,I4,I3,I2,I3,F6.3,I4,F6.3,I4,F6.3,I4,F6.3,I4)
    DO 503 I=1,IX,4
503 READ INPUT TAPE 4,501, ID,KM,KN,MZ,T(IR,I),T(IR,I+1),T(IR,I+2),
    XT(IR,I+3)
501 FORMAT(I5,I4,I3,I2,4F10.3)
    GO TO 488
C RADAR SELECTION

```

```

505 J = JR
    K = KR
    ASSIGN 678 TO JOHNST
    LKRP = NJ(K)
    LKLP = NJ(K)
    JJ = 1
    KK = 1
    IF (NOR(J)) 400,489,400
400 IF (NOR(K)) 401,489,401
401 IL=1
404 L = LR(IL)
    IF (L) 402,403,402
402 IF (NOR(L)) 674,403,674
403 IL = IL + 1
    IF (IL - NR + 2) 404,404,489
674 KRJ=0
    KRK=0
    KRL=0
    IALRJK=0
    IALRJL=0
    IALRKJ=0
    IALRKL=0
    IALRLJ=0
    IALRLK=0
    IFC=0
    ICONL=ICONL+1
    KNIT=0
    LL= 1
    IP=0
C   ROTATING ANGLES FOR J + K
31 KRQ=-1
    MD = J
    ME = NJ(J)
32 DO 33 IT = 1,ME
    RT(MD,IT) = T(MD,IT) + RA(K,J)
    IF(RT(MD,IT)-TPI) 34,35,35
35 RT(MD,IT)= RT(MD,IT)-TPI
34 IF(RT(MD,IT)) 36,33,33
36 RT(MD,IT) = RT(MD,IT) + TPI
33 CONTINUE
    IF(KRQ)37,38,38
37 KRQ=1
    MD= K
    ME = NJ(K)
    GO TO 32
C   SORTING RT(J,JJ) + RT(K,KK)
38 KRQ=-1
    MD= J
39 IC=1
    IF(NJ(MD)-1)45,45,40
40 IE=IC+1

```

```

41 IF(RT(MD,IC)-RT(MD,IE)) 42,42,43
42 IE=IE+1
   IF(IE-NJ(MD)) 41,41,44
43 TEMP= RT(MD,IE)
   RT(MD,IE)= RT(MD,IC)
   RT(MD,IC)= TEMP
   GO TO 42
44 IC=IC+1
   IF(IC-NJ(MD)) 40,45,45
45 IF(KRQ)46,47,47
46 KRQ=1
   MD=K
   GO TO 39
C   INTERMEDIATE FOR SORTED ROTATED AZIMUTHS
47 IF(IM3) 52,553,52
553 KRQ=-1
   ME=NJ(J)
   MD =J
   WRITE OUTPUT TAPE 6,801,KLOCK
801 FORMAT (17H           KLOCK = 15)
51 DO 48 MO=1,ME,4
48 WRITE OUTPUT TAPE 6,49,MD,ME,RT(MD,MO),RT(MD,MO+1),RT(MD,MO+2),
   XRT(MD,MO+3)
49 FORMAT(15H J, JJ,RT(J, JJ)= 214,4F8.2)
   IF(KRQ)50,52,52
50 KRQ=1
   MD=K
   ME= NJ(K)
   GO TO 51
C   CHECK RIGHT PLANE J AND K AZIMUTHS AGAINST ROTATED JK AND KJ
C   AZIMUTH LIMITS
52 ASSIGN 405 TO LT
   IF (RT(J, JJ) - RQLIM(J, K)) 53,54,54
53   KLM = -1
56 IF (RT(K, KK) - RRLIM(K, J)) 57,58,58
57 RF = RRLIM (K, J)
   IALRKJ=IALRKJ+1
   ASSIGN 59 TO MNP
60 ASSIGN 56 TO ICBM
   JD = K
   NSTR= KK
   ASSIGN 135 TO MNO
   ASSIGN 131 TO MOP
   GO TO 82
58 IF(RT(K, KK) - RQLIM(J, K)) 64,65,65
65 LKRP= KK-1
   IALRJK=IALRJK+1
   LKSR= KK-1
   RF = RQLIM (J, K)
   ASSIGN 321 TO MNP

```

```

GO TO 60
C INCREMENT STROBE NUMBERS - RIGHT PLANE
59 KK=KK+1
IF(KK-LKRP) 320,320,321
320 GO TO ICBM,(56,58)
321 JJ=JJ+1
ASSIGN 678 TO JOHNST
IF(JJ -NJ(J)) 322,322,722
322 KK=1
GO TO 52
C CHECK LEFT PLANE J AND K AZIMUTHS AGAINST ROTATED JK AND KJ
C AZIMUTH LIMITS
54 IF(KLM) 370,371,371
370 KK=LKRP+1
GO TO 372
371 KK=1
372 KLM=1
ASSIGN 678 TO JOHNST
LSK=KK
329 IF (RT(J,JJ) - RRLIM(J,K)) 349,349,328
328 IF (RT(K,KK) - RRLIM(J,K)) 348,348,333
349 ASSIGN 350 TO MNP
IALRJK=IALRJK+1
JD = J
NSTR = JJ
RF = RRLIM(J,K)
GO TO 347
333 IF (RT(K,KK) - RQLIM(K,J)) 382,335,335
335 ASSIGN 350 TO MNP
IALRJK=IALRJK+1
RF = RQLIM(J,K)
GO TO 346
348 ASSIGN 330 TO MNP
IALRKJ=IALRKJ+1
RF = RRLIM(J,K)
346 JD = K
NSTR = KK
347 ASSIGN 135 TO MNO
ASSIGN 131 TO MOP
GO TO 82
C INCREMENT STROBE NUMBERS - LEFT PLANE
350 JJ=JJ+1
IF(JJ -NJ(J)) 380,380,722
380 KK=LSK
GO TO 329
330 KK=KK+1
IF(KK-LKLP) 328,328,350
722 ASSIGN 489 TO KSLBD
GO TO 63
C CHECK INTERSECTION FEASIBILITY

```

```

C   LEFT PLANE
382 RTJ=RT(K,KK)
    RTK=RT(J,JJ)
    GO TO 325
C   RIGHT PLANE
64  RTJ= RT(J,JJ)
    RTK= RT(K,KK)
325 IF(RTJ-RTK)1001,1001,1000
1001 IFC = IFC + 1
    IF(IM4)326,950,326
950 WRITE OUTPUT TAPE 6,1013,J,JJ,(RT(J,JJ)),K,KK,(RT(K,KK))
1013 FORMAT (30H NO INTERSECTION FEASIBILITY 214,F5.2,214,F5.2)
    IF (KLM) 326,326,330
326 ASSIGN 58 TO ICBM
    GO TO 321
C   CHECK RANGE FROM J
1000 RF=RJPF(RX(J),RX(K),RY(J),RY(K),RT(J,JJ),RT(K,KK))
    IF(RF-RANGLT)76,76,77
77  JD=J
    KRJ=KRJ+1
    NSTR =JJ
331 IF(KLM) 78,78,79
78  ASSIGN 321 TO MNP
681 ASSIGN 131 TO MOP
332 ASSIGN 130 TO MNO
    GO TO 82
79  ASSIGN 130 TO MNO
    ASSIGN 131 TO MOP
334 ASSIGN 330 TO MNP
C   INTERMEDIATE OUTPUT FOR AZIMUTH LIMIT AND RANGE REJECTIONS
82  IF(IM5) 128,559,128
559 GO TO MNO,(130,135)
130 WRITE OUTPUT TAPE 6,1003,JD
1003 FORMAT (31H RANGE LIMIT REJECTION ON RADAR I2)
    GO TO 129
135 WRITE OUTPUT TAPE 6, 1004, JD,NSTR
1004 FORMAT(45H SECTOR LIMIT REJECTION ON RADAR AND STROBE = 2I4)
129 GO TO MOP,(131,132,136,137)
131 WRITE OUTPUT TAPE 6,1005,JD,NSTR,RT(JD,NSTR),RF
1005 FORMAT(33H RADAR,STROBE,ROTATED AZIMUTH,RF= 2I4,F6.2,F16.2)
    GO TO 128
132 WRITE OUTPUT TAPE 6,1006,JD,RLP
1006 FORMAT(18H RADAR L,RF OF L = I4.F8.2)
    GO TO 128
136 WRITE OUTPUT TAPE 6,1007,JD,NSTR,JF,TE1,TE2,TE3
1007 FORMAT(48H R1,STB,R2,RT(R1,STB),RQLIM(R1,R2),RRLIM(R1,R2)=3I4,
X3F8.2)
    GO TO 128
137 WRITE OUTPUT TAPE 6,1008,JD,NSTR,TE1,TE2,TE3
1008 FORMAT(36H R1,R2,RQDA,QLIM(R1,R2),RLIM(R1,R2)=2I4,3F8.2)

```

```

128 GO TO LT,(405,407)
407 IL = IL + 1
    IF (IL - NR + 2) 406,406,405
405 ASSIGN 405 TO LT
    IL=1
    GO TO MNP,(59,321,330,350)
C   CHECK RANGE FROM K
    76 RF=RKPF(RX(J),RX(K),RY(J),RY(K),RT(J,JJ),RT(K,KK))
    IF(IM1) 299,298,299
298 WRITE OUTPUT TAPE 6,297,J,JJ,(RT(J,JJ)),K,KK,(RT(K,KK)),RF
297 FORMAT(33H J,JJ,RT(J,JJ),K,KK,RT(K,KK),RF=2I3,F6.3,2I3,F6.3,F9.2)
299 IF(RF-RANGLT)83,83,84
    84 JD=K
    KRK=KRK+1
    NSTR=KK
    GO TO JOHNST,(678,331)
    678 IF(DISF(RX(J),RX(K),RY(J),RY(K))-RANGLT) 331,331,677
    677 IF(KLM) 680,680,695
    680 ASSIGN 59 TO MNP
    ASSIGN 58 TO ICBM
    GO TO 681
    695 ASSIGN 350 TO MNP
    ASSIGN 131 TO MOP
    ASSIGN 130 TO MNO
    GO TO 82
C   CALCULATE COORDINATES OF INTERSECTION POINTS
    83 XP = RX(K) +RF* SIN(RT(K,KK) - RA(K,J))
    YP = RY(K) +RF* COS(RT(K,KK) - RA(K,J))
    IF(KLM) 688,688,692
    688 ASSIGN 331 TO JOHNST
    GO TO 693
    692 ASSIGN 678 TO JOHNST
    693 KNIT = KNIT+1
    406 ASSIGN 407 TO LT
C   CHECK RANGE FROM L
    L = LR(IL)
    IF (L) 408,352,408
    408 RLP=DISF(XP,RX(L),YP,RY(L))
    IF(RLP- RANGLT) 88,88,87
    87 JD=L
    KRL=KRL+1
    RSV=RLP
    IF(KLM) 337,337,338
    337 ASSIGN 132 TO MOP
    ASSIGN 130 TO MNO
    ASSIGN 321 TO MNP
    GO TO 82
    338 ASSIGN 130 TO MNO
    ASSIGN 132 TO MOP
    GO TO 334
C   CALCULATE RQDA

```

```

88 RQDA=ATN1F((RX(L)-XP),(YP-RY(L)))
   NSL=1
   IF(RQDA) 192,193,192
192 RQDA= TPI - RQDA
193 IF(IM7)100,840,100
840 WRITE OUTPUT TAPE 6,1009,XP,YP,RQDA
1009 FORMAT(12H XP,YP,RQDA= 3F8.2)
C   CHECK SECTOR OVERLAP
100 IF(A(L,LL)-B(L,LL))101,101,102
101 IF (B(L,LL) - RQDA)103,118,104
103 LL=LL+1
   NSL=NSL+1
   IF(LL-NJ(L))100,100,1060
1060 NSL=NSL-1
106 IF(IM7)352,1111,352
1111 WRITE OUTPUT TAPE 6,1010,L,LL,A(L,LL),B(L,LL),RQDA
1010 FORMAT(22H L,LL,ALPHA,BETA,RQDA= 2I4,3F8.2)
352 LL=1
   IL = IL + 1
   IF (IL - NR + 2) 406,406,61
61 IF(KLM) 354,354,330
354 ASSIGN 58 TO ICBM
   IL=1
   LL=1
   GO TO 59
104 IF(A(L,LL) - RQDA) 118,118,106
102 IF (A(L,LL) - RQDA) 118,118,107
107 IF (B(L,LL) - RQDA) 103,118,118
C   DATA STORAGE FOR SUCCESSFUL TRIANGULATION
118 IP=IP+1
   XA(IP)=XP
   YA(IP)=YP
   NSLC(IP)= NSL
   JS(IP)= J
   JJS(IP)=JJ
   KV(IP)= K
   KKS(IP)=KK
   LS(IP)= L
   LLS(IP)=LL
   STAFJ(IP)= T(J,JJ)
   STAFK(IP)=T(K,KK)
   STAFL(IP)=T(L,LL)
   CG(IP)= RQDA
   ASSIGN 405 TO LT
   IF(IP- 25) 61,720,720
720 ASSIGN 725 TO KSLBD
C   DATA OUTPUT
63 WRITE OUTPUT TAPE 3,1027
1027 FORMAT(79H1 RUN CLK TT NAT RFJ RFK RFL NIF ALRJK KJ JL
6 KL LJ LK ATP CON)
   MX=22

```

```

MY=23
WRITE OUTPUT TAPE 3,1012,ID,KLOCK,MX,IP,KRJ,KRK,KKL,IFC,IALRJK,IAL
1RKJ,IALRJL,IALRKL,IALRLJ,IALRLK,KNIT,ICONL
1012 FORMAT (I5,I4,I3,5I5,I7,7I5)
WRITE OUTPUT TAPE 3,1023
1023 FORMAT(77H RUN CLK T TRG      XA      YA      XA      YA      XA
X  YA      XA      YA)
MARG=- 3
MX=21
MA=7
DO 73 MO = 1,IP,4
MARG= MARG+4
WRITE OUTPUT TAPE 3, 493,ID,KLOCK,MX,MARG,XA(MO),YA(MO),XA(MO+1),
XYA(MO+1),XA(MO+2),YA(MO+2),XA(MO+3),YA(MO+3)
IF(SENSE SWITCH 1) 220,73
220 IF (IP) 1021,73,1021
1021 PUNCH          1020,ID,KLOCK,MX,MA,MARG,XA(MO),YA(MO),XA(MO+1)
7,YA(MO+1),XA(MO+2),YA(MO+2),XA(MO+3),YA(MO+3)
1020 FORMAT (I5,I4,I3,2I2,8F6.1)
73 CONTINUE
1022 FORMAT(I5,I4,I3,2I2,8F8.2)
WRITE OUTPUT TAPE 3,166
166 FORMAT(75H RUN CLK TT TGT      J      JJ TA(J)      K      KK TA(K)      NSL
X L      LL TA(L) RQDA)
MAR=0
DO 1100 MZ=1,IP
MAR=MAR+1
1100 WRITE OUTPUT TAPE 3,1026,ID,KLOCK,MY,MAR,JS(MZ),JJS(MZ),(STAFJ(MZ)
X),KV(MZ),KKS(MZ),(STAFK(MZ)),NSLC(MZ),LS(MZ),LLS(MZ),(STAF(LMZ)),
XCG(MZ)
1026 FORMAT(I5,I4,I3,I4,2I5,F6.3,2I5,F6.3,3I5,2F6.3)
DO 1025 MO=1,IP
JS(MO)=0
JJS(MO)=0
STAFJ(MO)=0.
KV(MO)=0
KKS(MO)=0
NSLC(MO)=0
STAFK(MO)=0.
LS(MO)=0
LLS(MO)=0
STAF(LM)=0.
CG(MO)=0.
XA(MO)=0.
1025 YA(MO)=0.
IF(MAXLIM-KLOCK)456,456,216
216 GO TO KSLBD, (725,489)
725 IP=0
GO TO 61
END(0,1,1,0,1)

```

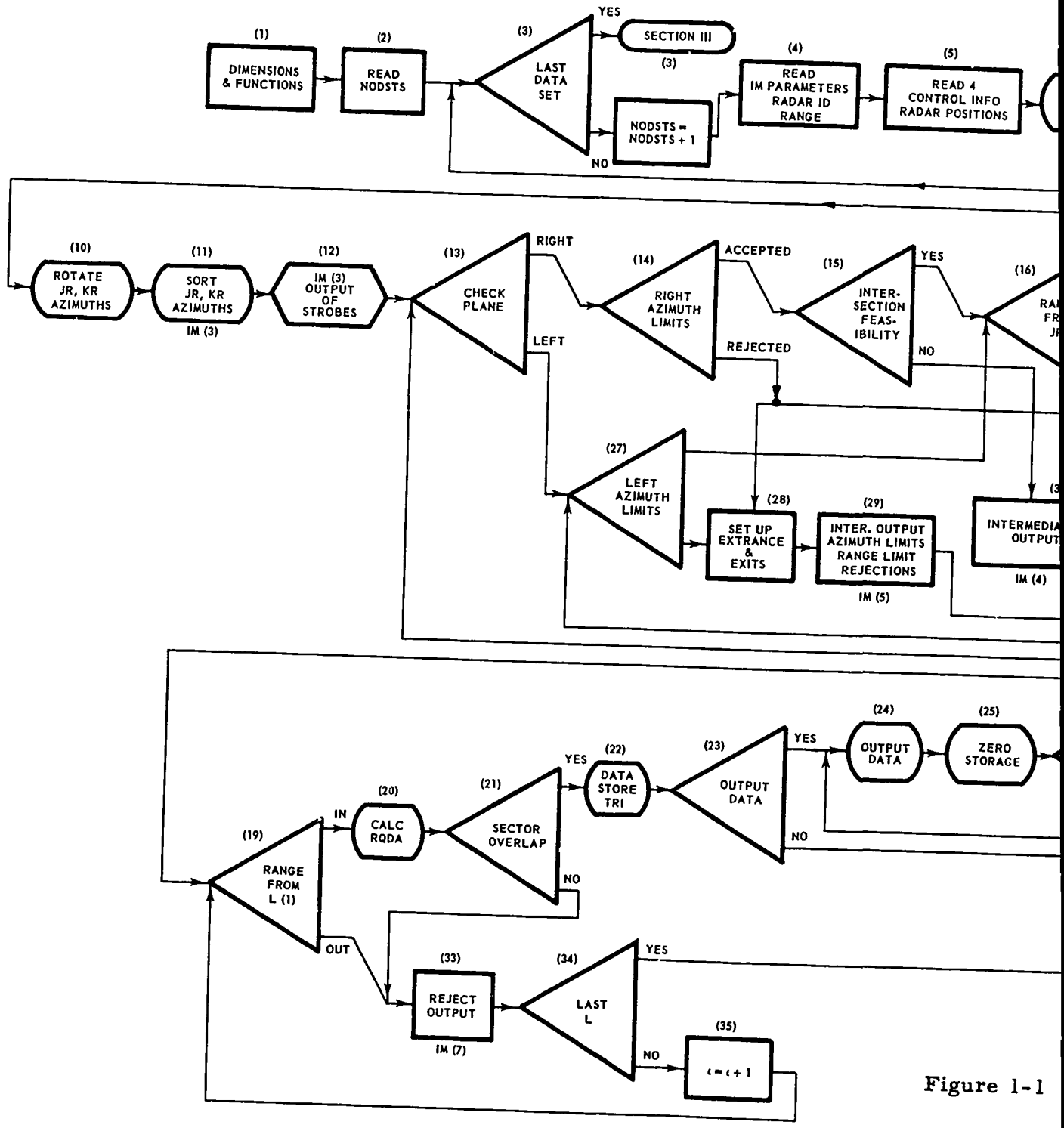


Figure 1-1

A

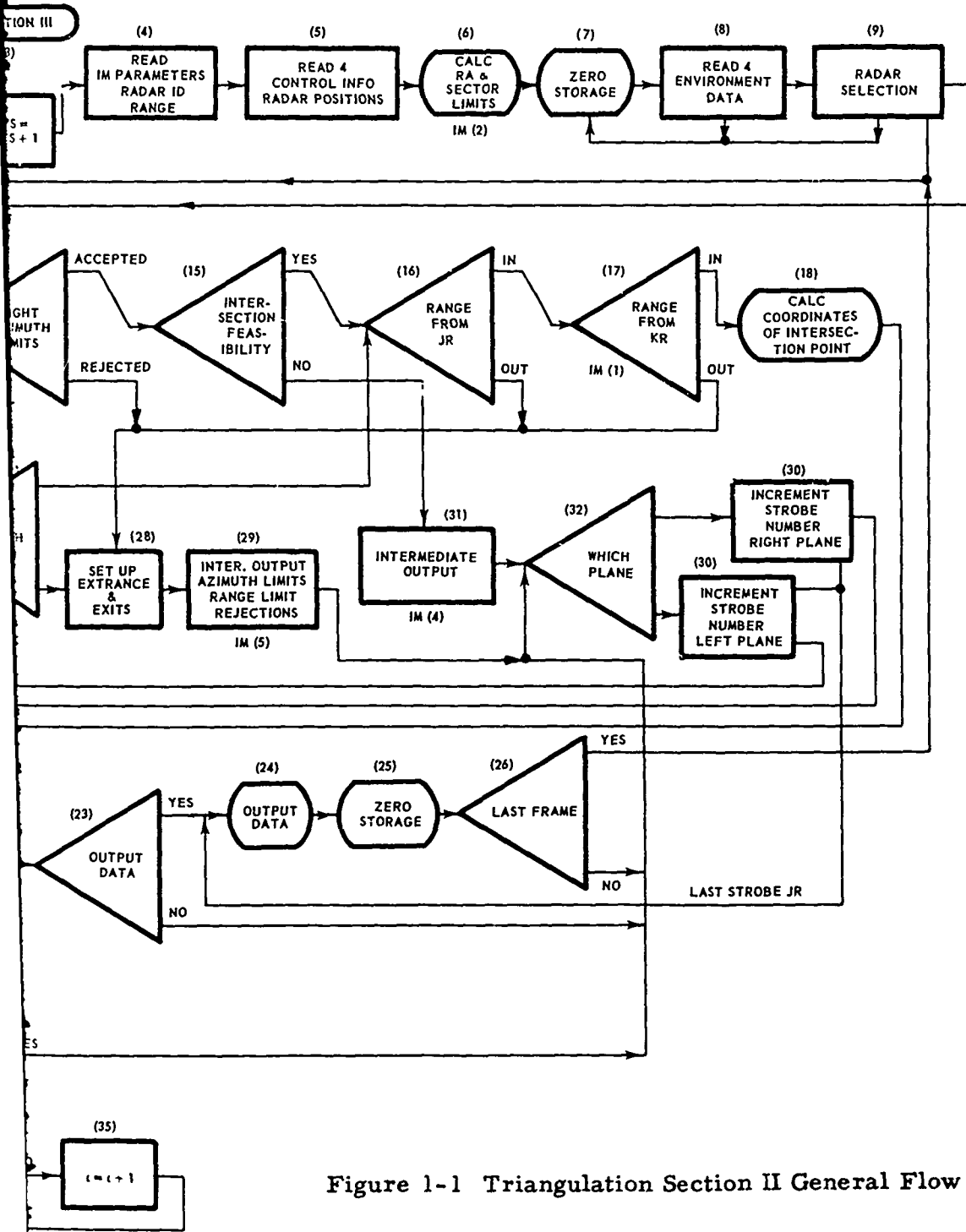


Figure 1-1 Triangulation Section II General Flow Diagram

B

1.3 SECTION III - DATA REDUCTION

1.3.1 Input/Output

There are eleven input cards prepared for Section II as follows:

Card 1.	Number of data sets being processed
Card 2	IM switches
Cards 3 & 4	Histogram switches
Cards 5 & 6	Maximum value of intervals
Cards 7 & 8	Interval size
Card 9	Target and sector cutoffs
Card 10	Number of frames to be processed
Card 11	Radar ID and range.

Card 1 The number of data sets is punched as a fixed point variable in the first five columns of the card.

NODSTS

1-5

Card 2 There are four IM switches considered "on" if a zero is punched; or "off", if a non-zero is punched in their field. The field for each switch is five columns designated as follows:

IM(1)	IM(2)	IM(3)	IM(4)
1-5	6-10	11-15	16-20

The function of each switch is as follows:

IM(1) Calculate means and standard deviation of histograms for each frame and output information on tape 6.

IM(2) List the true targets with the nearest apparent target and vice versa for each frame on tape 6.

IM(3) List misses, hits and false alarms for each frame on tape 6.

IM(4) Not used.

Cards 3 & 4 There are fifteen possible histograms which may be calculated depending upon the setting of the histogram "switches". Like the IM switches they are considered "on" if a zero is punched, or "off" if a non-zero is punched in their field. The field for each switch is seven columns distributed between cards 3 and 4 as follows:

Card 3	IH(1)	IH(2)	IH(3)	IH(4)	IH(5)	IH(6)	IH(7)
	1-7	8-14	15-21	22-28	29-35	36-42	43-49
		IH(8)	IH(9)				
		50-56	57-63				

Card 4	IH(10)	IH(11)	IH(12)	IH(13)	IH(14)	IH(15)
	1-7	8-14	15-21	22-28	29-35	26-42

The histograms are all frequency distribution tables. The descriptions of the fifteen histograms are as follows:

Histogram ID Description

1. Misses: frequency; number of true targets; distribution; distance between true target and nearest apparent.
2. Hits: frequency; number of true targets; distribution; distance between true target and nearest apparent target.
3. False alarms: frequency; number of apparent targets; distribution; distance between apparent target and nearest true target.
4. Misses and Hits: Accumulation of data from histograms 1 and 2.
5. Hits and false alarms: Accumulation of data from histograms 2 and 3.
6. Strobe width: frequency; number of strobes: distribution; width of strobes in degrees. A separate histogram is calculated for each radar.

7. Strobe width: An accumulation of the data from all the radars in histograms 6.
8. Number of targets per sector: frequency; number of sectors with X targets: distribution: number of targets. A separate histogram is calculated for each radar.
9. Number of targets per sector: An accumulation of the data from all radars in histograms 8.
10. Number of sectors with K targets or more: This histogram is the same as number 8, except that sectors with less than K targets (input data KTARG) are omitted.
11. Number of sector with K targets or more: This histogram is the same as number 9, except that sector with less than K targets are omitted.
12. Number of targets in sectors of width L degrees or greater: This histogram is the same as number 8, except that sectors of less than L degrees (input data SWLD) are omitted.
13. Number of targets in sectors of width L degrees or greater: This histogram is the same as number 9, except that sectors of less than L degrees are omitted.
14. Number of strobe sectors: This is not a histogram, only a list of the number of sectors detected for each radar.
15. Number of strobe sectors: A list of the number of stobes detected by all radars.

Cards 5 and 6: Associated with each histogram designated in cards 3 and 4 is a maximum distribution interval. These values are punched in floating point notation of the form XXX.XX and are distributed on cards 5 and 6 as follows:

Card 5: AMXM(1) AMXM(2) AMXM(3) AMXM(4)
 1-7 8-14 15-21 22-28

AMXM(5) AMXM(6) AMXM(7) AMXM(8) AMXM(9)
29-35 36-42 43-49 50-56 57-63

Card 6: AMXM(10) AMXM(11) AMXM(12) AMXM(13)
1-7 8-14 15-21 22-28

AMXM(14) AMXM(15)
29-35 36-42

The maximum distribution interval designates the range of the distribution. If a value happens to be greater than the maximum, the datum will be placed in the last interval.

Cards 7 & 8 Each histogram also has an interval size associated with it. These values are punched in floating point notation in the form XXXXX.X and are distributed on cards 7 and 8 as follows:

Card 7: QNTA(1) QNTA(2) QNTA(3) QNTA(4) QNTA(5)
1-7 8-14 15-21 22-28 29-35

QNTA(6) QNTA(7) QNTA(8) QNTA(9)
36-42 43-49 50-56 57-63

Card 8: QNTA(10) QNTA(11) QNTA(12) QNTA(13)
1-7 8-14 15-21 22-28

QNTA(14) QNTA(15)
29-35 36-42

Card 9 This card contains cut-off data used in histograms 10, 11, 12, and 13. KTARG is the minimum number of targets of interest and SWLD is the minimum sector width of interest. KTARG is a fixed point number and SWLD is a floating point number in the form XX.XXX. Their fields are designated.

Allowing the interval size as an input parameter makes it possible for the analyst to have any spread he desires. However, there can be no more than twenty (20) intervals in each histogram. The computer determines the number of intervals by dividing AMXM by QNTA. It is the analyst's responsibility to assure that the AMXM and QNTA are chosen to give no more than twenty intervals.

On the card as follows:

Card 9: KTARG SWLD
 1-5 6-11

Card 10: Card 10 contains the number of frames in the run. It is specified as a fixed point number in the first 5 columns.

Card 10: MAXLIM
 1-5

Card 11: Card 11 contains the radar ID's and the range. The ID's are in fixed point notations and the range is in floating point notation in the form XXXXX.X. Their fields are as follows:

Card 11: JR KR LRA LRB LRC RANGE
 1-5 6-10 11-15 16-20 21-25 26-32

The range is designated in miles.

There are three types of data outputted from Section III, on-line data (rates, and histograms of sums), tape 2 data (X and Y coordinates of hits), and tape 6 data (intermediate data).

On-line data are outputted from the computer directly on the tabulator. There are two types of on-line output. (1) Frame by frame and totals outputted after all the frames have been completed. (2) At the end of the run are the histograms of the totals for the misses and hits and any other histogram specified by the input data. The first two histograms misses and hits, are standardized and outputted below the histograms with an S in the margin. Means and standard deviations are also calculated for these two histograms and are outputted to the right of the histograms.

Tape 2 data are outputted for future use by auxiliary routines.

Tape 6 data are outputted under the control of the IM switches (see input data). This data is used mostly for checking purposes, however, this data may also be used for auxiliary investigations by writing an appropriate input routine.

1.3.2 Glossary

<u>AMXM</u>	Maximum value of histogram intervals. Any data values higher than AMXM are collected in the last interval.
<u>AQT</u>	Temporary floating point storage of interval headings of histograms. LQT is the fixed point storage ID. These two are equivalent.
<u>AVE</u>	Average of data in histograms. Only outputted on IM (1).
<u>BMA</u>	Difference between the upper and lower limits of a strobe sector read in from tape 4 (calculated in Section I).
<u>DATA</u>	Identification index of items being histogrammed.
<u>DEH</u> <u>DEM</u>	Temporary storage used when calculating the mean and standard deviation for histograms 1 and 2.
<u>DEV</u>	Standard deviation for histograms. Only outputted on IM(1).
<u>DH</u> <u>DH1</u> <u>DH2</u>	Temporary storage used when calculating the mean and standard deviation for histograms 1 and 2.
<u>DISQ</u>	Distance between an apparent target and its nearest true target.
<u>DSJQ</u>	Distance between a true target and its nearest apparent target.
<u>DM, DM1,</u> <u>DM2</u>	Temporary storage used when calculating the mean and standard deviation for histograms 1 and 2.
<u>DRT</u>	Ratio of number of apparent target to numbers of true targets in range of 3 radars.
<u>DSCL1</u> <u>DSCL2</u> <u>DSCL3</u>	Distance between true and apparent targets in classes 1, 2 and 3. (Misses, hits and false alarms.)

DTRAT Ratio of number of apparent target to number of true targets.

FAC1,
FAC2 Temporary storage used when standardizing histograms 1 and 2.

I A general index used in Do loops.

ICLS1,
ICLS2,
ICLS3 Temporary indices used to keep track of true targets in classes 1, 2 and 3. (Misses, hits and false alarms.)

ID Run number.

IH A dimensioned index identifying which histograms are being calculated.

IHT Calculated histogram interval into which a given datum falls.

IM Intermediate output selectors. IM(1), IM(2), IM(3), IM(4).

IP Number of apparent target in a frame.

IR Temporary storage of radar ID while reading in strobe information.

IRT An index which keeps track of the number of radars which detected in a given frame.

ISAVE A dimensioned index storing the number of strobes for a radar.

ISP Number of apparent targets read in from a given block of data.

IX Temporary index used on input for number of strobes from a given radar.

J General index used in Do loops.

JCLS1, Temporary indices used to keep track of apparent targets
JCLS2, in classes 1, 2 and 3. (Misses, hits and false alarms.)
JCLS3

J4, J5, JE Temporary index used when setting up histogram data.

JR ID of base radar.

KING Assigned GO TO Constant used to select proper input state-
 ment when reading in apparent targets from tape 3.

KLOCK Frame number.

KNT A count of lines of output, used for page restoring.

KO A dump storage for eliminating unwanted data on tapes 3 and 4.

KORAN An assigned GO TO constant used to determine whether the
 histograms for all the radars are done.

KOUNT A count of the number of targets in range for a frame.

KQ A temporary index used in setting up data for the histograms.

KR ID of one of the base radars.

KRO An index used to keep track of which histogram is being
 calculated.

KRU An assigned GO TO constant used to select histograms
 "strobe width" and "number of targets per sector."

KSUM Storage for accumulating items such as number of elements
 in the classes, miss rate, hit rate, etc. KSUM is the fixed
 point items and RSUM is used for floating point items. The
 two are equivalent.

KTARG Input constant designating number of target per sector.
 Used as a lower cut-off. i. e., KTARG = 4 means any
 sector with less than 4 targets is disregarded in the
 histogram.

KTLOCK Intermediate storage of frame number used for comparison in reading in data from tapes 3 and 4.

KXC
KXD
KXE Assigned GO TO constants used in selection nearest apparent target to a true target and vice versa.

KXG
KXL
KXP Assigned GO TO constants used in setting up data for histograms.

KI Temporary index used when setting up histogram data.

L A count of the number of items in class 2 and class 3.

LBJ A temporary constant used to output histograms of all radar data combined.

LH Storage of number of items in the intervals of a histogram.

LQT See AQT.

LRA
LRB
LRC ID of secondary radars.

LSUM Accumulation of items in LH for histograms of all radars.

M A count of the number of items in class 1.

MA, MB Temporary indices used for output.

MAR A dump storage for eliminating data not wanted from tape 4.

MAX Number of intervals in a histogram. Can be from 1 to 20.

MAXLIM Total number of frames in the run. Input data.

ME A temporary index used in sorting targets.

MOP A temporary index used in reading input data from tapes 3 and 4.

MX A temporary constant used to designate which class is being outputted in IM(3).

M1SUM
M2SUM The histograms of classes 1 and 2 are standardized for comparison. M1SUM and M2SUM are the standardizations for LSUM(1) and LSUM(2).

N A variable indicating which histogram is being calculated.

NA Number of targets in a strobe sector calculated in Section I. Inputted from tape 4.

NATI A table identifying the true target closest to a given apparent target and vice versa, for tabling.

NECL1
NECL2
NECL3 Number of elements in class 1, class 2 and class 3.

NFARAT Ratio of number of false alarms to number of apparent targets in a frame.

NG A switching constant used to indicate that all the true or apparent targets have been sorted.

NHTRAT Ratio of the number of hits to the number of apparent targets.

NIL A dump storage for eliminating data not wanted from tape 4.

NMSRAT Ratio of number of misses to the number of real targets.

NO Number of true targets.

NODSTS Number of runs being processed in a given pass.

NR Number of radars.

NTJ A table identifying the apparent target closest to a given true target and vice versa for outputting.

NU A temporary index used in setting up data for histograms.

QNTA The size of an interval in a histogram.

RANGE Maximum distance between a radar and a target for detection.

RSUM See KSUM.

RX X coordinate of a radar.

RY Y coordinate of a radar.

SD Temporary storage used in finding closest true target to an apparent target and vice versa.

SIG1
SIG2 Standard deviation for histograms 1 and 2.

SM1 Floating notation for sum of elements in class 1 for a frame.

SUM Sum of items in a histogram. Used to calculate mean.

SWLD Strobe width in degrees. An input parameter used as a minimum strobe width when calculating some histograms.

T A strobe from a given radar.

TO1, TO2 Floating notation for number of elements in class 1 and class 2 for a run.

T1, T2 Temporary storage used in sorting true and apparent targets.

X X coordinate of a true target.

XA X coordinate of an apparent target.

XCLK Frame number in floating notation.

XH Temporary storage in calculating mean and standard deviation for histogram of hits.

<u>XID</u>	Run number in floating notation.
<u>XM</u>	Temporary storage in calculating mean and standard deviation for histogram of misses.
<u>Y</u>	Y coordinate of a true target.
<u>YA</u>	Y coordinate of an apparent target.

1.3.3 FORTRAN Listing

```

DIMENSION IH(15), AMXM(15), QNTA(15), NSSPD(5), RX(8), RY(8), X( 50),
XY( 50), BMA(5,100), NA(5,100), ISAVE(5), T(5,100), XA(200), YA(200),
XDISQ(50), NTJ(50), DJSQ(50), NATI(50), ICLS2(50), JCLS2(50), DCLS2(50),
XICLS1(50), JCLS1(50), DCLS1(50), ICLS3(50), JCLS3(50), DCLS3(50)
DIMENSION DATA(125), LH( 25), AQT( 25), LQT( 25), LSUM(15,20), KSUM(15)
X, RSUM(15)
DIMENSION M1SUM(20), M2SUM(20)
EQUIVALENCE (KSUM, RSUM)
EQUIVALENCE (LQT, AQT)
READ 1000, NODSTS
1000 FORMAT(I5)
REWIND 2
REWIND 3
REWIND 4
422 IF(NODSTS) 90,90,91
90 REWIND 3
92 REWIND 4
END FILE 2
REWIND 2
END FILE 5
REWIND 5
END FILE 6
REWIND 6
PAUSE 1
91 NODSTS = NODSTS-1
TOI = 0
DO 140 I = 1,15
KSUM(I) = 0
DO 140 II = 1,20
140 LSUM (I,II) = 0
READ 978, IM1, IM2, IM3, IM4
978 FORMAT(4I5)
C SPECIFY HISTOGRAM PARAMETERS
READ 991 , (IH(I), I = 1,15)
991 FORMAT (9I7)

```

```

      READ 998, (AMXM(I),I=1,15)
998  FORMAT (9F7.2)
      READ 992,(QNTA(I),I=1,15)
992  FORMAT(9F7.1)
      READ 994,KTARG,SWLD
994  FORMAT(I5,I6.3)
      READ INPUT TAPE 4,1002,ID,NO,NR,MAXLIM
1002 FORMAT (4I15)
      READ 1000,MAXLIM
      READ 1001 , JR, KR, LRA, LRB, LRC, RANGE
1001 FORMAT (5I5,F7.1)
500  ASSIGN 11 TO KING
      KNT = 0
      IRT=0
      DO 1 I= 1,NR,4
    1  READ INPUT TAPE 4,1006,ID,KLOCK,MX,NIL,RX(I),RY(I),RX(I+1),RY(I+1)
      X,RX(I+2),RY(I+2),RX(I+3),RY(I+3)
1006 FORMAT(I5,I4,I3,I4,8F8.2)
    13 IF(KNT) 70,71,70
    70 KNT = KNT -1
      GO TO 72
    71 KNT = 49
      PRINT 981,ID,NO
981  FORMAT(16HICASE NUMBER      I5,10H.   NT      I5,57H      TRIANGULATION
      X SECTION 3. BENDIX SYSTEMS DIVISION 704)
      PRINT 982
982  FORMAT(76H CLOCK      K      NMS      NHT      NFS NFARAT NHTRAT NMSRAT
      XNDTRAT      NDR      KOUNT)
    72 MOP = 1
1008 FORMAT(80H
      X
      DO 2 I=1,NO,4
    2  READ INPUT TAPE 4,1006,ID,KLOCK, ...MAR,X(I),Y(I),X(I+1),Y(I+1),
      XX(I+2),Y(I+2),X(I+3),Y(I+3)
980  DO 985 I=1,NR
985  ISAVE(I)=C
    9  READ INPUT TAPE 4,1039,KO,MX,IR,IX
1039 FORMAT (I9,I3,I2,I60)
      IF(MX-18) 6,6,4
    6  ISAVE(IR)= IX
      DO 10 I = 1,IX,4
    10 READ INPUT TAPE 4,1008
      DO 7 I=1,IX,4
    7  READ INPUT TAPE 4,1010,ID,KLOCK,MX,KO,BMA(IR,I),NA(IR,I),BMA(IR,I+
      X1),NA(IR,I+1),BMA(IR,I+2),NA(IR,I+2),BMA(IR,I+3),NA(IR,I+3)
1010 FORMAT(I5,I4,I3,I5,F6.3,I4,F6.3,I4,F6.3,I4,F6.3,I4)
    06 DO 8 I=1,IX,4
    8  READ INPUT TAPE 4,1014,KO,T(IR,I),T(IR,I+1),T(IR,I+2),T(IR,I+3)

```

```

1014 FORMAT(I14,4F10.3)
      GO TO 9
      4 IP = 0
        IF(IX-2)66,66,14
      14 GO TO KING,(11,18)
      66 IF(IX)13,13,68
      68 KRO=6
        GO TO 49
      11 READ INPUT TAPE 3,1008
        READ INPUT TAPE 3,1016,KO,KTLOCK,KO,ISP,KO,KO,KO,KO,KO,KO,KO,KO,KO
        X,KO,KO,KO
1016 FORMAT (I5,I4,I3,5I5,I7,7I5)
      IF (KTLOCK - KLOCK) 18,18,81
      18 ASSIGN 11 TO KING
        IP = IP + ISP
        READ INPUT TAPE 3,1008
        DO 15 I=MOP,IP,4
      15 READ INPUT TAPE 3, 1018,KO,XA(I),YA(I),XA(I+1),YA(I+1),XA(I+2),YA(
        9I+2),XA(I+3),YA(I+3)
1018 FORMAT(I16,8F8.2)
      READ INPUT TAPE 3,1008
      MOP=IP +1
      DO 16 I = 1,ISP
      16 READ INPUT TAPE 3,1008
        IF(ISP-24)89,89,19
      19 ASSIGN 18 TO KING
        GO TO 11
      81 ASSIGN 18 TO KING
C      SORTING TRUE AND APPARENTS BY MAGNITUDE OF Y-COORDINATE
      89 KRO = 1
        IF (IP) 66,66,67
      67 ME = NO - 1
      23 NG=0
        DO 24 I=1,ME
          IF(Y(I+1)-Y(I)) 25,24,24
      25 T1=Y(I+1)
          T2=X(I+1)
          Y(I+1)=Y(I)
          X(I+1)=X(I)
          Y(I)=T1
          X(I)=T2
          NG=1
      24 CONTINUE
        IF(NG) 26,26,23
      26 ME=IP-1
        IF (ME) 27,30,27
      27 NG=0

```

```

DO 28 I=1,ME
IF(YA(I+1)-YA(I)) 29,28,28
29 T1=YA(I+1)
T2= XA(I+1)
YA(I+1)=YA(I)
XA(I+1)=XA(I)
YA(I)=T1
XA(I)=T2
NG=1
28 CONTINUE
IF(NG) 30,30,27
C FINDING CLOSEST AT TO TT AND VICE VERSA
30 MA=NO
MB=IP
ASSIGN 33 TO KXC
ASSIGN 36 TO KXD
ASSIGN 40 TO KXE
32 DO 41 I=1,MA
J=1
GO TO KXC,(33,34)
33 DISQ(I) = ((X(I) - XA(J))**2 + (Y(I) - YA(J))**2)**.5
NTJ(I)=J
GO TO 35
34 DJSQ(I) = ((XA(I) - X(J))**2 + (YA(I) - Y(J))**2)**.5
NATI(I)=J
35 DO 31 J=2,MB
GO TO KXD,(36,37)
36 SD = ((X(I) - XA(J))**2 + (Y(I) - YA(J))**2)**.5
IF(SD-DISQ(I)) 38,31,31
38 DISQ(I)=SD
NTJ(I)=J
GO TO 31
37 SD = ((XA(I) - X(J))**2 + (YA(I) - Y(J))**2)**.5
IF(SD-DJSQ(I)) 39,31,31
39 DJSQ(I)=SD
NATI(I)=J
31 CONTINUE
41 CONTINUE
GO TO KXE,(40,42)
40 ASSIGN 34 TO KXC
ASSIGN 37 TO KXD
ASSIGN 42 TO KXE
MA= IP
MB=NG
GO TO 32
42 IF(IM2) 45,700,45
C INTERMEDIATE OF TABLES BEFORE CLASSIFICATION

```

```

700 WRITE OUTPUT TAPE 6,702,NO,1D,KLOCK
702 FORMAT(55H X XA Y YA DIST I J NOTRT=
9I4,I5,I5)
DO 704 I=1,NO
MB=NTJ(I)
704 WRITE OUTPUT TAPE 6,703,(X(I)),(XA(MB)),(Y(I)),(YA(MB)),DISQ(I),
9I,MB
703 FORMAT(5F8.3,2I4)
WRITE OUTPUT TAPE 6,712,IP
712 FORMAT(55H XA X YA Y DIST J I NOAPT=
XI4)
DO 705 I=1,IP
MB=NATI(I)
705 WRITE OUTPUT TAPE 6,703,(XA(I)),(X(MB)),(YA(I)),(Y(MB)),DJSQ(I),
9I,MB
45 L=0
M=0
C CLASS II CONTENTS
DO 43 I=1,NO
MC=NTJ(I)
IF(NATI(MC)-1) 46,47,46
47 NATI(MC)=-1
L=L+1
ICLS2(L)=I
JCLS2(L)=MC
DCLS2(L)=DISQ(I)
GO TO 43
C CLASS I CONTENTS
46 M=M+1
ICLS1(M)=I
JCLS1(M)=MC
DCLS1(M)=DISQ(I)
43 CONTINUE
NECL1=M
NECL2=L
C CLASS III CONTENTS
L=0
DCLS3(L)=0
JCLS3(L)=0
ICLS3(L)=0
DO 44 I=1,IP
IF(NATI(I)) 44,44,48
48 L=L+1
DCLS3(L)=DJSQ(I)
JCLS3(L)=I
ICLS3(L)=NATI(I)
44 CONTINUE
NECL3=L

```

```

XID = ID
XCLK = KLOCK
DO 141 I = 1,NECL2
MB = JCLS2(I)
MA=ICLS2(I)
141 WRITE OUTPUT TAPE 2,989,KLOCK,XID,XCLK,MA,MB,DCLS2(I),X(MA),Y(MA),
XXA(MB),YA(MB)
989 FORMAT (I5,F10.7,F5.0,2I8,5F8.2)
IF(IM3)49,701,49
C INTERMEDIATE OF CLASSES
701 MX=1
ASSIGN 711 TO KXC
706 WRITE OUTPUT TAPE 6,707,MX
707 FORMAT( 4H CLS12,48H X XA Y YA DIST I
X J)
GO TO KXC,(711,713,715)
711 DO 708 I=1,NECL1
MA=ICLS1(I)
MB=JCLS1(I)
708 WRITE OUTPUT TAPE 6,709,(X(MA)),(XA(MB)),(Y(MA)),(YA(MB)),DCLS1(I)
X,MA,MB
709 FORMAT(F12.3,4F8.3,2I4)
MX=2
ASSIGN 713 TO KXC
GO TO 706
713 DO 710 I=1,NECL2
MA=ICLS2(I)
MB=JCLS2(I)
710 WRITE OUTPUT TAPE 6,709,(X(MA)),(XA(MB)),(Y(MA)),(YA(MB)),DCLS2(I)
6,MA,MB
MX=3
ASSIGN 715 TO KXC
GO TO 706
715 IF (NECL3) 49,49,716
716 DO 714 I=1,NECL3
MB=JCLS3(I)
MA=ICLS3(I)
714 WRITE OUTPUT TAPE 6,709,(X(MA)),(XA(MB)),(Y(MA)),(YA(MB)),DCLS3(I)
8,MB,MA
C COMPUTE HISTOGRAMS
49 N=KRO
296 FORMAT(1H1)
505 IF (IH(N)) 99,345,99
345 GO TO (51,52,53,54,55,56,57,58,59,60,61,62,63,64,65 ),N
51 MX=1
NU=NECL1
DO 121 I=1,NU

```

```

121 DATA(I)=DCLS1(I)
125 MAX = (AMXM(N)/QNTA(N)) + 1
    ASSIGN 99 TO KORAN
    GO TO 100
52  MX=2
    NU= NECL2
    DO 123 I=1,NU
123 DATA(I)= DCLS2(I)
    GO TO 125
53  MX=3
    NU=NECL3
    DO 119 I=1, NU
119 DATA(I)=DCLS3(I)
    GO TO 125
54  MX=4
    NU=NO
    DO 127 I=1,NU
127 DATA(I)=DISQ(I)
    GO TO 125
55  MX=5
    NU=IP
    DO 129 I=1,NU
129 DATA(I)=DJSQ(I)
    GO TO 125
56  ASSIGN 860 TO KXP
193 ASSIGN 861 TO KORAN
861 IRT=IRT+1
    IF(IRT-NR) 131,131,133
131 IF(ISAVE(IRT)) 861,861,380
380 NU=ISAVE(IRT)
    DO 132 I=1,NU
    GO TO KXP,(860,166)
860 DATA(I)= BMA(IRT,I)*57.2957795
    GO TO 132
166 DATA(I)=FLOCATF(INA(IRT,I))
132 CONTINUE
862 MAX = (AMXM(N)/QNTA(N)) + 1 .
    GO TO 100
133 IRT=0
    GO TO 99
57  ASSIGN 863 TO KRU
168 NU=0
    DO 135 I=1,NR
135 NU=NU+ISAVE(I)
    KQ=1
    K1=0
    DO 136 I=1,NR
    IF(ISAVE(I)) 136,136,381

```

```

381 ME=ISAVE(I)
      J4=0
      DO 137 II=KQ,ME
        J4=J4+1
        GO TO KRU,(863,167)
863 DATA(II)=BMA(I,J4)*57.2957795
      GO TO 137
167 DATA(II)=FLOATF(NA(I,J4))
137 CONTINUE
      KQ=ME+1
      K1=ISAVE(I)
136 CONTINUE
      GO TO 125
      58 ASSIGN 166 TO KXP
      GO TO 193
      59 ASSIGN 167 TO KRU
      GO TO 168
      60 ASSIGN 182 TO KXG
183 ASSIGN 172 TO KORAN
172 IRT=IRT+1
      IF(IRT-NR) 170,170,133
170 IF(ISAVE(IRT)) 172,172,383
383 NU=ISAVE(IRT)
      JE=0
      DO 171 I=1,NU
        GO TO KXG,(182,181)
182 IF(NA(IRT,I)-KTARG) 171,173,173
173 JE=JE+1
        DATA(JE)=BMA(IRT,I)*57.2957795
        GO TO 171
181 IF(BMA(IRT,I)-SWLD) 171,184,184
184 JE=JE+1
        DATA(JE)=FLOATF(NA(IRT,I))
        GO TO 171
171 CONTINUE
397 NU=JE
      MAX=(AMXM(N)/QNTA(N))+1.
      GO TO 100
      61 ASSIGN 177 TO KXL
175 KQ=1
      K1=0
      J4=0
      DO 176 I=1,NR
        IF(ISAVE(I)) 176,176,384
384 ME=ISAVE(I)+K1
      J5=0
      DO 179 II=KQ,ME
        J5=J5+1

```

```

        GO TO KXL,(177,187)
177 IF(NA(I,J5)-KTARG) 179,178,178
178 J4=J4+1
        DATA(J4)=BMA(I,J5)*57.2957795
        GO TO 179
187 IF(BMA(I,J5)-SWLD) 179,188,188
188 J4=J4+1
        DATA(J4)=FLOATF(NA(I,J5))
179 CONTINUE
        KG=ME+1
        K1=ISAVE(I)+K1
176 CONTINUE
        NU=J4
        GO TO 125
        62 ASSIGN 181 TO KXL
        GO TO 183
        63 ASSIGN 187 TO KXL
        GO TO 175
        64 ASSIGN 99 TO KORAN
        GO TO 164
        65 NU=0
        DO 208 I=1,NR
208 NU=ISAVE(I)+NU
        ASSIGN 99 TO KORAN
        GO TO 165
C COMPUTE HISTOGRAMS
100 DO 110 I= 1,20
        AGT(I) = 0
110 LH(I)=0
        IF(DATA(I))301,300,301
301 DO 111I=1,NU
        IHT = (DATA(I)/QNTA(N)) + 1
        IF (IHT - MAX) 112,113,113
113 IHT = MAX - 1
112 LH(IHT)=LH(IHT)+1
111 LSUM(N,IHT) = LSUM(N,IHT) + 1
        IF(IM1) 300,116,300
116 SUM = 0
        DO 114 I=1,NU
114 SUM=SUM+DATA(I)
        AVE=SUM/FLOATF(NU)
        SUM=0.
        DO 115 I=1,NU
115 SUM=SUM+(DATA(I)-AVE)**2
        DEV=SQRTF(SUM/FLOATF(NU))
        GO TO(151,151,151,151,151,156,157,158,159,160,161,162,163,164,165)
        X,N
151 WRITE OUTPUT TAPE 6,251,MX

```

```

251 FORMAT(31H HISTOGRAM OF ELEMENTS OF CLASS12)
GO TO 290
156 WRITE OUTPUT TAPE 6,256,IRT
256 FORMAT(43H HISTOGRAM OF STROBE WIDTH IN DEGREES/RADAR12,6H/FRAME)
GO TO 290
157 LBJ=6
WRITE OUTPUT TAPE 6,256,LBJ
GO TO 290
158 WRITE OUTPUT TAPE 6,258,IRT
258 FORMAT(44H HISTOGRAM OF NUMBER OF TARGETS/SECTOR/RADAR12,6H/FRAME)
GO TO 290
159 LBJ=6
WRITE OUTPUT TAPE 6,258,LBJ
GO TO 290
160 WRITE OUTPUT TAPE 6,260,KTARG,IRT
260 FORMAT(33H HISTOGRAM OF NO. OF SECTORS WITH13,22H OR MORE TARGETS/
XRADAR12,6H/FRAME)
GO TO 290
161 LBJ=6
WRITE OUTPUT TAPE 6,260,KTARG,LBJ
GO TO 290
162 WRITE OUTPUT TAPE 6,262,SWLD,IRT
262 FORMAT(44H HISTOGRAM OF NO. OF TARGETS/SECTOR OF WIDTHF7.2,14H OR
XMORE/RADAR12,6H/FRAME)
GO TO 290
163 LBJ=6
WRITE OUTPUT TAPE 6,262,SWLD,LBJ
GO TO 290
164 DO 263 I=1,NR
263 WRITE OUTPUT TAPE 6,264,I,(ISAVE(I))
264 FORMAT(25H NO. OF ENTRIES FOR RADAR12,3H ISI3)
GO TO 300
165 WRITE OUTPUT TAPE 6,265,NU
265 FORMAT(39H TOTAL NO. OF ENTRIES FOR ALL RADARS ISI4)
GO TO 300
290 MX=30+N
WRITE OUTPUT TAPE 6,289
289 FORMAT(44H RUN CLK TT MAX QNTA MEAN STD DEV)
WRITE OUTPUT TAPE 6,293,1D,KLOCK,MX,MAX,QNTA(N),AVE,DEV
293 FORMAT(I5,I4,I3,I7,F7.1,2F8.2)
DO 411 I=1,MAX
AQT(I+1)=AQT(I)+QNTA(N)
411 LQT(I) = AQT(I)
WRITE OUTPUT TAPE 6,295,(LQT(I+1),I=1,20)
WRITE OUTPUT TAPE 6,295,(LH(I),I=1,20)
295 FORMAT(20I4)
300 GO TO KORAN,(99,861,172)
99 IF(N-15)410,423,423

```

```

410 N=N+1
    GO TO 505
423 IRT=0
    KOUNT=0
    DO 450 I=1,NO
        IF(((X(I)-RX(JR))**2+(Y(I)-RY(JR))**2)**.5-RANGE)451,451,450
451 IF(((X(I)-RX(KR))**2+(Y(I)-RY(KR))**2)**.5-RANGE)452,452,450
452 IF(((X(I)-RX(LRA))**2+(Y(I)-RY(LRA))**2)**.5-RANGE) 453,453,454
454 IF(LRB) 455,450,455
455 IF(((X(I)-RX(LRB))**2+(Y(I)-RY(LRB))**2)**.5-RANGE)453,453,456
456 IF (LRC) 457, 450, 457
457 IF(((X(I)-RX(LRC))**2+(Y(I)-RY(LRC))**2)**.5-RANGE) 453, 453,450
453 KOUNT=KOUNT+1
450 CONTINUE
    DTRAT = FLOATF ((IP*1000)/NO)
    NMSRAT = (NECL1*1000)/NO
    NHTRAT = (NECL2*1000)/IP
    NFARAT = (NECL3*1000)/IP
    DRT = FLOATF(IP*1000)/KOUNT)
    SMI=NECL1
    TO1=TO1+SMI
    IF(NECL1) 433,434,433
434 IF(KOUNT) 435,433,435
435 IF(NECL2) 433,436,433
436 NECL1 = NO
433 PRINT 983,KLOCK,IP,NECL1,NECL2,NECL3,NFARAT,NHTRAT,
    XNMSRAT,DTRAT,DRT,KOUNT
983 FORMAT (I6,7I7,2F7.0,I7)
    KSUM(1) = KSUM(1) + IP
    KSUM(2) = KSUM(2) + NECL1
    KSUM(3) = KSUM(3) + NECL2
    KSUM(4) = KSUM(4) + NECL3
    KSUM(5) = KSUM(5) + NFARAT
    KSUM(6) = KSUM(6) + NHTRAT
    KSUM(7) = KSUM(7) + NMSRAT
    RSUM(8) = RSUM(8) + DTRAT
    RSUM(9) = RSUM(9) + DRT
    KSUM(10)= KSUM(10)+ KOUNT
    RSUM(11) = RSUM(11) + (DRT**2.)
    NECL1 = 0
    NECL2 = 0
    NECL3 = 0
    IF(KLOCK-MAXLIM) 13,421,421
421 RSUM(8) = RSUM(8)/1000.
    RSUM(9) = RSUM(9)/1000.
    RSUM(11) = RSUM(11)/1000000.
    PRINT 995,(KSUM(I),I = 1,11)
995 FORMAT(6HOSUMS 7I7,2F7.3,I7,F10.6,14H = SUM SQS NDR)

```

```

PRINT 981, ID, NO
PRINT 921
921 FORMAT (37H0 FREQUENCY OF OCCURENCE HISTOGRAMS)
PRINT 922
922 FORMAT(104H HIST.      1  2  3  4  5  6  7  8  9 10 11 1
X2 13 14 15 16 17 18 19 20 SIGMA MEAN)
TO2 = KSUM(3)
DH1 = 0
DM1 = 0
DH = 0
DM = 0
DH2 = 0
DM2 = 0
DEH = QNTA(2)*.5
DEM=QNTA(1)*.5
DO 416 M = 1,20
DH = FLOATF(LSUM(2,M))*DEH
DH1 = (DH * DEH) + DH1
DH2 = DH2 + DH
DM = FLOATF (LSUM(1,M)) * DEM
DM1 = (DM * DEM) + DM1
DM2 = DM2 + DM
DEH = DEH + QNTA(2)
416 DEM = DEM + QNTA(1)
XM = DM2/TO1
SIG1 = ((DM1/TO1) - (XM**2.))**.5
XH = DH2/TO2
SIG2 = ((DH1/TO2) - (XH**2.))**.5
FAC1 = 1000./TO1
FAC2 = 1000./TO2
DO 417 M = 1,20
M1SUM(M) = (FAC1*FLOATF(LSUM(1,M))) + .5
417 M2SUM(M) = (FAC2*FLOATF(LSUM(2,M))) + .5
PRINT 970, (LSUM(1,II),II = 1,20),SIG1,XM
970 FORMAT (4H0 1 18,1914,F10.3,F9.3)
PRINT 971, (M1SUM(I),I = 1,20)
971 FORMAT (4H S 1 18,1914)
PRINT 972, (LSUM(2,II),II = 1,20),SIG2,XH
972 FORMAT (4H0 2 18,1914,F10.3,F9.3)
PRINT 973, (M2SUM(I),I = 1,20)
973 FORMAT (4H S 2 18,1914)
I = 3
427 IF(IH(I)) 428,424,428
424 PRINT 297,I,(LSUM(I,II),II = 1,20)
297 FORMAT (1H0 14,18,1914)
428 I = I+1
IF(I - 15) 427,427,422
END (0,1,1,0,1)

```

SECTION 2

FORTRAN PROGRAM FOR SIMULATION OF THE CORRELATION SYSTEM

The complete system simulation program has evolved from the initial model with additional modifications. These modifications include an environment generator for maneuvering targets, a modification to accept the new environment generator, modification to obtain miss-distance, and a modification for raid size estimate.

The original program and its modifications are described in the following five sections. The sixth section is a description of the off-line historical data analysis program. Each of these sections give a glossary of terms, FORTRAN listing, and flow charts.

Input to the simulation program is made by cards only. For each individual case, there are three input cards plus one for each target in the raid.

1. CARD 1

K1, K2, K3, K4, K5, K6, K7, K8, K9, K0 These inputs are used to control the optional output. An integer causes output, while a zero will bypass output. The outputs associated with each "switch" are listed below.

K1 -- the frequency distribution of target clusters

K2 -- no longer in use

K3 -- the X and Y coordinates of all targets (true and apparent)

K4 -- no longer in use

K5 -- no longer in use

K6 — if the number of the true targets scored as "misses", the number of the nearest apparent target and the distance between them

K7 — the target numbers of targets in a "hit" pair and the distance between them, along with the locations of these targets

K8 — the number of the apparent targets scored as "false alarms", number of the nearest true, and the distance between

K9 — no longer in use

K0 — the frequency distribution of the signal-to-noise ratio

2. CARD 2

NT, CLOCKS, CLOCKL, CASE, AZMUL, DELMUL

NT — the number of targets in the raid

CLOCKS — the frame number at which the simulation is to begin calculation

CLOCKL — the last frame the simulation is to evaluate

CASE — the case number (used for identification)

AZMUL — a parameter which controls the range of the azimuth uncertainty

DELMUL — a parameter which controls the range of the delta uncertainty

3. CARD 3

RMAX, RMIN, RMIN2, AZRES, DELRES

RMAX — the maximum range at which the radar can detect a target (horizon cut-off)

RMIN — the range at which the noise level is no longer a function range

AZRES - this is related to the azimuth resolution capabilities of the radar. If the difference in azimuth of two or more, targets is less than this parameter, the targets are not resolved in azimuth

DELRES - this is related to the delta resolution capabilities of the radar. If the difference in delta of two, or more, targets are less than this parameter, the targets are not resolved in azimuth (delta is the difference in path length between target-omni and target-search radar)

4. CARD 4

X(I), Y(I), DELX(I), DELY(I)

X(I) - the X coordinate of the target

Y(I) - the Y coordinate of the target

DELX(I) - the distance the target will travel in a direction parallel to the X axis in a single frame

DELY(I) - the distance the target will travel in a direction parallel to Y axis in a single frame

There are NT CARD 4's per case. There must be one for each target in the raid. To run successive cases, all four types of cards must be repeated for each case.

Through-out the simulation, frequency distributions are generated and stored. They are part of the optional and/or normal output as follows:

Normal Program Output

1. Per Frame (printed-on-line, one line per frame)
 - a. Frame number
 - b. Number of apparent targets (number of detections) in the frame
 - c. Number of targets in Sector 1

- d. Number of targets in Sector 2
- e. Number of targets behind the radar
- f. Number of "misses"
- g. Number of "hits"
- h. Number of "false alarms"
- i. False alarm rate (number of "false alarms" divided by the number of detections)
- j. Hit rate (number of "hits" divided by the number of true targets)
- k. Miss rate (number of "misses" divided by the number of true targets)
- l. Detection ratio (number of detections divided by the number of true targets)
- m. Number of azimuth unresolved clusters
- n. Total number of targets involved in azimuth and delta unresolved clusters during this frame
- o. Number of azimuth unresolved clusters that are completely resolved in delta
- p. Number of targets detected
- q. An indicator of the performance of the random number generator (a computed theoretical number of detections)
- r. The number of targets that are jammed (how often is the noise too great for the signal to "heard")
- s. Number of targets detected divided by number of targets in-range

2. Per Case (Printed on-line)

- a. Summations of items 2, 3, 4, 5, 6, 7, 8, 14, 15, and 19
- b. Frequency distribution of miss distance between hit pairs, and miss distances that are associated with miss pairs (two (2) separate distributions)

Optional Program Output

1. Per Frame

- a. The X, Y coordinates of all true and apparent targets
- b. The target number of the true targets scored as "misses", the number of the nearest apparent target and the distance between them
- c. The target number of targets in a "hit" pair and the distance between them, along with the locations of those targets
- d. The number of the apparent targets scored as "false alarms", the number of the nearest true target, and the distance between them

2. Per Case

- a. Frequency distribution of signal-to-noise ratio
- b. Frequency distribution of all clusters encountered

Most of the optional frame output is on magnetic tape. Each output is independent and any combination may be obtained. The frequency distributions are listed on the on-line printer and are not on tape.

2.1 ORIGINAL PROGRAM

2.1.1 Glossary

2. 1. 1. 1 Subscripted Terms

<u>AZ</u>	Azimuth of the target, measured clockwise, at the search radar from west
<u>DELX</u>	The X increment. The distance the target will travel in a direction parallel to the X axis in a single frame
<u>DELY</u>	The Y increment. The distance the target will travel in a direction parallel to the Y axis in a single frame
<u>DIFX</u>	The initial X coordinate minus 500
<u>DIFY</u>	The initial Y coordinate minus 500
<u>DISFS</u>	The distance in nautical miles between the true and apparent target in a "false alarm" pair
<u>DISH</u>	The number of "hits" in a given hit distance interval of the frequency distribution
<u>DISHT</u>	The distance in nautical miles between the true and apparent targets in a "hit" pair
<u>DISMS</u>	The distance in nautical miles between the true and apparent targets in a "miss" pair
<u>DIST</u>	The number of misses in a given miss distance interval of the frequency distribution of "miss" pairs
<u>LAZ</u>	The target number of the J^{th} target in an azimuth unresolved list
<u>LAZDEL</u>	The target number in an azimuth-delta unresolved list
<u>LASDEX</u>	The number of targets in an azimuth-delta unresolved cluster
<u>LJTH</u>	The number of the target in an azimuth unresolved cluster, from which the targets are not resolved. The "target of interest"

<u>LJTHX</u>	The number of targets in a given cluster. Used for the frequency distribution of clusters
<u>LTOT</u>	The number of targets left in the final cluster after all delta resolved targets have been removed and all targets not azimuth-delta resolved from the azimuth-delta unresolved cluster are added
<u>LTOTX</u>	The same as <u>LTOT</u> except that this is the summation over the entire run. Used for frequency distribution
<u>M</u>	The dummy location used to store the sector number of the target and other information
<u>NFSA</u>	The target number of the apparent target in a "false alarm" pair
<u>NFST</u>	The target number of the true target in a "false alarm" pair
<u>NHTA</u>	The target number of the apparent target in a "hit" pair
<u>NHTT</u>	The target number of the true target in a "hit" pair
<u>NMSA</u>	The target number of the apparent target in a "miss" pair
<u>NMST</u>	The target number of the true target in a "miss" pair
<u>NRESOL</u>	The list of true target numbers of the unresolved targets
<u>NSIGNO</u>	An interval of the signal to noise ratio in the signal to noise frequency distribution
<u>NTTOAA</u>	The number of the apparent target closest to a given true

R1 Range from the target to omni 1
R2 Range from the target to omni 2
RO Range from the target to the search radar
X The X coordinate of the true target
XAP The X coordinate of the apparent target
Y The Y coordinate of the true target
YAP The Y coordinate of the apparent target

2. 1. 1. 2 Non-Subscripted Variables

ADELTA The difference in ranges from radar and from omni
 to jammer (used only in delta unresolved clusters)
AKI The sum of the squares of NDR
AK9 The sum of NDR's
AMB The sum of the weighting factor times the noise of
 the target of all targets in an azimuth unresolved
 cluster
AMBL The same as AMB except the noise has minimum
 restrictions placed on it
ANDR The same as NDR except it is in floating point.
 The number of detections divided by the number
 of targets in range
AND2 ANDR squared
ANGLE The angle of the apparent target in polar coor-
 dinates
ANO A random number multiplied by AZMUL

<u>AOSIGA</u>	Noise to Signal ratio at the omni receiver
<u>AOSIGO</u>	Noise to Signal ratio at the search radar
<u>AZMUL</u>	Scales the magnitude of the uncertainty in the azimuth of the apparent target
<u>AZRES</u>	The azimuth limits which two or more targets must be within to be clarified as azimuth unresolved
<u>A2T</u>	The azimuth of the target. Different reference than AZ
<u>BARH</u>	The mean distance between "hit" pairs. This is for the entire run
<u>BARM</u>	The mean distance between "miss" pairs. This is for the entire run
<u>BNO</u>	A random number multiplied by <u>DELMUL</u>
<u>CASE</u>	An arbitrary number assigned for identification of input parameters
<u>CLOCK</u>	The number of the frame which is being evaluated
<u>CLOCKL</u>	The last frame that the simulation is to evaluate
<u>CLOCKS</u>	The frame number with which the simulation is to begin
<u>CONV</u>	Conversion factor to change radians to degrees
<u>DELMUL</u>	Constant used to fix the magnitude of the delta uncertainty for an apparent target
<u>DELRES</u>	The delta limits which two or more targets must be within to be classified as delta unresolved

<u>DELTA</u>	The difference in ranges from jammer to search radar, and from jammer to omni receiver
<u>DIFRX</u>	The difference in the true X and the apparent X coordinate
<u>DIFRY</u>	The difference in the true Y and the apparent Y coordinate
<u>DISSQ</u>	Shortest distance squared between a true target and an apparent target. Used in scoring only
<u>DISTA</u>	Squared root of <u>DISSQ</u>
<u>DIXH</u>	Summation of "hits" that occur in the run
<u>DIXM</u>	Summation of "misses" that occur in the run
<u>FLJ</u>	A weighting factor which gives more weight to the targets closest to the J^{th} target and less weight to those near the edge of the beam (used for azimuth unresolved clusters only)
<u>FRACT</u>	The reciprocal of the range squared of the target in an azimuth unresolved cluster
<u>FRE1</u>	Summation of "hits" in a given interval times the midpoint of that interval
<u>FRE2</u>	Summation of "misses" in a given interval times the midpoint of that interval
<u>FRED</u>	Midpoint of the interval (upper limit - lower limit) / 2; this is for "miss" intervals
<u>FRED2</u>	Summation of "misses" in a given interval times the midpoint of that interval, the quantity squared
<u>FREQ</u>	Midpoint of "hit" interval

FREQ2 Same as FRED2 only it applied to "hits"

GAIN Gain in side lobes divided by the gain in the search radar

HIG, HIGH, HIGHR Arbitrary upper limits for frequency distributions

IAP, ID, IND, I, IJ, ITR These are general index registers

J The index register that controls the number of the target of interest

K The number of apparent targets in a given frame

KABAS An index used to denote an unresolved target

KEN Output type number. Used on output tape 3 for identification

K0, K1, K2, K3, K4, K5, K6, K7, K8, K9 (see detailed write-up) Used to control optional output

KK Summation of apparent targets taken over the entire run

KK2 Summation of NSUMM1 for run (targets in sector 1)

KK3 Summation of NSUMM2 for run (targets in sector 2)

KK4 Summation of "missed" targets for run

KK5 Summation of "hit" targets for run

KK6 Summation of "false alarms" for run

KK7 Summation of KOUNT7 for run

KK8 Summation of KOUNT8 for run

KLOO Index register used for targets in a delta unresolved cluster

KNTSUM An indicator of the performance of the random number generator. Should agree closely with KOUNT5

KOUNT1 The number of targets whose probability of detection is unity

KOUNT2 The number of targets whose probability of detection is 0.6

KOUNT3 The number of targets whose probability of detection is 0.2

KOUNT4 Number of targets whose probability of detection is zero

KOUNT5 Number of targets detected (equal to K)

KOUNT6 Number of azimuth unresolved clusters

KOUNT7 Total number of targets that are not azimuth or delta resolved

KOUNT8 The number of azimuth unresolved clusters that are resolved in delta

KOUNT9 The number of times that the jammers succeed in completely jamming the search radar (not just denying range information)

L Number of targets in a given azimuth unresolved cluster

LINDX Same as L

LL1 Summation of the LJTHX's

<u>LL2</u>	Summation of the <u>LAZDEX</u> 's
<u>LL3</u>	Summation of the <u>LTOTX</u> 's
<u>LXX</u>	Standardized occurrence of "hits" in hit frequency distribution
<u>LYY</u>	Standardized occurrence of "misses" in miss frequency distribution
<u>MDS</u>	Equal to $\text{DISMS}(\text{NMS}) / 5$; this is the index used to compute the miss frequency distribution
<u>MHS</u>	Equal to $\text{DISHT}(\text{NHT}) / 4$; this is the index used to compute the hit frequency distribution
<u>NATOTT</u>	Number of the true target that is nearest to a given apparent target (used as an index)
<u>NDR</u>	Detection ratio; the number of detections divided by the number of targets in-range (in tenths of percent)
<u>NDTRAT</u>	The number of detections divided by the total number of targets in raid (in tenths of percent)
<u>NEEL</u>	Number of targets in an azimuth-delta unresolved cluster
<u>NFARAT</u>	Number of "false alarms" divided by the number of targets detected (in tenths of percent)
<u>NFS</u>	Number of "false alarms" in a frame
<u>NHTRAT</u>	Number of "hits" divided by the number of detections (in tenths of percent)
<u>NHT</u>	The number of "hits" in a given frame

<u>NMSRAT</u>	The number of "misses" divided by the total number of targets in the raid (in tenths of percent)
<u>NMS</u>	Number of "misses" in a given frame
<u>NS</u>	Equal to signal-to-noise ratio divided by 500; used as the index register in computing signal to noise frequency distribution
<u>NSUMM1</u>	Number of targets in Sector 1 during a given frame
<u>NSUMM2</u>	Number of targets in Sector 2 during a given frame
<u>NSUMM3</u>	Number of targets behind the radar during a given frame
<u>NT</u>	The number of true targets taking part in the raid
<u>O</u>	A dummy symbolic address used to initialize the random number generator
<u>RADOMA</u>	The summation of the reciprocal of the range squared, when $RO(I) < RMIN$, $RO(I)$ is replaced by $RMIN$
<u>RANGE</u>	The distance from the apparent target to the search radar
<u>RMAX</u>	Maximum range at which the radar can "see" a target (horizon cut-off)
<u>RMIN</u>	The distance at which noise from jammer is no longer a function of distance
<u>RMIN2</u>	The reciprocal of <u>RMIN</u> squared
<u>SIGMAH</u>	Sigma of the "hits" or the standard deviation of the distance between hit pairs
<u>SIGMAN</u>	Sigma of the "misses" or the standard deviation of the distance between miss pairs
<u>SIGNO</u>	Signal to noise ratio

SUM Summation of the number of "hits"

SUML1 The summation of the reciprocal of the squares of the ranges from omni one to target; if $R1(i) < RMIN$, $RMIN$ replaces $R1(i)$ (this is the noise at omni 1)

SUML2 The summation of the reciprocal of the squares of the ranges from omni 2 to the target if $R2(I) < RMIN$, $RMIN$ replaces $R2(I)$

SUMLO Same as SUMO but with minimum restrictions

SUMO The summation of the reciprocal of the squares of the ranges from the search radar to target (no minimum restriction)

TARGML The signal of the target of interest (equal to FRACT)

TDIFRX Temporary storage for the distance between the X coordinate of the true target and the X coordinate of the apparent (used in the scoring process only)

TDIFRY Same as TDIFRX only it is the difference in Y

X1 X coordinate of omni 1

X2 X coordinate of omni 2

XAPP X coordinate of the apparent target

XLOW, XLO, XHIG, XLOWR Interval limits for frequency distribution

XO X coordinate of search radar

XTRU X coordinate of a true target while simulation is testing for nearest apparent

XX Counter used for controlling output; whenever XX is larger than 50, the program will restore the page on the online printer

Y1 Y coordinate of omni 1

Y2 Y coordinate of omni 2

YAPP Y coordinate of apparent target

YO Y coordinate of search radar

YTRU Y coordinate of true target used while testing for nearest apparent

2.1.2 FORTRAN Listing

```

DIMENSION LUTHX(50),LUTYX(50),LAZDEX(50),NSIGNO(155)
DIMENSION NTICAA(100),DISTT(100),NMST(100),NMSA(100),DISMS(100),
INHTI(100),NHTA(100),DISHI(100),NFST(100),NFSA(100),DISFS(100)
DIMENSION DIST(31),DISH(31)
DIMENSION DIFX(100),DIFY(100)
DIMENSION X(100),Y(100),DELX(100),DELY(100),M(100),RO(100),RI(100),
I,K2(100),AZ(100)
DIMENSION NRESOL(100),LUTH(50),LAZ(50),XAP(100),YAP(100),LAZDEL(50),
LTOF(50)
READ 400,X1,Y1,X2,Y2,GAIN
READ 400,SIG1,SIG2,SIG3,CON1,CON2
1500 DO 5353 I=1,50
LUTX(I)=0
5333 LAZDEX(I)=0
SUX=0.
SUM=0.
FRE1=0.
FRE2=0.
FRLO=0.
FREQ=0.
FREQ2=0.
FREQ2=0.
DXH=0.
DXM=0.
PRINT 845

```

```

DO 8050 I =1,31
DISH(I)=0.
8050 DIST(I)=0.
5002 FORMAT(90H
XDATA. BENDIX SYSTEMS DIVISION 704.)
DO 5334 I=1,50
5334 LJTHX(I)=0
DO 8000 I=1,153
8000 NSIGNO(I)=0
O=0.
READ 350,K1,K2,K3,K4,K5,K6,K7,K8,K9,K0
350 FORMAT(10I1)
READ 3,NT,CLOCKS,CLOCKL,CASE,AZMUL,DELMUL
XX=CLOCKS
3 FORMAT(I5,5F10.1)
READ 400,RMAX,RMIN,RMIN2,AZRES,DELRES
PRINT 5002
PRINT 700,CASE,NT
700 FORMAT(12H CASE NUMBERF8.0,4H NTI4)
PRINT 1100,CLOCKS,CLOCKL,AZMUL,DELMUL
1100 FORMAT(16H CLOCKS F4.0,7H CLOCKL F6.0,6H AZMUL F8.6,7H DELMU
1LF8.6)
PRINT 1101,RMAX,RMIN,RMIN2,AZRES,DELRES
1101 FORMAT((14H RMAX F6.0,5H RMIN F6.0,6H RMIN2 F8.7,6H AZRES F8.
17,7H DELRES F8.6)///)
PRINT 701
701 FORMAT((118H CLOCK K NSUMM1 NSUMM2 NSUMM3 NMS NHT NFS NFARAT NHT
1RAT NMSRAT NDTRAT KOUNT6 KOUNT7 KQUNT8 KOUNT5 KNISUM KOUNT9 NDR
2 )///)
400 FORMAT(5F10.2)
READ 4,((X(I),Y(I),DELX(I),DELY(I)),I=1,NT)
4 FORMAT(4F10.2)
CONV=57.29577
XO=500.
YO=500.
KK=0
KK1=0
KK2=0
KK3=0
KK4=0
KK5=0
KK6=0
KK7=0
KK8=0
AK9=0.
AK1=0.
CLOCK=CLOCKS
IF(CLOCKS) 5,7,5

```

TLQ 8 PROGRAM

```

5      DO 6 I=1,NT
      X(I)=X(I)+CLOCKS*DEL X(I)
6      Y(I)=Y(I)+CLOCKS*DEL Y(I)
7      KOUNT1=0
      KOUNT2=0
      KOUNT3=0
      KOUNT4=0
      KOUNT5=0
      KOUNT6=0
      KOUNT7=0
      KOUNT8=0
      KOUNT9=0
      NSUMM1=0
      NSUMM2=0
      NSUMM3=0
      SUMLO=0.
      SUMO =0.
      SUML1=0.
      SUML2=0.
      DO 50 I=1,NT
      DIFX(I)=X(I)-XO
      DIFY(I)=Y(I)-YO
      RO(I)=SQRTF(DIFX(I)**2+DIFY(I)**2)
      IF(RO(I)-RMAX) 9,9,8
9      IF(RO(I)-RMIN) 500,500,501
500    SUMLO=SUMLO+RMIN2
      GO TO 502
501    SUMLO=SUMLO+1./RO(I)**2
502    SUMO=SUMO+1./RO(I)**2
8      R1(I)=SQRTF((X(I)-X1)**2+(Y(I)-Y1)**2)
      IF(R1(I)-RMAX) 11,11,650
11     IF(R1(I)-RMIN) 652,651,651
651    SUML1=SUML1+1./R1(I)**2
      GO TO 650
652    SUML1=SUML1+RMIN2
650    CONTINUE
13     R2(I)=SQRTF((X(I)-X2)**2+(Y(I)-Y2)**2)
      IF(R2(I)-RMAX) 15,15,660
15     IF(R2(I)-RMIN) 662,661,661
661    SUML2=SUML2+1./R2(I)**2
      GO TO 660
662    SUML2=SUML2+RMIN2
660    CONTINUE
      IF(DIFY(I)) 18,18,19
18     M(I)=3
      NSUMM3=NSUMM3+1
      GO TO 50

```

```

19   IF(R0(I)-RMAX) 40,40,22
40   IF(DIFX(I))20,23,23
20   IF(R1(I)-RMAX) 21,21,22
22   M(I)=4
     GO TO 50
21   M(I)=1
     NSUMM1=NSUMM1+1
     GO TO 26
23   IF(R2(I)-RMAX) 25,25,22
25   M(I)=2
     NSUMM2=NSUMM2+1
26   AZ(I)=ATN1P(DIFY(I),-DIFX(I))
50   CONTINUE
     DO 56 I=1,50
     LJTH(I)=0
     LAZ(I)=0
     LAZDFL(I)=0
56   LTOT(I)=0
     K=0
     LINDX=0
     DO 100 J=1,NT
     IF(M(J)) 100,58,58
58   IF(XABSF(M(J))-2) 60,60,100
60   L=0
     DO 76 I=1,NT
     IF(XABSF(M(I))-XABSF(M(J))) 76,62,76
62   IF((XABSF(AZ(J)-AZ(I)))-AZRFS) 64,64,76
64   L=L+1
     NRESOL(L)=I
76   CONTINUE
72   SIDLO=SUMO
     AMB=0.
     AMBL=0.
74   IF(XABSF(M(J))-1) 75,71,75
75   RADOMA=SUML2
     GO TO 77
71   RADOMA=SUML1
77   DO 88 I=1,L
     KBAS=NRESOL(I)
     FRACT=1./RO(KBAS)**2
     FLJ=1.-ABSF((AZ(J)-AZ(KBAS))/AZRFS)
     AMB=AMB+FLJ*FRACT
     IF(FRACT-RMIN2) 275,275,274
274   FRACT=RMIN2
275   AMBL=AMBL+FLJ*FRACT
     IF(J-KBAS) 88,276,88
276   TARGML=FRACT
     IF(XABSF(M(J))-1) 83,85,83

```

```

83   IF(R2(KBAS)-RMIN) 680,680,681
680  RANGMA=RMIN**2
      GO TO 682
681  RANGMA=R2(KBAS)**2
682  CONTINUE
      DELTA=RO(J)-R2(J)
      GO TO 88
85   IF(R1(KBAS)-RMIN) 690,690,691
690  RANGMA=RMIN**2
      GO TO 692
691  RANGMA=R1(KBAS)**2
692  CONTINUE
      DELTA=RO(J)-R1(J)
88   CONTINUE
      IF(AMBL-(SUMO-AMB)*GAIN)4000,89,89
4000  KOUNT9=KOUNT9+1
      GO TO 100
89   AOSIGO=((SUMO-AMB)*GAIN+AMBL-TARGML)/TARGML
      AOSIGA=(RANGMA*ADOMA)-1.
      SIGNO=1./(2.+AOSIGO+AOSIGA+AOSIGO*AOSIGA)
      NS=SIGNO*500.
      NS=NS+1
      IF(NS-152) 8002,8002,8001
8001  NSIGNO(152)=NSIGNO(152)+1
      GO TO 8003
8002  NSIGNO(NS)=NSIGNO(NS)+1
8003  CONTINUE
      IF(SIGNO-SIG1) 99,101,101
101  KOUNT1=KOUNT1+1
      GO TO 112
99   IF(SIGNO-SIG2) 103,102,102
102  KOUNT2=KOUNT2+1
      IF(RAM2BF(0)-.6) 112,250,250
103  IF(SIGNO-SIG3) 105,104,104
104  KOUNT3=KOUNT3+1
      IF(RAM2BF(0)-.2) 112,250,250
105  KOUNT4=KOUNT4+1
250  CONTINUE
      GO TO 100
112  KOUNT5=KOUNT5+1
      IF(L-1) 113,113,114
114  KOUNT6=KOUNT6+1
      LINDX=LINDX+1
      IF(LINDX-50) 73,73,122
73   LJTH(LINDX) =J
      LJTHX(L)=LJTHX(L)+1
      LAZ(LINDX) =L
      GO TO 122

```

```

113 A2T=AZ (J)
115 ANO=(RAM2BF(O)-.5)*AZMUL
    A2T=A2T+ANO
    BNO=(RAM2BF(O)-.5)*DELMUL
    DELTA=DFLTA+ BNO
    IF(XABSF(M(J))-1) 116,116,117
116 ANGLF=A2T+CON1
    GO TO 118
117 ANGLF=A2T-CON2
118 RANGE=(2500.-DELTA**2)/(100.*COSF(ANGLF)-2.*DELTA)
1298 CONTINUE
    K=K+1
120 XAP(K)=XO-RANGE*COSF(A2T)
121 YAP(K)=YO+RANGE*SINF(A2T)
    GO TO 100
122 NEEL=0
    DO 130 I=1,L
    KBAS=NRESOL(I)
    IF(XABSF(M(J))-1) 123,123,124
124 IF(ABSF(RO(KBAS)-R2(KBAS)-DELTA)-DFLRES) 127,127,130
123 IF(ABSF(RO(KBAS)-R1(KBAS)-DFLTA)-DELRES) 127,127,130
127 NFFL=NFFL+1
    M(KBAS)=-M(KBAS)
    NRESOL(NEEL)=NRESOL(I)
130 CONTINUE
    LAZDEX(NEEL)=LAZDEX(NFEL)+1
    IF(NFEL-1) 517,517,132
517 KOUNT8=KOUNT8+1
    IF(LINDX-50)7000,7000,113
7000 LAZDFL(LINDX)=1
    LTOT(LINDX)=1
    GO TO 113
    132 KLOO=1
    IF(LINDX-50)133,133,134
133 LAZDFL(LINDX)=NFEL
134 IF(NRESOL(KLOO)-J) 135,150,135
135 IND=NRESOL(KLOO)
    IF(XABSF(M(J))-1) 136,136,137
136 ADELTA=RO(IND)-R1(IND)
    GO TO 140
137 ADELTA=RO(IND)-R2(IND)
140 DO 149 I=1,NT
    IF(M(I)) 149,777,777
777 IF(XABSF(M(I))-XABSF(M(J)))149,141,149
141 IF(ABSF(AZ(I)-AZ(KLOO))-AZRFS) 142,142,149
142 IF(XABSF(M(J))-1) 146,146,143
146 IF(ABSF(RO(I)-R1(I)-ADELTA)-DELRFSS) 144,144,149
144 NEEL=NFFL+1

```

```

NRESOL(NEEL)=I
M(I)=-M(I)
GO TO 149
143 IF(ABSF(RO(I)-R2(I)-ADELTA)-DELRES) 144,144,149
149 CONTINUE
150 KLOO=KLOO+1
IF(KLOO-NEFL) 134,134,152
152 LTOTX(NEEL)=LTOTX(NEEL)+1
IF(LINDX-50)155,155,156
155 LTOT(LINDX)=NEEL
156 SUMAZ=0.
DENOM=NEEL
KOUNT7=KOUNT7+NEEL
A2T=0.
DO 160 I=1,NEEL
KBAS=NRESOL(I)
IF(AZ(KBAS)-A2T) 160,160,277
277 A2T=AZ(KBAS)
IF(XABSF(M(J))-1)278,278,279
278 DELTA=RO(KBAS)-R1(KBAS)
GO TO 160
279 DELTA=RO(KBAS)-R2(KBAS)
160 CONTINUE
GO TO 115
100 CONTINUE
NMS=0
NHT=0
NFS=0
IF(K)570,571,570
571 DO 572 ITR=1,NT
NMS=NMS+1
NMST(NMS)=ITR
NMSA(NMS)=-0
572 DISMS(NMS)=-0
GO TO 575
570 DO 168 ITR=1,NT
IAP=1
NTTOAA(ITR) =1
XTRU=X(ITR)
YTRU=Y(ITR)
DIFRX=ABSF(XTRU-XAP(IAP))
DIFRY=ABSF(YTRU-YAP(IAP))
DISSQ=DIFRY**2+DIFRX**2
DO 166 IAP=2,K
TDIFRX=ABSF(XTRU-XAP(IAP))
TDIFRY=ABSF(YTRU-YAP(IAP))
IF(TDIFRX-DIFRX) 165,164,164

```

```

164 IF(TDIFRY-DIFRY) 165,166,166
165 TDISSQ=TDIFRX**2+TDIFRY**2
IF(TDISSQ-DISSQ) 167,166,166
167 NTTOAA(ITR)=IAP
DISSQ=TDISSQ
DIFRX=TDIFRX
DIFRY=TDIFRY
166 CONTINUE
168 DISTT(ITR)=SQRTF(DISSQ)
171 DO 178 IAP=1,K
ITR=1
NATOTT=1
YAPP=YAP(IAP)
XAPP=XAP(IAP)
DIFRX=ABSF(X(ITR)-XAPP)
DIFRY=ABSF(Y(ITR)-YAPP)
DISSQ=DIFRX**2+DIFRY**2
172 DO 175 ITR=2,NT
TDIFRX=ABSF(X(ITR)-XAPP)
TDIFRY=ABSF(Y(ITR)-YAPP)
IF(TDIFRX-DIFRX) 169,170,170
170 IF(TDIFRY-DIFRY) 169,175,175
169 TDISSQ=TDIFRX**2+TDIFRY**2
IF(TDISSQ-DISSQ) 173,175,175
173 NATOTT=ITR
DISSQ=TDISSQ
DIFRX=TDIFRX
DIFRY=TDIFRY
175 CONTINUE
DISTA=SQRTF(DISSQ)
IF(XABSF(NTTOAA(NATOTT))-IAP) 176,177,176
177 NHT=NHT+1
NHTT(NHT)=NATOTT
NHTA(NHT)=IAP
DISHT(NHT)=DISTT(NATOTT)
MHD=DISHT(NHT)*4.
MHD=MHD+1
IF(MHD-31) 1404,1405,1405
1405 DISH(31)=DISH(31)+1.
GO TO 1406
1404 DISH(MHD)=DISH(MHD)+1.
1406 CONTINUE
NTTOAA(NATOTT)=-NTTOAA(NATOTT)
GO TO 178
176 NFS=NFS+1
NFST(NFS)=NATOTT
NFSA(NFS)=IAP
DISFS(NFS)=SQRTF(DISSQ)

```

```

178   CONTINUE
      DO 440 ITR=1,NT
      IF(NTTOAA(ITR)) 440,440,442
442   NMS=NMS+1
      NMST(NMS)=ITR
      NMSA(NMS)=NTTOAA(ITR)
      DISMS(NMS)=DISTT(ITR)
      MDS=DISMS(NMS)/5.
      MDS=MDS+1
      IF(MDS-31) 1400,1401,1401
1401  DIST(31)=DIST(31)+1.
      GO TO 1402
1400  DIST(MDS)=DIST(MDS)+1.
1402  CONTINUE
440   CONTINUE
575   CONTINUE
      NFARAT=(1000*NFS)/K
      NMSRAT=(1000*NMS)/NT
      NDTRAT=(1000*K)/NT
      NHTRAT=(1000*NHT)/K
      KNTSUM=(10*KOUNT1+6*KOUNT2+2*KOUNT3)
      NDR=(1000*K)/(NSUMM1+NSUMM2)
      ANDR=NDR
      ANDR=ANDR/1000.
      AND2=ANDR**2
      AK1=AK1+AND2
      XX=XX+1.
      IF(XX-(51.+CLOCKS))844,843,844
843  PRINT 845
      XX=CLOCKS
      PRINT 5002
      PRINT 700,CASE,NT
845  FORMAT(IH1)
      PRINT 701
844  CONTINUE
      PRINT 702,CLOCK,K,NSUMM1,NSUMM2,NSUMM3,NMS,NHT,NFS,NFARAT,NHTRAT,
1NMSRAT,NDTRAT,KOUNT6,KOUNT7,KOUNT8,KOUNT5,KNTSUM,KOUNT9,NDR
      KK=KK+K
      KK2=KK2+NSUMM1
      KK3=KK3+NSUMM2
      KK4=KK4+NMS
      KK5=KK5+NHT
      KK6=KK6+NFS
      KK7=KK7+KOUNT6
      KK8=KK8+KOUNT8
      AK9=AK9+ANDR
702  FORMAT(F6.0,I4,I4,3I7,2I4,I5,3I7,I6,3I7,2I8,I5)
      IF(K3) 8075,8075,8076

```

```

8076 CONTINUE
WRITE OUTPUT TAPE 4,900,CASE,CLOCK,NT,K
900 FORMAT(F11.6,F10.0,2I5)
DO 901 I=1,NT
901 WRITE OUTPUT TAPE 4,902,X(I),Y(I)
902 FORMAT(2F15.4)
IF(K) 6000,6001,6000
6000 DO 903 I=1,K
903 WRITE OUTPUT TAPE 4,902,XAP(I),YAP(I)
6001 CONTINUE
8075 CONTINUE
359 IF(K6) 361,361,362
362 KEN=6
IF(NMS) 2100,2200,2100
2100 DO 906 I=1,NMS
906 WRITE OUTPUT TAPE 3,907,KEN,CASE,CLOCK,NMST(I),NMSA(I),DISMS(I)
2200 CONTINUE
361 IF(K7) 363,363,364
364 KFN=7
IF(NHT) 2101,2102,2101
2101 DO 908 I=1,NHT
IS=NHTA(I)
ID=NHTT(I)
908 WRITE OUTPUT TAPE 3,907,KEN,CASE,CLOCK,NHTT(I),NHTA(I),DISHT(I),
IX(ID),Y(ID),XAP(IS),YAP(IS)
2102 CONTINUE
363 IF(K8) 365,365,366
366 KEN=8
IF(NFS) 2103,2104,2103
2103 DO 909 I=1,NFS
909 WRITE OUTPUT TAPE 3,907,KEN,CASE,CLOCK,NFST(I),NFSA(I),DISFS(I)
365 CONTINUE
2104 CONTINUE
CLOCK=CLOCK+1.
IF(CLOCK-CLOCKL) 999,999,301
999 DO 998 I=1,NT
Y(I)=Y(I)+DFLY(I)
998 X(I)=X(I)+DFLX(I)
GO TO 7
301 KEN=0
5555 FORMAT(4H TOTI6,I5,I6,I13,2I5,I31,I14,F28.3)
PRINT 5555,KK,KK2,KK3,KK4,KK5,KK6,KK7,KK8,AK9
PRINT 6500,AK1
6500 FORMAT(F120.6)
5557 FORMAT(7H TOTALI14,I9.1I5)
PRINT 845
PRINT 1666

```

```

PRINT 700,CASE,NT
PRINT 5002
1666 FORMAT(56H                HITS                MISSES
1)
PRINT 1667
1667 FORMAT(70H          LOWER  UPPER  NUMBER  STD FREQ          LOWER  UPPER  NUM
1BER  STD FREQ  )
1408 FORMAT(37H          HITS                MISSES  )
1409 FORMAT(46H LOWER  UPPER  NUMBER          LOWER  UPPER  NUMBER  )
XLOW=0.
HIGH=5.
XLO=0.
HIG=.25
DO 1668 I=1,31
SUX=SUX+DIST(I)
1668 SUM=SUM+DISH(I)
SUX=1000./SUX
SUM=1000./SUM
DO 1411 I=1,31
LXX=DISH(I)*SUM+.5
LYY=DIST(I)*SUX+.5
PRINT 1410,XLO,HIG,DISH(I),LXX,XLOW,HIGH,DIST(I),LYY
FREQ=(HIG-XLO)/2.+XLO
FRED=(HIGH-XLOW)/2.+XLOW
FREQ2=FREQ2+FREQ**2*DISH(I)
FRED2=FRED2+FRED**2*DIST(I)
FRE1=FRE1+FREQ*DISH(I)
FRE2=FRE2+FRED*DIST(I)
DXH=DXH+DISH(I)
DXM=DXM+DIST(I)
XLOW=HIGH
HIGH=HIGH+5.
XLO=HIG
1411 HIG=HIG+.25
1410 FORMAT(F10.2,F7.2,F8.0,I10,F10.0,F7.0,F8.0,I10)
SIGMAH=SQRTF(FREQ2/DXH-(FRE1/DXH)**2)
SIGMAM=SQRTF(FRED2/DXM-(FRE2/DXM)**2)
BARH=FRE1/DXH
BARM=FRE2/DXM
PRINT 1669,SIGMAH,SIGMAM,BARH,BARM
1669 FORMAT(13H          SIGMA HITF8.3,11H SIGMA MISSF8.3,9H MEAN HITF9.3,10H
1 MEAN MISSF9.3)
IF(K0) 369,369,370
370 HIGH=.002
PRINT 5000
5000 FORMAT (1H1)
PRINT 5002
PRINT 700,CASE,NT

```

```

450 PRINT 450
    FORMAT(48H          FREQUENCY DISTRIBUTION SIGNAL TO NOISE )
    PRINT 451
451 FORMAT(42H          LIMITS                                LIMITS )
    PRINT 452
452 FORMAT(86H LOWER   UPPER   NUMBER   LOWER   UPPER   NUMBER
1     LOWER   UPPER   NUMBER )
    XLOW=.0
    XLOWR=.102
    XLO=.204
    XHIG=.206
    HIGHR=.104
    DO 453 I=1,51
    PRINT 454,XLOW,HIGH,NSIGNO(I),XLOWR,HIGHR,NSIGNO(I+51),XLO,XHIG,
    INSIGNO(I+101)
454 FORMAT(F6.3,F9.3,I8,F13.3,F9.3,I8,F13.3,F9.3,I8)
    XLO=XHIG
    XHIG=XHIG+.002
888 FORMAT(F6.1,F9.3,I8,F13.3,F9.3,I8)
    XLOW=HIGH
    HIGH=HIGH+.002
    XLOWR=HIGHR
453 HIGHR=HIGHR+.002
369 CONTINUE
465 FORMAT(17H DURING THIS RAIDI4,53H AZIMUTH UNRESOLVE SECTORS WERE F
XOUND WHICH CONTAINEDI5,8H TARGETS)
    IF(K1) 460,460,461
461 PRINT 5000
    PRINT 700,CASE,NT
    PRINT 472
    LL1=0
    LL2=0
    LL3=0
    DO 464 I=2,50
    LL1=LL1+LJTHX(I)
    LL2=LL2+LAZDEX(I)
    LL3=LL3+LTOTX(I)
5556 PRINT 462,I,LJTHX(I),LAZDEX(I),LTOTX(I),I
464 PRINT 5557,LL1,LL2,LL3
472 FORMAT(57H NO TARGETS   AZ UNK   AZ+DEL   AZ+DEL+OTHERS   NO TARGET
IS )
462 FORMAT(I12,2I9,I15,I9)
460 CONTINUE
    PAUSE 1
    GO TO 1500
907 FORMAT(I5,F10.5,F5.0,2I8,5F8.2)
904 FORMAT(I5,F10.5,F5.0,5I8)
    FND(0,1,0,0,1)

```

This page was left intentionally blank.

DIMENSION STATEMENTS

NTTOAA (100), DISTT (100), NMST (100), X (100)
NMSA (100), DISMS (100), NHTT (100), Y (100)
NHTA (100), DISHT (100), NFST (100), M (100)
NFSA (100), DISFS (100), DIFX (100), RO (100)
DIFY (100), DELX (100), DELY (100), R1 (100)
R2 (100), AZ (100), NRESOL (100) NSIGNO (100)
LJTH (50), LAZ (50), XAP (100), YAP (100)
LAZDEL (50), LTOT (50), LJTHX (50), LTOTX (50)
LAZDEX (50), DIST (50), DISH (50)

1500

START

READ 4
X (I) Y (I) DELX (I)
DELY (I) I = 1, NT

PRINT 5002
PRINT 700, CASE, NT
PRINT 1100, CLOCKS
CLOCKL, AZMUL,
DELMUL
PRINT 1101 RMAX
RMIN, RM12, AZRES
DELRES
PRINT 701

R
A

CONV = 57.29577
XO = 500.
YO = 500.
GAIN = .001
X1 = 456.6985
X2 = 475.
Y1 = 475.
Y2 = 543.3015
KK = 0 KK2 = 0
KK3 = 0 KK4 = 0
KK5 = 0 KK6 = 0
KK7 = 0 KK8 = 0
AK9 = 0 AK1 = 0

CLOCK=
CLOCKS

IS
CLOCKS
0

YES

A.

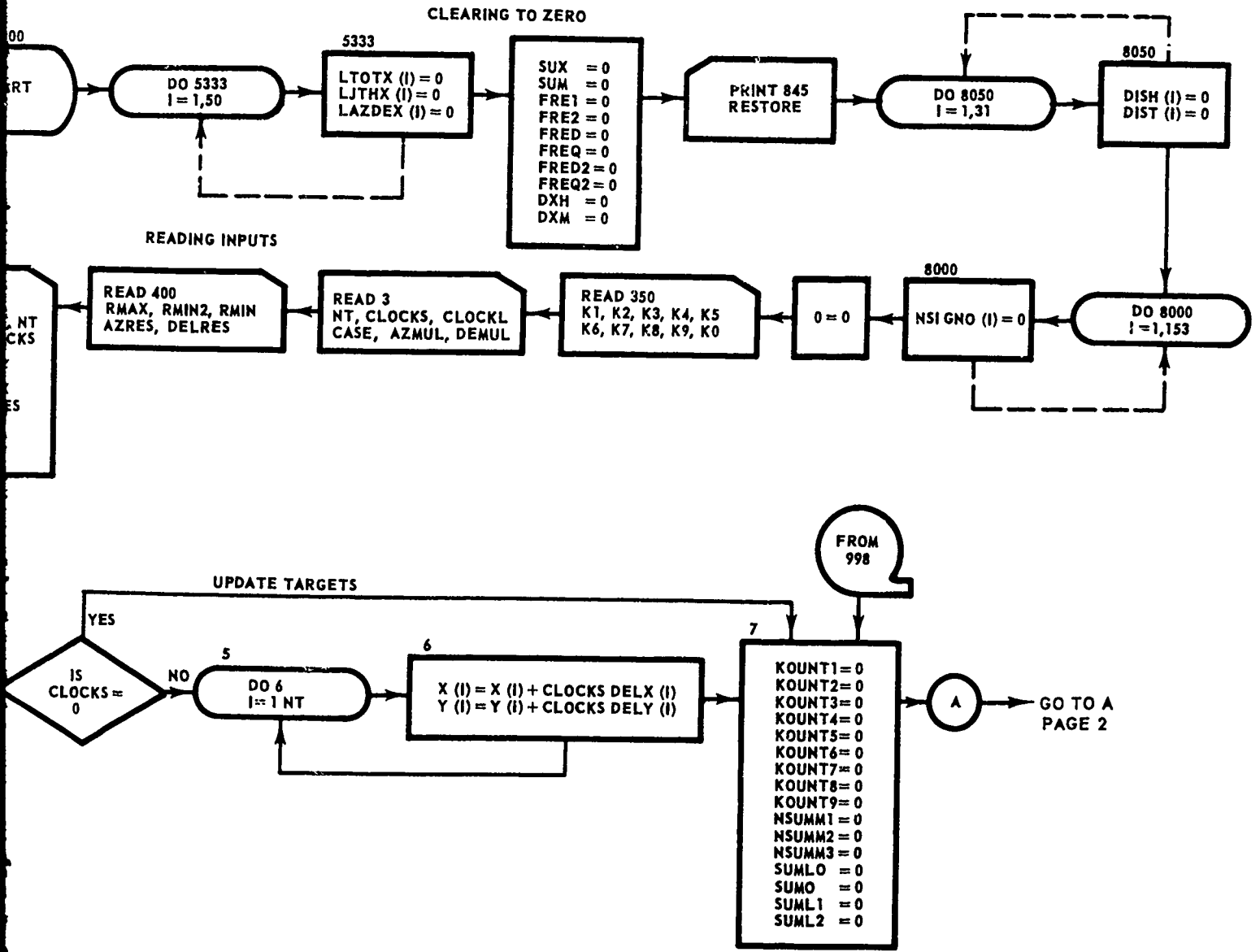
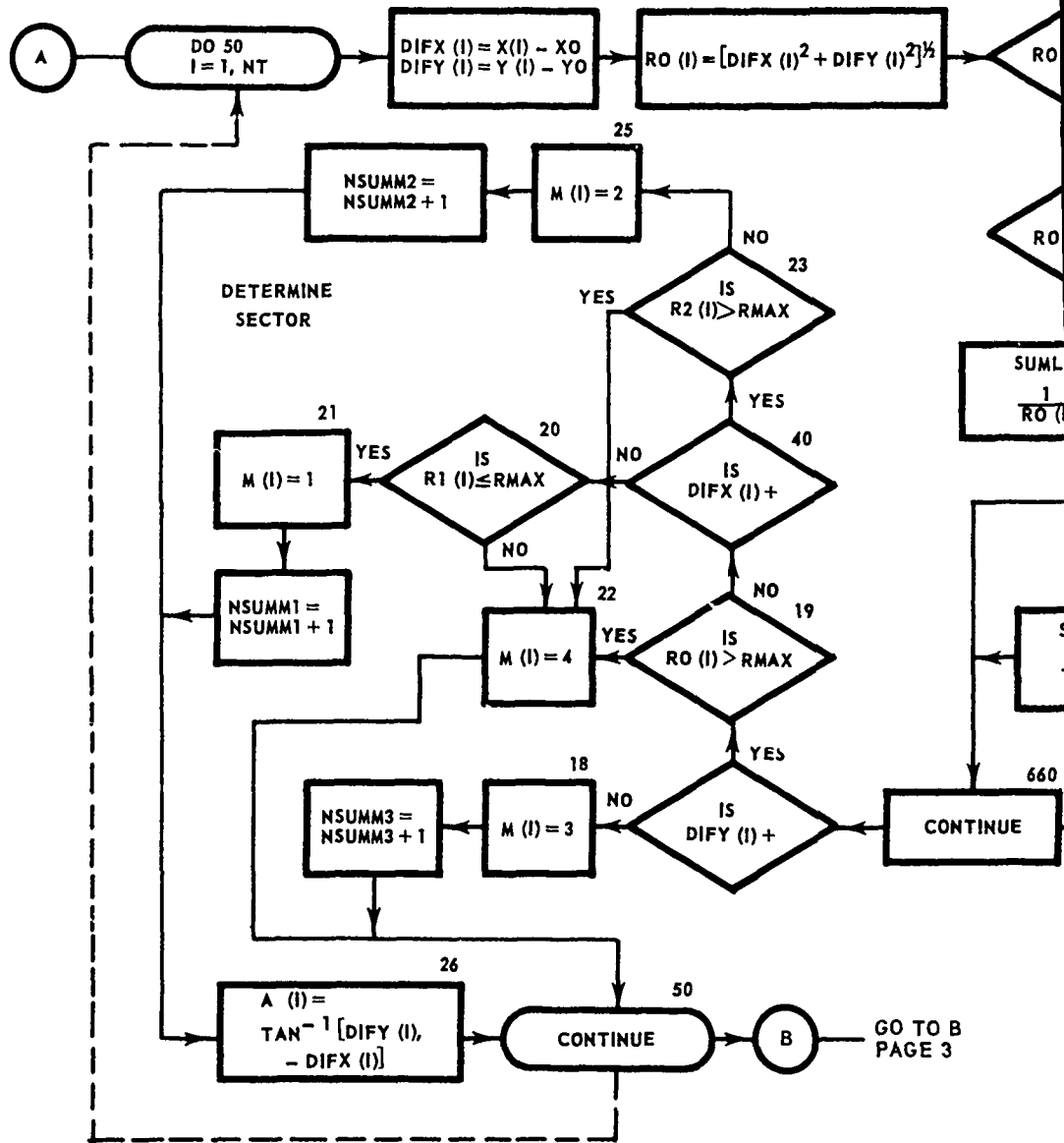


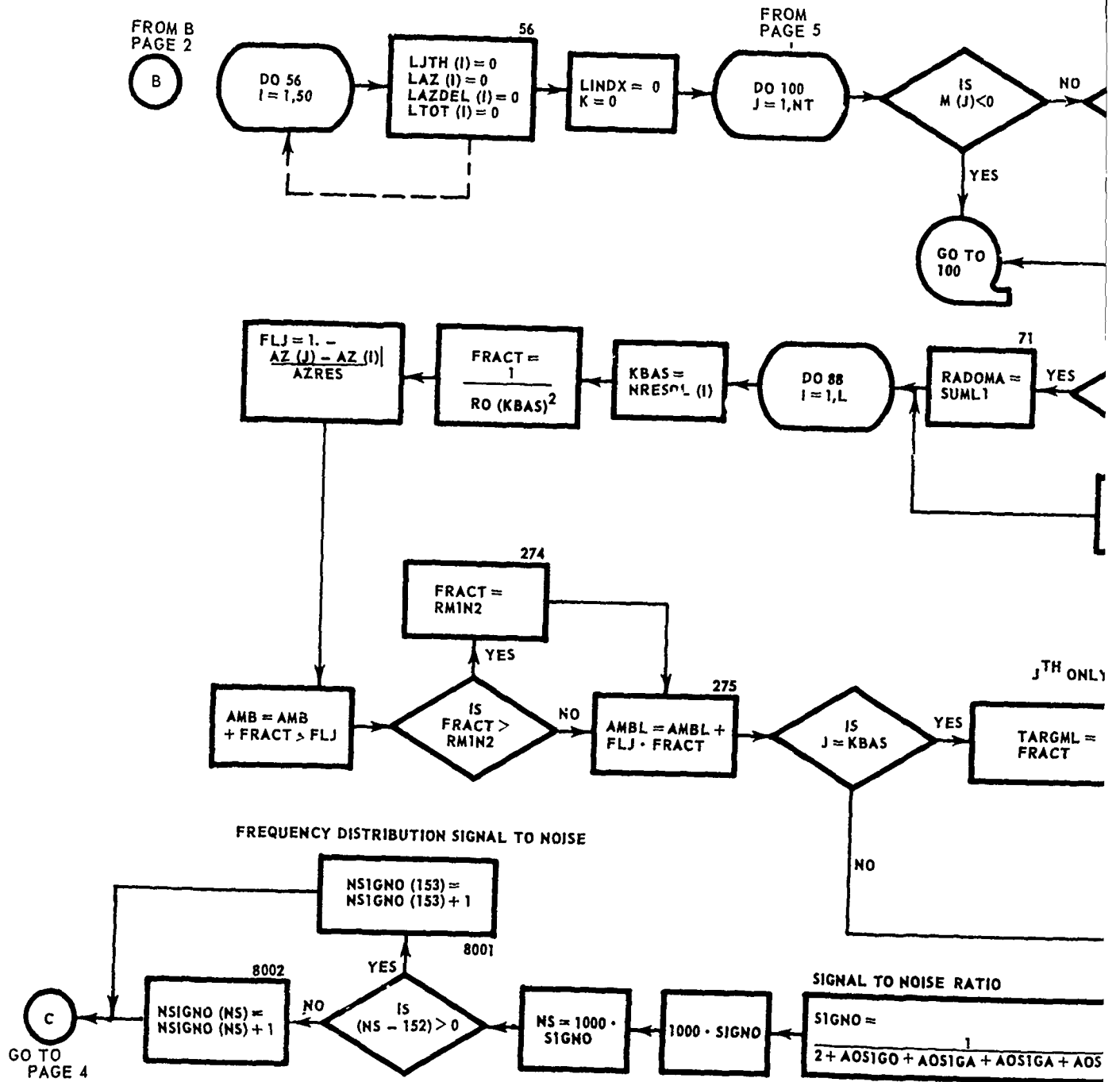
Figure 2-1 Original Program Flow Diagram

B.

FROM A
PAGE 1



A.



A₁

TESTING AZIMUTH RESOLUTION

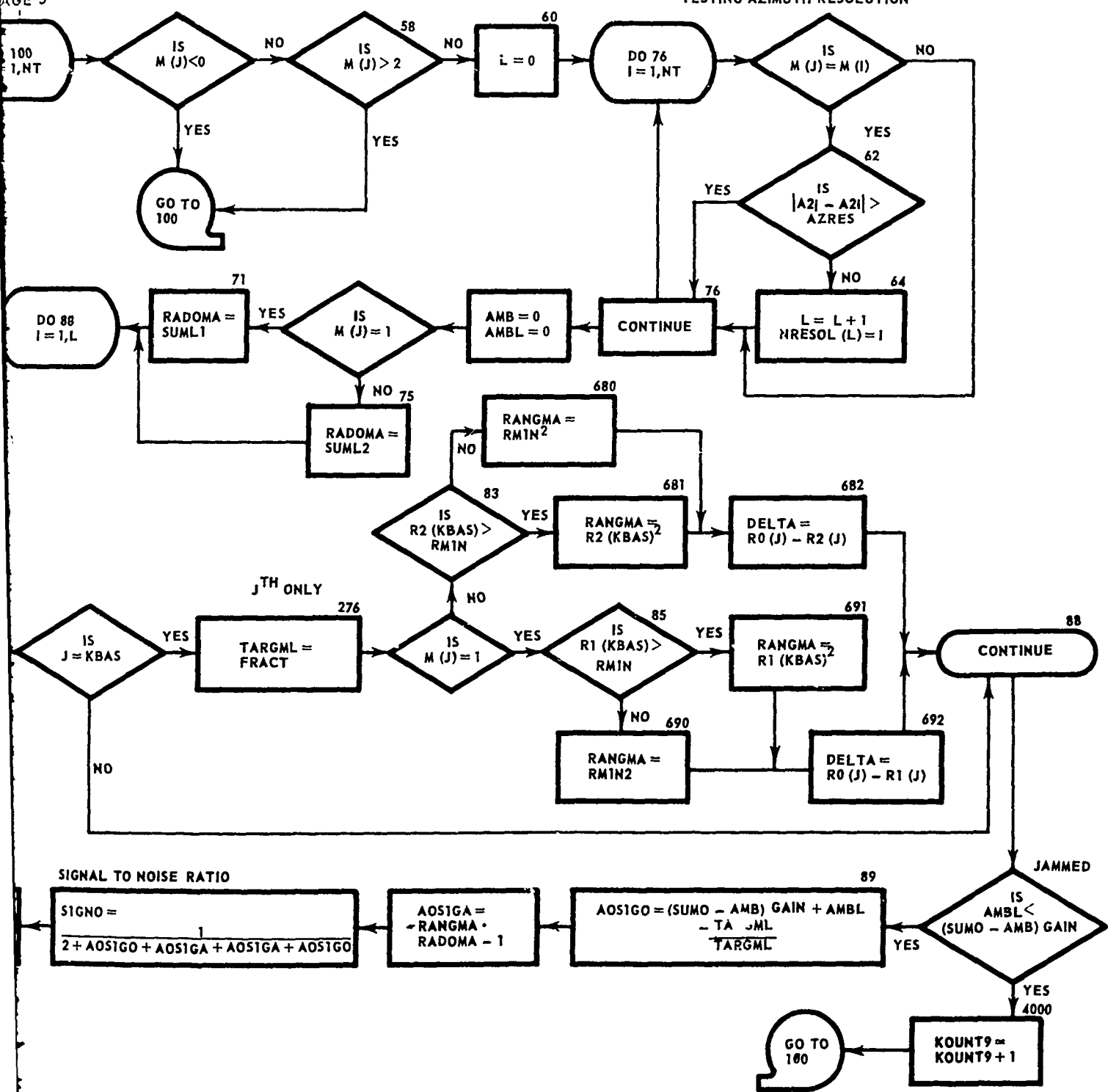
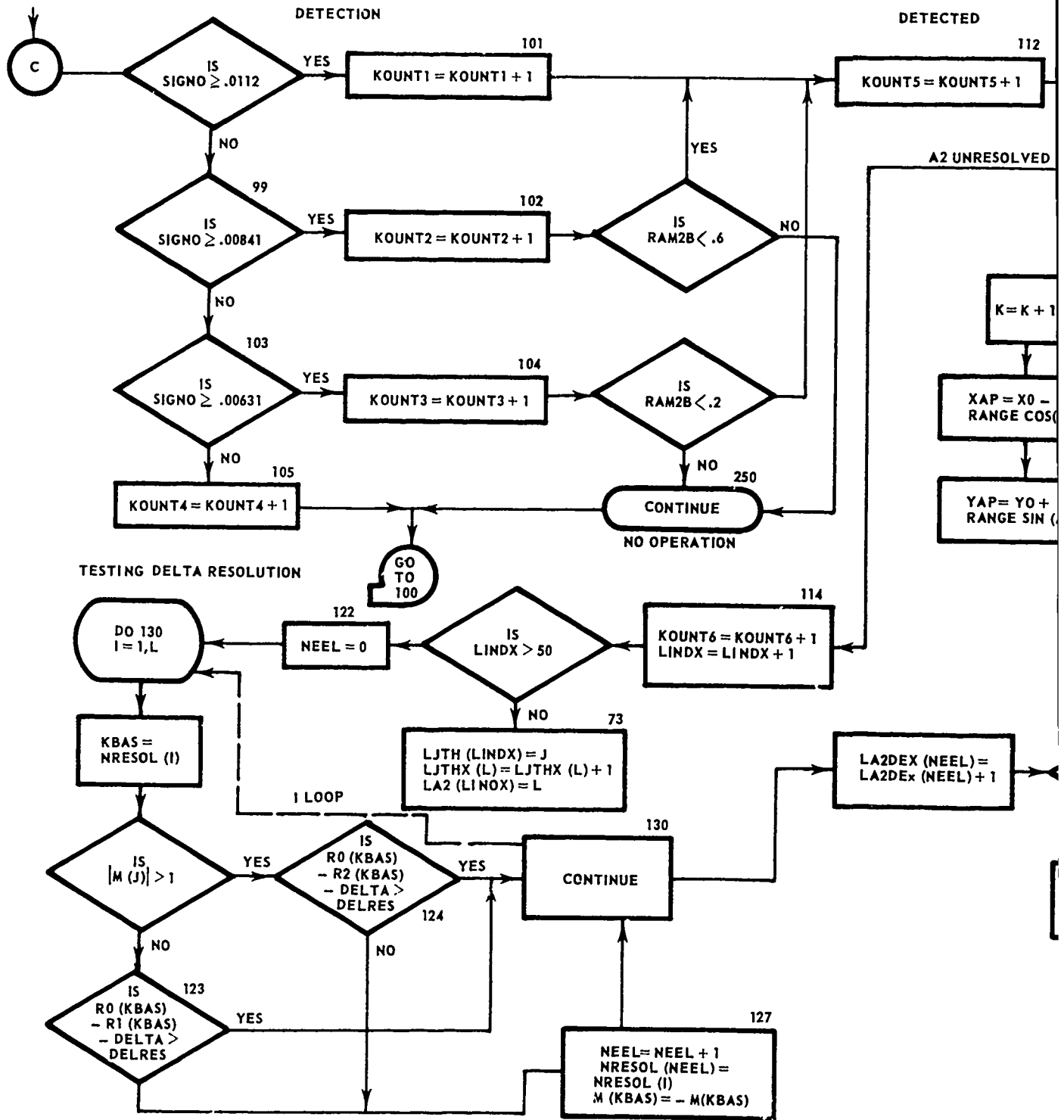


Figure 2-1 (Cont.)

E.

FROM PAGE 3



A.

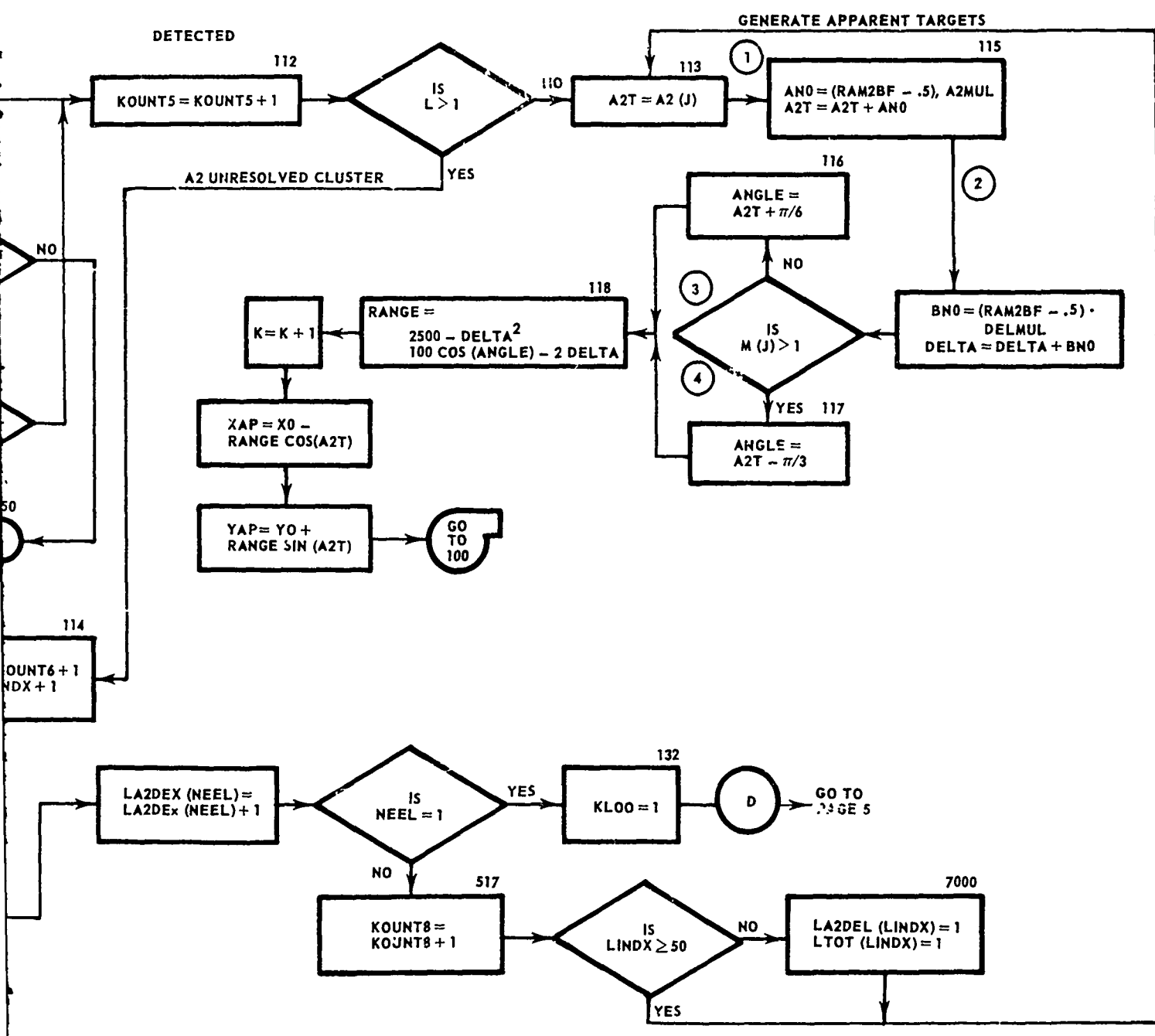
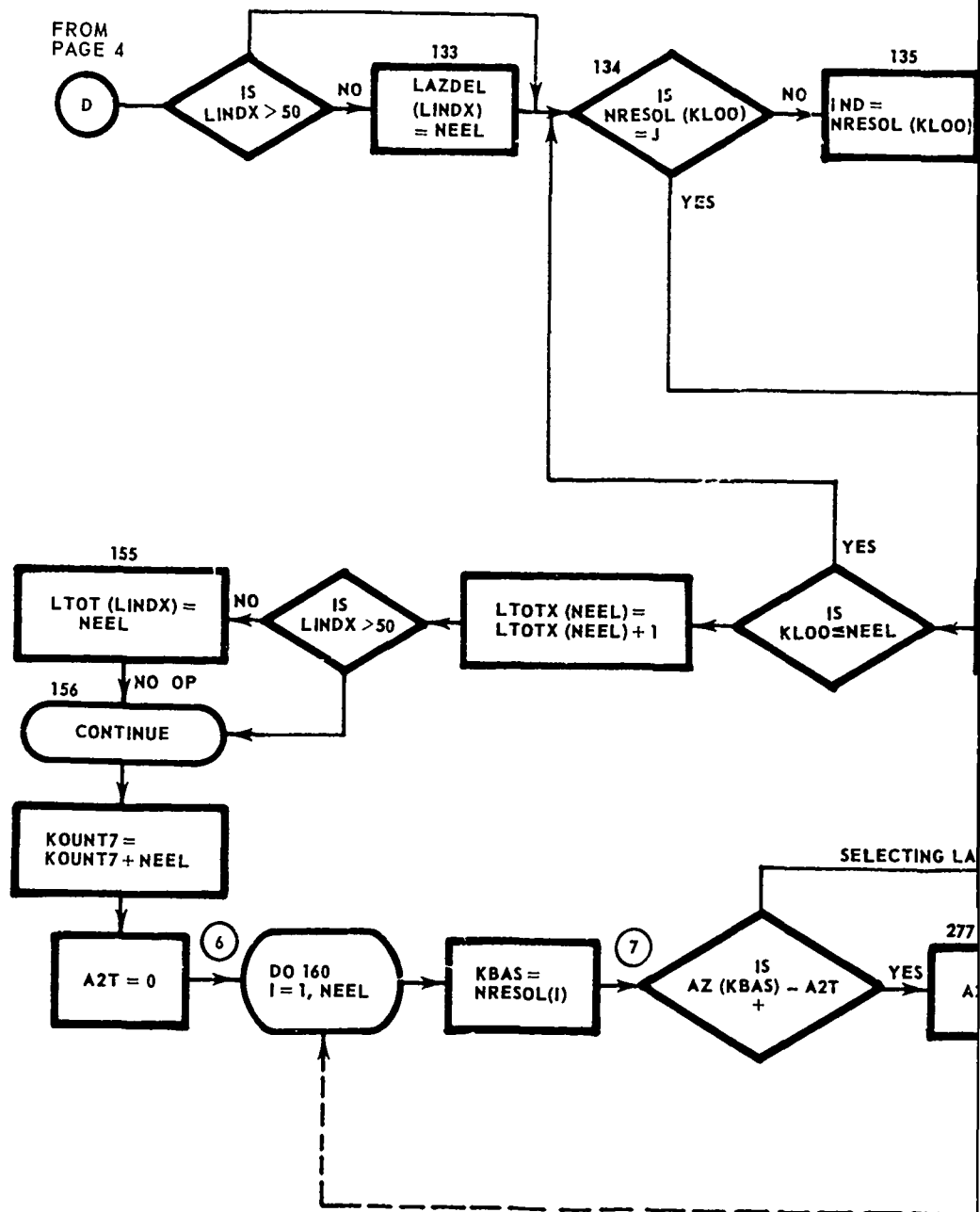


Figure 2-1 (Cont.)

B.

FROM PAGE 4



A.

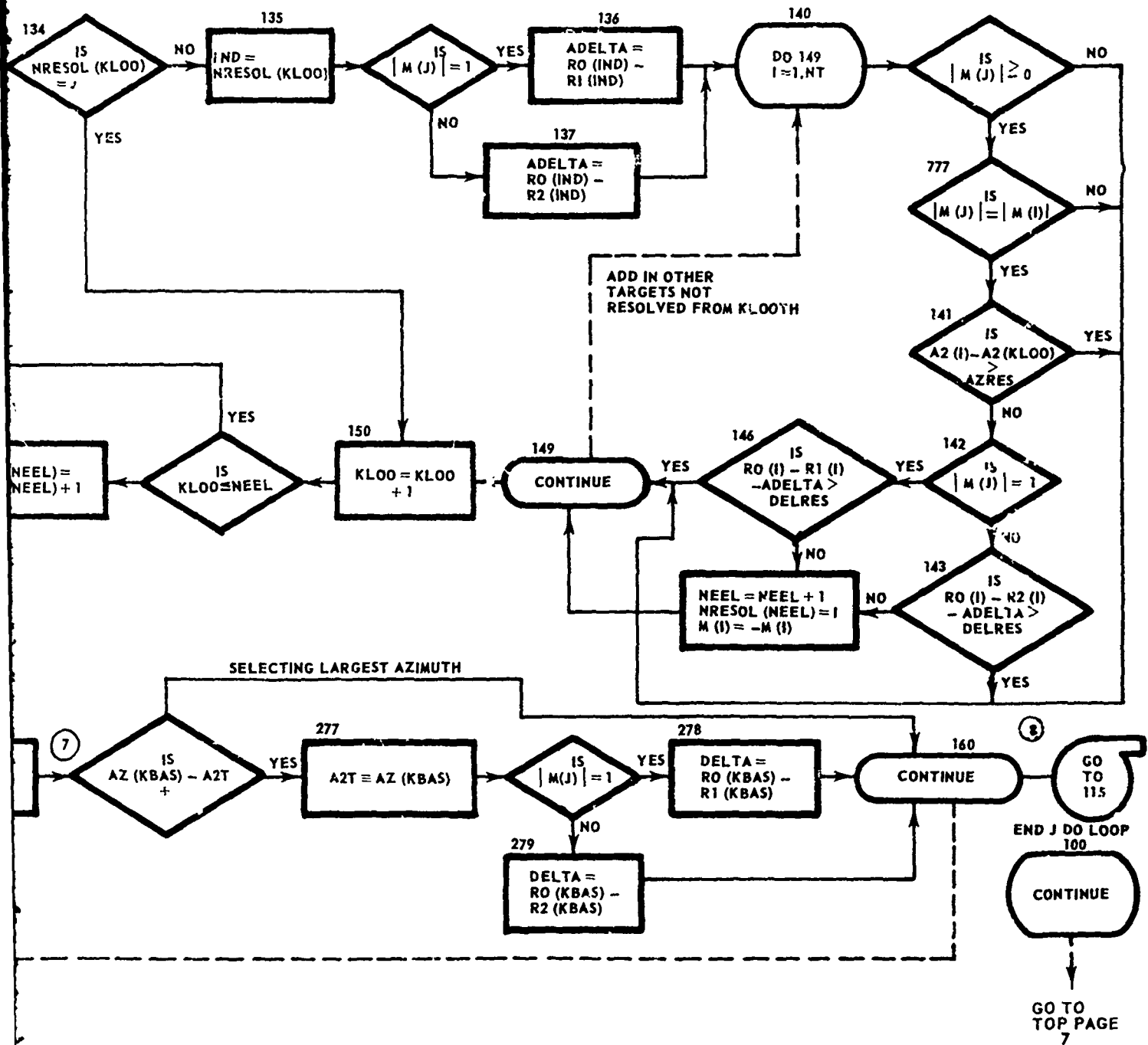
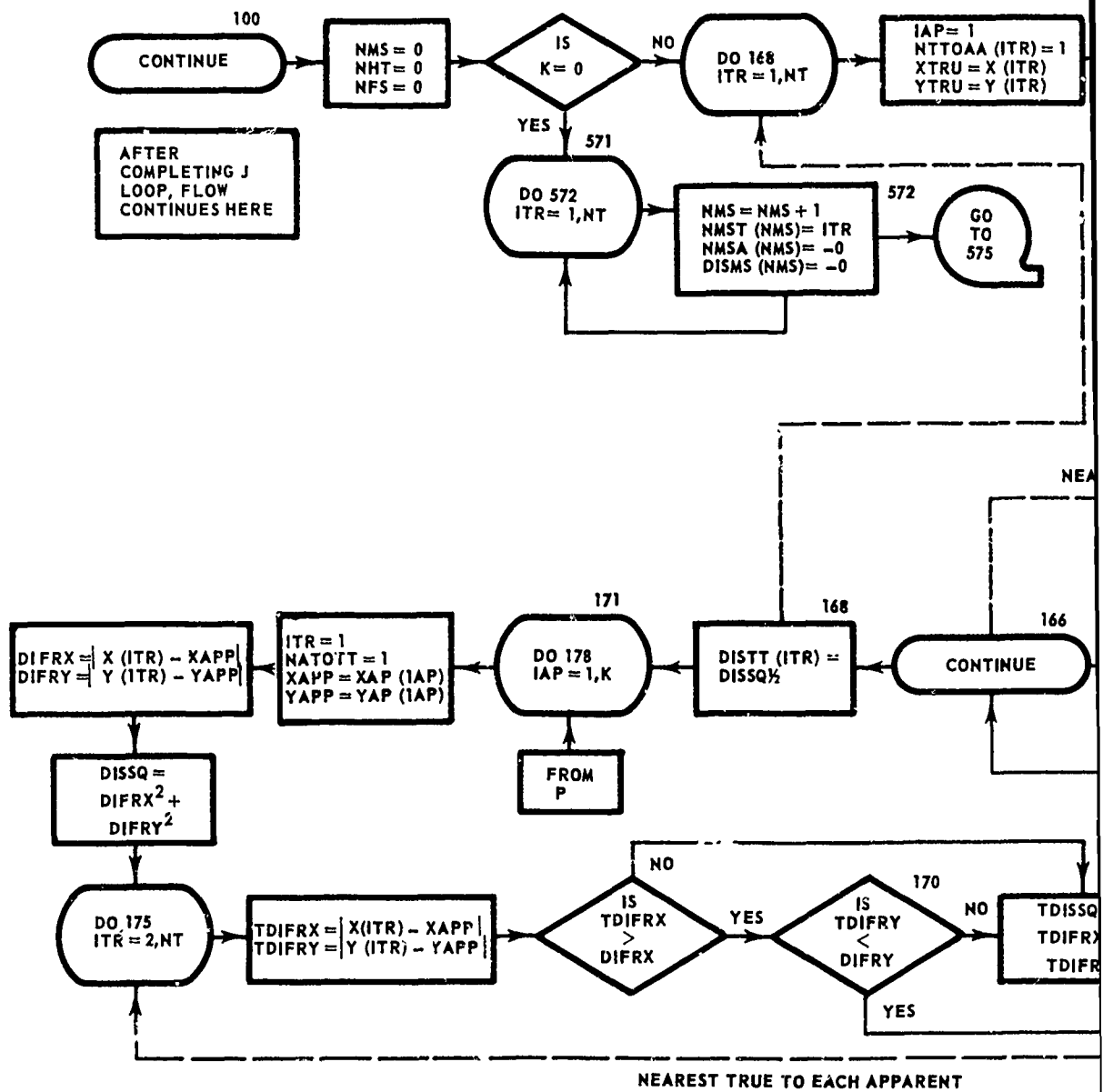
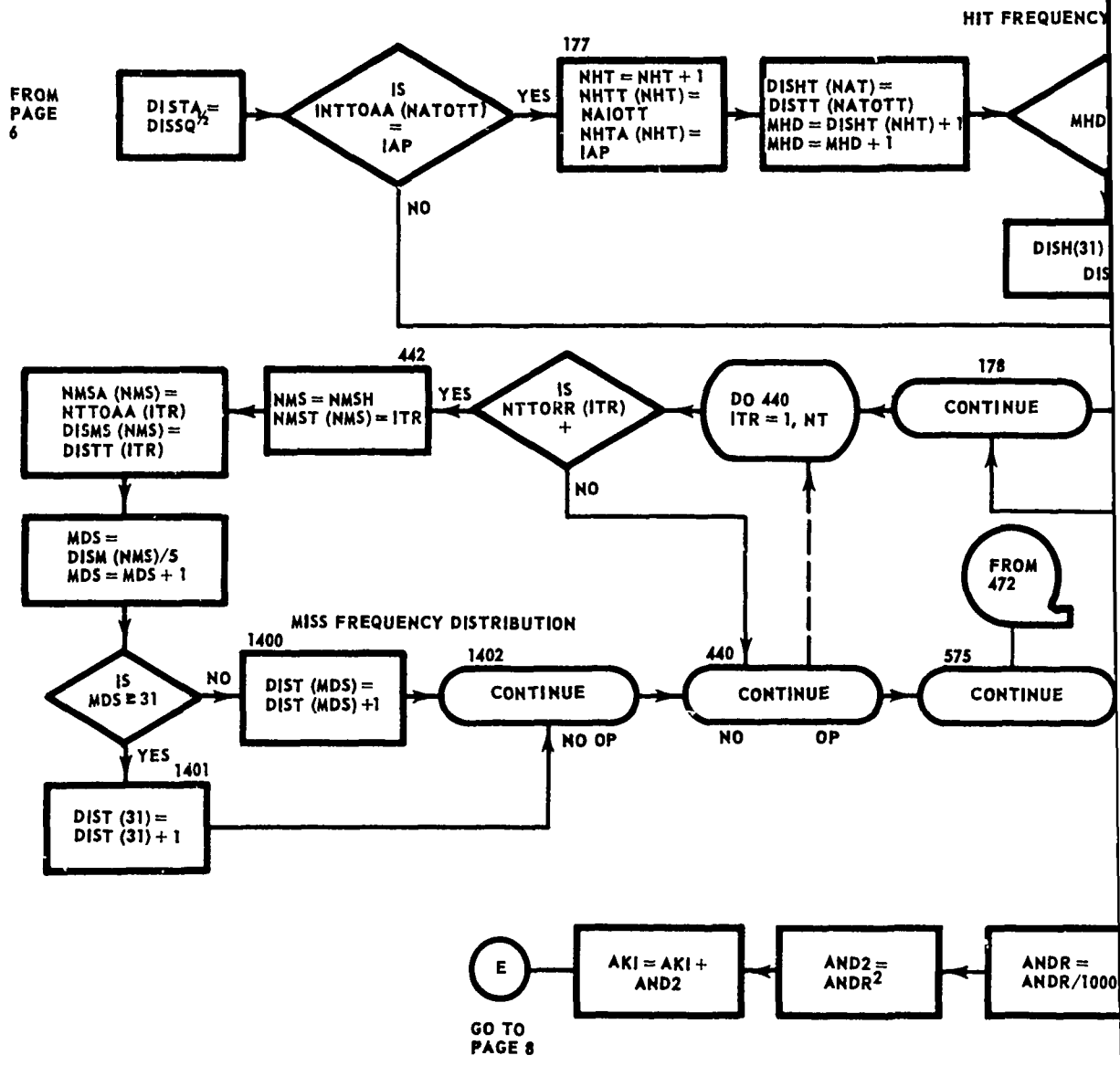


Figure 2-1 (Cont.)

B.



A.



A.

HIT FREQUENCY DISTRIBUTION

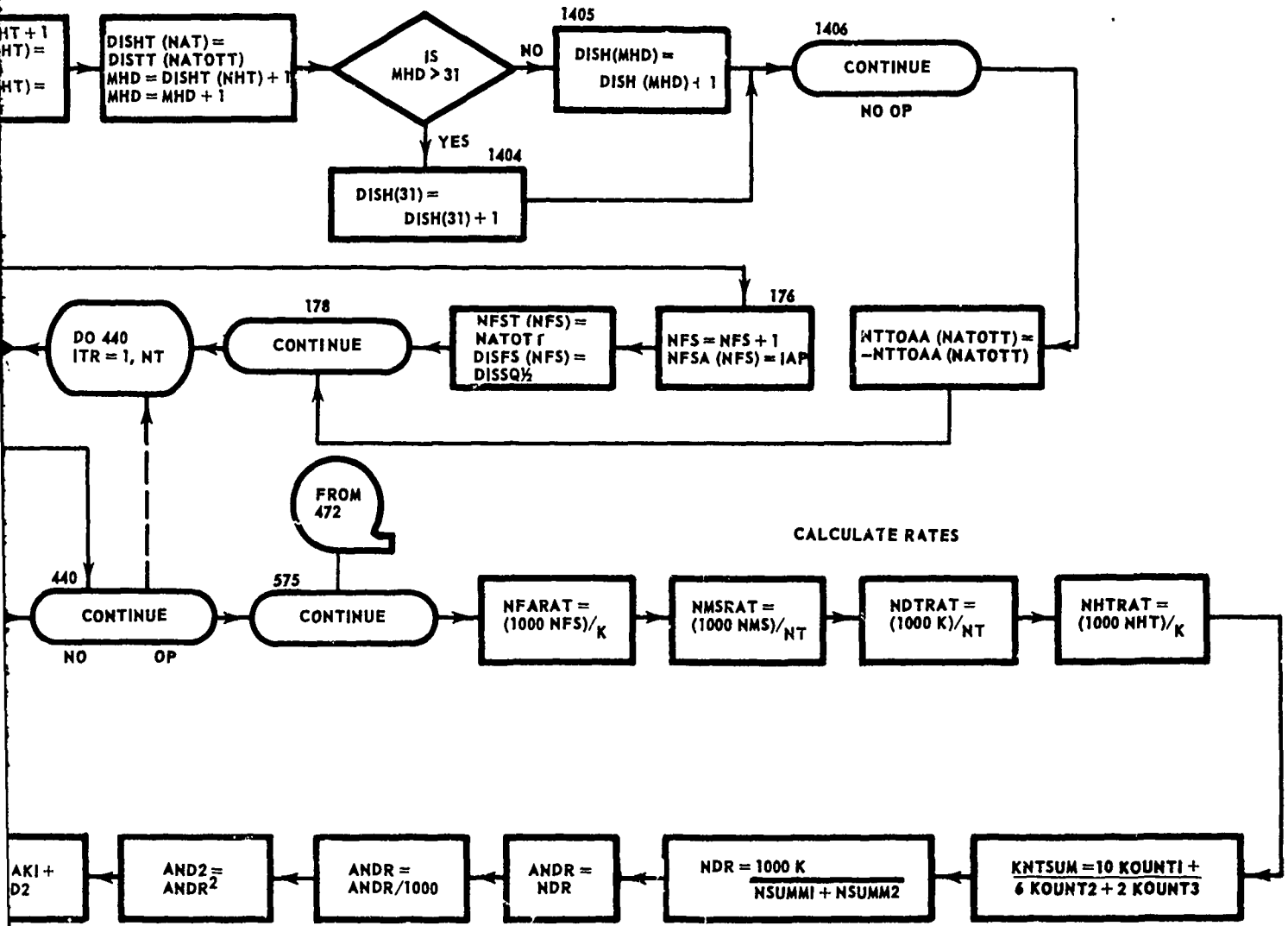


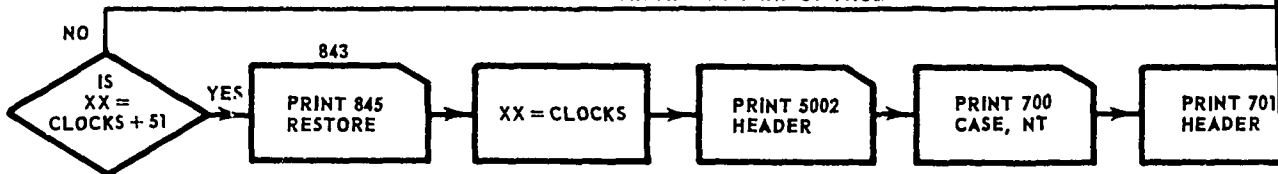
Figure 2-1 (Cont.)

B.

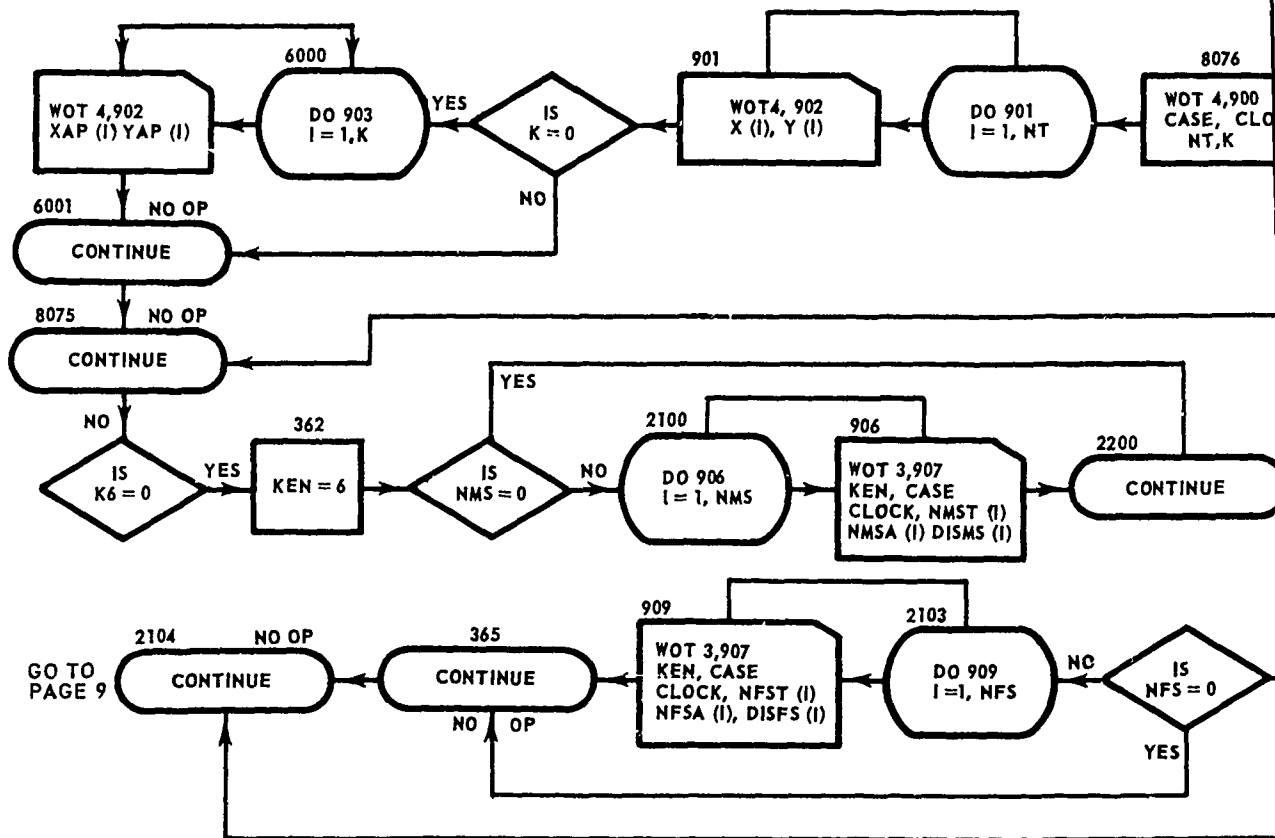
FROM PAGE 7

E

TESTING FOR END OF PAGE



WRITING SPECIAL TAPES



A

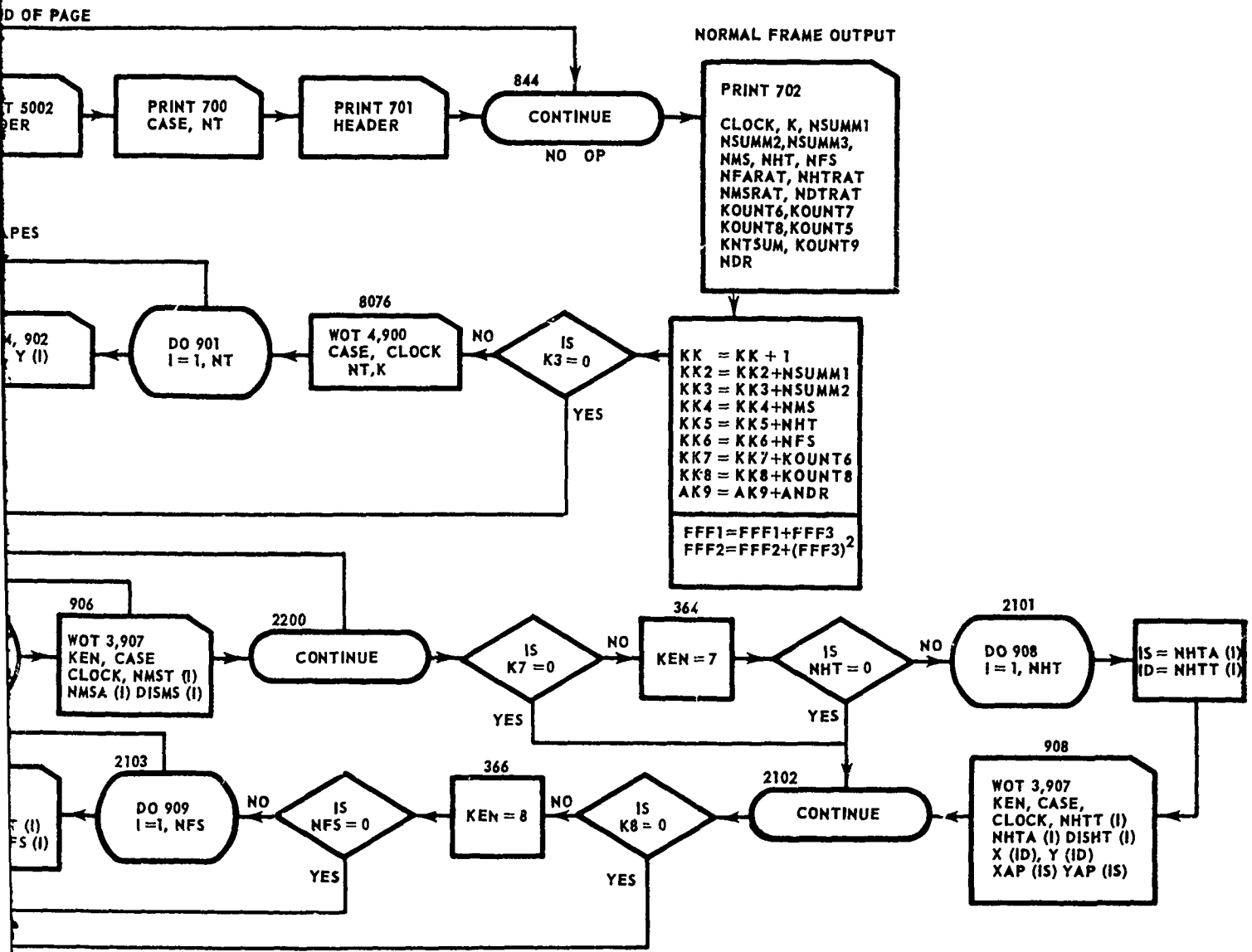
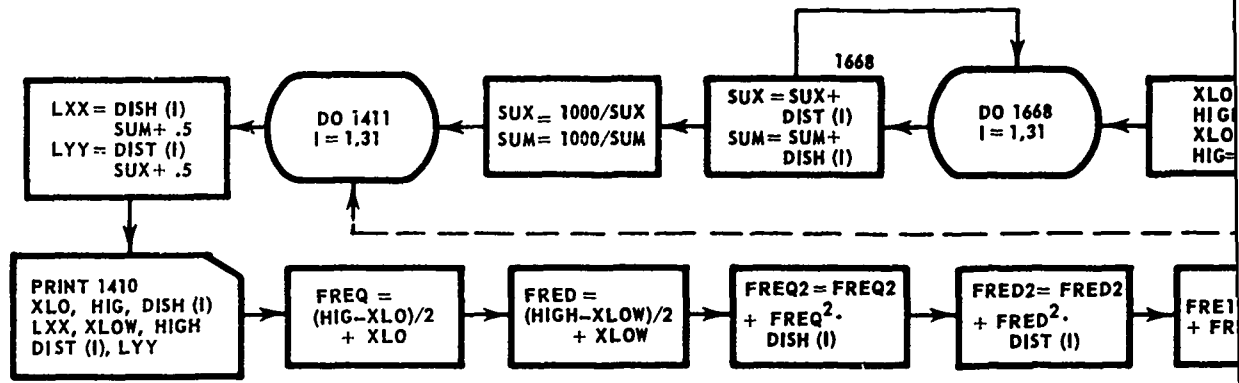
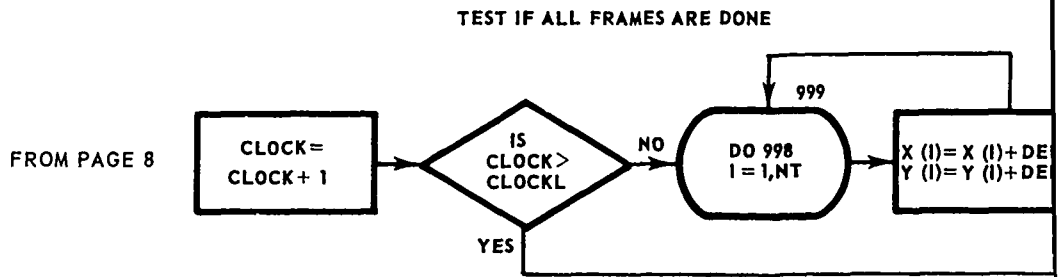


Figure 2-1 (Cont.)

B.



GO TO PAGE 10

A.

FRAMES ARE DONE

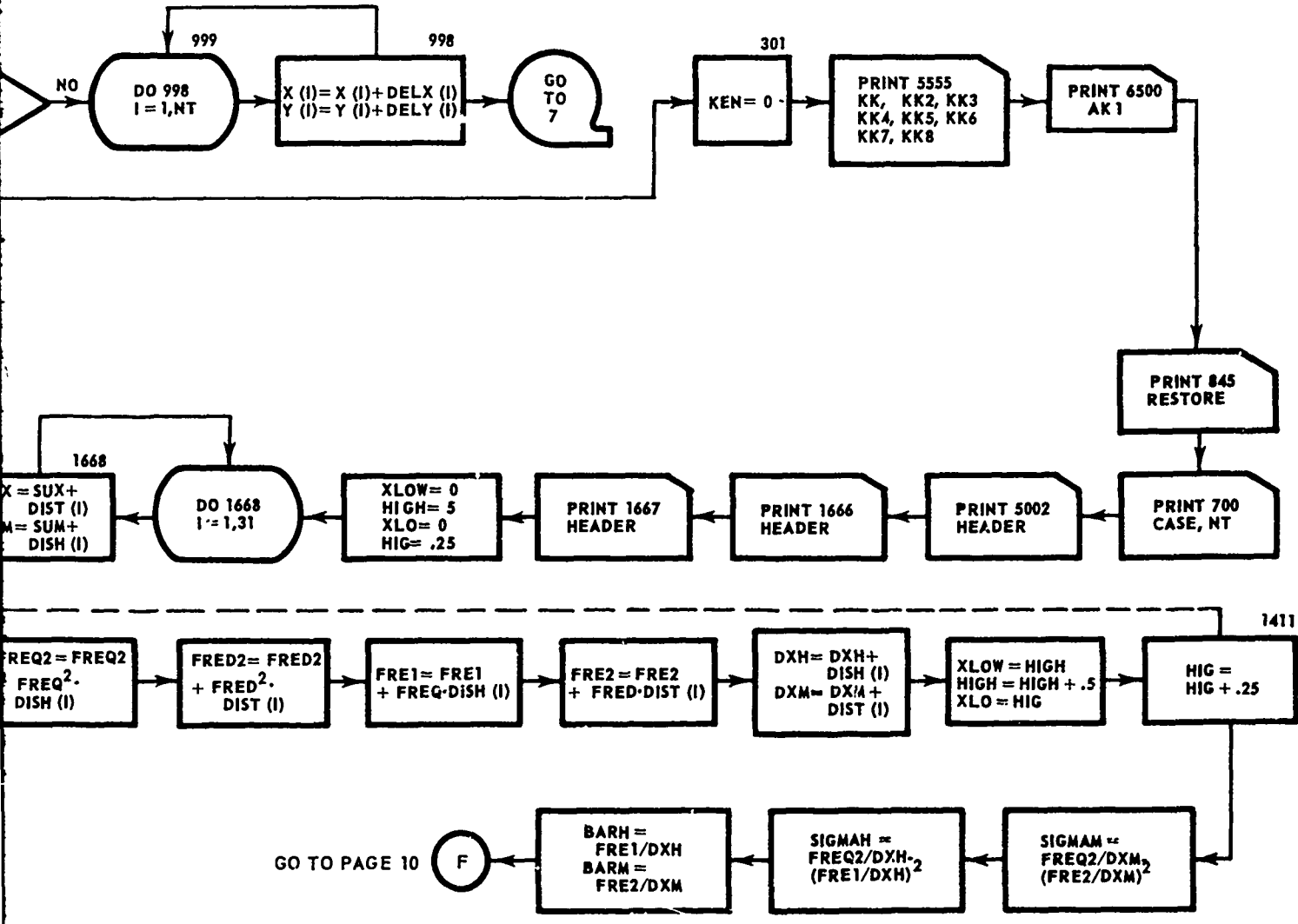
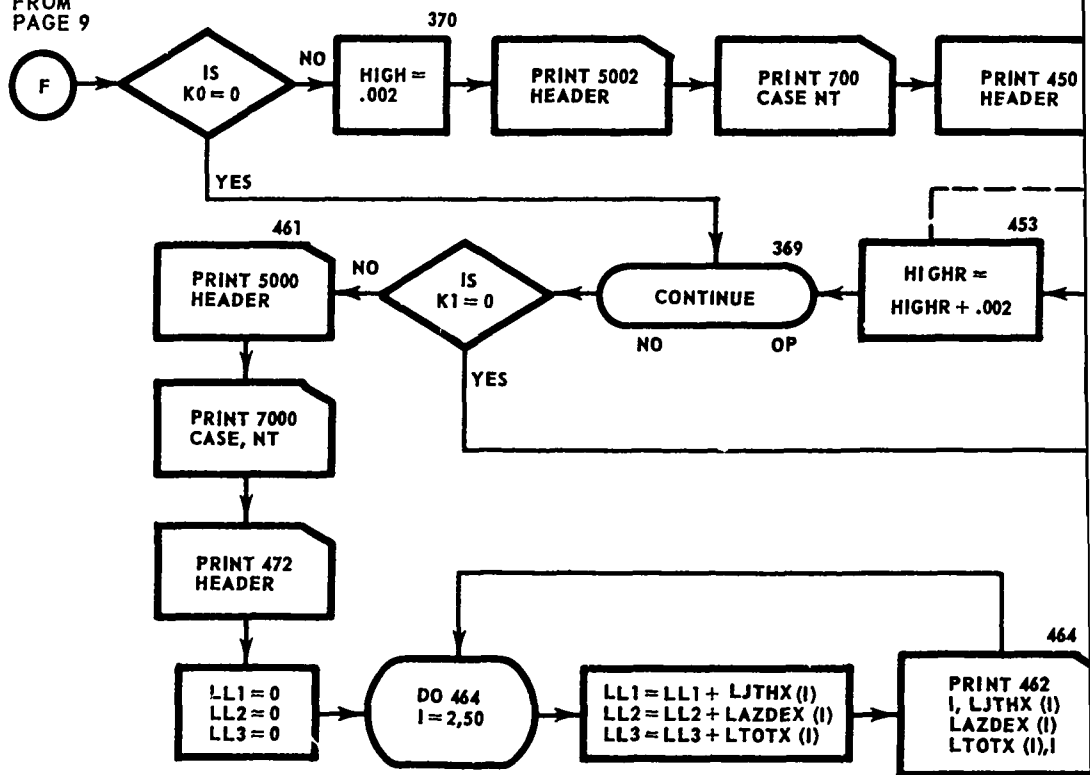


Figure 2-1 (Cont.)

B.

FROM
PAGE 9



Fig

A.

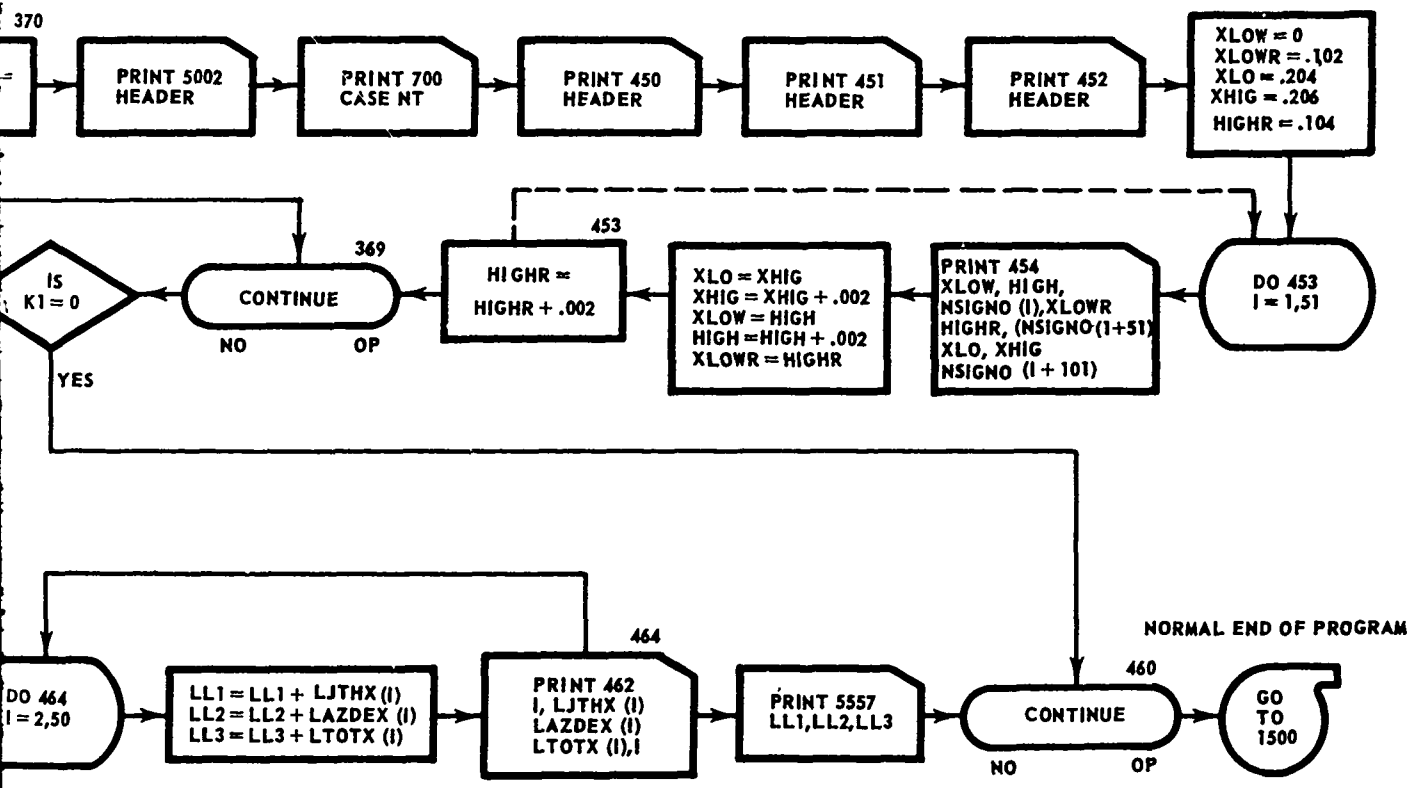


Figure 2-1 (Cont.)

B.

2.2 ENVIRONMENT GENERATOR MODIFICATION FOR MANEUVERING TARGETS

2.2.1 Glossary

<u>ALPHA 1</u>	The target heading measured from due East counter clockwise to the tail of the target direction line (Northbound = 180° , Southbound = 0° , etc.)
<u>ALT</u>	Altitude of target
<u>ANGLE</u>	The angle used to calculate the new target position after the target has turned through θ degrees
<u>BETA</u>	Angle of bank for a specific target turn
<u>CHORD</u>	The length of the straight line connecting two points of the flight path of a specific target
<u>CID</u>	Curve identification. CID = 0 implies a turn to the right, CID = 1 implies a left turn
<u>CLOCK</u>	Identification number (outputted) for each set of target positions
<u>G</u>	Force acting on a target during turn due to gravity (commonly referred to as g-loading)
<u>HEAD</u>	Flight direction of target recorded as the angle measured in degrees from due North (compass North), clockwise to the front of the direction line of the target. (North bound = 0° , Southbound = 180° , etc.)
<u>I</u>	Index for frame loop
<u>ITAG</u>	Target flight plan control. This parameter keeps track of where each target is with respect to its flight plan
<u>J</u>	Target index

M A counter used to permit the program to run 30 frames before commencing a test for out of range boundaries on the targets

N Subscript for flight leg identification

NF Number of frames. When NF is entered as zero the program runs until all the targets are out of radar range. (Since NF is a "DO LOOP Limit" the program must transfer around the respective LOOP when NF = 0.) When NF > 0 the program stops after NF frames (radar revolutions)

NT Number of targets; NT at present is set for a maximum of 60 targets

NTOR Number of targets out of radar range

RATETN Angular rate of turn in radians for a specific target and a specific turn

SEC Number of seconds target is to fly in a specific turn

SECST Number of seconds target is to fly in a specific straight course

THETA The angle a target must turn to complete a frame

THETAL The angle remaining to be turned through to complete a specific curved course

V Target velocity

X X coordinate target position

XLAST X coordinate target position

XPART The X distance remaining to travel in order to finish either a straight line course, or a curved course

Y Y coordinate target position

YLAST Y coordinate target position

YPART The Y distance remaining to travel in order to finish either a straight line course or a curved course

2.2.2 Environment Generator FORTRAN Listing

```

DIMENSION V(60),X(60),Y(60),ALT(60),SECST(5,60),SEC(5,60),HEAD(5,6) 002
X0),G(5,60),CID(5,60),ITAG(60), XLAST(60),YLAST 003
X(60), BETA(5,60), RMAX(60),RMIN(60),RMIN2(60), RATEFN(5,60) 004
DIMENSION THETA(60), CHORD(60), ANGLE(60), ALPHA1(60), THETA1(60) 005
EQUIVALENCE(BETA,G) 006
EQUIVALENCE(ALPHA1,HEAD) 007
READ 10,NT,NF,ID 008
WRITE OUTPUT TAPE 2,10,NT,NF,ID 009
10 FORMAT(3I5) 010
DO 14 J=1,NT 011
READ 12, (X(J), Y(J), ALT(J), V(J)) 012
12 FORMAT (4F14.4) 013
READ 16, SECST(1,J), HEAD(1,J), G(1,J), SECST( 2 ,J), HEAD( 2 ,J), 014
X( 2 ,J), CID(1,J), CID( 2 ,J) 015
READ 16,SECST(3,J),HEAD(3,J),G(3,J),SECST(4,J),HEAD(4,J),G(4,J), 016
CID(3,J),CID(4,J),SECST(5,J),HEAD(5,J) 017
13 FORMAT(2F9.3,F7.3 ,2F9.3,F7.3,2F2.0,2F9.3) 018
IF (SENSE SWITCH1)300,14 019
300 PRINT302 020
302 FORMAT(55H0DATA LISTED BELOW IS INPUTS ARRANGED AS IN INPUTFORMAT) 021
PRINT303,X(J),Y(J),ALT(J),V(J) 022
PRINT304,SECST(1,J),HEAD(1,J),G(1,J),SECST(2,J),HEAD(2,J),G(2,J) 023
1 CID(1,J),CID(2,J) 024
PRINT304,SECST(3,J),HEAD(3,J),G(3,J),SECST(4,J),HEAD(4,J),G(4,J), 025
1 CID(3,J),CID(4,J),SECST(5,J),HEAD(5,J) 026
303 FORMAT( 4F16.6) 027
304 FORMAT(2F9.3,F7.3,2F9.3,F7.3,2F2.0,2F9.3) 028
14 CONTINUE 029
N=1 030
DO 20 J=1,NT 031
HEAD(N,J)=270.-HEAD(N,J) 032
ITAG(J)=1 033
RMAX(J)= 1.23*SQRTF(ALT(J)) 034
RMIN(J)= -340.906 +SQRTF((340.906)**2+(9200./6080.)*ALT(J)+(ALT(J) 035
X/6080.))**2) 036
RMIN2(J)=1./ (RMIN(J))**2 037
V(J)=V(J)/3600. 038
XLAST(J) = X(J) 039
YLAST(J) = Y(J) 040
IF(SENSE SWITCH1)132,130 041
132 PRINT 134,NT,RMAX(J),RMIN(J),RMIN2(J) 042
134 FORMAT(5H NT=I5,6H RMAX=F16.8,6H RMIN=F16.8,7H RMIN2=F16.8) 043
130 WRITE OUTPUT TAPE 2,22,RMAX(J),RMIN(J),RMIN2(J) 044
22 FORMAT(3E16.8) 045
20 CONTINUE 046
CALCULATE NUMBER OF SECONDS IN THE TARGET TURNS PLUS RATE OF TURN 047

```

136	IF(SENSE SWITCH 1) 520,106	048
520	PRINT 92	049
92	FORMAT(30H BETA RATETN SEC)	050
106	DO 26 N=1,4	051
	DO 26 J=1,NT	052
	BETA(N,J)= ATANF(SQRTF((G(N,J)**2-1.)))	053
25	RATETN(N,J)=(SINF(BETA(N,J)))/(188.82*V(J)*COSF(BETA(N,J)))	054
	HEAD(N+1,J)=270.-HEAD(N+1,J)	055
	IF(SENSE SWIT H3)400,401	056
400	PRINT 402, HEAD(N,J)	057
402	FORMAT(7H HEAD = F10.3)	058
401	IF(CID(N,J))200,202,200	059
202	IF(HEAD(N+1,J)-HEAD(N,J))204,206,206	060
206	SEC(N,J)=(6.2832-(HEAD(N+1,J)-HEAD(N,J))*0.0174533)/RATETN(N,J)	061
	GO TO 208	062
204	SEC(N,J)=((HEAD(N,J)-HEAD(N+1,J))*0.0174533)/RATETN(N,J)	063
	GO TO 208	064
200	IF(HEAD(N+1,J)-HEAD(N,J))210,210,212	065
210	SEC(N,J)=(6.2832-(HEAD(N,J)-HEAD(N+1,J))*0.0174533)/RATETN(N,J)	066
	GO TO 208	067
212	SEC(N,J)=(HEAD(N+1,J)-HEAD(N,J))*0.0174533 /RATETN(N,J)	068
208	IF(SENSE SWITCH 1)86,26	069
86	PRINT '88,BETA(N,J),RATETN(N,J),SEC(N,J)	070
88	FORMAT(3F10.3)	071
26	CONTINUE	072
	N=1	073
	CLOCK =0.	074
	WRITE OUTPUT TAPE 2,501,CLOCK	075
501	FORMAT(F6.0)	076
	DO 120 J=1,NT	077
	ALPHA1(J)=HEAD(N,J)*0.0174533	078
120	WRITE OUTPUT TAPE 2,38,X(J),Y(J)	079
	NTOR=0	080
	M=0	081
	DO 90 I=1,NF	082
505	CLOCK=CLOCK+1.	083
	WRITE OUTPUT TAPE 2,501,CLOCK	084
	DO 80 J=1,NT	085
	N = ITAG(J)	086
	IF (SECST(N,J)-15.)34,36,36	087
36	SECST(N,J)=SECST(N,J)-15.	088
	X(J)=XLAST(J)-15.*V(J)*COSF(ALPHA1(J))	089
	Y(J)=YLAST(J)-15.*V(J)*SINF(ALPHA1(J))	090
	XLAST(J)=X(J)	091
	YLAST(J)=Y(J)	092
	IF(SENSE SWITCH2)94,76	093
94	PRINT 96,J,N,SECST(N,J),X(J),Y(J)	094
96	FORMAT(8H0 TARGET I2,10HIS ON THE I2,21HST. LINE LEG. IT HAS F8.3,1	095
	X6HSEC LEFT. XCOOR=F8.3,7H YCOOR=F8.3)	096

	GO TO 76	097
34	IF (SECST(N,J)) 40,42,40	098
40	XPART=SECST(N,J)*V(J)*COSF(ALPHA1(J))	099
	YPART=SECST(N,J)* V(J)*SINF(ALPHA1(J))	100
42	IF (ITAG(J)-5) 44,46,44	101
46	PRINT 48,J	102
48	FORMAT(11H TARGET NO I2,32H IS TRYING TO ENTER A 5TH CURVE.)	103
	PAUSE 7	104
44	IF (SEC(N,J)-15.+SECST(N,J)) 50,52,52	105
52	SEC(N,J)= SEC(N,J)-15.+SECST(N,J)	106
	THETA(J) = (15.-SECST(N,J))*RATETN(N,J)	107
	SECST(N,J)=0	108
	XLAST(J) = X (J)-XPART	109
	YLAST(J)=Y(J)-YPART	110
	CHORD(J)=(2.*V(J)*SINF(THETA(J)/2.))/RATETN(N,J)	111
	TARGET TO TURN CLOCKWISE OR COUNTER-CLOCKWISE	112
	IF(CID(N,J)) 54,56,54	113
56	ANGLE(J)=THETA(J)/2.+1.570797-ALPHA1(J)	114
	ALPHA1(J)=ALPHA1(J)-THETA(J)	115
	IF(ALPHA1(J)) 220,218,218	116
220	ALPHA1(J)=6.2832+ALPHA1(J)	117
	GO TO 218	118
54	ANGLE(J)=1.570797-THETA(J)/2.-ALPHA1(J)	119
	ALPHA1(J)=ALPHA1(J)+THETA(J)	120
	IF(ALPHA1(J)-6.2832) 218,218,214	121
214	ALPHA1(J)=ALPHA1(J)-6.2832	122
218	X(J)=XLAST(J)-CHORD(J)*SINF(ANGLE(J))	123
	Y(J)=YLAST(J)-CHORD(J)*COSF(ANGLE(J))	124
	XLAST(J)=X(J)	125
	YLAST(J)=Y(J)	126
	IF (SENSE SWITCH 2) 98,76	127
98	PRINT 100,J,N, SEC(N,J),CID(N,J)	128
100	FORMAT'9H0 TARGET I2,11H HAS ENTERED I2,19H CURVED LEG. IT HAS F8.3,1	129
	XOHSEC. LEFT.,8H CID=F2.0)	130
	PRINT 102,XLAST(J),YLAST(J)	131
102	FORMAT(12H0 LAST POINT=F10.3)	132
	PRINT 104,THETA(J),CHORD(J),ANGLE(J),ALPHA1(J),X(J),Y(J)	133
104	FORMAT(8H THETA=F10.3,7H CHORD=F10.3,7H ANGLE=F10.3 JH ALPHA1=F10	134
	X.3,7H XCOOR=F10.3,7H YCOOR=F10.3)	135
	GO TO 76	136
50	IF (SEC(N,J)) 60,62,60	137
60	THETA1(J)= SEC(N,J)* RATETN(N,J)	138
	IF(CID(N,J)) 64,66,64	139
66	ANGLE(J)=THETA1(J)/2.+1.5707965-ALPHA1(J)	140
	ALPHA1(J)=ALPHA1(J)-THETA1(J)	141
	IF(ALPHA1(J)) 224,68,68	142
224	ALPHA1(J)=6.2832+ALPHA1(J)	143
	GO TO 68	144
64	ANGLE(J)= 1.5707965 - THETA1(J)/2. -ALPHA1(J)	145

ALPHA1(J)=ALPHA1(J)+THETA1(J)	146
IF(ALPHA1(J)-6.2832)68,68,228	147
228 ALPHA1(J)=ALPHA1(J)-6.2832	148
68 CHORD(J)=2.*V(J)*SINF(THETA1(J)/2.)/RATETN(N,J)	149
XPART=XPART+CHORD(J)*SINF(ANGLE(J))	150
YPART=YPART+CHORD(J)*COSF(ANGLE(J))	151
ITAG(J)= ITAG(J)+1	152
XLAST(J)=X(J)-XPART	153
YLAST(J)=Y(J)-YPART	154
N= ITAG(J)	155
SECST(N,J)=SECST(N,J)-15.+SEC(N-1,J)+SECST(N-1,J)	156
GO TO 70	157
62 ITAG(J) = ITAG(J)+1	158
N= ITAG(J)	159
70 X(J)=XLAST(J)-(15.-SEC(N-1,J)-SECST(N-1,J))*V(J)*COSF(ALPHA1(J))	160
Y(J)=YLAST(J)-(15.-SEC(N-1,J)-SECST(N-1,J))*V(J)*SINF(ALPHA1(J))	161
SEC(N-1,J)=0	162
IF(SENSE SWITCH 2)108,76	163
108 PRINT 110,J,N	164
110 FORMAT(8H TARGET I2,15H HAS ENTERED THE I2,13H ST. LINE LEG.)	165
PRINT 112,XPART,YPART,ALPHA1(J),CHORD(J),XLAST(J),YLAST(J)	166
112 FORMAT(8H XPART=F8.3,8H YPART=F8.3,9H ALPHA1=F8.3,7H CHORD=F8.3	167
X,7H XLAST=F8.3,7H YLAST=F8.3)	168
76 XLAST(J)=X(J)	169
YLAST(J)=Y(J)	170
XPART=0	171
YPART=0	172
IF(SENSE SWITCH 1)138,140	173
138 PRINT 310,X(J),Y(J)	174
310 FORMAT(2F16.8)	175
140 WRITE OUTPUT TAPE 2,38,X(J),Y(J)	176
38 FORMAT(2E16.8)	177
IF(M-30)80,508,508	178
508 IF(Y(J)-500.)509,509,510	179
510 IF(SQRTF((X(J)-500.)**2+(Y(J)-500.)**2)-RMAX(J)) 80,509,509	180
509 NTOR=NTOR+1	181
IF(NTOR-NT)80,513,513	182
80 CONTINUE	183
IF(NF)90,516,90	184
516 NTOR=0	185
M=M+1	186
GO TO 505	187
90 CONTINUE	188
513 CLOCK=-CLOCK	189
WRITE OUTPUT TAPE 2,501,CLOCK	190
PRINT 330	191
330 FORMAT(52H NORMAL FRAME HALT REACHED. SAVE INFO. THANKS ROD)	192
PAUSE 1	193
END(0,1,0,0,1)	194

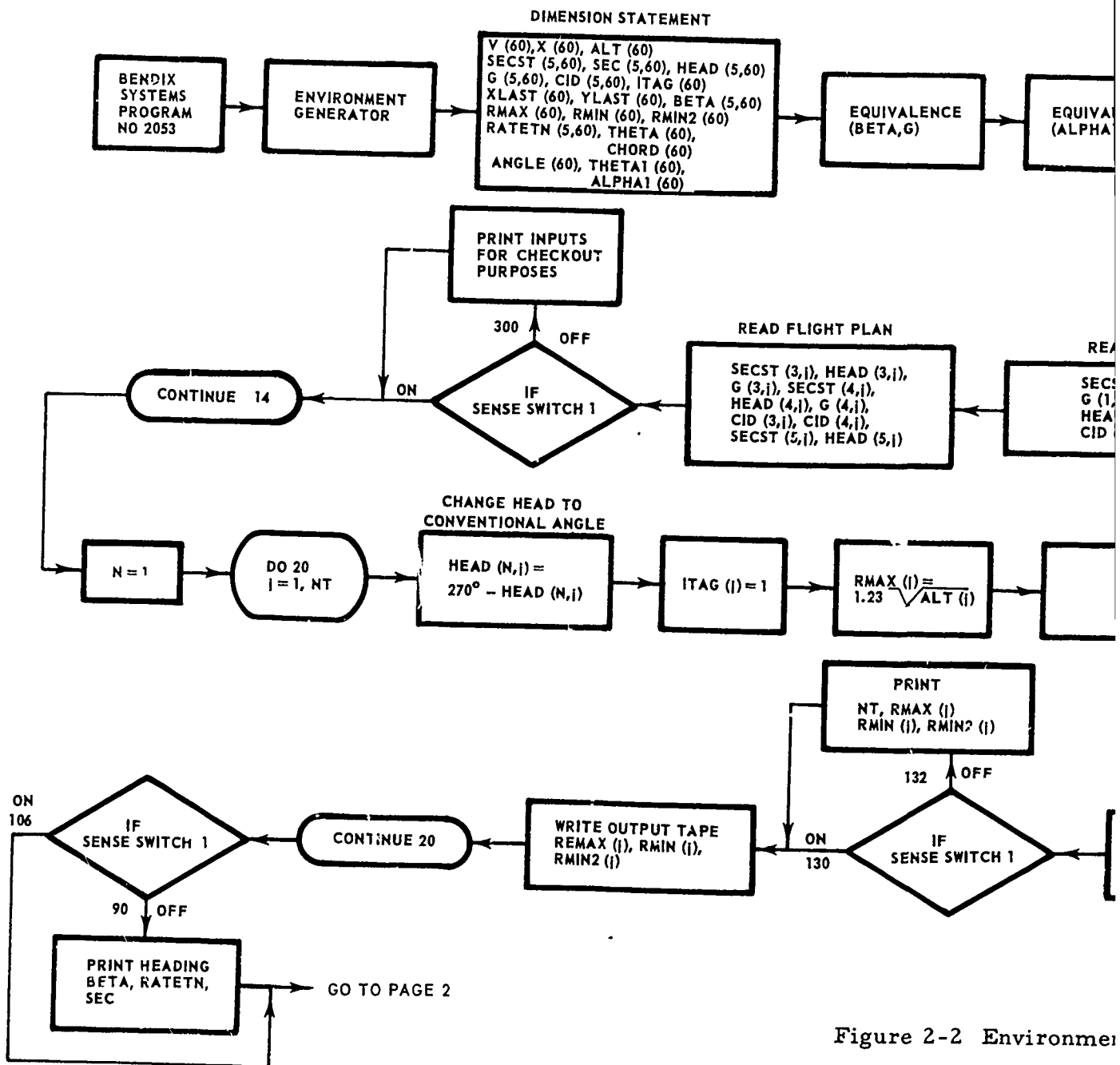


Figure 2-2 Environment

A.

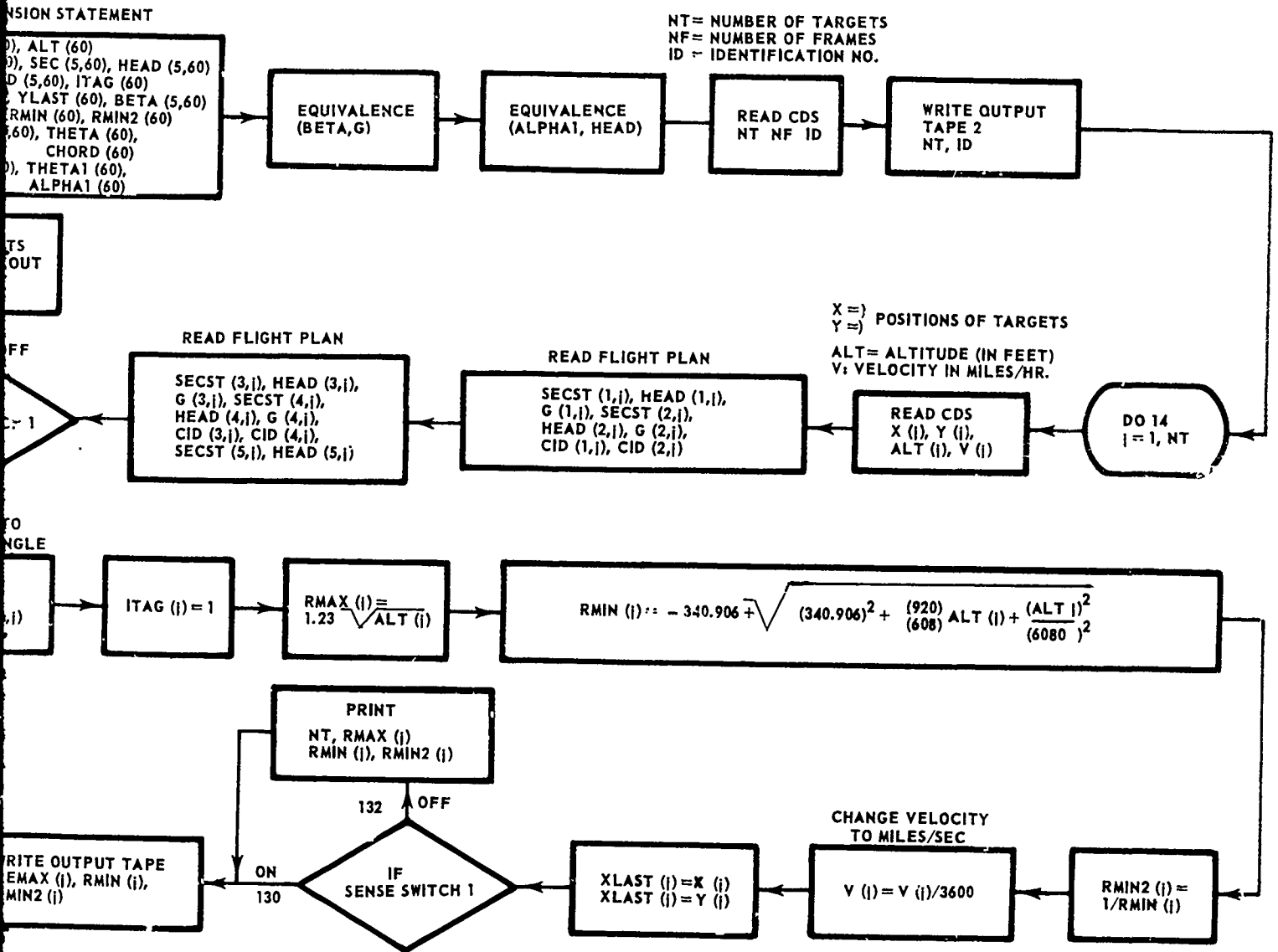
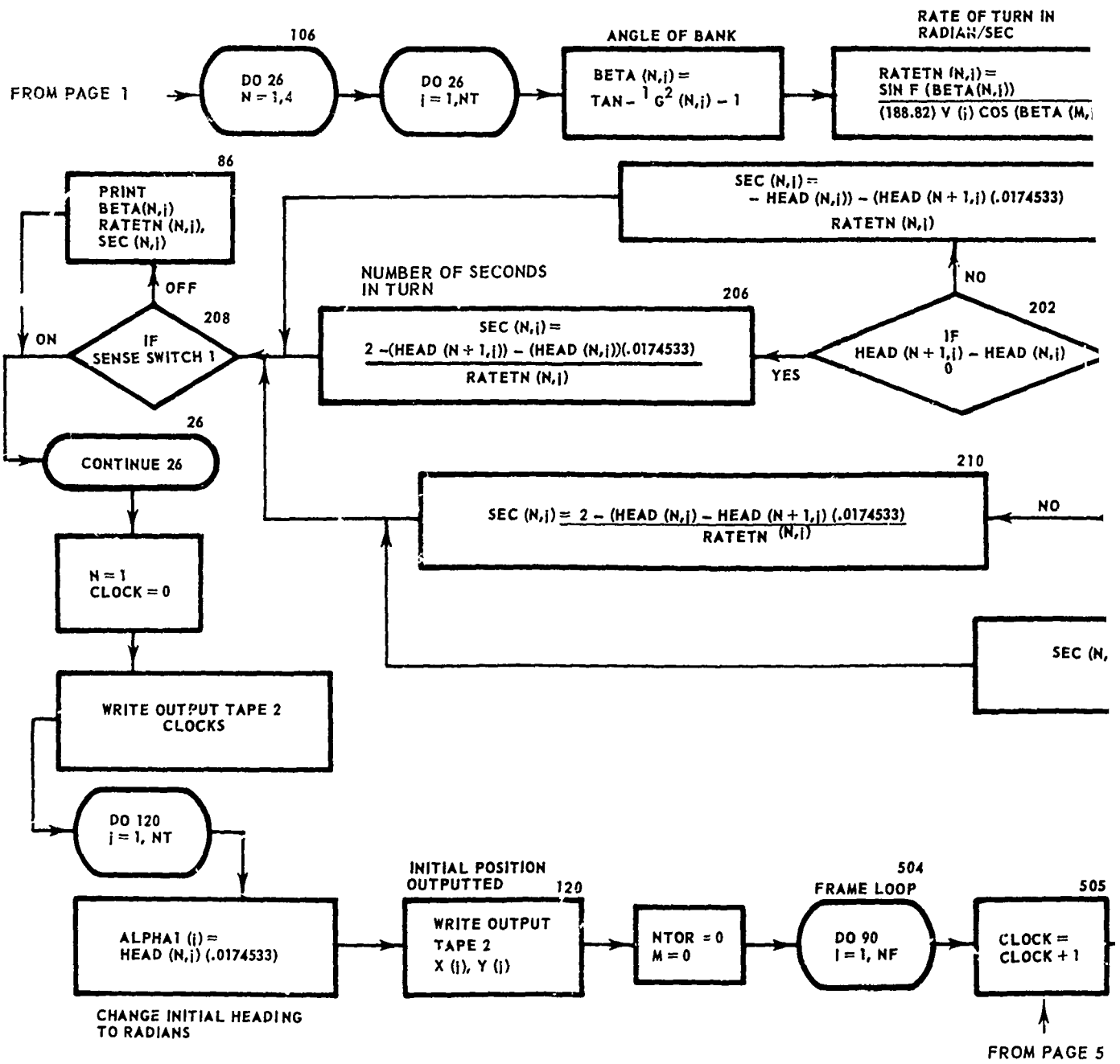


Figure 2-2 Environment Generator Modification Program Flow

B.



Figure

A,

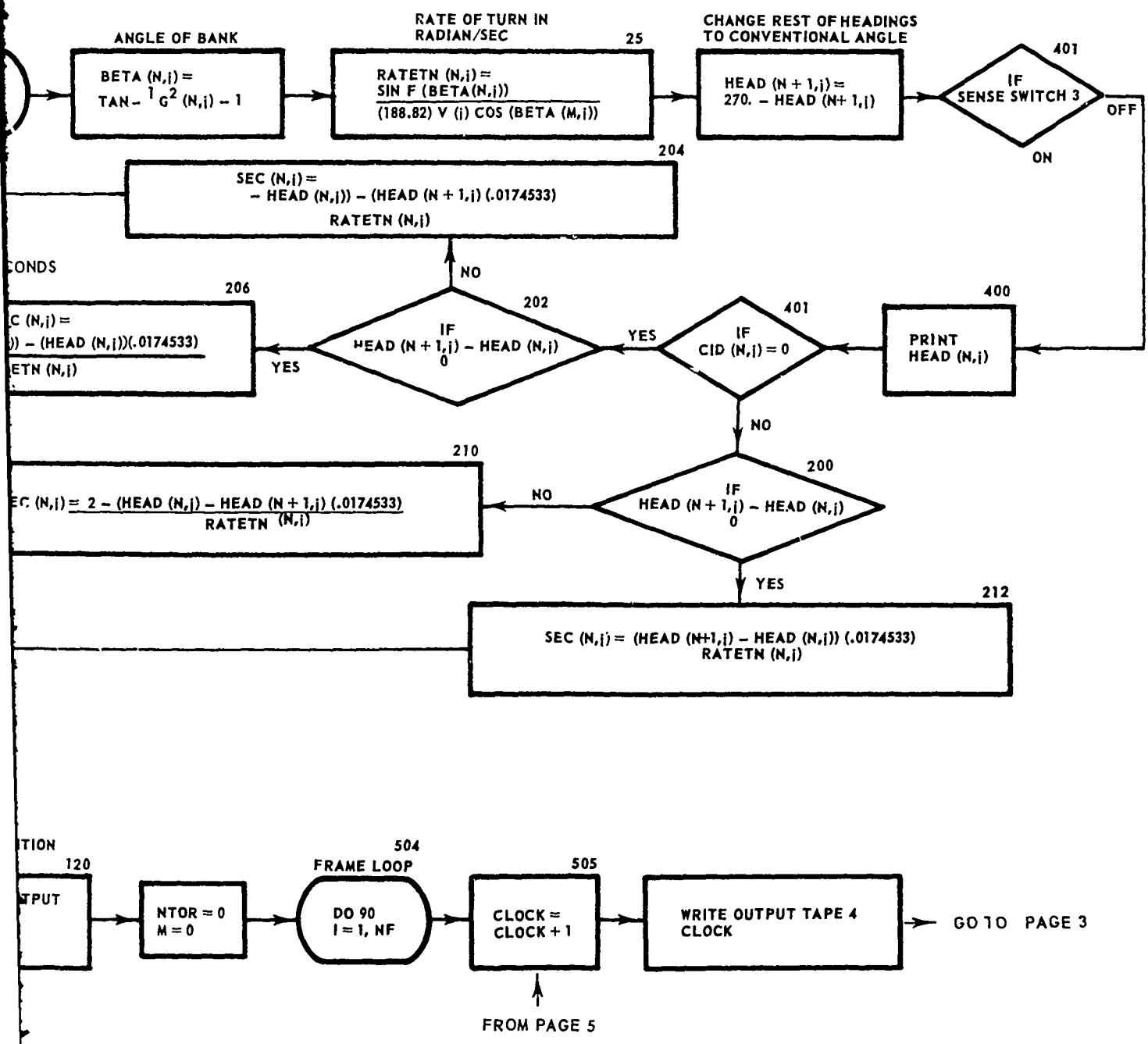
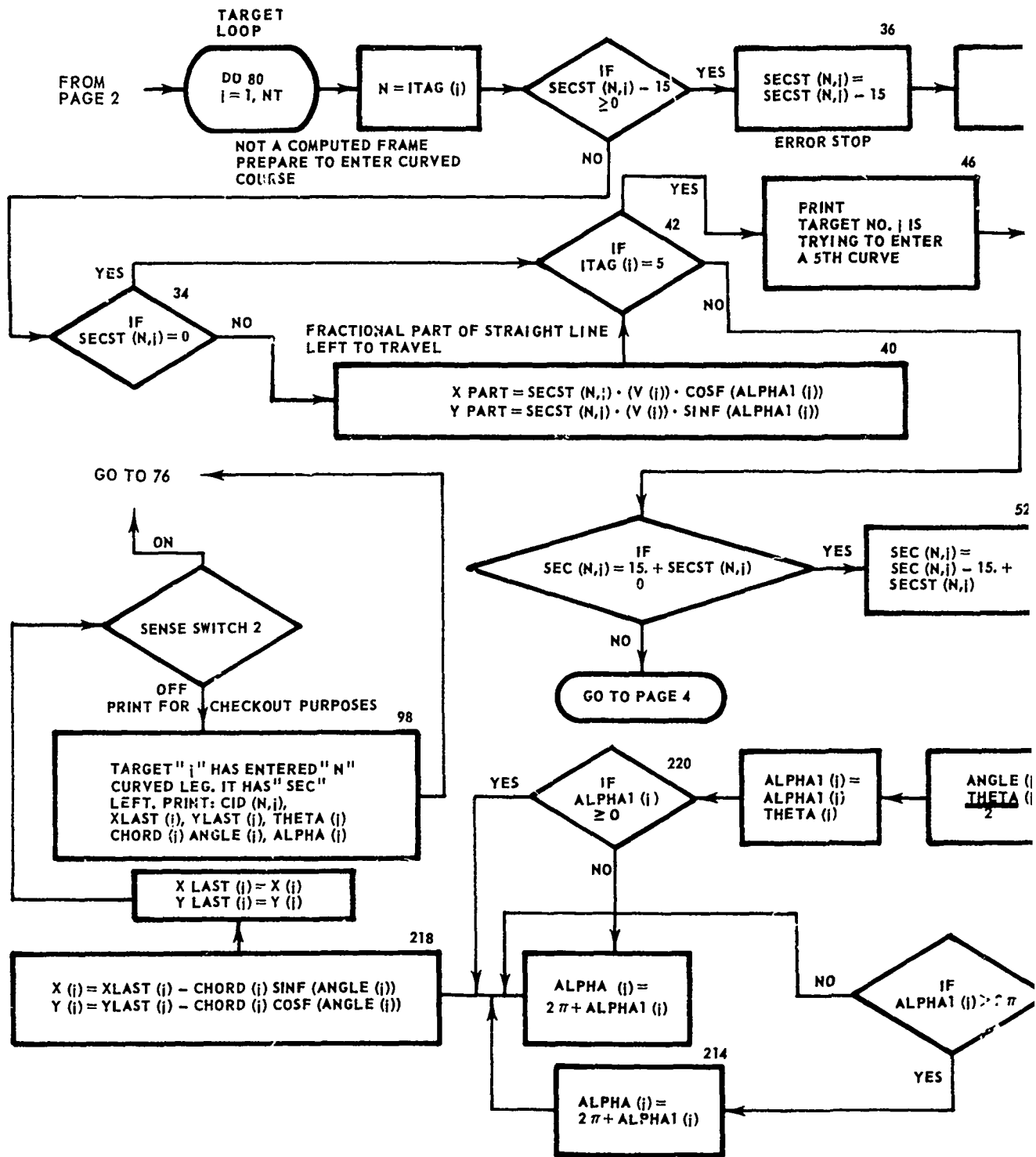


Figure 2-2 (Cont.)



A₁

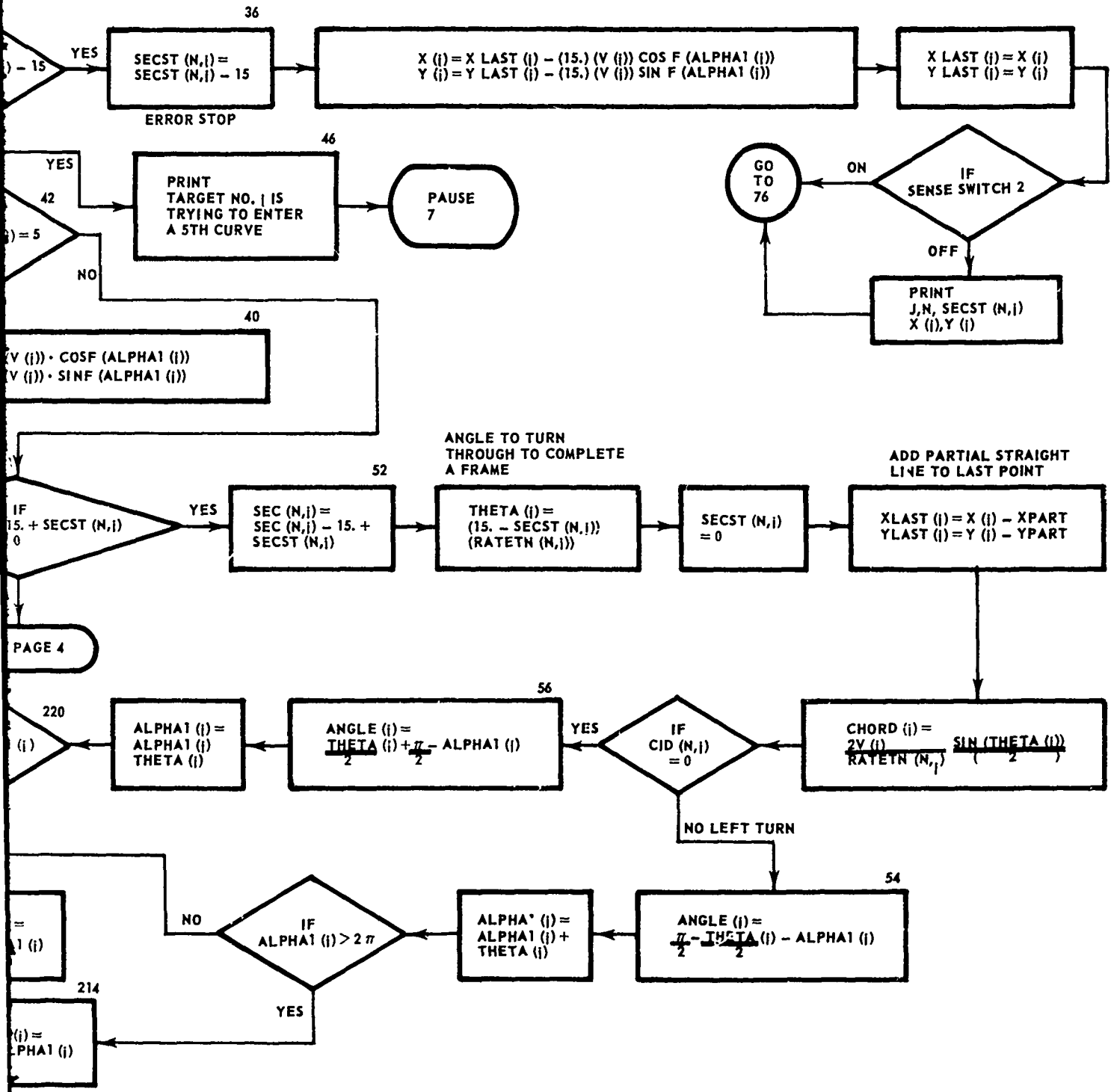
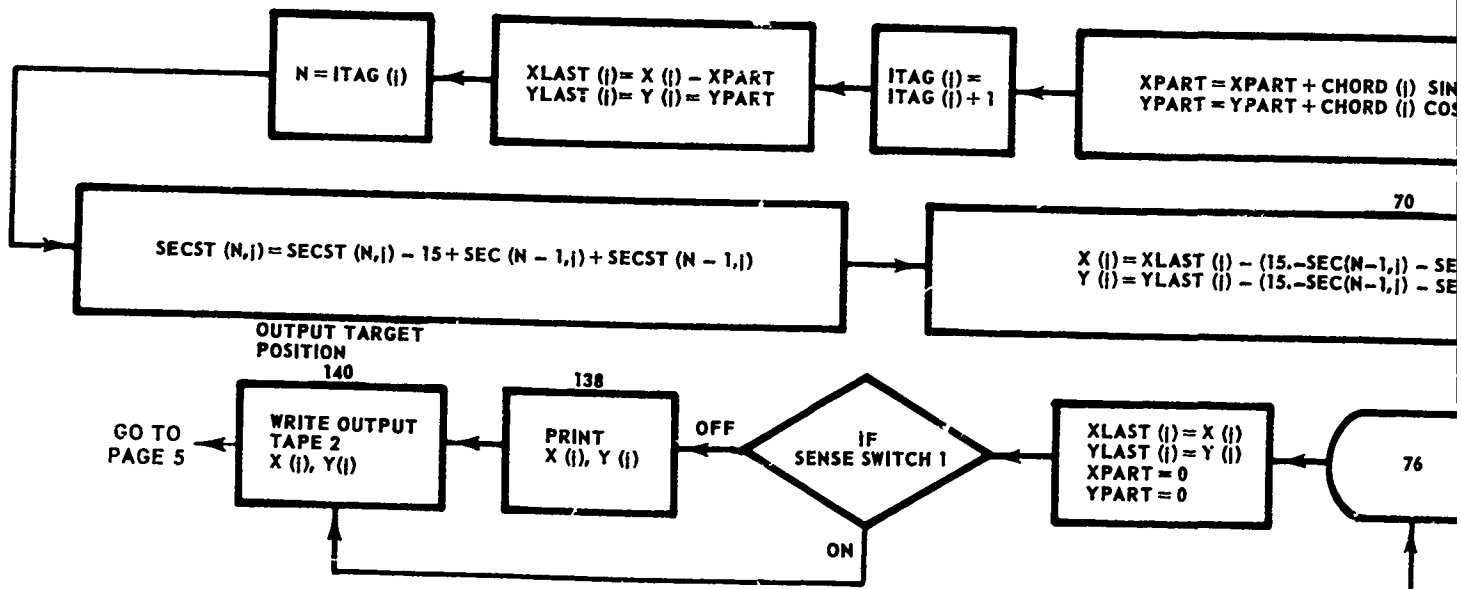
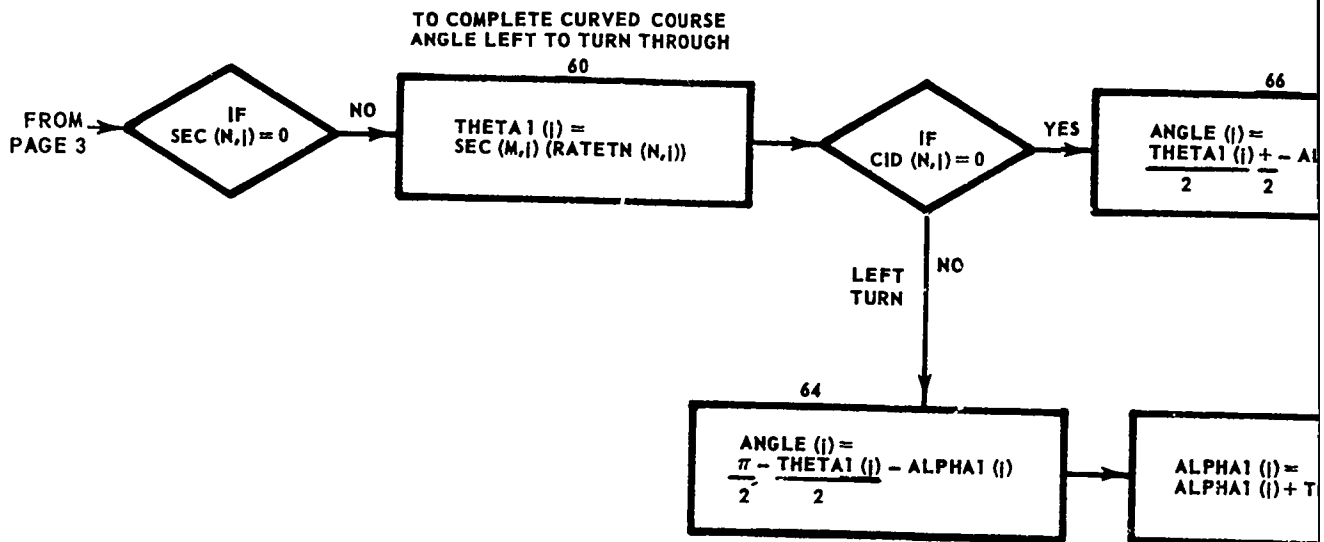


Figure 2-2 (Cont.)

P.



A.

ED COURSE
N THROUGH

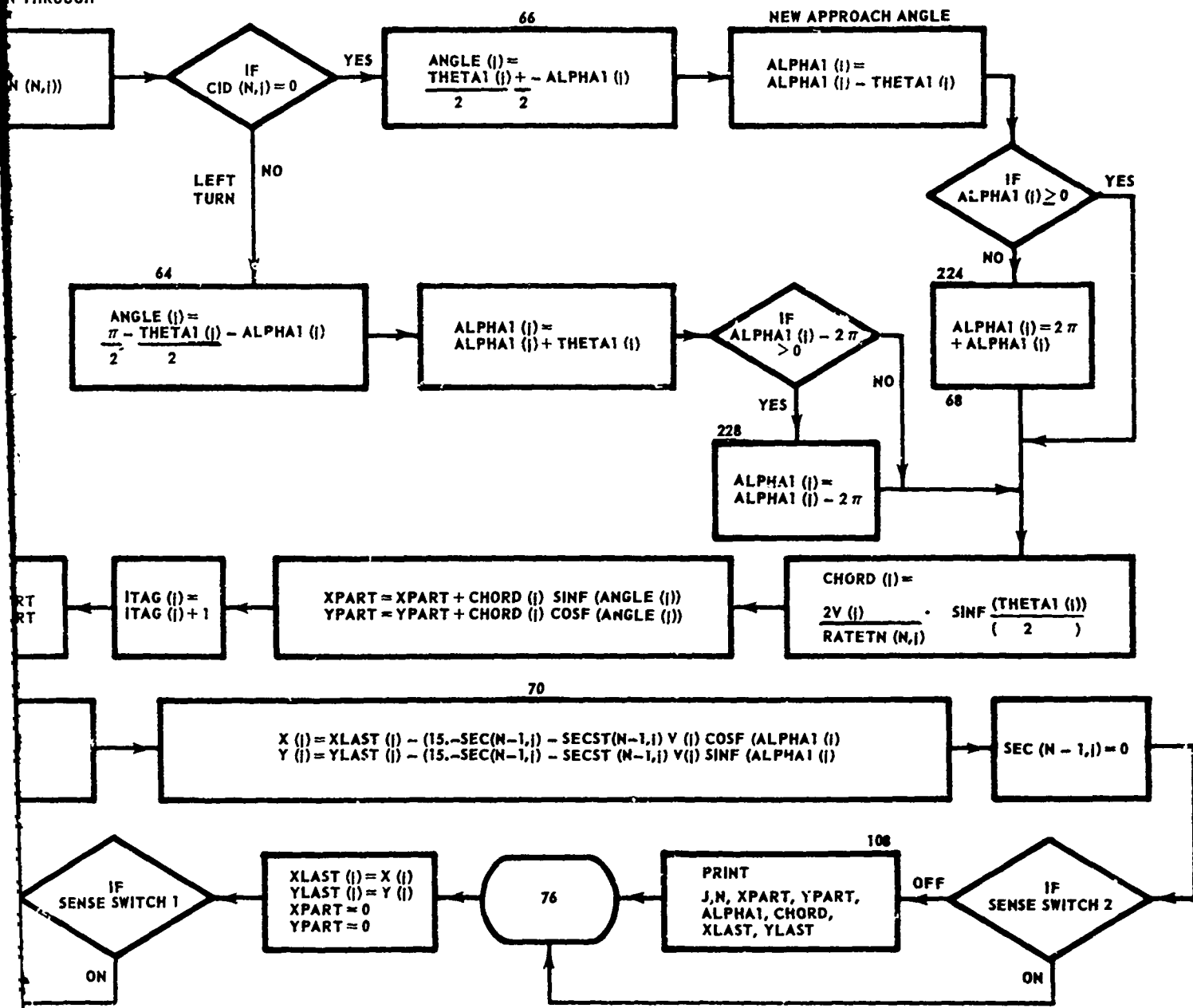
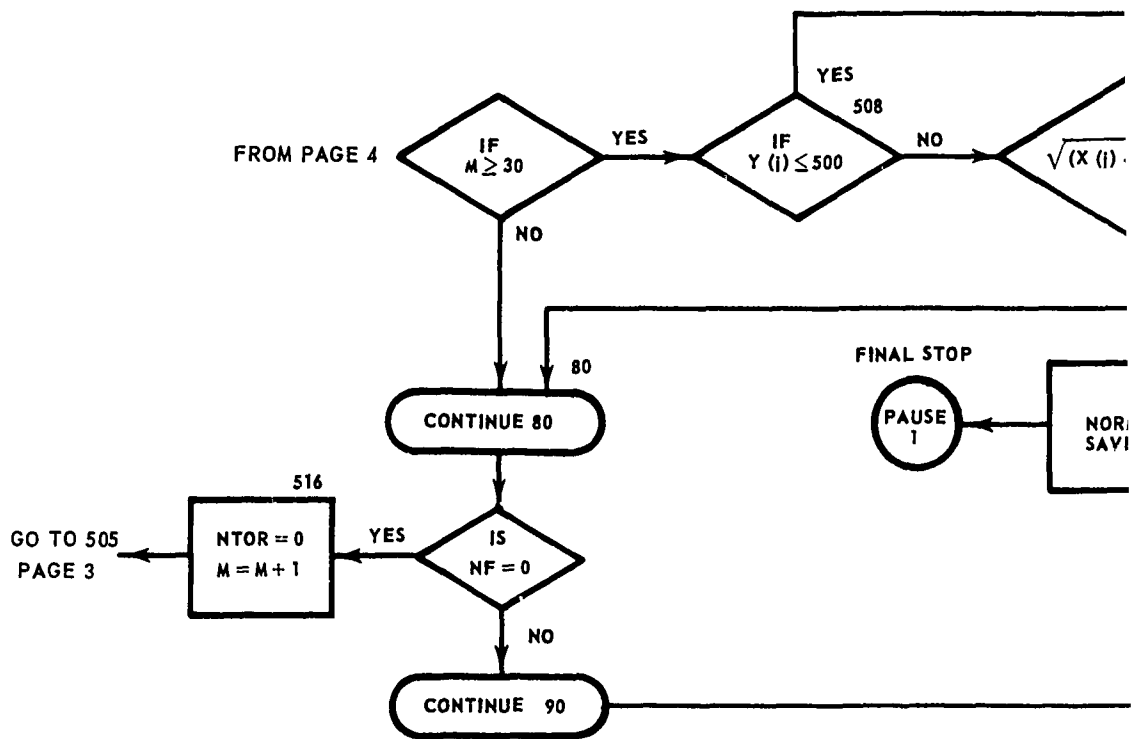


Figure 2-2 (Cont.)

B



A.

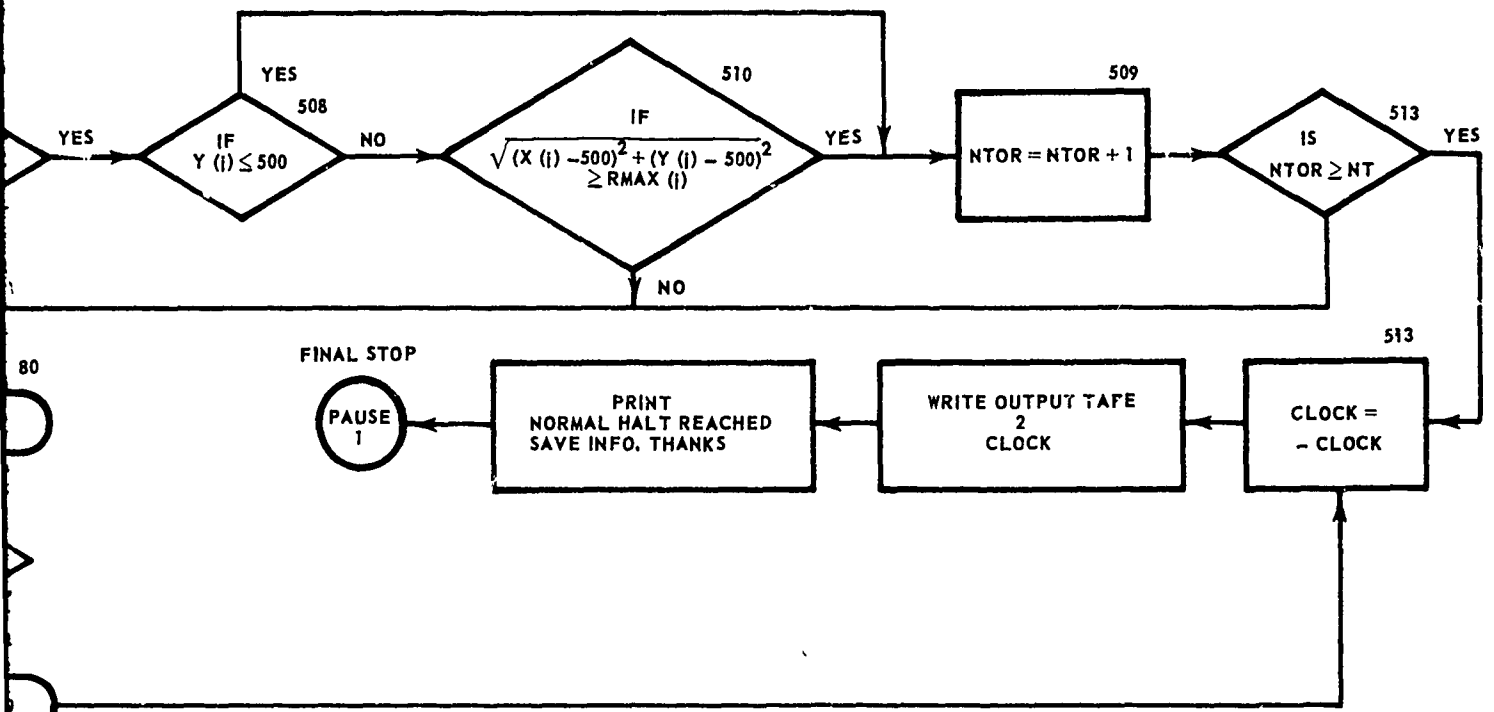


Figure 2-2 (Cont.)

B.

2.3 MODIFICATION TO ACCEPT NEW ENVIRONMENT GENERATOR

2.3.1 FORTRAN Listing

```
DIMENSION LJTHX(50),LTOTX(50),LAZDEX(50),NSIGNO(153)
DIMENSION NTTOAA(100),DISTT(100),NMST(100),NMSA(100),DISMS(100),
INHTT(100),NHTA(100),DISHT(100),NFST(100),NFSA(100),DISFS(100)
DIMENSION DIST(31),DISH(31)
DIMENSION DIFX(100),DIFY(100)
DIMENSION X(100),Y(100),RMIN1(100),RMIN2(100),RMAX(100),M(100),
IRO(100),R1(100),R2(100),AZ(100)
DIMENSION NRESOL(100),LJTH(50),LAZ(50),XAP(100),YAP(100),LAZDEL(50
1),LTOT(50)
READ 5050,MONTH,NDAY,NYEAR
5050 FORMAT(3I2)
1500 DO 5333 I=1,50
      LTOTX(I)=0
5333 LAZDEX(I)=0
      SUX=0.
      SUM=0.
      FRE1=0.
      FRE2=0.
      FRED=0.
      FREQ=0.
      FRED2=0.
      FREQ2=0.
      DXH=0.
      DXM=0.
PRINT 845,MONTH,NDAY,NYEAR
DO 8050 I=1,31
8050 DIST(I)=0.
5002 FORMAT(90H
XDATA. BENDIX SYSTEMS DIVISION 704.)
DO 5334 I=1,50
5334 LJTHX(I)=0
DO 8000 I=1,153
8000 NSIGNO(I)=0
      O=0.
READ 350,K1,K2,K3,K4,K5,K6,K7,K8,K9,K0
350 FORMAT(10I1)
READ 3,CASE,AZMUL,DELMUL,AZRES,DELRES,GAIN
3 FORMAT(6F10.1)
REWIND 2
READ INPUT TAPE 2,1150,NT,NF,ID
      FORTRAN LISTING 2

1150 FORMAT(3I5).
```

TLQ 8 PROGRAM

```

DO 1151 I=1,NT
1151 READ INPUT TAPE 2,1172,RMAX(I),RMIN(I),RMIN2(I)
      CLOCKL=NF
      CLOCKS=0.
      XX=CLOCKS
      PRINT 5002
      PRINT 700,CASE,NT
700   FORMAT(12H CASE NUMBERF8.0,4H NTI4)
      PRINT 1100,CLOCKL,AZMUL,DELMUL,AZRES,DELRES,GAIN
1100  FORMAT((16H           CLOCKLF5.0,6H AZMULF8.6,7H DELMULF8.6,6H AZRE
1SF8.7,7H DELRESF8.6,5H GAINF8.6)///)
      PRINT 701
701   FORMAT((118H CLOCK   K NSUMM1 NSUMM2 NSUMM3 NMS NHT NFS NFARAT NHT
1RAT NMSRAT NDTRAT KOUNT6 KOUNT7 KOUNT8 KOUNT5 KNTSUM KOUNT9 NDR
2 )///)
400   FORMAT(5F10.2)
1152  FORMAT(3E16.8)
      CONV=57.29577
      XO=500.
      YO=500.
      X1=456.6985
      X2=475.
      Y1=475.
      Y2=543.3015
      KK=0
      KK1=0
      KK2=0
      KK3=0
      KK4=0
      KK5=0
      KK6=0
      KK7=0
      KK8=0
      AK9=0.
      AK1=0.
599   FORMAT(F6.0)
7     READ INPUT TAPE 2,599,CLOCK
      IF(CLOCK) 301,1177,1177
1177  DO 1153 I=1,NT
1153  READ INPUT TAPE 2, 1154,X(I),Y(I)
1154  FORMAT(2E16.8)
      KOUNT1=0
      KOUNT2=0
      KOUNT3=0
      KOUNT4=0

```

FORTRAN LISTING 3

```

KOUNT5=0
KOUNT6=0
KOUNT7=0
KOUNT8=0
KOUNT9=0
NSUMM1=0
NSUMM2=0
NSUMM3=0
SUMLO=0.
SUMO =0.
SUML1=0.
SUML2=0.
DO 50 I=1,NT
DIFX(I)=X(I)-X0
DIFY(I)=Y(I)-Y0
RO(I)=SQRTF(DIFX(I)**2+DIFY(I)**2)
IF(RO(I)-RMAX(I)) 9,9,8
9 IF(RO(I)-RMIN(I)) 500,500,501
500 SUMLO=SUMLO+RMIN2(I)
GO TO 502
501 SUMLO=SUMLO+1./RO(I)**2
502 SUMO=SUMO+1./RO(I)**2
8 R1(I)=SQRTF((X(I)-X1)**2+(Y(I)-Y1)**2)
IF(R1(I)-RMAX(I)) 11,11,650

11 IF(R1(I)-RMIN(I)) 652,651,651
651 SUML1=SUML1+1./R1(I)**2
GO TO 650
652 SUML1=SUML1+RMIN2(I)
650 CONTINUE
13 R2(I)=SQRTF((X(I)-X2)**2+(Y(I)-Y2)**2)
IF(R2(I)-RMAX(I)) 15,15,660
15 IF(R2(I)-RMIN(I)) 662,661,661
661 SUML2=SUML2+1./R2(I)**2
GO TO 660
662 SUML2=SUML2+RMIN2(I)
660 CONTINUE
IF(DIFY(I)) 18,18,19
18 M(I)=3
NSUMM3=NSUMM3+1
GO TO 50
19 IF(RO(I)-RMAX(I)) 40,40,22
40 IF(DIFX(I))20,23,23
20 IF(R1(I)-RMAX(I)) 21,21,22
22 M(I)=4
GO TO 50
21 M(I)=1

```

FORTRAN LISTING 4

```

NSUMM1=NSUMM1+1
GO TO 26
23 IF(R2(I)-RMAX(I)) 25,25,22
25 M(I)=2
NSUMM2=NSUMM2+1
26 AZ(I)=ATN1F(DIFY(I),-DIFX(I))
50 CONTINUE
DO 56 I=1,50
LJTH(I)=0
LAZ(I)=0
LAZDEL(I)=0
56 LTOT(I)=0
K=0
LINDX=0
DO 100 J=1,NT
IF(M(J)) 100,58,58
58 IF(XABSF(M(J))-2) 60,60,100
60 L=0
DO 76 I=1,NT
IF(XABSF(M(I))-XABSF(M(J))) 76,62,76
62 IF((XABSF(AZ(J)-AZ(I)))-AZRES) 64,64,76
64 L=L+1
NRESOL(L)=I
76 CONTINUE
72 SIDLO=SUMO
AMB=0.
AMBL=0.
74 IF(XABSF(M(J))-1) 75,71,75
75 RADOMA=SUML2
GO TO 77
71 RADOMA=SUML1
77 DO 88 I=1,L
KBAS=NRESOL(I)
FRACT=1./RO(KBAS)**2
FLJ=1.-ABSF((AZ(J)-AZ(KBAS))/AZRES)
AMB=AMB+FLJ*FRACT
IF(FRACT-RMIN2(KBAS)) 275,275,274
274 FRACT=RMIN2(KBAS)
275 AMBL=AMBL+FLJ*FRACT
IF(J-KBAS) 88,276,88
276 TARGML=FRACT
IF(XABSF(M(J))-1) 83,85,83
83 IF(R2(KBAS)-RMIN(KBAS)) 680,680,681
680 RANGMA=RMIN(KBAS)**2
GO TO 682
681 RANGMA=R2(KBAS)**2
FORTRAN LISTING 5

```

```

682 CONTINUE
   DELTA=RO(J)-R2(J)
   GO TO 88
85  IF(R1(KBAS)-RMIN(KBAS)) 690,690,691
690 RANGMA=RMIN(KBAS)**2
   GO TO 692
691 RANGMA=R1(KBAS)**2
692 CONTINUE
   DELTA=RO(J)-R1(J)
88  CONTINUE
   IF(AMBL-(SUMO-AMB)*GAIN)4000,89,89
4000 KOUNT9=KOUNT9+1
   GO TO 100
89  AOSIGO=((SUMO-AMB)*GAIN+AMBL-TARGML)/TARGML
   AOSIGA=(RANGMA*ADOMA)-1.
   SIGNO=1./(2.+AOSIGO+AOSIGA+AOSIGO*AOSIGA)
   NS=SIGNO*500.
   NS=NS+1
   IF(NS-152) 8002,8002,8001
8001 NSIGNO(152)=NSIGNO(152)+1
   GO TO 8003
8002 NSIGNO(NS)=NSIGNO(NS)+1
8003 CONTINUE
   IF(SIGNO-.0112) 99,101,101
101  KOUNT1=KOUNT1+1
   GO TO 112
99  IF(SIGNO-.00841) 103,102,102
102  KOUNT2=KOUNT2+1
   IF(RAM2BF(O)-.6) 112,250,250
103  IF(SIGNO-.00631) 105,104,104
104  KOUNT3=KOUNT3+1
   IF(RAM2BF(O)-.2) 112,250,250
105  KOUNT4=KOUNT4+1
250  CONTINUE
   GO TO 100
112  KOUNT5=KOUNT5+1
   IF(L-1) 113,113,114
114  KOUNT6=KOUNT6+1
   LINDX=LINDX+1
   IF(LINDX-50) 73,73,122
73  LJTH(LINDX) =J
   LJTHX(L)=LJTHX(L)+1
   LAZ(LINDX) =L
   GO TO 122
113  A2T=AZ (J)
115  ANO=(RAM2BF(O)-.5)*AZMUL

```

FORTRAN LISTING 6

```

A2T=A2T+ANO
BNO=(RAM2BF(O)-.5)*DELMUL
DELTA=DELTA+ BNO
IF(XABSF(M(J))-1) 116,116,117
116 ANGLE=A2T+.5235938
GO TO 118
117 ANGLE=A2T-1.0471976
118 RANGE=(2500.-DELTA**2)/(100.*COSF(ANGLE)-2.*DELTA)
1298 CONTINUE
K=K+1
120 XAP(K)=XO-RANGE*COSF(A2T)
121 YAP(K)=YO+RANGE*SINF(A2T)
GO TO 100
122 NEEL=0
DO 130 I=1,L
KBAS=NRESOL(I)
IF(XABSF(M(J))-1) 123,123,124
124 IF(ABSF(RO(KBAS)-R2(KBAS)-DELTA)-DELRES) 127,127,130
123 IF(ABSF(RO(KBAS)-R1(KBAS)-DELTA)-DELRES) 127,127,130
127 NEEL=NEEL+1
M(KBAS)=-M(KBAS)
NRESOL(NEEL)=NRESOL(I)
130 CONTINUE
LAZDFX(NFEL)=LAZDEX(NEEL)+1
IF(NFEL-1) 517,517,132
517 KOUNT8=KOUNT8+1
IF(LINDX-50)7000,7000,113
7000 LAZDEL(LINDX)=1
LTOT(LINDX)=1
GO TO 113
132 KLOO=1
IF(LINDX-50)133,133,134
133 LAZDEL(LINDX)=NEEL
134 IF(NRESOL(KLOO)-J) 135,150,135
135 IND=NRESOL(KLOO)
IF(XABSF(M(J))-1) 136,136,137
136 ADELTA=RO(IND)-R1(IND)
GO TO 140
137 ADELTA=RO(IND)-R2(IND)
140 DO 149 I=1,NT
IF(M(I)) 149,777,777
777 IF(XABSF(M(I))-XABSF(M(J)))149,141,149
141 IF(ABSF(AZ(I)-AZ(KLOO))-AZRES) 142,142,149
142 IF(XABSF(M(J))-1) 146,146,143
146 IF(ABSF(RO(I)-R1(I)-ADELTA)-DELRES) 144,144,149
144 NEFL=NEEL+1
FORTRAN LISTING 7

NRESOL(NEEL)=I
M(I)=-M(I)
GO TO 149

```

```

143 IF(ABSF(RO(I)-R2(I)-ADELTA)-DELRES) 144,144,149
149 CONTINUE
150 KLOO=KLOO+1
    IF(KLOO-NEEL) 134,134,152
152 LTOTX(NEEL)=LTOTX(NEEL)+1
    IF(LINDX-50)155,155,156
155 LTOT(LINDX)=NEEL
156 SUMAZ=0.
    DENOM=NEEL
    KOOUNT7=KOOUNT7+NEEL
    A2T=0.
    DO 160 I=1,NEEL
    KBAS=NRESOL(I)
    IF(AZ(KBAS)-A2T) 160,160,277
277 A2T=AZ(KBAS)
    IF(XABSF(M(J))-1)278,278,279
278 DELTA=RO(KBAS)-R1(KBAS)
    GO TO 160
279 DELTA=RO(KBAS)-R2(KBAS)
160 CONTINUE
    GO TO 115
100 CONTINUE
    NMS=0
    NHT=0
    NFS=0
    IF(K)570,571,570
571 DO 572 ITR=1,NT
    NMS=NMS+1
    NMST(NMS)=ITR
    NMSA(NMS)=-0
572 DISMS(NMS)=-0
    GO TO 575
570 DO 168 ITR=1,NT
    IAP=1
    NTTOAA(ITR) =1
    XTRU=X(ITR)
    YTRU=Y(ITR)
    DIFRX=ABSF(XTRU-XAP(IAP))
    DIFRY=ABSF(YTRU-YAP(IAP))
    DISSQ=DIFRY**2+DIFRX**2
    DO 166 IAP=2,K
    TDIFRX=ABSF(XTRU-XAP(IAP))
    TDIFRY=ABSF(YTRU-YAP(IAP))
    FORTRAN LISTING 8

    IF(TDIFRX-DIFRX) 165,164,164
164 IF(TDIFRY-DIFRY)165,166,166
165 TDISSQ=TDIFRX**2+TDIFRY**2
    IF(TDISSQ-DISSQ) 167,166,166

```

```

167  NTTOAA(ITR)=IA
      DISSQ=TDISSQ
      DIFRX=TDIFRX
      DIFRY=TDIFRY
166  CONTINUE
168  DISTT(ITR)=SQRTF(DISSQ)
171  DO 178 IAP=1,K
      ITR=1
      NATOTT=1
      YAPP=YAP(IAP)
      XAPP=XAP(IAP)
      DIFRX=ABSF(X(ITR)-XAPP)
      DIFRY=ABSF(Y(ITR)-YAPP)
      DISSQ=DIFRX**2+DIFRY**2
172  DO 175 ITR=2,NT
      TDIFRX=ABSF(X(ITR)-XAPP)
      TDIFRY=ABSF(Y(ITR)-YAPP)
      IF(TDIFRX-DIFRX) 169,170,170
170  IF(TDIFRY-DIFRY) 169,175,175
169  TDISSQ=TDIFRX**2+TDIFRY**2
      IF(TDISSQ-DISSQ) 173,175,175
173  NATOTT=ITR
      DISSQ=TDISSQ
      DIFRX=TDIFRX
      DIFRY=TDIFRY
175  CONTINUE
      DISTA=SQRTF(DISSQ)
      IF(XABSF(NTTOAA(NATOTT))-IAP) 176,177,176
177  NHT=NHT+1
      NHTT(NHT)=NATOTT
      NHTA(NHT)=IAP
      DISHT(NHT)=DISTT(NATOTT)
      MHD=DISHT(NHT)*4.
      MHD=MHD+1
      IF(MHD-31) 1404,1405,1405
1405  DISH(31)=DISH(31)+1.
      GO TO 1406
1404  DISH(MHD)=DISH(MHD)+1.
1406  CONTINUE
      NTTOAA(NATOTT)=-NTTOAA(NATOTT)
      GO TO 178
176  NFS=NFS+1
      FORTRAN LISTING 9

      NFST(NFS)=NATOTT
      NFSA(NFS)=IAP
      DISFS(NFS)=SQRTF(DISSQ)
178  CONTINUE
      DO 440 ITR=1,NT
      IF(NTTOAA(ITR)) 440,440,442

```

```

442  NMS=NMS+1
      NMS7(NMS)=ITR
      NMSA(NMS)=NTTOAA(ITR)
      DISMS(NMS)=DISTT(ITR)
      MDS=DISMS(NMS)/5.
      MDS=MDS+1
      IF(MDS-31) 1400,1401,1401
1401  DIST(31)=DIST(31)+1.
      GO TO 1402
1400  DIST(MDS)=DIST(MDS)+1.
1402  CONTINUE
440   CONTINUE
575   CONTINUE
      NFARAT=(1000*NFS)/K
      NMSRAT=(1000*NMS)/NT
      NDTRAT=(1000*K)/NT
      NHTRAT=(1000*NHT)/K
      KNTSUM=(10*KOUNT1+6*KOUNT2+2*KOUNT3)
      NDR=(1000*K)/(NSUMM1+NSUMM2)
      ANDR=NDR
      ANDR=ANDR/1000.
      AND2=ANDR**2
      AK1=AK1+AND2
      XX=XX+1.
      IF(XX-(51.+CLOCKS))844,843,844
843   PRINT 845,MONTH,NDAY,NYEAR
      XX=CLOCKS
      PRINT 5002
      PRINT 700,CASE,NT
845   FORMAT(108H1
1
22)
      PRINT 701
844   CONTINUE
      PRINT 702,CLOCK,K,NSUMM1,NSUMM2,NSUMM3,NMS,NHT,NFS,NFARAT,NHTRAT,
INMSRAT,NDTRAT,KOUNT6,KOUNT7,KOUNT8,KOUNT5,KNTSUM,KOUNT9,NDR
      KK=KK+K
      KK2=KK2+NSUMM1
      KK3=KK3+NSUMM2
      KK4=KK4+NMS
      FORTRAN LISTING 10

      KK5=KK5+NHT
      KK6=KK6+NFS
      KK7=KK7+KOUNT6
      KK8=KK8+KOUNT8
      AK9=AK9+ANDR
702   FORMAT(F6.0,I4,I4,3I7,2I4,I5,3I7,I6,3I7,2I8,I5)

```

```

      IF(K3) 8075,8075,8076
8076  CONTINUE
      WRITE OUTPUT TAPE 4,900,CASE,CLOCK,NT,K
900   FORMAT(F11.6,F10.0,2I5)
      DO 901 I=1,NT
901   WRITE OUTPUT TAPE 4,902,X(I),Y(I)
902   FORMAT(2F15.4)
      IF(K) 6000,6001,6000
6000  DO 903 I=1,K
903   WRITE OUTPUT TAPE 4,902,XAP(I),YAP(I)
6001  CONTINUE
8075  CONTINUE
359   IF(K6) 361,361,362
362   KEN=6
      IF(NMS) 2100,2200,2100
2100  DO 906 I=1,NMS
906   WRITE OUTPUT TAPE 3,907,KEN,CASE,CLOCK,NMST(I),NMSA(I),DISMS(I)
2200  CONTINUE
361   IF(K7) 363,363,364
364   KEN=7
      IF(NHT) 2101,2102,2101
2101  DO 908 I=1,NHT
      IS=NHTA(I)
      ID=NHTT(I)
908   WRITE OUTPUT TAPE 3,907,KEN,CASE,CLOCK,NHTT(I),NHTA(I),DISHT(I),
IX(ID),Y(ID),XAP(IS),YAP(IS)
2102  CONTINUE
363   IF(K8) 365,365,366
366   KEN=8
      IF(NFS) 2103,2104,2103
2103  DO 909 I=1,NFS
909   WRITE OUTPUT TAPE 3,907,KEN,CASE,CLOCK,NFST(I),NFSA(I),DISFS(I)
365   CONTINUE
2104  CONTINUE
      GO TO 7
301   KEN=0
5555  FORMAT(4H TOTI6,I5,I6,I13,2I5,I31,I14,F28.3)
      PRINT 5555,KK,KK2,KK3,KK4,KK5,KK6,KK7,KK8,AK9
      PRINT 6500,AK1
6500  FORMAT(F120.6)
      FORTRAN LISTING 11

5557  FORMAT(7H TOTALI14,I9,I15)
      PRINT 845,MONTH,NDAY,NYEAR
      PRINT 700,CASE,NT
      PRINT 5002
      PRINT 1666
1666  FORMAT(56H
      HITS
      MISSES
      1)

```

```

PRINT 1667
1667 FORMAT(70H LOWER UPPER NUMBER STD FREQ LOWER UPPER NUM
1BER STD FREQ )
1408 FORMAT(37H HITS MISSES )
1409 FORMAT(46H LOWER UPPER NUMBER LOWER UPPER NUMBER )
XLOW=0.
HIGH=5.
XLO=0.
HIG=.25
DO 1668 I=1,31
SUX=SUX+DIST(I)
1668 SUM=SUM+DISH(I)
SUX=1000./SUX
SUM=1000./SUM
DO 1411 I=1,31
LXX=DISH(I)*SUM+.5
LYY=DIST(I)*SUX+.5
PRINT 1410,XLO,HIG,DISH(I),LXX,XLOW,HIGH,DIST(I),LYY
FREQ=(HIG-XLO)/2.+XLO
FRED=(HIGH-XLOW)/2.+XLOW
FREQ2=FREQ2+FREQ**2*DISH(I)
FRED2=FRED2+FRED**2*DIST(I)
FRE1=FRE1+FREQ*DISH(I)
FRE2=FRE2+FRED*DIST(I)
DXH=DXH+DISH(I)
DXM=DXM+DIST(I)
XLOW=HIGH
HIGH=HIGH+5.
XLO=HIG
1411 HIG=HIG+.25
1410 FORMAT(F10.2,F7.2,F8.0,I10,F10.0,F7.0,F8.0,I10)
SIGMAH=SQRTF(FREQ2/DXH-(FRE1/DXH)**2)
SIGMAM=SQRTF(FRED2/DXM-(FRE2/DXM)**2)
BARH=FRE1/DXH
BARM=FRE2/DXM
PRINT 1669,SIGMAH,SIGMAM,BARH,BARM
1669 FORMAT(13H SIGMA HITF8.3,11H SIGMA MISSF8.3,9H MEAN HITF9.3,10H
1 MEAN MISSF9.3)
IF(K0) 369,369,370
FORTRAN LISTING 12

370 HIGH=.002
PRINT 845,MONTH,NDAY,NYEAR
5000 FORMAT (1H1)
PRINT 5002
PRINT 700,CASE,NT
PRINT 450
450 FORMAT(48H FREQUENCY DISTRIBUTION SIGNAL TO NOISE )

```

```

PRINT 451
451  FORMAT(42H          LIMITS                LIMITS )
      PRINT 452
452  FORMAT(86H LOWER    UPPER    NUMBER    LOWER    UPPER    NUMBER
1     LOWER    UPPER    NUMBER )
      XLOW=.0
      XLOWR=.102
      XLO=.204
      XHIG=.206
      HIGHR=.104
      DO 453 I=1,51
      PRINT 454,XLOW,HIGH,NSIGNO(I),XLOWR,HIGHR,NSIGNO(I+51),XLO,XHIG,
INSIGNO(I+101)
454  FORMAT(F6.3,F9.3,I8,F13.3,F9.3,I8,F13.3,F9.3,I8)
      XLO=XHIG
      XHIG=XHIG+.002
888  FORMAT(F6.1,F9.3,I8,F13.3,F9.3,I8)
      XLOW=HIGH
      HIGH=HIGH+.002
      XLOWR=HIGHR
453  HIGHR=HIGHR+.002
369  CONTINUE
      465  FORMAT(17H DURING THIS RAID14,53H AZIMUTH UNRESOLVE SECTORS WERE F
      XOUND WHICH CONTAINED15,8H TARGETS)
      IF(K1) 460,460,461
461  PRINT 845,MONTH,NDAY,NYEAR
      PRINT 700,CASE,NT
      PRINT 472
      LL1=0
      LL2=0
      LL3=0
      DO 464 I=2,50
      LL1=LL1+LJTHX(I)
      LL2=LL2+LAZDEX(I)
5556 LL3=LL3+LTOTX(I)
464  PRINT 462,I,LJTHX(I),LAZDEX(I),LTOTX(I),I
      PRINT 5557,LL1,LL2,LL3
472  FORMAT(57H NO TARGETS    AZ UNR    AZ+DEL    AZ+DEL+OTHERS    NO TARGET
15 )
      FORTRAN LISTING 13

462  FORMAT(112,219,115,19)
460  CONTINUE
      PAUSE 1
      GO TO 1500
907  FORMAT(15,F10.5,F5.0,2I8,5F8.2)
904  FORMAT(15,F10.5,F5.0,5I8)
      END(0,1,1,0,1)

```

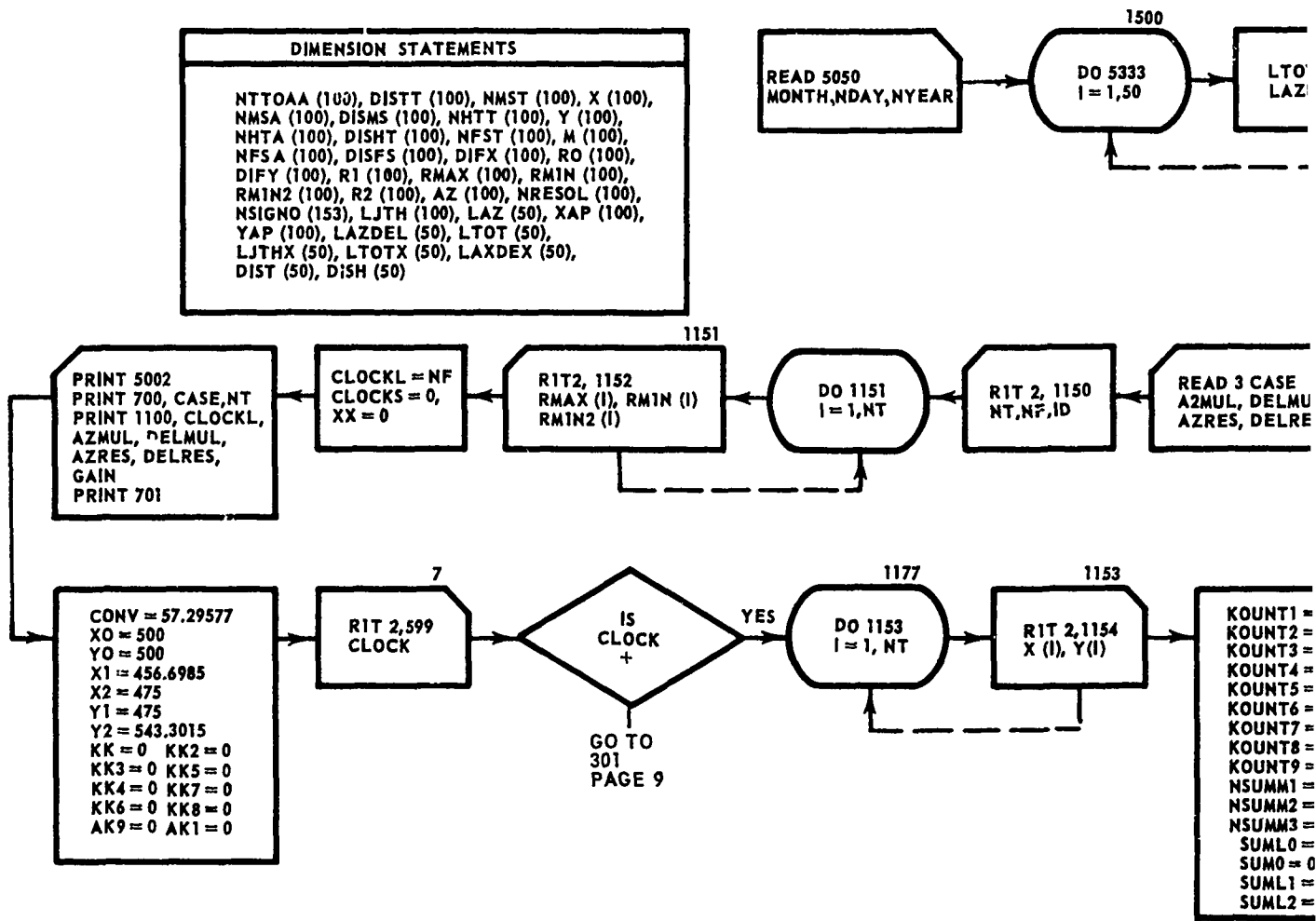


Figure 2-3 Acc
P1

A.

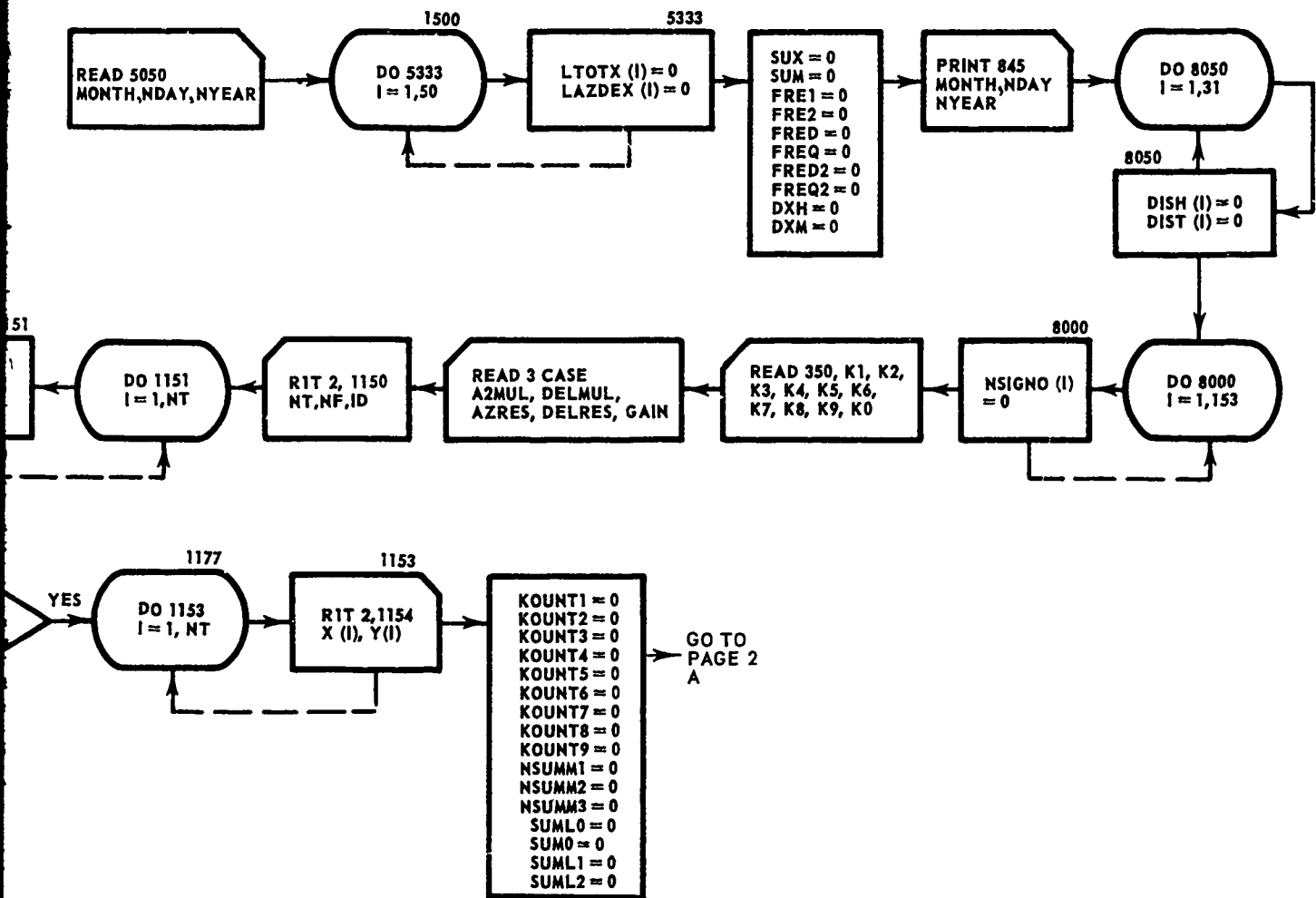
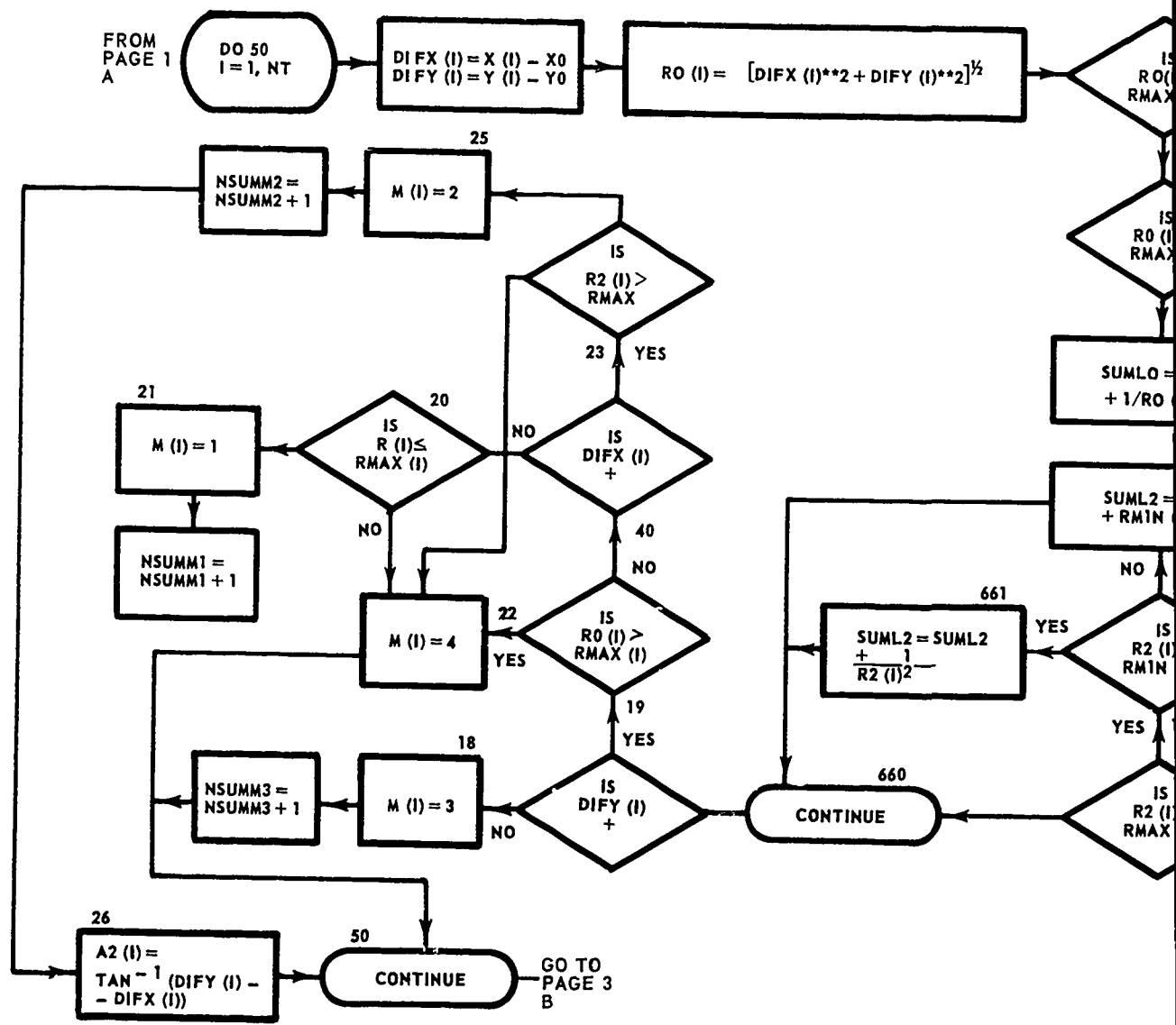


Figure 2-3 Accept New Environment Generator Program Flow Diagram

B.

FROM PAGE 1
A



A.

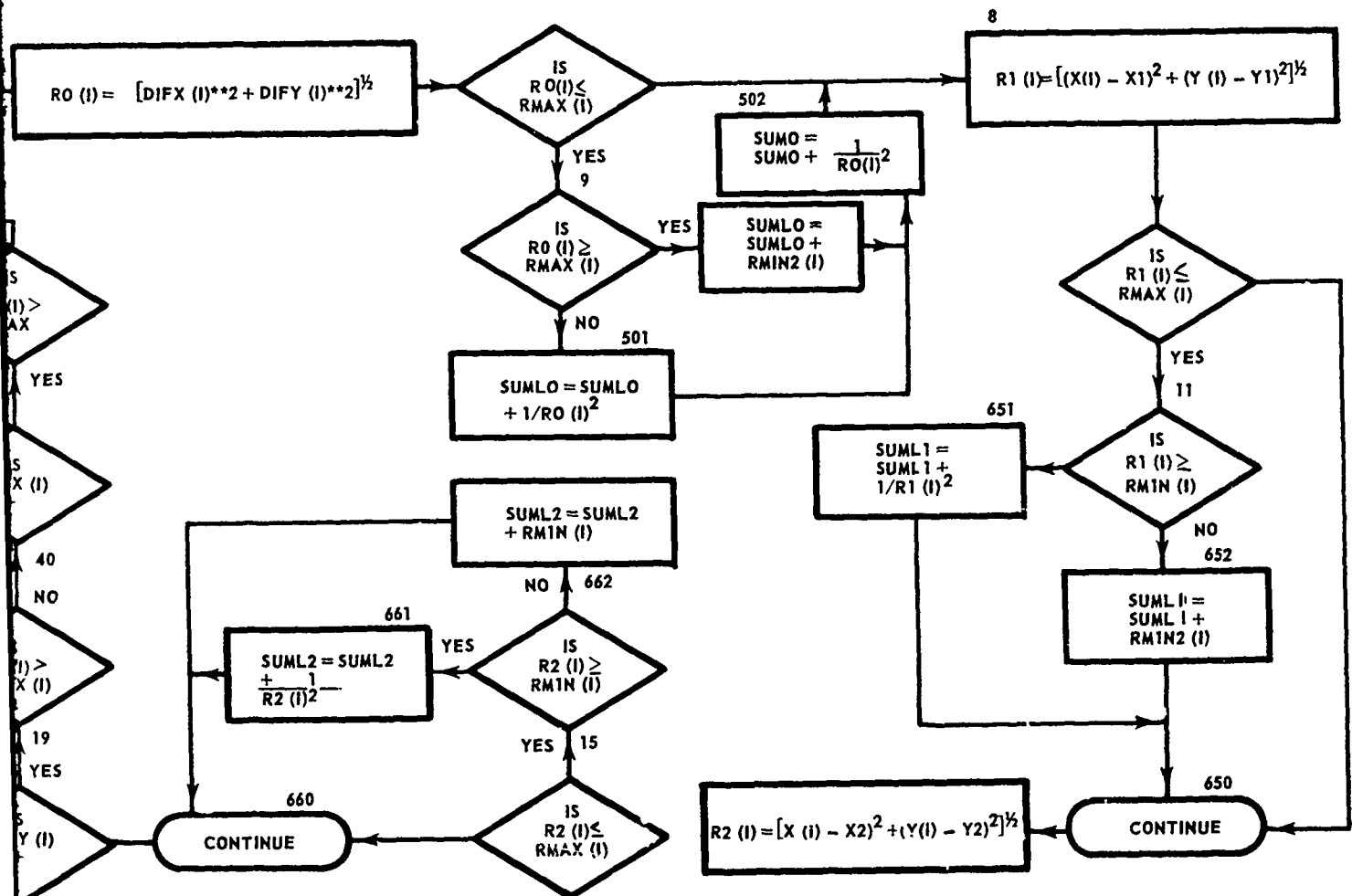
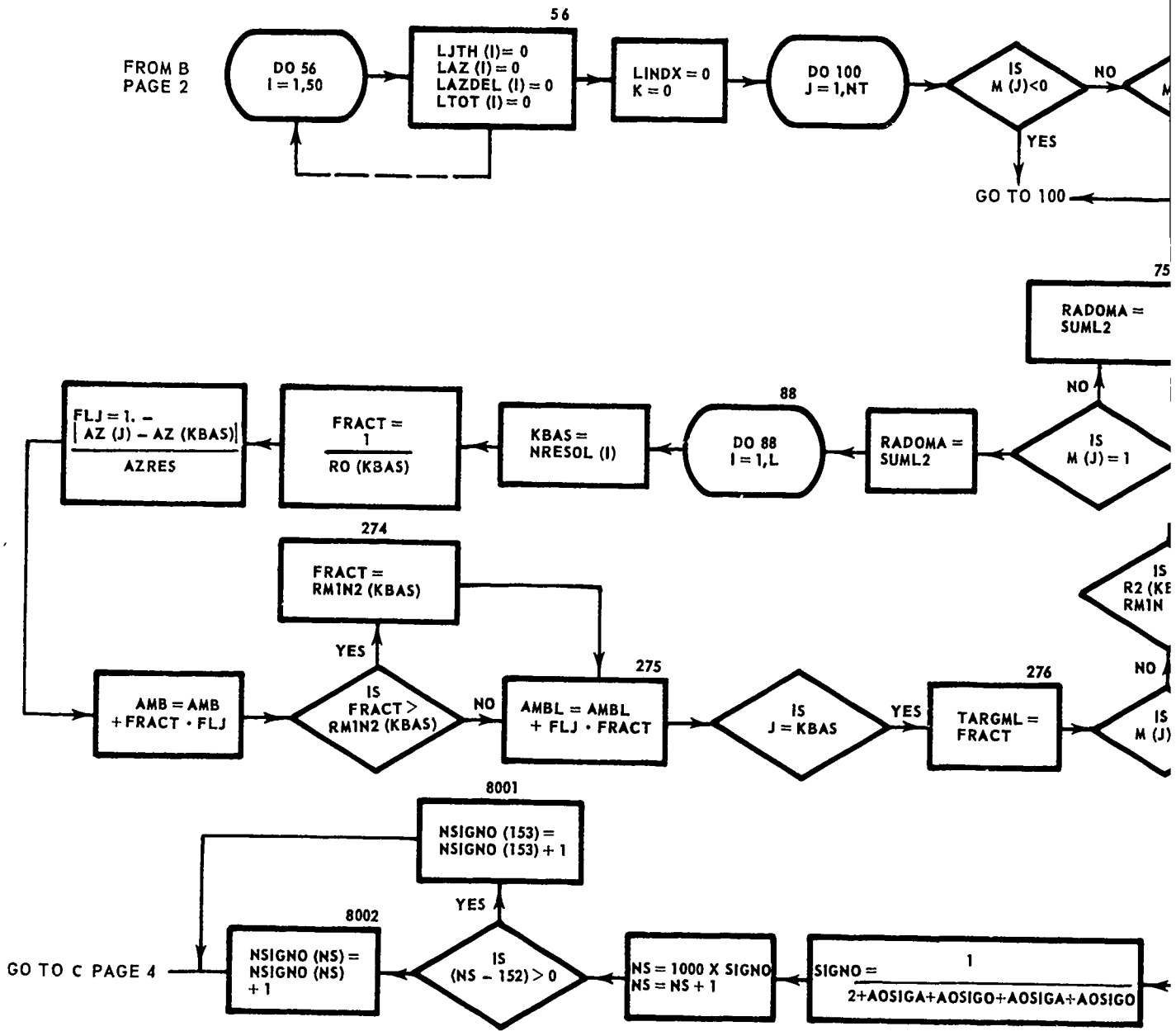


Figure 2-3 (Cont.)

B

FROM B
PAGE 2



A

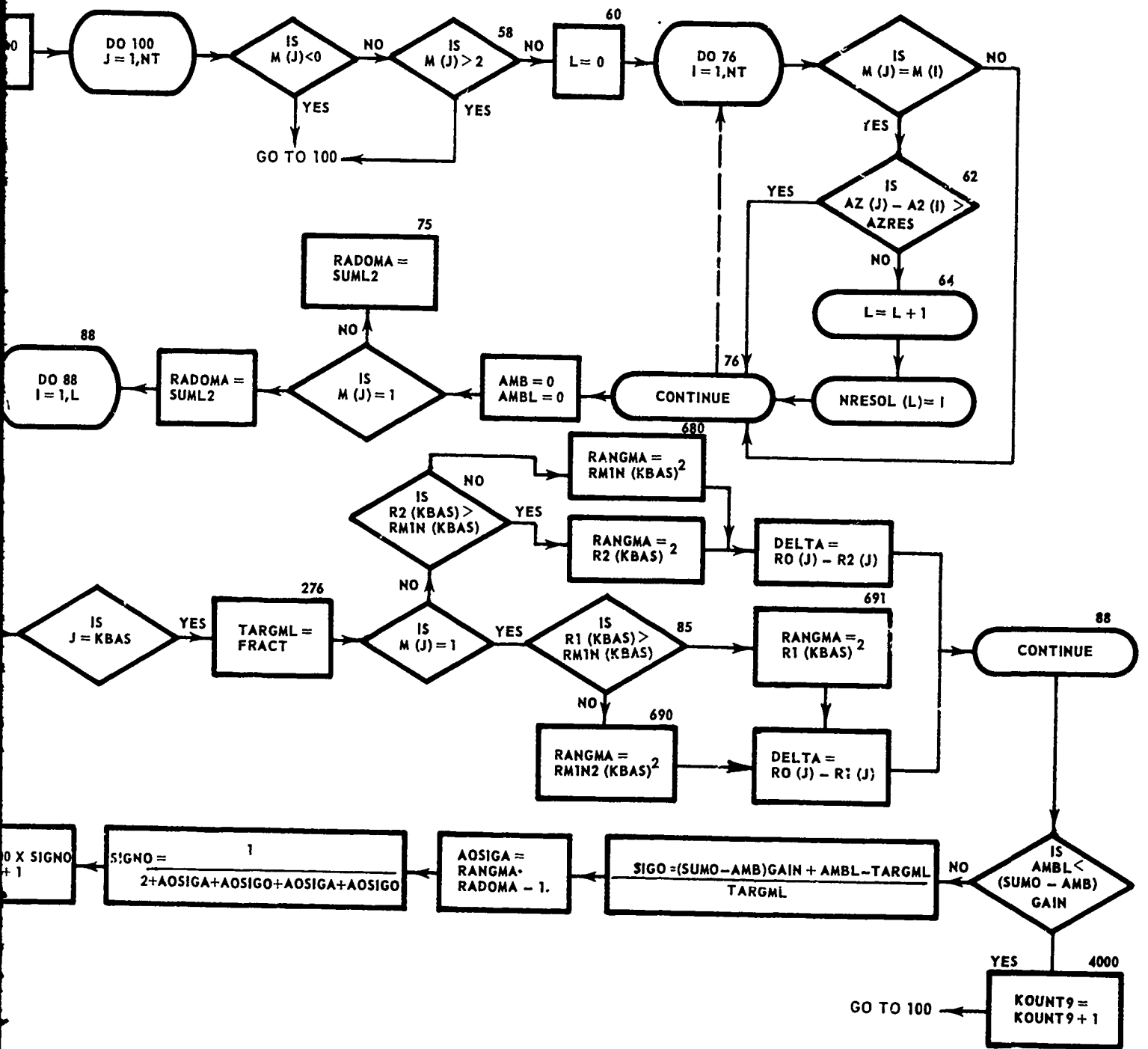
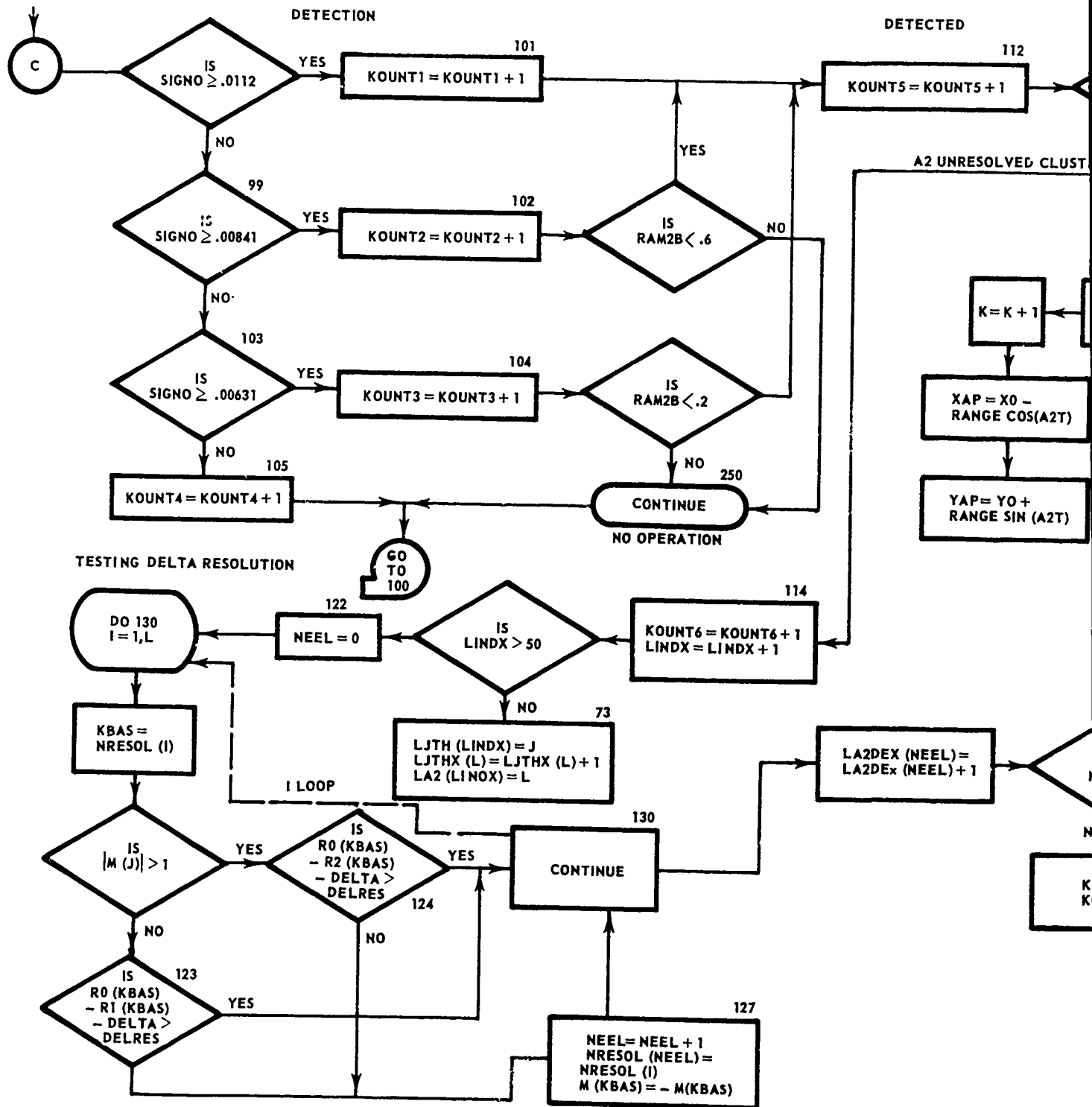


Figure 2-3 (Cont.)

FROM PAGE 3



A.

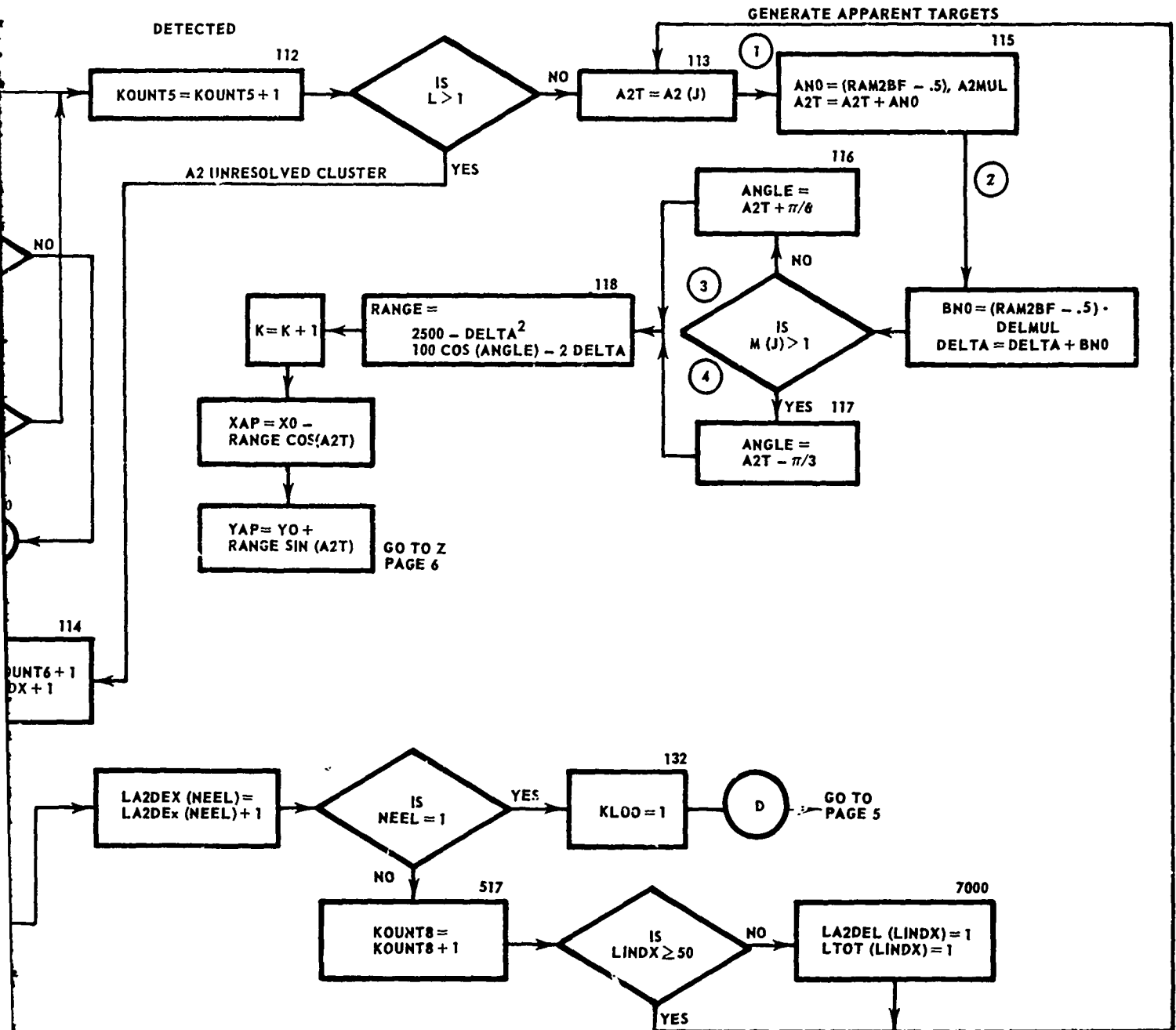
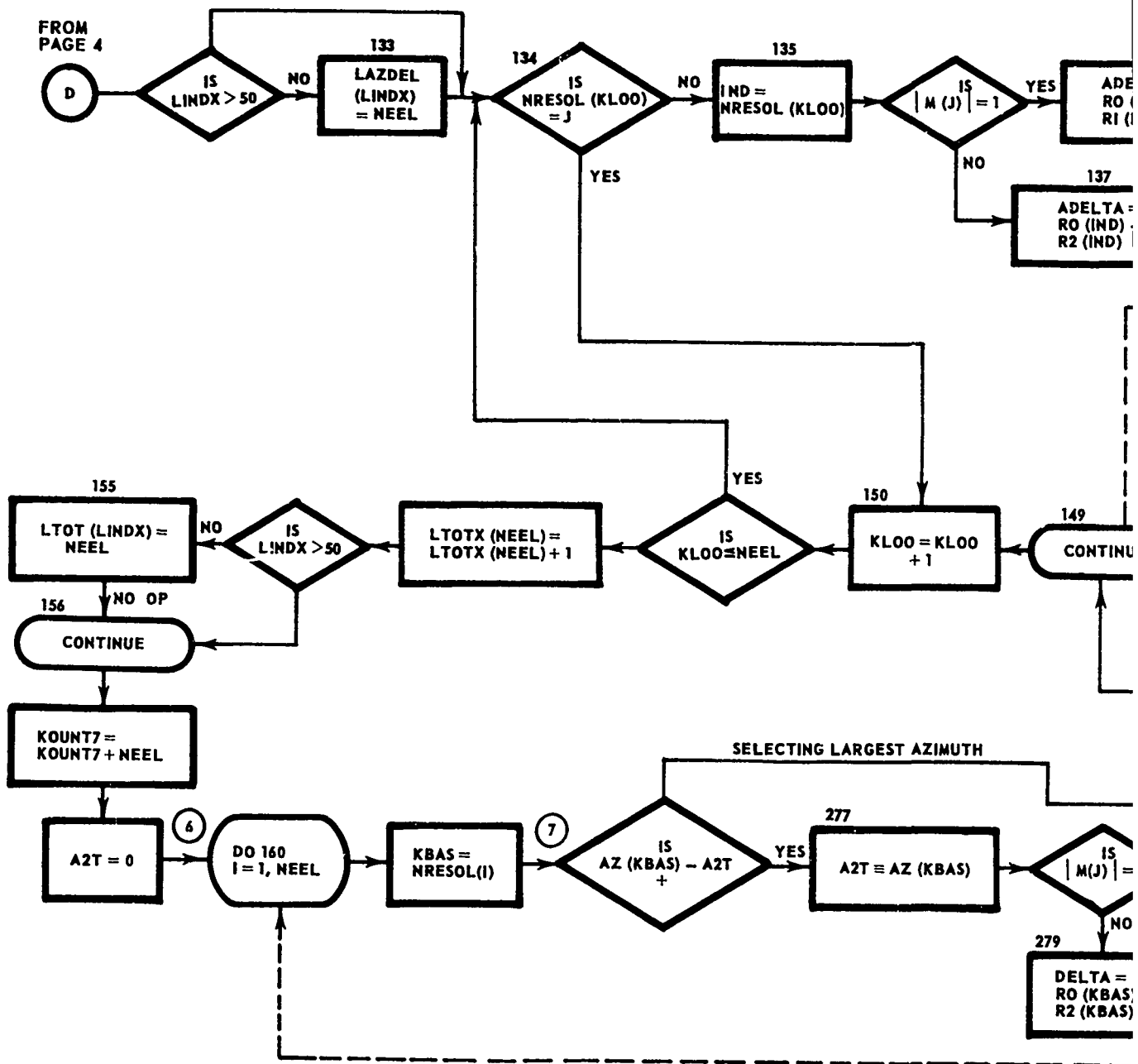


Figure 2-3 (Cont.)

B.



Figure

A.

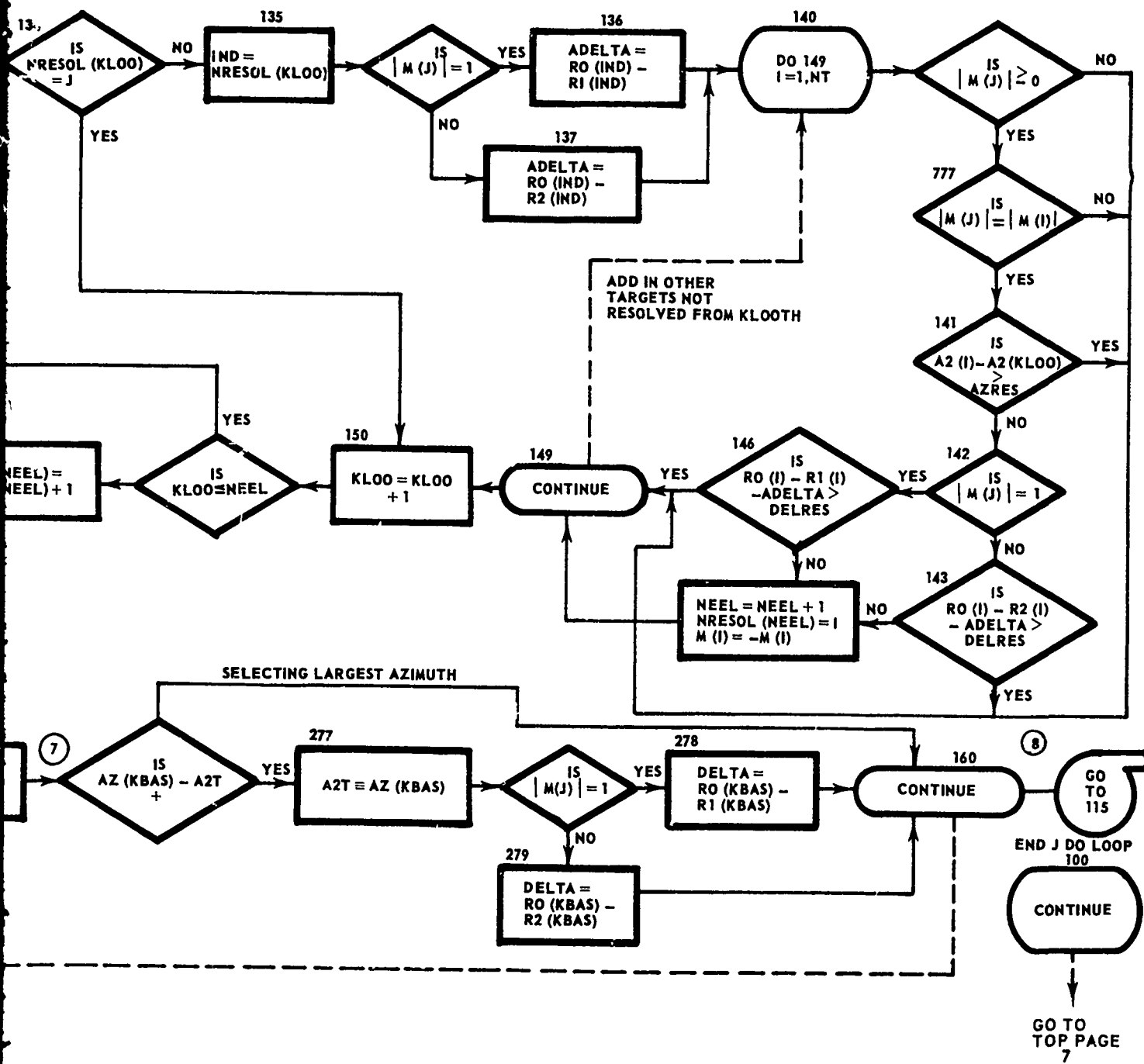
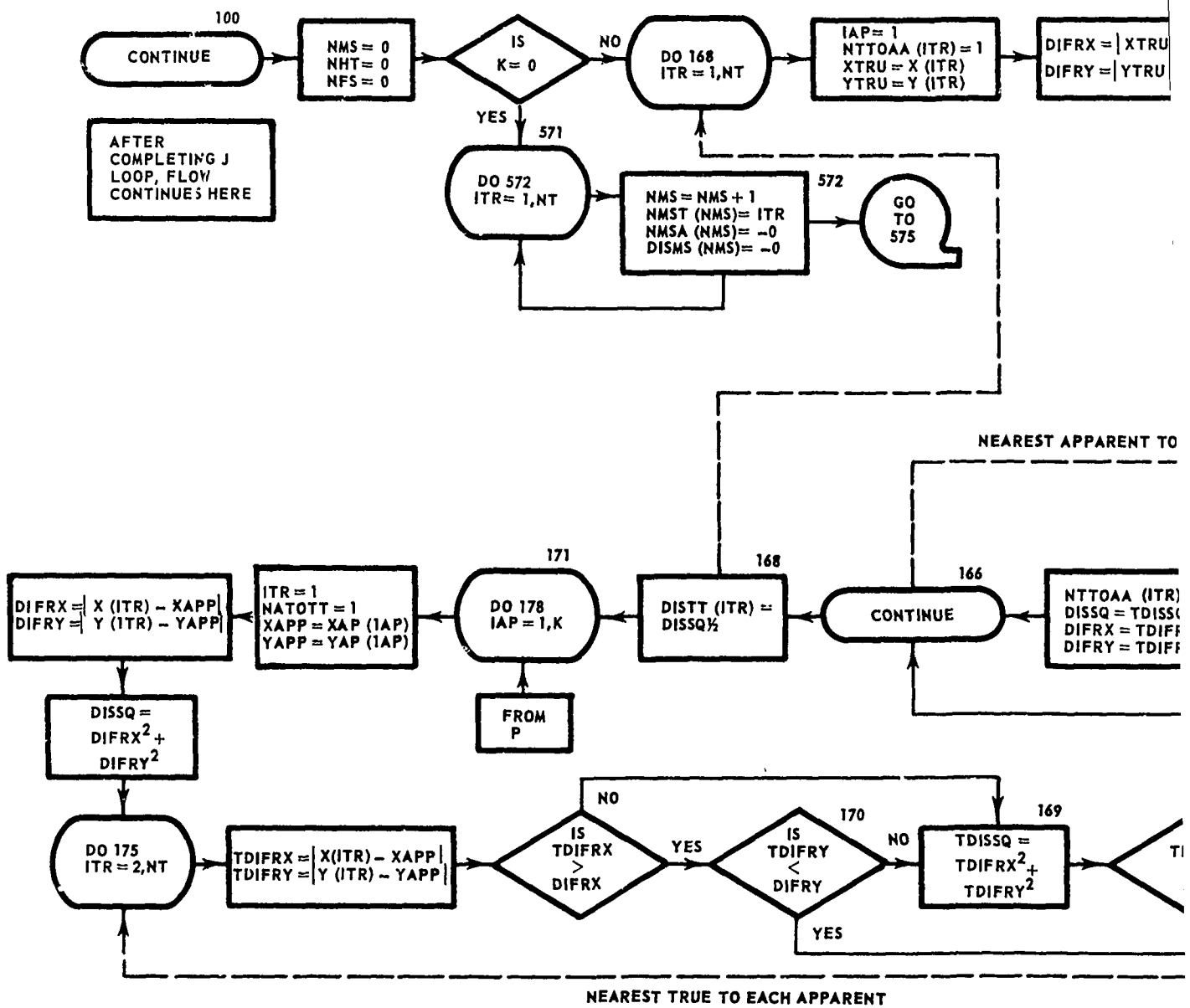


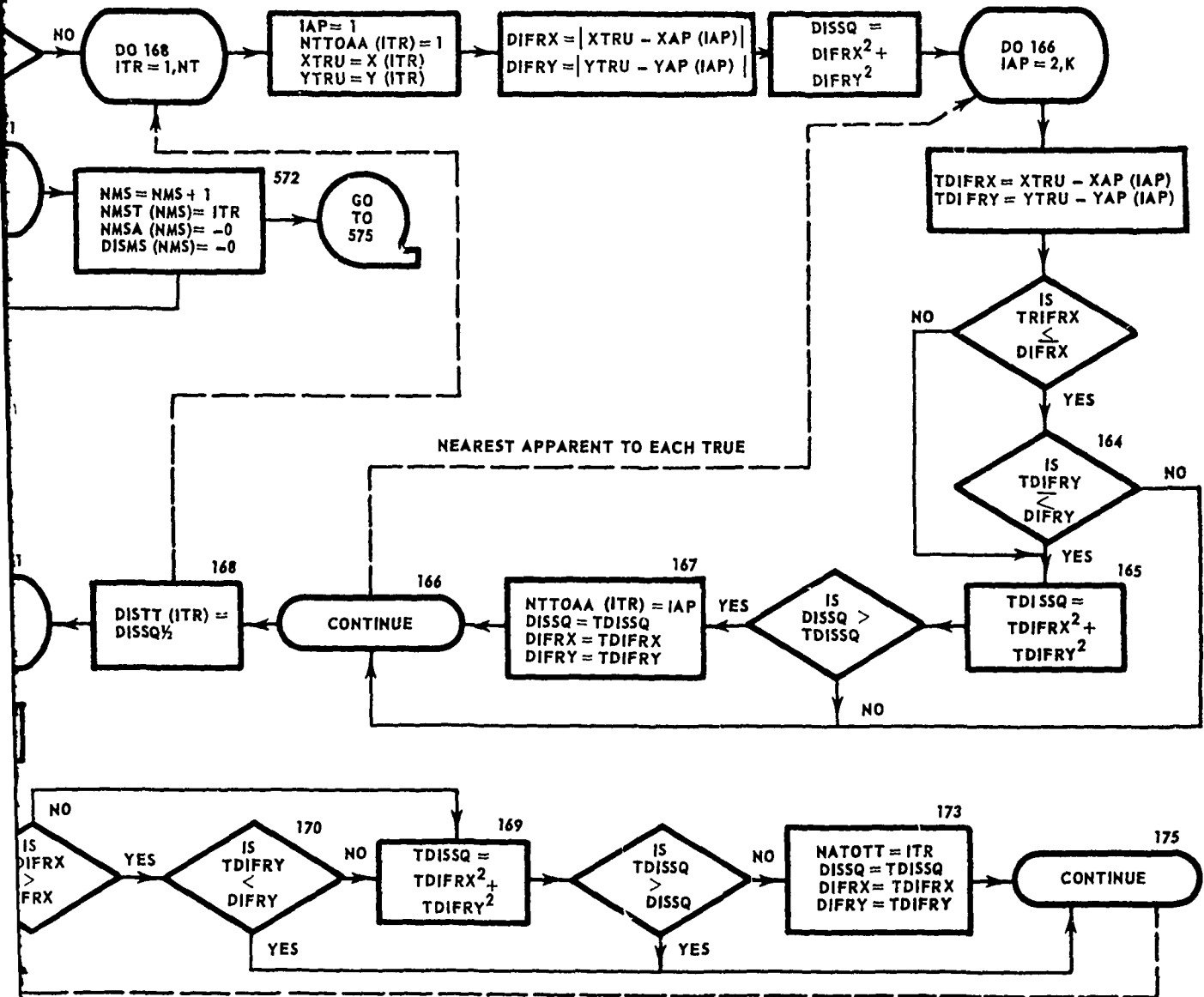
Figure 2-3 (Cont.)

B.



Fig

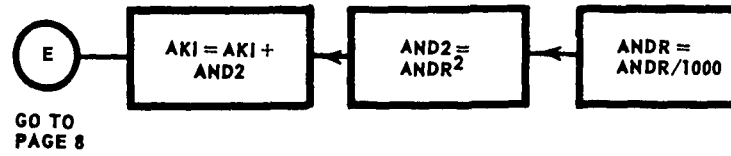
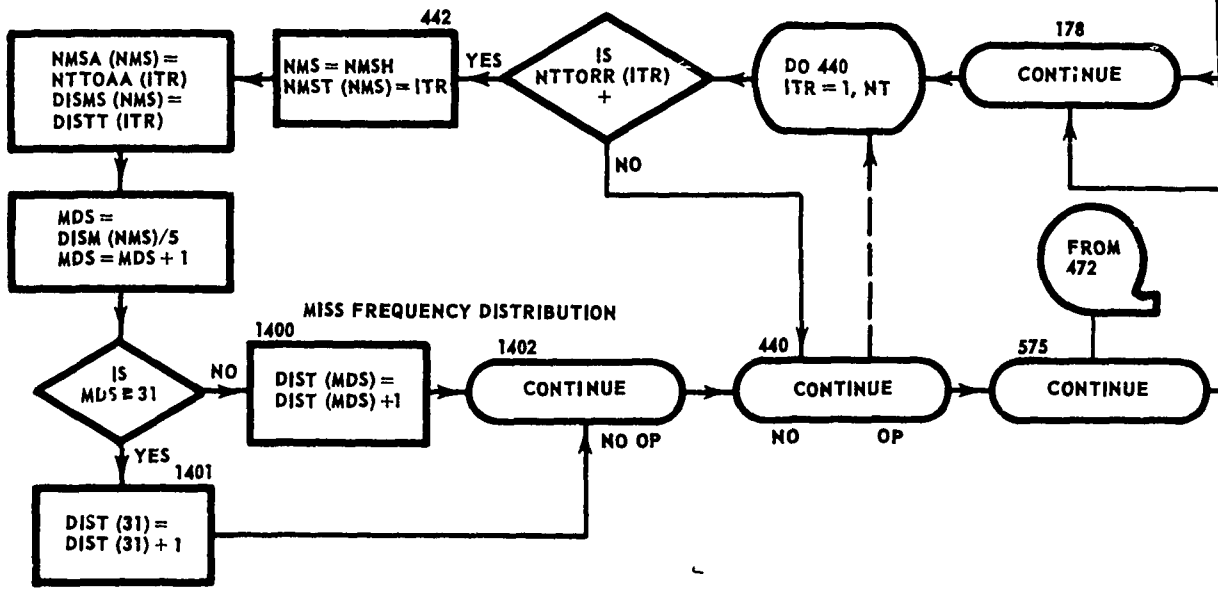
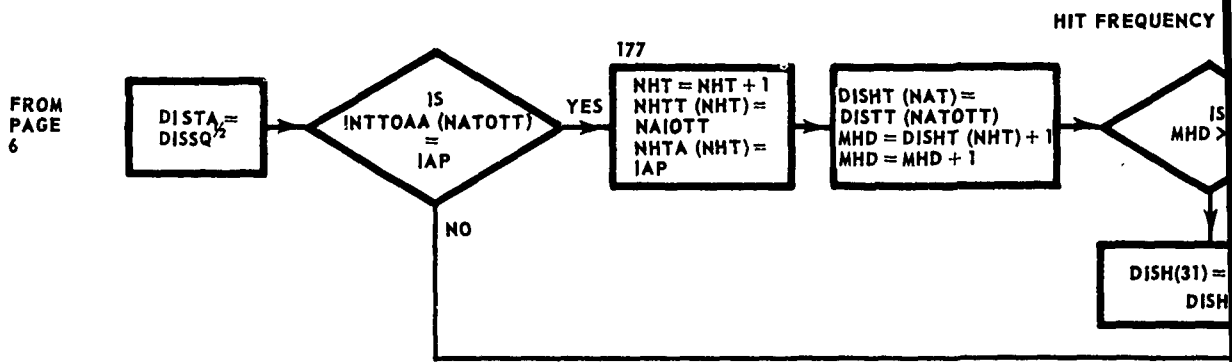
A.



GO TO PAGE 7

Figure 2-3 (Cont.)

B.



A.

HIT FREQUENCY DISTRIBUTION

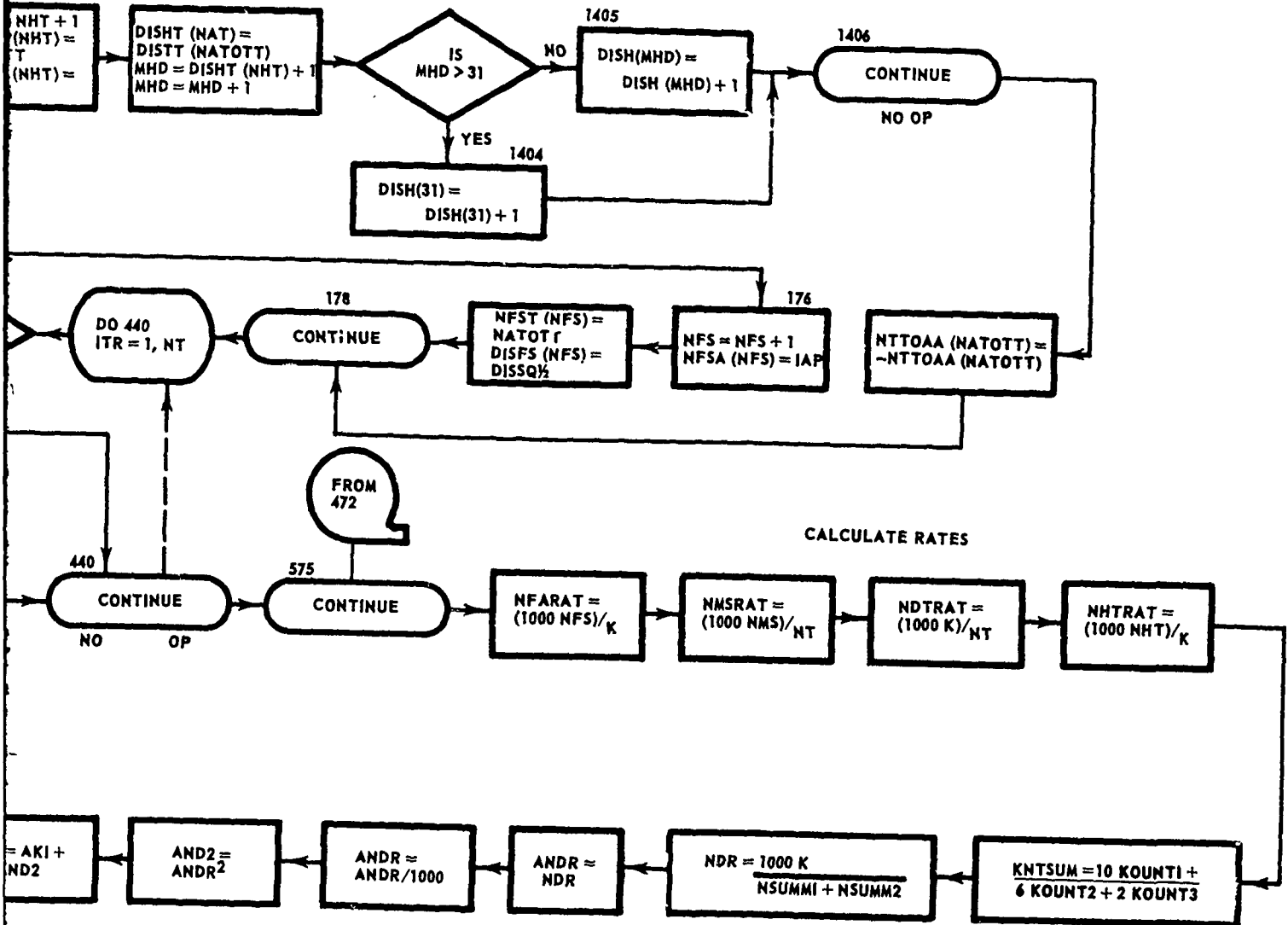


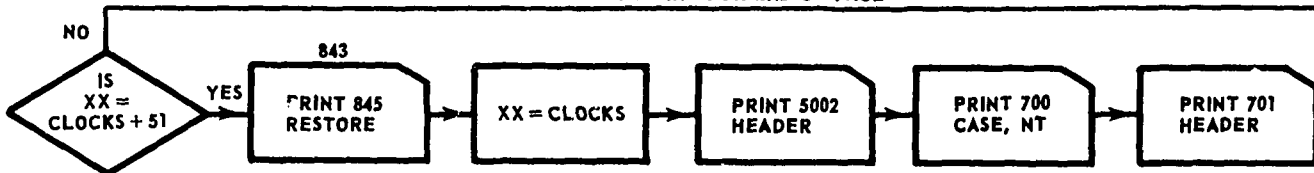
Figure 2-3 (Cont.)

B

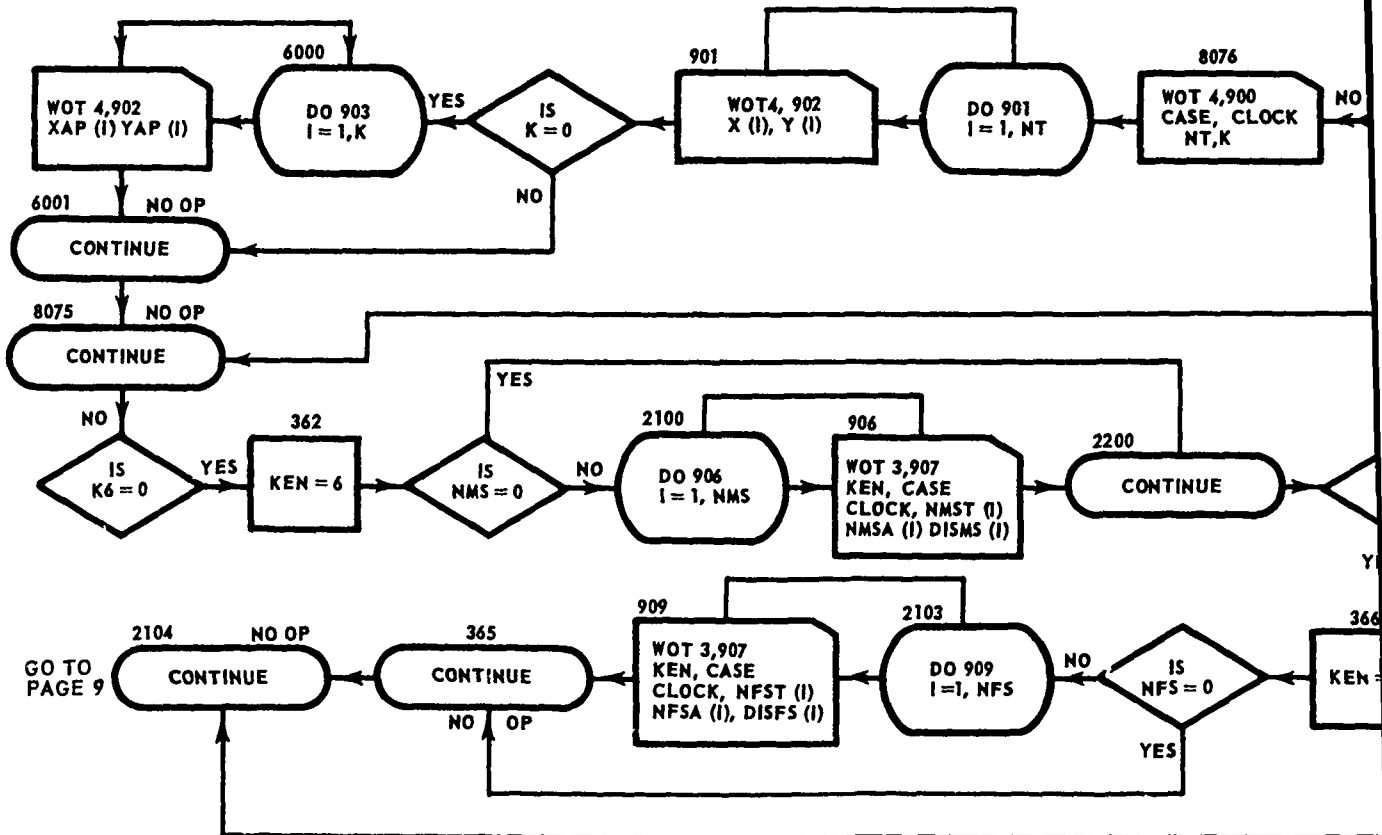
TESTING FOR END OF PAGE

FROM PAGE 7

E



WRITING SPECIAL TAPES



A.

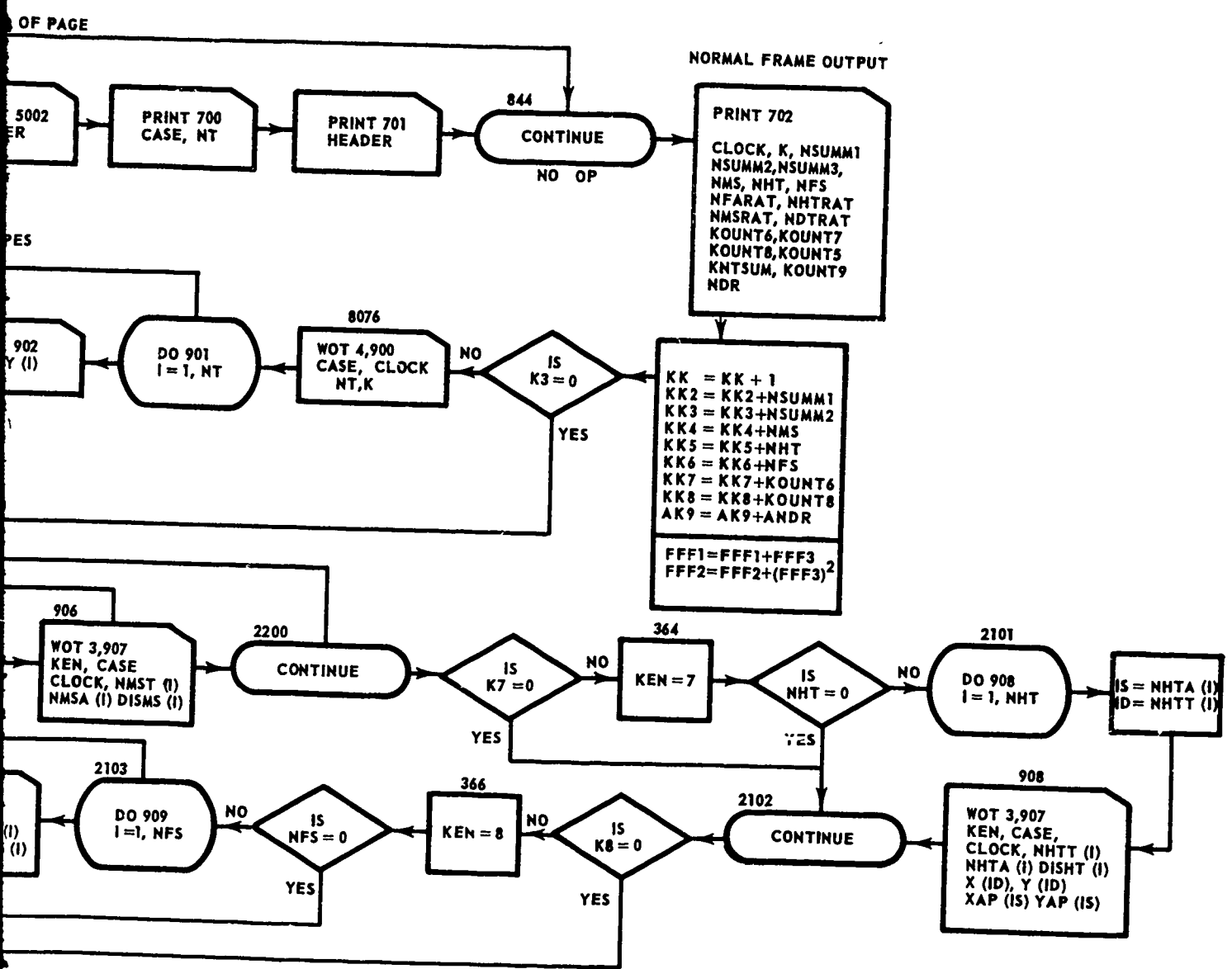
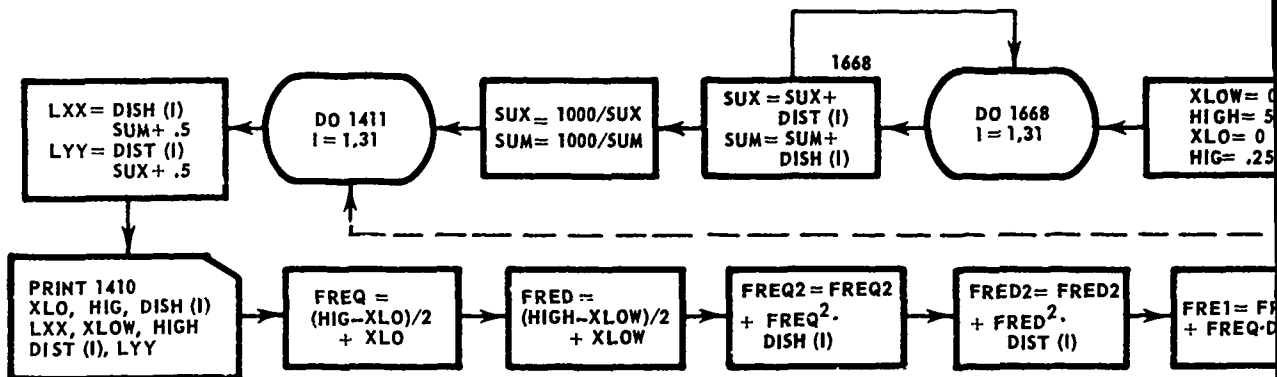
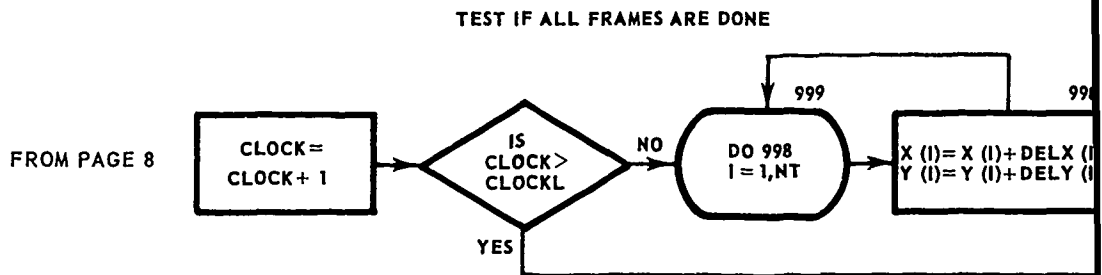


Figure 2-3 (Cont.)

B.



GO TO PAGE 10

A.

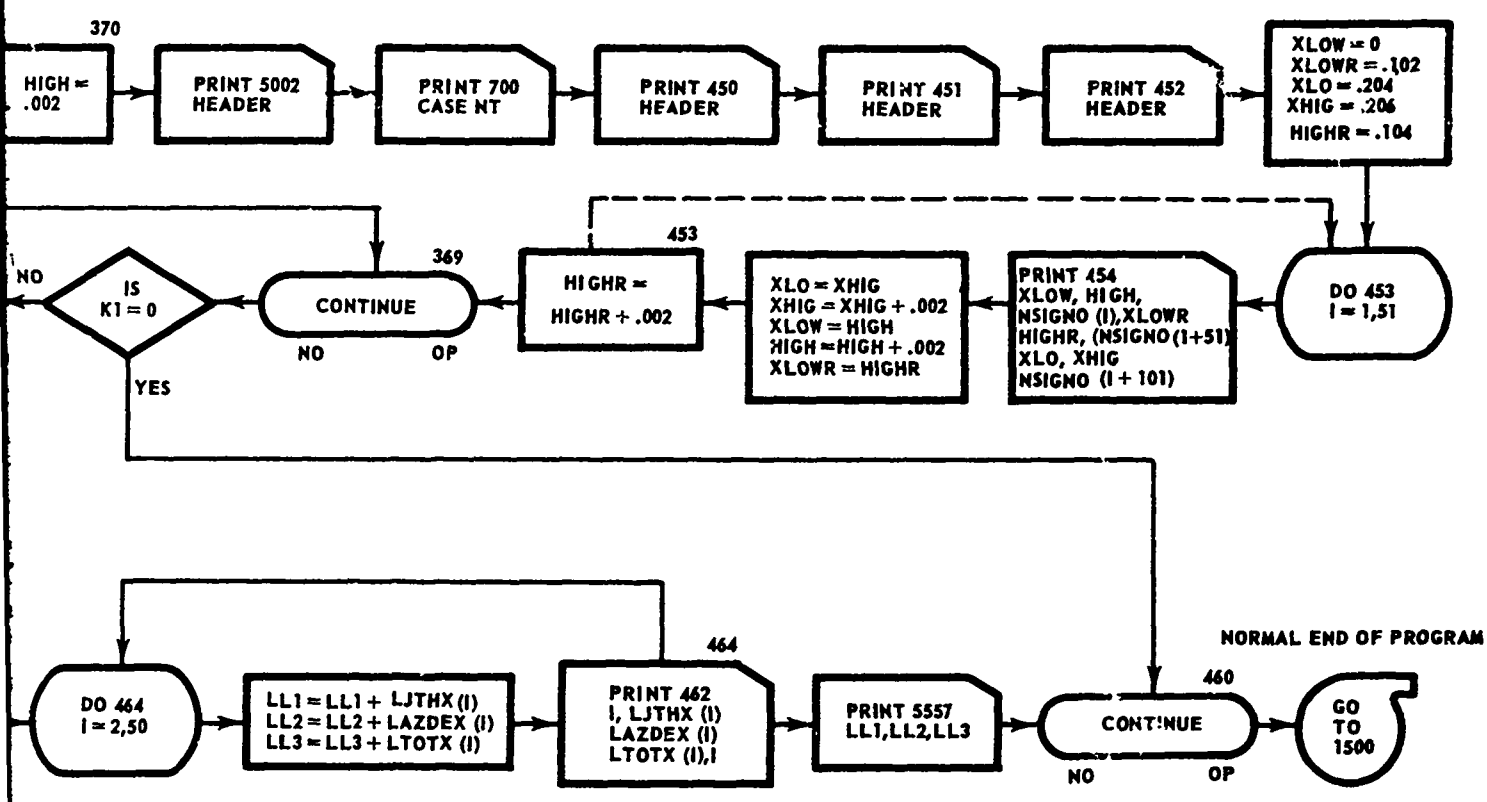


Figure 2-3 (Cont.)

B.

2.4 MODIFICATION TO OBTAIN MISS-DISTANCE

The function of this sub-program is to read one or more of the binary tapes produced by the continuity sub-program and to compute and make entries in the two-dimensional RAnge/AZimuth frequency table. This frequency table may be thought of as a ruled plane resembling a "checker board." The horizontal dimensions will be in terms of RAnge intervals while the vertical scaling will be in terms of AZimuth. Any particular block thus, represents a specific "closed" interval of range and azimuth. If each RA/AZ block is ruled so that it contains three separate lines for three specific entries, then each detected target will have range and azimuth values that will define the target into one of the RA/AZ blocks in the RA/AZ frequency table. The program examines the range and azimuth of each detected target and makes an entry into the first line of the corresponding block in the frequency table. This first line in each block, XOB SVN, is a count of the detected targets which appear in the defined RA/AZ block for a specific analysis. The remaining two lines contain the arithmetic mean of the summation and the standard deviation respectively of either of three particular variables depending on the input control constant. These "mean" and "deviation" values may be functions of one of the following:

1. The difference between the true and the apparent target in a "hit pair"
2. The difference between the ranges of the true and the apparent target in a "hit pair"
3. The difference between the azimuths of the true target and the apparent target in a "hit pair"

The precise content of "mean" and "deviation" is a function of the control constant, K_1 , which is read in at the start of each analysis.

The program reads the input that indicates which binary tape to process (CASE 2), the initial frame (START 2), and the end frame (END 2); it then initiates a "search tape 4 routine" to select the desired starting point on the input tape. Each individual detected target is examined and entries are made into each of the three respective lines in the corresponding RA/AZ block. Upon processing the frame information,

a test is made to determine if the last frame has been processed. If the last frame has not been processed, the program recycles back to read another frame from tape and processes this as described above. When the last frame has been processed, the program enters the "output sections." The RA/AZ frequency table (discussed previously) is printed on-line and punched onto cards.

2.4.1 Glossary

<u>AMAZ</u>	Arithmetic mean of the azimuth error in (radians)
<u>AMMD</u>	Arithmetic mean of miss distance in (nautical miles)
<u>AMRE</u>	Arithmetic mean of the range error in (nautical miles)
<u>AND</u>	The summation of the NDR's
<u>AND2</u>	The summation of the squares of NDR
<u>ANDR</u>	NDR expressed in floating point
<u>AZD</u>	The absolute difference in azimuth between a true target and its associated apparent
<u>AZTHER</u>	The summation of the azimuth error for a given Jth target
<u>AZTHSQ</u>	The summation of the squares of the azimuths for a given Jth target
<u>DIST</u>	The distance between a true and its apparent target
<u>DMISSQ</u>	The summation of the squares of the miss distance for a given Jth target
<u>DMISSR</u>	The summation of the miss distances for a given J th target

<u>FRQOBS</u>	Frequency of observation (i. e. , how many times the J th target was detected)
<u>MONTH</u>	The number of the month
<u>NDAY</u>	The day of the month
<u>NYEAR</u>	The last two digits of the year
<u>RNGER</u>	The summation of the range errors for a given Jth target
<u>RNGSQ</u>	The summation of the squares of the range errors for a given Jth target
<u>STDAZ</u>	The standard deviation of azimuth error
<u>STDMD</u>	The standard deviation of the miss distance
<u>STDRE</u>	The standard deviation of the range error
<u>VARAZ</u>	The variance of the azimuth error calculated by the formula
	$\zeta^2 = \frac{N \sum X^2 - (\sum X)^2}{N(N-1)}$
<u>VARMD</u>	The variance of the miss distance
<u>VARRE</u>	The variance of the range error

2.4.2 FORTRAN Listing

```

DIMENSION LJTH(50),LAZ(50),LAZDEL(50),LTOT(50)
DIMENSION X(100),Y(100),M(100),DIFX(100),DIFY(100),RO(100),R2(100)
1      ,DELY(100),AZ(100),NRESOL(100),FRQOBS(100),DMISSR(100),
2DMISSQ(100),RNGER(100),RNGSQ(100),AZTHER(100),AZTHSQ(100),R1(100)
DIMENSION DELX(100)
READ 5050,MONTH,NDAY,NYEAR
1500  DO 5333 I=1,100
      FRQOBS(I)=0.
      DMISSR(I)=0.
      DMISSQ(I)=0.
      RNGER(I)=0.

```

```

RNGSQ(I)=0.
AZTHER(I)=0.
5333 AZTHSQ(I)=0.
AND=0.
AND2=0.
5050 FORMAT(3I2)
PRINT 845,MONTH,NDAY,NYEAR
O=0.
READ 3,NT,KLOCKL,CASE,GAIN
3 FORMAT(2I5,2F10.1)
READ 400,AZMUL,DELMUL,AZRES,DELRES,RMAX,RMIN,RMIN2
400 FORMAT(7F10.1)
PRINT 5002
5002 FORMAT(78H AN/TLQ-8 SIMULATION MEASURES OF MISS DISTANCES
1 BSD PROGRAM BX2055)
PRINT 700,CASE,NT
700 FORMAT(11H CASE F5.0, 6H NT I5)
PRINT 1100,KLOCKL,AZMUL,DELMUL,AZRES,DELRES
1100 FORMAT(12H CLOCKLI4,7H AZMUL F9.6,7H DELMULF7.4,6H AZRESF9.6
1,7H DELRESF7.4)
PRINT 1101,GAIN,RMAX,RMIN,RMIN2
1101 FORMAT(10H GAINF7.4,5H RMAXF7.2,5H RMINF6.2,6H RMIN2F9.6)
PRINT 905
905 FORMAT(96H FRAME APPS NSUMM1 NSUMM2 NSUMM3 NDTRAT KOUNT6 KOU
INT7 KOUNT8 KOUNT5 KNTSUM KOUNT9 NDR )
READ 4,((X(I),Y(I),DELX(I),DELY(I)),I=1,NT)
4 FORMAT(4F10.2)
X0=500.
Y0=500.
X1=456.6985
X2=475.
Y1=475.
Y2=543.3015
KK=0
XX=0.
KK2=0

```

FORTRAN LISTING PAGE 2

```

KK3=0
KK7=0
KK8=0
KK9=0
KLOCK=0
7 KOUNT1=0
KOUNT2=0
KOUNT3=0
KOUNT4=0
KOUNT5=0
KOUNT6=0

```

```

KOUNT8=0
KOUNT9=0
NSUMM1=0
NSUMM2=0
NSUMM3=0
SUMLO=0.
SUMO=0.
SUML1=0.
SUML2=0.
DO 50 I=1,NT
DIFX(I)=X(I)-XO
DIFY(I)=Y(I)-YO
RO(I)=SQRTF(DIFX(I)**2+DIFY(I)**2)
IF(RO(I)-RMAX) 9,9,8
9 IF(RO(I)-RMIN) 500,500,501
500 SUMLO=SUMLO+RMIN2
GO TO 502
501 SUMLO=SUMLO+1./RO(I)**2
502 SUMO=SUMO+1./RO(I)**2
8 R1(I)=SQRTF((X(I)-X1)**2+(Y(I)-Y1)**2)
IF(R1(I)-RMAX) 11,11,650
11 IF(R1(I)-RMIN) 652,651,651
651 SUML1=SUML1+1./R1(I)**2
GO TO 650
652 SUML1=SUML1+RMIN2
650 CONTINUE
13 R2(I)=SQRTF((X(I)-X2)**2+(Y(I)-Y2)**2)
IF(R2(I)-RMAX) 15,15,660
15 IF(R2(I)-RMIN) 662,661,661
661 SUML2=SUML2+1./R2(I)**2
GO TO 660
662 SUML2=SUML2+RMIN2
660 CONTINUE
IF(DIFY(I)) 18,18,19
FORTRAN LISTING PAGE 3

18 M(I)=3
NSUMM3=NSUMM3+1
GO TO 50
19 IF(RO(I)-RMAX) 40,40,22
40 IF(DIFX(I))20,23,23
20 IF(R1(I)-RMAX) 21,21,22
22 M(I)=4
GO TO 50
21 M(I)=1
NSUMM1=NSUMM1+1
GO TO 26
23 IF(R2(I)-RMAX) 25,25,22

```

```

25   M(I)=2
      NSUMM2=NSUMM2+1
26   AZ(I)=ATN1F(DIFY(I),-DIFX(I))
50   CONTINUE
      DO 56 I=1,50
          LJTH(I)=0
          LAZ(I)=0
          LAZDEL(I)=0
56   LTOT(I)=0
      K=0
      LINDX=0
      DO 100 J=1,NT
          IF(M(J)) 100,58,58
58   IF(XABSF(M(J))-2) 60,60,100
60   L=0
      DO 76 I=1,NT
          IF(XABSF(M(I))-XABSF(M(J))) 76,62,76
62  IF((ABSF(AZ(J)-AZ(I)))-AZRES) 64,64,76
64  L=L+1
      NRESOL(L)=I
76  CONTINUE
72  SIDLO=SUMO
      AMB=0.
      AMBL=0.
74  IF(XABSF(M(J))-1) 75,71,75
75  RADOMA=SUML2
      GO TO 77
71  RADOMA=SUML1
77  DO 88 I=1,L
          KBAS=NRESOL(I)
          FRACT=1./RO(KBAS)**2
          FLJ=1.-ABSF((AZ(J)-AZ(KBAS))/AZRES)
          AMB=AMB+FLJ*FRACT
          IF(FRACT-RMIN2) 275,275,274
              FORTRAN LISTING PAGE 4

274  FRACT=RMIN2
275  AMBL=AMBL+FLJ*FRACT
          IF(J-KBAS) 88,276,88
276  TARGML=FRACT
          IF(XABSF(M(J))-1) 83,85,83
83  IF(R2(KBAS)-RMIN) 680,680,681
680  RANGMA=RMIN**2
          GO TO 682
681  RANGMA=R2(KBAS)**2
682  CONTINUE
          DELTA=RO(J)-R2(J)
          GO TO 88
85  IF(R1(KBAS)-RMIN) 690,690,691

```

```

690 RANGMA=RMIN**2
GO TO 692
691 RANGMA=R1(KBAS)**2
692 CONTINUE
DELTA=RO(J)-R1(J)
88 CONTINUE
IF(AMBL-(SUMO-AMB)*GAIN)4000,89,89
4000 KOUNT9=KOUNT9+1
GO TO 100
89 AOSIGO=((SUMO-AMB)*GAIN+AMBL-TARGML)/TARGML
AOSIGA=(RANGMA*RADOMA)-1.
SIGNO=1./(2.+AOSIGO+AOSIGA+AOSIGO*AOSIGA)
IF(SIGNO-.0112) 99,101,101
101 KOUNT1=KOUNT1+1
GO TO 112
99 IF(SIGNO-.00841) 103,102,102
102 KOUNT2=KOUNT2+1
IF(RAM2BF(0)-.6) 112,250,250
103 IF(SIGNO-.00631) 105,104,104
104 KOUNT3=KOUNT3+1
IF(RAM2BF(0)-.2) 112,250,250
105 KOUNT4=KOUNT4+1
250 CONTINUE
GO TO 100
112 KOUNT5=KOUNT5+1
IF(L-1) 113,113,114
114 KOUNT6=KOUNT6+1
LINDX=LINDX+1
IF(LINDX-50) 73,73,122
73 LJTH(LINDX) =J
LAZ(LINDX) =L
GO TO 122
113 A2T=AZ (J)
FORTRAN LISTING PAGE 5
115 ANO=(RAM2BF(0)-.5)*AZMUL
A2T=A2T+ANO
BNO=(RAM2BF(0)-.5)*DELMUL
DELTA=DELTA+ BNO
IF(XABS(M(J))-1) 116,116,117
116 ANGLE=A2T+.5235988
GO TO 118
117 ANGLE=A2T-1.0471976
118 RANGE=(2500.-DELTA**2)/(100.*COSF(ANGLE)-2.*DELTA)
1298 CONTINUE
K=K+1
120 XAP =XO-RANGE*COSF(A2T)
121 YAP =YO+RANGE*SINF(A2T)
IF(SENSE SWITCH 2 ) 1666,1667
1666 WRITE OUTPUT TAPE 4,1668,J,X(J) ,Y(J),RO(J),AZ(J),XAP,YAP,RANGE,A2

```

```

1T
1668 F( R MAT( I5,8E14.5)
1667 CONTINUE
      FRQOBS(J)= FRQOBS(J)+1.
      DIST= SQRTF((X(J)-XAP  )**2+(Y(J)-YAP  )**2)
      DMISSQ(J)= DMISSQ(J) + DIST**2
      DMISSR(J)= DMISSR(J) + DIST
      RD= ABSF(RO(J)-RANGE)
      RNGER(J)=RNGER(J)+RD
      RNGSQ(J)=RNGSQ(J)+RD**2
      AZD= ABSF(AZ(J)-A2T)
      AZTHER(J)=AZTHER(J)+AZD
      AZTHSQ(J)=AZTHSQ(J)+AZD**2
      IF(SENSE SWITCH 1) 1400,1401
1400 PUNCH 912,CASE,KLOCK,J,RO(J),AZ(J),DIST,RD,AZD
1401 CONTINUE
912  FORMAT(F8.0,I5,I4,F10.3,F10.6,F11.5,F11.5,F10.6)
      GO TO 100
122  NEEL=0
      DO 130 I=1,L
      KBAS=NRESOL(I)
      IF(XABSF(M(J))-1) 123,123,124
124  IF(ABSF(RO(KBAS)-R2(KBAS)-DELTA)-DELRES) 127,127,130
123  IF(ABSF(RO(KBAS)-R1(KBAS)-DELTA)-DELRES) 127,127,130
127  NEEL=NEEL+1
      M(KBAS)=-M(KBAS)
      NRESOL(NEEL)=NRESOL(I)
130  CONTINUE
      IF(NEEL-1) 517,517,132
517  KOUNT8=KOUNT8+1
      IF(LINDX-50)7000,7000,113
          FORTRAN LISTING PAGE 6

7000 LAZDEL(LINDX)=1
      LTOT(LINDX)=1
      GO TO 113
132  KLOO=1
      IF(LINDX-50)133,133,134
133  LAZDEL(LINDX)=NEEL
134  IF(NRESOL(KLOO)-J) 135,150,135
135  IND=NRESOL(KLOO)
      IF(XABSF(M(J))-1) 136,136,137
136  ADELTA=RO(IND)-R1(IND)
      GO TO 140
137  ADELTA=RO(IND)-R2(IND)
140  DO 149 I=1,NT
      IF(M(I)) 149,777,777
777  IF(XABSF(M(I))-XABSF(M(J)))149,141,149

```

```

141   IF(ABSF(AZ(I)-AZ(KLOO))-AZRES) 142,142,149
142   IF(XABSF(M(J))-1) 146,146,143
146   IF(ABSF(R0(I)-R1(I)-ADELTA)-DELRES) 144,144,149
144   NEEL=NEEL+1
      NRESOL(NEEL)=1
      M(I)=-M(I)
      GO TO 149
143   IF(ABSF(R0(I)-R2(I)-ADELTA)-DELRES) 144,144,149
149   CONTINUE
150   KLOO=KLOO+1
      IF(KLOO-NEEL) 134,134,152
152   IF(LINDX-50)155,155,156
155   LTOT(LINDX)=NEEL
156   SUMAZ=0.
      DENOM=NEEL
      KOOUNT7=KOOUNT7+NEEL
      A2T=0.
      DO 160 I=1,NEEL
      KBAS=NRESOL(I)
      IF(AZ(KBAS)-A2T) 160,160,277
277   A2T=AZ(KBAS)
      IF(XABSF(M(J))-1)278,278,279
278   DELTA=R0(KBAS)-R1(KBAS)
      GO TO 160
279   DELTA=R0(KBAS)-R2(KBAS)
160   CONTINUE
      GO TO 115
100   CONTINUE
      NDTRAT=(1000*K)/NT
      KNTSUM=10*KOOUNT1+6*KOOUNT2+2*KOOUNT3
      NDR=(1000*K)/(NSUMM1+NSUMM2)
      FORTRAN LISTING PAGE 7

      XX=XX+1.
      IF(XX-52.) 571,571,570
570   PRINT 845,MONTH,NDAY,NYEAR
      PRINT905
      XX=0.
571   PRINT 906,KLOCK,K,NSUMM1,NSUMM2,NSUMM3,NDTRAT,KOOUNT6,KOOUNT7,KOOUNT8
1, KOOUNT5,KNTSUM,KOOUNT9,NDR
906   FORMAT(I6,4I7,7I8,I6)
      KK=KK+K
      KK2=KK2+NSUMM1
      KK3=KK3+NSUMM2
      KK7=KK7+KOOUNT6
      KK8=KK8+KOOUNT8
      KK9=KK9+KOOUNT9
      ANDR=NDR
      ANDR=ANDR/1000.
      AND2=AND2+ANDR**2

```

```

AND=AND+ANDR
KLOCK=KLOCK+1
IF (KLOCK-KLOCKL) 949,949,950
949 DO 900 I=1,NT
X(I)=X(I)+DELX(I)
900 Y(I)=Y(I)+DELY(I)
GO TO 7
950 PRINT 907, KK, KK2, KK3, KK7, KK8, KK9, AND
907 FORMAT(7H TOTALS I6, 2I7, I23, I16, I24, F9.3)
PRINT 914, AND2
914 FORMAT(90H
1 SUM OF SQUARES NDR F11.6)
PRINT 845, MONTH, NDAY, NYEAR
PRINT 908
908 FORMAT(95H MISS DISTANCE RANG
1E ERROR AZIMUTH ERROR)
PRINT 909
909 FORMAT(101H TARGET FREQ OBS MEAN ST DEV VAR MEAN
1 ST DEV VAR MEAN ST DEV VAR)
DO 901 I=1,NT
AMMD= DMISSR(I)/FRQOBS(I)
VARMD=(FRQOBS(I)*DMISSQ(I)-DMISSR(I)**2)/(FRQOBS(I)*(FRQOBS(I)-1.))
1)
STDMD=SQRTF(VARMD)
AMRE= RNGER(I)/FRQOBS(I)
VARRE=(FRQOBS(I)*RNGSQ(I)-RNGER(I)**2)/(FRQOBS(I)*(FRQOBS(I)-1.))
STDRE=SQRTF(VARRE)
AMAZ=AZTHER(I)/FRQOBS(I)
VARAZ=(FRQOBS(I)*AZTHSQ(I)-AZTHER(I)**2)/(FRQOBS(I)*(FRQOBS(I)-1.))
FORTRAN LISTING PAGE 8
1)
STDAZ=SQRTF(VARAZ)
PRINT 911, I, FRQOBS(I), AMMD, STDMD, VARMD, AMRE, STDRE, VARRE, AMAZ, STDAZ
1, VARAZ
PUNCH 913, CASE, I, RO(I), AZ(I), FRQOBS(I), DMISSR(I), DMISSQ(I), RNGER(
I), RNGSQ(I), AZTHER(I), AZTHSQ(I)
901 CONTINUE
PRINT 959
PAUSE 1
GO TO 1500
959 FORMAT(83H NOTE MEAN AND ST DEV OF AZIMUTH ERROR ARE MULTIPLIED B
1Y 100 AND VARIANCE. BY 1000 )
845 FORMAT(108H1
1 DATE RUN I3, 1H/I2, 1H/I
22)
911 FORMAT(I5, F10.0, F11.3, F10.3, F9.3, F11.3, F10.3, F9.3, 2P2F10.6, 3PF9.6)
913 FORMAT(F6.0, I3, F6.1, F7.4, F4.0, F7.1, F7.1, F9.4, F8.2, 1PF7.4, 2PF8.4)
END(0, 1, 0, 0, 1)

```

FLOW DIAGRAM ERROR MEASURES

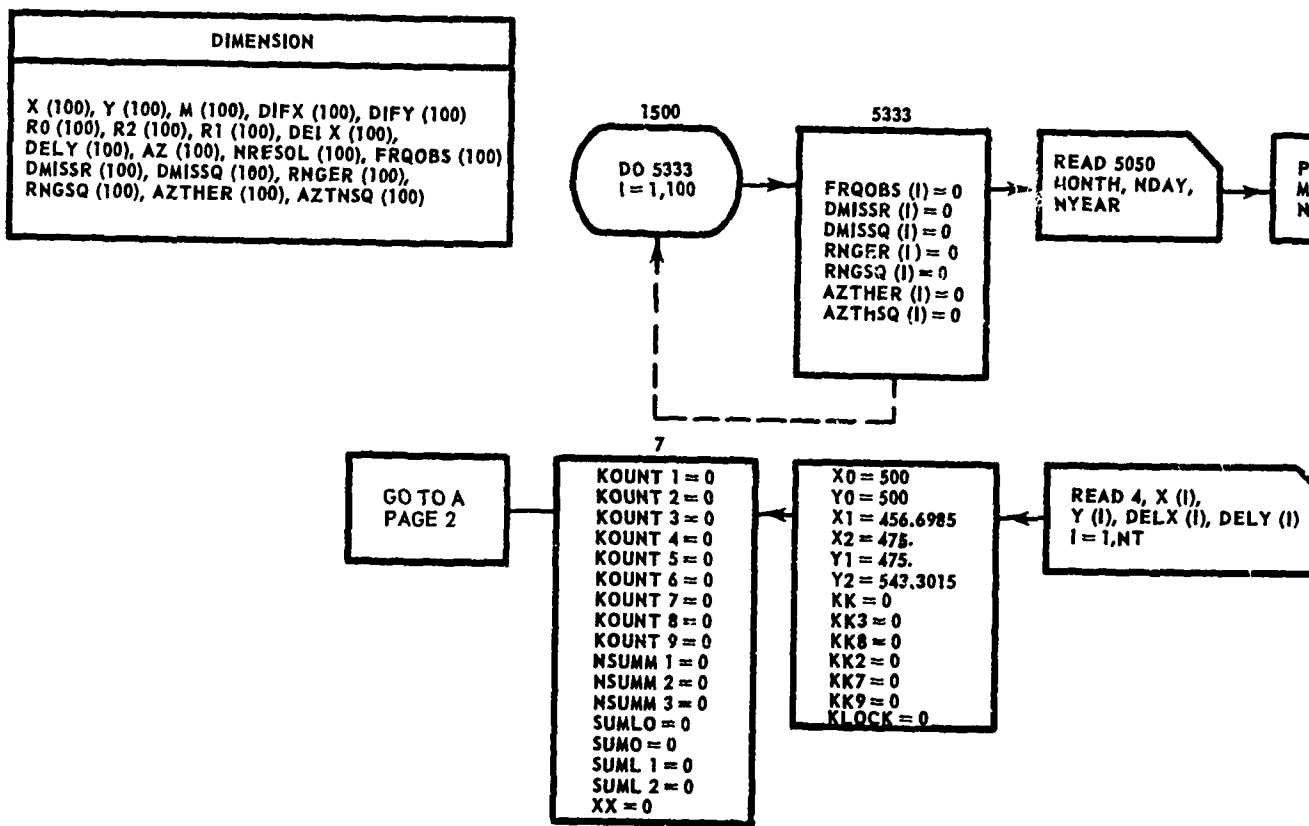


Figure 2-4 Flow Diagram

A.

FLOW DIAGRAM ERROR MEASURES

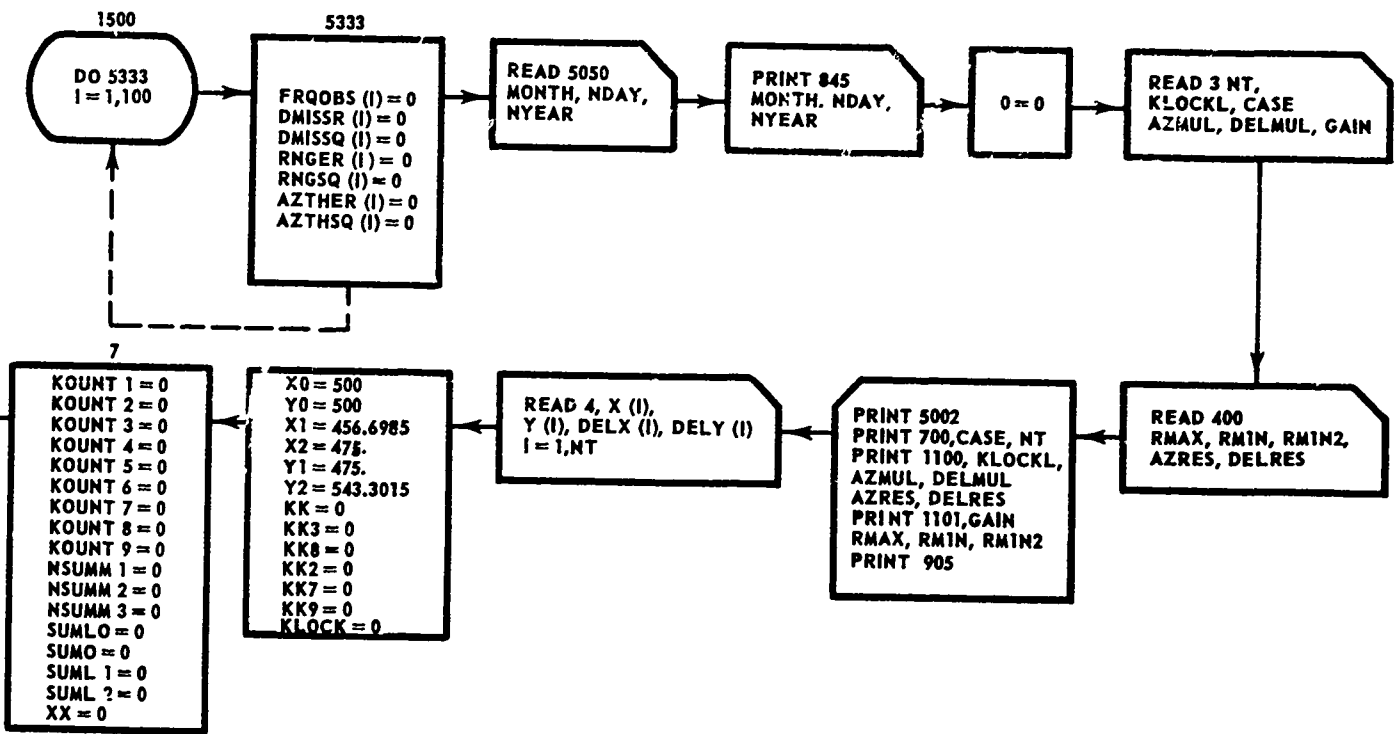
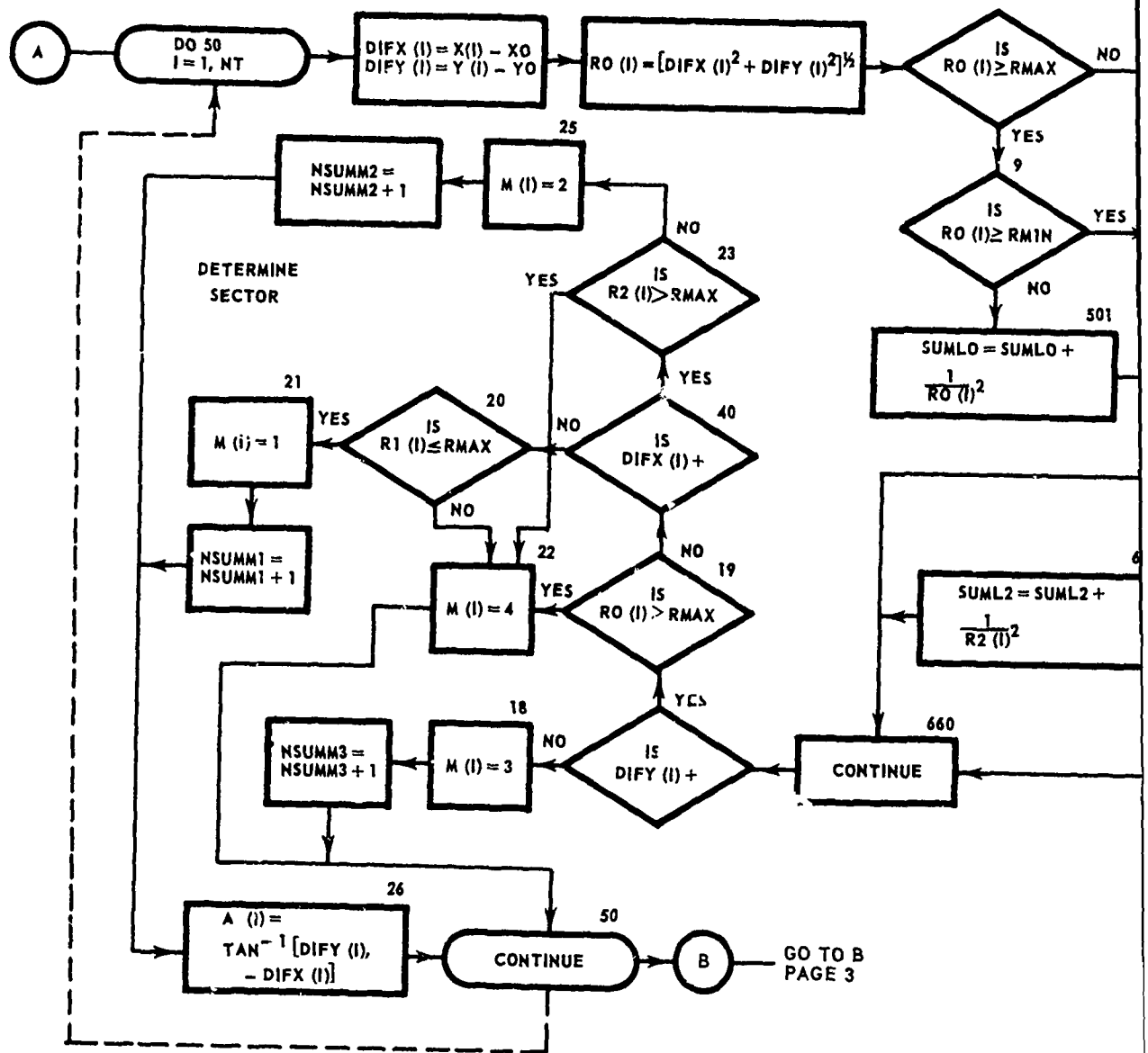


Figure 2-4 Flow Diagram Error Measures

B.

FROM A
PAGE 1



A.

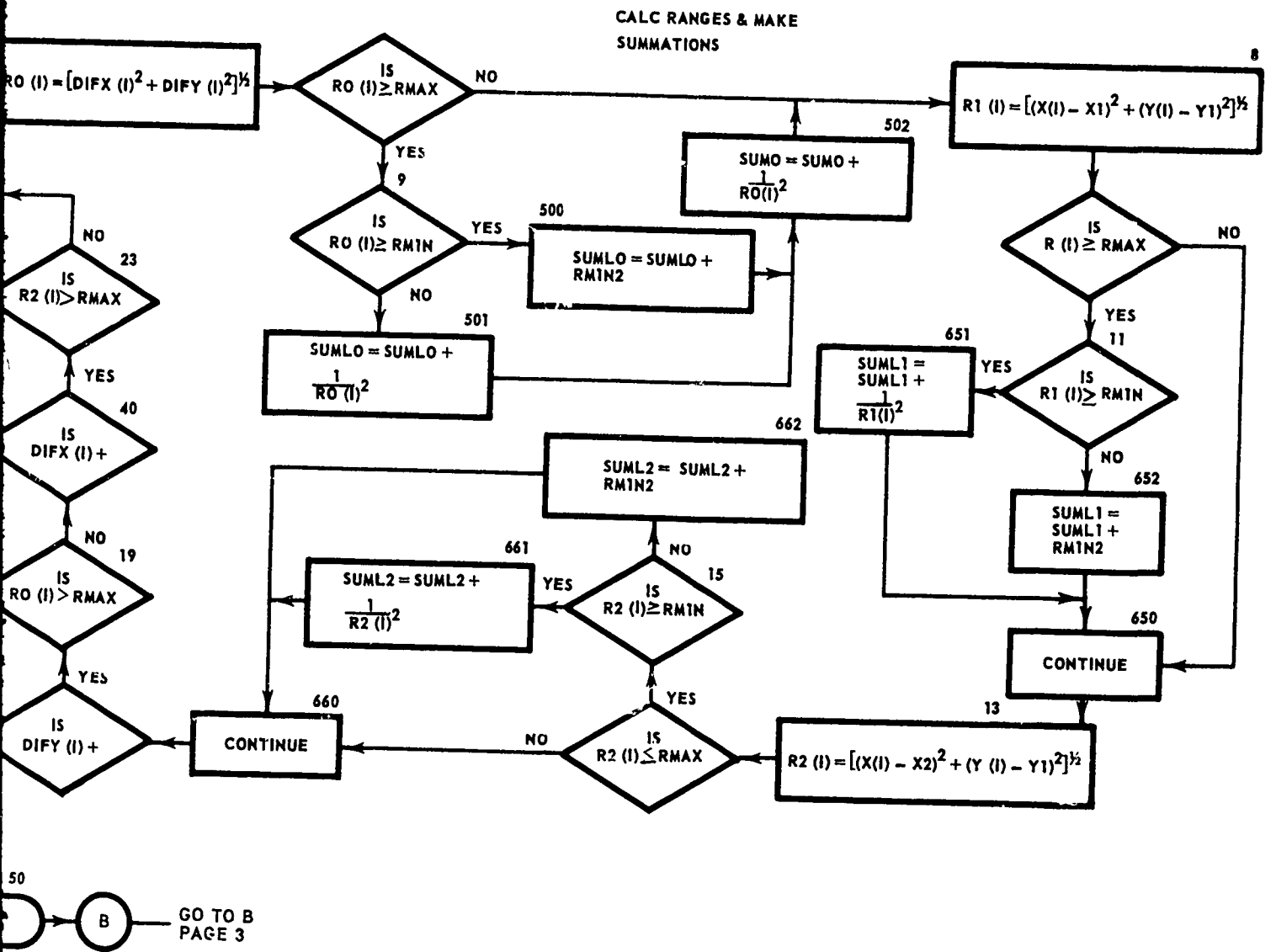
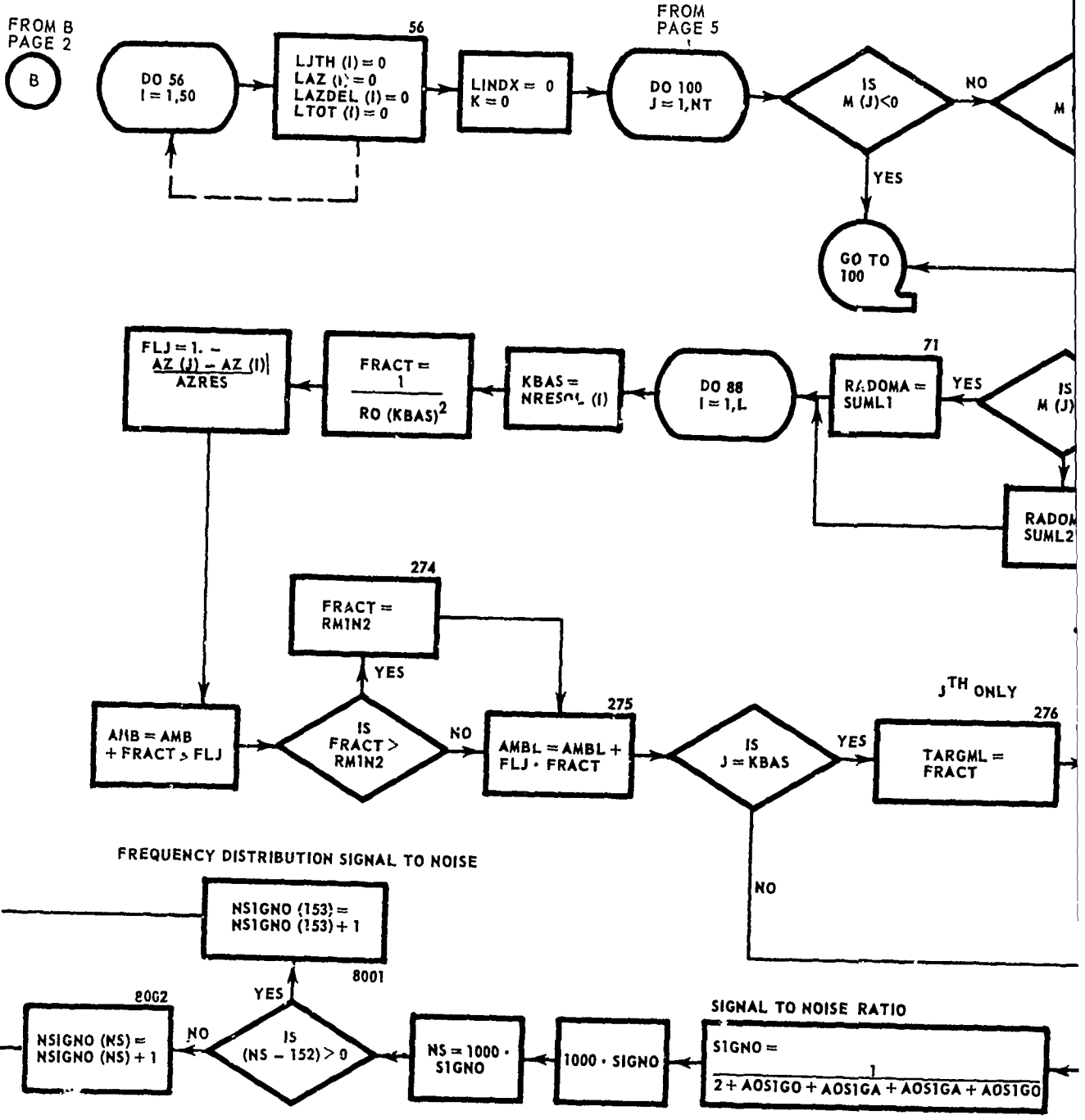


Figure 2-4 (Cont.)

B.



A.

FROM PAGE 5

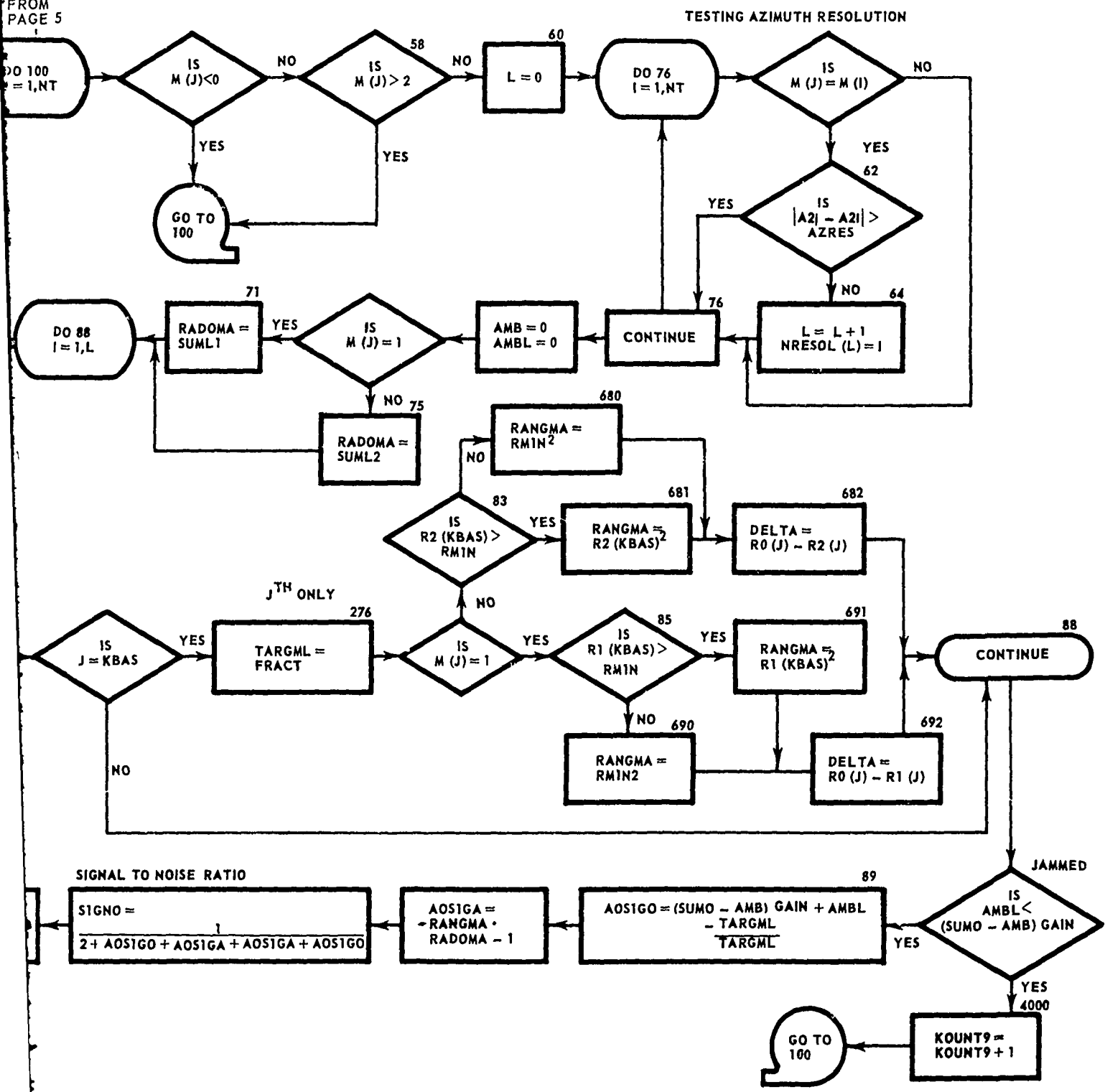


Figure 2-4 (Cont.)

B.

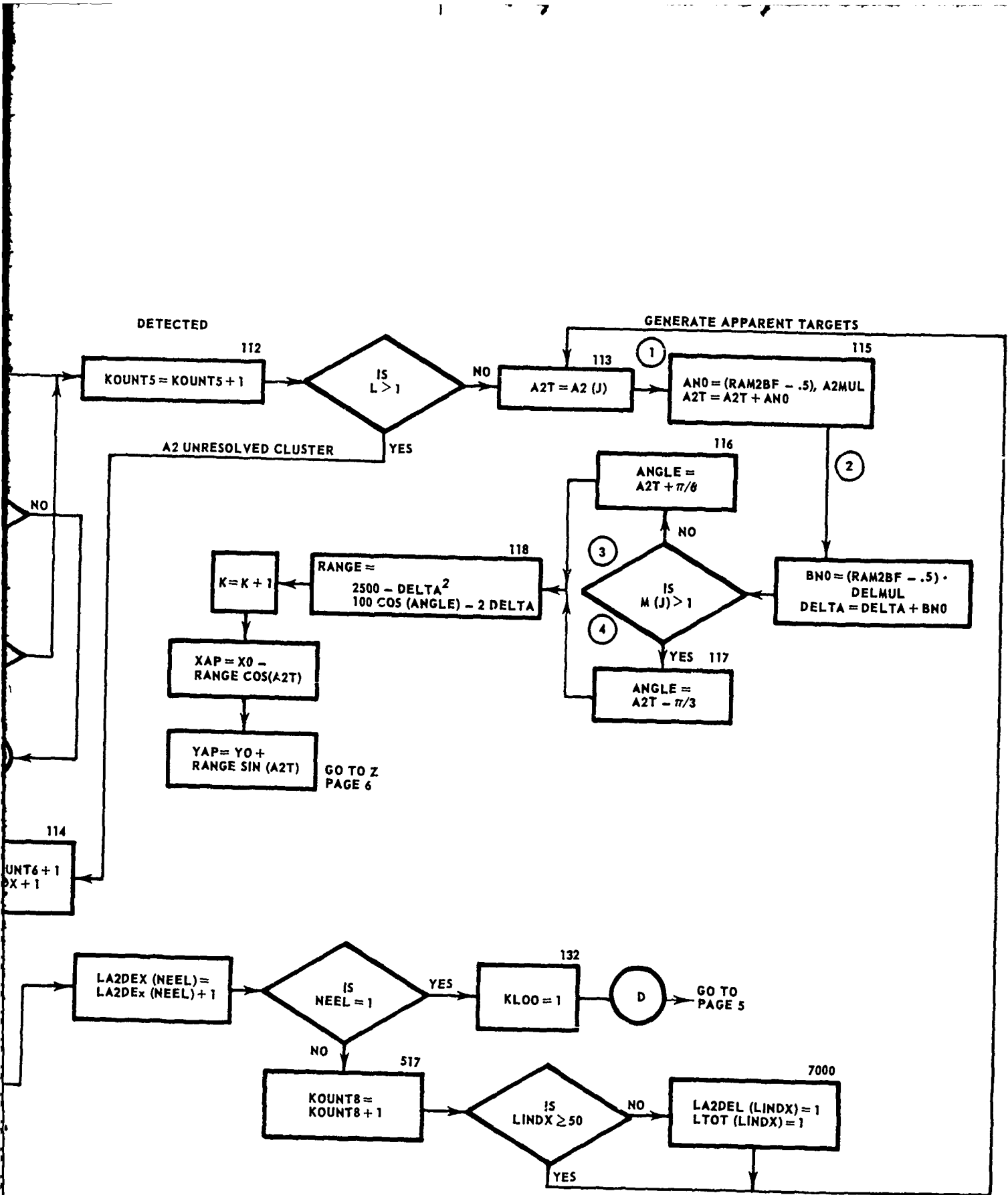
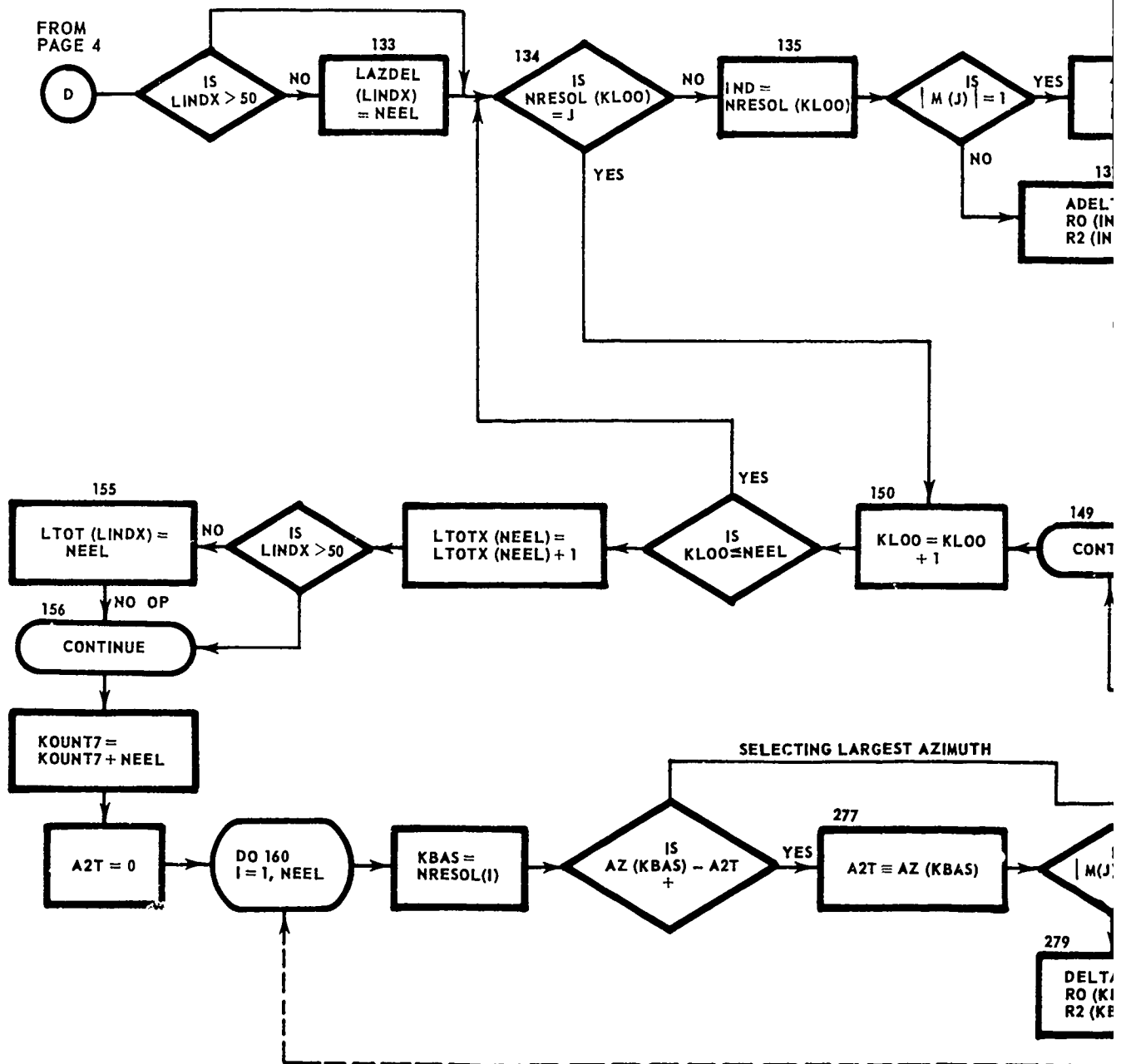


Figure 2-4 (Cont.)

B.



Figur

A.

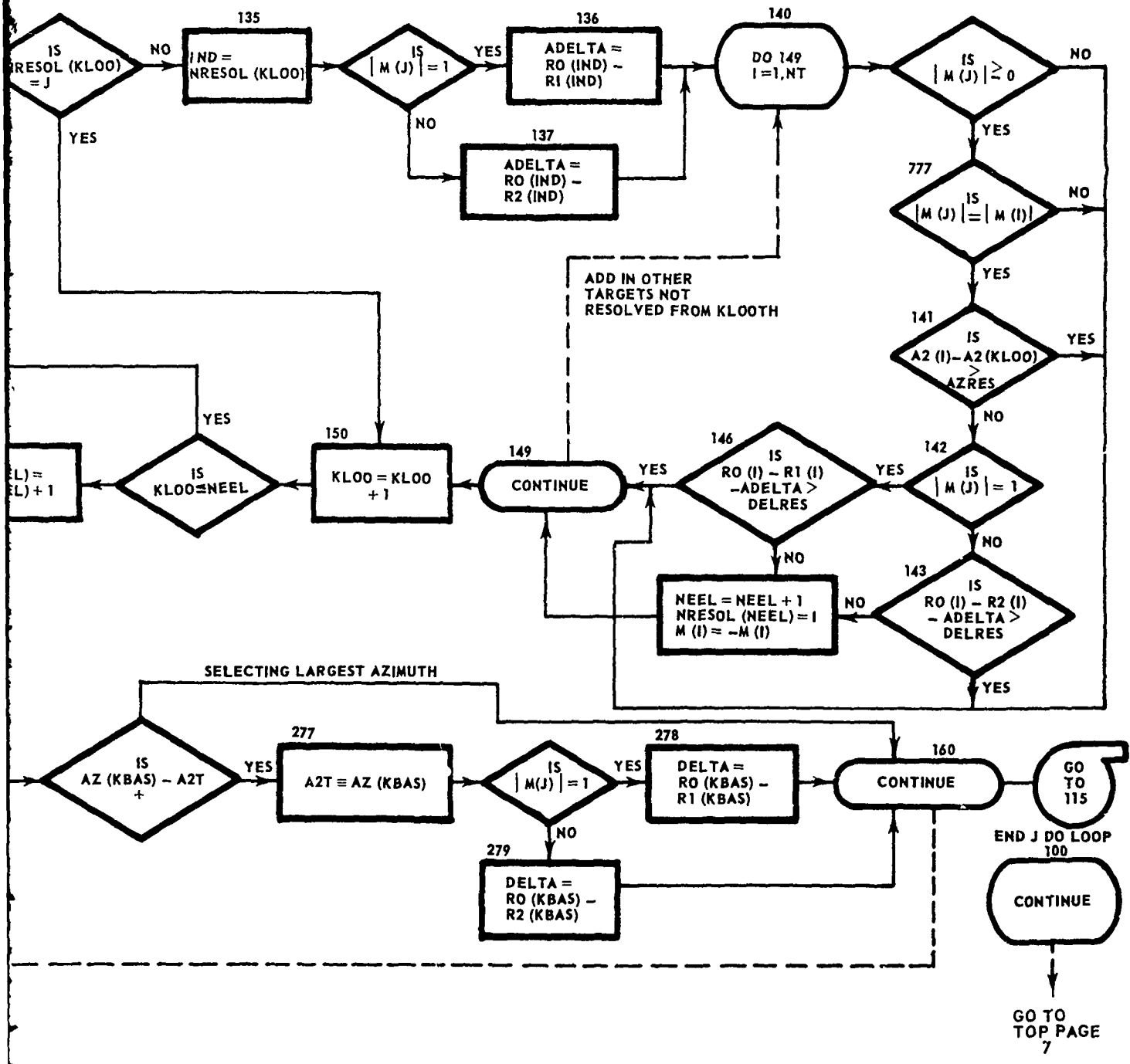


Figure 2-4 (Cont.)

B.

FROM Z
PAGE 4

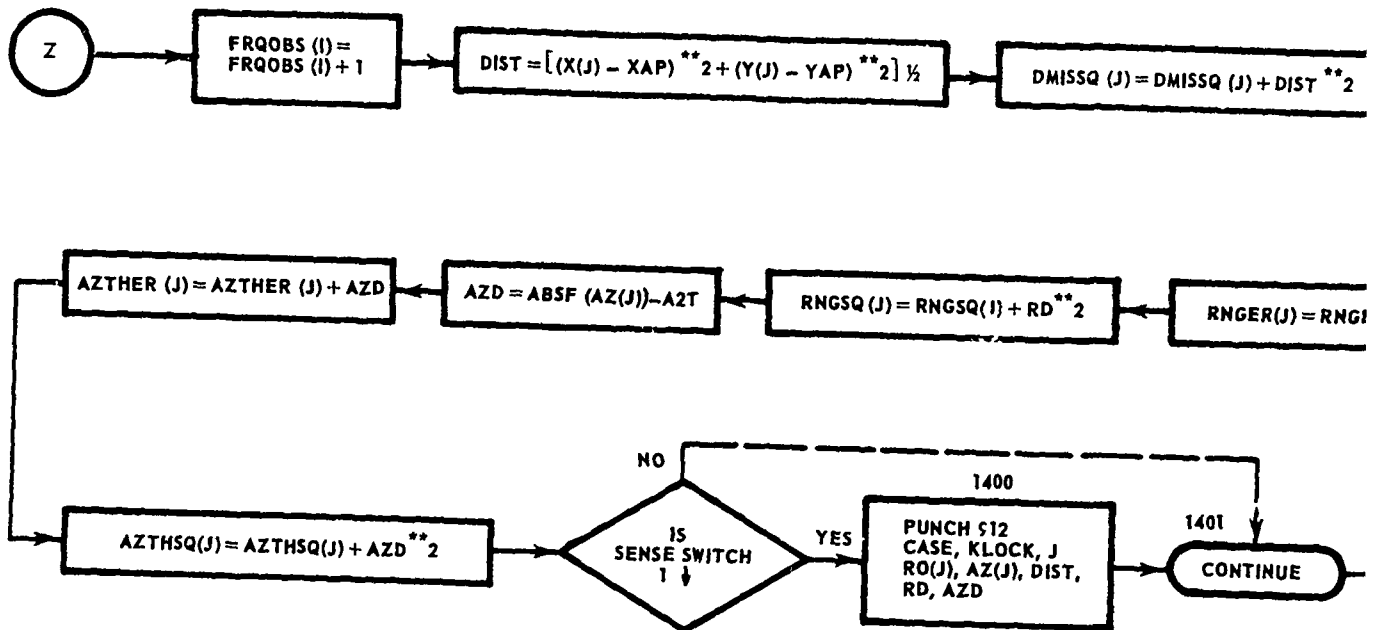


Figure 2-4 (Co)

A.

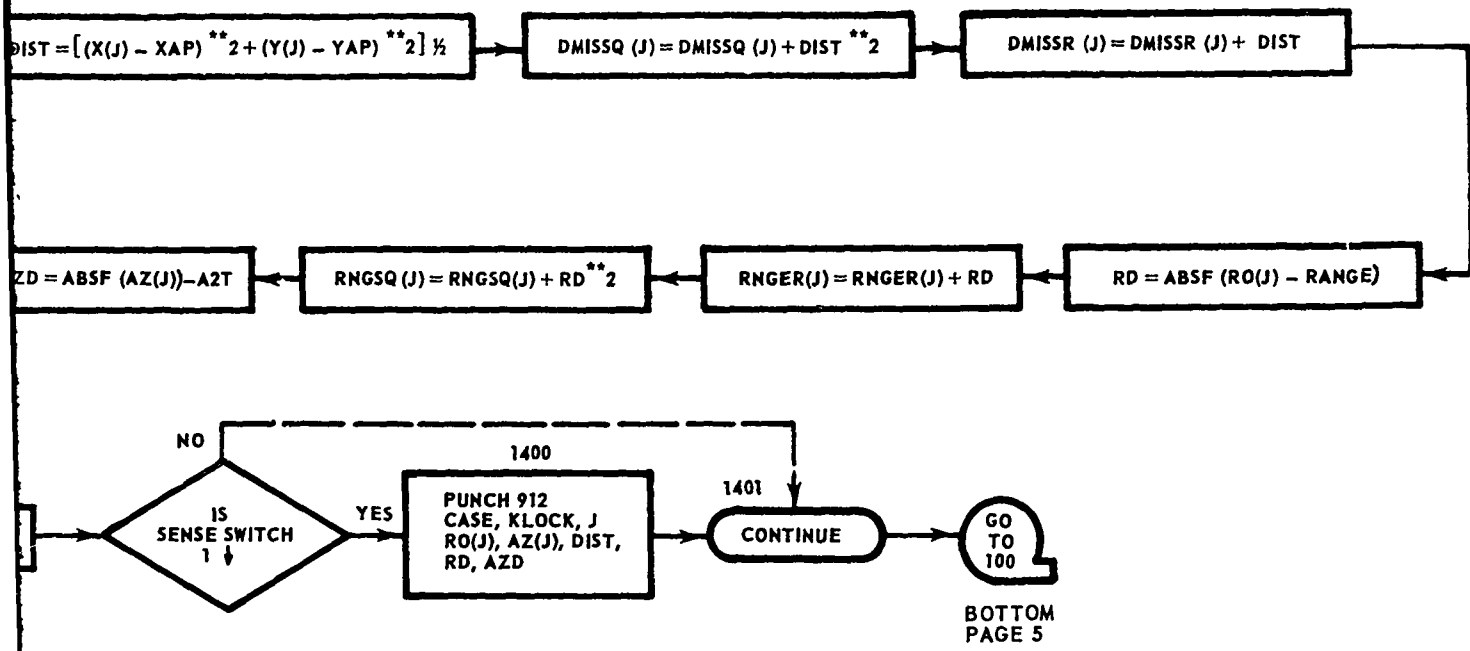
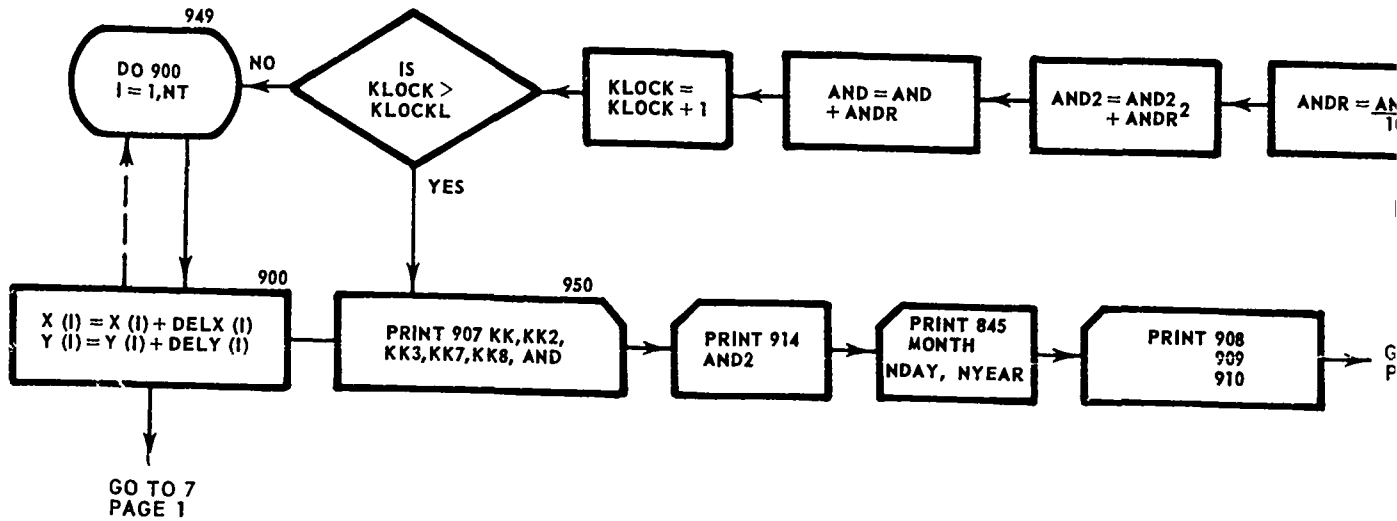
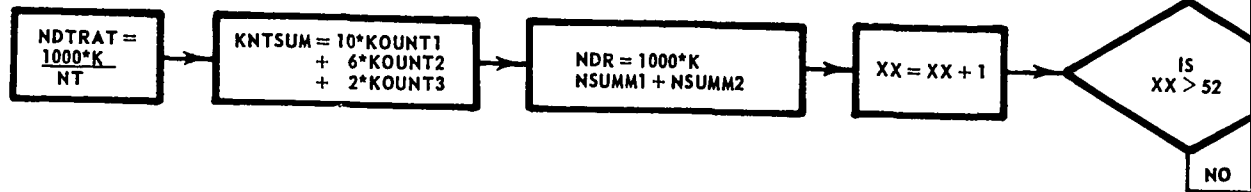


Figure 2-4 (Cont.)

B.

FROM PAGE 5 BOTTOM



A.

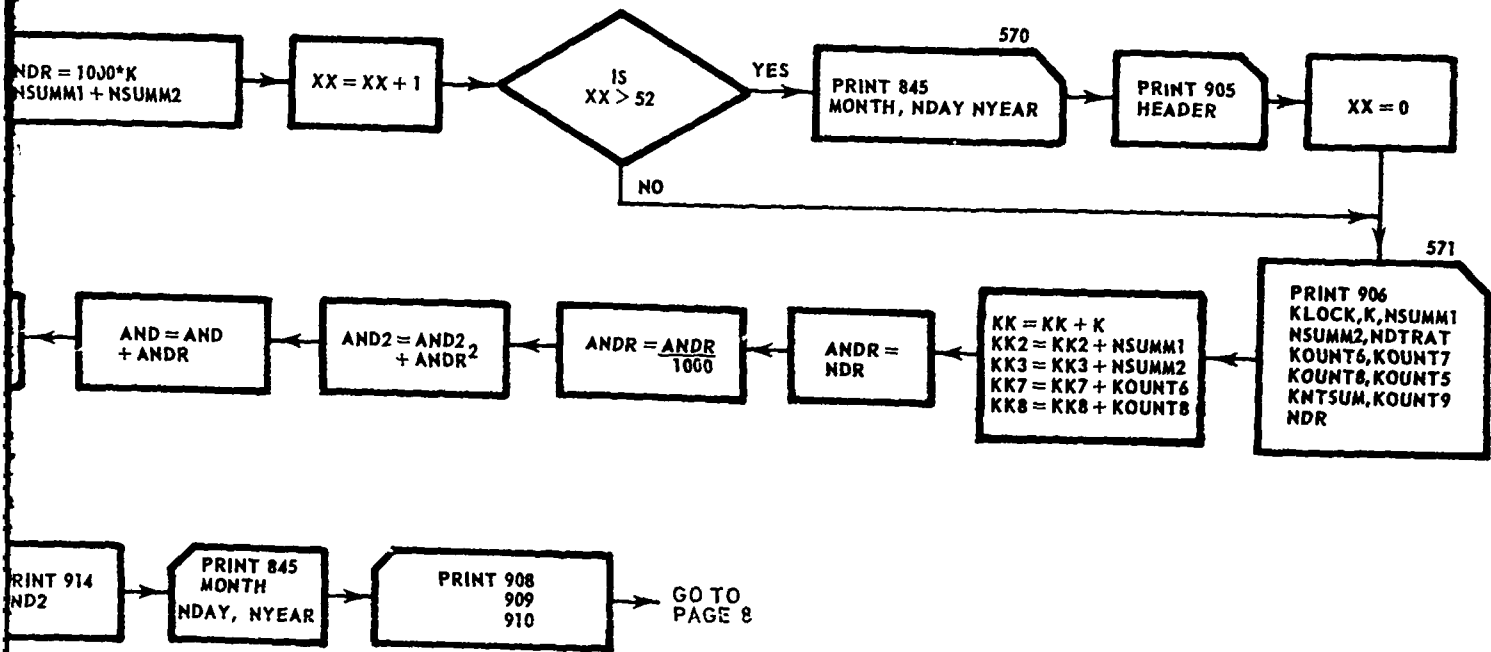


Figure 2-4 (Cont.)

B.

FROM
BOTTOM
PAGE 7

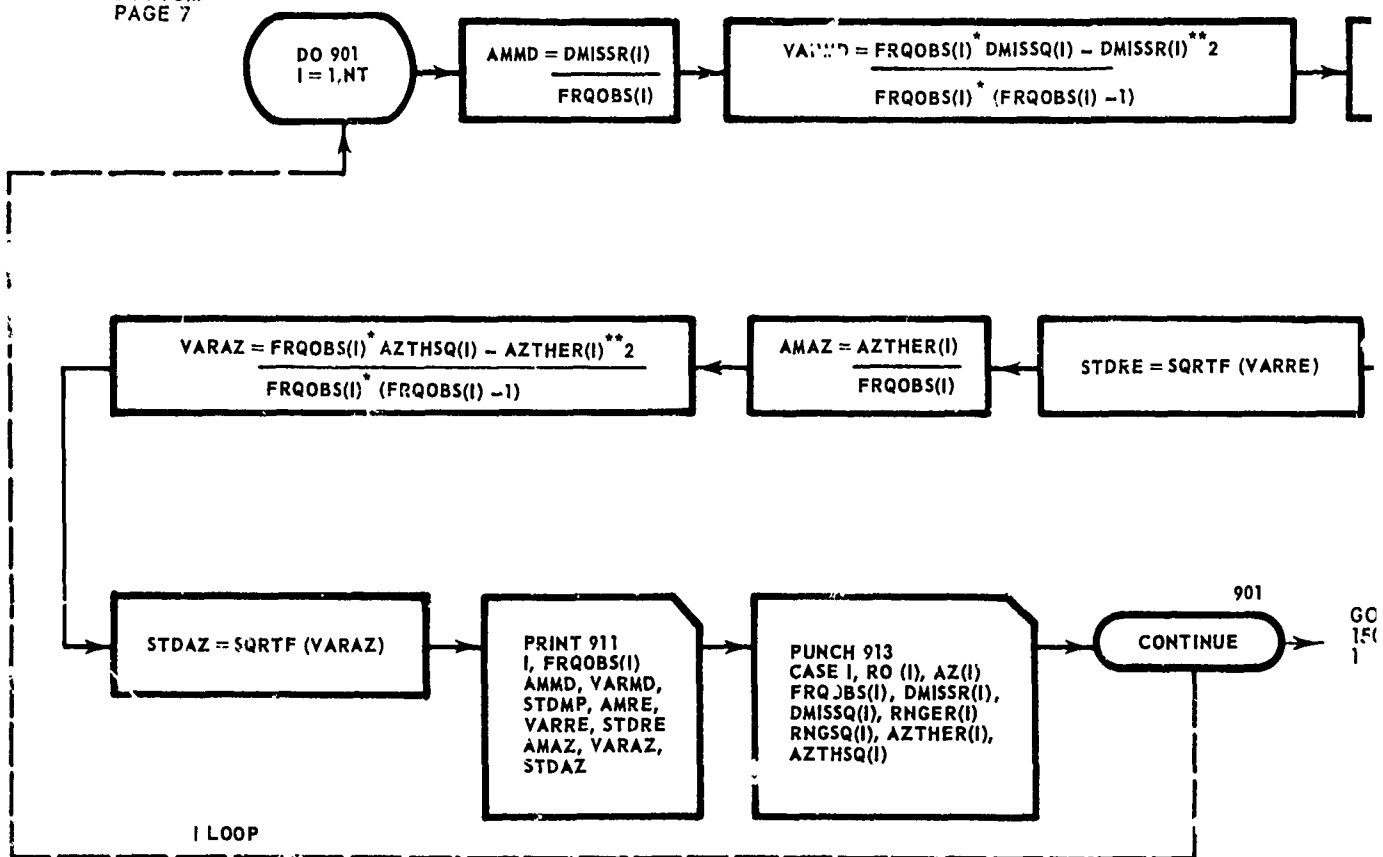


Figure 2-4 (

A

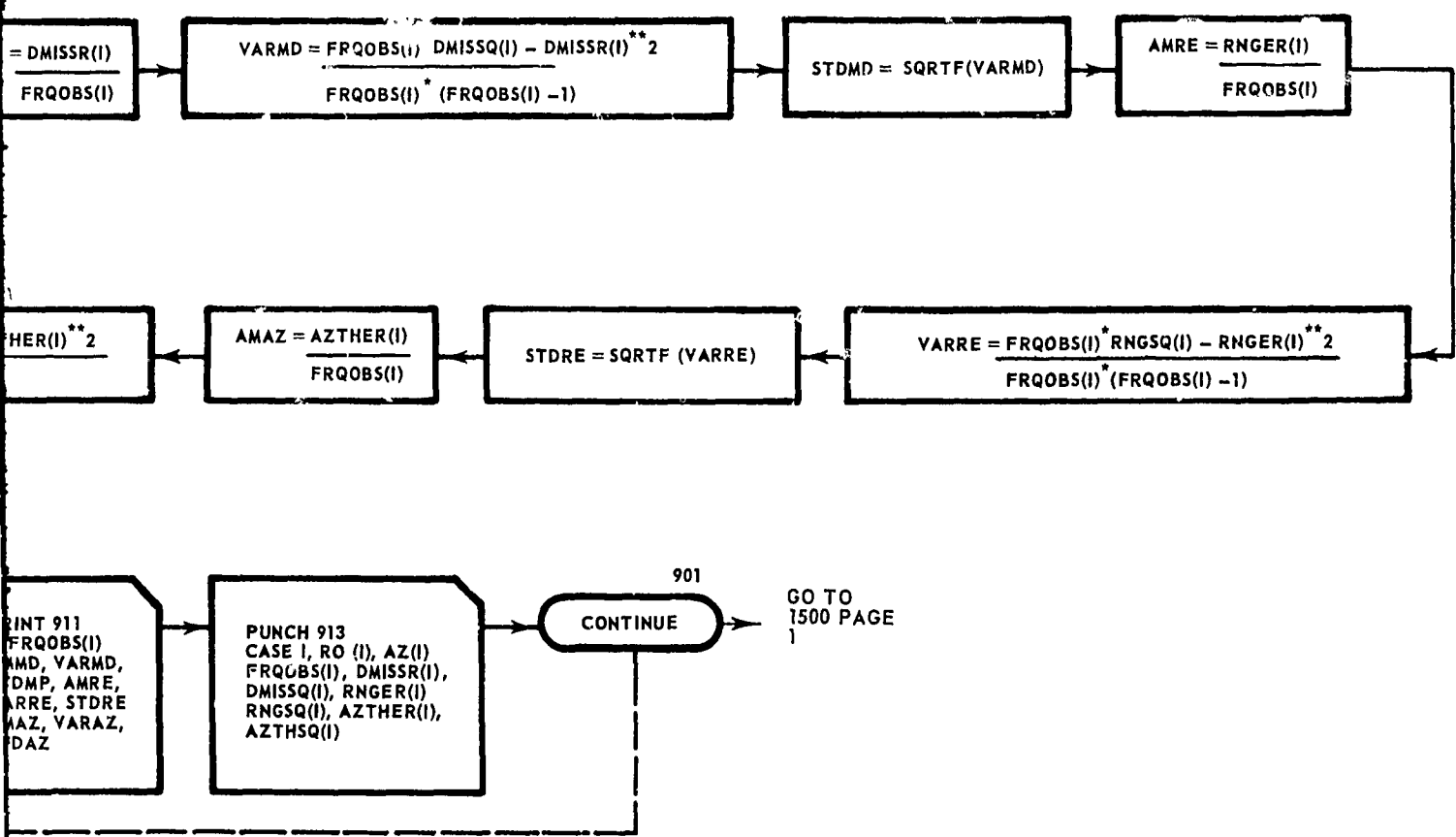


Figure 2-4 (Cont.)

B.

2.5 RAID SIZE ESTIMATION PROGRAM

The Raid Size Estimation Program is a modification to the original simulation program to include a process for more realistically estimating the number of targets in the unresolved (AZIMUTH and DELTA) clusters.

The addition of the counter, NESDT (which is used to count the new estimated number of detections), is the principal facility of the "Estimation Program." Originally, when a cluster of targets that were unresolved with respect to the radar was encountered, the scoring section reported only one target detected, regardless of the actual number in the cluster. Since the width of the unresolved cluster is almost always dependent upon the azimuths of the respective members of the cluster, one can give a minimum estimation of the number contained in the cluster. For example; if a cluster extends a width of 30° , and the azimuth resolution of the radar (AZRES) is 3° , it appears evident that there are $30^\circ/3^\circ$ or 10 radar beam widths in the clustered sector.

The program examines each target per frame to determine if a cluster is reported. If a target is not contained in a cluster, the detection counter (NESDT) is increased by one. When a cluster is reported the program proceeds to compute the width (SECTWD) of the clustered sector. This is accomplished by examining the respective azimuths of the targets in the sector and determining the largest and smallest azimuth of the group. Once the width has been obtained it is divided by the radar beam size (AZRES) to obtain the number of radar beams contained in the clustered sector. It should be noted that this result ($\frac{\text{SECTWD}}{\text{AZRES}}$) should be increased by one to yield the actual minimum number of targets contained in the cluster. (e. g. : if 10 beam widths are contained in the cluster then at least 11 targets are required to construct this sector width).

After completion of the detection analysis, the new estimated number of detections (NESDT), and the new detection rate (NESTR) are printed in tabular form along with the original output.

An optional punched card output has been added which may be used to obtain detection plots, and perhaps utilized in later analysis. The card format contains the case number (CASE), the number of targets in range (NSUM12), the number of apparent targets (K), the number of

detections (KOUNT5), the detection rate (NDR), the new estimated number of detections (NESDT), and the new estimated detection rate (NESTR).

The Historical Output Tape 4 has been modified to include, in addition to the coordinates of the true and apparent targets (X, Y, XAP, YAP), the sector width (SECTWD), the range (RANGE), and the azimuth (A2T1) of the true target.

The sum and the sum of squares of the tabular columns of NESTR are also computed and printed at the end of the tabular output.

2.5.1 Glossary

<u>ARANGE</u>	Target range (same as RANGE)
<u>A2T1</u>	Azimuth of the apparent target (same as A2T)
<u>CON1</u>	The angle from Omni 1 to the horizontal, with the main radar as vertex
<u>CON2</u>	The angle from Omni 2 to the horizontal, with the main radar as vertex
<u>FFF1</u>	Sum of the variable NESTR over all of the frames examined
<u>FFF2</u>	Sum of the squares of the variable NESTR over all of the frames examined
<u>FFF3</u>	Same as NESTR
<u>NESDT</u>	Counter containing the new number of detections per frame
<u>NESTR</u>	New rate of detection
<u>NSUM12</u>	Sum of NSUMM1 and NSUMM2
<u>NWIDE</u>	The value used to estimate the minimum number of targets in an unresolved (AZIMUTH and DELTA) cluster
<u>SCTWD1</u>	Same as SECTWD

SECTWD

The sector width of the AZIMUTH and DELTA unresolved cluster, plus the value AZRES

SMALL

The smallest azimuth contained in the unresolved (AZIMUTH and DELTA) cluster

XLARG

The largest azimuth value contained in the unresolved (AZIMUTH and DELTA) cluster

2.5.2 FORTRAN Listing

```
DIMENSION LJTHX(50),LTOTX(50),LAZDEX(50),NSIGNU(153)
DIMENSION NTT0AA( 60),DIST( 60),NMST( 60),NMSA( 60),DISMS( 60),
INHTT( 60),NHTA( 60),DISH( 60),NFST( 60),NFA( 60),DISFS( 60)
DIMENSION DIST(31),DISH(31)
DIMENSION DIRX( 60),DIRY( 60)
DIMENSION X( 60),Y( 60),RMIN( 60),RMIN2( 60),RMAX( 60),M( 60),
IRU( 60),R1( 60),R2( 60),AZ( 60)
DIMENSION NRESOL( 60),LJTH(50),LAZ(50),XAP( 60),YAP( 60),LAZDEL(50
1),LTOT(50),SCTWD1( 60),AZT1( 60),ARANGE( 60)
READ 5050,MONTH,NDAY,NYEAR
5050 FORMAT(3I2)
READ 222,X1,Y1,X2,Y2,CUN1,CUN2
222 FORMAT(6F10.1)
1500 DO 5333 I=1,50
LTOTX(I)=0
5333 LAZDEX(I)=0
SUX=0.
SUM=0.
FRE1=0.
FRE2=0.
FREU=0.
FREQ=0.
FREU2=0.
FREQ2=0.
DXH=0.
DXM=0.
NESTR1=0
NESTR2=0
PRINT 845,MONTH,NDAY,NYEAR
DO 8050 I=1,31
DISH(I)=0.
8050 DIST(I)=0.
5002 FORMAT(90H
XDATA, BENDIX SYSTEMS DIVISION 704.)
DO 5334 I=1,50
```

FIG 3 PROGRAM

```

3334 LJTHX(1)=0
DU 8000 I=1,155
8000 NSIGNO(I)=0
O=0.
READ 350,K1,K2,K3,K4,K5,K6,K7,K8,K9,K0
350 FORMAT(10I1)
READ 3,CASE,AZMUL,DELMUL,AZRES,DELRES,GAIN
3 FORMAT(6F10.1)
REWIND 2
READ INPUT TAPE 2,1150,NT,NF,LD
1150 FORMAT(3I5)
DU 1151 I=1,NT
1151 READ INPUT TAPE 2,1152,RMAX(I),RMIN(I),RMIN2(I)
CLOCKL=NF
CLOCKS=0.
XX=CLOCKS
PRINT 5002

PRINT 700,CASE,NT
700 FORMAT(12H CASE NUMBERF8.0,4H NTI4)
PRINT 1100,CLOCKL,AZMUL,DELMUL,AZRES,DELRES,GAIN
1100 FORMAT((16H CLOCKLF5.0,6H AZMULF8.6,7H DELMULF5.0,6H AZRE
1SF8.7,7H DELRESF8.6,5H GAINF8.6)///)
PRINT 701
701 FORMAT(116H CLOCK K NSUMM1 NSUMM2 NSUMM3 NMS NH1 NPS NPARAI NH:R
1AI NMSRAI NDIAI K6 K7 K8 K9 KNTSUM KY NDK NESTR RESDI)
400 FORMAT(5F10.2)
1152 FORMAT(3E16.8)
X0=500.
Y0=500.
KK=0
KK1=0
KK2=0
KK3=0
KK4=0
KK5=0
KK6=0
KK7=0
KK8=0
AKY=0.
AK1=0.
FFF1=0.
FFF2=0.
333 FORMAT(F8.0)
1 READ INPUT TAPE 2,333,CLOCK
IF(CLOCK) 301,1177,1177

```

```

1177 DO 1153 I=1,NT
1153 READ INPUT TAPE 20 1154,X(I),Y(I)
1154 FORMAT(2L16.8)
      KOUNT1=0
      KOUNT2=0
      KOUNT3=0
      KOUNT4=0
      KOUNT5=0
      KOUNT6=0
      KOUNT7=0
      KOUNT8=0
      KOUNT9=0
      NSJMM1=0
      NSUMM2=0
      NSUMM3=0
      SUMLO=0.
      SUMO =0.
      SUML1=0.
      SUML2=0.
      NESDT=0
      NESTK=0
      DO 50 I=1,NT
      DIFX(I)=X(I)-X0
      DIFY(I)=Y(I)-Y0
      RO(I)=SQRT(DIFX(I)**2+DIFY(I)**2)
      IF(RO(I)-RMAX(I)) 9,9,8
      IF(RO(I)-RMIN(I)) 500,500,501
500 SUMLO=SUMLO+RMINZ(I)
      GO TO 502
501 SUMLO=SUMLO+1./RO(I)**2
502 SUMO=SUMO+1./RO(I)**2
      R1(I)=SQRT((X(I)-X1)**2+(Y(I)-Y1)**2)
      IF(R1(I)-RMAX(I)) 11,11,600
11 IF(R1(I)-RMIN(I)) 602,602,601
601 SUML1=SUML1+1./R1(I)**2
      GO TO 650
602 SUML1=SUML1+RMINZ(I)
650 CONTINUE
15 R2(I)=SQRT((X(I)-X2)**2+(Y(I)-Y2)**2)
      IF(R2(I)-RMAX(I)) 15,15,660
15 IF(R2(I)-RMIN(I)) 662,662,661
661 SUML2=SUML2+1./R2(I)**2
      GO TO 660
662 SUML2=SUML2+RMINZ(I)
660 CONTINUE

```

```

18 IF(DIFY(I)) 18,18,19
18 M(I)=3
   NSUMM3=NSUMM3+1
   GO TO 50
19 IF(K0(I)-KMAX(I)) 40,40,22
40 IF(DIFX(I)) 20,23,23
20 IF(R1(I)-RMAX(I)) 21,21,22
22 M(I)=4
   GO TO 50
21 M(I)=1
   NSUMM1=NSUMM1+1
   GO TO 26
23 IF(K2(I)-RMAX(I)) 25,25,22
23 M(I)=2
   NSUMM2=NSUMM2+1
26 AZ(I)=ATN1F(DIFY(I),-DIFX(I))
50 CONTINUE
   NSUM12=NSUMM1+NSUMM2
   DO 56 I=1,50
   LJTH(I)=0
   LAZ(I)=0
   LAZDEL(I)=0
56 LTOT(I)=0
   K=0
   LINDX=0
   DO 100 J=1,N1
   IF(M(J)) 100,50,50
58 IF(XABS(M(J))-2) 60,60,100
60 L=0
   DO 70 I=1,N1
   IF(XABS(M(I))-XABS(M(J))) 76,62,76
62 IF((ABS(AZ(J)-AZ(I)))-AZRES) 64,64,76
64 L=L+1
   NRESOL(L)=1
70 CONTINUE
72 SUML=SUMO
   AMO=0.
   AMOL=0.
74 IF(XABS(M(J))-1) 75,71,75
75 KADOMA=SUML2
   GO TO 77
77 KADOMA=SUML1
   DO 80 I=1,L
   KDAS=NRESOL(I)
   FRACT=1./K0(KDAS)**2
   FLJ=1.-ALSF((AZ(J)-AZ(KDAS))/AZRES)
   AMO=AMO+FLJ*FRACT
   IF(1-FRACT<RMINZ(KDAS)) 275,275,274
274 FRACT=RMINZ(KDAS)

```

```

275  AMDL=AMBL+FLJ*FRACT
      IF(J-KBAS) 88,276,88
276  TARGML=FRACT
      IF(XABS(M(J))-1) 83,85,83
83   IF(R2(KBAS)-RMIN(KBAS)) 680,680,681
680  RANGMA=RMIN(KBAS)**2
      GO TO 602
601  RANGMA=R2(KBAS)**2
602  CONTINUE
      DELTA=RO(J)-R2(J)
      GO TO 88
85   IF(R1(KBAS)-RMIN(KBAS)) 690,690,691
690  RANGMA=RMIN(KBAS)**2
      GO TO 692
691  RANGMA=R1(KBAS)**2
692  CONTINUE
      DELTA=RO(J)-R1(J)
86   CONTINUE
      IF (AMDL-(SUMO-AMB)*GAIN)4000,89,89
4000  KOUNT9=KOUNT9+1
      GO TO 100
89   AOSIGO=((SUMO-AMB)*GAIN+AMBL-TARGML)/TARGML
      AOSIGA=(RANGMA*(RADOMA)-1.
      SIGNO=1./(2.+AOSIGO+AOSIGA+AOSIGO*AOSIGA)
      NS=SIGNO*500.
      NS=NS+1
      IF(NS-102) 8002,8002,8001
8001  NSIGNO(102)=NSIGNO(102)+1
      GO TO 8003
8002  NSIGNO(NS)=NSIGNO(NS)+1
8003  CONTINUE
      I:(SIGNO=.0112) 99,101,101
101  KOUNT1=KOUNT1+1
      GO TO 112
99  I:(SIGNO=.00041) 103,102,102
102  KOUNT2=KOUNT2+1
      I:(RAN20:(0)-.06) 112,250,250
103  I:(SIGNO=.00021) 103,104,104
104  KOUNT3=KOUNT3+1
      I:(RAN20:(0)-.02) 112,250,250
103  KOUNT4=KOUNT4+1
250  CONTINUE
      GO TO 100
112  KOUNT5=KOUNT5+1
      IF(L-1) 113,113,114
114  KOUNT6=KOUNT6+1

```

```

LINDX=LINDX+1
IF (LINDX=50) 73,75,122
75  LJTH(LINDX) =J
    LJTHX(L)=LJTHX(L)+1
    LAZ(LINDX) =L
    GO TO 122
113  AZT=AZ (J)
    SECTWD=AZRES
    NWIDE=1
115  ANO=(KAM2DF (U)-.5)*AZMUL
    NRESOL=NRESOL+NWIDE
    IF (SENSE SWITCH2)1719,1720
1719 WRITE OUTPUT TAPE 5,1714,0,SECTWD,NWIDE,XLARG,SMALL
1714 FORMAT(12H TARGET NO.=15,8H SECTWD=10.4,7H NWIDE=1.00, XLARGE=PIU
X.3, 7H SMALL=PIU.5)

1720 AZT=AZT+ANO
    BNU=(KAM2DF (U)-.5)*DELRES
    DELTA=DELTA+BNU
    IF (XABS(M(J))-1) 116,116,117
116  ANGLE=AZT+CON1
    GO TO 118
117  ANGLE=AZT-CON2
118  RANGE=(2500.-DELTA**2)/(100.*COS(Angle)-2.*DELTA)
1298 CONTINUE
    K=K+1
120  XAP(K)=XU-RANGE*COS(AZT)
121  YAP(K)=YU+RANGE*SIN(AZT)
    AZT1(K)=AZT
    SECTWD(K)=SECTWD
    KRANGE(K)=RANGE
    GO TO 100
122  NEEL=0
    DO 130 I=1,L
    KBAS=NRESOL(I)
    IF (XABS(M(J))-1) 123,125,124
124  I: (ABS.(RU(KBAS)-R2(KBAS)-DELTA)-DELRES) 127,127,130
123  I: (ABS.(RU(KBAS)-R1(KBAS)-DELTA)-DELRES) 127,127,130
127  NEEL=NEEL+1
    M(KBAS)=-M(KBAS)
    NRESOL(NEEL)=NRESOL(I)
130  CONTINUE
    LAZDEX(NEEL)=LAZDEX(NEEL)+1
    IF (NEEL=1) 517,517,132
517  KOUNTB=KOUNTB+1
    IF (LINDX=50)7000,7000,113

```

```

7000 LAZDEL(LINDX)=1
      LTOT(LINDX)=1
      GO TO 113
132  KLOO=1
      IF(LINDX-50)133,133,134
133  LAZDEL(LINDX)=NEEL
134  IF(NRESOL(KLOO)-J) 135,150,135
135  IND=NRESOL(KLOO)
      IF(XABSF(M(J))-1) 136,136,137
136  DELTA=RO(IND)-R1(IND)
      GO TO 140
137  DELTA=RO(IND)-R2(IND)
140  DO 149 I=1,NT
      IF(M(I)) 149,777,777
141  IF(XABSF(M(I))-XABSF(M(J)))149,141,149
142  IF(ABSF(AZ(I)-AZ(KLOO))-AZRES) 142,142,149
143  IF(XABSF(M(J))-1) 146,146,143
144  IF(ABSF(RO(I)-R1(I)-DELTA)-DELRES) 144,144,149
144  NEEL=NEEL+1
      NRESOL(NEEL)=1
      M(I)=-M(I)
      GO TO 149
143  IF(ABSF(RO(I)-R2(I)-DELTA)-DELRES) 144,144,149
149  CONTINUE
150  KLOO=KLOO+1
      IF(KLOO=NEEL) 134,134,152
152  LTOTX(NEEL)=LTOTX(NEEL)+1
      IF(LINDX-50)153,155,156
153  LTOT(LINDX)=NEEL
156  SUMAZ=0.
      DENOM=NEEL

      KOUNT7=KOUNT7+NEEL
      AZT=0.
      IF(SENSE SWITCH2)1715,1716
1715 WRITE OUTPUT TAPE 5,1713
1713 FORMAT(41H      J  AZIMUTH  NO. OF UNRESOLVED TARGET)
1716 DO 160 I=1,NEEL
      KBAS=NRESOL(I)
      IF(SENSE SWITCH2)1717,1718
1717 WRITE OUTPUT TAPE 5,1712,J,AZ(KBAS),NRESOL(I)
1712 FORMAT(15,F10.3,I24)
1718 IF(AZ(KBAS)-AZI)160,160,277
277  AZI=AZ(KBAS)
      IF(XABSF(M(J))-1)278,278,279
278  DELTA=RO(KBAS)-R1(KBAS)
      GO TO 160

```

```

270 DELTA=R0(KBAS)-RZ(KBAS)
180 CONTINUE
KBAS=NRESOL(1)
XLARG=AZ(KBAS)
SMALL=AZ(KBAS)
DO 1710 I=2,NRES
KBAS=NRESOL(I)
IF(AZ(KBAS)-SMALL)1705,1704,1704
1704 IF(XLARG-AZ(KBAS))1705,1710,1710
1705 XLARG=AZ(KBAS)
GO TO 1710
1703 SMALL=AZ(KBAS)
1710 CONTINUE
SECTWD=XLARG-SMALL+AZRES
NWIDL=SECTWD/AZRES+.5
GO TO 115
100 CONTINUE
NMS=0
NIT=0
NFS=0
IF(KID/1000.100.0)
171 DO 172 ITR=1,NT
NMS=NMS+1
NAST(NMS)=ITR
NMSA(NMS)=-0
172 DISMS(NMS)=-0
GO TO 575
170 DO 180 IIR=1,NI
IAP=1
NTUAA(IIR)=1
XIRU=X(IIR)
YIRU=Y(IIR)
DIFRX=ABS(XIRU-XAP(IAP))
DIFRY=ABS(YIRU-YAP(IAP))
DISSQ=DIFRX**2+DIFRY**2
DO 180 IAP=2,N
TDIFRX=ABS(XIRU-XAP(IAP))
TDIFRY=ABS(YIRU-YAP(IAP))
IF(TDIFRX-DIFRX) 165,164,164
164 IF(TDIFRY-DIFRY)165,166,166
165 DISSQ=TDIFRX**2+TDIFRY**2
IF(TDISSQ-DISSQ) 167,166,166
167 NTUAA(IIR)=IAP
DISSQ=DISSQ
DIFRX=TDIFRX
DIFRY=TDIFRY

```

```

160 CONTINUE
168 DISTT(ITR)=SQRTF(DISSQ)
171 DO 170 IAP=1,K
    ITR=1
    NATOTT=1
    YAPP=YAP(IAP)
    XAPP=XAP(IAP)
    DIFRX=ABSF(X(ITR)-XAPP)
    DIFRY=ABSF(Y(ITR)-YAPP)
    DISSQ=DIFRX**2+DIFRY**2
172 DO 170 ITR=2,N1
    TDIFRX=ABSF(X(ITR)-XAPP)
    TDIFRY=ABSF(Y(ITR)-YAPP)
    IF(TDIFRX-DIFRX) 169,170,170
170 IF(TDIFRY-DIFRY) 169,175,175
169 TDISSQ=TDIFRX**2+TDIFRY**2
    IF(TDISSQ-DISSQ) 173,170,170
175 NATOTT=ITR
    DISSQ=TDISSQ
    DIFRX=TDIFRX
    DIFRY=TDIFRY
175 CONTINUE
    DISTA=SQRTF(DISSQ)
    IF(XABSF(NTTOAA(NATOTT))-IAP) 176,177,176
177 NHT=NHT+1
    NATTT(NHT)=NATOTT
    NHTA(NHT)=IAP
    DISHT(NHT)=DISTT(NATOTT)
    MHD=MHD+1
    IF(MHD=31) 1404,1405,1405
1405 DISH(31)=DISH(31)+1.0
    GO TO 1406
1404 DISH(MHD)=DISH(MHD)+1.0
1406 CONTINUE
    NTTOAA(NATOTT)=-NTTOAA(NATOTT)
    GO TO 178
176 NFS=NFS+1
    NFST(NFS)=NATOTT
    NFSA(NFS)=IAP
    DISFS(NFS)=SQRTF(DISSQ)
170 CONTINUE
    DO 440 ITR=1,NT
    IF(NTTOAA(ITR)) 440,440,442
442 NMS=NMS+1
    NMST(NMS)=ITR
    NMSA(NMS)=NTTOAA(ITR)
    DISMS(NMS)=DISTT(ITR)
    MDS=DISMS(NMS)/5.0
    MDS=MDS+1
    IF(MDS=31) 1400,1401,1401

```

```

1401 DIST(31)=DIST(31)+1.
      GO TO 1402
1400 DIST(MDS)=DIST(MDS)+1.
1402 CONTINUE
440  CONTINUE
275  CONTINUE
      NFARAT=(1000*NFS)/K
      NMSRAT=(1000*NMS)/NT
      NDTRAT=(1000*K)/NT
      NHTRAT=(1000*NHT)/K

```

```

      KNTSUM=(10*KOUNT1+6*KOUNT2+2*KOUNT3)
      NDR=(1000*K)/(NSUMM1+NSUMM2)
      NESTR=(1000*NESDT)/NSUM12
      ANDR=NDR
      ANDR=ANDR/1000.
      AND2=ANDR**2
      FFF3=NESTR
      FFF3=FFF3/1000.
      AK1=AK1+AND2
      XX=XX+1.
      IF(XX-(51.+CLOCKS))844,843,844
843  PRINT 845,MONTH,NDAY,NYEAR
      XX=CLOCKS
      PRINT 5002
      PRINT 700,CASE,NT
845  FORMAT(10GH)
                                     DATE RUN IS,1M/12,1M/1
1
22)
      PRINT 701
844  CONTINUE
      PRINT 702,CLOCK,K,NSUMM1,NSUMM2,NSUMM3,NMS,NHT,NFS,NFARAT,NHTRAT,
X,NMSRAT,NDTRAT,KOUNT6,KOUNT7,KOUNT8,KOUNT5,KNTSUM,KOUNT9,NDR,NESTR,
X,NESDT
      IF(SENSE SWITCH 1)1707,1708
1707 PUNCH 1709,CASE,CLOCK,NSUM12,K,KOUNT5,NDR,NESTR,NESDT
1709 FORMAT(2F6.0,6I6)
1708 KK=KK+K
      KK2=KK2+NSUMM1
      KK3=KK3+NSUMM2
      KK4=KK4+NMS
      KK5=KK5+NHT
      KK6=KK6+NFS
      KK7=KK7+KOUNT6
      KK8=KK8+KOUNT8
      AK9=AK9+ANDR

```

```

FFF1=FFF1+FFF3
FFF2=FFF2+(FFF3)**2
702 FORMAT(F6.0,214,317,214,15,317,16,314,216,315)
IF(K3) 8075,8075,8076
8076 CONTINUE
WRITE OUTPUT TAPE 4,900,CASE,CLOCK,NT,K
900 FORMAT(F11.0,F10.0,215)
DO 901 I=1,NT
901 WRITE OUTPUT TAPE 4,902,X(I),Y(I)
902 FORMAT(5F14.4)
IF(K) 6000,6001,6000
6000 DO 903 I=1,K
903 WRITE OUTPUT TAPE 4,902,XAP(I),YAP(I),SCTWD1(I),AZT1(I),ARANGE(I)
6001 CONTINUE
8075 CONTINUE
359 IF(K6) 361,361,362
362 KEN=6
IF(NMS) 2100,2200,2100
2100 DO 906 I=1,NMS
906 WRITE OUTPUT TAPE 3,907,KEN,CASE,CLOCK,NMST(I),NMSA(I),DISMS(I)
2200 CONTINUE
361 IF(K7) 363,363,364
364 KEN=7
IF(NHT) 2101,2102,2101
2101 DO 908 I=1,NHT
IS=NHTA(I)

ID=NHTT(I)
908 WRITE OUTPUT TAPE 3,907,KEN,CASE,CLOCK,NHTT(I),NHTA(I),DISMT(I),
IX(ID),Y(ID),XAP(IS),YAP(IS)
2102 CONTINUE
363 IF(K8) 365,365,366
366 KEN=8
IF(NFS) 2103,2104,2103
2103 DO 909 I=1,NFS
909 WRITE OUTPUT TAPE 3,907,KEN,CASE,CLOCK,NFST(I),NNSA(I),DISFS(I)
365 CONTINUE
2104 CONTINUE
GO TO 7
301 KEN=0
PRINT 5DD5,KK1,KK2,KK3,KK4,KK5,KK6,KK7,KK8
5DD5 FORMAT(4H TOT16,I5,I5,I13,2I5,I5,I14)
PRINT 1711,AK9,AK1,FFF1,FFF2
1711 FORMAT(13H SUM OF NDR= F10.4,23H SUM OF SQUARES OF NDR=F12.6,14H
XSUM OF NESTR=F10.4,25H SUM OF SQUARES OF NESTR=F12.6)
5DD1 FORMAT(7H TOTALI14,I5,I5)
PRINT 845,MONTH,NDAY,NT,EAR

```

```

PRINT 700,CASE,NT
PRINT 5002
PRINT 1066
1000 FORMAT(56H          HITS          MISSES
1)
PRINT 1067
1007 FORMAT(70H      LOWER UPPER NUMBER STD FREQ      LOWER UPPER NUM
1067)
1400 FORMAT(37H      HITS          MISSES )
1409 FORMAT(46H LOWER UPPER NUMBER      LOWER UPPER NUMBER
XLOW=0.
HIGH=5.
XLO=0.
HIG=0.25
DO 1668 I=1,31
SUX=SUX+DIST(I)
1008 SUM=SUM+DISH(I)
SUX=1000./SUX
SUM=1000./SUM
DO 1411 I=1,31
LXX=DISH(I)*SUM+.5
LYY=DISH(I)*SUX+.5
PRINT 1410,XLO,HIG,DISH(I),LXX,XLOW,HIGH,DISH(I),LYY
FREQ=(HIGH-XLO)/2.+XLO
FREQ2=(HIGH-XLOW)/2.+XLOW
FREQ2=FREQ2+FREQ**2*DISH(I)
FREQ2=FREQ2+FREQ**2*DISH(I)
FREQ1=FREQ1+FREQ*DISH(I)
FREQ2=FREQ2+FREQ*DISH(I)
DXH=DXH+DISH(I)
DXM=DXM+DISH(I)
XLOW=HIGH
HIGH=HIGH+5.
XLO=HIG
1411 HIG=HIG+.25
1410 FORMAT(F10.2,F7.2,F8.0,I10,F10.0,F7.0,F8.0,I10)
SIGMAH=SQRT(FREQ2/DXH-(FREQ1/DXH)**2)
SIGMAM=SQRT(FREQ2/DXM-(FREQ2/DXM)**2)
BARM=FREQ1/DXH
BARM=FREQ2/DXM
PRINT 1069,SIGMAH,SIGMAM,BARM,BARM

1009 FORMAT(13H      SIGMA HIT F8.3,I10H SIGMA MISS F8.3,I10H MEAN HIT F8.3,I10H
1 MEAN MISS F9.3)
IF(K0) 369,369,370
370 HIGH=.002
PRINT 845,MONTH,NDAY,NEYAR
5000 FORMAT(11H)

```

```

PRINT 5002
PRINT 700,CASE,NT
PRINT 450
450 FORMAT(48H          FREQUENCY DISTRIBUTION SIGNAL TO NOISE )
PRINT 451
451 FORMAT(42H          LIMITS          LIMITS )
PRINT 452
452 FORMAT(80H LOWER    UPPER    NUMBER    LOWER    UPPER    NUMBER
1    LOWER    UPPER    NUMBER )
XLOW=.0
XLOWR=.102
XLO=.204
XHI=.206
HIGHK=.104
DO 453 I=1,51
PRINT 454,XLOW,HIGH,NSIGNO(I),XLOWR,HIGHK,NSIGNO(I+51),XLO,XHI,
INSIGNO(I+101)
454 FORMAT(F6.3,F9.3,I0,F13.3,F9.3,I0,F13.3,F9.3,I0)
XLO=XHI
XHI=XHI+.002
455 FORMAT(F6.3,F9.3,I0,F13.3,F9.3,I0)
XLOW=HIGH
HIGH=HIGH+.002
XLOWR=HIGHR
456 HIGHK=HIGHK+.002
457 CONTINUE
458 FORMAT(11H DURING THIS RAIBD4.25M AZIMUTH UNRESOLVE SECTORS WERE F
XOUND WHICH CONTAINED 15.8H TARGETS)
IF (N1) 460,460,461
461 PRINT 845,MONDAY,NTLAK
PRINT 700,CASE,NT
PRINT 472
LL1=0
LL2=0
LL3=0
DO 464 I=2,50
LL1=LL1+LJTHX(I)
LL2=LL2+LAZDEX(I)
466 LL3=LL3+LTOTX(I)
464 PRINT 462,I,LJTHX(I),LAZDEX(I),LTOTX(I),1
PRINT 5557,LL1,LL2,LL3
472 FORMAT(57H NO TARGETS    AZ UNR    AZ+DEL    AZ+DEL+OTHERS    NO TARGET
IS )
462 FORMAT(I12,2I9,I15,I9)
460 CONTINUE
PAUSE 1
GO TO 1500
507 FORMAT(13,F10.5,F5.0,2I8,5F8.2)
904 FORMAT(15,F10.5,F5.0,5I6)
END(0,1,0,0,1)

```

This page was left intentionally blank.

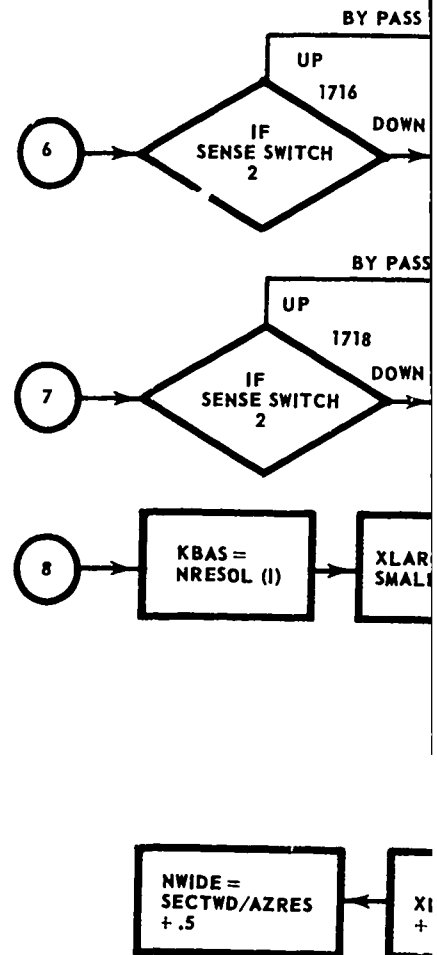
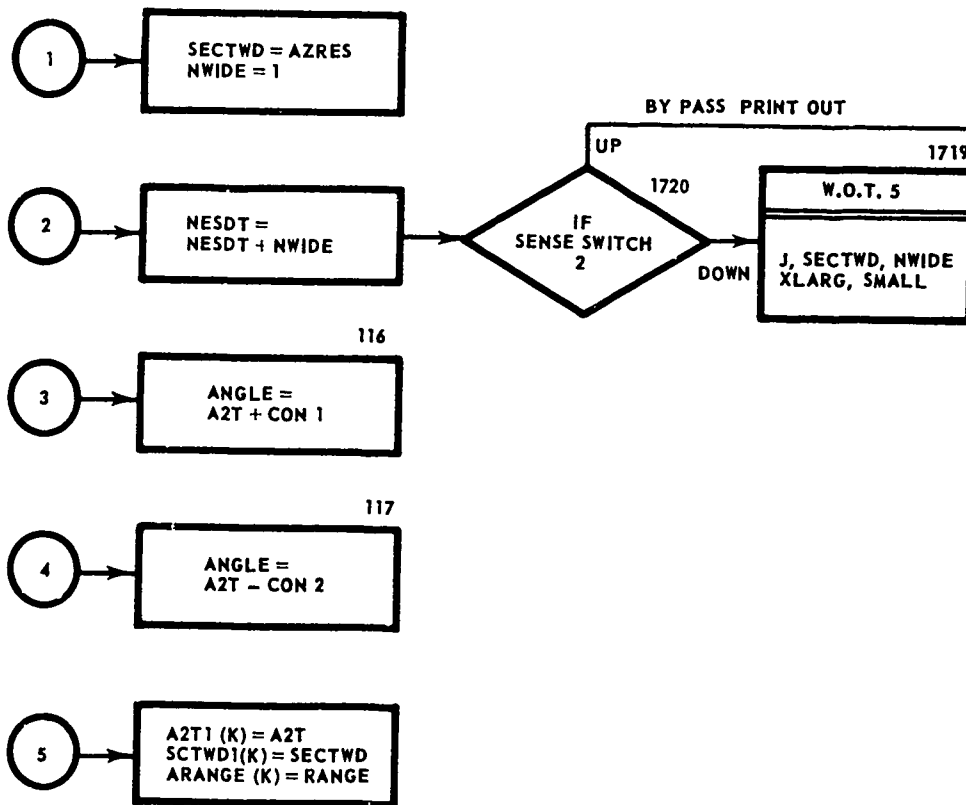


Figure 2-5 A

A₁

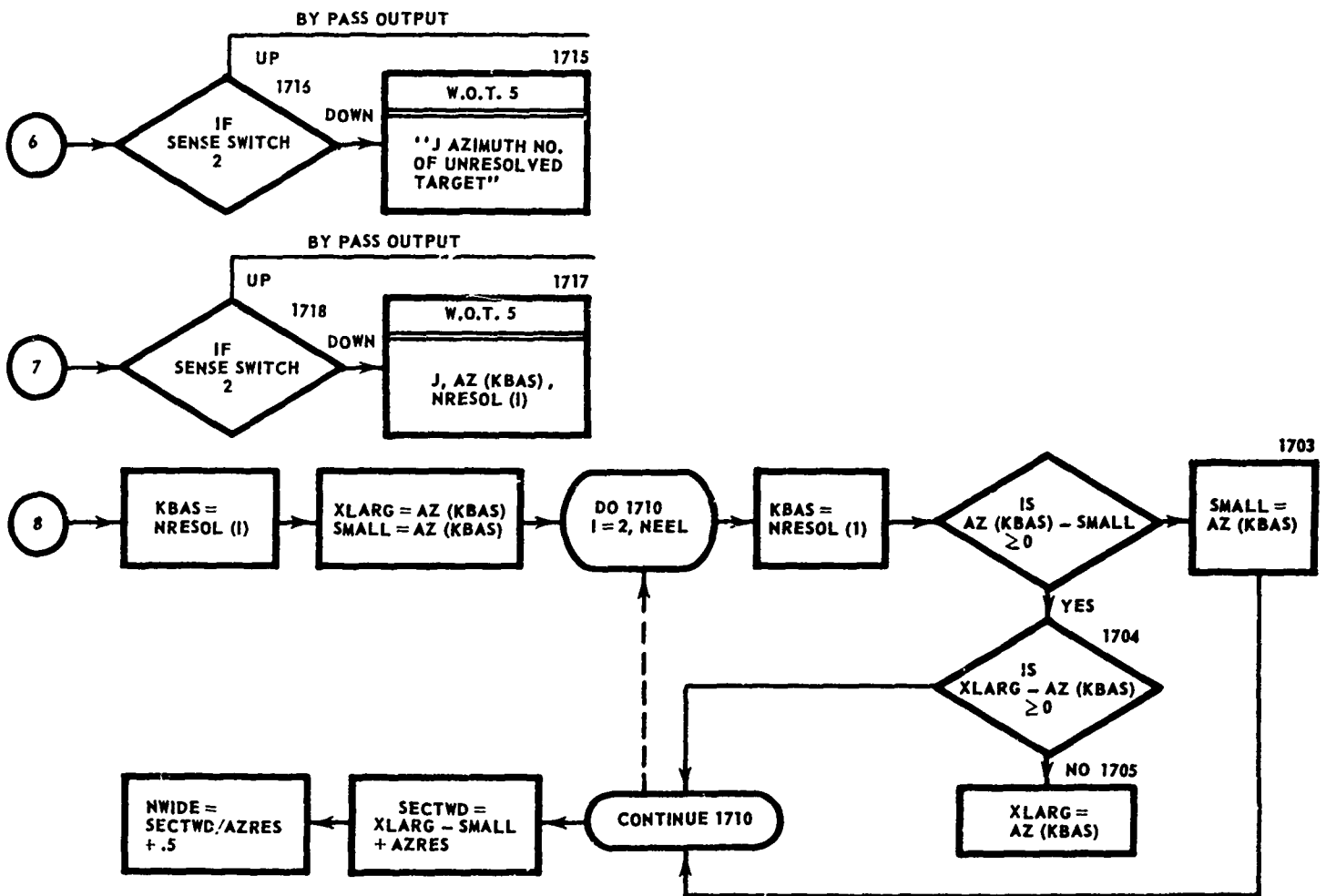


Figure 2-5 Modifications to Basic Program for Raid Size Estimation

B.

DIMENSION STATEMENTS

NTTOAA (100), DISTT (100), NMST (100), X (100),
 NMSA (100), DISMS (100), NHTT (100), Y (100),
 NHTA (100), DISHT (100), NFST (100), M (100),
 NFS A (100), DISFS (100), DIFX (100), RO (100),
 DIFY (100), R1 (100), RMAX (100), RM1N (100),
 RM1N2 (100), R2 (100), AZ (100), NRESOL (100),
 NSIGNO (153), LJTH (100), LAZ (50), XAP (100),
 YAP (100), LAZDEL (50), LTOT (50),
 LJTHX (50), LTOTX (50), LAXDEX (50),
 DIST (50), DISH (50)

READ 5050
 MONTH,NDAY,NYEAR

READ 222
 X1, Y1, X2, Y2,
 CON1, CON2

DO 5333
 I = 1,50

LTO
 LAZ

PRINT 5002
 PRINT 700, CASE,NT
 PRINT 1100, CLOCKL,
 AZMUL, DELMUL,
 AZRES, DELRES,
 GAIN
 PRINT 701

CLOCKL = NF
 CLOCKS = 0,
 XX = 0

RIT2, 1152
 RMAX (I), RM1N (I)
 RM1N2 (I)

DO 1151
 I = 1,NT

RIT 2, 1150
 NT,NF,ID

READ 3 CASE
 AZMUL, DELMU
 AZRES, DELRE

CONV = 57.29577
 XO = 500
 YO = 500
 X1 = 456.6985
 X2 = 475
 Y1 = 475
 Y2 = 543.3015
 KK = 0 KK2 = 0
 KK3 = 0 KK5 = 0
 KK4 = 0 KK7 = 0
 KK6 = 0 KK8 = 0
 AK9 = 0 AK1 = 0

RIT 2,599
 CLOCK

IS
 CLOCK
 +
 GO TO
 301
 PAGE 9

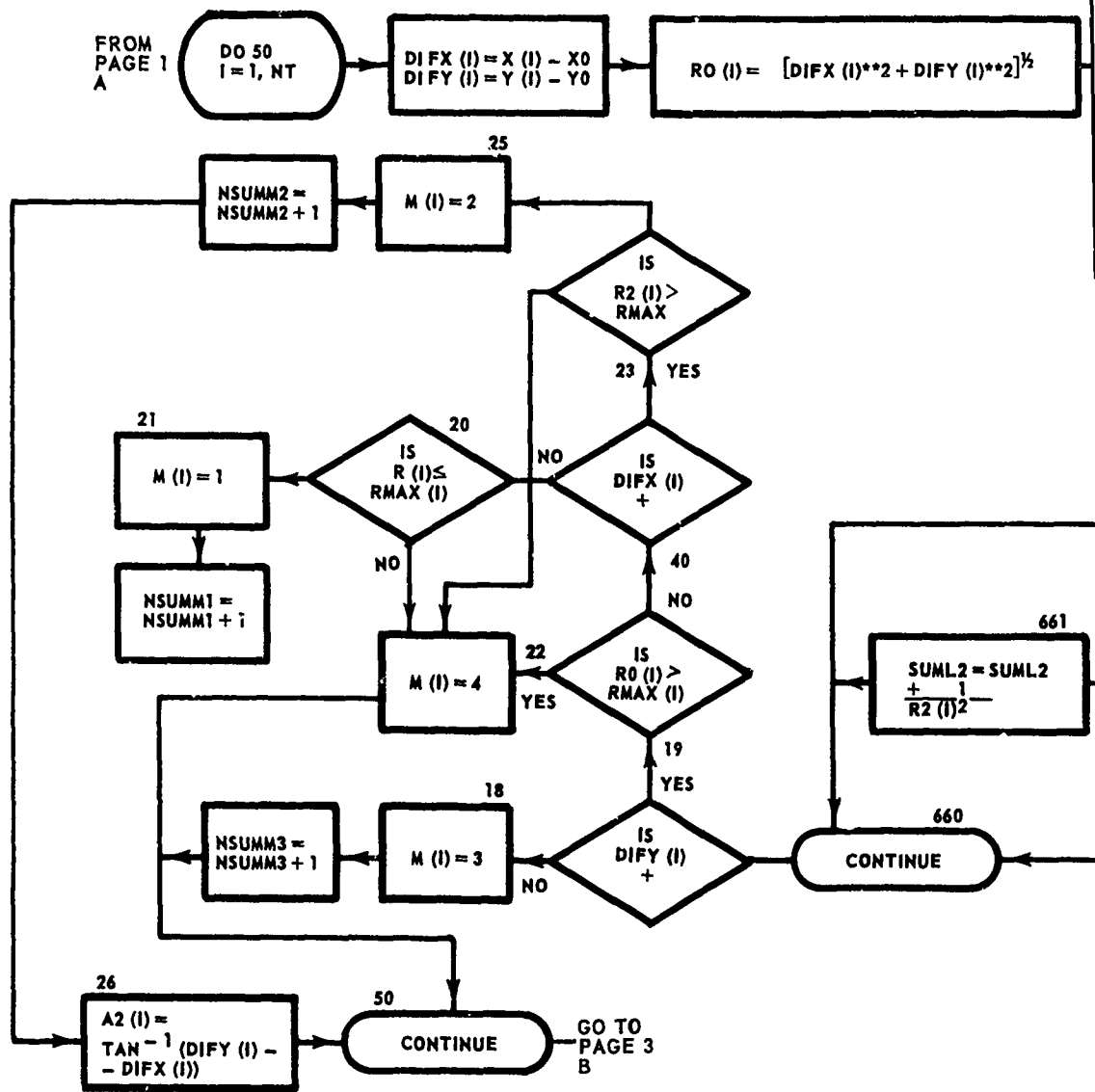
DO 1153
 I = 1, NT

RIT 2,1154
 X (I), Y(I)

KOUNT1 =
 KOUNT2 =
 KOUNT3 =
 KOUNT4 =
 KOUNT5 =
 KOUNT6 =
 KOUNT7 =
 KOUNT8 =
 KOUNT9 =
 NSUMM1 =
 NSUMM2 =
 NSUMM3 =
 SUMLO =
 SUMO = 0
 SUML1 =
 SUML2 =
 NESDT =
 NESTR =

A.

FROM
PAGE 1
A



A.

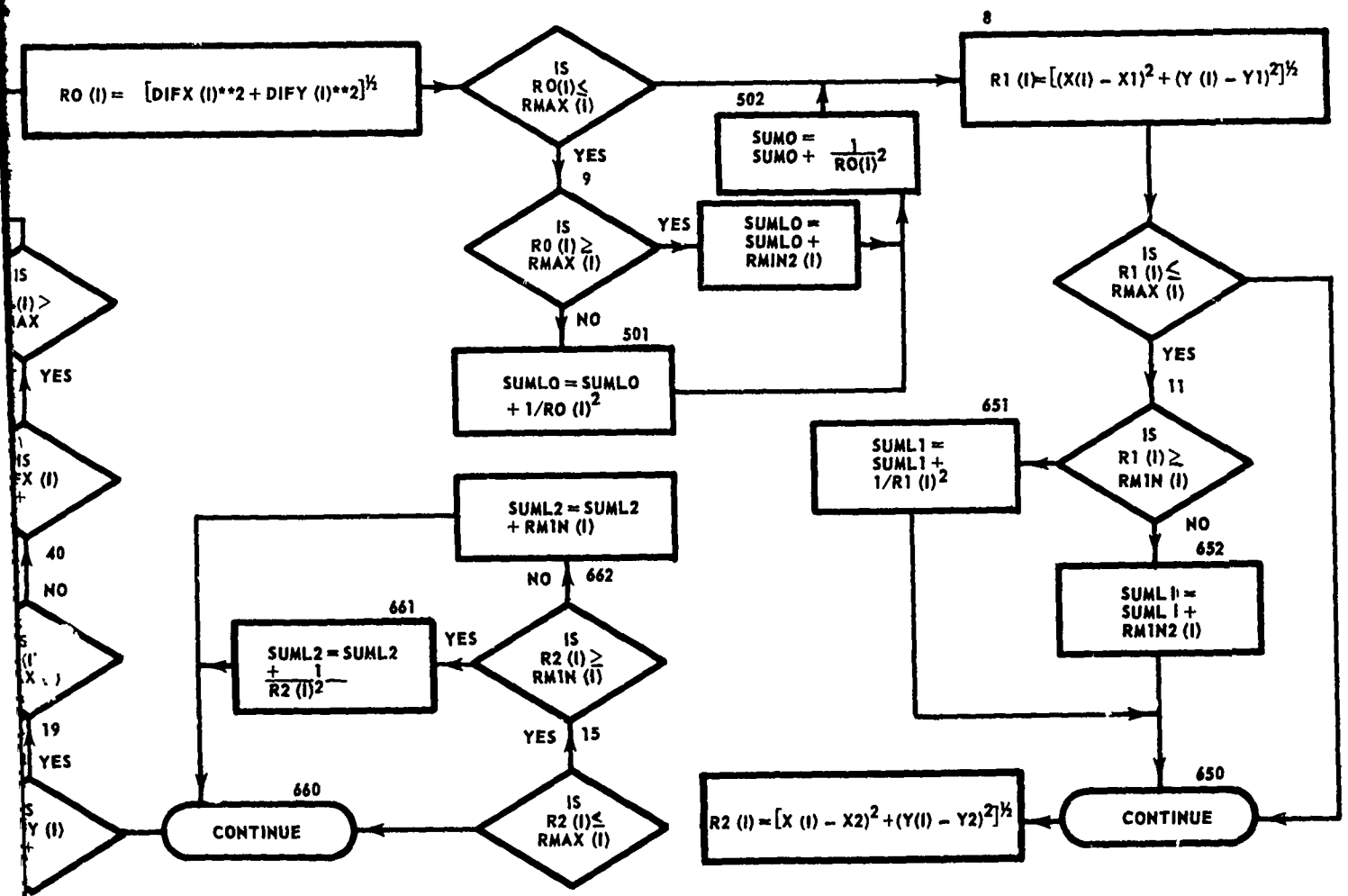
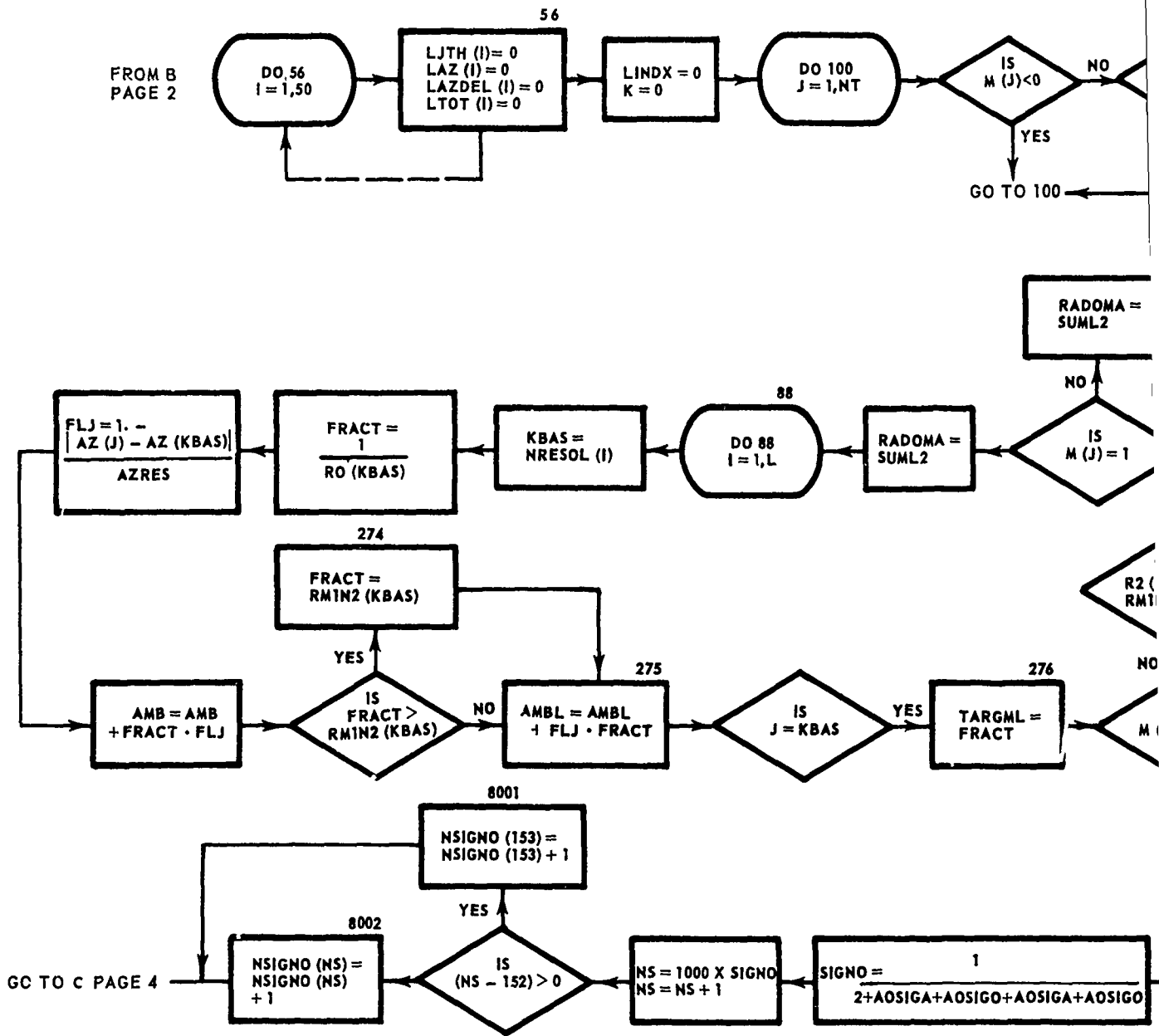


Figure 2-5 (Cont.)

B

FROM B
PAGE 2



GC TO C PAGE 4

A₁

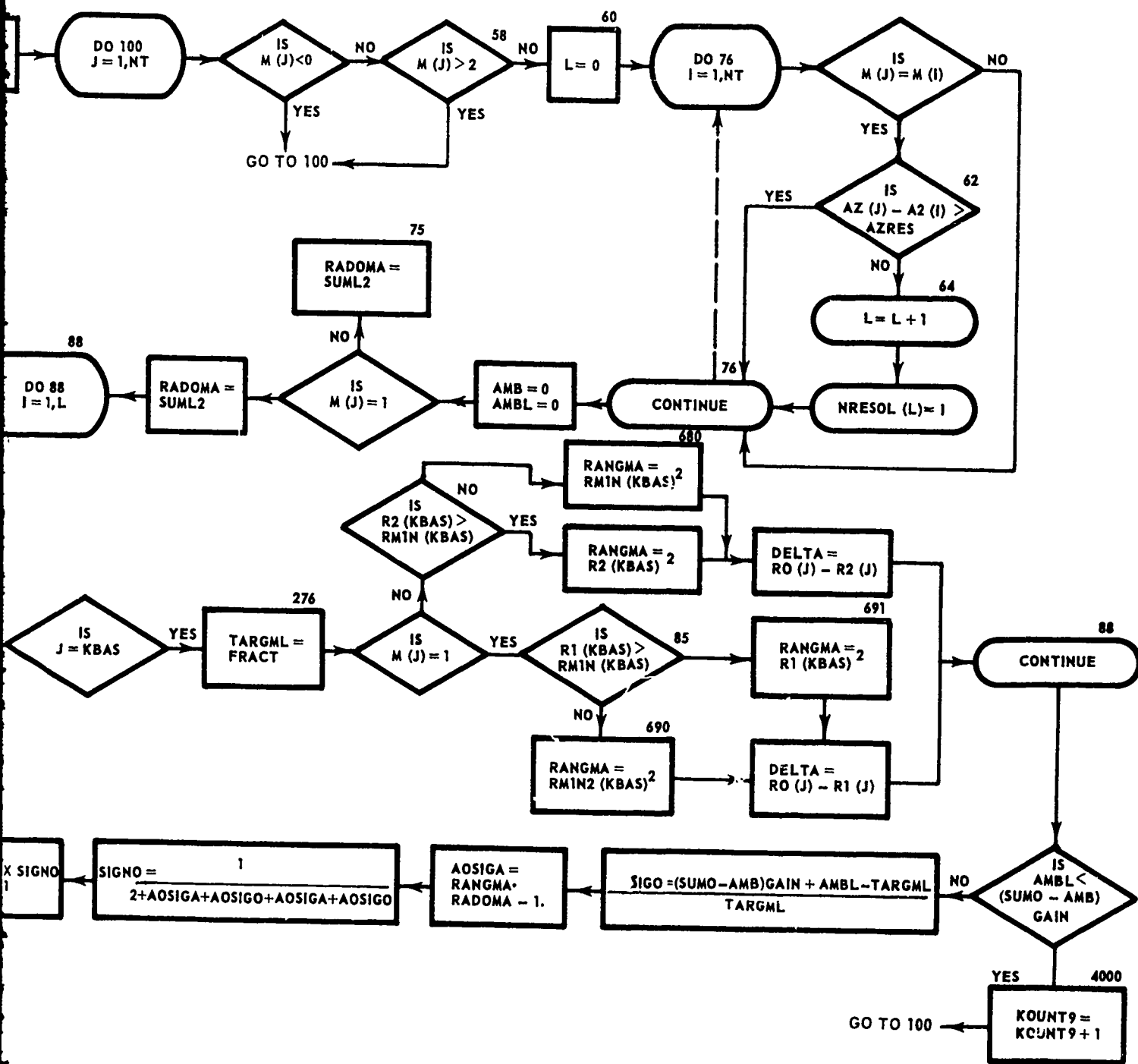
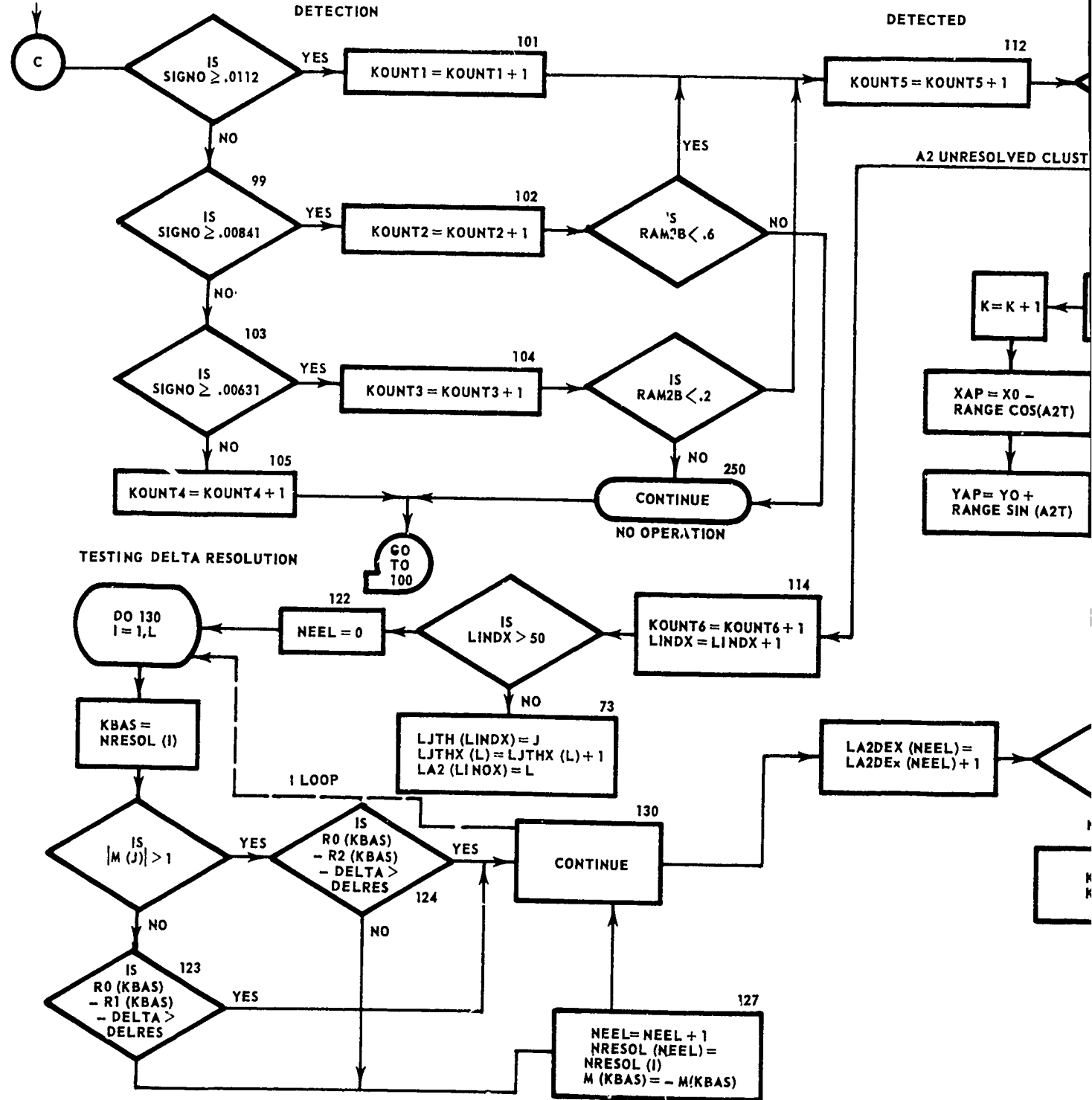


Figure 2-5 (Cont.)

B.

FROM PAGE 3



A.

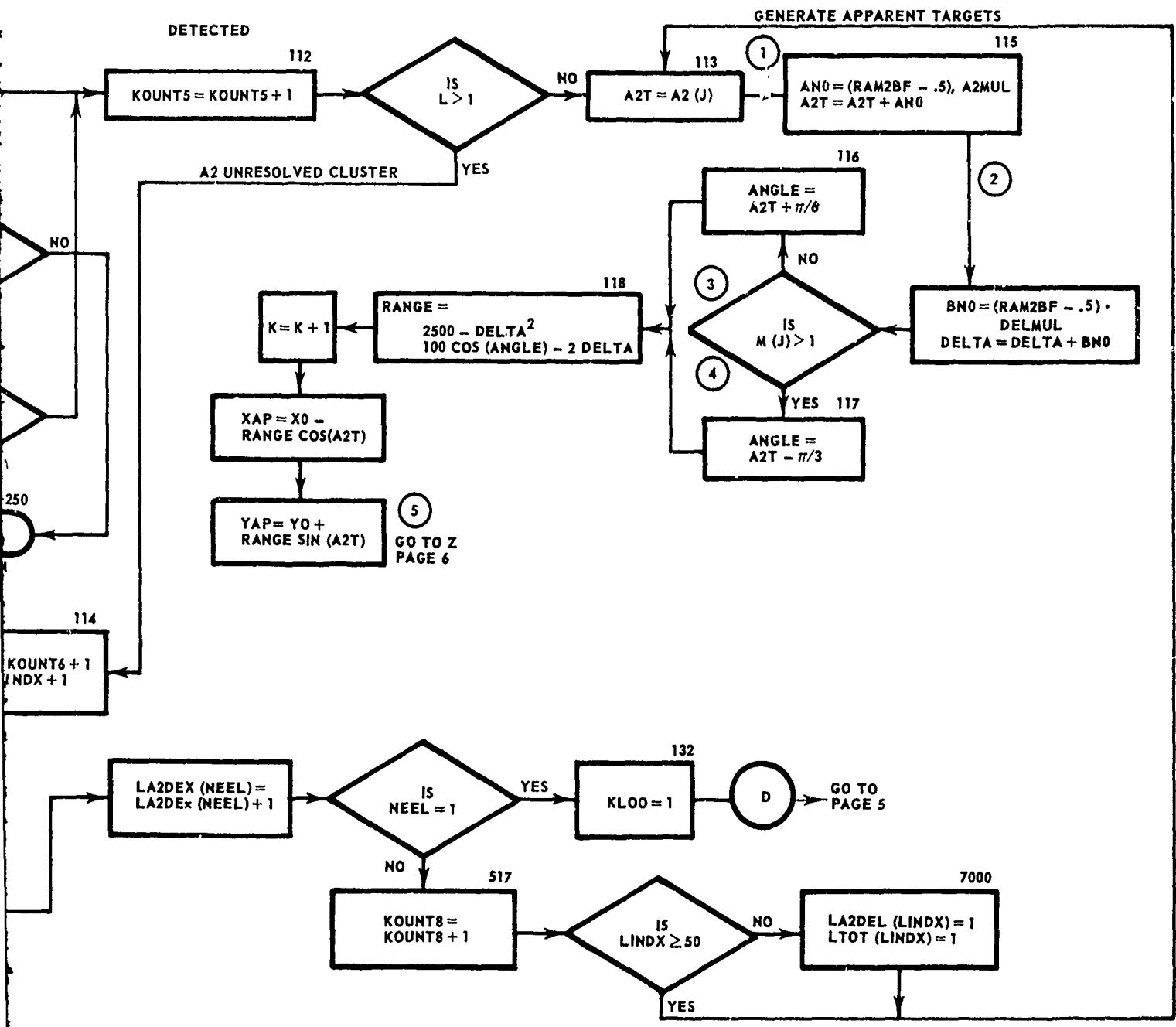
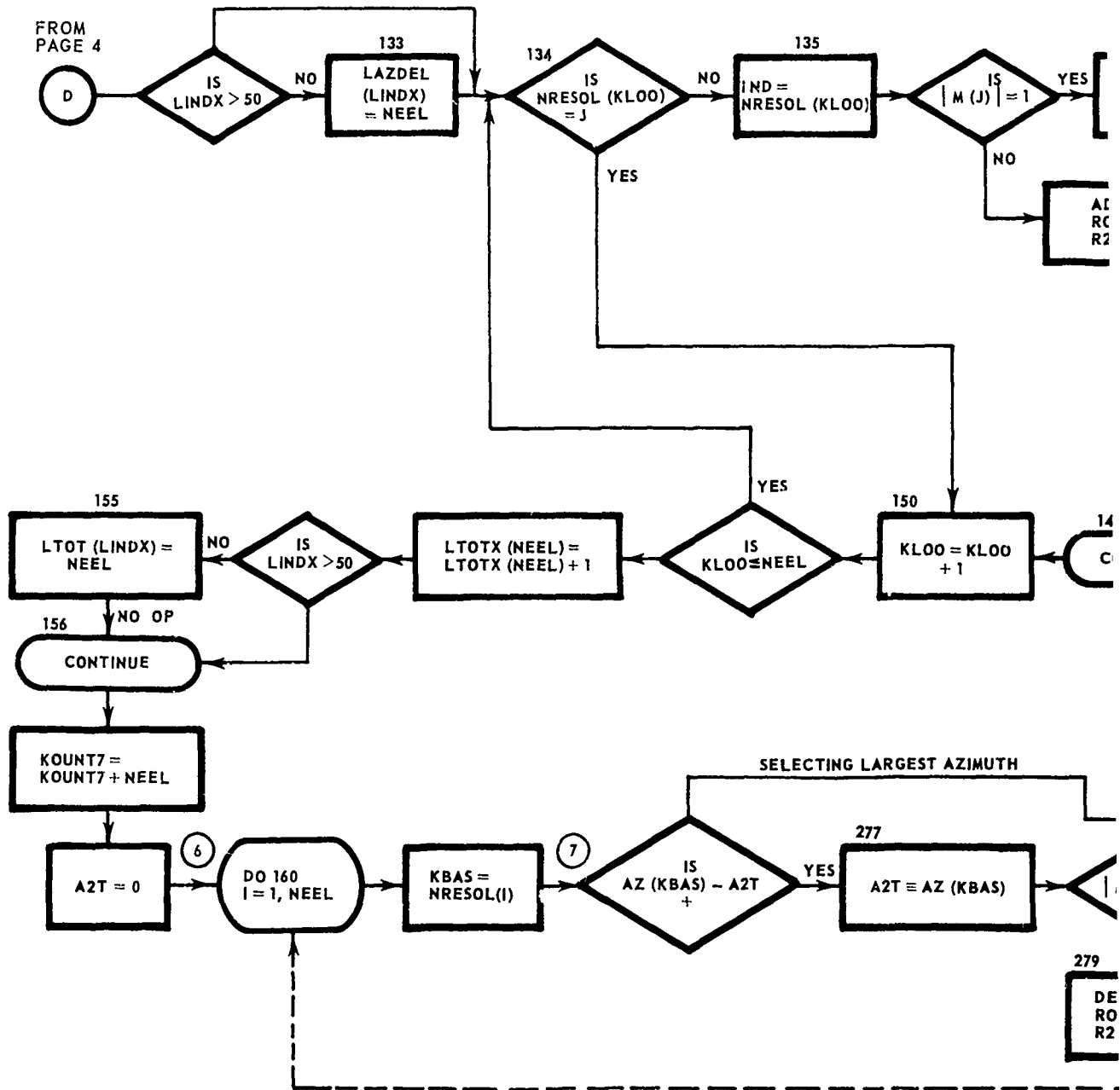


Figure 2-5 (Cont.)

B.

FROM PAGE 4



A₁

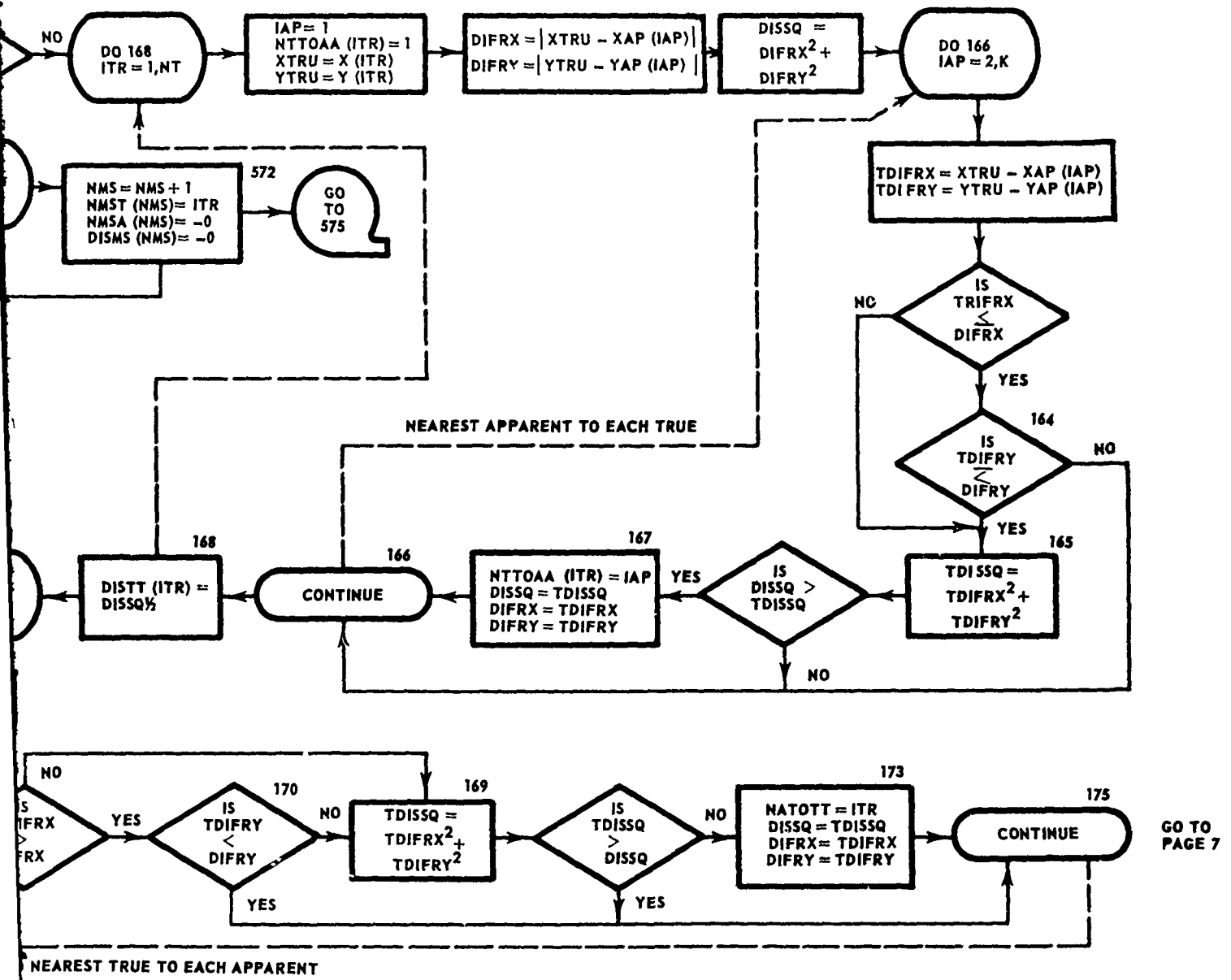
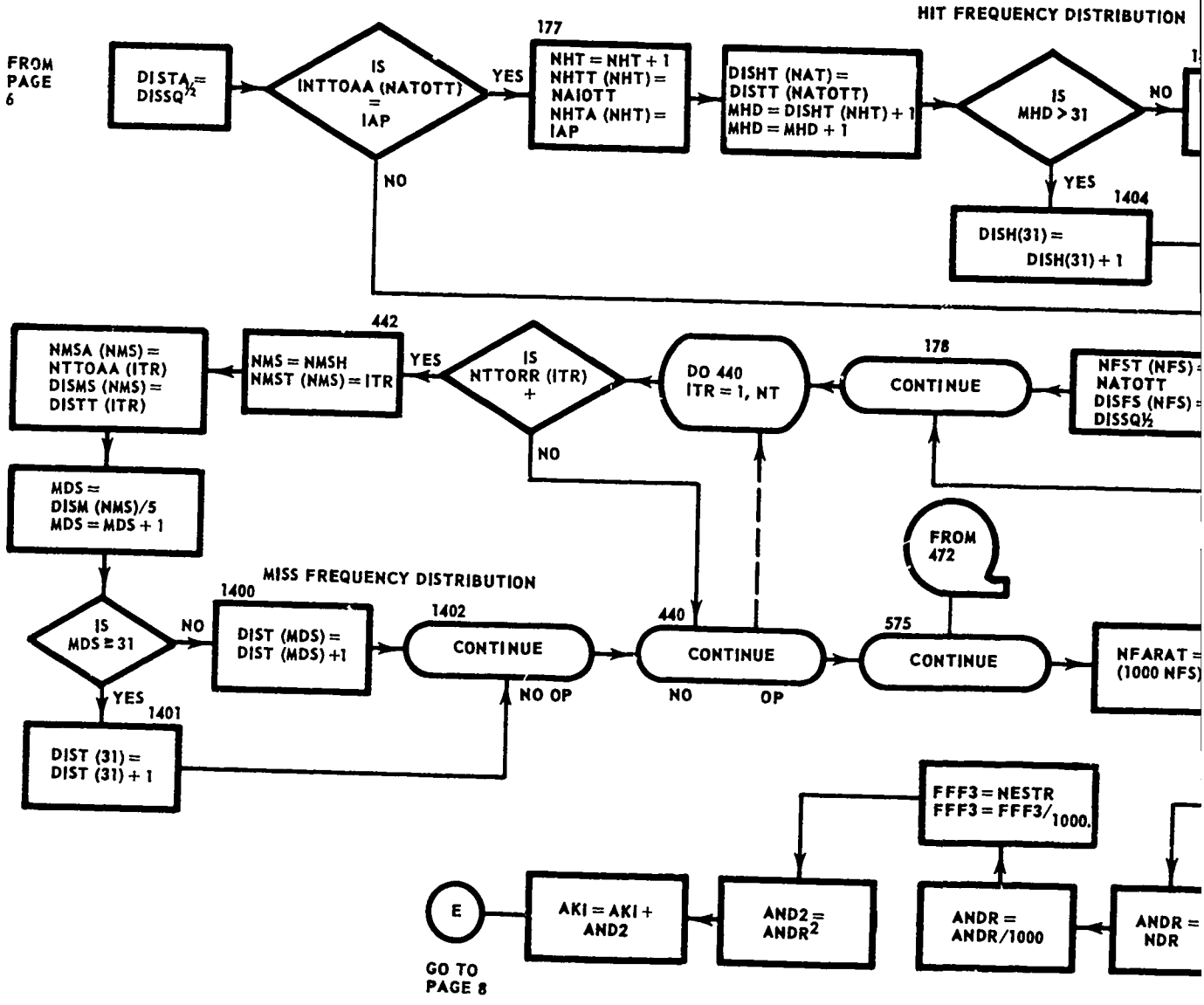


Figure 2-5(Cont.)

B.



A.

HIT FREQUENCY DISTRIBUTION

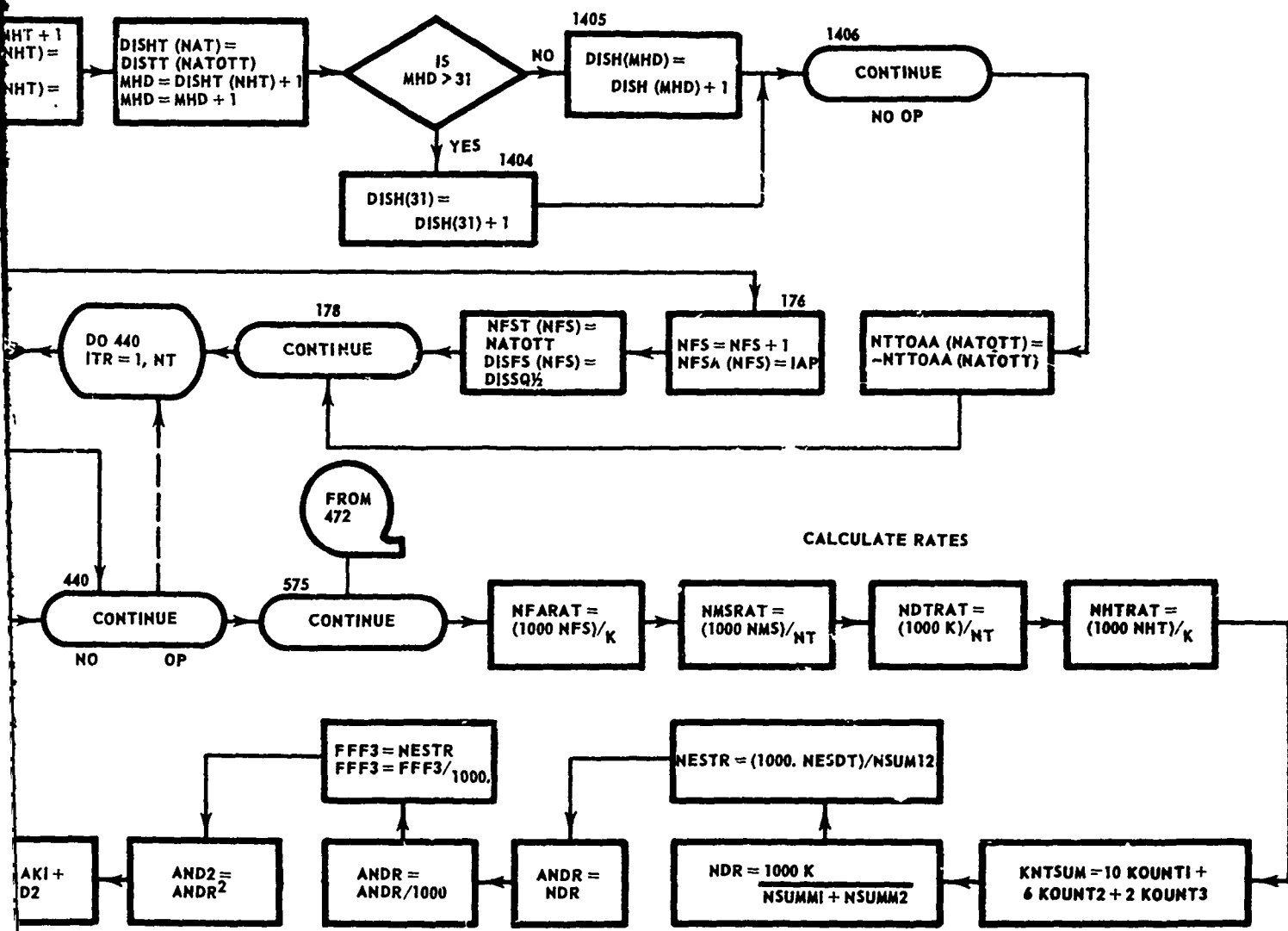


Figure 2-5 (Cont.)

B,

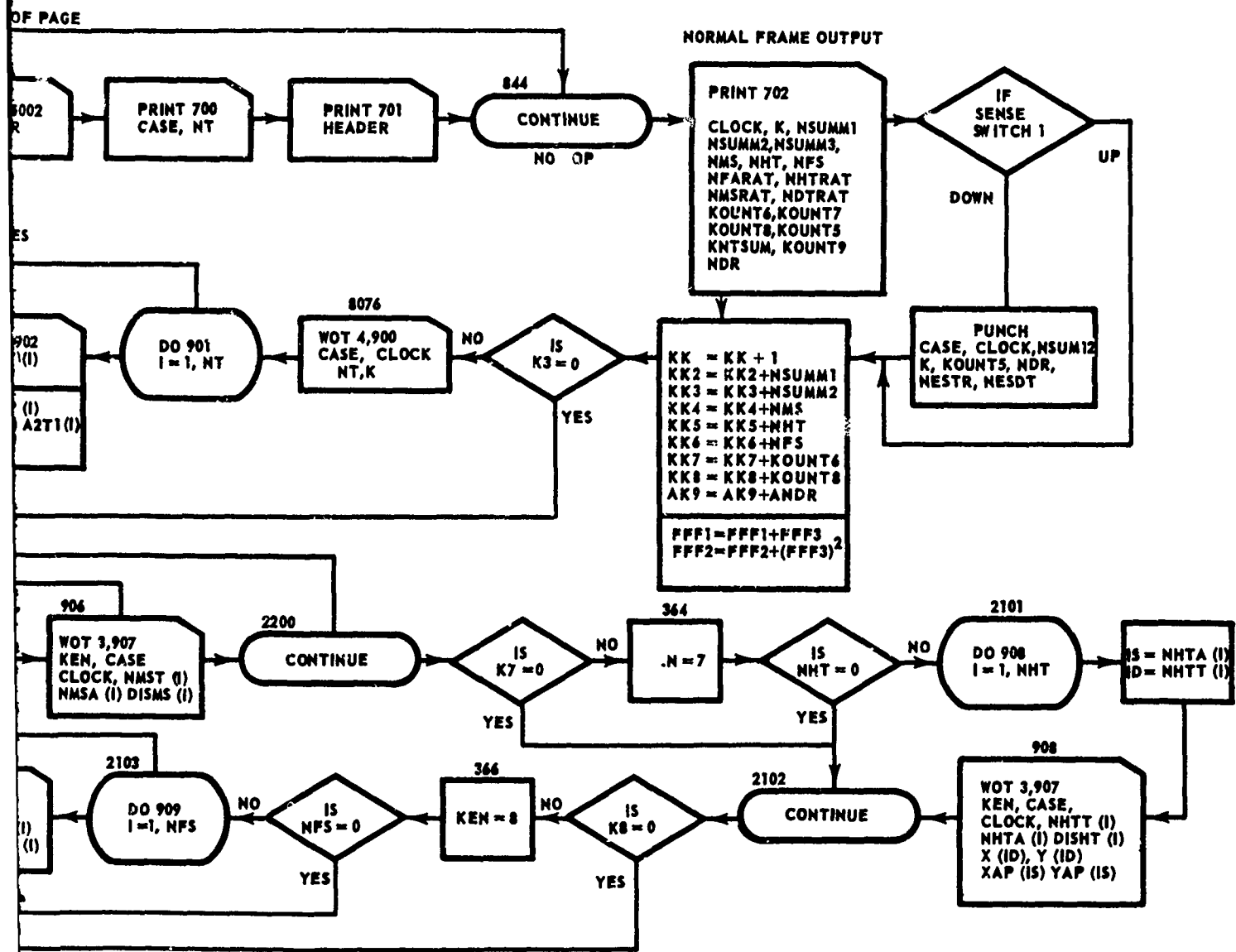


Figure 2-5 (Cont.)

B.

FRAMES ARE DONE

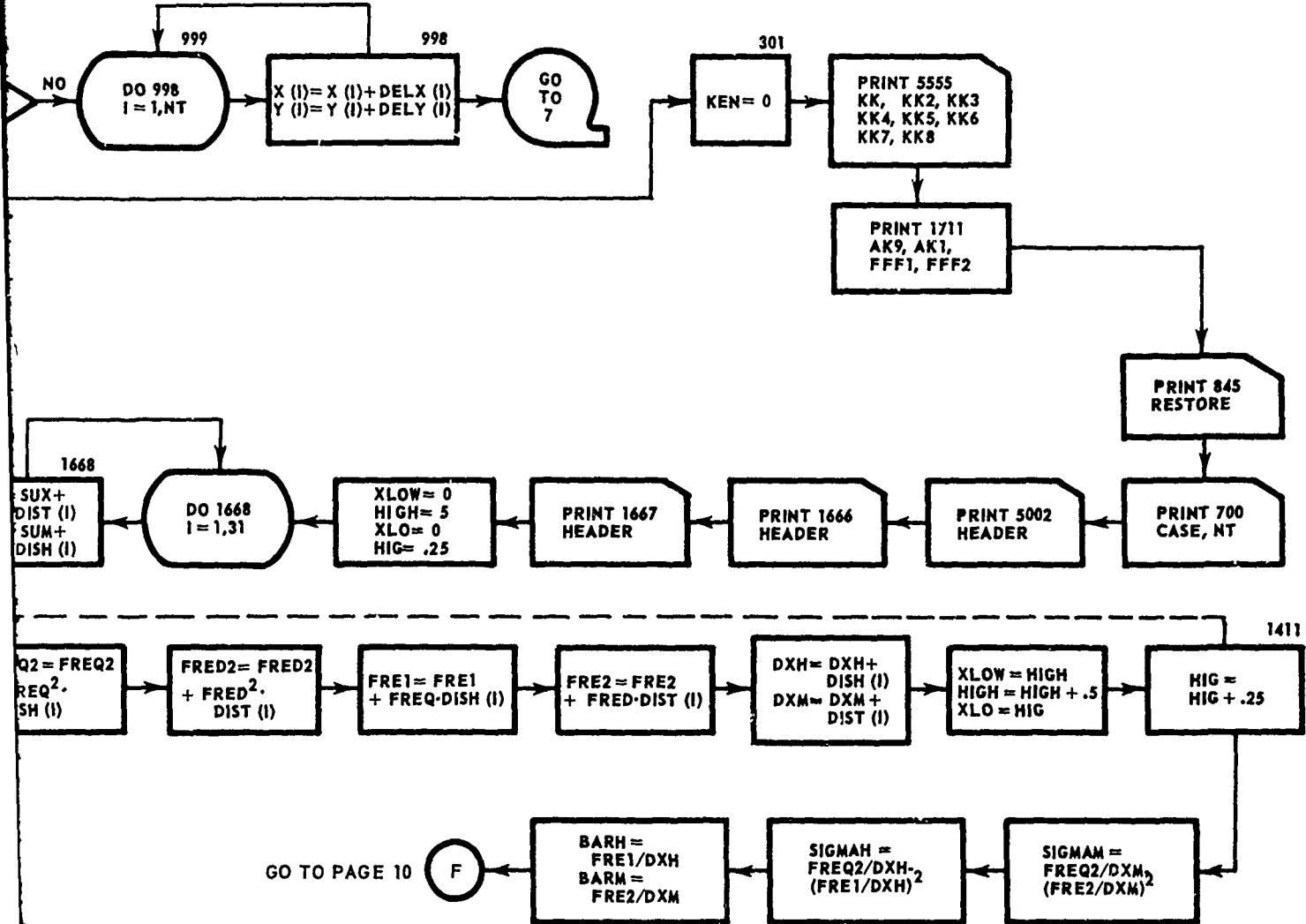


Figure 2-5 (Cont.)

B.

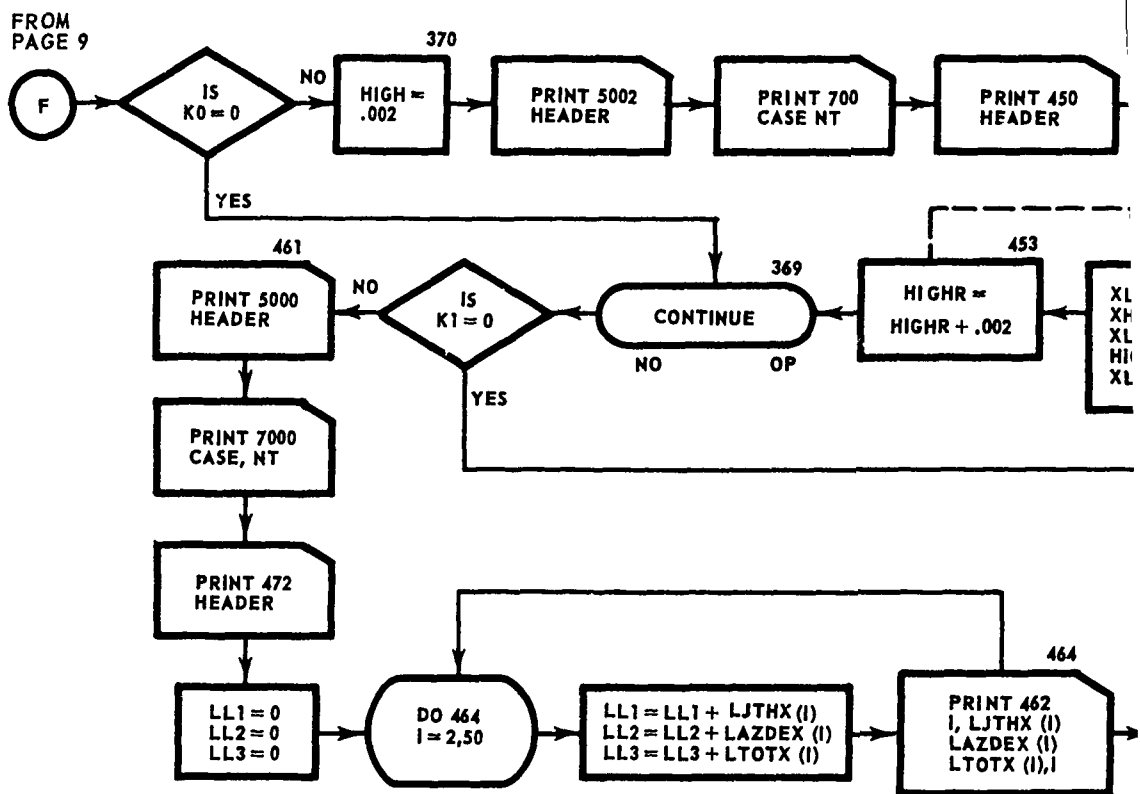


Figure 2.

Ai

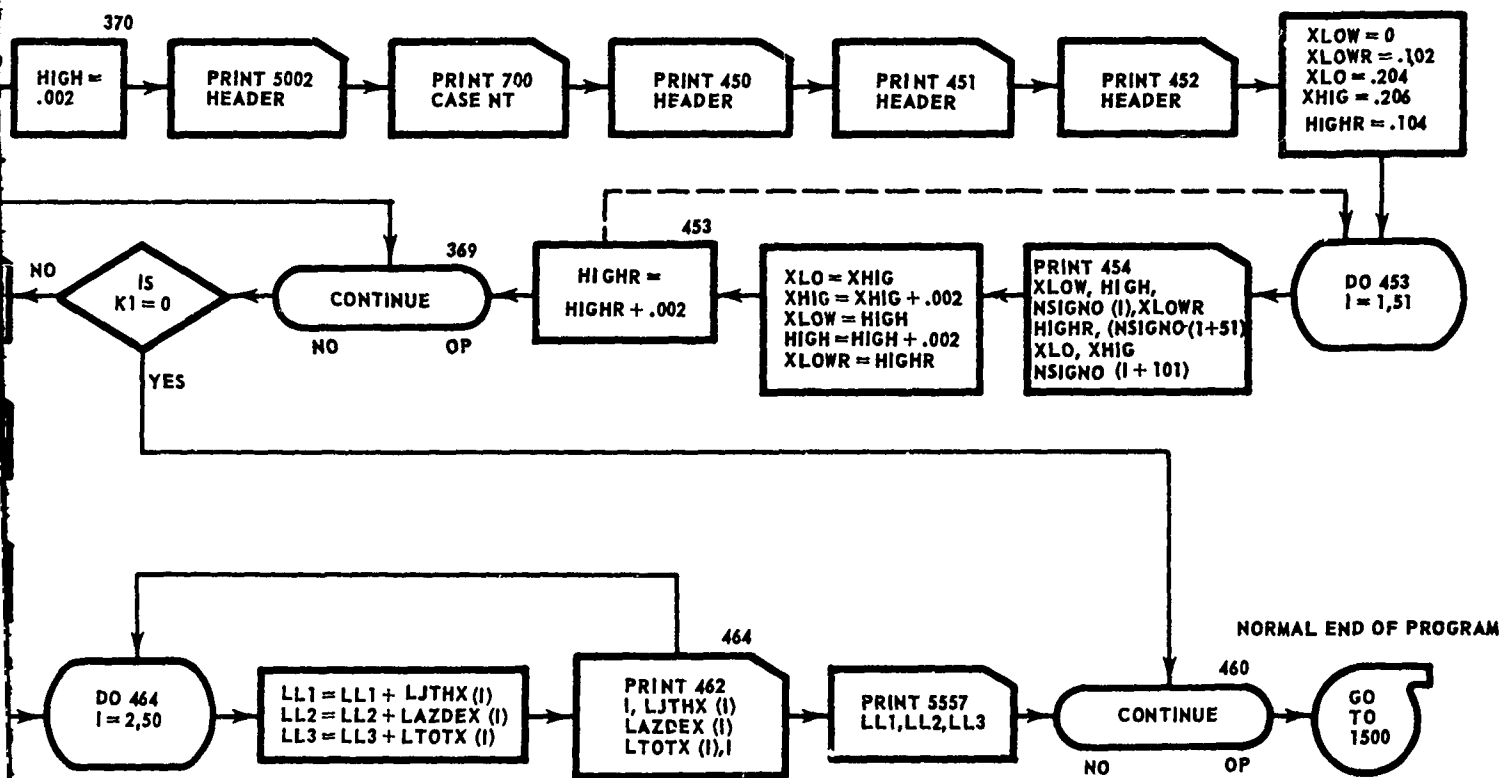


Figure 2-5 (Cont.)

B.

2.6 HISTORICAL DATA PROGRAM

Inputs to the Historical Data Program are from tape and cards. The tape input contains the information to be analyzed. The input cards contains the information designating the specific case and specific frames on tape to be analyzed. At the start and finish of each subprogram, a hollerith card is read and printed. These hollerith cards are used to convey specific operating instructions to the machine operator. The hollerith cards are themselves inputs and care should be taken to keep them in correct order.

1. Card inputs to the continuity subprogram

Card 1) Hollerith Card (read at beginning of analysis)

Card 2) Variable Card CASE1 (F5.0), START1 (F5.0),
END1(F5.0), START3 (F5.0),
END3 (F5.0), NT

Card 3) Hollerith Card (read at end of analysis)

2. Card inputs to the miss distance subprogram

Card 1) Hollerith Card (read at beginning of subprogram)

Card 2) Variable Card CASE2 (F4.0), START2 (F4.0),
END2 (F4.0), RAINTR (F3.0),
AZINTR (F3.0), K1 (12)

Card 3) Hollerith Card (read at end of analysis)

Output from the continuity subprogram will be printed, or punched cards, or tape.

2.6.1 Glossary

A

A dummy storage location used read the X variable from tape while searching tape for desired position where analysis begins

<u>AMDIST</u>	Arithmetic mean, computed on the number of entires in each of the specific RA/AZ blocks for either (1) the missed distance in a "hit" pair, (2) the difference in ranges in a "hit" pair, or (3) the difference in azimuths in a "hit" pair
<u>AZ</u>	True target azimuth
<u>AZAPP</u>	Apparent target azimuth
<u>AZINTR</u>	Number of azimuth intervals used in the miss distance analysis
<u>AZMID</u>	Midpoints of the various azimuth intervals
<u>B</u>	Dummy location, used to store the Y variable while reading tape to locate the correct starting position
<u>C</u>	Dummy location, used to store the XAP variable while reading tape to locate the correct starting position
<u>CASE</u>	Identification number for a case
<u>CASE 1</u>	The case to be analyzed in the continuity subprogram
<u>CASE 2</u>	The case to be analyzed in the miss-distance subprogram
<u>CLOCK</u>	Frame identification
<u>CLOCKA</u>	Frame identification
<u>COUNTR</u>	A counter used to determine (and correct) the possibility of encountering blank frames while processing tapes
<u>D</u>	Dummy location used to store the XAP variable while reading tape to locate the correct starting position

<u>DENOM</u>	The total number of frames processed
<u>DISHT</u>	The distance between a true target and an apparent target in a "hit" pair
<u>DIST</u>	The summation of the specific difference variable (HARRY) for each of the RA/AZ blocks
<u>DIST 1</u>	Square of DIST ($DIST^2$)
<u>DIST 2</u>	This variable is the number of observed targets (XOBSVN) in a RA/AZ section multiplied by the sum of squares of HARRY ($\Sigma (HARRY)^2$)
<u>DISTSQ</u>	Summation of $(HARRY)^2$
<u>E</u>	Dummy location used to store the YAP variable while input tape is searched for correct starting position
<u>END 1</u>	Last frame to be processed with respect to the binary tape output section
<u>END 2</u>	Last frame to be processed with respect to the miss-distance subprogram
<u>END 3</u>	Last frame to be processed with respect to the continuity subprogram analysis
<u>HARRY</u>	The miss-distance subprogram is capable of performing analysis in either the three separate modes (i. e., (1) the frequency analysis is performed with respect to the distance between the true and the apparent target in a "hit" pair (2) the frequency analysis is performed with respect to the difference in ranges between the true and apparent in a "hit" pair (3) the frequency analysis is performed with respect to the differences in ranges between the true and apparent target in a "hit" pair); HARRY serves as a common location for each of these cases

II The number of range intervals (RAINTR) used as a DO LOOP limit

JJ The number of azimuth intervals (AZINTR) used as a DO LOOP limit

KEN Identification number for different types of historical data which were written previously on tape 3 by TLQ-8 program

KMAX Number of detections per frame

K1 Control constant used to control the specific analytic modes

1) K1 = 1 analysis is performed on DISHT

2) K1 = 2 analysis is performed on "difference in ranges" (see HARRY)

3) K1 = 3 analysis is performed on "difference in azimuths" (see HARRY)

MCASE Same as CASE

MF The number of frames encountered during the tape analysis in which no detections occurred

MISS The total number of frames missed per target

N A variable used to record runs of length "N" where N is the number of consecutive missed frames ($0 \leq N \leq 15$)

NFREQ The number of times a specific target was missed for a length of "N" consecutive frames

NHTA The number of the apparent target in a "hit" pair

NHTT A counter used to tabulate the number of consecutive missed frames for the individual targets

<u>NT</u>	Number of targets
<u>N1</u>	Dummy location used to store the variable NHTT while searching tape for the correct tape
<u>N2</u>	Dummy location used to store the variable NHTA while searching tape for the correct tape
<u>RA</u>	True target range
<u>RAMID</u>	Midpoints of the range intervals
<u>RAINTR</u>	Number of range intervals used in the miss distance subprogram
<u>RAPP</u>	Range of the apparent target
<u>RATE</u>	The ratio of the total number of times a target was missed to the maximum number of misses possible
<u>RAIVSZ</u>	The specific size of the range intervals
<u>STADEV</u>	Standard deviation
<u>START 1</u>	Starting frame for the binary tape output
<u>START 2</u>	Starting frame for analysis in the miss distance subprogram
<u>START 3</u>	Starting frame for the continuity analysis
<u>TOP</u>	See MISS
<u>X</u>	X coordinate of the true target
<u>XAP</u>	X coordinate of the apparent target
<u>XOBSVN</u>	The number of targets observed in a specific RA/AZ interval

XOBSII

(XOBSVN)(XOBSVN-1)

Y

Y coordinate of the true target

YAP

Y coordinate of the apparent target

2.6.2 FORTRAN Listing

```
DIMENSION NMISS(60),NFREQ(60,15),NHIT(60),NHIA(60),DISHT(60),X(60)
1,Y(60),XAP(60),YAP(60),RA(60),AZ(60),RAPP(60),AZAPP(60),XOBSVN(25,
218),DIST(25,18),DISTSQ(25,18),RAMID(18),AZMID(25),AMDIST(60),
3VARNCE(60),STADDEV(60),M(60)
5 READ 6
6 FORMAT(72H
X
PRINT 6
PAUSE 1
IF(SENSE SWITCH2)20,10
20 REWIND 3
END FILE 4
REWIND 4
GO TO 5
10 IF(SENSE SWITCH 1)40,30
30 READ 31,CASE1,START1,END1,START3,END3,NT
31 FORMAT(5F5.0,13)
COUNTR=START1
MF=0
DO 50 J=1,NT
M(J)=0
NMISS(J)=0
DO 50 N=1,15
50 NFRFO(J,N)=0
SENSE LIGHT 1
SENSE LIGHT 2
SENSE LIGHT 3
SENSE LIGHT 4
I=1
60 READ INPUT TAPE 3,61,KEN,CASE,CLOCKA,N1,N2,A,B,C,D,E
61 FORMAT(15,F10.5,F5.0,2I8,5F8.2)
62 IF(SENSE LIGHT 1)65,75
65 SENSE LIGHT 1
IF(CASE1-CASE)71,70,71
```

```

71 IF(KEN-7)72,60,72
72 PRINT 73,KEN,CLOCKA
73 FORMAT(39H KEN ISNOT EQUAL TO 7. CHECK TAPE. KEN=I3,7H CLOCK=I4)
  PAUSE 2
70 IF(SENSE LIGHT 1)75,75
75 IF(SENSE LIGHT 2)80,85
85 IF(SENSE LIGHT 3)402,403
402 SENSE LIGHT 3
  GO TO 401
80 SENSE LIGHT 2
  IF(START1-CLOCKA)400,400,71
400 IF(SENSE LIGHT 2)401,401
401 IF(START3-CLOCKA+1.)81,416,87
81 MF=CLOCKA-START3
  DO 82 K=1,MF
  DO 82 J=1,NT
82 NMISS(J)=NMISS(J)+1
  COUNTR=COUNTR+FLOATF(MF)
  MF=0
416 IF(SENSE LIGHT 3)403,403
403 IF(CASE1-CASE)92,87,92
92 IF(END3-CLOCK)93,87,93
93 MF=END3-CLOCK
87 IF(I-1)89,88,89
88 CLOCK=CLOCKA
89 IF(CLOCKA-CLOCK)95,91,95
95 COUNTR=COUNTR+1.
  IF(CLOCKA-COUNTR)96,90,96
96 MF=CLOCKA-COUNTR
  COUNTR=COUNTR+FLOATF(MF)
  GO TO 90
91 NHTT(I)=N1
  NHTA(I)=N2
  DISHT(I)=A
  X(I)=B
  Y(I)=C
  XAP(I)=D
  YAP(I)=E
  I=I+1
  GO TO 60
90 KMAX=I-1
  WRITE TAPE 4,CASE1,CLOCK,KMAX
  WRITE TAPE 4,(X(I),Y(I),XAP(I),YAP(I),DISHT(I),I=1,KMAX)
  IF(SENSE LIGHT 3)412,404
412 SENSE LIGHT 3
  GO TO 55

```

```

404 IF(SENSE LIGHT 4)407,101
407 SENSE LIGHT 4
405 DO 150 I=1,KMAX
      J=NHTT(I)
      IF(NMISS(J))150,150,146
146 N=NMISS(J)
      IF(N-15)148,148,147
147 MM=N-15
      M(J)=M(J)+MM
      N=15
148 NFREQ(J,N)=NFREQ(J,N)+1
150 NMISS(J)=-1
      DO 155 J=1,NT
155 NMISS(J)=NMISS(J)+1
      IF(MF)102,406,98
102 PRINT 103,COUNTR,CLOCKA
103 FORMAT(24H MF IS NEGATIVE. COUNTR=F4.0,8H CLOCKA=F4.0)
      PAUSE 11
      98 DO 99 K=1,MF
          DO 99 J=1,NT
          99 NMISS(J)=NMISS(J)+1
          MF=0
406 IF(END3-CLOCKA)408,101,101
408 IF(SENSE LIGHT 4)101,101
101 IF(CASE1-CASE)165,156,165
156 IF(END1-CLOCKA)165,55,55
55 I=1
      NHTT(I)=N1
      NHTA(I)=N2
      DISHT(I)=A
      X(I)=B
      Y(I)=C
      XAP(I)=D
      YAP(I)=E
      GO TO 62
165 DO 170 J=1,NT
      IF(NMISS(J))170,170,166
166 N=NMISS(J)
      IF(N-15)168,168,167
167 MM=N-15
      M(J)=M(J)+MM
      N=15
168 NFREQ(J,N)=NFREQ(J,N)+1
170 CONTINUE
      DENOM=END3-START3+1.
      PRINT 171,CASE1,DENOM
171 FORMAT(17H CASE ANALYZED=F4.0,26H MAXIMUM NO. OF MISSES IS F4.0)

```

```

PRINT 172
172 FORMAT(120H TARGET NO.  MISSES  RATE  RUNS OF LENGTH N= 1  2
X 3  4  5  6  7  8  9  10  11  12  13  14  15 TOTAL MISSES)
MCASF=CASE
DO 176 J=1,NT
MISS=0
DO 175 N=1,15
175 MISS=MISS+(N*NFREQ(J,N))
TOP=MISS+M(J)
RATE=TOP/DENOM
PRINT 173,J,MISS,RATE,(NFREQ(J,N),N=1,15),M(J)
173 FORMAT(I10,I8,F9.3,I24,I4I4,I9)
176 PUNCH174,MCASE,J,MISS,RATE,(NFREQ(J,N),N=1,15)
174 FORMAT(I3,I2,I3,F4.2,I5I4)
GO TO 5
40 READ 41,CASE2,START2,END2,RAINTR,AZINTR,K1
41 FORMAT(3F4.0,2F3.0,I2)
RAIVSZ= 250./RAINTR
AZIVSZ=3.1415927/AZINTR
IF(SENSE SWITCH 4)410,411
411 DO 178 I=1,25
DO 178 J=1,18
XOBSVN(I,J)=0.
DIST(I,J)=0.
178 DISTSQ(I,J)=0.
410 SENSE LIGHT 3
SENSE LIGHT 4
K1=K1-2
179 READ TAPE 4,CASE ,CLOCK,KMAX
IF(SENSE LIGHT 3)185,180
185 SENSE LIGHT 3
IF(CASE2-CASE)195,190,195
190 IF(SENSE LIGHT 3)205,205
195 READ TAPE 4,A,B,C
GO TO 179
180 IF(CASE2-CASE)200,205,200
200 PRINT 201,CASE,CASE2,CLOCK
201 FORMAT(94H CASE NO. HAS BEEN RECORDED WRONG. ERROR COULD BE IN SEA
XRCHING OR RECORDING TAPE 3 OR 4. CASE=F4.0,7H CASE2=F4.0,7H CLOCK=
XF4.0)
PAUSE 3
205 IF(SENSE LIGHT 4)206,210
206 SENSE LIGHT 4
IF(CLOCK-START2)195,207,207
207 IF(SENSE LIGHT 4)210,210
210 READ TAPE 4,(RA(K),AZ(K),RAPP(K),AZAPP(K),DISHT(K),K=1,KMAX)

```

```

      IF(SENSE SWITCH 3) 392,393
392 DO 390 K=1,KMAX
390 PRINT 391,RA(K),AZ(K),RAPP(K),AZAPP(K),DISHT(K)
391 FORMAT(5F10.3)
393 IF(KMAX) 220,242,220
220 DO 230 K=1,KMAX
      DIFX=RA(K)-500.
      DIFY=AZ(K)-500.
      RA(K)=SQRTF(DIFX**2+DIFY**2)
      AZ(K)=ATN1F(DIFY,-DIFX)
      DIFX=RAPP(K)-500.
      DIFY=AZAPP(K)-500.
      RAPP(K)=SQRTF(DIFX**2+DIFY**2)
230 AZAPP(K)=ATN1F(DIFY,-DIFX)
      DO 240 K=1,KMAX
      J=(RA(K)/RAIVSZ)+1.
      I=(AZ(K)/AZIVSZ)+1.
      XOB SVN(I,J)=XOB SVN(I,J)+1.
      IF(K) 231,232,233
231 HARRY=DISHT(K)
      GO TO 234
233 HARRY= ABSF(AZ(K)-AZAPP(K))
      GO TO 234
232 HARRY=ABSF(RA(K)-RAPP(K))
234 DIST(I,J)=DIST(I,J)+HARRY
      IF(SENSE SWITCH 3) 300,240
300 PRINT 301,KMAX,J,I,XOB SVN(I,J),RA(K),RAPP(K),AZ(K),AZAPP(K),HARRY
301 FORMAT(3I4,6F10.3)
240 DIST SQ(I,J)=DIST SQ(I,J)+HARRY**2
242 IF(END2-CLOCK) 243,245,179
243 PRINT 244,CLOCK,CASE,END2
244 FORMAT(56H ENDING FRAME NO. HAS BEEN BYPASSED. CHECK TAPES. CLOCK=
      XF4.0,6H CASE=F4.0,6H END2=F4.0)
      PAUSE 4
245 II=RAINTR
      JJ=AZINTR
      IF(SFNSE SWITCH 5) 5,808
808 RAMID(1)=RAIVSZ/2.
      AZMID(1)=AZIVSZ/2.
      DO 250 I=2,II
250 RAMID(I)=RAMID(I-1)+RAIVSZ
      DO 255 I=2,JJ
255 AZMID(I)=AZMID(I-1)+AZIVSZ
      IF(K) 260,261,262
260 PRINT 263

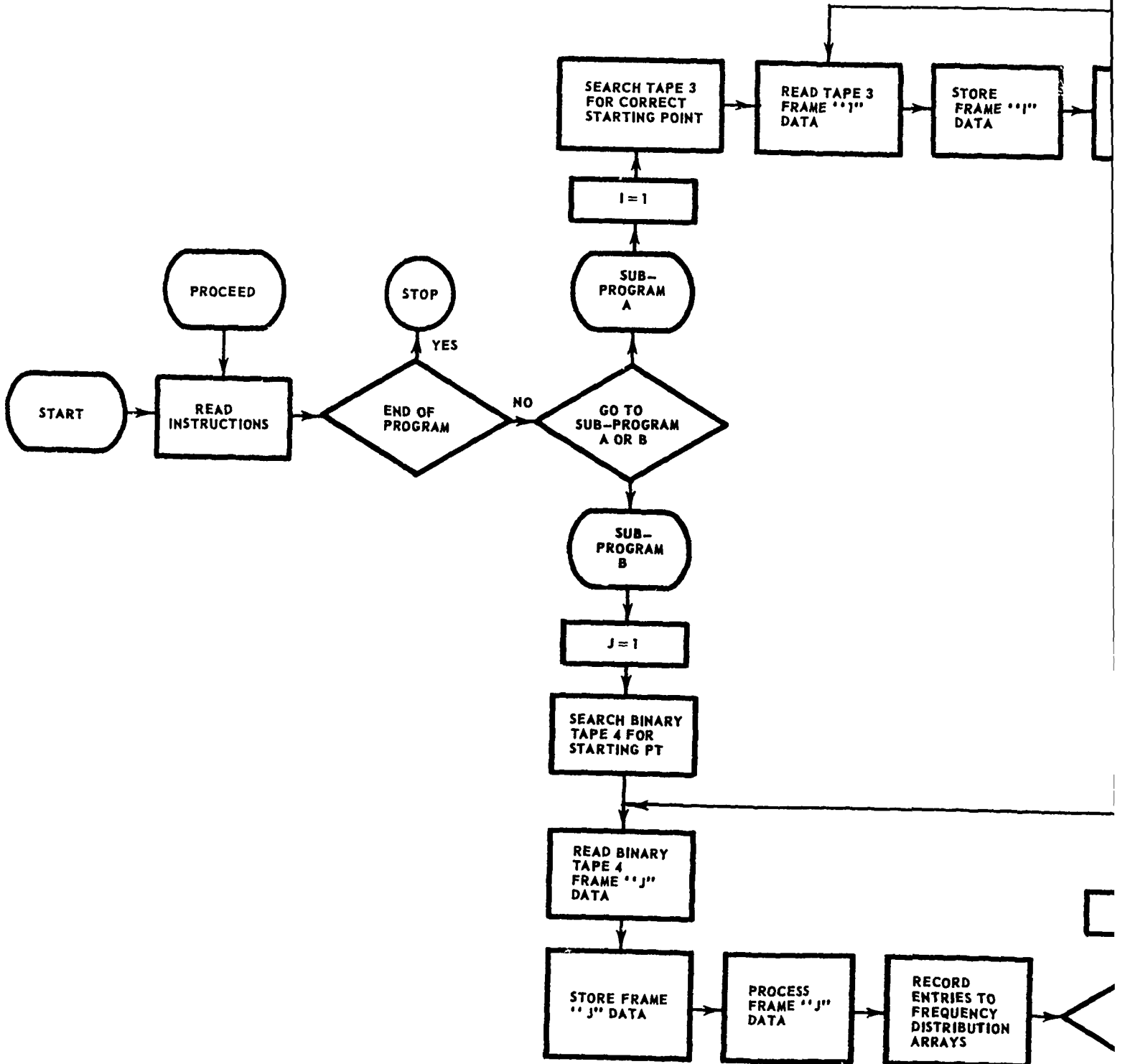
```

```

263 FORMAT(62H MISS DISTANCE ANALYSIS
X RANGE)
GO TO 269
261 PRINT 264
264 FORMAT(62H RANGE ERROR ANALYSIS
X RANGE)
GO TO 269
262 PRINT 265
265 FORMAT(62H AZIMUTH ERROR ANALYSIS
X RANGE)
269 PRINT 270,(RAMID(I),I=1,II)
270 FORMAT( 9H AZIMUTH  F6.1,17F6.1)
DO 285 I=1,II
DO 271 J=1,II
AMDIST(J)=DIST(I,J)/XOBSVN(I,J)
DIST1= DIST(I,J)**2
DIST2= XOBSVN(I,J)*DISTSQ(I,J)
XOBS11=XOBSVN(I,J)*(XOBSVN(I,J)-1.)
VARNCE(J)=(DIST2-DIST1)/XOBS11
STADEV(J)=SQRTF(VARNCE(J))
IF(SENSE SWITCH 3)271,289
289 PUNCH 272,CASE2,AZMID(I),RAMID(J),XOBSVN(I,J),AMDIST(J),STADEV(J),
X VARNCE(J),DIST1,DIST2
271 CONTINUE
272 FORMAT(F5.0,F6.3,F7.1,F6.0,3F7.3,2F10.2)
AZMID(I)=AZMID(I)/.0174533
PRINT 273,((XOBSVN(I,J)),J=1,II)
273 FORMAT(F13.0,17F6.0)
IF(K1)274,274,278
274 PRINT 275,AZMID(I),(AMDIST(J),J=1,II)
275 FORMAT(F6.1,F10.3,17F6.3)
PRINT 276,(STADEV(J),J=1,II)
276 FORMAT(F16.3,17F6.3)
GO TO 285
278 PRINT 279,AZMID(I),(AMDIST(J),J=1,II)
279 FORMAT(F7.1,F10.4,17F6.4)
280 PRINT 281,(STADEV(J),J=1,II)
281 FORMAT(F16.4,17F6.4)
285 CONTINUE
GO TO 5
END(0,1,0,0,1)

```

This page was left intentionally blank.



A.

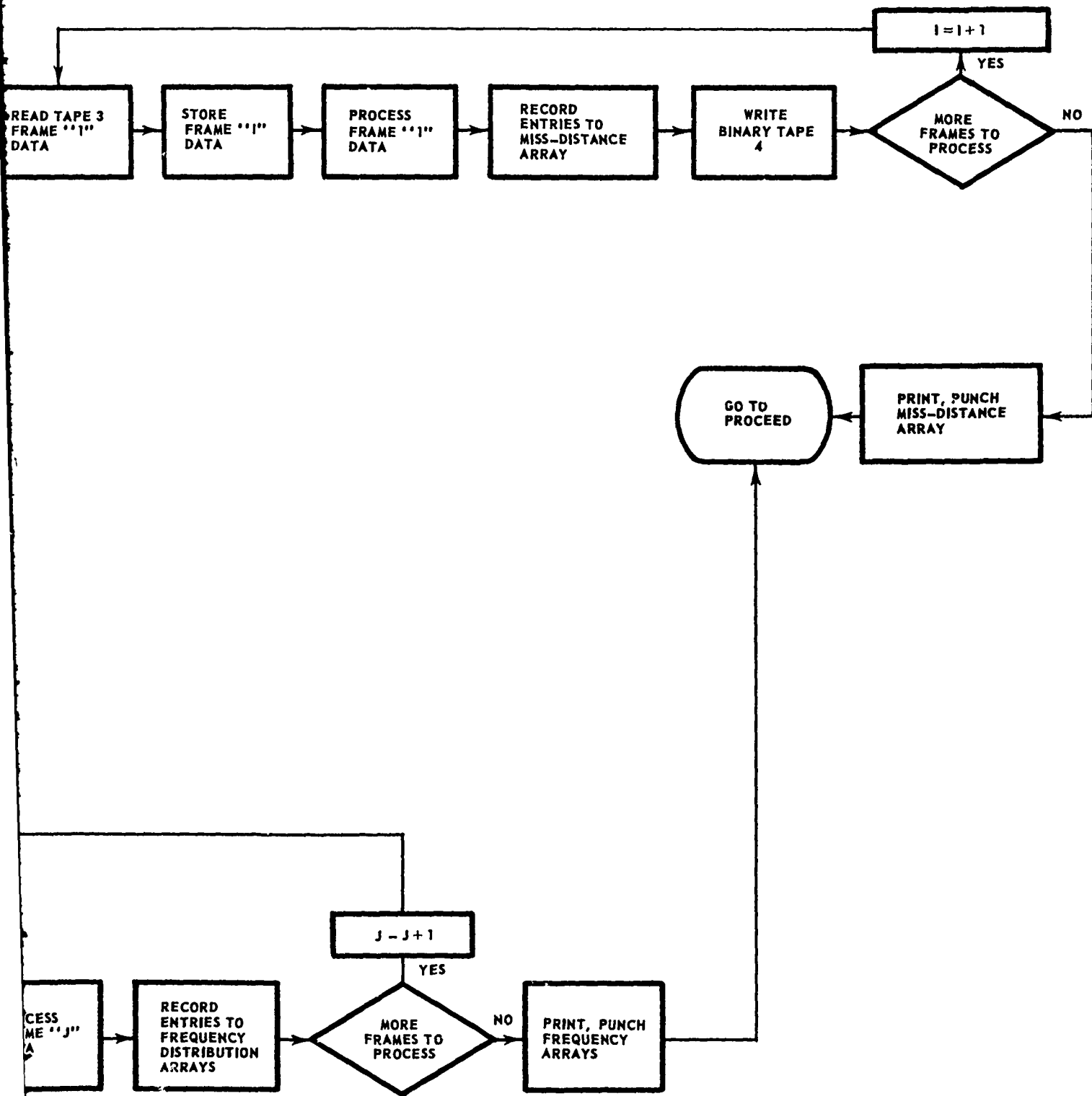
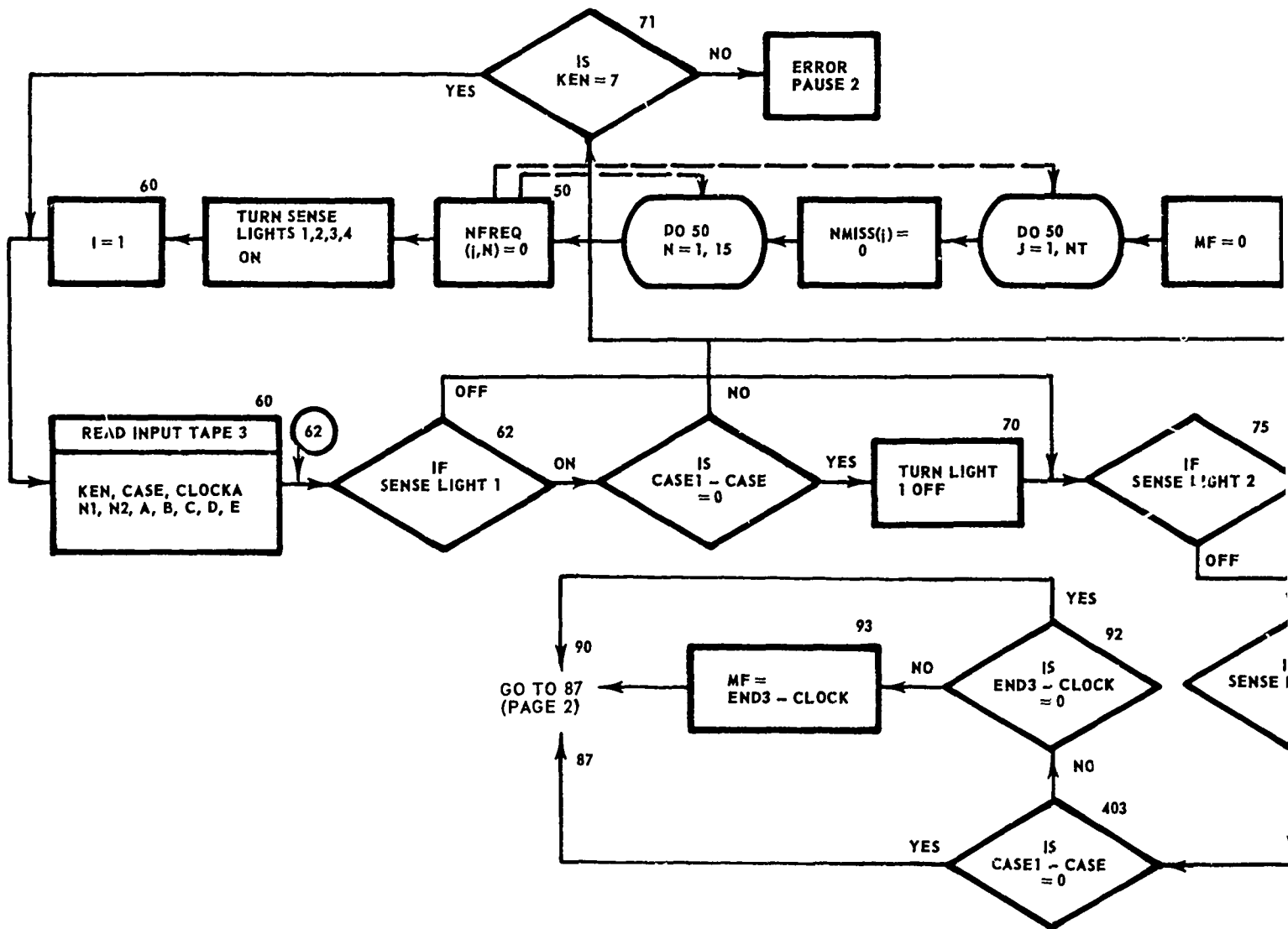
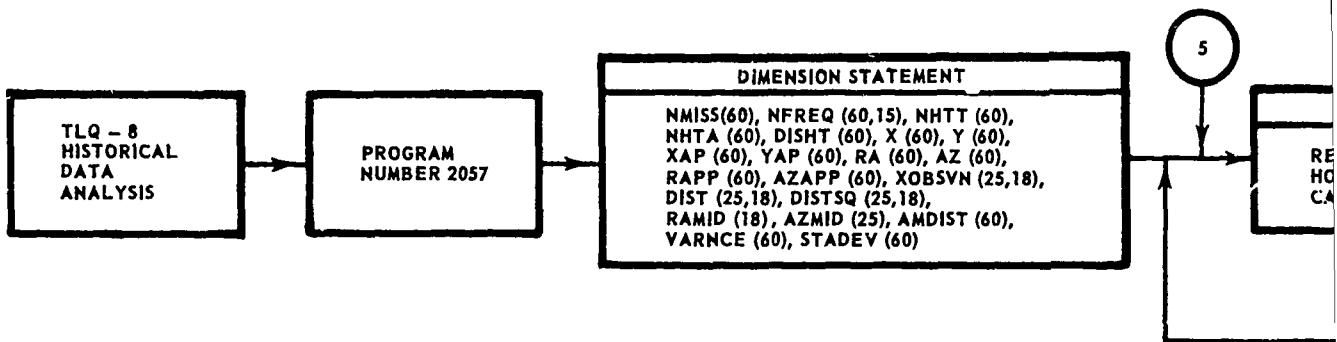


Figure 2-6 General Flow Charts

B.



A.

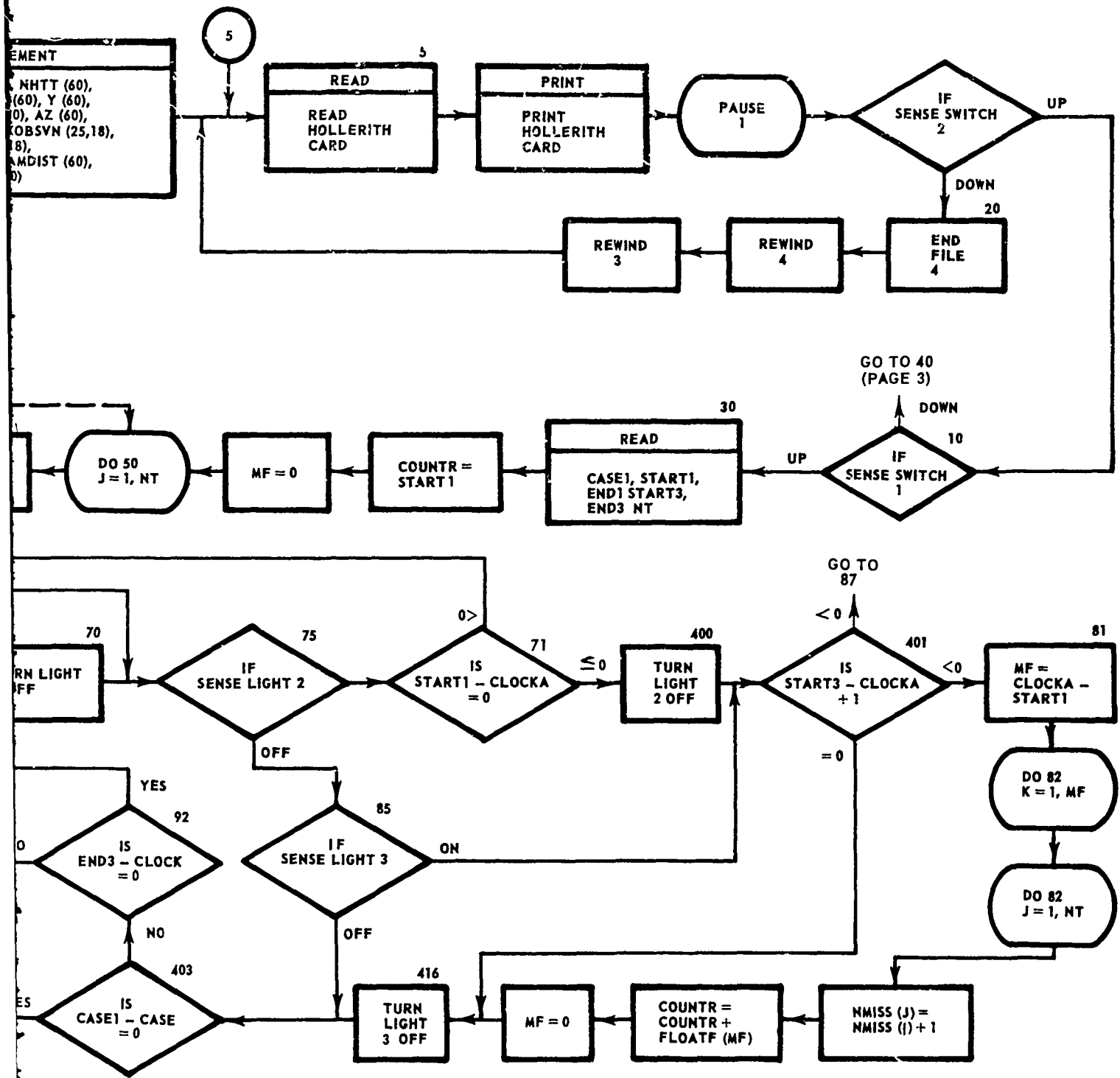
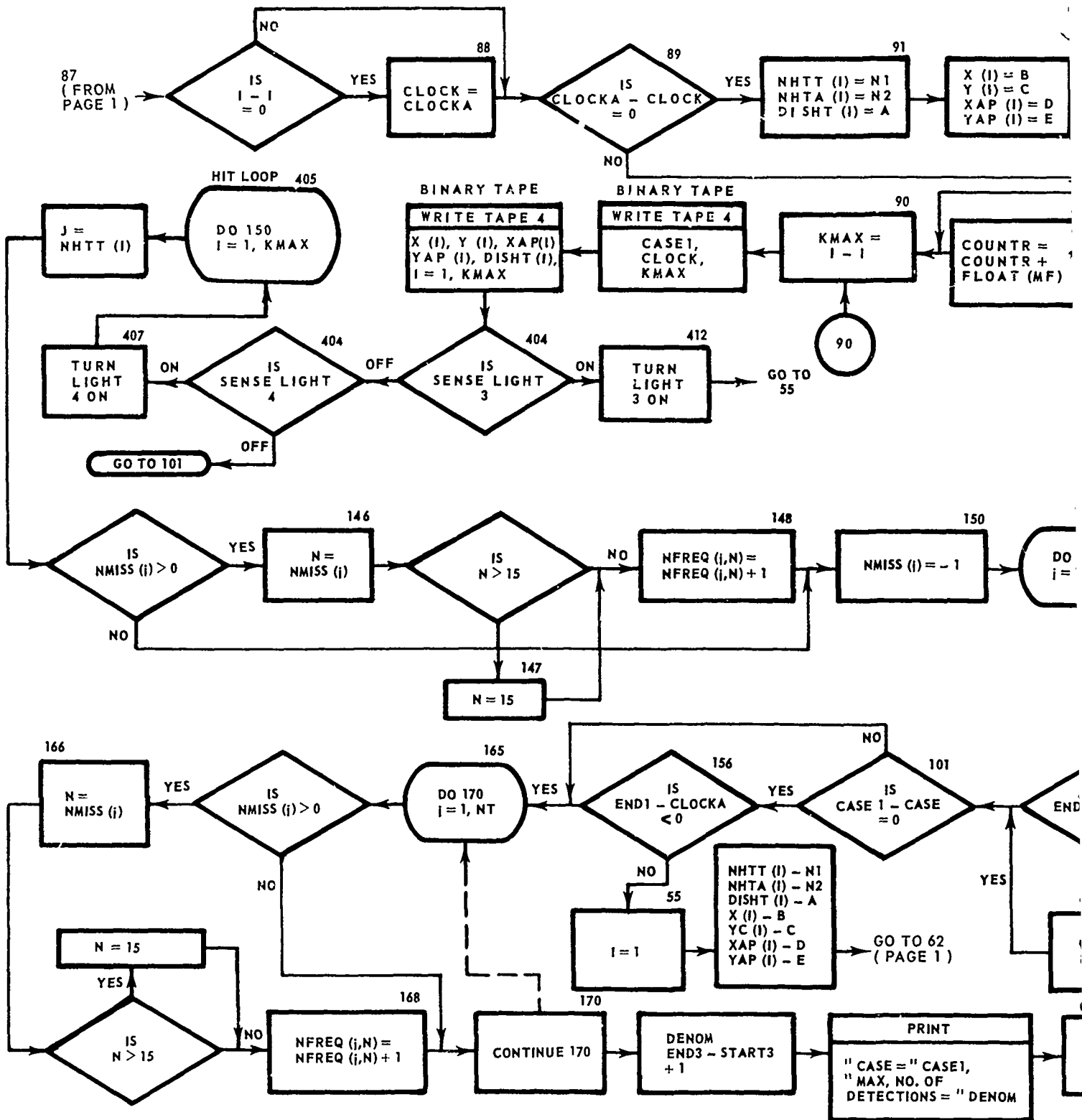


Figure 2-7 Historical Data Analysis Flow Charting

B.



A1

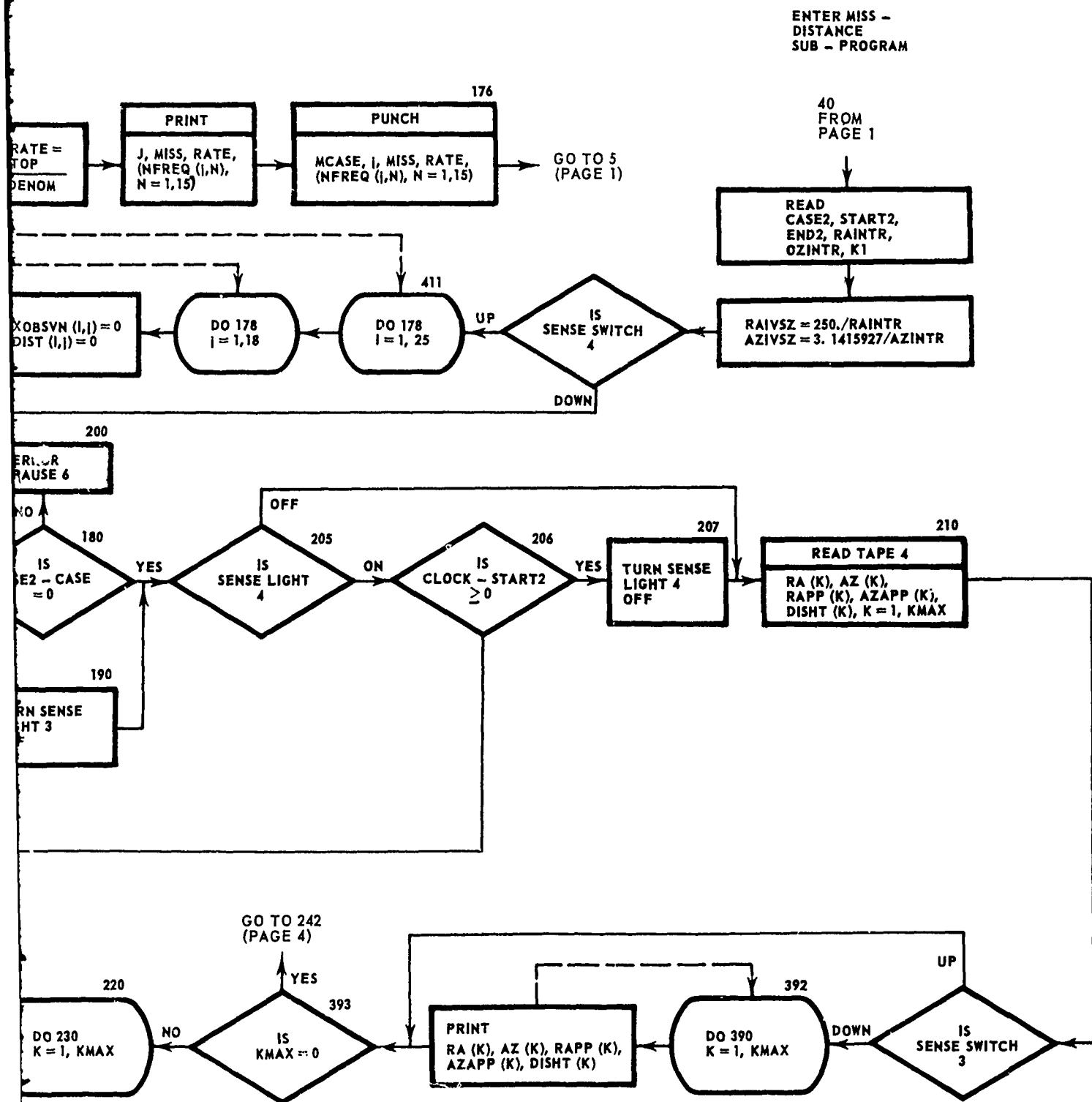


Figure 2-7 (Cont.)

B.

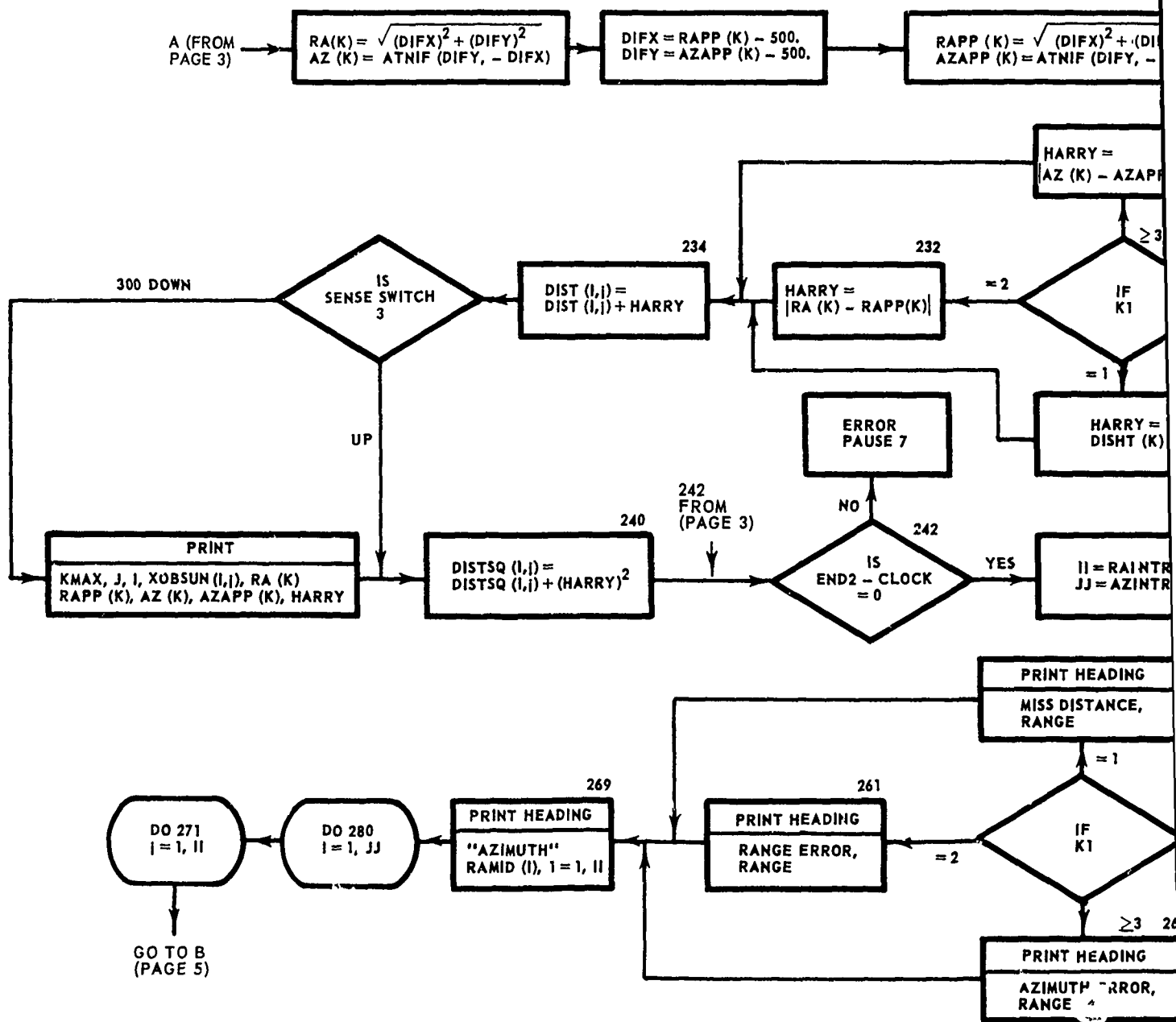


Figure 2-

A.

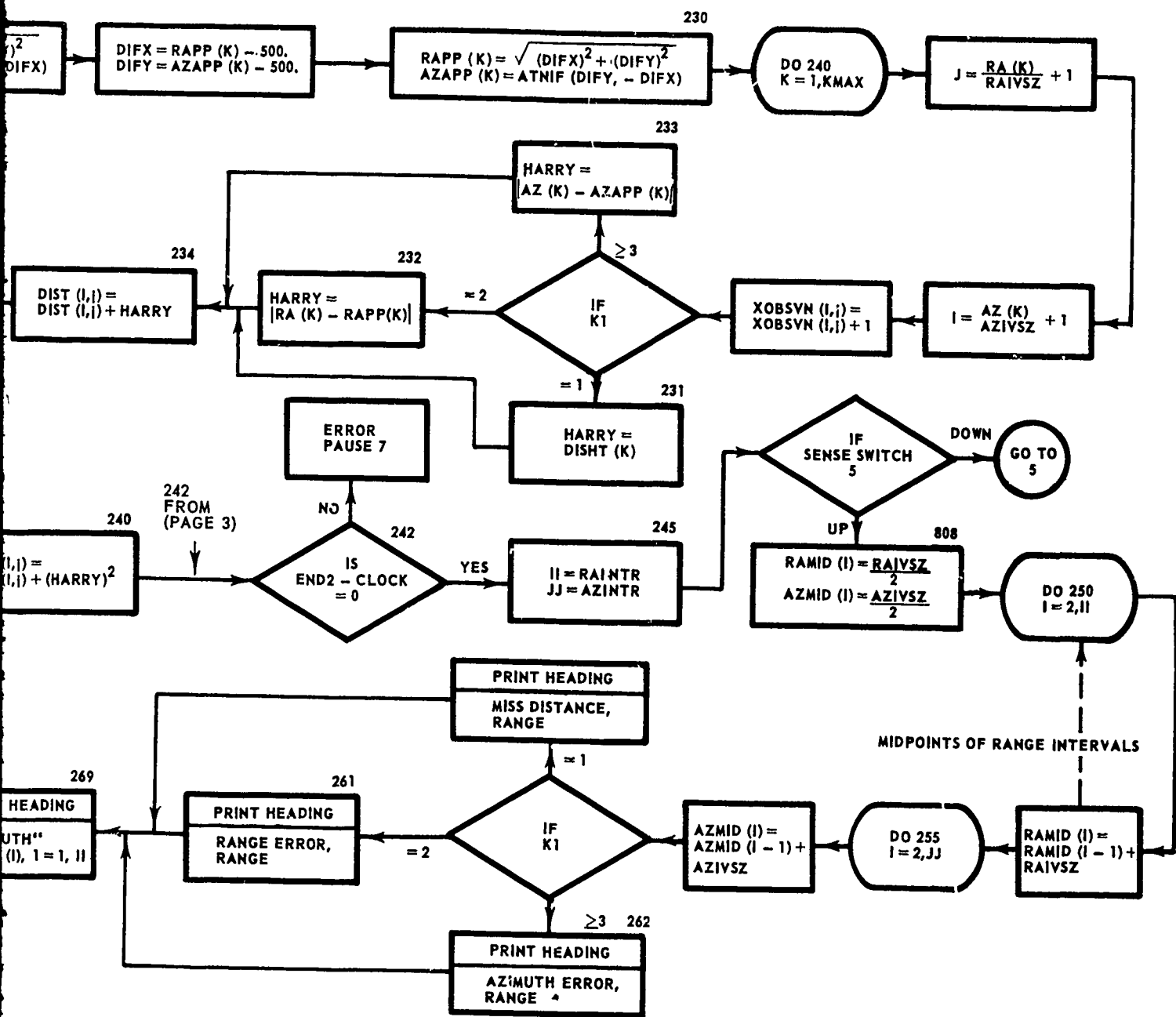


Figure 2-7 (Cont.)

B.

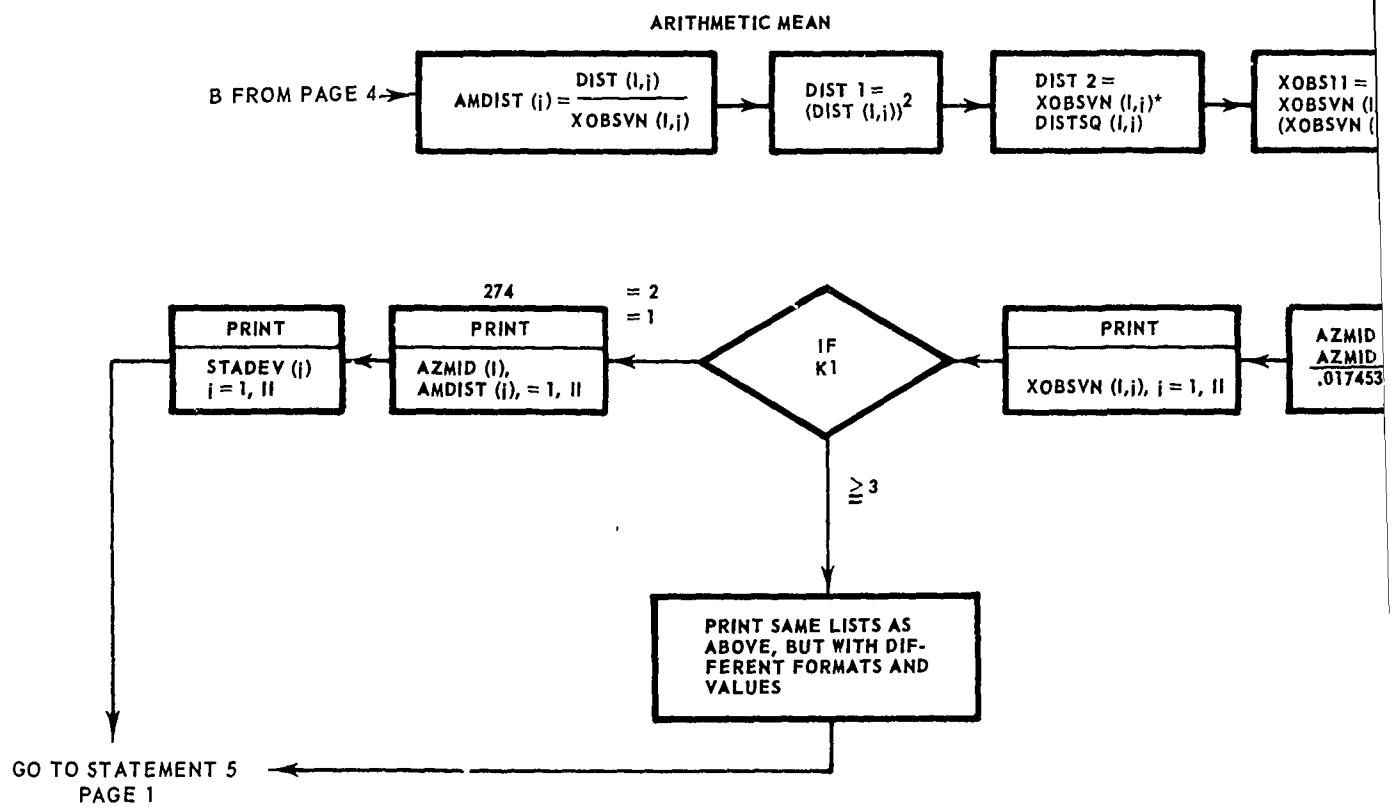


Figure 2-7

A.

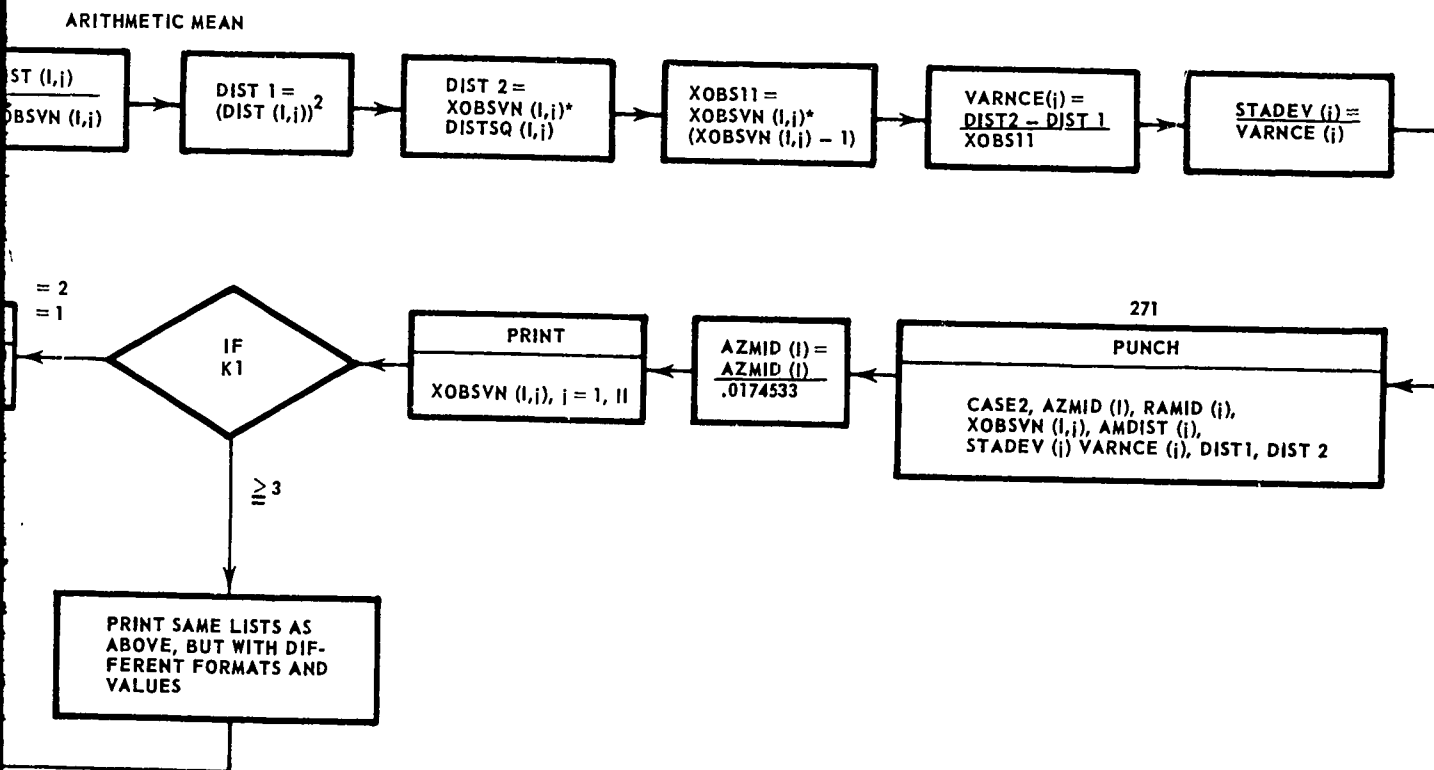


Figure 2-7 (Cont.)

B.