REPORT NO. DOT-TSC-OST-74-3. IVB

STRATEGIC CONTROL ALGORITHM DEVELOPMENT
Volume IVB: Computer Program Report (Concluded)

R. L. Erwin
M. J. Omoth
W. H. Galer
D. Hartnell
A. L. Yarrington et al.

AUGUST 1974

FINAL REPORT

DOCUMENT IS AVAILABLE TO THE PUBLIC THROUGH THE NATIONAL TECHNICAL INFORMATION SERVICE, SPRINGFIELD, VIRGINIA 22151.

Prepared for

U.S. DEPARTMENT OF TRANSPORTATION
OFFICE OF THE SECRETARY
Office of the Assistant Secretary
For Systems Development and Technology
Office of Systems Engineering
Washington DC 20590

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
DOT-TSC-OST-74-3. IVB		

4. Title and Subtitle	5. Report Date
STRATEGIC CONTROL ALGORITHM DEVELOPMENT Volume IVB: Computer Program Report (Concluded)	August 1974
	6. Performing Organization Code
7. Author(s) R.L. Erwin, M.J. Omoth, W.H. Galer D. Hartnell, A.L. Yarrington et al.	8. Performing Organization Report No. DOT-TSC-OST-74-3. IVB
9. Performing Organization Name and Address Boeing Commercial Airplane Company* P.O. Box 3707 Seattle WA 98124	10. Work Unit No. (TRAIS) OS404/R4509
	11. Contract or Grant No. DOT-TSC-538-4
	13. Type of Report and Period Covered Final Report Nov. 1972 - Dec. 1973
12. Sponsoring Agency Name and Address Department of Transportation Office of the Secretary Office of the Ass't. Sec. for Sys. Dev. & Tech. Office of Systems Engineering Washington DC 20590	14. Sponsoring Agency Code

15. Supplementary Notes

*Under contract to: U.S. Department of Transportation
Transportation Systems Center, Kendall Sq.
Cambridge MA 02142

16. Abstract

A description of the strategic algorithm evaluation model is presented, both at the user and programmer levels. The model representation of an airport configuration, environmental considerations, the strategic control algorithm logic, and the airplane simulation model are delineated, together with data inputs and outputs. Detailed instructions for running the model include the input deck setup. Listings of the complete program, as well as detailed logic flow charts and a variable dictionary, are included. Program storage requirements and machine dependence considerations are also discussed.

Volume IVA includes sections 1 through 4.2.15;
Volume IVB includes sections 4.2.16 through 5.

17. Key Words	18. Distribution Statement
Air traffic control Computer modeling Airport operations analysis Airport operations simulation	DOCUMENT IS AVAILABLE TO THE PUBLIC THROUGH THE NATIONAL TECHNICAL INFORMATION SERVICE, SPRINGFIELD, VIRGINIA 22151.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	272	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

PREFACE

The strategic Control Algorithm Development program is a first study in the evolution of the strategic control concept. Previous work accomplished during the concept formulation stage of the Advanced Air Traffic Management System (AATMS) indicated that this technique held potential benefit for accommodating high traffic demands projected for the 1990's and beyond. The present effort explored the feasibility of basic strategic arrival control via analysis and fast-time simulation. This work included the design of a basic arrival control algorithm which accomplished sequencing, scheduling and generation of conflict-free four-dimensional flight paths for assignment to each arrival in the demand scenario.

In addition to the basic algorithm design and testing, tasks were accomplished to determine potential airports for application of strategic control; assess the resulting benefits; make a preliminary estimate of data processing requirements; and refine the concept. A Research, Development, Test and Evaluation (RDT&E) program was also developed.

In accomplishing this study it was necessary to provide an integration of technologies in the study team. As strategic control is primarily designed for automatic operation, it is necessary to understand airplane performance capability, wind and temperature effect, avionics capability, and computing technology, as well as comprehensive understanding of the Air Traffic Control environment. Successful integration of these technologies resulted in considerable insight into the requirements imposed on strategic flight path generation.

In the future, the evolution of the strategic control concept will require studies designed to establish the feasibility, requirements, and algorithms for strategic departure and en route airplanes. Further refinement of the basic arrival strategy and means of accommodating system perturbations will need to be accomplished. Real-time simulations, including those using strategically equipped airplane(s) will provide a logical test-bed for concept demonstration and testing.

The work of the following personnel is acknowledged: A.F. Norwood, Chief, ATC and Electronics, presenting the executive level and ensuring full company support to the program and coordination with other Boeing ATC-related activities; E.A. Delanty, algorithm design; R.W. Schwab, evaluation model design, S.G. Datar, evaluation model design; R.O. Barnes, terminal and airspace requirements; E.A. Olmstead, data processing requirements; J.T. Burghart, benefits analysis; J.M. Bedregal programming and analysis; W.L. Chu, programming and analysis; H.F. Lee, programming and analysis; E.D. Ramer, programming and analysis; J.M. Sherwin, programming and analysis supervision; R.L. Swanson, programming and analysis; J. Yonekawa, engineering support.

CONTENTS - VOLUME IVB

Page

4.2.16 Subroutine HST 265
4.2.17 Subroutine LIMITS 267
4.2.18 Subroutine NORMAL 269
4.2.19 Subroutine RDAERO 270
4.2.20 Subroutine RDATC 272
4.2.21 Subroutine RDAVNC 274
4.2.22 Subroutine RDDSCR 275
4.2.23 Subroutine RDDSTR 277
4.2.24 Subroutine RDGEOM 279
4.2.25 Subroutine RDSTAT 282
4.2.26 Subroutine RDWTHR 283
4.2.27 Subroutine READIN 285
4.2.28 Subroutine REFBUG 287
4.2.29 Subroutine RTP 289
4.2.30 Subroutine SCHLD1 302
4.2.31 Subroutine SCHLD2 305
4.2.32 Subroutine SCHLD3 309
4.2.33 Subroutine SCHWT 313
4.2.34 Subroutine SEQUEN 315
4.2.35 Subroutine SETN 318
4.2.36 Subroutine SHELL 321
4.2.37 Subroutine SHELLX 323
4.2.38 Subroutine SORT 324
4.2.39 Subroutine STATIC 326
4.2.40 Program STRAD 334
4.2.41 Subroutine STUFF1 340
4.2.42 Subroutine STUFF2 345
4.2.43 Subroutine STUFF3 348
4.2.44 Subroutine TASCAS 350
4.2.45 Function TBLU 351
4.2.46 Subroutine TFIND 353
4.2.47 Subroutine TIMAA 356
4.2.48 Subroutine TIMAD 358
4.2.49 Subroutine TIMDA 360
4.2.50 Subroutine TIMDD 362
4.2.51 Subroutine TIMFND 364
4.2.52 Subroutine TOMIAF 367
4.2.53 Subroutine TRAFIC 368
4.2.54 Subroutine TRAFWT 376
4.2.55 Subroutine TRKGOM 378
4.2.56 Subroutine UNIFORM 382
4.2.57 Subroutine WETHER 383
4.2.58 Subroutine WTHTST 386
4.2.59 Subroutine WTOUT 388
4.3 Program Labeled Error Stops 392

CONTENTS - VOLUME IVB (concluded)

Section		Page
4.4	Machine Dependencies	392
4.4.1	Random-Number Generation	392
4.4.2	Program Storage Requirements	393
5.0	STRATEGIC ALGORITHM EVALUATION MODEL LISTING	395

TABLES - VOLUME IVB

		Page
4-5	Machine Dependencies	393
5-1	Program Listing	396

4.2.16 Subroutine HST

4.2.16.1 Abstract

Subroutine HST computes the appropriate index for entries in histogram tables.

4.2.16.2 Data Interfaces

HST Inputs

Through Calling Sequence:

XINIT	Initial table value
XINC	Increment
NENT	Number of slots in table
X POINT	Entry value

HST Outputs

Through Calling Sequence:

INDIX	Index of entry location in table

HST Calls

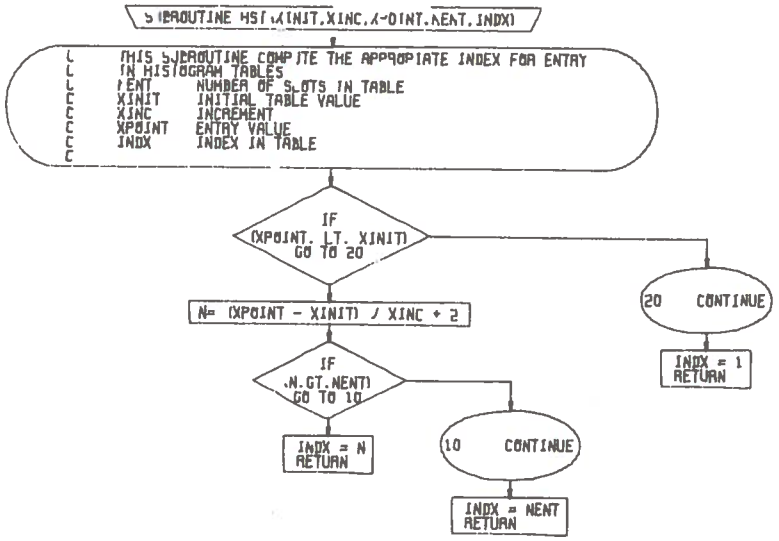
None

HST Called by

FLIGTX

4.2.16.3 HST Variables

None


SUBROUTINE HST(XINIT,XINC,XPOINT,NENT,INDX)
C THIS SUBROUTINE COMPUTE THE APPROPIATE INDEX FOR ENTRY
C IN HISTOGRAM TABLES
C NENT NUMBER OF SLOTS IN TABLE
C XINIT INITIAL TABLE VALUE
C XINC INCREMENT
C XPOINT ENTRY VALUE
C INDX INDEX IN TABLE
C
IF (XPOINT. LT. XINIT) GO TO 20
N= (XPOINT - XINIT) / XINC + 2
IF (N.GT.NENT) GO TO 10
INDX = N RETURN
10 CONTINUE
INDX = NENT RETURN
20 CONTINUE
INDX = 1 RETURN

4.2.17 Subroutine LIMITS

4.2.17.1 Abstract

Subroutine LIMITS is entered with a specified probability distribution (IDIST), its parameters (P1, P2), the lower and upper limits (XLOW, HIGH) of the variable to be generated, and a limit switch (N). The routine generates a variable value (A) according to the designated probability distribution and the preset limits.

4.2.17.2 Data Interface

LIMITS Inputs

Through Calling Sequence:

IDIST	Name of the probability distribution
P1	First parameter of the probability distribution
P2	Second parameter
XLOW	Lower limit of the variable values
HIGH	Upper limit of the variable values
N	Limit checking switch when N = 1 checking for lower limit only N = 3 checking for upper limit only N = 2 checking for both limits

LIMITS Outputs

Through Calling Sequence:

A	Random variate generated from the specified distribution and within the required bounds

LIMITS Calls

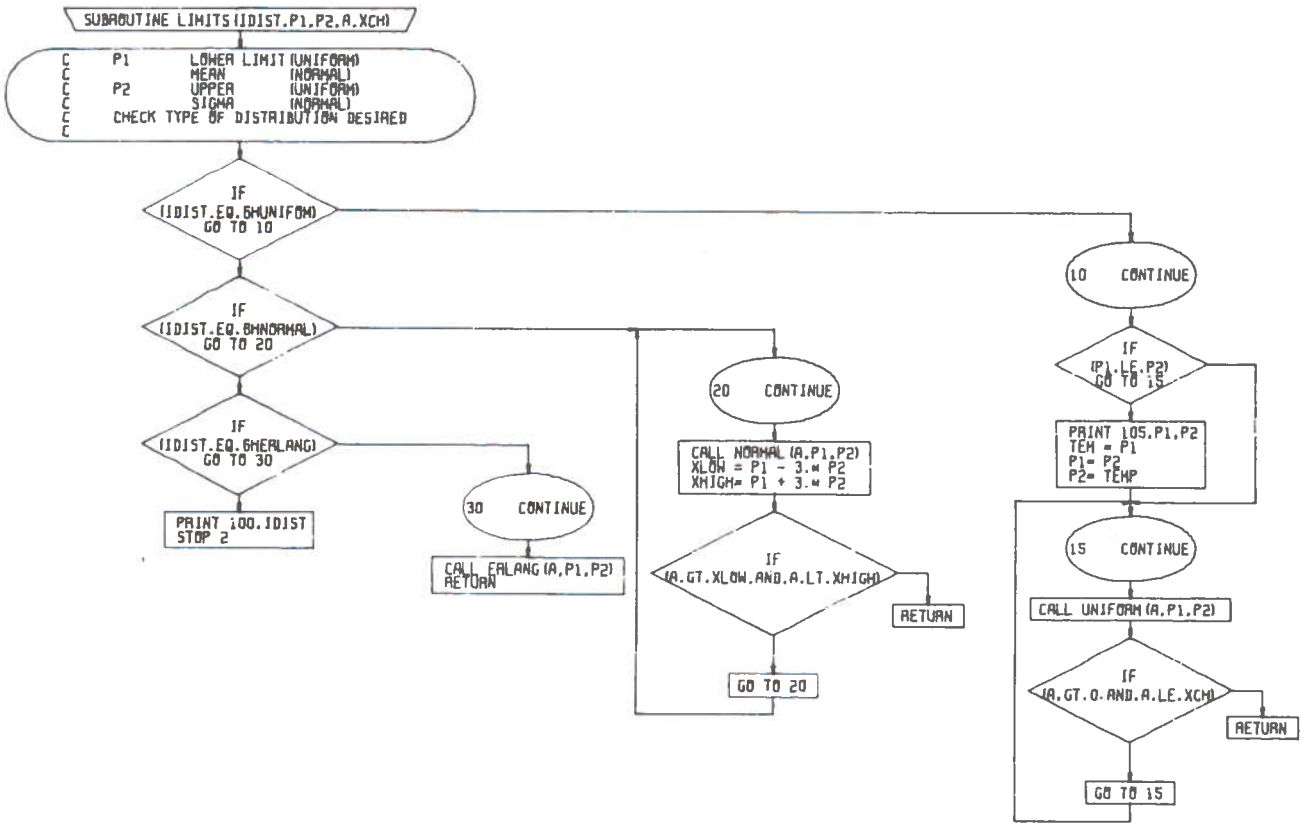
ERLANG, NORMAL, UNIFORM

LIMITS Called by

TRAFIC

4.2.17.3 LIMITS Variables

None


SUBROUTINE LIMITS(IDIST,P1,P2,A,XCM)
C P1 LOWER LIMIT (UNIFORM)
C MEAN (NORMAL)
C P2 UPPER (UNIFORM)
C SIGMA (NORMAL)
C CHECK TYPE OF DISTRIBUTION DESIRED
C
IF (IDIST.EQ.6HUNIFOM) GO TO 10
IF (IDIST.EQ.6HNORMAL) GO TO 20
IF (IDIST.EQ.6HERLANG) GO TO 30
PRINT 100,IDIST
STOP 2
30 CONTINUE
CALL ERLANG(A,P1,P2)
RETURN
20 CONTINUE
CALL NORMAL(A,P1,P2)
XLOW = P1 - 3.* P2
XHIGH= P1 + 3.* P2
IF (A.GT.XLOW.AND.A.LT.XHIGH)
RETURN
GO TO 20
10 CONTINUE
IF (P1.LE.P2) GO TO 15
PRINT 105,P1,P2
TEM = P1
P1= P2
P2= TEMP
15 CONTINUE
CALL UNIFORM(A,P1,P2)
IF (A.GT.0.AND.A.LE.XCM)
RETURN
GO TO 15

4.2.18 Subroutine NORMAL

4.2.18.1 Abstract

Subroutine NORMAL is entered with the mean (P1) and standard deviation (P2) of the normal distribution. It returns a normally distributed random variate sampled from a distribution with mean P1 and standard deviation P2.

4.2.18.2 Data Interfaces

NORMAL Inputs

Through Calling Sequence:

P1	Mean of the normal distribution
P2	Standard deviation of the normal distribution

Through Common Statements:

/RANDOM/	IU	Starter of the seed of the random-number generator
	IX	Multiplier of the seed of the random-number generator

NORMAL Outputs

Through Calling Sequence

A	The sampled random variate

NORMAL Calls

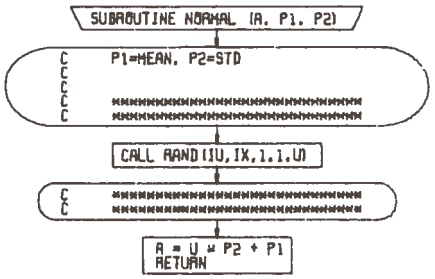
RAND

NORMAL Called by

TRAFIC, LIMITS

4.2.18.3 NORMAL Variables

U	A random variable generated by RAND, sampled from a normal distribution with zero mean and standard deviation 1



4.2.19 Subroutine RDAERO

4.2.19.1 Abstract

Subroutine RDAERO reads the aero profile data for each arrival airplane type.

4.2.19.2 Data Interfaces

RDAERO Inputs

Through Common Statements:

/AEROIN/	All variables
/TRAFIN/	NYTPEA

RDAERO Outputs

Through Common Statements:

/AEROIN/	All variables

Through Printing:

/AEROIN/	All variables except NAERO
/TRAFIN/	NMTYPA

RDAERO Calls

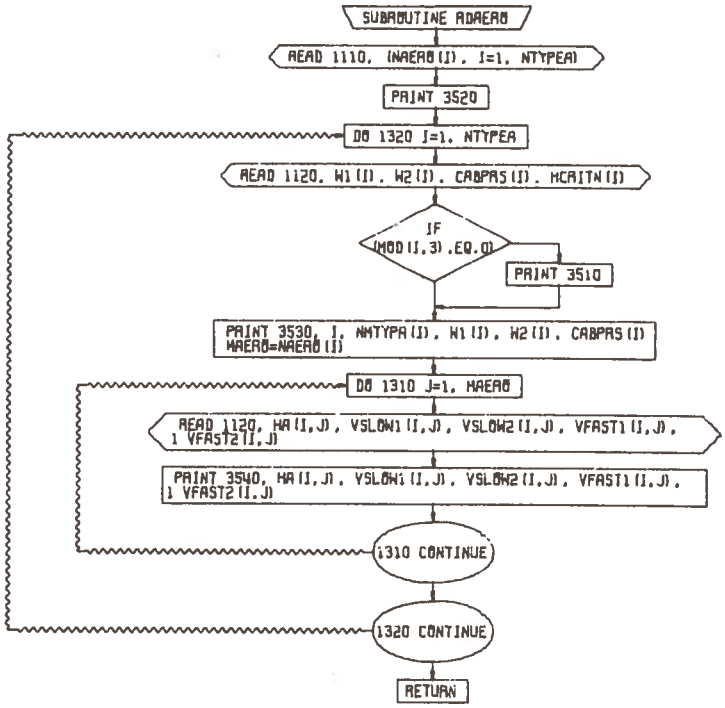
None

RDAERO Called by

READIN

4.2.19.3 RDAERO Variables

None


SUBROUTINE RDAERA
READ 1110, (NAERA(I), I=1, NTYPEA)
PRINT 3520
DO 1320 I=1, NTYPEA
READ 1120, W1(I), W2(I), CABPAS(I), HCRITN(I)
IF
(MOD(I,3).EQ.0)
PRINT 3510
PRINT 3530, I, NMTYPA(I), W1(I), W2(I), CABPAS(I)
NAERA=NAERA(I)
DO 1310 J=1, NAERA
READ 1120, HA(I,J), VSLOW1(I,J), VSLOW2(I,J), VFAST1(I,J),
1 VFAST2(I,J)
PRINT 3540, HA(I,J), VSLOW1(I,J), VSLOW2(I,J), VFAST1(I,J),
1 VFAST2(I,J)
1310 CONTINUE
1320 CONTINUE
RETURN

4.2.20 Subroutine RDATC

4.2.20.1 Abstract

Subroutine RDATC reads in the ATC constraint data and scheduler selection switch.

4.2.20.2 Data Interfaces

RDATC Inputs

Through Read Inputs:
/ATCIN/ All variables

Through Common Statements:
/TRAFIN/ NMTYPA, NMTYPD, NTYPEA, NTYPED

RDATC Outputs

Through Common Statements:
/ATCIN/ All variables

Through Printing:
/ATCIN/ All variables
/TRAFIN/ NMTYPA, NMTYPD

RDATC Calls

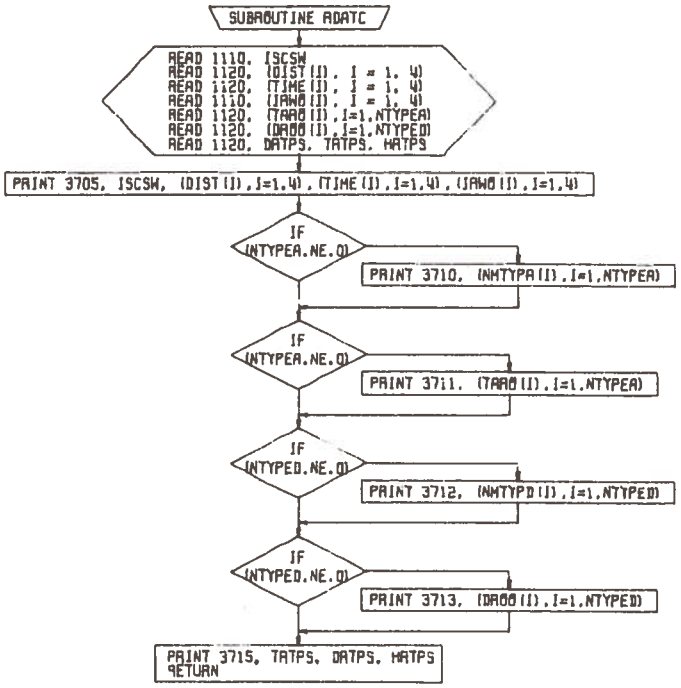
None

RDATC Called by

READIN

4.2.20.3 RDATC Variables

None


SUBROUTINE RDATC
READ 1110, ISCSW
READ 1120, (DIST(I), I = 1, 4)
READ 1120, (TIME(I), I = 1, 4)
READ 1110, (JAW0(I), I = 1, 4)
READ 1120, (TAR0(I),I=1,NTYPEA)
READ 1120, (DR00(I),I=1,NTYPED)
READ 1120, DATPS, TATPS, HATPS
PRINT 3705, ISCSW, (DIST(I),I=1,4),(TIME(I),I=1,4),(JAW0(I),I=1,4)
IF (NTYPEA.NE.0)
PRINT 3710, (NHTYPA(I),I=1,NTYPEA)
IF (NTYPEA.NE.0)
PRINT 3711, (TAR0(I),I=1,NTYPEA)
IF (NTYPED.NE.0)
PRINT 3712, (NHTYPD(I),I=1,NTYPED)
IF (NTYPED.NE.0)
PRINT 3713, (DR00(I),I=1,NTYPED)
PRINT 3715, TATPS, DATPS, HATPS
RETURN

4.2.21 Subroutine RDAVNC

4.2.21.1 Abstract

Subroutine RDAVNC reads the avionics input data, which consist of the navigation and control systems statistical error parameters.

4.2.21.2 Data Interfaces

RDAVNC Inputs

Through Read Inputs:

/AVION/ All variables

RDAVNC Outputs

Through Common Statements and Printing:

/AVION/ All variables

RDAVNC Calls

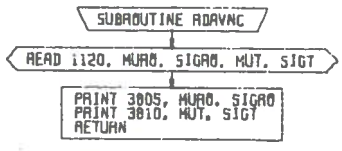
None

RDAVNC Called by

READIN

4.2.21.3 RDAVNC Variables

None



4.2.22 Subroutine RDDSCR

4.2.22.1 Abstract

Subroutine RDDSCR reads the discrete traffic input data.

4.2.22.2 Data Interfaces

RDDSCR Inputs

Through Read Inputs:

/TRAFIN/	NTYPEA, NTYPED, TASIGM, TDSIGM, SPSIGM, CVREF, CWVREF, NCVREF, NMTYPA, NMTYPD
/TRAFOT/	NACA, NACD, ALT, TSECSD, TSECSA, LFIX, ITYPEA, SPEED, GWA, GWD, ITYPED
/RANDOM/	IU, IX

RDDSCR Outputs

Through Common Statements:

Above, plus

/TRAFIN/	HOUR

Through Printing:

/TRAFIN/	NYPEA, TASIGM, TDSIGM, SPSIGM, NMTYPA, CVREF, CWVREF

RDDSCR Calls

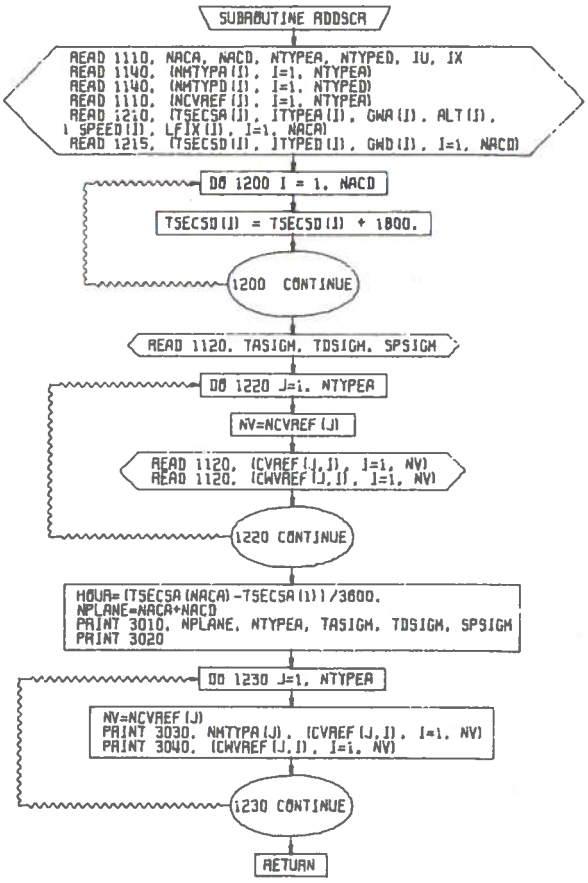
None

RDDSCR Called by

READIN

4.2.22.3 RDDSCR Variables

NPLANE	Total arrivals and departures


SUBROUTINE ADDSCR
READ 1110, NACA, NACD, NTYPEA, NTYPED, IU, IX
READ 1140, (NMTYPA(I), I=1, NTYPEA)
READ 1140, (NMTYPD(I), I=1, NTYPED)
READ 1110, (NCVREF(I), I=1, NTYPEA)
READ 1210, (TSECSA(I), ITYPEA(I), GWA(I), ALT(I),
SPEED(I), LFIX(I), I=1, NACA)
READ 1215, (TSECSD(I), ITYPED(I), GWD(I), I=1, NACD)
DO 1200 I = 1, NACD
TSECSD(I) = TSECSD(I) + 1800.
1200 CONTINUE
READ 1120, TASIGM, TDSIGM, SPSIGM
DO 1220 J=1, NTYPEA
NV=NCVREF(J)
READ 1120, (CVREF(J,I), I=1, NV)
READ 1120, (CWVREF(J,I), I=1, NV)
1220 CONTINUE
HOUR=(TSECSA(NACA)-TSECSA(1))/3600.
NPLANE=NACA+NACD
PRINT 3010, NPLANE, NTYPEA, TASIGM, TDSIGM, SPSIGM
PRINT 3020
DO 1230 J=1, NTYPEA
NV=NCVREF(J)
PRINT 3030, NMTYPA(J), (CVREF(J,I), I=1, NV)
PRINT 3040, (CWVREF(J,I), I=1, NV)
1230 CONTINUE
RETURN

4.2.23 Subroutine RDDSTR

4.2.23.1 Abstract

Subroutine RDDSTR reads the distribution traffic input data.

4.2.23.2 Data Interface

RDDSTR Inputs

Through Read Inputs:

/TRAFIN/	All variables except ENTRY, SPSIGM, SWTRAF, TASIGM, TDSIGM
/RANDOM/	IU, IZ

RDDSTR Outputs

Through Common Statements:

/TRAFIN/	All variables except TASIGM, TDSIGM, SPSIGM, SWTRAF, HOUR, NTYPEA, NTYPED, ENTRY, NCVREF, NCALT

RDDSTR Calls

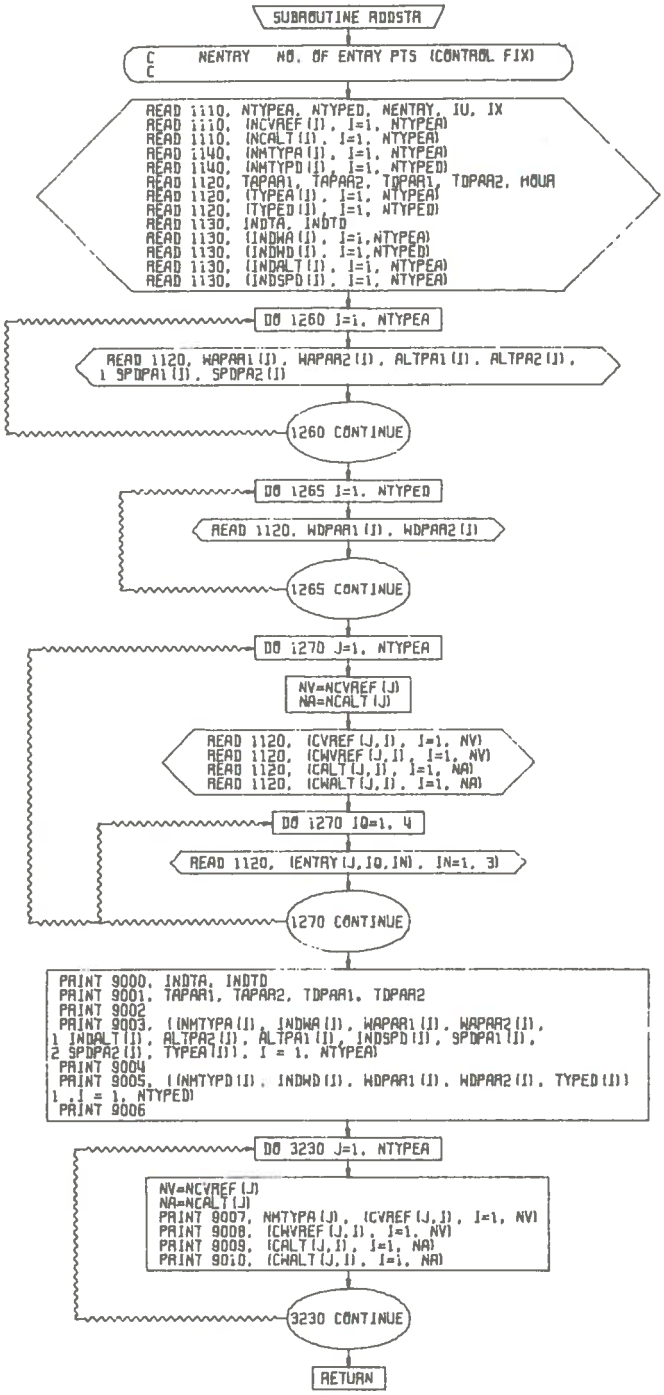
None

RDDSTR Called by

READIN

4.2.23.3 RDDSTR Variables

NENTRY	Number of entry points in the terminal area


SUBROUTINE RDDSTR
C NENTRY NO. OF ENTRY PTS (CONTROL FIX)
C
READ 1110, NTYPEA, NTYPED, NENTRY, IU, IX
READ 1110, (NCVREF(I), I=1, NTYPEA)
READ 1110, (NCALT(I), I=1, NTYPEA)
READ 1140, (NMTYPA(I), I=1, NTYPEA)
READ 1140, (NMTYPD(I), I=1, NTYPED)
READ 1120, TAPAA1, TAPAA2, TDPAA1, TDPAA2, HOUR
READ 1120, (TYPEA(I), I=1, NTYPEA)
READ 1120, (TYPED(I), I=1, NTYPED)
READ 1130, INDTA, INDTD
READ 1130, (INDWA(I), I=1,NTYPEA)
READ 1130, (INDWD(I), I=1,NTYPED)
READ 1130, (INDALT(I), I=1, NTYPEA)
READ 1130, (INDSPD(I), I=1, NTYPEA)
DO 1260 I=1, NTYPEA
READ 1120, WAPAR1(I), WAPAR2(I), ALTPA1(I), ALTPA2(I),
1 SPDPA1(I), SPDPA2(I)
1260 CONTINUE
DO 1265 I=1, NTYPED
READ 1120, WDPAR1(I), WDPAR2(I)
1265 CONTINUE
DO 1270 J=1, NTYPEA
NV=NCVREF(J)
NA=NCALT(J)
READ 1120, (CVREF(J,I), I=1, NV)
READ 1120, (CWVREF(J,I), I=1, NV)
READ 1120, (CALT(J,I), I=1, NA)
READ 1120, (CWALT(J,I), I=1, NA)
DO 1270 IQ=1, 4
READ 1120, (ENTRY(J,IQ,IN), IN=1, 3)
1270 CONTINUE
PRINT 9000, INDTA, INDTD
PRINT 9001, TAPAR1, TAPAR2, TDPAR1, TDPAR2
PRINT 9002
PRINT 9003, ((NMTYPA(I), INDWA(I), WAPAR1(I), WAPAR2(I),
1 INDALT(I), ALTPA2(I), ALTPA1(I), INDSPD(I), SPDPA1(I),
2 SPDPA2(I), TYPEA(I)), I = 1, NTYPEA)
PRINT 9004
PRINT 9005, ((NMTYPD(I), INDWD(I), WDPAR1(I), WDPAR2(I), TYPED(I))
1 ,I = 1, NTYPED)
PRINT 9006
DO 3230 J=1, NTYPEA
NV=NCVREF(J)
NA=NCALT(J)
PRINT 9007, NMTYPA(J), (CVREF(J,I), I=1, NV)
PRINT 9008, (CWVREF(J,I), I=1, NV)
PRINT 9009, (CALT(J,I), I=1, NA)
PRINT 9010, (CWALT(J,I), I=1, NA)
3230 CONTINUE
RETURN

4.2.24 Subroutine RDGEOM

4.2.24.1 Abstract

Subroutine RDGEOM reads the terminal area geometry inputs used by the model to determine flightpath distances from entry fixes to the threshold.

4.2.24.2 Data Interfaces

RDGEOM Inputs

Through Read Inputs:
/GEOMIN/ All variables

RDGEOM Outputs

Through Common Statements:
/GEOMIN/ All variables

Through Printing:
/GEOMIN/ All variables

RDGEOM Calls

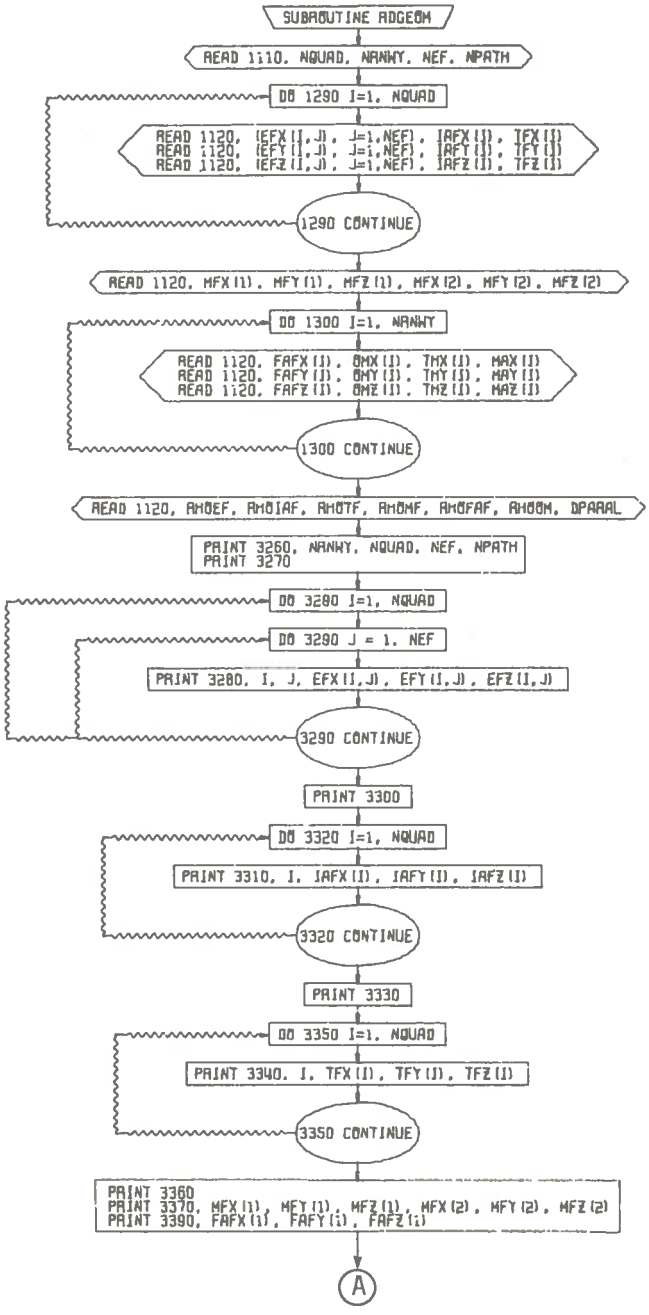
None

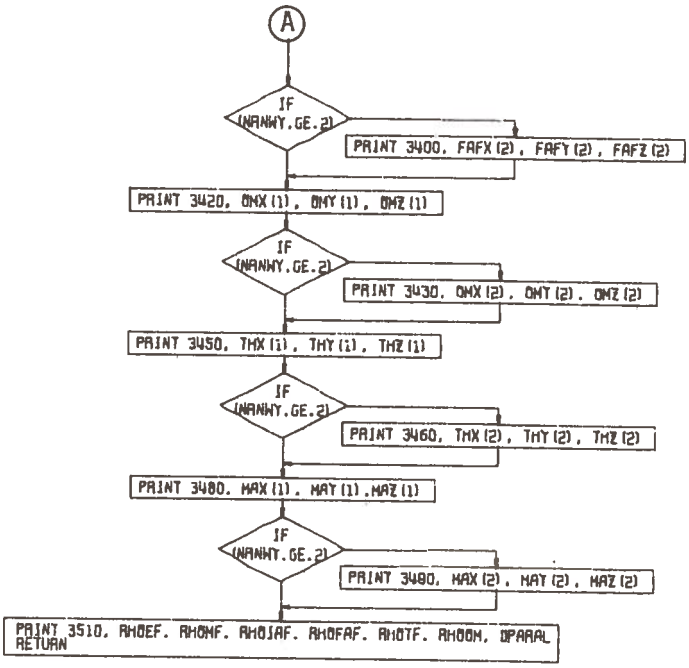
RDGEOM Called by

READIN

4.2.24.3 RDGEOM Variables

None


SUBROUTINE RDGEOM
READ 1110, NQUAD, NANWY, NEF, NPATH
DO 1290 I=1, NQUAD
READ 1120, (EFX(I,J), J=1,NEF), IAFX(I), TFX(I)
READ 1120, (EFY(I,J), J=1,NEF), IAFY(I), TFY(I)
READ 1120, (EFZ(I,J), J=1,NEF), IAFZ(I), TFZ(I)
1290 CONTINUE
READ 1120, MFX(1), MFY(1), MFZ(1), MFX(2), MFY(2), MFZ(2)
DO 1300 I=1, NANWY
READ 1120, FAFX(I), GMX(I), TMX(I), MAX(I)
READ 1120, FAFY(I), GMY(I), TMY(I), MAY(I)
READ 1120, FAFZ(I), GMZ(I), TMZ(I), MAZ(I)
1300 CONTINUE
READ 1120, RHOEF, RHOIAF, RHOTF, RHOMF, RHOFAF, RHOGM, DPARAL
PRINT 3260, NANWY, NQUAD, NEF, NPATH
PRINT 3270
DO 3280 I=1, NQUAD
DO 3290 J = 1, NEF
PRINT 3280, I, J, EFX(I,J), EFY(I,J), EFZ(I,J)
3290 CONTINUE
PRINT 3300
DO 3320 I=1, NQUAD
PRINT 3310, I, IAFX(I), IAFY(I), IAFZ(I)
3320 CONTINUE
PRINT 3330
DO 3350 I=1, NQUAD
PRINT 3340, I, TFX(I), TFY(I), TFZ(I)
3350 CONTINUE
PRINT 3360
PRINT 3370, MFX(1), MFY(1), MFZ(1), MFX(2), MFY(2), MFZ(2)
PRINT 3390, FAFX(1), FAFY(1), FAFZ(1)
A


A
IF (NANWY.GE.2)
PRINT 3400, FAFX(2), FAFY(2), FAFZ(2)
PRINT 3420, OMX(1), OMY(1), OMZ(1)
IF (NANWY.GE.2)
PRINT 3430, OMX(2), OMY(2), OMZ(2)
PRINT 3450, THX(1), THY(1), THZ(1)
IF (NANWY.GE.2)
PRINT 3460, THX(2), THY(2), THZ(2)
PRINT 3480, MAX(1), MAY(1), MAZ(1)
IF (NANWY.GE.2)
PRINT 3490, MAX(2), MAY(2), MAZ(2)
PRINT 3510, RHOEF, RHOMF, RHOJAF, RHOFAF, RHOTF, RHOOM, DPARAL
RETURN

4.2.25 Subroutine RDSTAT

4.2.25.1 Abstract

Subroutine RDSTAT reads in the parameters for summary statistics and output tables.

4.2.25.2 Data Interfaces

RDSTAT Inputs

Through Read Inputs:
/STATIN/ All variables

RDSTAT Outputs

Through Common Statements:
/STATIN/ All variables

Through Printing:
/STATIN/ All variables

RDSTAT Calls

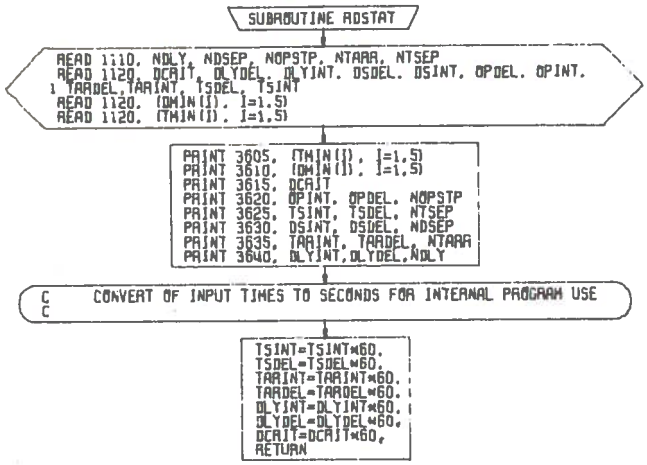
None

RDSTAT Called by

READIN

4.2.25.3 RDSTAT Variables

None



4.2.26 Subroutine RDWTHR

4.2.26.1 Abstract

Subroutine RDWTHR reads the wind and temperature data used in the program.

4.2.26.2 Data Interface

RDWTHR Inputs

Through Read Inputs:

/TEMPER/	All variables except TEMP, TEMP1, TEMP2, TSTAND
/WIND/	All variables except WINDN, WINDE, WINDN1, WINDN2, WINDE1, WINDE2, WINMAG, WINANG

RDWTHR Outputs

Through Common Statements:

/TEMPER/	All variables except TEMP, TSTAND
/WIND/	All variables except WINDE, WINDN, WINANG, WINMAG

Through Printing:

Same as input

RDWTHR Calls

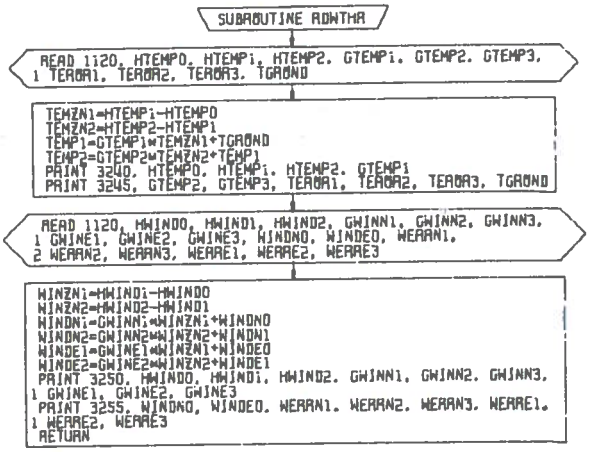
None

RDWTHR Called by

READIN

4.2.26.3 RDWTHR Variables

None


SUBROUTINE ROWTHR
READ 1120, HTEMP0, HTEMP1, HTEMP2, GTEMP1, GTEMP2, GTEMP3,
1 TEA0A1, TEA0A2, TEA0A3, TGA0ND
TEMZN1=HTEMP1-HTEMP0
TEMZN2=HTEMP2-HTEMP1
TEMP1=GTEMP1*TEMZN1+TGA0ND
TEMP2=GTEMP2*TEMZN2+TEMP1
PRINT 3240, HTEMP0, HTEMP1, HTEMP2, GTEMP1
PRINT 3245, GTEMP2, GTEMP3, TEA0A1, TEA0A2, TEA0A3, TGA0ND
READ 1120, HWIND0, HWIND1, HWIND2, GWINN1, GWINN2, GWINN3,
1 GWINE1, GWINE2, GWINE3, WINDN0, WINDE0, WERAN1,
2 WERAN2, WERAN3, WERRE1, WERRE2, WERRE3
WINZN1=HWIND1-HWIND0
WINZN2=HWIND2-HWIND1
WINDN1=GWINN1*WINZN1+WINDN0
WINDN2=GWINN2*WINZN2+WINDN1
WINDE1=GWINE1*WINZN1+WINDE0
WINDE2=GWINE2*WINZN2+WINDE1
PRINT 3250, HWIND0, HWIND1, HWIND2, GWINN1, GWINN2, GWINN3,
1 GWINE1, GWINE2, GWINE3
PRINT 3255, WINDN0, WINDE0, WERAN1, WERAN2, WERAN3, WERRE1,
1 WERRE2, WERRE3
RETURN

4.2.27 Subroutine READIN

4.2.27.1 Abstract

Subroutine READIN controls the reading, storage, and printing of all input data (except the first card, which is read by STRAD). READIN does this by calling a series of subroutines which, in turn, read particular data sets.

4.2.27.2 Data Interface

READIN Inputs (Data Read)

Through Read Input:

SWTRAF	Code word equal to DISTRB or DISCRT. If distribution traffic is input, DISTRB is used; otherwise, discrete input (DISCRT) is used.

READIN Outputs

SWTRAF	Stored in COMMON/TRAFIN/
READIN	Also prints the input data format heading

READIN Calls

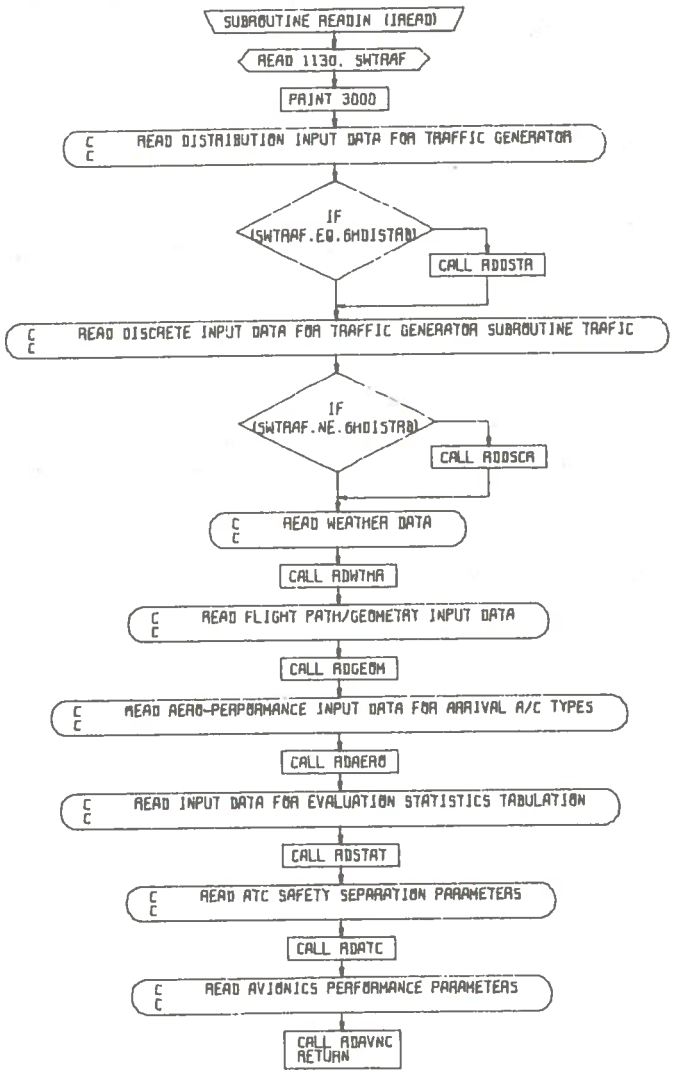
RDDSTR, RDDSCR, RDWTHR, RDGEOM, RDAERO, RDSTAT, RDATC, RDAVNC

READIN Called by

STRAD

4.2.27.3 READIN Variables

None


SUBROUTINE READIN (IREAD)
READ 1130, SWTRAF
PRINT 3000
C
C READ DISTRIBUTION INPUT DATA FOR TRAFFIC GENERATOR
IF
(SWTRAF.EQ.6HDISTRB)
CALL RDDSTR
C
C READ DISCRETE INPUT DATA FOR TRAFFIC GENERATOR SUBROUTINE TRAFIC
IF
(SWTRAF.NE.6HDISTRB)
CALL RDDSCR
C
C READ WEATHER DATA
CALL RDWTHR
C
C READ FLIGHT PATH/GEOMETRY INPUT DATA
CALL RDGEOM
C
C READ AERO-PERFORMANCE INPUT DATA FOR ARRIVAL A/C TYPES
CALL RDAERO
C
C READ INPUT DATA FOR EVALUATION STATISTICS TABULATION
CALL RDSTAT
C
C READ ATC SAFETY SEPARATION PARAMETERS
CALL RDATC
C
C READ AVIONICS PERFORMANCE PARAMETERS
CALL RDAVNC
RETURN

4.2.28 Subroutine REFBUG

4.2.28.1 Abstract

Subroutine REFBUG takes a flightpath angle corresponding to the final approach course, finds the along-track wind component, determines VBUGG, and finally determines the groundspeed equivalent of the bug airspeed corrected for the temperature forecast.

4.2.28.2 Data Interfaces

REFBUG Inputs

Through Common Statements:

/ATMO/	DENSRF
/GEOMIN/	OMZ(1)
/GEOMOT/	ANGOM(1)
/TRAFOT/	VREF(-)
/WIND/	WINMAG, WINANG

REFBUG Outputs

Through Common Statements:

/TRAFOT/	VBUGG(-)

REFBUG Calls

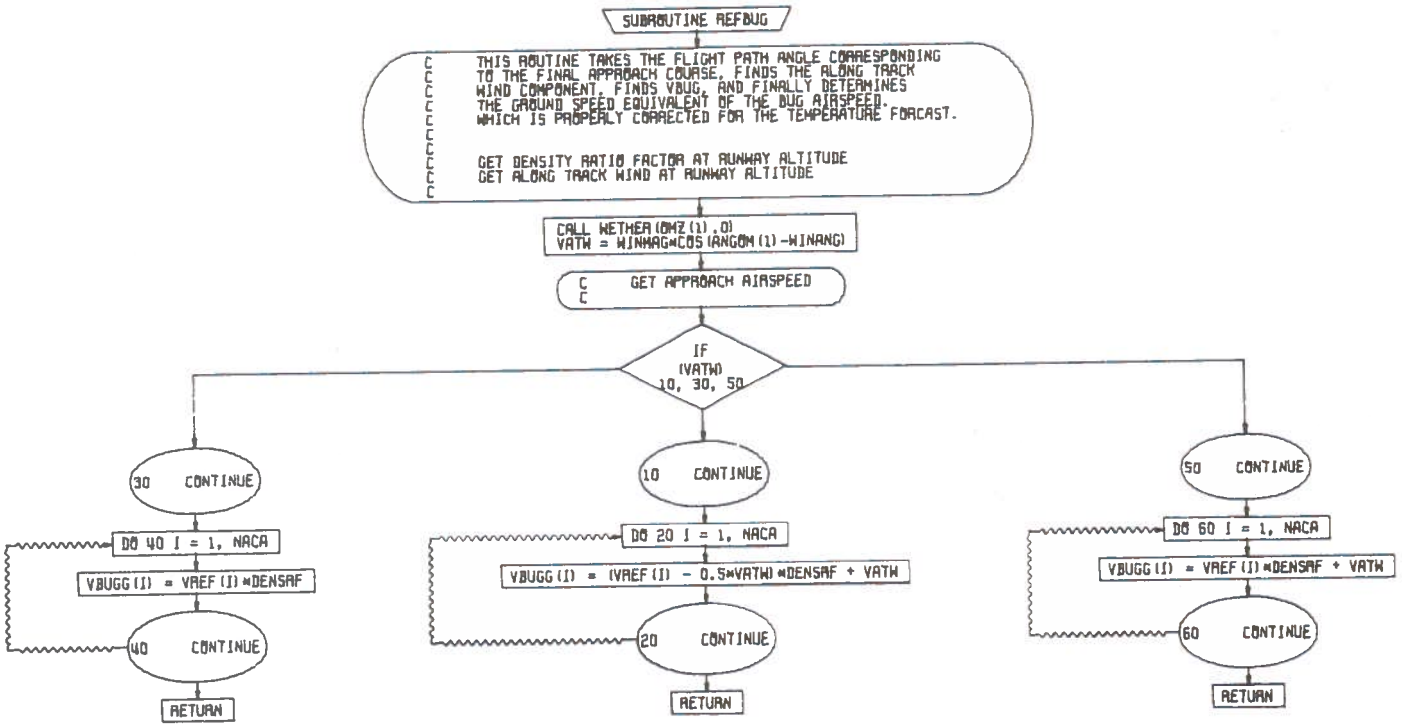
WETHER	
COS	FORTRAN library function

REFBUG Called by

STRAD

4.2.28.3 REFBUG Variables

VATW	Along-track wind velocity (in knots)


SUBROUTINE REFBUG
C THIS ROUTINE TAKES THE FLIGHT PATH ANGLE CORRESPONDING
C TO THE FINAL APPROACH COURSE, FINDS THE ALONG TRACK
C WIND COMPONENT, FINDS VBUG, AND FINALLY DETERMINES
C THE GROUND SPEED EQUIVALENT OF THE BUG AIRSPEED,
C WHICH IS PROPERLY CORRECTED FOR THE TEMPERATURE FORCAST.
C
C
C GET DENSITY RATIO FACTOR AT RUNWAY ALTITUDE
C GET ALONG TRACK WIND AT RUNWAY ALTITUDE
C
CALL WETHER(DMZ(1),0)
VATW = WINMAG*COS(ANGOM(1)-WINANG)
C
C GET APPROACH AIRSPEED
IF
(VATW)
10, 30, 50
30 CONTINUE
DO 40 I = 1, NACA
VBUGG(I) = VREF(I)*DENSAF
40 CONTINUE
RETURN
10 CONTINUE
DO 20 I = 1, NACA
VBUGG(I) = (VREF(I) - 0.5*VATW)*DENSAF + VATW
20 CONTINUE
RETURN
50 CONTINUE
DO 60 I = 1, NACA
VBUGG(I) = VREF(I)*DENSAF + VATW
60 CONTINUE
RETURN

4.2.29 Subroutine RTP

4.2.29.1 Abstract

Subroutine RTP (IFLITX, NSAMP) determines for each airplane in the arrival list a conflict-free descent path, descent velocities, and waypoint times to be made good from the entry fix to the threshold. (Data from the IAF to the threshold are not computed in RTP but are collected into the route-time profile in this subroutine.)

4.2.29.2 Data Interfaces

RTP Inputs

Through Calling Sequence:

IFLITX	Print control flag that, when used with NSAMP, determines whether current sample is to be printed
NSAMP	Index of the current sample

Through Common Statements:

/AEROOT/	HCRIT, JMAX, VFAST(-), HB(-), VSLOW(-), SPKNOT
/AERO1O/	VCG(IQUAD,1), TCSUM(IQUAD), HC(IQUAD,-), TC(IQUAD,-)
/GEOMIN/	NRNWY, THZ(IRNWY)
/GEOMOT/	DOM(IRNWY), DEF(IQUAD, INFIX, IPATH), DMF(-), DIAF(IQUAD), DTF(IQUAD), DFAF(IRNWY)
/RTPO/	DDP
/TRAFOT/	LFIX(IS), KEY, NACA, TSECAA(IS), VBUGG(IS), SLT(IS), ALT(IS)
/CNSTNT/	ITMAX, GRADI, RTPTST
/RTPOUT/	All variables

RTP Outputs

Through Common Statements:

/RTPO/	DLF, HIAF, INFIX, IQUAD, LFIX, TIAF
/RTPOUT/	All variables
/TRAFOT/	LFIX(IS)

RTP Calls

AERO	Computes aerodynamic characteristics of the airplane whose RTP is being computed
CNFLCTG	Detects passing conflicts between two airplanes on the same flightpath
CNFLCTR	Detects time, distance, and altitude conflicts between two airplanes on the same flightpath

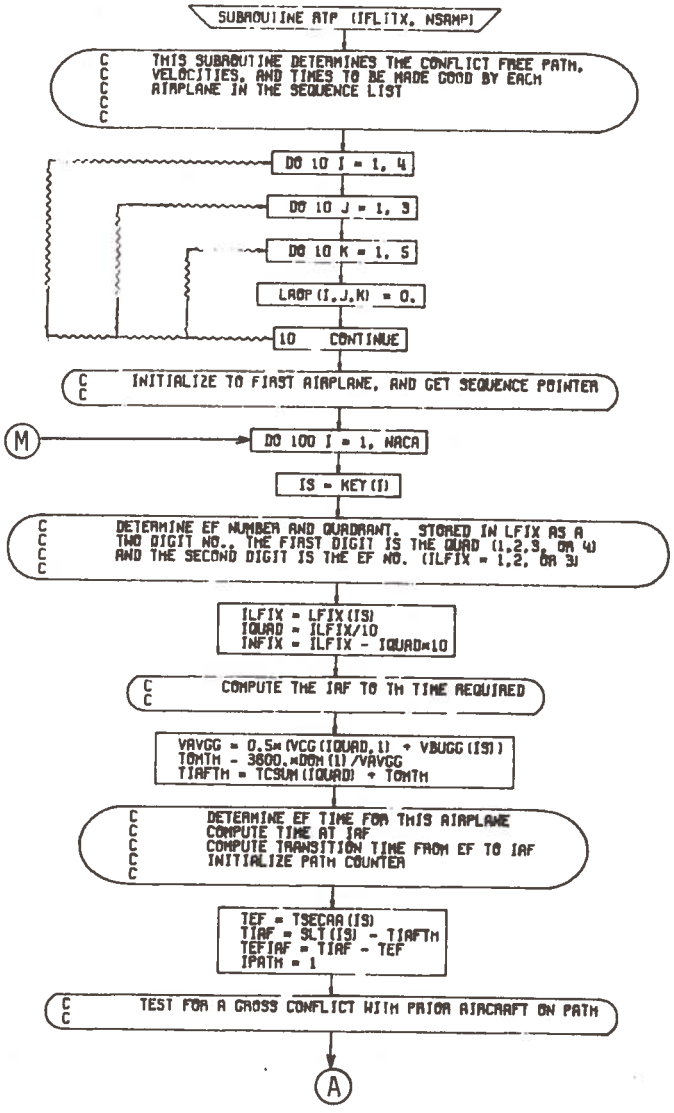
STUFF1	Creates above-critical-altitude letdown for constant calibrated or constant calibrated/constant Mach RTP
STUFF2	Creates above-critical-altitude letdown for constant Mach RTP
STUFF3	Creates below-critical-altitude letdown for constant calibrated airspeed RTP
TFIND	Transforms the RTP into a transition time duration
WETHER	Computes along-track forecast winds at a given altitude

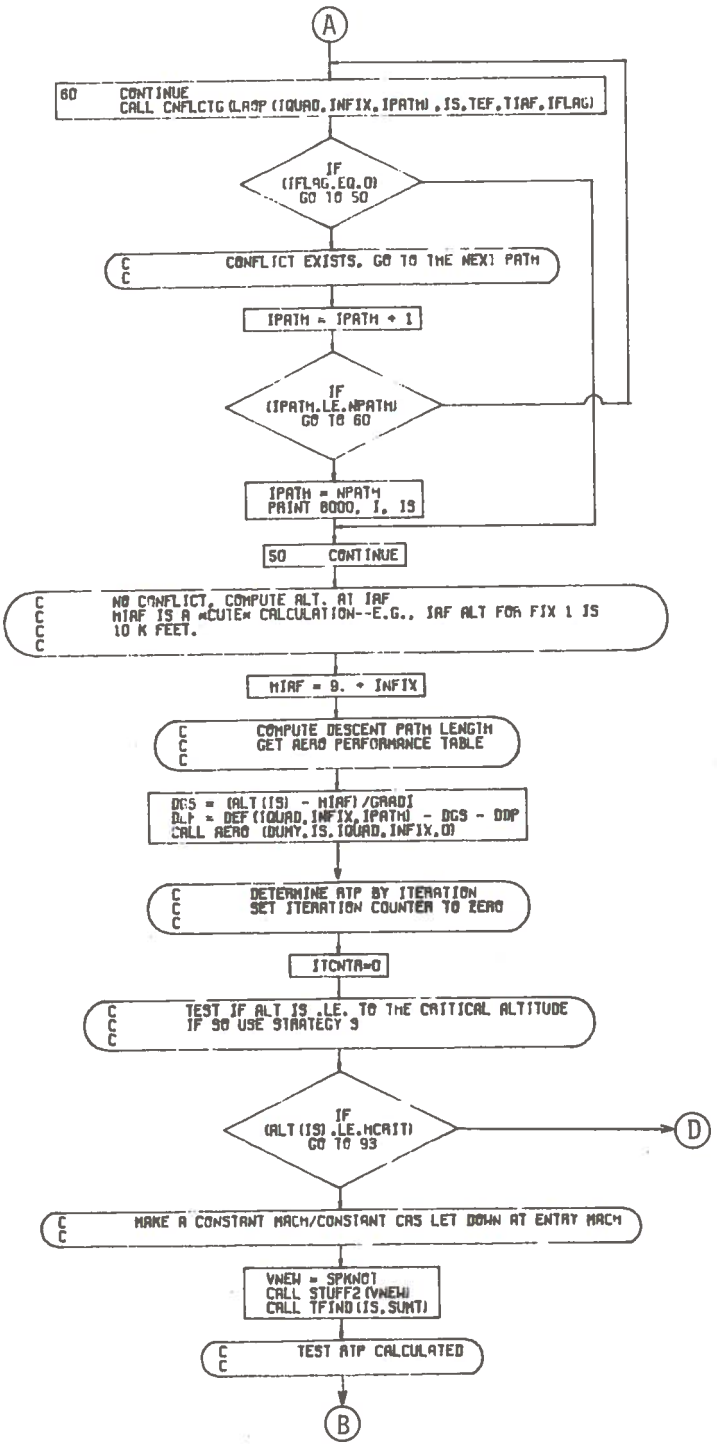
RTP Called by

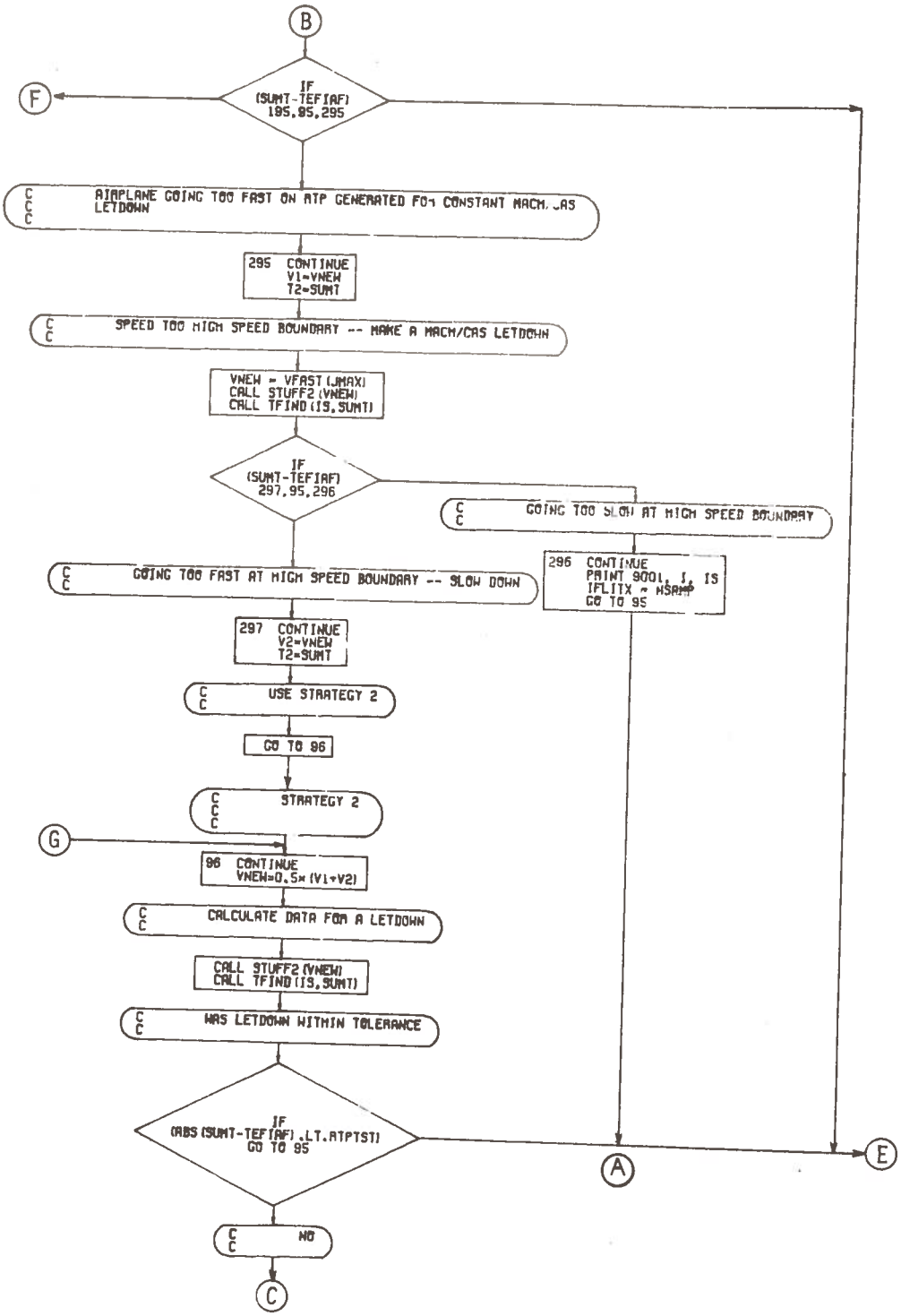
STRAD	Main program

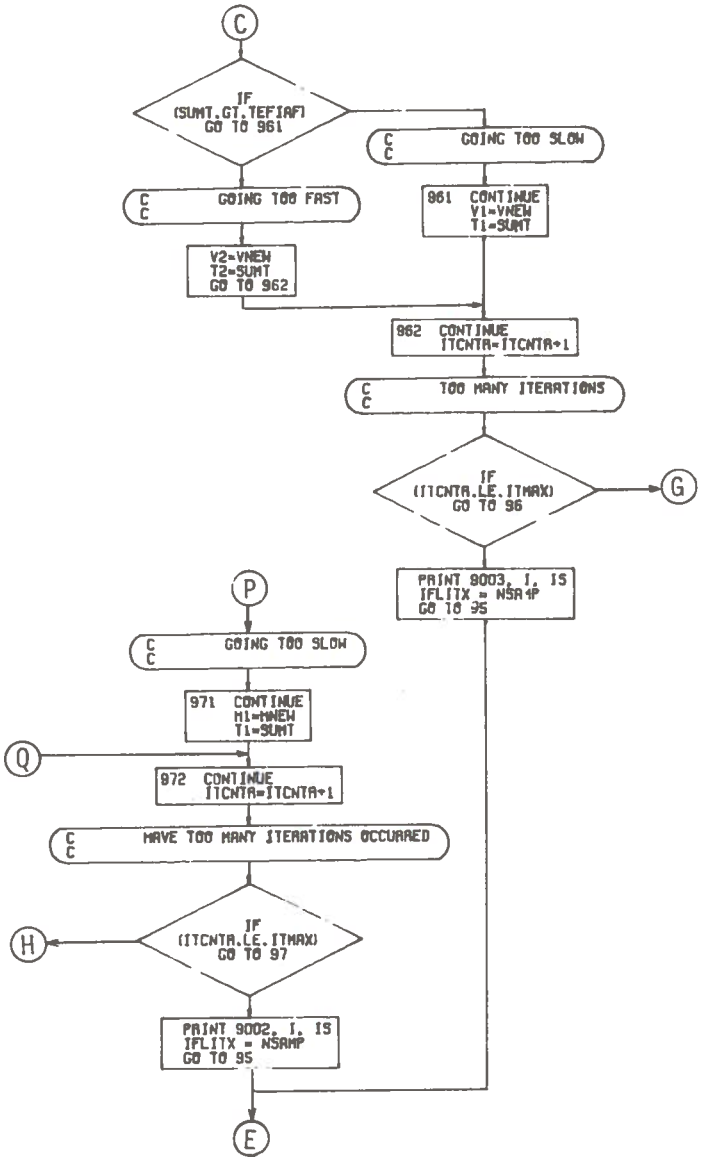
4.2.29.3 RTP Variables

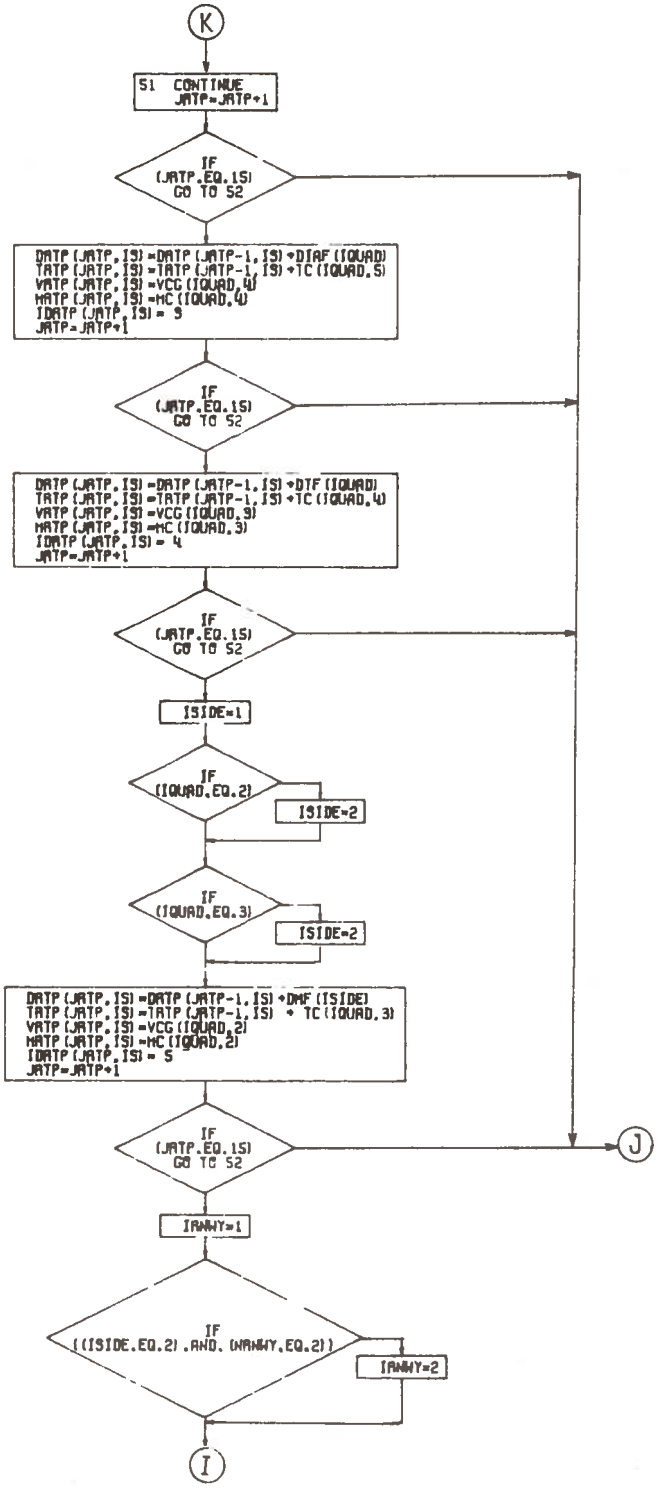
DGS	Descent glide slope
IFLAG	Conflict flag. If IFLAG equals 1, there is a conflict. If it equals zero, there is none.
IPATH	Parallel path number. IPATH = 1 is the primary path; IPATH = 2 through 5 are parallel offset paths.
IRNWY	Number of runway to be used (1 or 2)
ITCNTR	Number of iterations taken to develop an RTP to acceptable accuracy (within RTPTST)
LOAP (IQUAD, INFIX, IPATH)	Contains the index (IS) of the prior airplane being considered on this path. If zero, there is no prior airplane on the path.
SUMT	RTP computed EF to IAF transition time
TEF	Same as TSECAA(IS): actual time of arrival for airplane IS at the entry fix
TEFIAF	Transition time from EF to IAF
TIAFTH	Transition time from IAF to threshold
TOMTH	Transition time from OM to threshold
VAVGG	Average speed from OM to threshold

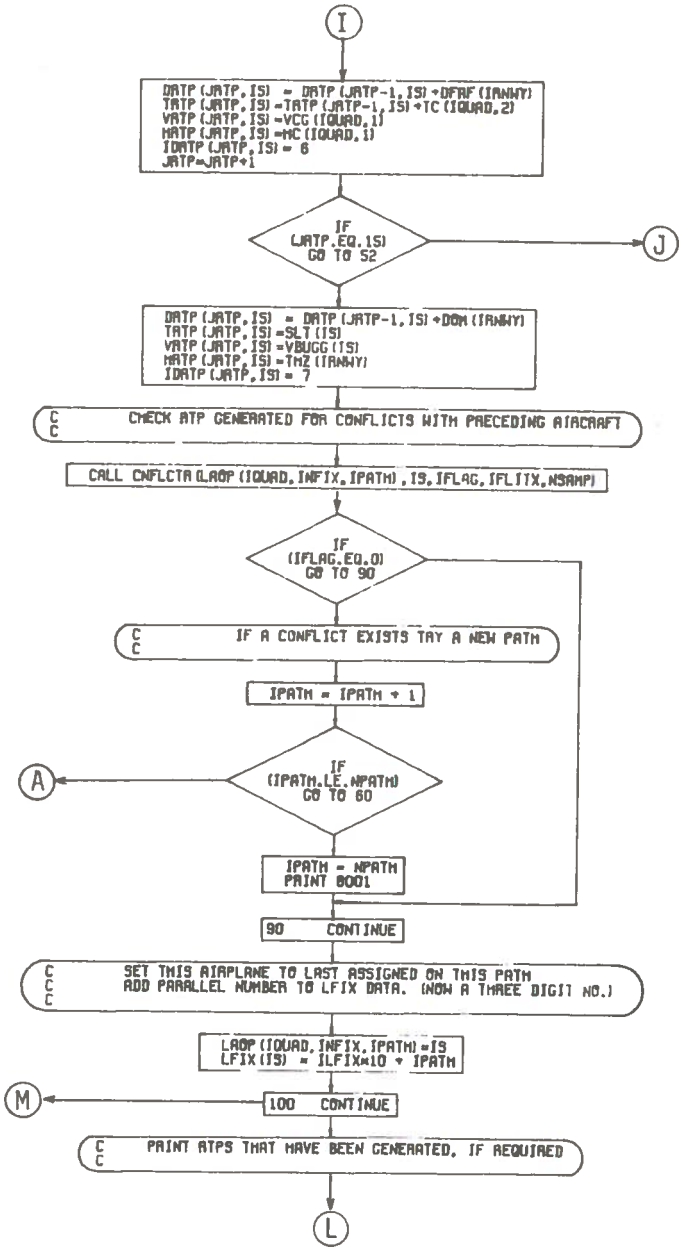

SUBROUTINE ATP (IFLITX, NSAMP)
C THIS SUBROUTINE DETERMINES THE CONFLICT FREE PATH,
C VELOCITIES, AND TIMES TO BE MADE GOOD BY EACH
C AIRPLANE IN THE SEQUENCE LIST
C
DO 10 I = 1, 4
DO 10 J = 1, 3
DO 10 K = 1, 5
LADP(I,J,K) = 0.
10 CONTINUE
C INITIALIZE TO FIRST AIRPLANE, AND GET SEQUENCE POINTER
C
M
DO 100 I = 1, NACA
IS = KEY(I)
C DETERMINE EF NUMBER AND QUADRANT. STORED IN LFIX AS A
C TWO DIGIT NO., THE FIRST DIGIT IS THE QUAD (1,2,3, OR 4)
C AND THE SECOND DIGIT IS THE EF NO. (ILFIX = 1,2, OR 3)
C
ILFIX = LFIX(IS)
IQUAD = ILFIX/10
INFIX = ILFIX - IQUAD*10
C COMPUTE THE IAF TO TH TIME REQUIRED
C
VAVGG = 0.5*(VCG(IQUAD,1) + VBUGG(IS))
TOMTH = 3600.*DDM(1)/VAVGG
TIAFTH = TCSUM(IQUAD) + TOMTH
C DETERMINE EF TIME FOR THIS AIRPLANE
C COMPUTE TIME AT IAF
C COMPUTE TRANSITION TIME FROM EF TO IAF
C INITIALIZE PATH COUNTER
C
TEF = TSECAA(IS)
TIAF = SLT(IS) - TIAFTH
TEFIAF = TIAF - TEF
IPATH = 1
C TEST FOR A GROSS CONFLICT WITH PRIOR AIRCRAFT ON PATH
C
A

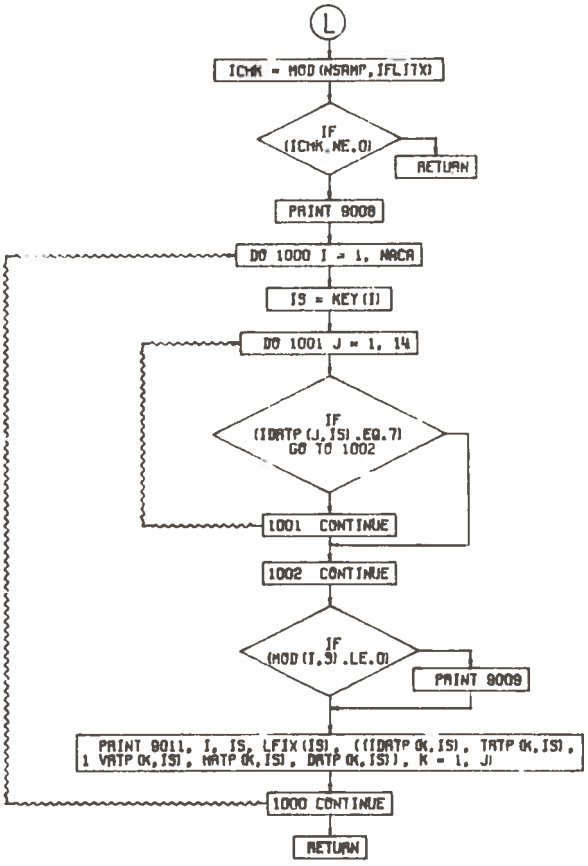

A
60 CONTINUE
CALL CNFLCTG (LAOP (IQUAD, INFIX, IPATH) , IS, TEF, TIAF, IFLAG)
IF
(IFLAG.EQ.0)
GO TO 50
C
C CONFLICT EXISTS, GO TO THE NEXT PATH
IPATH = IPATH + 1
IF
(IPATH.LE.NPATH)
GO TO 60
IPATH = NPATH
PRINT 8000, I, IS
50 CONTINUE
C
C NO CONFLICT, COMPUTE ALT. AT IAF
C HIAF IS A "CUTE" CALCULATION--E.G., IAF ALT FOR FIX 1 IS
C 10 K FEET.
C
HIAF = 9. + INFIX
C
C COMPUTE DESCENT PATH LENGTH
C GET AERO PERFORMANCE TABLE
C
DGS = (ALT (IS) - HIAF) /GRADI
DDP = DEF (IQUAD, INFIX, IPATH) - DGS - DDP
CALL AERO (DUMY, IS, IQUAD, INFIX, 0)
C
C DETERMINE ATP BY ITERATION
C SET ITERATION COUNTER TO ZERO
C
ITCNTR=0
C
C TEST IF ALT IS .LE. TO THE CRITICAL ALTITUDE
C IF SO USE STRATEGY 3
C
IF
(ALT (IS) .LE.HCRIT)
GO TO 93
D
C
C MAKE A CONSTANT MACH/CONSTANT CAS LET DOWN AT ENTRY MACH
C
VNEW = SPKNOT
CALL STUFF2 (VNEW)
CALL TFIND (IS, SUMT)
C
C TEST ATP CALCULATED
C
B

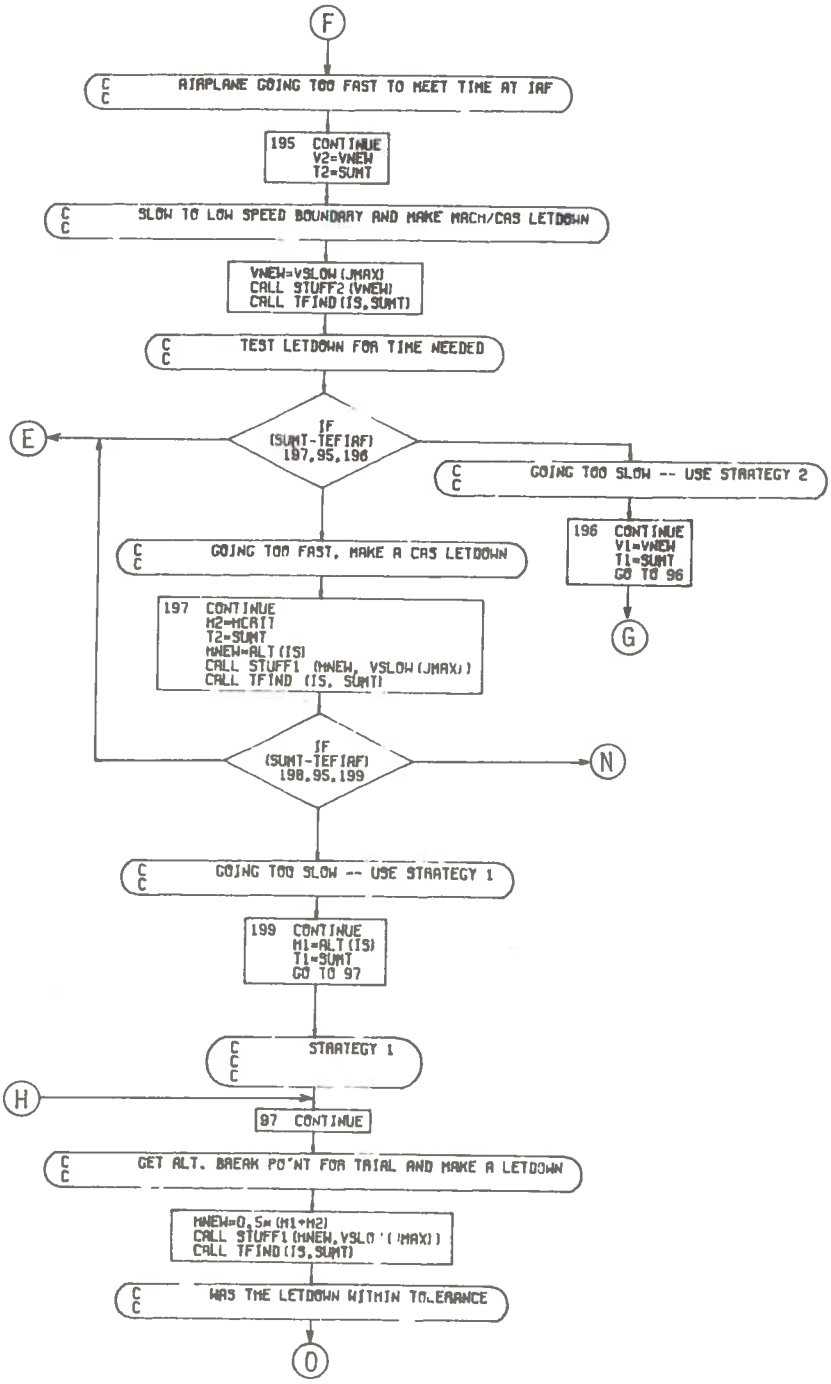

B
IF
(SUMT-TEFIAF)
195,95,295
F
C
C
C
AIRPLANE GOING TOO FAST ON RTP GENERATED FOR CONSTANT MACH/CAS
LETDOWN
295 CONTINUE
V1=VNEW
T2=SUMT
C
C
SPEED TOO HIGH SPEED BOUNDARY -- MAKE A MACH/CAS LETDOWN
VNEW = VFAST(JMAX)
CALL STUFF2(VNEW)
CALL TFIND(IS,SUMT)
IF
(SUMT-TEFIAF)
297,95,296
C
C
GOING TOO SLOW AT HIGH SPEED BOUNDARY
296 CONTINUE
PRINT 9001, I, IS
IFLITX = HSRMP
GO TO 95
C
C
GOING TOO FAST AT HIGH SPEED BOUNDARY -- SLOW DOWN
297 CONTINUE
V2=VNEW
T2=SUMT
C
C
USE STRATEGY 2
GO TO 96
C
C
C
STRATEGY 2
G
96 CONTINUE
VNEW=0.5*(V1+V2)
C
C
CALCULATE DATA FOR A LETDOWN
CALL STUFF2(VNEW)
CALL TFIND(IS,SUMT)
C
C
WAS LETDOWN WITHIN TOLERANCE
IF
(ABS(SUMT-TEFIAF).LT.RTPTST)
GO TO 95
A
E
C
C
NO
C

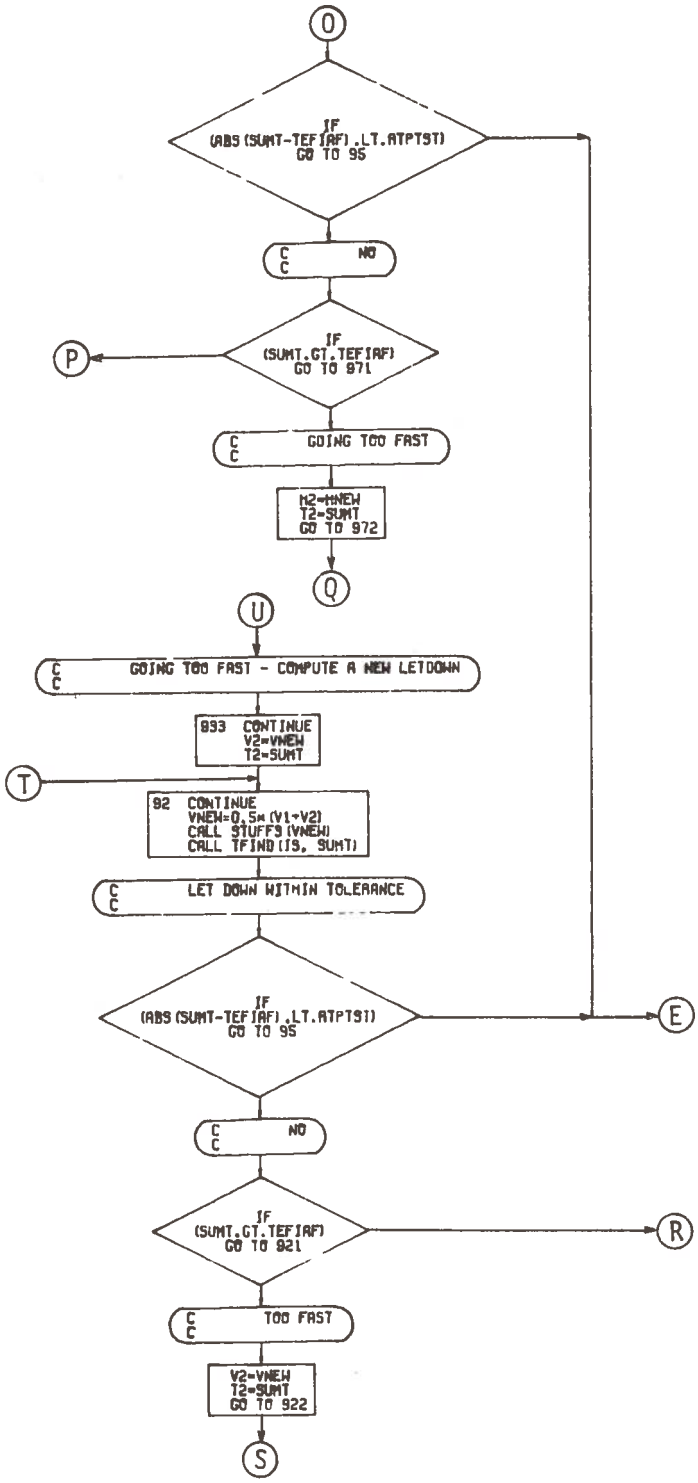

C
IF
(SUMT.GT.TEFIAF)
GO TO 961
C
C GOING TOO SLOW
C
C GOING TOO FAST
961 CONTINUE
V1=VNEW
T1=SUMT
V2=VNEW
T2=SUMT
GO TO 962
962 CONTINUE
ITCNTA=ITCNTA+1
C
C TOO MANY ITERATIONS
IF
(ITCNTA.LE.ITMAX)
GO TO 96
G
PRINT 9003, I, IS
IFLITX = NSAMP
GO TO 95
P
C
C GOING TOO SLOW
971 CONTINUE
M1=MNEW
T1=SUMT
Q
972 CONTINUE
ITCNTA=ITCNTA+1
C
C HAVE TOO MANY ITERATIONS OCCURRED
IF
(ITCNTA.LE.ITMAX)
GO TO 97
H
PRINT 9002, I, IS
IFLITX = NSAMP
GO TO 95
E

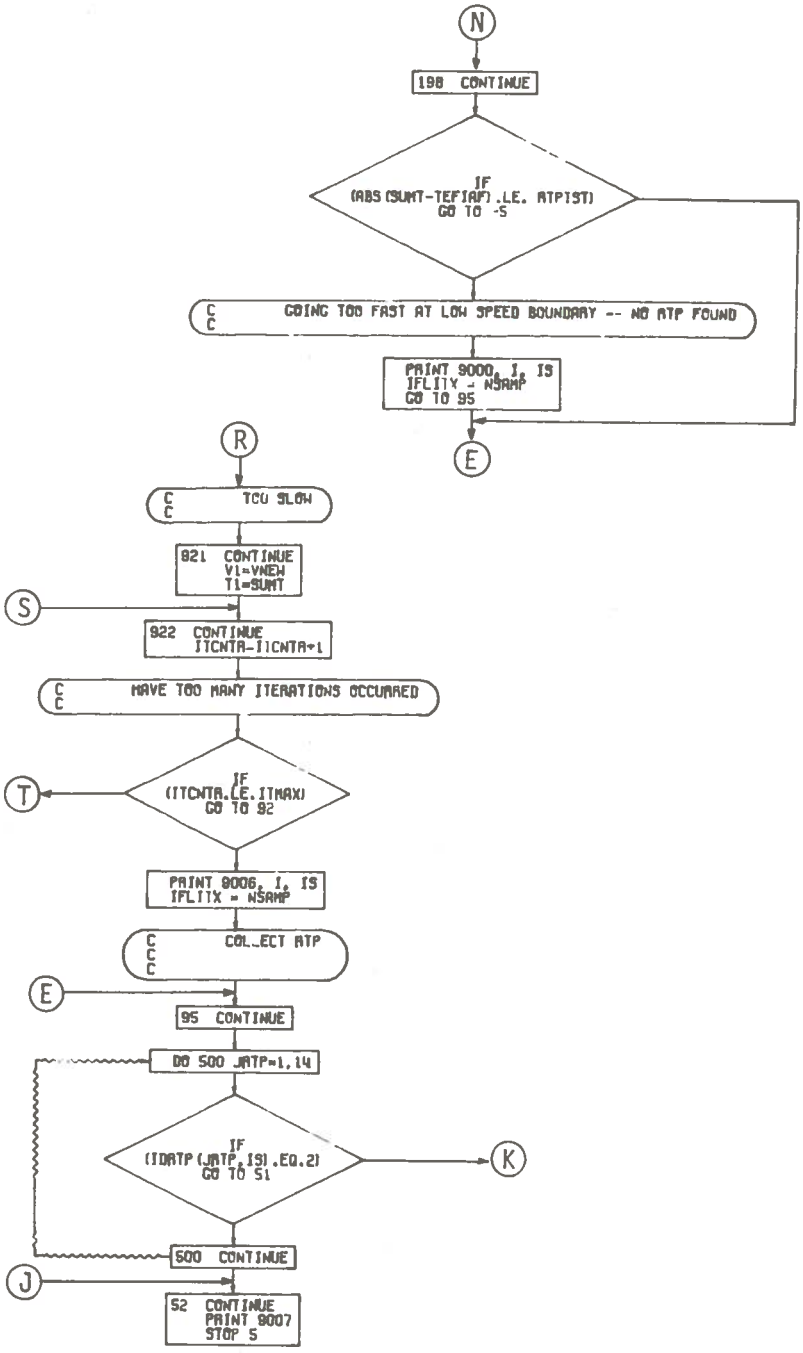

K
51 CONTINUE
JATP=JATP+1
IF
(JATP.EQ.15)
GO TO 52
DATP(JATP,IS)=DATP(JATP-1,IS)+DIAF(IQUAD)
TATP(JATP,IS)=TATP(JATP-1,IS)+TC(IQUAD,5)
VATP(JATP,IS)=VCG(IQUAD,4)
HATP(JATP,IS)=HC(IQUAD,4)
IDATP(JATP,IS)=3
JATP=JATP+1
IF
(JATP.EQ.15)
GO TO 52
DATP(JATP,IS)=DATP(JATP-1,IS)+DTF(IQUAD)
TATP(JATP,IS)=TATP(JATP-1,IS)+TC(IQUAD,4)
VATP(JATP,IS)=VCG(IQUAD,3)
HATP(JATP,IS)=HC(IQUAD,3)
IDATP(JATP,IS)=4
JATP=JATP+1
IF
(JATP.EQ.15)
GO TO 52
ISIDE=1
IF
(IQUAD.EQ.2)
ISIDE=2
IF
(IQUAD.EQ.3)
ISIDE=2
DATP(JATP,IS)=DATP(JATP-1,IS)+DMF(ISIDE)
TATP(JATP,IS)=TATP(JATP-1,IS)+TC(IQUAD,3)
VATP(JATP,IS)=VCG(IQUAD,2)
HATP(JATP,IS)=HC(IQUAD,2)
IDATP(JATP,IS)=5
JATP=JATP+1
IF
(JATP.EQ.15)
GO TO 52
J
IRNWY=1
IF
((ISIDE.EQ.2).AND.(NRNWY.EQ.2))
IRNWY=2
I

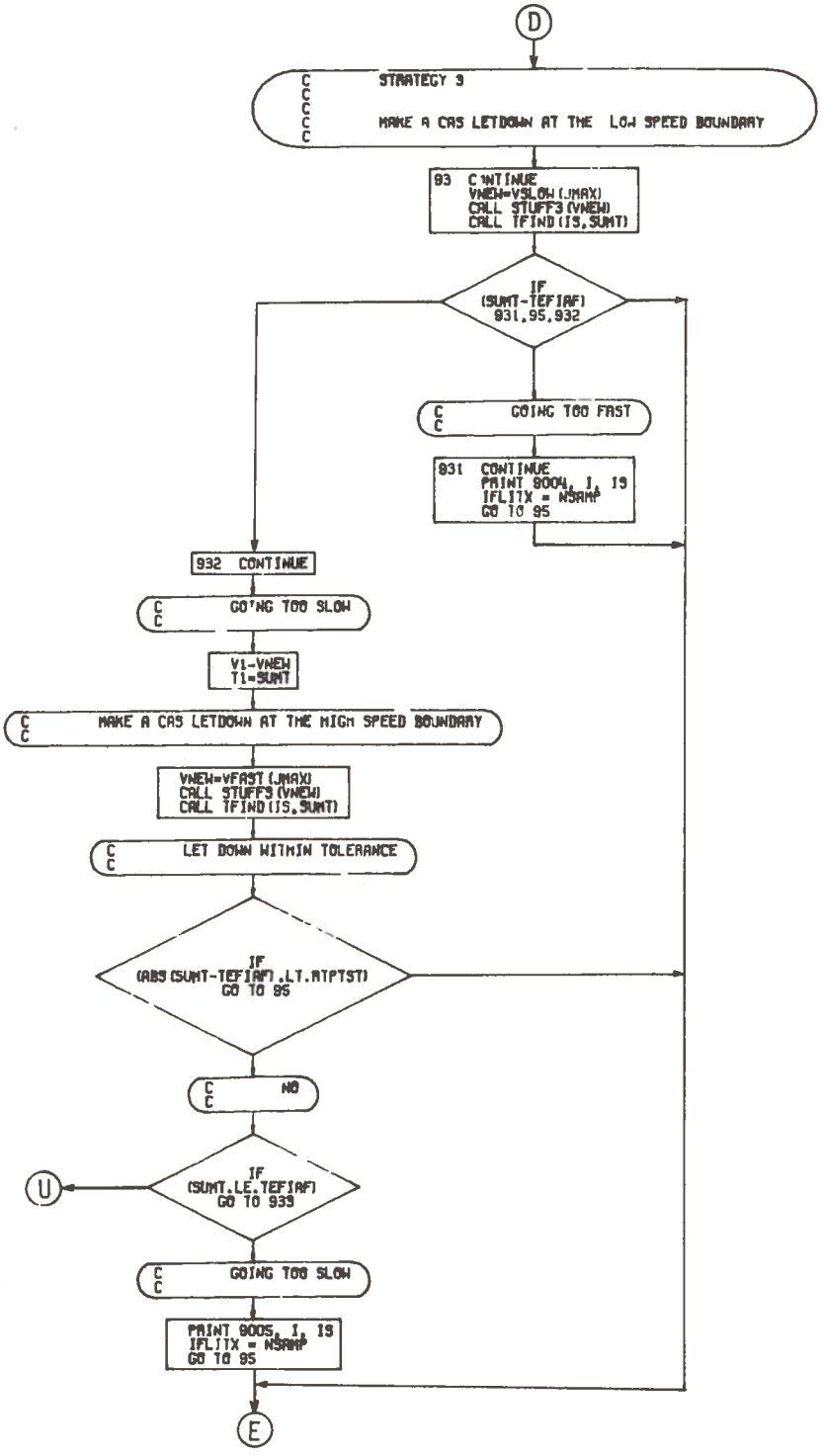

I
DATP(JATP,IS) = DATP(JATP-1,IS)+DFAF(IAWAY)
TATP(JATP,IS)=TATP(JATP-1,IS)+TC(IQUAD,2)
VATP(JATP,IS)=VCG(IQUAD,1)
HATP(JATP,IS)=HC(IQUAD,1)
IDATP(JATP,IS) = 6
JATP=JATP+1
IF (JATP.EQ.1S) GO TO 52
J
DATP(JATP,IS) = DATP(JATP-1,IS)+DDH(IAWAY)
TATP(JATP,IS)=SLT(IS)
VATP(JATP,IS)=VBUGG(IS)
HATP(JATP,IS)=THZ(IAWAY)
IDATP(JATP,IS) = 7
C CHECK ATP GENERATED FOR CONFLICTS WITH PRECEDING AIRCRAFT
CALL CNFLCTR LAOP(IQUAD,INFIX,IPATH),IS,IFLAG,IFLITX,NSAMP)
IF (IFLAG.EQ.0) GO TO 90
C IF A CONFLICT EXISTS TRY A NEW PATH
IPATH = IPATH + 1
IF (IPATH.LE.MPATH) GO TO 60
A
IPATH = MPATH
PRINT 8001
90 CONTINUE
C SET THIS AIRPLANE TO LAST ASSIGNED ON THIS PATH
C ADD PARALLEL NUMBER TO LFIX DATA. (NOW A THREE DIGIT NO.)
LAOP(IQUAD,INFIX,IPATH)=IS
LFIX(IS) = ILFIX*10 + IPATH
M
100 CONTINUE
C PRINT ATPS THAT HAVE BEEN GENERATED, IF REQUIRED
L


L
ICHK = MOD(NSAMP,IFLITX)
IF
(ICHK.NE.0)
RETURN
PRINT 9008
DO 1000 I = 1, NACA
IS = KEY(I)
DO 1001 J = 1, 14
IF
(IDATP(J,IS).EQ.7)
GO TO 1002
1001 CONTINUE
1002 CONTINUE
IF
(MOD(I,5).LE.0)
PRINT 9009
PRINT 9011, I, IS, LFIX(IS), ((IDATP(K,IS), TATP(K,IS),
1 VATP(K,IS), HATP(K,IS), DATP(K,IS)), K = 1, J)
1000 CONTINUE
RETURN


F
C
C AIRPLANE GOING TOO FAST TO MEET TIME AT IAF
195 CONTINUE
V2=VNEW
T2=SUMT
C
C SLOW TO LOW SPEED BOUNDARY AND MAKE MACH/CAS LETDOWN
VNEW=VSLOW(JMAX)
CALL STUFF2 (VNEW)
CALL TFIND(IS,SUMT)
C
C TEST LETDOWN FOR TIME NEEDED
IF
(SUMT-TEFIAF)
197,95,196
E
C
C GOING TOO SLOW -- USE STRATEGY 2
196 CONTINUE
V1=VNEW
T1=SUMT
GO TO 96
G
C
C GOING TOO FAST, MAKE A CAS LETDOWN
197 CONTINUE
M2=MCRIT
T2=SUMT
MNEW=ALT(IS)
CALL STUFF1 (MNEW, VSLOW(JMAX))
CALL TFIND (IS, SUMT)
IF
(SUMT-TEFIAF)
198,95,199
N
C
C GOING TOO SLOW -- USE STRATEGY 1
199 CONTINUE
M1=ALT(IS)
T1=SUMT
GO TO 97
C
C
C STRATEGY 1
H
97 CONTINUE
C
C GET ALT. BREAK PO'NT FOR TRIAL AND MAKE A LETDOWN
MNEW=0.5*(M1+M2)
CALL STUFF1 (MNEW,VSLO'(JMAX))
CALL TFIND(IS,SUMT)
C
C WAS THE LETDOWN WITHIN TOLERANCE
O


O
IF
(ABS(SUMT-TEFIAF).LT.ATPTST)
GO TO 95
C NO
C
IF
(SUMT.GT.TEFIAF)
GO TO 971
P
C GOING TOO FAST
C
M2=MNEW
T2=SUMT
GO TO 972
Q
U
C GOING TOO FAST - COMPUTE A NEW LETDOWN
C
893 CONTINUE
V2=VNEW
T2=SUMT
T
92 CONTINUE
VNEW=0.5*(V1+V2)
CALL STUFF3(VNEW)
CALL TFIND(I9, SUMT)
C LET DOWN WITHIN TOLERANCE
C
IF
(ABS(SUMT-TEFIAF).LT.ATPTST)
GO TO 95
E
C NO
C
IF
(SUMT.GT.TEFIAF)
GO TO 921
R
C TOO FAST
C
V2=VNEW
T2=SUMT
GO TO 922
S


N
198 CONTINUE
IF
(ABS(SUMT-TEFIAF).LE. ATPIST)
GO TO 95
C
C GOING TOO FAST AT LOW SPEED BOUNDARY -- NO ATP FOUND
PRINT 9000, I, IS
IFLITX = NSAMP
GO TO 95
E
R
C
C TOO SLOW
821 CONTINUE
V1=VNEW
T1=SUMT
S
922 CONTINUE
ITCNTA=ITCNTA+1
C
C HAVE TOO MANY ITERATIONS OCCURRED
IF
(ITCNTA.LE.ITMAX)
GO TO 92
T
PRINT 9006, I, IS
IFLITX = NSAMP
C
C COLLECT ATP
C
E
95 CONTINUE
DO 500 JATP=1,14
IF
(IDATP(JATP,IS).EQ.2)
GO TO 51
K
500 CONTINUE
J
52 CONTINUE
PRINT 9007
STOP 5


D
C STRATEGY 9
C
C MAKE A CAS LETDOWN AT THE LOW SPEED BOUNDARY
93 CONTINUE
VNEW=VSLOW(JMAX)
CALL STUFF3(VNEW)
CALL TFIND(IS,SUMT)
IF
(SUMT-TEFIAF)
931,95,932
C GOING TOO FAST
931 CONTINUE
PRINT 9004, I, IS
IFLITX = NSAMP
GO TO 95
932 CONTINUE
C GOING TOO SLOW
V1=VNEW
T1=SUMT
C MAKE A CAS LETDOWN AT THE HIGH SPEED BOUNDARY
VNEW=VFAST(JMAX)
CALL STUFF3(VNEW)
CALL TFIND(IS,SUMT)
C LET DOWN WITHIN TOLERANCE
IF
(ABS(SUMT-TEFIAF).LT.ATPTST)
GO TO 95
C NO
IF
(SUMT.LE.TEFIAF)
GO TO 939
U
C GOING TOO SLOW
PRINT 9005, I, IS
IFLITX = NSAMP
GO TO 95
E

4.2.30 Subroutine SCHLD1

4.2.30.1 Abstract

Subroutine SCHLD1 (IFLITX, ISAMP) determines the scheduled landing times (SLT) and scheduled departure times (SDT) for arrivals and departures in the traffic list. Scheduler 1 gives arrivals absolute priority over departures, which are scheduled as arrival gaps become available. The scheduler calls the subroutines TIMAA, TIMAD, TIMDD, and TIMDA to determine runway minimum constraint times.

4.2.30.2 Data Interface

SCHLD1 Inputs

Through Calling Sequence:

IFLITX	Print control parameter for intermediate output (INLITX = N implies every Nth sample is printed)
ISAMP	Sample number

Through Common Statements:

/TRAFOT/	NACA, NACD, KEY(I), EELT(I), TSECAD(J), ELLT(I), TSECAA(I)

SCHLD1 Outputs

Through Calling Sequence (of SCHWT):

DLYA	Arrival delay array
DLYD	Departure dealy array
DLYEF	Entry fix hold time delay for arrivals

Through Common Statements:

/TRAFOT/	SLT(I), TSECAA(I), SDT(J)

SCHLD1 Calls

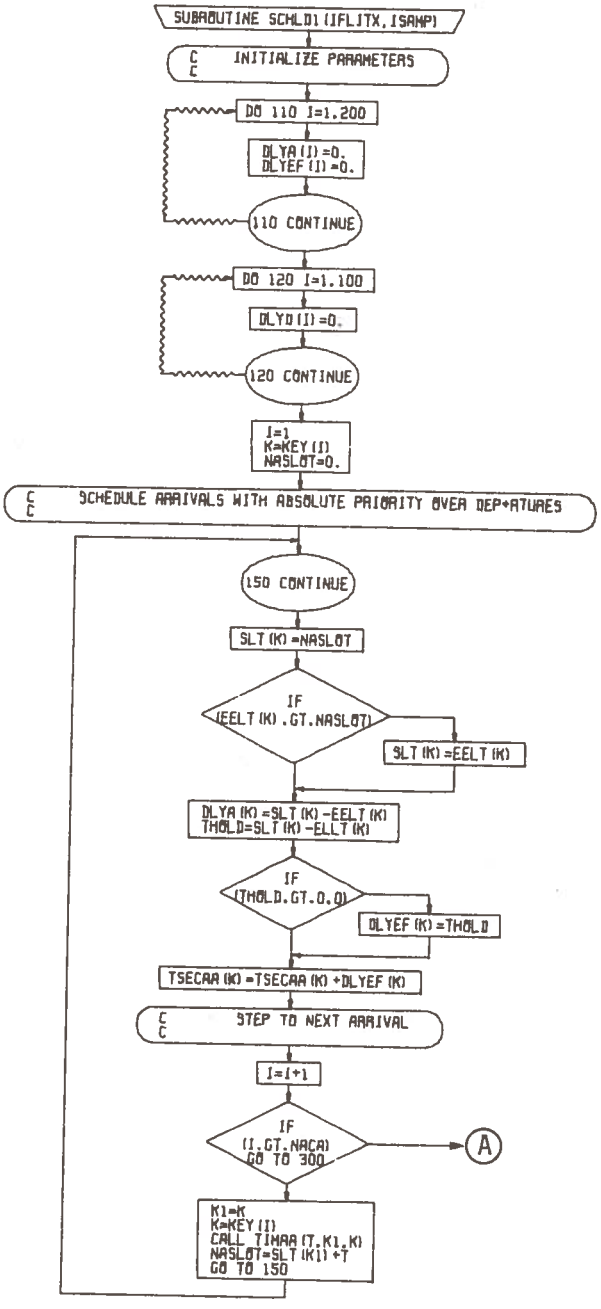
TIMAA, TIMAD, TIMDA, TIMDD, SCHWT

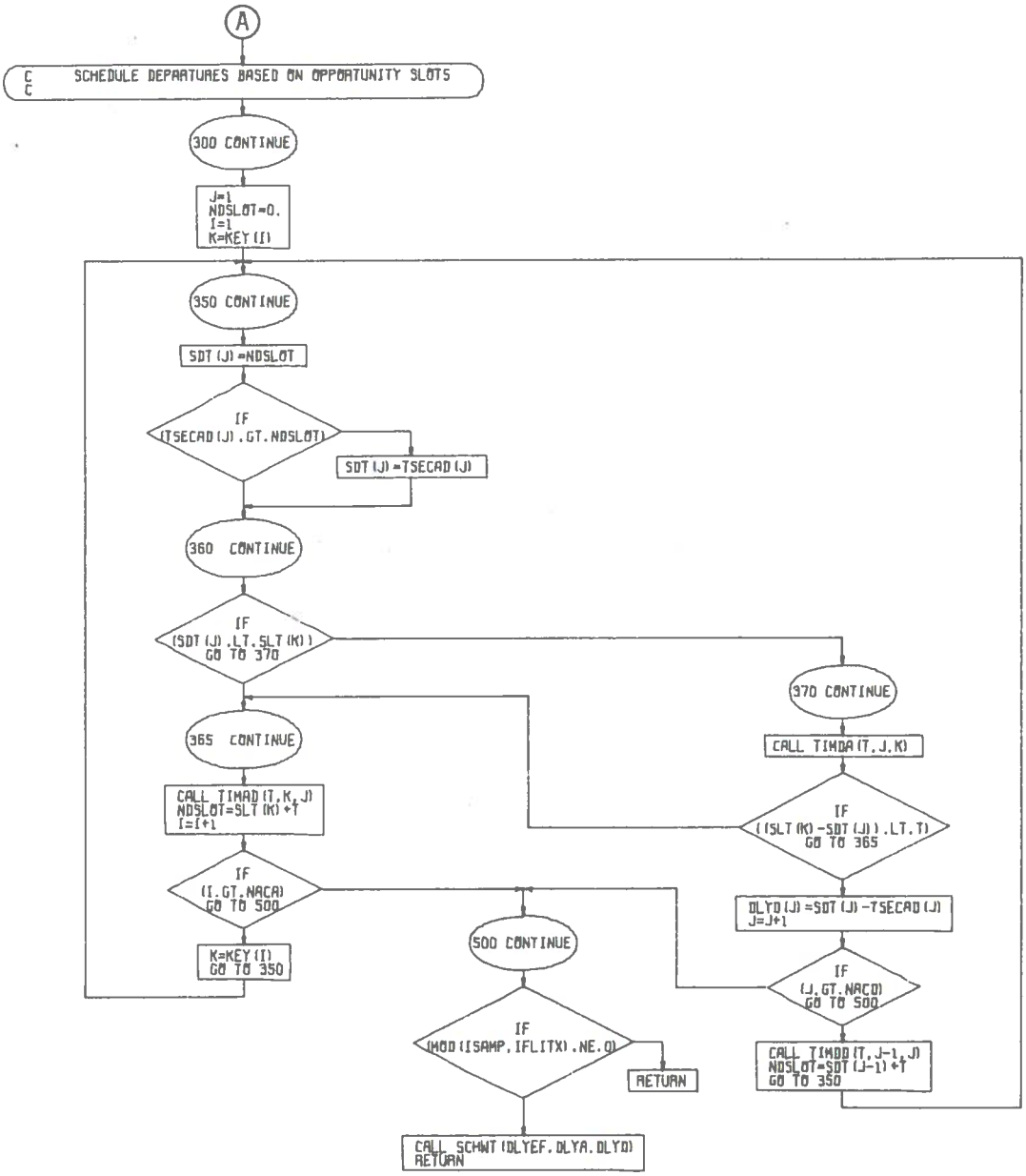
SCHLD1 Called by

STRAD

4.2.30.3 SCHLD1 Variables

DLYA	Arrival scheduled delay [SLT(I) - EELT(I)]
DLYD	Departure scheduled delay [SDT(J) - TSECAD(J)]
DLYEF	Entry fix delay
NASLOT	Next available arrival slot time for runway scheduling
NDSLOT	Next available departure slot time for runway scheduling
T	Interoperation minimum constraint time
THOLD	Hold time of airplane being scheduled


SUBROUTINE SCHLD1 (IFLITX, ISAMP)
C INITIALIZE PARAMETERS
C
DO 110 I=1,200
DLYA(I)=0.
DLYEF(I)=0.
110 CONTINUE
DO 120 I=1,100
DLYD(I)=0.
120 CONTINUE
I=1
K=KEY(I)
NASLOT=0.
C SCHEDULE ARRIVALS WITH ABSOLUTE PRIORITY OVER DEPARTURES
C
150 CONTINUE
SLT(K)=NASLOT
IF (EELT(K).GT.NASLOT)
SLT(K)=EELT(K)
DLYA(K)=SLT(K)-EELT(K)
THOLD=SLT(K)-ELLT(K)
IF (THOLD.GT.0.0)
DLYEF(K)=THOLD
TSECAA(K)=TSECAA(K)+DLYEF(K)
C STEP TO NEXT ARRIVAL
C
I=I+1
IF (I.GT.NACA) GO TO 300
A
K1=K
K=KEY(I)
CALL TIMAA(T,K1,K)
NASLOT=SLT(K1)+T
GO TO 150


A
C SCHEDULE DEPARTURES BASED ON OPPORTUNITY SLOTS
C
300 CONTINUE
J=1
NDSLOT=0.
I=1
K=KEY(I)
350 CONTINUE
SDT(J)=NDSLOT
IF (TSECAD(J).GT.NDSLOT)
SDT(J)=TSECAD(J)
360 CONTINUE
IF (SDT(J).LT.SLT(K)) GO TO 370
370 CONTINUE
CALL TIMDA(T,J,K)
365 CONTINUE
CALL TIMAD(T,K,J)
NDSLOT=SLT(K)+T
I=I+1
IF ((SLT(K)-SDT(J)).LT.T) GO TO 365
IF (I.GT.NACA) GO TO 500
DLYD(J)=SDT(J)-TSECAD(J)
J=J+1
K=KEY(I)
GO TO 350
500 CONTINUE
IF (J.GT.NACD) GO TO 500
IF (MOD(ISAMP,IFLITX).NE.0)
RETURN
CALL TIMDD(T,J-1,J)
NDSLOT=SDT(J-1)+T
GO TO 350
CALL SCHWT(DLYEF,DLYA,DLYD)
RETURN

4.2.31 Subroutine SCHLD2

4.2.31.1 Abstract

Subroutine SCHLD2 (IFLITX, ISAMP) determines the scheduled landing times (SLT) and the scheduled departure times (SDT) for arrivals and departures in the traffic list. Scheduler 2 gives arrivals priority over departures, but leaves sufficient time between arrival-arrival pairs to allow for a departure insertion. Scheduler 2 calls subroutines TIMAA, TIMAD, TIMDA, and TIMDD to determine runway minimum constraint times.

4.2.31.2 Data Interface

SCHLD2 Inputs

Through Calling Sequence:

IFLITX	Print control parameter for intermediate output (IFLITX = N implies every Nth sample is printed)
ISAMP	Sample number

Through Common Statements:

/TRAFOT/	NACA, NACD, KEY(I), EELT(I), TSECAD(J), ELLT(I), TSECAA(I)

SCHLD 2 Outputs

Through Calling Sequence (of SCHWT):

DLYA	Arrival delay array
DLYD	Departure delay array
DLYEF	Entry fix hold time delay for arrivals

Through Common Statements:

/TRAFOT/	SLT(I), TSECAA(I), SDT(J)

SCHLD2 Calls

TIMAA, TIMAD, TIMDA, TIMDD, SCHWT

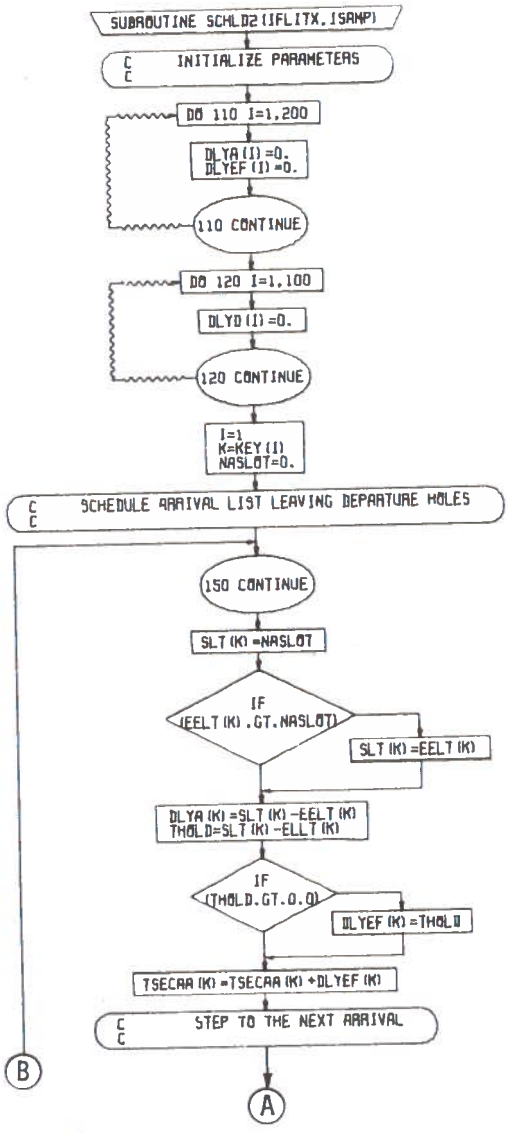
SCHLD2 Called by

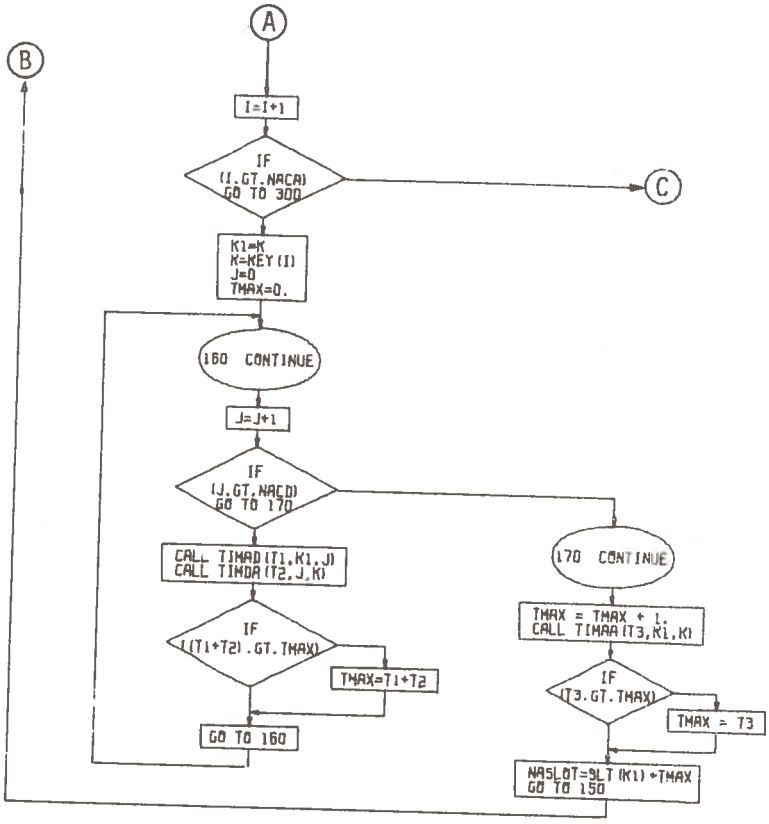
STRAD

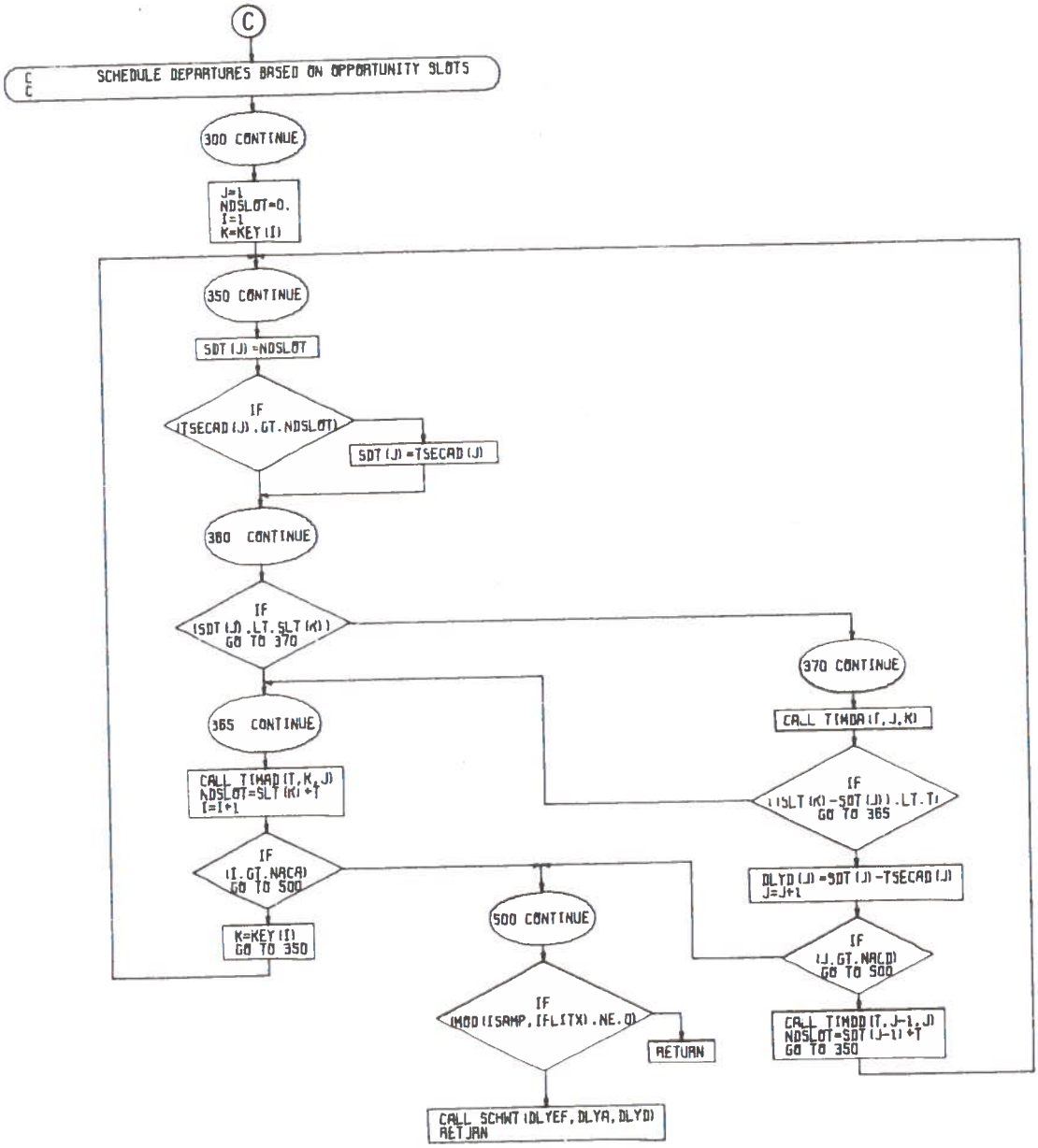
4.2.31.3 SCHLD2 Variables

DLYA	Arrival scheduled delay [SLT(I) - EELT(I)]
DLYD	Departure scheduled delay [SDT(J) - TSECAD(J)]
DLYEF	Entry fix delay
NASLOT	Next available arrival slot time for runway scheduling
NDSLOT	Next available departure slot time for runway scheduling

T1	Interoperation minimum constraint time
THOLD	Hold time of airplane being scheduled
TMAX	Total interoperation constraint time
T	Arrival-departure interoperation constraint time
T2	Departure-arrival interoperation constraint time




A
B
C
I=I+1
IF
(I.GT.NACA)
GO TO 300
K1=K
K=KEY(I)
J=0
TMAX=0.
160 CONTINUE
J=J+1
IF
(J.GT.NACD)
GO TO 170
CALL TIMAD(T1,K1,J)
CALL TIMDA(T2,J,K)
IF
((T1+T2).GT.TMAX)
TMAX=T1+T2
GO TO 160
170 CONTINUE
TMAX = TMAX + 1.
CALL TIMAA(T3,K1,K)
IF
(T3.GT.TMAX)
TMAX = T3
NASLOT=SLT(K1)+TMAX
GO TO 150


C
C SCHEDULE DEPARTURES BASED ON OPPORTUNITY SLOTS
C
300 CONTINUE
J=1
NDSLOT=0.
I=1
K=KEY(I)
350 CONTINUE
SDT(J)=NDSLOT
IF (TSECAD(J).GT.NDSLOT)
SDT(J)=TSECAD(J)
380 CONTINUE
IF (SDT(J).LT.SLT(K)) GO TO 370
365 CONTINUE
CALL TIMAD(T,K,J)
NDSLOT=SLT(K)+T
I=I+1
IF (I.GT.NACA) GO TO 500
K=KEY(I)
GO TO 350
370 CONTINUE
CALL TIMDA(T,J,K)
IF ((SLT(K)-SDT(J)).LT.T) GO TO 365
DLYD(J)=SDT(J)-TSECAD(J)
J=J+1
IF (J.GT.NACD) GO TO 500
CALL TIMDD(T,J-1,J)
NDSLOT=SDT(J-1)+T
GO TO 350
500 CONTINUE
IF (MOD(ISAMP,IFLITX).NE.0)
RETURN
CALL SCHWT(DLYEF,DLYA,DLYD)
RETJRN

4.2.32 Subroutine SCHLD3

4.2.32.1 Abstract

Subroutine SCHLD3 (IFLITX, ISAMP) determines the scheduled landing times (SLT) and scheduled departure times (SDT) for arrivals and departures in the traffic list. Scheduler 3 gives equal priority to arrivals and departures. As long as both arrivals and departures are available for scheduling, alternate operations will be assigned. Scheduler 3 calls for subroutines TIMAA, TIMAD, TIMDD, and TIMDA to determine runway minimum constraint times.

4.2.32.2 Data Interfaces

SCHLD3 Inputs

Through Calling Sequence:

IFLITX	Print control parameter for intermediate output (IFLITX = N implies every Nth sample is printed)
ISAMP	Sample number

Through Common Statements:

/TRAFOT/	NACA, NACD, KEY(I), EELT(I), TSECSD(J), ELLT(J), TSECAA(J)

SCHLD3 Outputs

Through Calling Sequence (of SCHWT):

DLYA	Arrival delay array
DLYD	Departure delay array
DLYEF	Entry fix hold time delay for arrivals

Through Common Statements:

/TRAFOT/	SLT(I), TSECAA(I), SDT(J)

SCHLD3 Calls

TIMAA, TIMAD, TIMDA, TIMDD, SCHWT

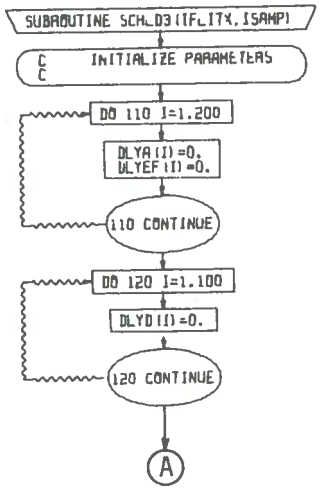
SCHLD3 Called by

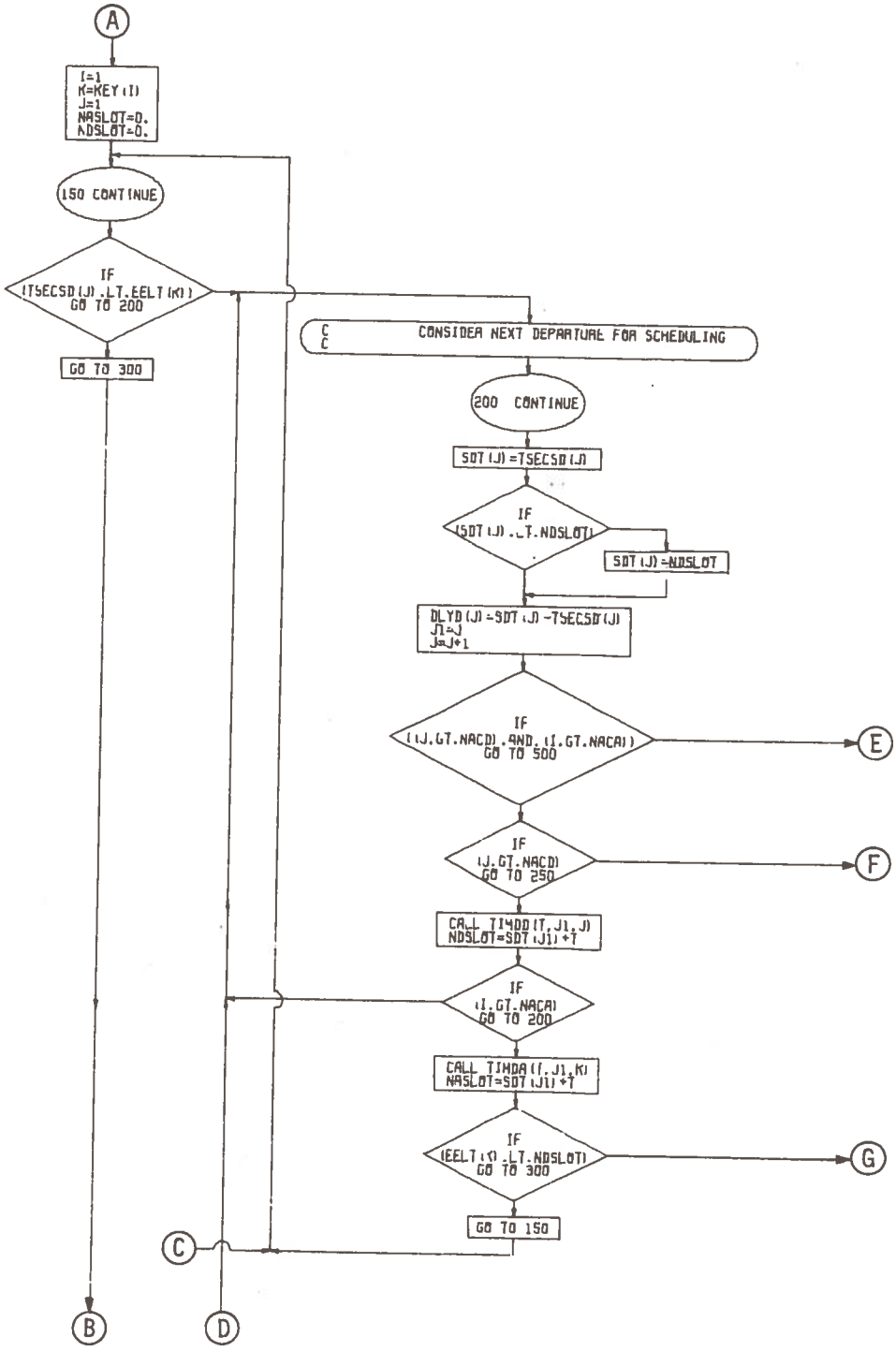
STRAD

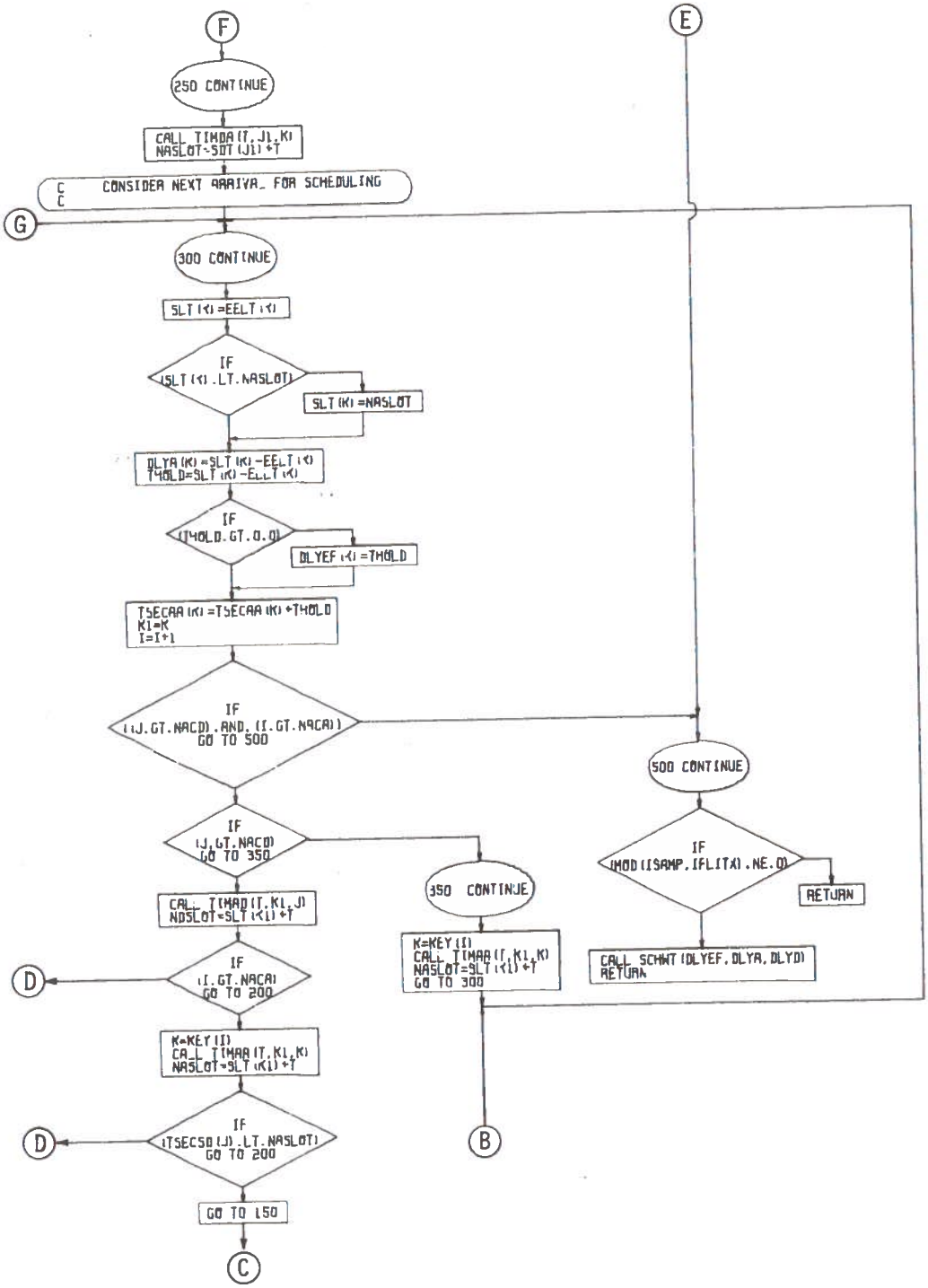
4.2.32.3 SCHLD 3 Variables

DLYA	Arrival scheduled delay [SLT(I) - EELT(I)]
DLYD	Departure scheduled delay [SDT(J) - TSECSD(J)]
DLYEF	Entry fix delay
NASLOT	Next available arrival slot time for runway scheduling

NDSLOT	Next available departure slot time for runway scheduling
T	Interoperation minimum constraint time
THOLD	Hold time of airplane being scheduled




A
I=1
K=KEY(I)
J=1
NASLOT=0.
NDSLOT=0.
150 CONTINUE
IF
(TSECSD(J).LT.EELT(K))
GO TO 200
GO TO 300
C
C CONSIDER NEXT DEPARTURE FOR SCHEDULING
200 CONTINUE
SDT(J)=TSECSD(J)
IF
(SDT(J).LT.NDSLOT)
SDT(J)=NDSLOT
DLYD(J)=SDT(J)-TSECSD(J)
J1=J
J=J+1
IF
((J.GT.NACD).AND.(I.GT.NACA))
GO TO 500
E
IF
(J.GT.NACD)
GO TO 250
F
CALL TIMDD(T,J1,J)
NDSLOT=SDT(J1)+T
IF
(I.GT.NACA)
GO TO 200
CALL TIMDA(T,J1,K)
NASLOT=SDT(J1)+T
IF
(EELT(K).LT.NDSLOT)
GO TO 300
G
GO TO 150
C
B
D


F
250 CONTINUE
CALL TIMDA(T,J1,K)
NASLOT=SDT(J1)+T
C CONSIDER NEXT ARRIVAL FOR SCHEDULING
C
G
300 CONTINUE
SLT(K)=EELT(K)
IF (SLT(K).LT.NASLOT)
SLT(K)=NASLOT
DLYA(K)=SLT(K)-EELT(K)
THOLD=SLT(K)-ELLT(K)
IF (THOLD.GT.0.0)
DLYEF(K)=THOLD
TSECAA(K)=TSECAA(K)+THOLD
K1=K
I=I+1
IF ((J.GT.NACD).AND.(I.GT.NACA))
GO TO 500
IF (J.GT.NACD)
GO TO 350
CALL TIMAD(T,K1,J)
NDSLOT=SLT(K1)+T
IF (I.GT.NACA)
GO TO 200
K=KEY(I)
CALL TIMAA(T,K1,K)
NASLOT=SLT(K1)+T
IF (TSECSD(J).LT.NASLOT)
GO TO 200
GO TO 150
C
D
D
350 CONTINUE
K=KEY(I)
CALL TIMAA(T,K1,K)
NASLOT=SLT(K1)+T
GO TO 300
B
E
500 CONTINUE
IF (MOD(ISAMP,IFLIT).NE.0)
RETURN
CALL SCHWT(DLYEF,DLYA,DLYD)
RETURN

4.2.33 Subroutine SCHWT

4.2.33.1 Abstract

Subroutine SCHWT prints the sequencer and scheduler output, which is divided into the arrival and departure runway algorithm data.

4.2.33.2 Data Interfaces

SCHWT Inputs

Through Calling Sequence:
AHOLDT, DLAY, DLAYD

Through Common Statements:
/TRAFOT/ NACA, NACD, EELT, ELLT, KEY, SDT, SLT

SCHWT Outputs

Through Printing:

AHOLDT, DLA, DLAYD, EELT, ELLT, KEY, SDT, SLT

SCHWT Calls

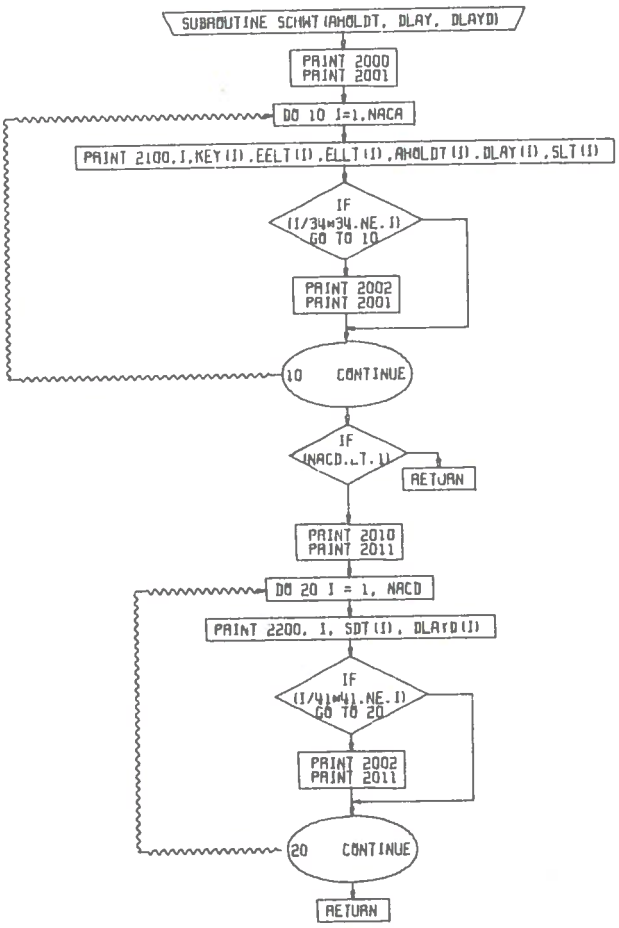
None

SCHWT Called by

SCHLD1, SCHLD2, SCHLD3

4.2.33.3 SCHWT Variables

AHOLDT	Arrival airplane hold time at the entry fix (in seconds)
DLAY	Arrival delay time (in seconds)
DLAYD	Departure delay time (in seconds)


SUBROUTINE SCHWT(AHOLDT, DLAY, DLAYD)
PRINT 2000
PRINT 2001
DO 10 I=1,NACA
PRINT 2100,I,KEY(I),EELT(I),ELLT(I),AHOLDT(I),DLAY(I),SLT(I)
IF
(I/34*34.NE.I)
GO TO 10
PRINT 2002
PRINT 2001
10 CONTINUE
IF
(NACD.LT.1)
RETURN
PRINT 2010
PRINT 2011
DO 20 I = 1, NACD
PRINT 2200, I, SDT(I), DLAYD(I)
IF
(I/41*41.NE.I)
GO TO 20
PRINT 2002
PRINT 2011
20 CONTINUE
RETURN

4.2.34 Subroutine SEQUEN

4.2.34.1 Abstract

Subroutine SEQUEN determines the earliest and latest runway time for each arrival in the traffic list. SEQUEN combines the outer marker to threshold time (from REFBUG) and the entry fix to initial approach time limits (from FASLOW) with the nominal IAF to outer marker time (from AERO1). The sequencer then reorders the traffic list based on earliest threshold capability.

4.2.34.2 Data Interfaces

SEQUEN Inputs

Through Calling Sequence:

IFLITX	Print control variable
NSAMP	Sample currently under consideration (used for print control)

Through Common Statements:

/AEROIN/	CABPRS
/AERO1O/	TCSUM, THPRES, VCG
/ATMO/	PRESUR
/GEOMIN/	NRNWY
/GEOMOT/	DOM
/TRAFOT/	NACA, ALT, LFIX, VBUGG, TSECAA

SEQUEN Outputs

Through Common Statements:

/TRAFOT/	KEY, ELLT, EELT

SEQUEN Calls

AERO	
REFBUG	
FASLOW	Provides fastest and slowest letdown times from the EF to the IAF
WETHER	Provides forecast atmospheric pressure at requested altitude

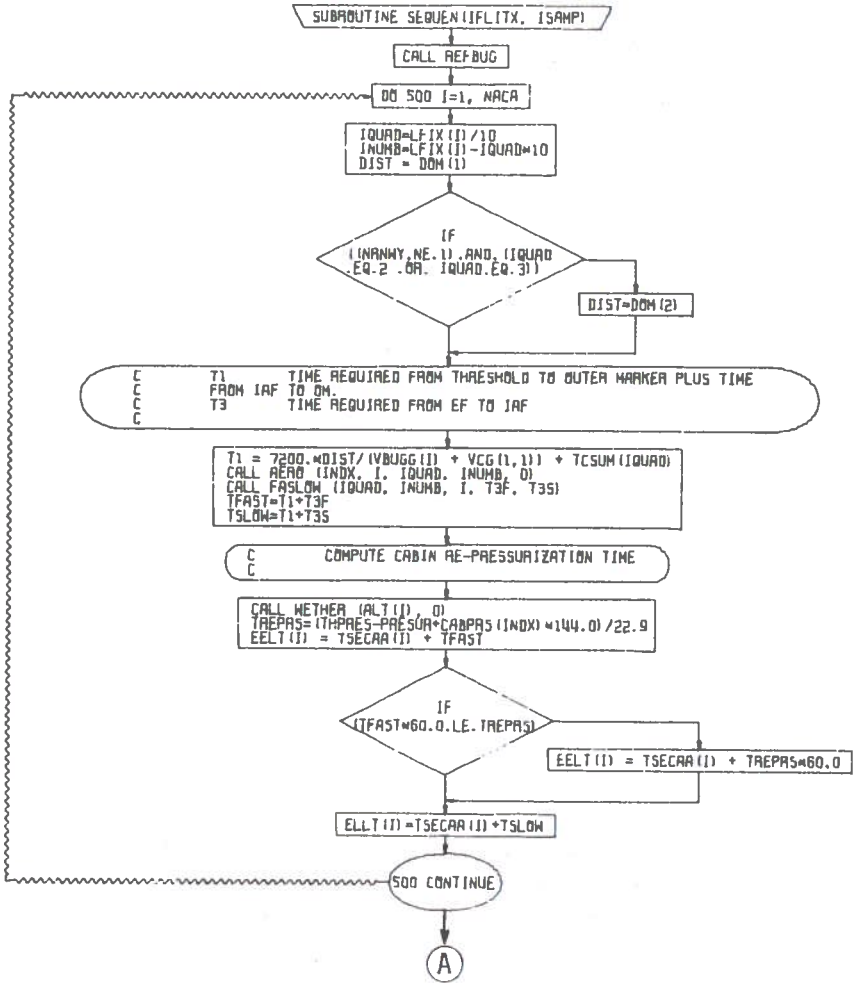
SEQUEN Called by

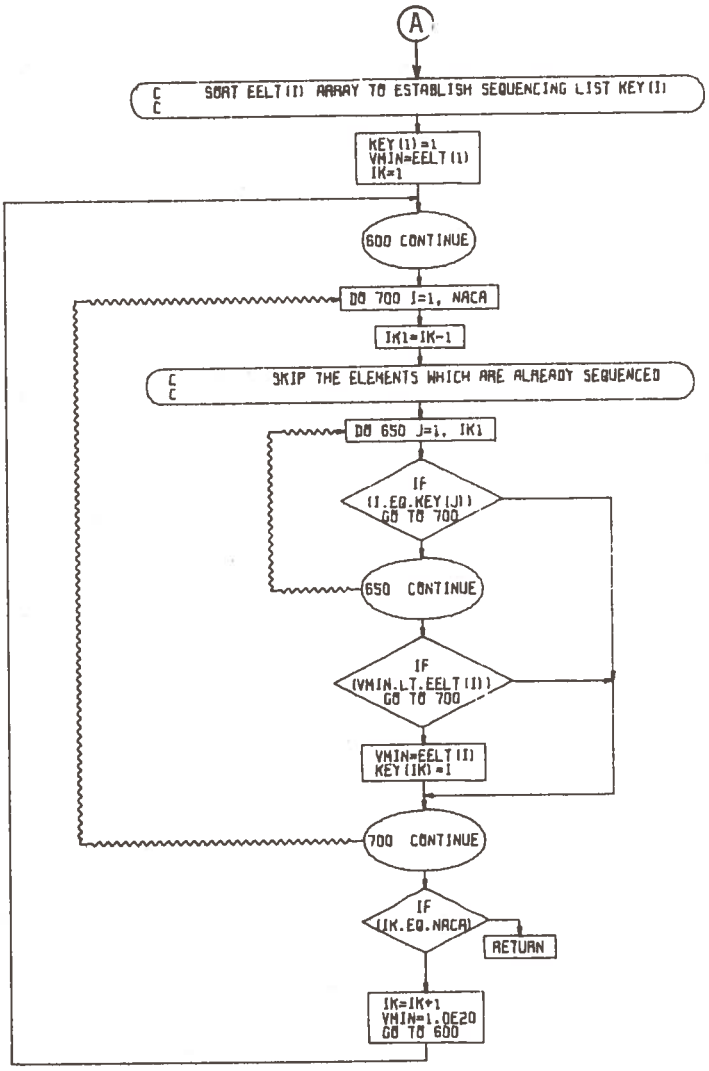
STRAD

4.2.34.3 SEQUEN Variables

INDX	Type of airplane being sequenced (from subroutine AERO)
INUMB	Entry fix number in quadrant IQUAD
IQUAD	Quadrant in which airplane is arriving

TFAST	Fastest time airplane can fly from EF to TH
TSLOW	Slowest time airplane can fly from EF to TH
T1	Outer marker (OM) to threshold (TH) time plus the IAF to OM time
T3F	Fastest EF to IAF time airplane can fly
T3S	Slowest EF to IAF time airplane can fly
VMIN	Sort variable for earliest estimated landing times. Used to create KEY(-).
DIST	Distance from threshold to outer marker for runway 1 or 2
TREPRS	Cabin repressurization time




A
C SORT EELT(I) ARRAY TO ESTABLISH SEQUENCING LIST KEY(I)
C
KEY(1)=1
VMIN=EELT(1)
IK=1
600 CONTINUE
DO 700 J=1, NACA
IK1=IK-1
C SKIP THE ELEMENTS WHICH ARE ALREADY SEQUENCED
C
DO 650 J=1, IK1
IF
(I.EQ.KEY(J))
GO TO 700
650 CONTINUE
IF
(VMIN.LT.EELT(I))
GO TO 700
VMIN=EELT(I)
KEY(IK)=I
700 CONTINUE
IF
(IK.EQ.NACA)
RETURN
IK=IK+1
VMIN=1.0E20
GO TO 600

4.2.35 Subroutine SETN

4.2.35.1 Abstract

Subroutine SETN initializes variables before each data set is run. The variables may be initialized to any value (zero is now used in the model).

4.2.35.2 Data Interfaces

SETN Inputs

Through Calling Sequence:
NX

Through Common Statements:
/FLGTXO/ All variables
/STATOT/ All variables

SETN Outputs

Through Common Statements:
/FLGTXO/ All variables
/STATOT/ All variables

SETN Calls

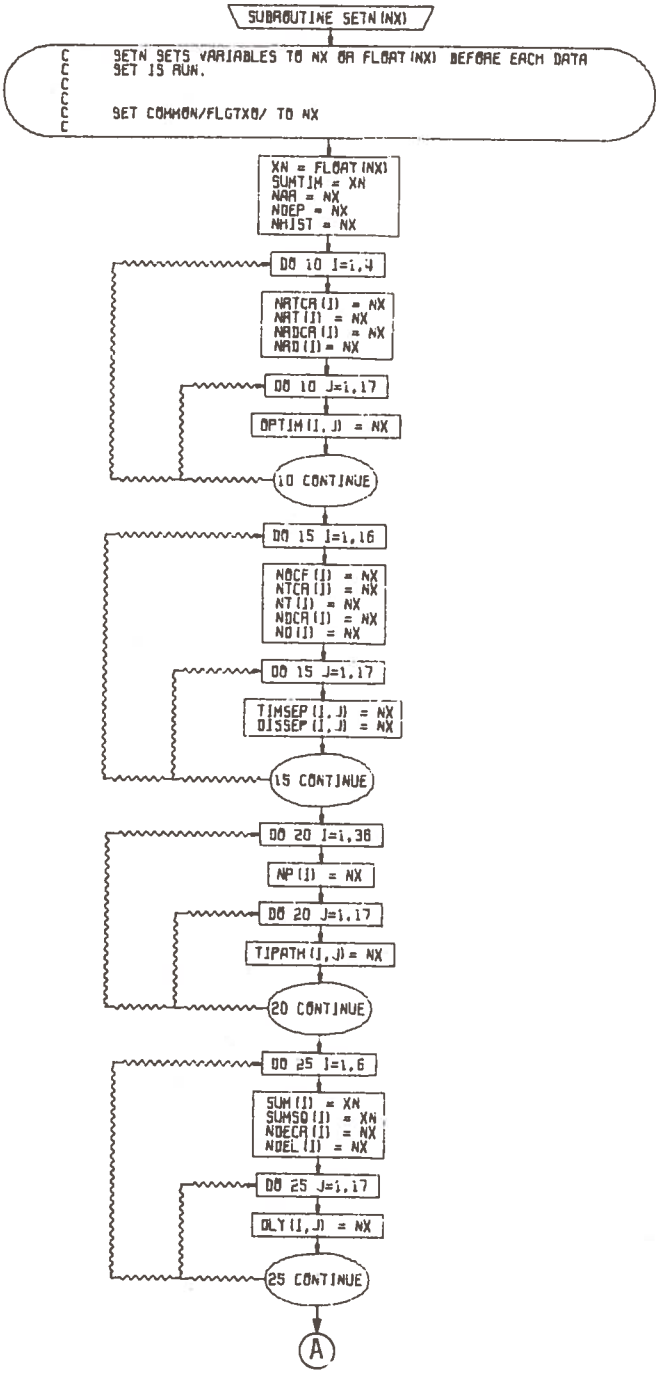
None

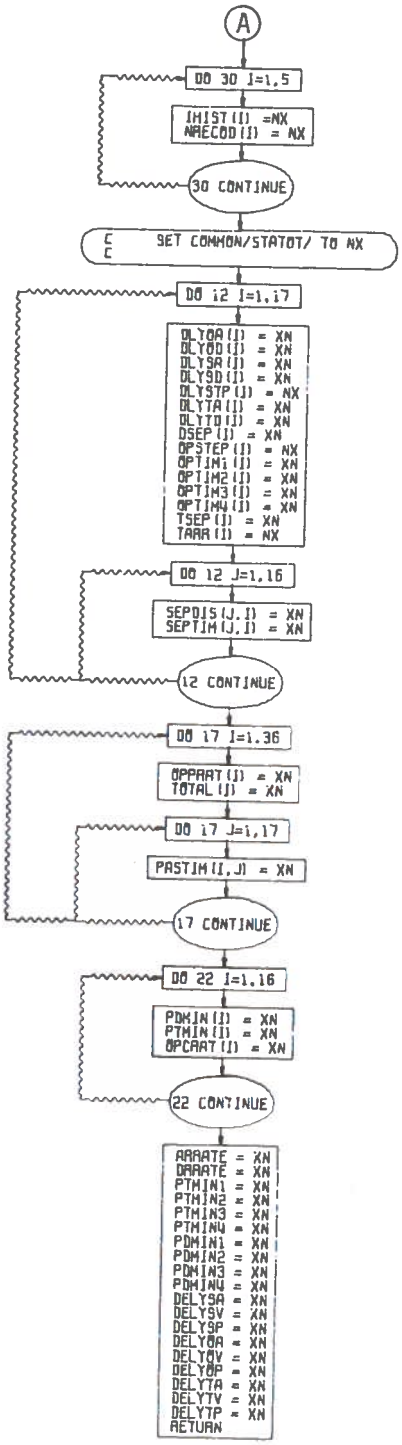
SETN Called by

STRAD

4.2.35.3 SETN Variables

XN	Floating point value of initialization parameter passed by the calling sequence


SUBROUTINE SETN(NX)
C SETN SETS VARIABLES TO NX OR FLOAT(NX) BEFORE EACH DATA
C SET IS RUN.
C
C
C SET COMMON/FLGTXO/ TO NX
C
XN = FLOAT(NX)
SUMTIM = XN
NAR = NX
NDEP = NX
NMIST = NX
DO 10 I=1,4
NRTCR(I) = NX
NRT(I) = NX
NRDCR(I) = NX
NRD(I) = NX
DO 10 J=1,17
OPTIM(I,J) = NX
10 CONTINUE
DO 15 I=1,16
NOCF(I) = NX
NTCR(I) = NX
NT(I) = NX
NDCR(I) = NX
ND(I) = NX
DO 15 J=1,17
TIMSEP(I,J) = NX
DISSEP(I,J) = NX
15 CONTINUE
DO 20 I=1,38
NP(I) = NX
DO 20 J=1,17
TIPATH(I,J) = NX
20 CONTINUE
DO 25 I=1,6
SUM(I) = XN
SUMSQ(I) = XN
NDECR(I) = NX
NDEL(I) = NX
DO 25 J=1,17
DLY(I,J) = NX
25 CONTINUE
A


A
DO 30 I=1,5
IHIST(I) =NX
NRECOD(I) = NX
30 CONTINUE
C SET COMMON/STATOT/ TO NX
C
DO 12 I=1,17
DLY0A(I) = XN
DLY0D(I) = XN
DLY9A(I) = XN
DLY9D(I) = XN
DLY9TP(I) = NX
DLYTA(I) = XN
DLYTD(I) = XN
DSEP(I) = XN
OPSTEP(I) = NX
OPTIM1(I) = XN
OPTIM2(I) = XN
OPTIM3(I) = XN
OPTIM4(I) = XN
TSEP(I) = XN
TARR(I) = NX
DO 12 J=1,16
SEPDIS(J,I) = XN
SEPTIM(J,I) = XN
12 CONTINUE
DO 17 I=1,36
OPPART(I) = XN
TOTAL(I) = XN
DO 17 J=1,17
PASTIM(I,J) = XN
17 CONTINUE
DO 22 I=1,16
PDMIN(I) = XN
PTMIN(I) = XN
OPCART(I) = XN
22 CONTINUE
ARRATE = XN
DARATE = XN
PTMIN1 = XN
PTMIN2 = XN
PTMIN3 = XN
PTMIN4 = XN
PDMIN1 = XN
PDMIN2 = XN
PDMIN3 = XN
PDMIN4 = XN
DELY9A = XN
DELY9V = XN
DELY9P = XN
DELY0A = XN
DELY0V = XN
DELY0P = XN
DELYTA = XN
DELYTV = XN
DELYTP = XN
RETURN

4.2.36 Subroutine SHELL

4.2.36.1 Abstract

Subroutine SHELL (X,K,N) orders X in ascending order. N is the length of X. K is an index array for ordering other arrays in the same order as X.

4.2.36.2 Data Interfaces

SHELL Inputs

Through Calling Sequence:

X	Array to be sorted in ascending order
N	Length of X
K	Array initially set so that K(1)=1, . . ., K(N) = N, then subsequently ordered the same as X after X is sorted

SHELL Outputs

Through Calling Sequence:

X	Now sorted
K	Now sorted

SHELL Calls

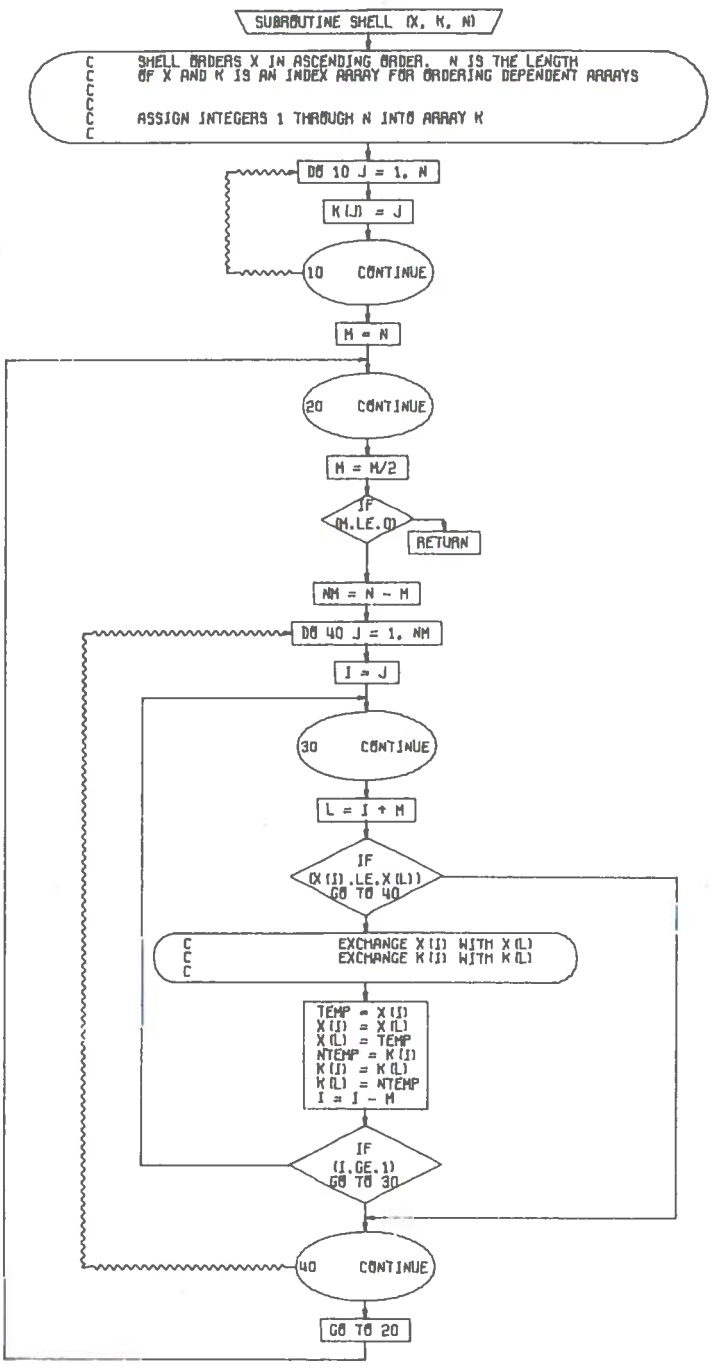
None

SHELL Called by

CNFLCTR

4.2.36.3 SHELL Variables

None


SUBROUTINE SHELL (X, K, N)
C SHELL ORDERS X IN ASCENDING ORDER. N IS THE LENGTH
C OF X AND K IS AN INDEX ARRAY FOR ORDERING DEPENDENT ARRAYS
C
C ASSIGN INTEGERS 1 THROUGH N INTO ARRAY K
DO 10 J = 1, N
K(J) = J
10 CONTINUE
M = N
20 CONTINUE
M = M/2
IF (M.LE.0)
RETURN
NM = N - M
DO 40 J = 1, NM
I = J
30 CONTINUE
L = I + M
IF (X(I).LE.X(L)) GO TO 40
C EXCHANGE X(I) WITH X(L)
C EXCHANGE K(I) WITH K(L)
C
TEMP = X(I)
X(I) = X(L)
X(L) = TEMP
NTEMP = K(I)
K(I) = K(L)
K(L) = NTEMP
I = I - M
IF (I.GE.1) GO TO 30
40 CONTINUE
GO TO 20

4.2.37 Subroutine SHELLX

4.2.37.1 Abstract

Subroutine SHELLX (Y,K,N) orders array Y according to the index array K.

4.2.37.2 Data Interfaces

SHELLX Inputs

Through Calling Sequence:

Y	Array to be reordered
K	Array containing indices determining how Y is to be reordered
N	Length of Y and K arrays

SHELLX Outputs

Through Calling Sequence:

Y	In reordered form

SHELLX Calls

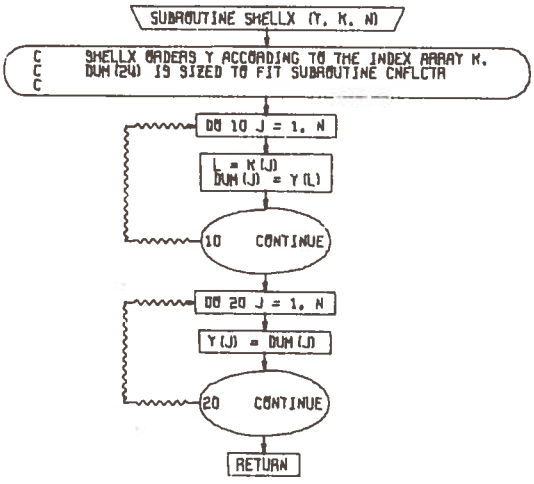
None

SHELLX Called by

CNFLCTR

4.2.37.3 SHELLX Variables

DUM(28)	Temporary storage array that must be at least as large as largest array used in SHELLX. DUM is sized to correspond to R1 and R2 in CNFLCTR.



4.2.38 Subroutine SORT

4.2.38.1 Abstract

Subroutine SORT (X,N) is a floating point sort routine that resequences an array in descending order.

4.2.38.2 Data Interfaces

SORT Inputs

Through Calling Sequence:

X	Array to be sorted
N	Length of array X

SORT Outputs

Through Calling Sequence:

X	The sorted array

SORT Calls

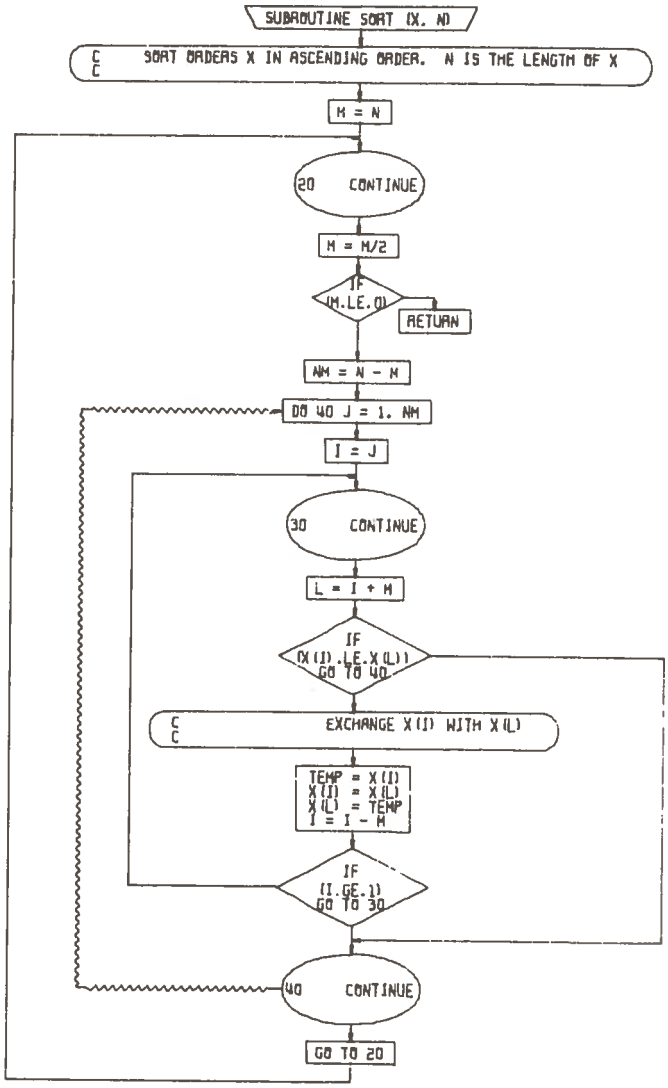
None

SORT Called by

Traffic

4.2.38.3 SORT Variables

None


SUBROUTINE SORT (X, N)
C SORT ORDERS X IN ASCENDING ORDER. N IS THE LENGTH OF X
C
M = N
20 CONTINUE
M = M/2
IF (M.LE.0)
RETURN
NM = N - M
DO 40 J = 1, NM
I = J
30 CONTINUE
L = I + M
IF (X(I).LE.X(L)) GO TO 40
C EXCHANGE X(I) WITH X(L)
C
TEMP = X(I)
X(I) = X(L)
X(L) = TEMP
I = I - M
IF (I.GE.1) GO TO 30
40 CONTINUE
GO TO 20

4.2.39 Subroutine STATIC

4.2.39.1 Abstract

The subroutine STATIC follows flight execution. The subroutine reduces all data generated by sampling to probability values, together with distribution means and variances. The subroutine output in COMMON/STATOT/ is passed to subroutine WTOUT for formatting and printing.

4.2.39.2 Data Interfaces

STATIC Inputs

Through Common Statements:

/FLGTXO/	All variables except IHIST, NRECOD, NHIST
/STATIN/	All variables except DMIN, TMIN, DCRIT

STATIC Outputs

Through Common Statements:

/STATOT/	All variables

STATIC Calls

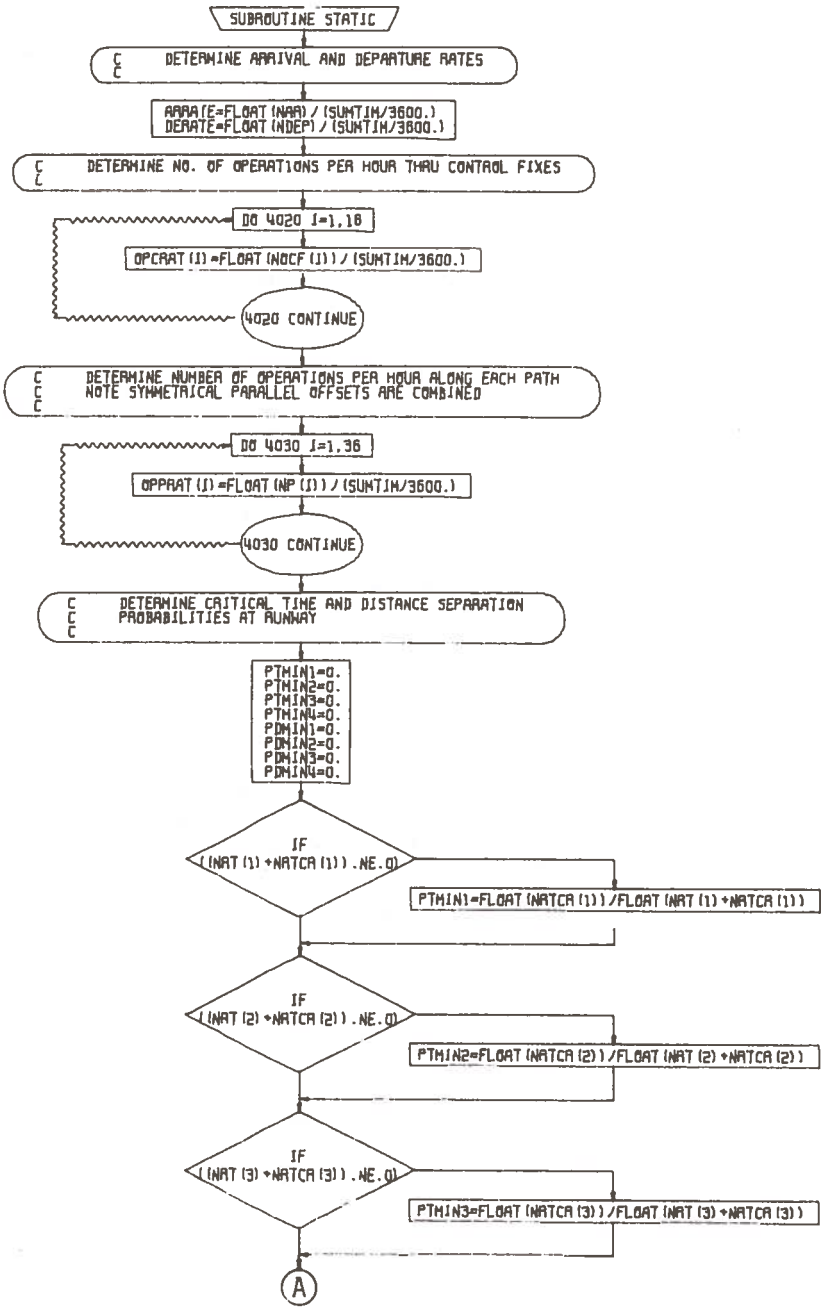
None

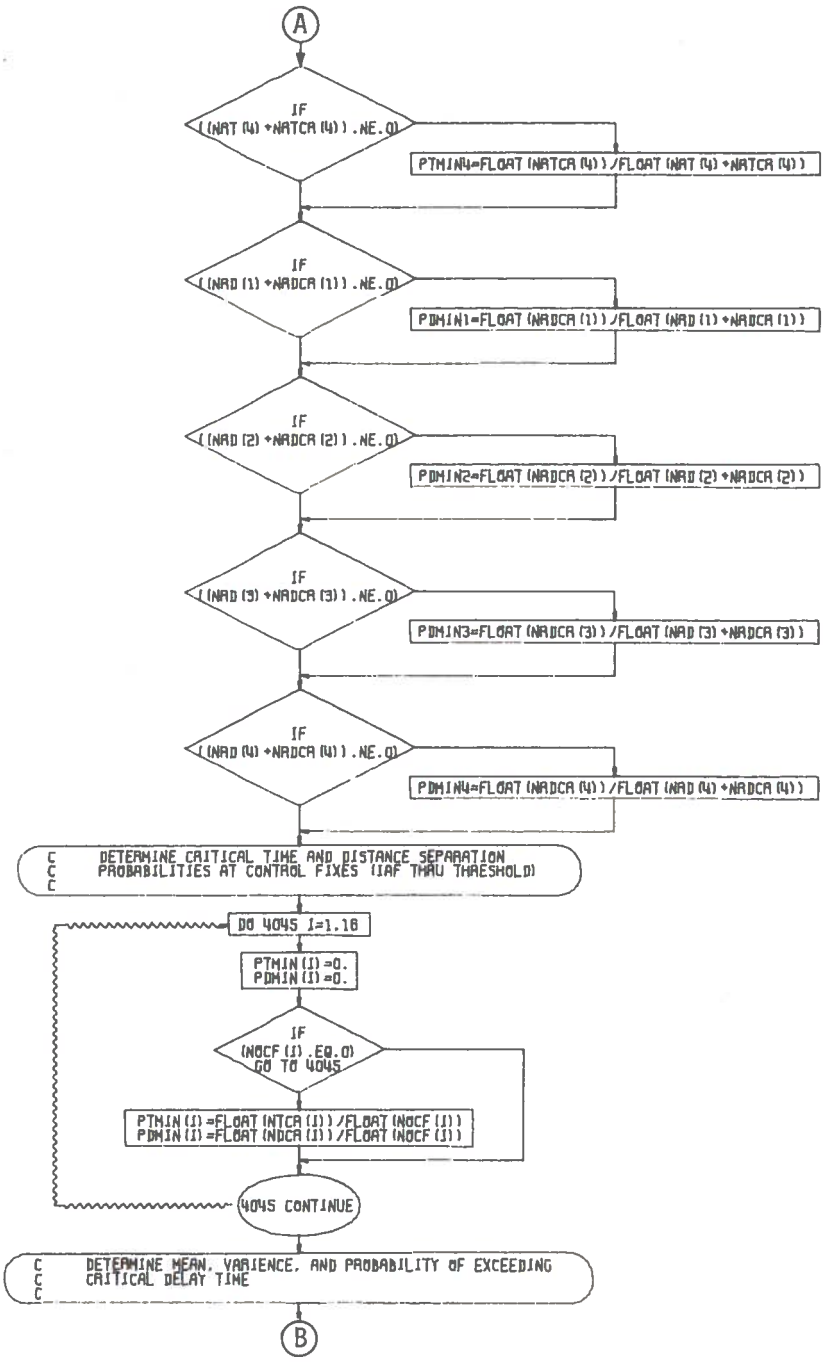
STATIC Called by

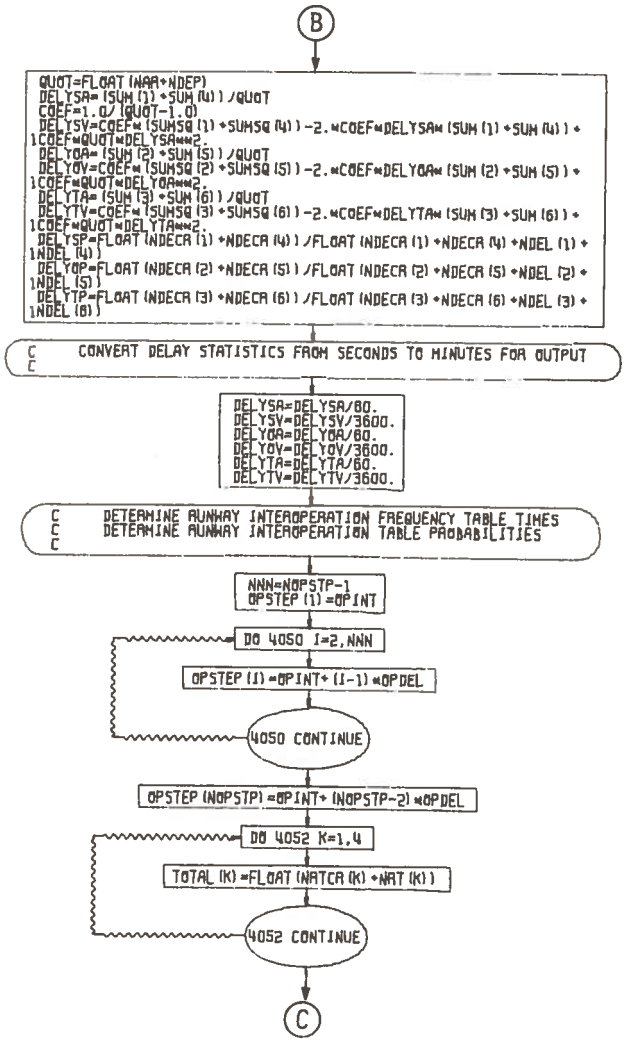
STRAD

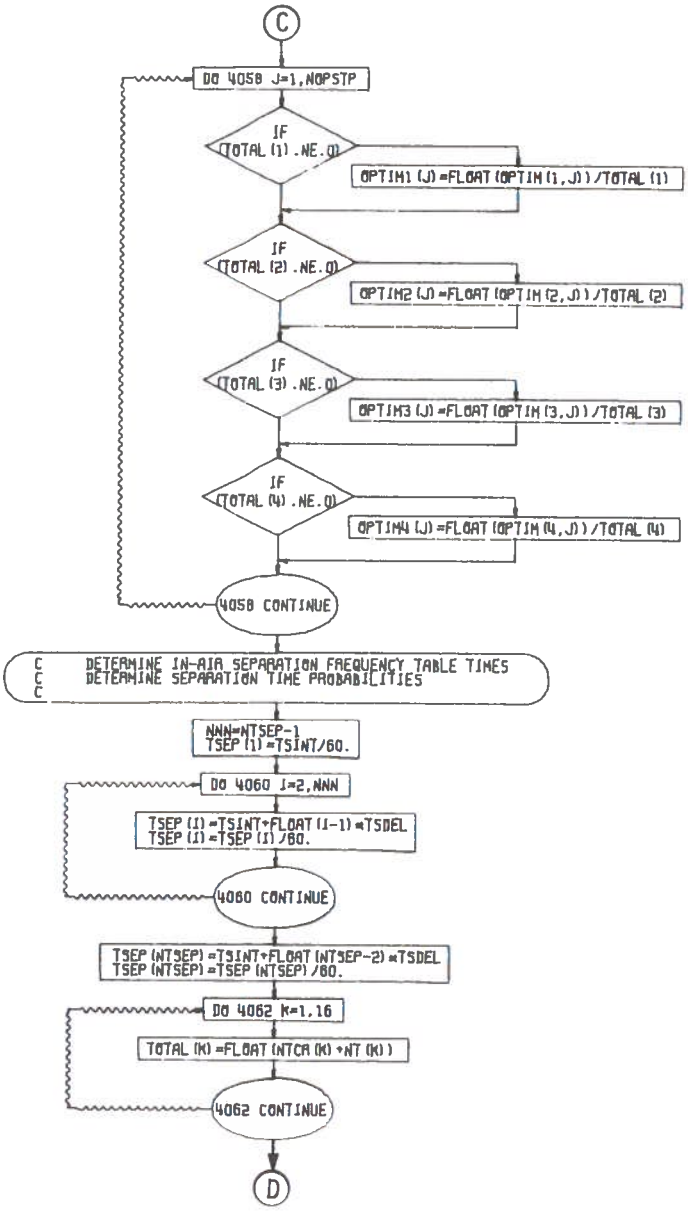
4.2.39.3 STATIC Variables

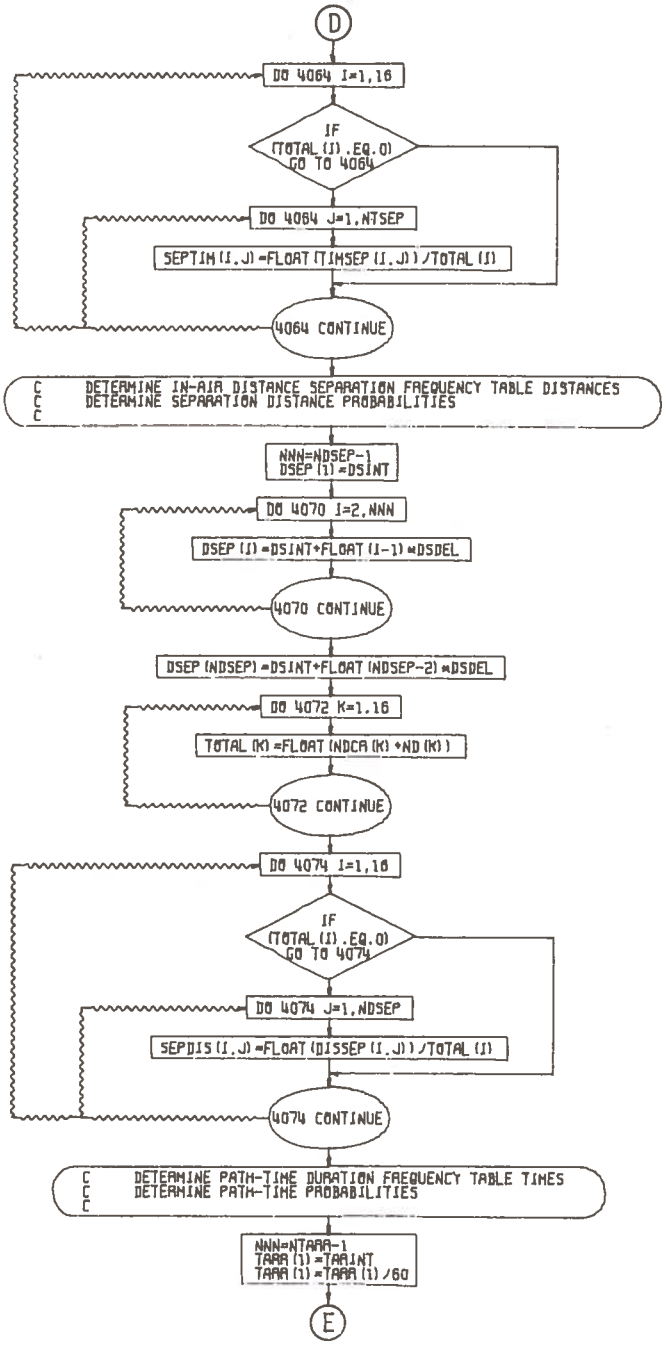
ARRNO	Number of arrivals. Same as NAR except ARRNO is real and NAR is an integer.
QUOT	NAR + NDEP (QUOT is real)

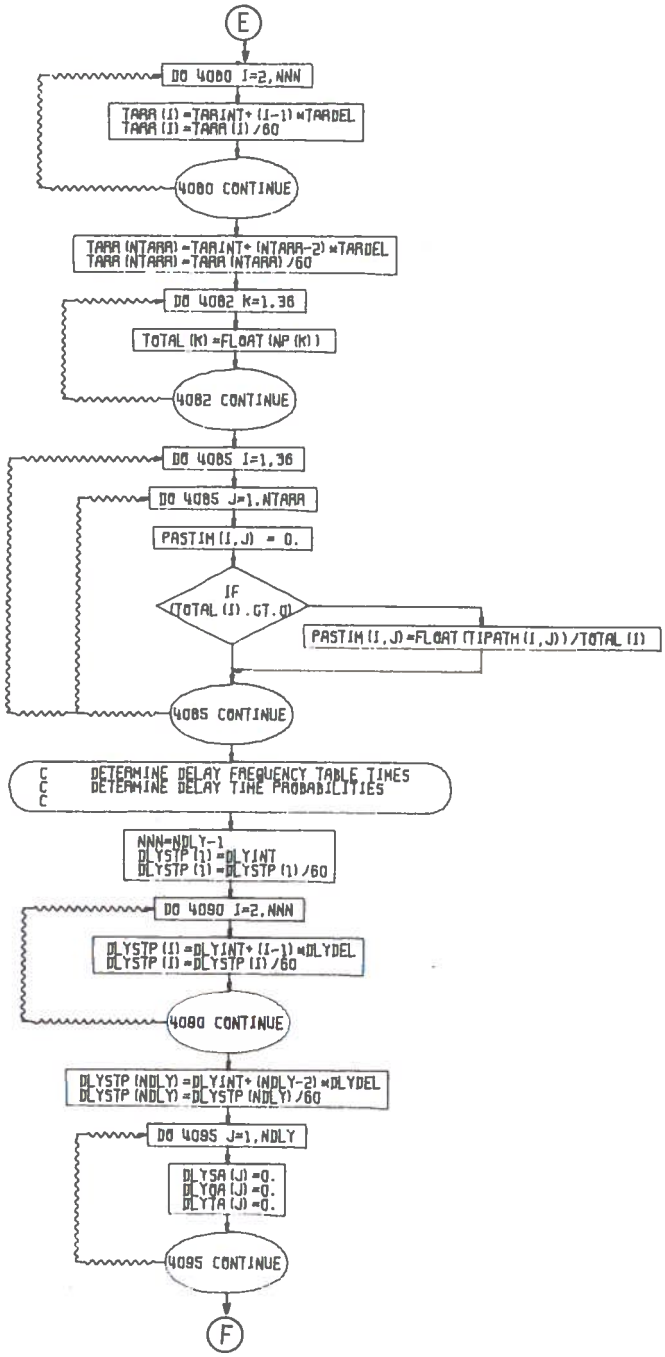

SUBROUTINE STATIC
C DETERMINE ARRIVAL AND DEPARTURE RATES
C
ARRATE=FLOAT(NAA) / (SUMTIM/3600.)
DERATE=FLOAT(NDEP) / (SUMTIM/3600.)
C DETERMINE NO. OF OPERATIONS PER HOUR THRU CONTROL FIXES
C
DO 4020 I=1,18
OPCRAT(I)=FLOAT(NOCF(I)) / (SUMTIM/3600.)
4020 CONTINUE
C DETERMINE NUMBER OF OPERATIONS PER HOUR ALONG EACH PATH
C NOTE SYMMETRICAL PARALLEL OFFSETS ARE COMBINED
C
DO 4030 J=1,36
OPPRAT(J)=FLOAT(NP(J)) / (SUMTIM/3600.)
4030 CONTINUE
C DETERMINE CRITICAL TIME AND DISTANCE SEPARATION
C PROBABILITIES AT RUNWAY
C
PTMIN1=0.
PTMIN2=0.
PTMIN3=0.
PTMIN4=0.
PDMIN1=0.
PDMIN2=0.
PDMIN3=0.
PDMIN4=0.
IF (NAT(1)+NATCA(1)) .NE. 0
PTMIN1=FLOAT(NATCA(1)) /FLOAT(NAT(1)+NATCA(1))
IF (NAT(2)+NATCA(2)) .NE. 0
PTMIN2=FLOAT(NATCA(2)) /FLOAT(NAT(2)+NATCA(2))
IF (NAT(3)+NATCA(3)) .NE. 0
PTMIN3=FLOAT(NATCA(3)) /FLOAT(NAT(3)+NATCA(3))
A

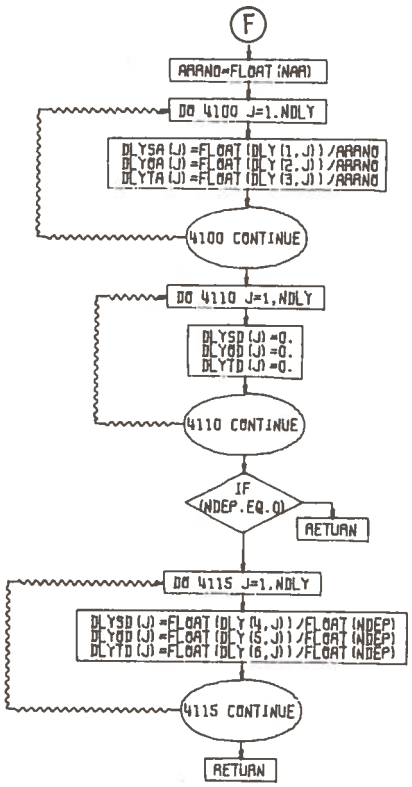

A
IF
((NAT(4)+NATCA(4)).NE.0)
PTMIN4=FLOAT(NATCA(4))/FLOAT(NAT(4)+NATCA(4))
IF
((NAD(1)+NADCA(1)).NE.0)
PDMIN1=FLOAT(NADCA(1))/FLOAT(NAD(1)+NADCA(1))
IF
((NAD(2)+NADCA(2)).NE.0)
PDMIN2=FLOAT(NADCA(2))/FLOAT(NAD(2)+NADCA(2))
IF
((NAD(3)+NADCA(3)).NE.0)
PDMIN3=FLOAT(NADCA(3))/FLOAT(NAD(3)+NADCA(3))
IF
((NAD(4)+NADCA(4)).NE.0)
PDMIN4=FLOAT(NADCA(4))/FLOAT(NAD(4)+NADCA(4))
C DETERMINE CRITICAL TIME AND DISTANCE SEPARATION
C PROBABILITIES AT CONTROL FIXES (IAF THRU THRESHOLD)
C
DO 4045 I=1,16
PTMIN(I)=0.
PDMIN(I)=0.
IF
(NOCF(I).EQ.0)
GO TO 4045
PTMIN(I)=FLOAT(NTCA(I))/FLOAT(NOCF(I))
PDMIN(I)=FLOAT(NDCA(I))/FLOAT(NOCF(I))
4045 CONTINUE
C DETERMINE MEAN, VARIENCE, AND PROBABILITY OF EXCEEDING
C CRITICAL DELAY TIME
C
B


B
QUOT=FLOAT(NAR+NDEP)
DELYSA=(SUM(1)+SUM(4))/QUOT
COEF=1.0/(QUOT-1.0)
DELYSV=COEF*(SUMSQ(1)+SUMSQ(4))-2.*COEF*DELYSA*(SUM(1)+SUM(4))+
1COEF*QUOT*DELYSA**2.
DELYOA=(SUM(2)+SUM(5))/QUOT
DELYOV=COEF*(SUMSQ(2)+SUMSQ(5))-2.*COEF*DELYOA*(SUM(2)+SUM(5))+
1COEF*QUOT*DELYOA**2.
DELYTA=(SUM(3)+SUM(6))/QUOT
DELYTV=COEF*(SUMSQ(3)+SUMSQ(6))-2.*COEF*DELYTA*(SUM(3)+SUM(6))+
1COEF*QUOT*DELYTA**2.
DELYSP=FLOAT(NDECA(1)+NDECA(4))/FLOAT(NDECA(1)+NDECA(4)+NDEL(1)+
1NDEL(4))
DELYOP=FLOAT(NDECA(2)+NDECA(5))/FLOAT(NDECA(2)+NDECA(5)+NDEL(2)+
1NDEL(5))
DELYTP=FLOAT(NDECA(3)+NDECA(6))/FLOAT(NDECA(3)+NDECA(6)+NDEL(3)+
1NDEL(6))
C CONVERT DELAY STATISTICS FROM SECONDS TO MINUTES FOR OUTPUT
C
DELYSA=DELYSA/60.
DELYSV=DELYSV/3600.
DELYOA=DELYOA/60.
DELYOV=DELYOV/3600.
DELYTA=DELYTA/60.
DELYTV=DELYTV/3600.
C DETERMINE RUNWAY INTEROPERATION FREQUENCY TABLE TIMES
C DETERMINE RUNWAY INTEROPERATION TABLE PROBABILITIES
C
NNN=NOPSTP-1
OPSTEP(1)=OPINT
DO 4050 I=2,NNN
OPSTEP(I)=OPINT+(I-1)*OPDEL
4050 CONTINUE
OPSTEP(NOPSTP)=OPINT+(NOPSTP-2)*OPDEL
DO 4052 K=1,4
TOTAL(K)=FLOAT(NATCA(K)+NAT(K))
4052 CONTINUE
C


C
DO 4058 J=1,NOPSTP
IF (TOTAL (1) .NE. 0)
OPTIM1 (J) =FLOAT (OPTIM (1, J)) /TOTAL (1)
IF (TOTAL (2) .NE. 0)
OPTIM2 (J) =FLOAT (OPTIM (2, J)) /TOTAL (2)
IF (TOTAL (3) .NE. 0)
OPTIM3 (J) =FLOAT (OPTIM (3, J)) /TOTAL (3)
IF (TOTAL (4) .NE. 0)
OPTIM4 (J) =FLOAT (OPTIM (4, J)) /TOTAL (4)
4058 CONTINUE
C DETERMINE IN-AIR SEPARATION FREQUENCY TABLE TIMES
C DETERMINE SEPARATION TIME PROBABILITIES
C
NNN=NTSEP-1
TSEP (1) =TSINT/60.
DO 4060 I=2,NNN
TSEP (I) =TSINT+FLOAT (I-1) *TSDEL
TSEP (I) =TSEP (I) /60.
4060 CONTINUE
TSEP (NTSEP) =TSINT+FLOAT (NTSEP-2) *TSDEL
TSEP (NTSEP) =TSEP (NTSEP) /60.
DO 4062 K=1,16
TOTAL (K) =FLOAT (NTCR (K) +NT (K))
4062 CONTINUE
D


D
DO 4064 I=1,16
IF
(TOTAL(I).EQ.0)
GO TO 4064
DO 4064 J=1,NTSEP
SEPTIM(I,J)=FLOAT(TIMSEP(I,J))/TOTAL(I)
4064 CONTINUE
C DETERMINE IN-AIR DISTANCE SEPARATION FREQUENCY TABLE DISTANCES
C DETERMINE SEPARATION DISTANCE PROBABILITIES
C
NNN=NDSEP-1
DSEP(1)=DSINT
DO 4070 I=2,NNN
DSEP(I)=DSINT+FLOAT(I-1)*DSDEL
4070 CONTINUE
DSEP(NDSEP)=DSINT+FLOAT(NDSEP-2)*DSDEL
DO 4072 K=1,16
TOTAL(K)=FLOAT(NDCA(K)+ND(K))
4072 CONTINUE
DO 4074 I=1,16
IF
(TOTAL(I).EQ.0)
GO TO 4074
DO 4074 J=1,NDSEP
SEPDIS(I,J)=FLOAT(DISSEP(I,J))/TOTAL(I)
4074 CONTINUE
C DETERMINE PATH-TIME DURATION FREQUENCY TABLE TIMES
C DETERMINE PATH-TIME PROBABILITIES
C
NNN=NTARR-1
TARR(1)=TARINT
TARR(1)=TARR(1)/60
E


E
DO 4080 I=2,NNN
TARR(I)=TARINT+(I-1)*TARDEL
TARR(I)=TARR(I)/60
4080 CONTINUE
TARR(NTARR)=TARINT+(NTARR-2)*TARDEL
TARR(NTARR)=TARR(NTARR)/60
DO 4082 K=1,36
TOTAL(K)=FLOAT(NP(K))
4082 CONTINUE
DO 4085 I=1,36
DO 4085 J=1,NTARR
PASTIM(I,J) = 0.
IF (TOTAL(I).GT.0)
PASTIM(I,J)=FLOAT(IPATH(I,J))/TOTAL(I)
4085 CONTINUE
C DETERMINE DELAY FREQUENCY TABLE TIMES
C DETERMINE DELAY TIME PROBABILITIES
C
NNN=NDLY-1
DLYSTP(1)=DLYINT
DLYSTP(1)=DLYSTP(1)/60
DO 4090 I=2,NNN
DLYSTP(I)=DLYINT+(I-1)*DLYDEL
DLYSTP(I)=DLYSTP(I)/60
4090 CONTINUE
DLYSTP(NDLY)=DLYINT+(NDLY-2)*DLYDEL
DLYSTP(NDLY)=DLYSTP(NDLY)/60
DO 4095 J=1,NDLY
DLYSA(J)=0.
DLYQA(J)=0.
DLYTA(J)=0.
4095 CONTINUE
F


F
ARRNO=FLOAT(NAA)
DO 4100 J=1,NDLY
DLYSA(J)=FLOAT(DLY(1,J))/ARRNO
DLYQA(J)=FLOAT(DLY(2,J))/ARRNO
DLYTA(J)=FLOAT(DLY(3,J))/ARRNO
4100 CONTINUE
DO 4110 J=1,NDLY
DLYSD(J)=0.
DLYQD(J)=0.
DLYTD(J)=0.
4110 CONTINUE
IF (NDEP.EQ.0)
RETURN
DO 4115 J=1,NDLY
DLYSD(J)=FLOAT(DLY(4,J))/FLOAT(NDEP)
DLYQD(J)=FLOAT(DLY(5,J))/FLOAT(NDEP)
DLYTD(J)=FLOAT(DLY(6,J))/FLOAT(NDEP)
4115 CONTINUE
RETURN

4.2.40 Program STRAD

4.2.40.1 Abstract

Program STRAD (INPUT, OUTPUT, TAPE5 = INPUT, TAPE1) is the controlling logic for the strategic algorithm and the evaluation model.

4.2.40.2 Data Interfaces

STRAD Inputs

Through Read Inputs:

NSAMP	Number of data samples to be simulated in a data set
IPRINT	Print control for data sample intermediate prints. IPRINT = N implies printing of every Nth sample. No printing occurs if IPRINT is zero.

STRAD Outputs

None

STRAD Calls

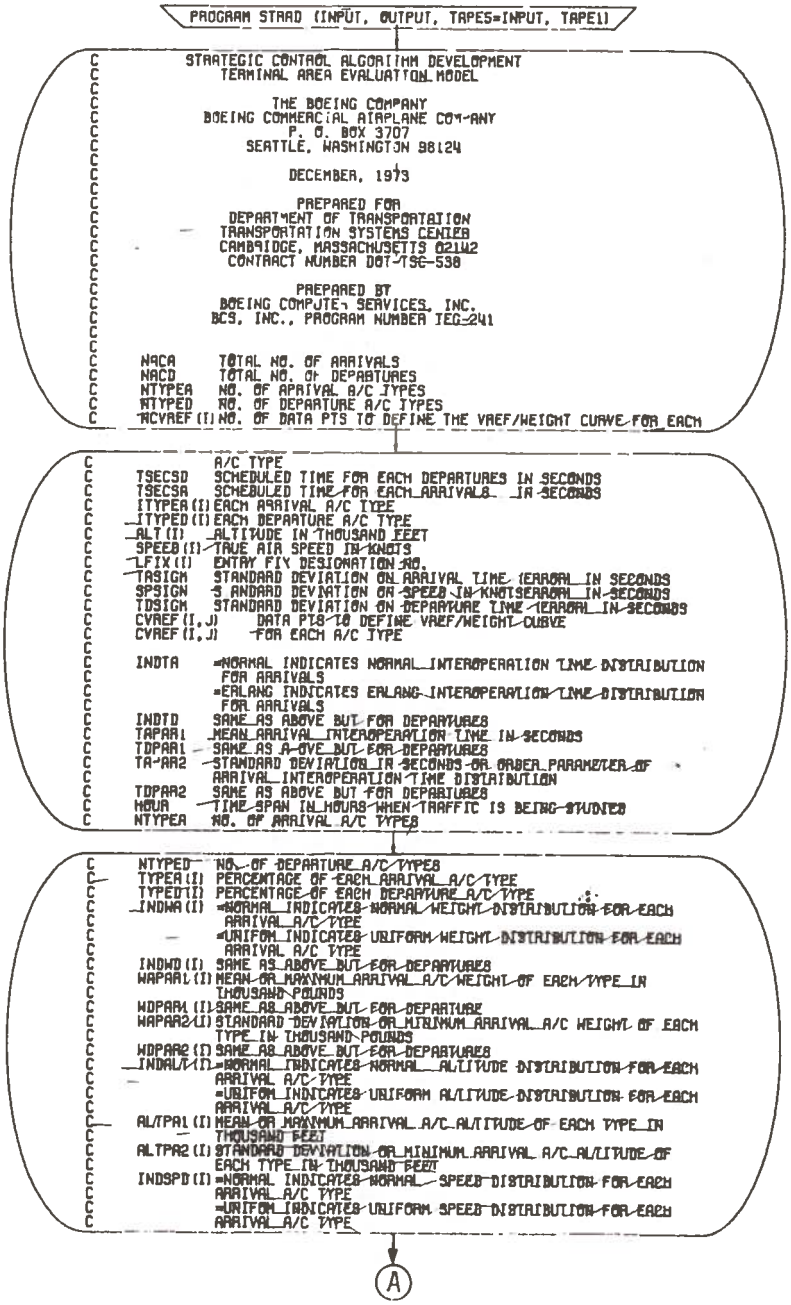
AERO1, DETAIL, FLIGTX, GEOMTY, READIN, RTP, SCHLD1, SCHLD2, SCHLD3, SEQUEN, SETN, STATIC, TRAFIC, WTHTST, WTOUT
All the subroutines called are evaluation or algorithm subroutines.

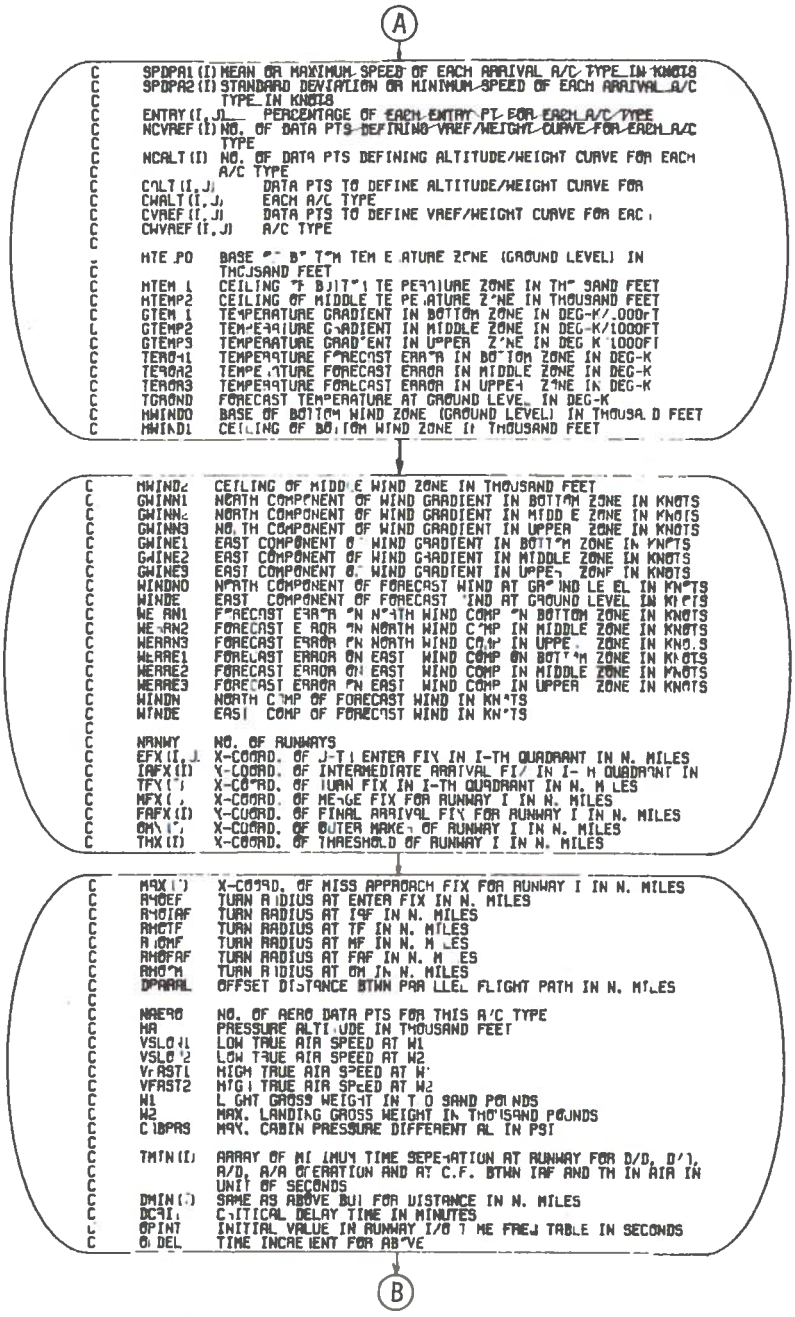
STRAD Called by

None

4.2.40.3 STRAD Variables

NSAMP, IPRINT


PROGRAM STRAD (INPUT, OUTPUT, TAPE5=INPUT, TAPE1)
C STRATEGIC CONTROL ALGORITHM DEVELOPMENT
C TERMINAL AREA EVALUATION MODEL
C
C THE BOEING COMPANY
C BOEING COMMERCIAL AIRPLANE COMPANY
C P. O. BOX 3707
C SEATTLE, WASHINGTON 98124
C
C DECEMBER, 1973
C
C PREPARED FOR
C DEPARTMENT OF TRANSPORTATION
C TRANSPORTATION SYSTEMS CENTER
C CAMBRIDGE, MASSACHUSETTS 02142
C CONTRACT NUMBER DOT-TSC-538
C
C PREPARED BY
C BOEING COMPUTER SERVICES, INC.
C BCS, INC., PROGRAM NUMBER IEG-241
C
C NACA TOTAL NO. OF ARRIVALS
C NACD TOTAL NO. OF DEPARTURES
C NTYPEA NO. OF ARRIVAL A/C TYPES
C NTYPED NO. OF DEPARTURE A/C TYPES
C NCVREF(I) NO. OF DATA PTS TO DEFINE THE VREF/WEIGHT CURVE FOR EACH
C A/C TYPE
C TSECSD SCHEDULED TIME FOR EACH DEPARTURES IN SECONDS
C TSECSA SCHEDULED TIME FOR EACH ARRIVALS IN SECONDS
C ITYPEA(I) EACH ARRIVAL A/C TYPE
C ITYPED(I) EACH DEPARTURE A/C TYPE
C ALT(I) ALTITUDE IN THOUSAND FEET
C SPEED(I) TRUE AIR SPEED IN KNOTS
C LFIX(I) ENTRY FIX DESIGNATION NO.
C TASIGM STANDARD DEVIATION ON ARRIVAL TIME (ERROR) IN SECONDS
C SPSIGM STANDARD DEVIATION ON SPEED IN KNOTS(ERROR) IN SECONDS
C TDSIGM STANDARD DEVIATION ON DEPARTURE TIME (ERROR) IN SECONDS
C CVREF(I,J) DATA PTS TO DEFINE VREF/WEIGHT CURVE
C CVREF(I,J) FOR EACH A/C TYPE
C
C INDTA =NORMAL INDICATES NORMAL INTEROPERATION TIME DISTRIBUTION
C FOR ARRIVALS
C =ERLANG INDICATES ERLANG INTEROPERATION TIME DISTRIBUTION
C FOR ARRIVALS
C INDTD SAME AS ABOVE BUT FOR DEPARTURES
C TAPAR1 MEAN ARRIVAL INTEROPERATION TIME IN SECONDS
C TDPAR1 SAME AS ABOVE BUT FOR DEPARTURES
C TA-AR2 STANDARD DEVIATION IN SECONDS OR ORDER PARAMETER OF
C ARRIVAL INTEROPERATION TIME DISTRIBUTION
C TDPAR2 SAME AS ABOVE BUT FOR DEPARTURES
C HOUR TIME SPAN IN HOURS WHEN TRAFFIC IS BEING STUDIED
C NTYPEA NO. OF ARRIVAL A/C TYPES
C NTYPED NO. OF DEPARTURE A/C TYPES
C TYPEA(I) PERCENTAGE OF EACH ARRIVAL A/C TYPE
C TYPED(I) PERCENTAGE OF EACH DEPARTURE A/C TYPE
C INDWA(I) =NORMAL INDICATES NORMAL WEIGHT DISTRIBUTION FOR EACH
C ARRIVAL A/C TYPE
C =UNIFOM INDICATES UNIFORM WEIGHT DISTRIBUTION FOR EACH
C ARRIVAL A/C TYPE
C INDWD(I) SAME AS ABOVE BUT FOR DEPARTURES
C WAPAR1(I) MEAN OR MAXIMUM ARRIVAL A/C WEIGHT OF EACH TYPE IN
C THOUSAND POUNDS
C WDPAR1(I) SAME AS ABOVE BUT FOR DEPARTURE
C WAPAR2(I) STANDARD DEVIATION OR MINIMUM ARRIVAL A/C WEIGHT OF EACH
C TYPE IN THOUSAND POUNDS
C WDPAR2(I) SAME AS ABOVE BUT FOR DEPARTURES
C INDALT(I) =NORMAL INDICATES NORMAL ALTITUDE DISTRIBUTION FOR EACH
C ARRIVAL A/C TYPE
C =UNIFOM INDICATES UNIFORM ALTITUDE DISTRIBUTION FOR EACH
C ARRIVAL A/C TYPE
C ALTPA1(I) MEAN OR MAXIMUM ARRIVAL A/C ALTITUDE OF EACH TYPE IN
C THOUSAND FEET
C ALTPA2(I) STANDARD DEVIATION OR MINIMUM ARRIVAL A/C ALTITUDE OF
C EACH TYPE IN THOUSAND FEET
C INDSPD(I) =NORMAL INDICATES NORMAL SPEED DISTRIBUTION FOR EACH
C ARRIVAL A/C TYPE
C =UNIFOM INDICATES UNIFORM SPEED DISTRIBUTION FOR EACH
C ARRIVAL A/C TYPE
A


A
SPDPA1(I) MEAN OR MAXIMUM SPEED OF EACH ARRIVAL A/C TYPE IN KNOTS
SPDPA2(I) STANDARD DEVIATION OR MINIMUM SPEED OF EACH ARRIVAL A/C TYPE IN KNOTS
ENTRY(I,J) PERCENTAGE OF EACH ENTRY PT FOR EACH A/C TYPE
NCVREF(I) NO. OF DATA PTS DEFINING VREF/WEIGHT CURVE FOR EACH A/C TYPE
NCALT(I) NO. OF DATA PTS DEFINING ALTITUDE/WEIGHT CURVE FOR EACH A/C TYPE
CALT(I,J) DATA PTS TO DEFINE ALTITUDE/WEIGHT CURVE FOR
CWALT(I,J) EACH A/C TYPE
CVREF(I,J) DATA PTS TO DEFINE VREF/WEIGHT CURVE FOR EAC
CWVREF(I,J) A/C TYPE
HTEMP0 BASE OF BOTTOM TEMPERATURE ZONE (GROUND LEVEL) IN THOUSAND FEET
HTEMP1 CEILING OF BOTTOM TEMPERATURE ZONE IN THOUSAND FEET
HTEMP2 CEILING OF MIDDLE TEMPERATURE ZONE IN THOUSAND FEET
GTEMP1 TEMPERATURE GRADIENT IN BOTTOM ZONE IN DEG-K/1000FT
GTEMP2 TEMPERATURE GRADIENT IN MIDDLE ZONE IN DEG-K/1000FT
GTEMP3 TEMPERATURE GRADIENT IN UPPER ZONE IN DEG K/1000FT
TERR01 TEMPERATURE FORECAST ERROR IN BOTTOM ZONE IN DEG-K
TERR02 TEMPERATURE FORECAST ERROR IN MIDDLE ZONE IN DEG-K
TERR03 TEMPERATURE FORECAST ERROR IN UPPER ZONE IN DEG-K
TGROND FORECAST TEMPERATURE AT GROUND LEVEL IN DEG-K
HWINDO BASE OF BOTTOM WIND ZONE (GROUND LEVEL) IN THOUSAND FEET
HWIND1 CEILING OF BOTTOM WIND ZONE IN THOUSAND FEET
HWIND2 CEILING OF MIDDLE WIND ZONE IN THOUSAND FEET
GWINN1 NORTH COMPONENT OF WIND GRADIENT IN BOTTOM ZONE IN KNOTS
GWINN2 NORTH COMPONENT OF WIND GRADIENT IN MIDDLE ZONE IN KNOTS
GWINN3 NORTH COMPONENT OF WIND GRADIENT IN UPPER ZONE IN KNOTS
GWINE1 EAST COMPONENT OF WIND GRADIENT IN BOTTOM ZONE IN KNOTS
GWINE2 EAST COMPONENT OF WIND GRADIENT IN MIDDLE ZONE IN KNOTS
GWINE3 EAST COMPONENT OF WIND GRADIENT IN UPPER ZONE IN KNOTS
WINDNO NORTH COMPONENT OF FORECAST WIND AT GROUND LEVEL IN KNOTS
WINDE EAST COMPONENT OF FORECAST WIND AT GROUND LEVEL IN KNOTS
WERRN1 FORECAST ERROR ON NORTH WIND COMP ON BOTTOM ZONE IN KNOTS
WERRN2 FORECAST ERROR ON NORTH WIND COMP IN MIDDLE ZONE IN KNOTS
WERRN3 FORECAST ERROR ON NORTH WIND COMP IN UPPER ZONE IN KNOTS
WERRE1 FORECAST ERROR ON EAST WIND COMP ON BOTTOM ZONE IN KNOTS
WERRE2 FORECAST ERROR ON EAST WIND COMP IN MIDDLE ZONE IN KNOTS
WERRE3 FORECAST ERROR ON EAST WIND COMP IN UPPER ZONE IN KNOTS
WINDN NORTH COMP OF FORECAST WIND IN KNOTS
WINDE EAST COMP OF FORECAST WIND IN KNOTS
NRNWY NO. OF RUNWAYS
EFX(I,J) X-COORD. OF J-TH ENTER FIX IN I-TH QUADRANT IN N. MILES
IAFX(I) X-COORD. OF INTERMEDIATE ARRIVAL FIX IN I-TH QUADRANT IN
TFX(I) X-COORD. OF TURN FIX IN I-TH QUADRANT IN N. MILES
MFX(I) X-COORD. OF MERGE FIX FOR RUNWAY I IN N. MILES
FAFX(I) X-COORD. OF FINAL ARRIVAL FIX FOR RUNWAY I IN N. MILES
OMX(I) X-COORD. OF OUTER MARKER OF RUNWAY I IN N. MILES
THX(I) X-COORD. OF THRESHOLD OF RUNWAY I IN N. MILES
MAX(I) X-COORD. OF MISS APPROACH FIX FOR RUNWAY I IN N. MILES
RADEF TURN RADIUS AT ENTER FIX IN N. MILES
RADIAF TURN RADIUS AT IAF IN N. MILES
RADTF TURN RADIUS AT TF IN N. MILES
RADMF TURN RADIUS AT MF IN N. MILES
RADFAF TURN RADIUS AT FAF IN N. MILES
RADOM TURN RADIUS AT OM IN N. MILES
DPARAL OFFSET DISTANCE BTWN PARALLEL FLIGHT PATH IN N. MILES
NAERO NO. OF AERO DATA PTS FOR THIS A/C TYPE
HA PRESSURE ALTITUDE IN THOUSAND FEET
VSLO1 LOW TRUE AIR SPEED AT W1
VSLO2 LOW TRUE AIR SPEED AT W2
VFAST1 HIGH TRUE AIR SPEED AT W1
VFAST2 HIGH TRUE AIR SPEED AT W2
W1 L GNT GROSS WEIGHT IN THOUSAND POUNDS
W2 MAX. LANDING GROSS WEIGHT IN THOUSAND POUNDS
CABPRS MAX. CABIN PRESSURE DIFFERENTIAL IN PSI
TMIN(I) ARRAY OF MINIMUM TIME SEPERATION AT RUNWAY FOR D/D, D/I, A/D, A/A OPERATION AND AT C.F. BTWN IAF AND TM IN AIR IN UNIT OF SECONDS
DMIN(I) SAME AS ABOVE BUT FOR DISTANCE IN N. MILES
DCRI CRITICAL DELAY TIME IN MINUTES
OPINT INITIAL VALUE IN RUNWAY I/O TIME FREQ TABLE IN SECONDS
OPDEL TIME INCREMENT FOR ABOVE
B

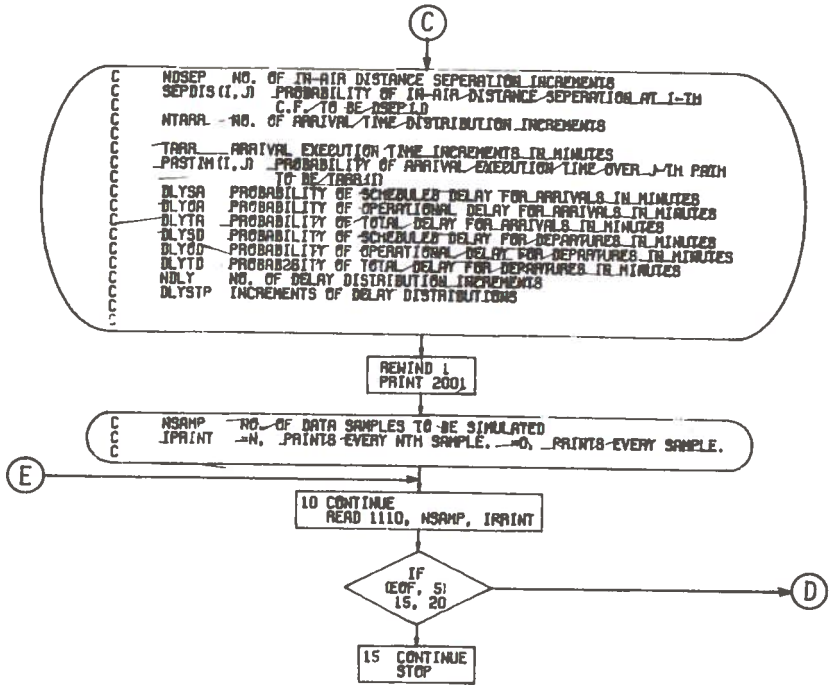
(B)

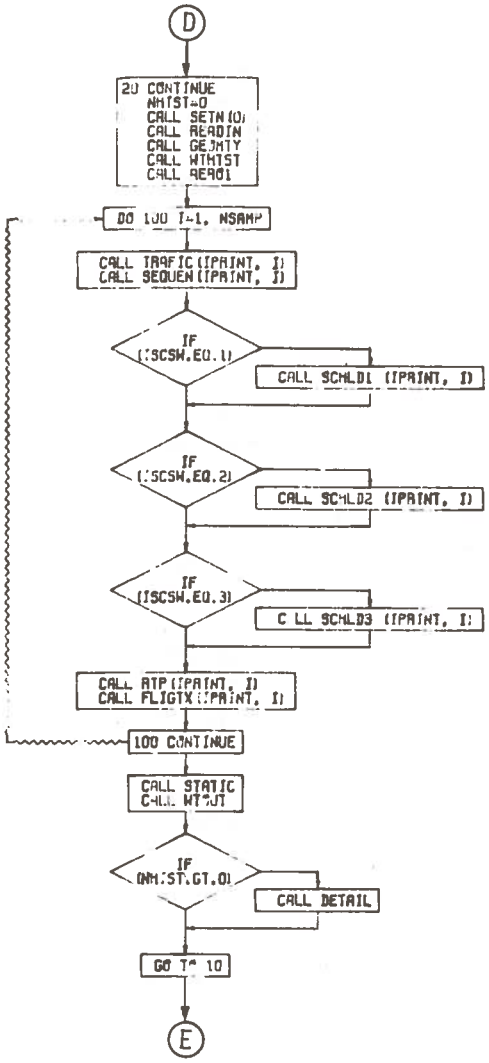
```
C     NOPSTP   NO. OF ENTRIES IN RUNWAY I/O TIME FREQ TABLE
C     TSINT    INITIAL VALUE OF IN-AIR TIME SEPERATION FREQ TABLE IN
C              MINUTES
C     TSDEL    INCREMENT (STEP SIZE) FOR ABOVE
C     NTSEP    NO. OF ENTRIES IN IN-AIR TIME SEPERATION FREQ TABLE
C     DSINT    INITIAL VALUE OF IN-AIR DISTANCE SEPERATION FREQ TABLE IN
C              N. MILES
C     DSDEL    INCREMENT (STEP SIZE) FOR ABOVE
C     NDSE     NO. OF ENTRIES IN IN-AIR DISTANCE SEPERATION FREQ TABLE
C     TARINT   INITIAL VALUE IN ARRIVAL EXECUTION TIME FREQ TABLE IN
C              MINUTES
C     TADEL    INCREMENT (STEP SIZE) FOR ABOVE
C     NTAR     NO. OF ENTRIES IN ARRIVAL EXECUTION TIME FREQ TABLE
C     DLYINT   INITIAL VALUE IN DELAY FREQ TABLE IN MINUTES
C     DLYDEL   INCREMENT (STEP SIZE) FOR ABOVE
C     NDLY     NO. OF ENTRIES IN DELAY FREQ TABLE
C
C     IRO      CONSTRAINT SWITCH (=0 FOR RUNWAY OCCUPANCY AND =1 FOR
C              ADDITIONAL IN-AIR CONSTRAINTS)
C     IATC     CONSTRAINT SWITCH (=0 FOR TIME AND =1 FOR DISTANCE
C              SEPARATION)
C     DSEP     NOMINAL DISTANCE SEPARATION FOR SCHEDULING IN N.MILES
C     TSEP     NOMINAL TIME SEPARATION FOR SCHEDULING IN SECONDS
C     TARO(I)  NOMINAL RUNWAY TIME FOR ITH ARRIVAL TYPE IN SECONDS
C     DROT(I)  NOMINAL RUNWAY TIME FOR ITH DEPARTURE TYPE IN SECONDS
C     DRTPS    MINIMUM IN-AIR DISTANCE SEPARATION USED IN ATP GENERATION
```

```
C     TRTPS    MINIMUM IN-AIR TIME SEP. USED IN ATP GENERATION IN SECOND
C     HATPS    MINIMUM IN-AIR ALTITUDE SEP. USED IN APT GENERATION(FTX-O
C
C     MURO     MEAN VALUE OF THE RADIAL COMPONENT OF THE NAVIGATION
C              ERROR DISTRIBUTION IN NAUTICAL MILES
C     SIGRO    STANDARD DEVIATION OF THE NAV ERROR DISTRIBUTION RADIAL
C              COMPONENT (N. MILES)
C     MUT      MEAN VALUE OF THE CONTROL ERROR DISTRIBUTION IN
C              SECONDS
C     SIGT     STANDARD DEVIATION OF CONTROL ERROR TIME DISTRIBUTION
C              IN SECONDS
C     ARRATE   NO. OF ARRIVALS PER HOUR
C     DERATE   NO. OF DEPARTURES PER HOUR
C     OPCRAT   NO. OF OPERATIONS PER HOUR THRU EACH CONTROL FIX
C     OPRAT    NO. OF OPERATIONS PER HOUR THRU EACH PATH SEGMENT
C     PTMIN1   PROBABILITY OF TIME SEPERATION BEING LESS THAN TMIN(1) FOR
C              D/D OPERATION
C     PTMIN2   PROBABILITY OF TIME SEPERATION BEING LESS THAN TMIN(2) FOR
C              D/A OPERATION
C     PTMIN3   PROBABILITY OF TIME SEPERATION BEING LESS THAN TMIN(3) FOR
C              A/D OPERATION
C     PTMIN4   PROBABILITY OF TIME SEPERATION BEING LESS THAN TMIN(4) FOR
C              A/A OPERATION
C     PDMIN1   PROBABILITY OF DISTANCE SEPERATION BEING LESS THAN DMIN(1)
C              FOR D/D OPERATION
C     PDMIN2   PROBABILITY OF DISTANCE SEPERATION BEING LESS THAN DMIN(2)
```

```
C              FOR D/A OPERATION
C     PDMIN3   PROBABILITY OF DISTANCE SEPERATION BEING LESS THAN DMIN(3)
C              FOR A/D OPERATION
C     PDMIN4   PROBABILITY OF DISTANCE SEPERATION BEING LESS THAN DMIN(4)
C              FOR A/A OPERATION
C     DELYSA   MEAN VALUE OF SCHEDULED DELAY IN MINUTES
C     DELYSV   VARIANCE OF SCHEDULED DELAY IN MINUTES
C     DELYSP   PROBABILITY OF SCHEDULED DELAY EXCEEDING D(CRITICAL)=S
C     DELYOA   MEAN VALUE OF OPERATIONAL DELAY IN MINUTES
C     DELYOV   VARIANCE OF OPERATIONAL DELAY IN MINUTES
C     DELYOP   PROBABILITY OF OPERATIONAL DELAY EXCEEDING D(CRITICAL)=O
C     DELYTA   MEAN VALUE OF TOTAL DELAY IN MINUTES
C     DELYTV   VARIANCE OF TOTAL DELAY IN MINUTES
C     DELYTP   PROBABILITY OF TOTAL DELAY EXCEEDING D(CRITICAL)=T
C     NPLANE   TOTAL NO. OF AIRPLANE EXECUTED THRU THE SYSTEM
C     OPSTEP   STEPS OF INTEROPERATION TIME IN MINUTES
C     NOPSTP   NO. OF INTEROPERATION TIME INCREMENTS
C     OPTIM1   INTEROPERATION TIME OF D/D OPERATION IN SECONDS
C     OPTIM2   INTEROPERATION TIME OF D/A OPERATION IN SECONDS
C     OPTIM3   INTEROPERATION TIME OF A/D OPERATION IN SECONDS
C     OPTIM4   INTEROPERATION TIME OF A/A OPERATION IN SECONDS
C     TSEP        IN-AIR TIME SEPERATION INCREMENTS IN MINUTES
C     NTSEP    NO. OF IN-AIR TIME SEPERATION INCREMENTS
C     SEPTIM(I,J)  PROBABILITY OF IN-AIR TIME SEPERATION AT I-TH C.F. TO
C                  BE TSEP(J)
C     DSEP        IN-AIR DISTANCE SEPERATION INCREMENTS IN N.M.
```

(C)


C
C NOSEP NO. OF IN-AIR DISTANCE SEPERATION INCREMENTS
C SEPDIS(I,J) PROBABILITY OF IN-AIR DISTANCE SEPERATION AT I-TH
C C.F. TO BE NSEP1,0
C NTARR NO. OF ARRIVAL TIME DISTRIBUTION INCREMENTS
C
C TARR ARRIVAL EXECUTION TIME INCREMENTS IN MINUTES
C PASTIM(I,J) PROBABILITY OF ARRIVAL EXECUTION TIME OVER J-TH PATH
C TO BE 1,NRA1,0
C DLYSA PROBABILITY OF SCHEDULED DELAY FOR ARRIVALS IN MINUTES
C DLYOA PROBABILITY OF OPERATIONAL DELAY FOR ARRIVALS IN MINUTES
C DLYTA PROBABILITY OF TOTAL DELAY FOR ARRIVALS IN MINUTES
C DLYSD PROBABILITY OF SCHEDULED DELAY FOR DEPARTURES IN MINUTES
C DLYOD PROBABILITY OF OPERATIONAL DELAY FOR DEPARTURES IN MINUTES
C DLYTD PROBABILITY OF TOTAL DELAY FOR DEPARTURES IN MINUTES
C NDLY NO. OF DELAY DISTRIBUTION INCREMENTS
C DLYSTP INCREMENTS OF DELAY DISTRIBUTIONS
C
REWIND 1
PRINT 2001
C NSAMP NO. OF DATA SAMPLES TO BE SIMULATED
C IPRINT =N, PRINTS EVERY NTH SAMPLE. =0, PRINTS EVERY SAMPLE.
C
10 CONTINUE
READ 1110, NSAMP, IPRINT
IF
(EOF, 5)
15, 20
15 CONTINUE
STOP
C
E
D


D
20 CONTINUE
NHIST=0
CALL SETN(0)
CALL READIN
CALL GEOMTY
CALL WTHIST
CALL AERO1
DO 100 I=1, NSAMP
CALL TRAFIC(IPRINT, I)
CALL SEQUEN(IPRINT, I)
IF (ISCSW.EQ.1)
CALL SCHLD1 (IPRINT, I)
IF (ISCSW.EQ.2)
CALL SCHLD2 (IPRINT, I)
IF (ISCSW.EQ.3)
CALL SCHLD3 (IPRINT, I)
CALL ATP(IPRINT, I)
CALL FLIGTX(IPRINT, I)
100 CONTINUE
CALL STATIC
CALL WTOUT
IF (NHIST.GT.0)
CALL DETAIL
GO TO 10
E

4.2.41 Subroutine STUFF1

4.2.41.1 Abstract

Subroutine STUFF1 (HNEW, VGND) sets up COMMON/RTPOUT/ with the variables to allow an RTP integration of subroutine TFIND. STUFF1 is used when an airplane arrives above the critical altitude and the airplane will make either a constant calibrated airspeed letdown or a constant Mach/constant calibrated letdown, with the break point at other than the critical altitude.

4.2.41.2 Data Interfaces

STUFF1 Inputs

Through Calling Sequence:

HNEW	Will determine the altitude at which the transition from Mach to calibrated airspeed is made. If HNEW is set equal to the arrival altitude, a calibrated airspeed letdown will result.
VGND	Groundspeed desired at the end of the level segment at the entry altitude. VGND should be set to VSLOW (JMAX) for normal algorithm operation.

Through Common Statements:

/AEROOT/	HCRIT, SPKNOT
/GEOMOT/	ANGEF(IQUAD, INFIX), ANGIAF(IQUAD)
/RTPO/	DDP, DLF, HIAF, INFIX, IQUAD, IS, TIAF
/TEMPER/	TEMP
/TRAFOT/	ALT(IS)
/WIND/	WINANG, WINMAG

STUFF1 Outputs

Through Common Statements:

/RTPOUT/	DRTP(-,IS), HRTP(-,IS), IDRTP(-,IS), TRTP(-,IS), VRTP(-,IS)

STUFF1 Calls

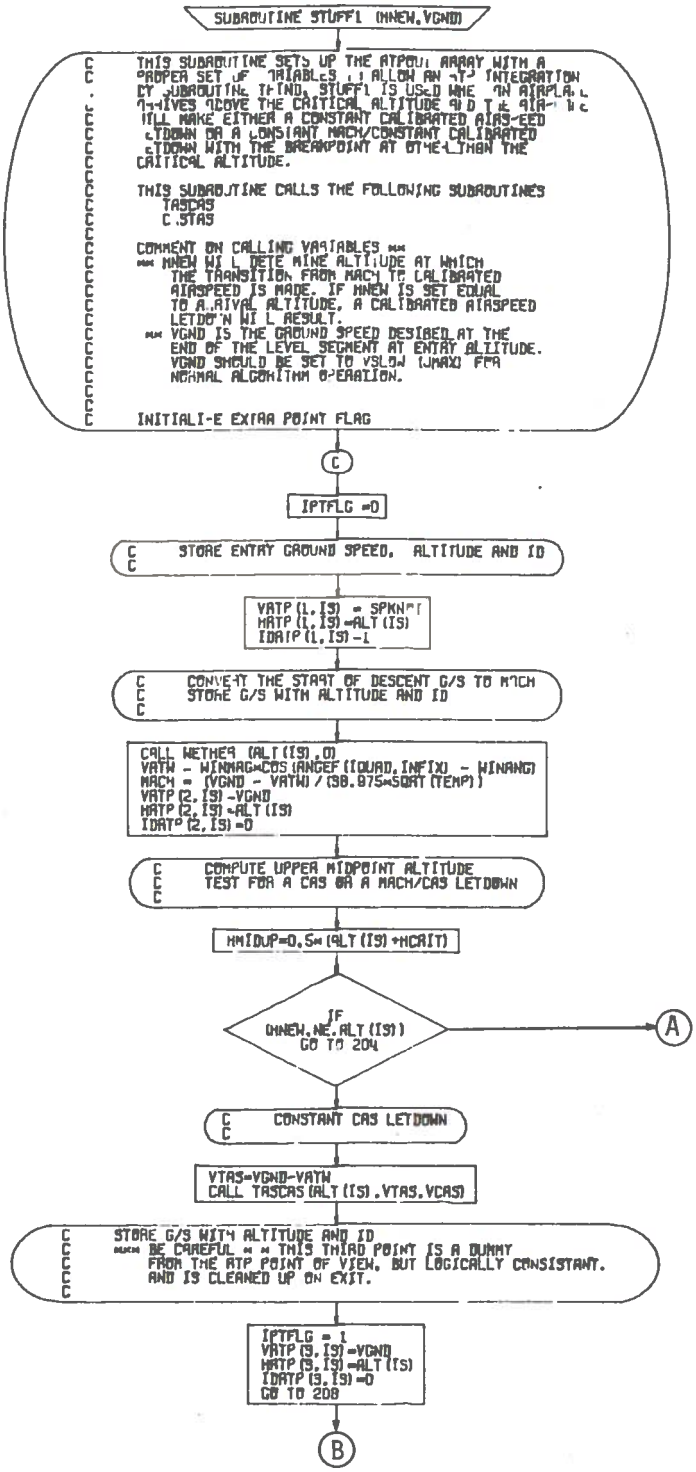
CASTAS	Converts calibrated airspeed to true airspeed at a given pressure altitude
SQRT	FORTRAN library function
TASCAS	Converts true airspeed to calibrated airspeed at a given pressure altitude
WETHER	Finds the forecast along-track wind at a given altitude

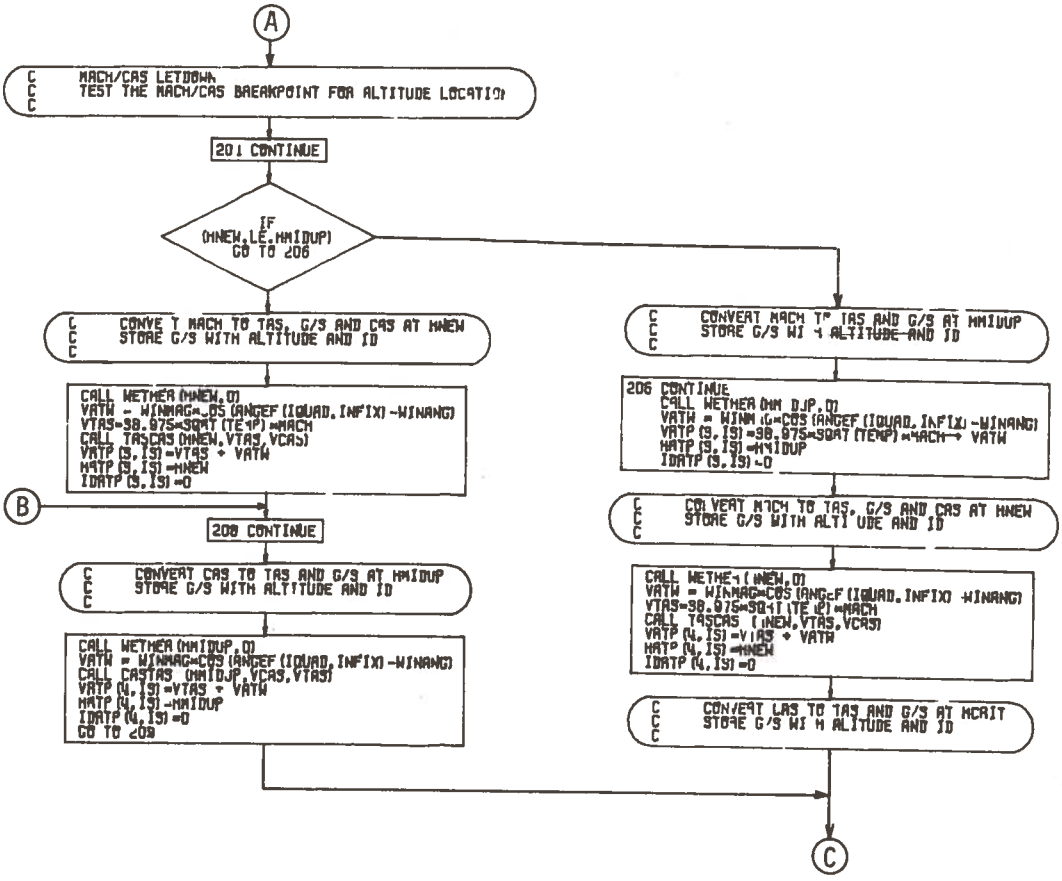
STUFF1 Called by

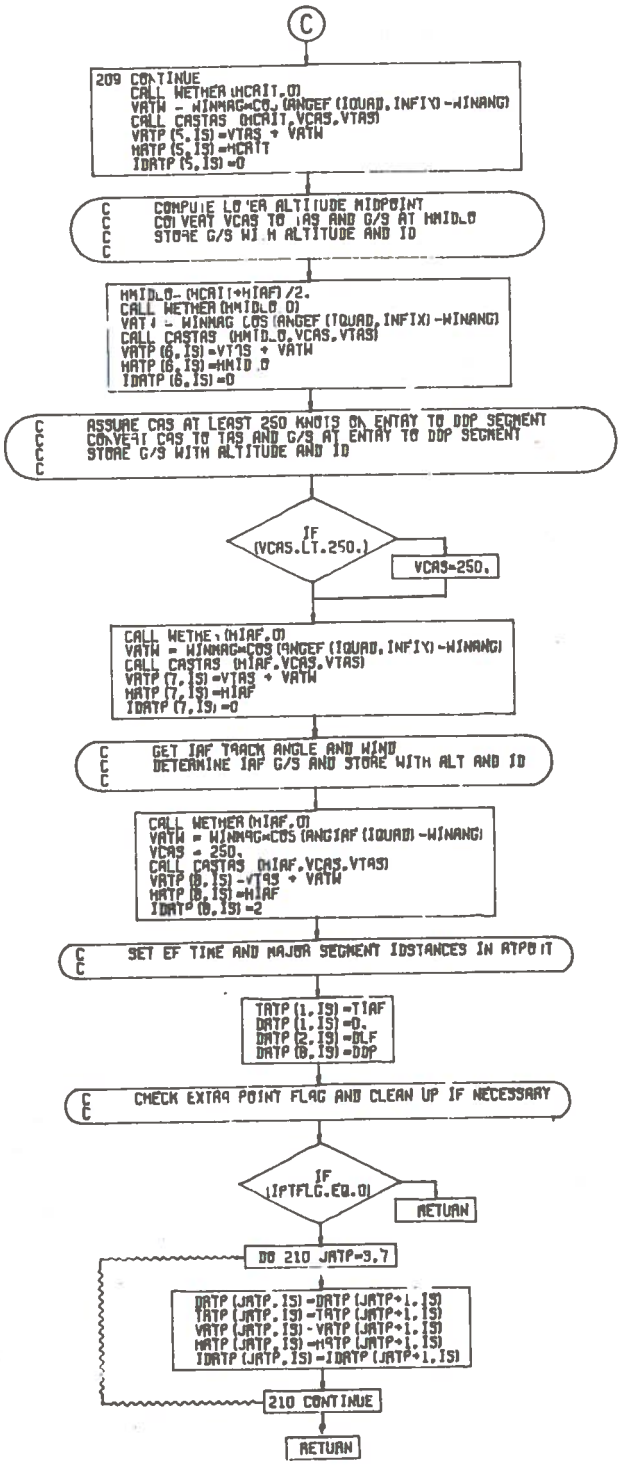
RTP, FASLOW

4.2.41.3 STUFF1 Variables

HMIDLO	Lower altitude midpoint
HMIDUP	Upper midpoint altitude
IPTFLG	Extra point in RTP flag; equals 1 if extra point present. Point removed before return to calling program.
MACH	Mach number associated with VGND in the calling sequence, corrected for along-track wind at the given altitude.
VATW	Along-track wind component (in knots)
VCAS	Calibrated airspeed (in knots)
VTAS	True airspeed (in knots)


SUBROUTINE STUFF1 (HNEW,VGND)
C THIS SUBROUTINE SETS UP THE ATPOL ARRAY WITH A
C PROPER SET OF VARIABLES TO ALLOW AN ATP INTEGRATION
C BY SUBROUTINE THIND. STUFF1 IS USED WHE AN AIRPLA
C GIVES ABOVE THE CRITICAL ALTITUDE AND T
C WILL MAKE EITHER A CONSTANT CALIBRATED AIRSPEED
C LETDOWN OR A CONSTANT MACH/CONSTANT CALIBRATED
C LETDOWN WITH THE BREAKPOINT AT OTHERWISE THAN THE
C CRITICAL ALTITUDE.
C
C THIS SUBROUTINE CALLS THE FOLLOWING SUBROUTINES
C TASCAS
C SIAS
C
C COMMENT ON CALLING VARIABLES ***
C *** HNEW WILL DETERMINE ALTITUDE AT WHICH
C THE TRANSITION FROM MACH TO CALIBRATED
C AIRSPEED IS MADE. IF HNEW IS SET EQUAL
C TO ARRIVAL ALTITUDE, A CALIBRATED AIRSPEED
C LETDOWN WILL RESULT.
C *** VGND IS THE GROUND SPEED DESIRED AT THE
C END OF THE LEVEL SEGMENT AT ENTRY ALTITUDE.
C VGND SHOULD BE SET TO VSLOW (VMAX) FOR
C NORMAL ALGORITHM OPERATION.
C
C
C INITIALIZE EXTRA POINT FLAG
C
IPTFLG =0
C STORE ENTRY GROUND SPEED, ALTITUDE AND ID
C
VATP(1,IS) = SPKNOT
HATP(1,IS) =ALT(IS)
IDATP(1,IS) =1
C CONVERT THE START OF DESCENT G/S TO MACH
C STORE G/S WITH ALTITUDE AND ID
C
CALL WETHER (ALT(IS),0)
VATW = WINMAG*COS(ANGEF(IQUAD,INFIX) - WINANG)
MACH = (VGND - VATW) / (38.975*SQRT(TEMP))
VATP(2,IS) =VGND
HATP(2,IS) =ALT(IS)
IDATP(2,IS) =0
C COMPUTE UPPER MIDPOINT ALTITUDE
C TEST FOR A CAS OR A MACH/CAS LETDOWN
C
HMIDUP=0.5*(ALT(IS)+HCRIT)
IF
(HNEW.NE.ALT(IS))
GO TO 204
A
C CONSTANT CAS LETDOWN
C
VTAS=VGND-VATW
CALL TASCAS(ALT(IS),VTAS,VCAS)
C STORE G/S WITH ALTITUDE AND ID
C *** BE CAREFUL *** THIS THIRD POINT IS A DUMMY
C FROM THE ATP POINT OF VIEW, BUT LOGICALLY CONSISTANT.
C AND IS CLEANED UP ON EXIT.
C
IPTFLG = 1
VATP(3,IS) =VGND
HATP(3,IS) =ALT(IS)
IDATP(3,IS) =0
GO TO 208
B


A
C MACH/CAS LETDOWN
C TEST THE MACH/CAS BREAKPOINT FOR ALTITUDE LOCATIO
C
201 CONTINUE
IF
(HNEW.LE.HMIDUP)
GO TO 206
C CONVE T MACH TO TAS, G/S AND CAS AT HNEW
C STORE G/S WITH ALTITUDE AND ID
C
CALL WETHER (HNEW,0)
VATW = WINMAG*.05 (ANGEF (IQUAD, INFIX) -WINANG)
VTAS=38.975*SQRT (TEMP) *MACH
CALL TASCAS (HNEW, VTAS, VCAS)
VATP (3, IS) =VTAS + VATW
HATP (3, IS) =HNEW
IDATP (3, IS) =0
B
208 CONTINUE
C CONVERT CAS TO TAS AND G/S AT HMIDUP
C STORE G/S WITH ALTITUDE AND ID
C
CALL WETHER (HMIDUP, 0)
VATW = WINMAG*COS (ANGEF (IQUAD, INFIX) -WINANG)
CALL CASTAS (HMIDUP, VCAS, VTAS)
VATP (4, IS) =VTAS + VATW
HATP (4, IS) =HMIDUP
IDATP (4, IS) =0
GO TO 209
C CONVERT MACH T* TAS AND G/S AT HMIDUP
C STORE G/S WI H ALTITUDE AND ID
C
206 CONTINUE
CALL WETHER (HM DUP, 0)
VATW = WINM AG*COS (ANGEF (IQUAD, INFIX) -WINANG)
VATP (3, IS) =38.975*SQAT (TEMP) *MACH+ VATW
HATP (3, IS) =HMIDUP
IDATP (3, IS) =0
C CONVERT MACH TO TAS, G/S AND CAS AT HNEW
C STORE G/S WITH ALTI UDE AND ID
C
CALL WETHER (HNEW, 0)
VATW = WINMAG*COS (ANGEF (IQUAD, INFIX) -WINANG)
VTAS=38.975*SQRT (TE P) *MACH
CALL TASCAS (HNEW, VTAS, VCAS)
VATP (4, IS) =V IAS + VATW
HATP (4, IS) =HNEW
IDATP (4, IS) =0
C CONVERT CAS TO TAS AND G/S AT HCRIT
C STORE G/S WI H ALTITUDE AND ID
C
C


C
209 CONTINUE
CALL WETHER (HCRIT,0)
VATW = WINMAG*COS (ANGEF (IQUAD, INFIX) -WINANG)
CALL CASTAS (HCRIT,VCAS,VTAS)
VATP (5, IS) =VTAS + VATW
HATP (5, IS) =HCRIT
IDATP (5, IS) =0
C COMPUTE LOWER ALTITUDE MIDPOINT
C CONVERT VCAS TO TAS AND G/S AT HMID_0
C STORE G/S WITH ALTITUDE AND ID
C
HMID_0= (HCRIT+HIAF) /2.
CALL WETHER (HMID_0 0)
VATW = WINMAG *COS (ANGEF (IQUAD, INFIX) -WINANG)
CALL CASTAS (HMID_0, VCAS, VTAS)
VATP (6, IS) =VTAS + VATW
HATP (6, IS) =HMID_0
IDATP (6, IS) =0
C ASSURE CAS AT LEAST 250 KNOTS ON ENTRY TO DDP SEGMENT
C CONVERT CAS TO TAS AND G/S AT ENTRY TO DDP SEGMENT
C STORE G/S WITH ALTITUDE AND ID
C
IF (VCAS.LT.250.)
VCAS=250.
CALL WETHER (HIAF,0)
VATW = WINMAG*COS (ANGEF (IQUAD, INFIX) -WINANG)
CALL CASTAS (HIAF, VCAS, VTAS)
VATP (7, IS) =VTAS + VATW
HATP (7, IS) =HIAF
IDATP (7, IS) =0
C GET IAF TRACK ANGLE AND WIND
C DETERMINE IAF G/S AND STORE WITH ALT AND ID
C
CALL WETHER (HIAF,0)
VATW = WINMAG*COS (ANGIAF (IQUAD) -WINANG)
VCAS = 250.
CALL CASTAS (HIAF,VCAS,VTAS)
VATP (8, IS) =VTAS + VATW
HATP (8, IS) =HIAF
IDATP (8, IS) =2
C SET ET TIME AND MAJOR SEGMENT DISTANCES IN ATPOINT
C
TATP (1, IS) =TIAF
DATP (1, IS) =0.
DATP (2, IS) =DLF
DATP (8, IS) =DDP
C CHECK EXTRA POINT FLAG AND CLEAN UP IF NECESSARY
C
IF (IPTFLG.EQ.0)
RETURN
DO 210 JATP=3,7
DATP (JATP, IS) =DATP (JATP+1, IS)
TATP (JATP, IS) =TATP (JATP+1, IS)
VATP (JATP, IS) =VATP (JATP+1, IS)
HATP (JATP, IS) =HATP (JATP+1, IS)
IDATP (JATP, IS) =IDATP (JATP+1, IS)
210 CONTINUE
RETURN

4.2.42 Subroutine STUFF2

4.2.42.1 Abstract

Subroutine STUFF2 (VNEW) sets up COMMON/RTPOUT/ with the variables to allow an RTP integration by subroutine TFIND. STUFF2 is used when an airplane arrives above the critical altitude and the descent will be made via constant Mach to the critical altitude and then via a constant calibrated airspeed.

4.2.42.2 Data Interfaces

STUFF2 Inputs

Through Calling Sequence:

VNEW	True airspeed, corrected for wind, of the airplane arriving at the EF

Through Common Statements:

/AEROOT/	HCRIT
/GEOMOT/	ANGEF(IQUAD,INFIX), ANGIAF(IQUAD)
/RTPO/	DDP, DLF, HIAF, INFIX, IQUAD, IS, TIAF
/TEMPER/	TEMP
/WIND/	WINANG, WINMAG
/TRAFOT/	ALT(IS)

STUFF2 Outputs

Through Common Statements:

/RTPOUT/	DRTP(-,IS), HRTP(-,IS), IDRTP(-,IS), TRTP(-IS), VRTP(-,IS)

STUFF2 Calls

CASTAS	Converts calibrated airspeed to true airspeed at a given pressure altitude
SQRT	FORTRAN library function
TASCAS	Converts true airspeed to calibrated airspeed at a given pressure altitude
WETHER	Finds the forecast along-track wind at a given altitude

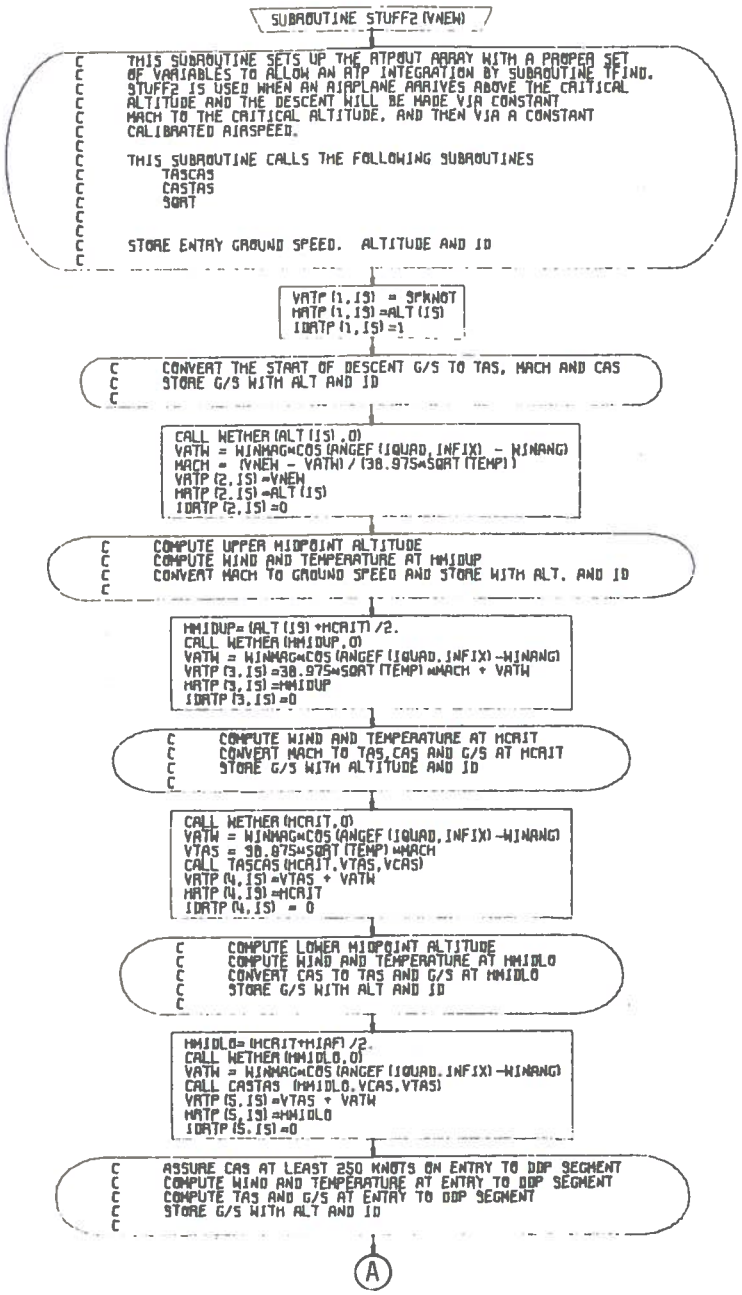
STUFF2 Called by

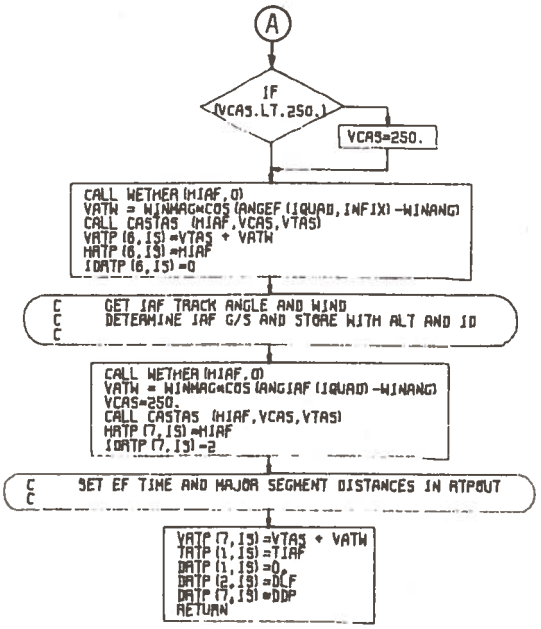
RTP, FASLOW

4.2.42.3 STUFF2 Variables

HMIDLO	Lower midpoint altitude
HMIDUP	Upper midpoint altitude

MACH2	Start of descent groundspeed (in Mach)
VATW	Along-track wind component (in knots)
VCAS	Calibrated airspeed (in knots)
VTAS	True airspeed (in knots)




A
IF
(VCAS.LT.250.)
VCAS=250.
CALL WETHER (HIAF,0)
VATW = WINMAG*COS (ANGEF (IQUAD,INFIX) -WINANG)
CALL CASTAS (HIAF,VCAS,VTAS)
VATP (6,IS) =VTAS + VATW
HATP (6,IS) =HIAF
IDATP (6,IS) =0
C GET IAF TRACK ANGLE AND WIND
C DETERMINE IAF G/S AND STORE WITH ALT AND ID
C
CALL WETHER (HIAF,0)
VATW = WINMAG*COS (ANGIAF (IQUAD) -WINANG)
VCAS=250.
CALL CASTAS (HIAF,VCAS,VTAS)
HATP (7,IS) =HIAF
IDATP (7,IS) =2
C SET EF TIME AND MAJOR SEGMENT DISTANCES IN ATPOUT
C
VATP (7,IS) =VTAS + VATW
TATP (1,IS) =TIAF
DATP (1,IS) =0.
DATP (2,IS) =DCF
DATP (7,IS) =DDP
RETURN

4.2.43 Subroutine STUFF3

4.2.43.1 Abstract

Subroutine STUFF3 (VNEW) sets up COMMON/RTPOUT/ with the variables to allow an RTP integration by subroutine TFIND. STUFF3 is used when an airplane arrives below the critical altitude, and the descent will be made via constant calibrated airspeed.

4.2.43.2 Data Interfaces

STUFF3 Inputs

Through Calling Sequence:

VNEW	True airspeed, corrected for wind, of the airplane arriving at the EF

Through Common Statements:

/GEOMOT/	ANGEF (IQUAD, INFIX), ANGIAF (IQUAD)
/RTPO/	DDP, DLF, HIAF, INFIX, IQUAD, IS, TIAF
/TRAFOT/	ALT (IS)
/WIND/	WINANG, WINMAG
/AEROOT/	SPKNOT

STUFF3 Outputs

Through Common Statements:

/RTPOUT/	DRTP (-, IS), HRTP (-, IS), IDRTP (-, IS), TRTP (-, IS), VRTP (-, IS)

STUFF3 Calls

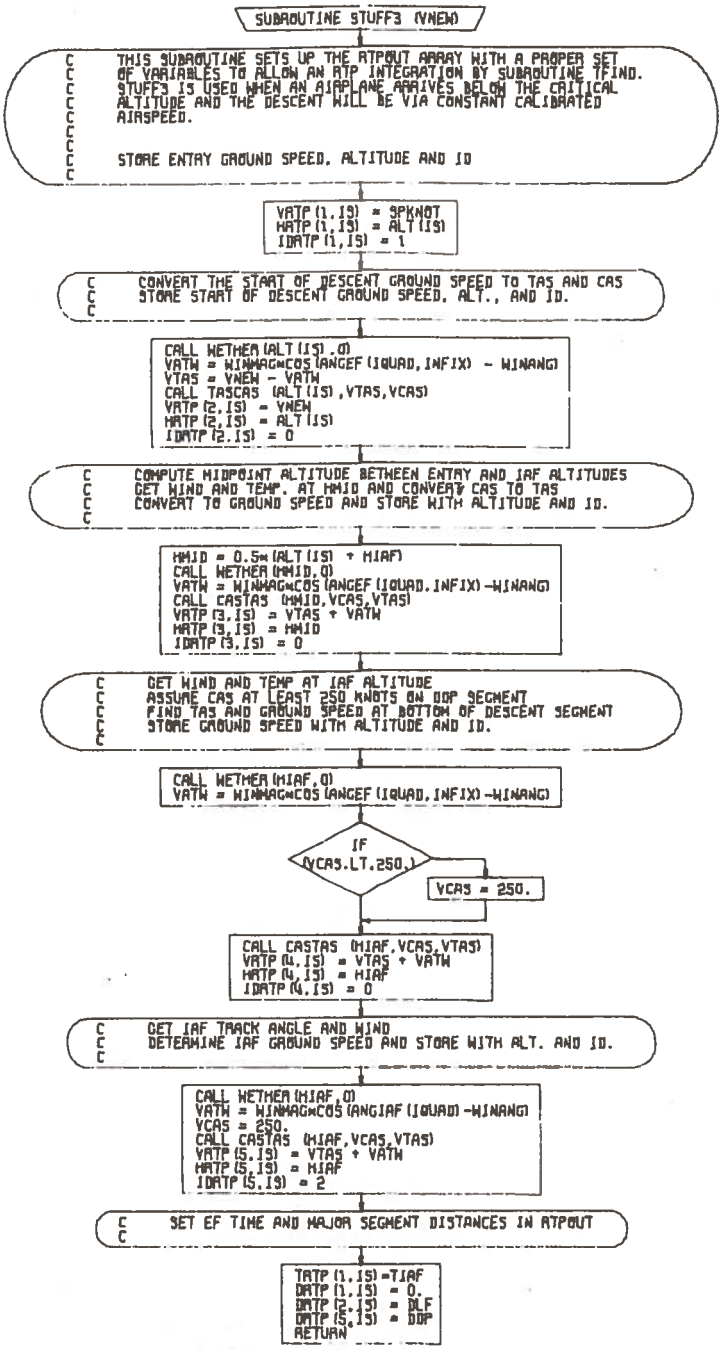
CASTAS	Converts calibrated airspeed to true airspeed at a given pressure altitude
SQRT	FORTRAN library function
TASCAS	Converts true airspeed to calibrated airspeed at a given pressure altitude
WETHER	Finds the forecast along-track wind at a given altitude

STUFF3 Called by

FASLOW, RTP

4.2.43.3 STUFF3 Variables

HMID	Midpoint altitude between entry and IAF altitudes
VATW	Along-track wind component (in knots)
VCAS	Calibrated airspeed (in knots)
VTAS	True airspeed (in knots)


SUBROUTINE STUFF3 (VNEW)
C THIS SUBROUTINE SETS UP THE RTPOUT ARRAY WITH A PROPER SET
C OF VARIABLES TO ALLOW AN RTP INTEGRATION BY SUBROUTINE TFIND.
C STUFF3 IS USED WHEN AN AIRPLANE ARRIVES BELOW THE CRITICAL
C ALTITUDE AND THE DESCENT WILL BE VIA CONSTANT CALIBRATED
C AIRSPEED.
C
C
C STORE ENTRY GROUND SPEED, ALTITUDE AND ID
C
VRTP(1,IS) = 9PKNOT
HRTP(1,IS) = ALT(IS)
IDRTP(1,IS) = 1
C CONVERT THE START OF DESCENT GROUND SPEED TO TAS AND CAS
C STORE START OF DESCENT GROUND SPEED, ALT., AND ID.
C
CALL WETHER(ALT(IS),0)
VATH = WINMAG*COS(ANGEF(IQUAD,INFIX) - WINANG)
VTAS = VNEW - VATH
CALL TASCAS (ALT(IS),VTAS,VCAS)
VRTP(2,IS) = VNEW
HRTP(2,IS) = ALT(IS)
IDRTP(2,IS) = 0
C COMPUTE MIDPOINT ALTITUDE BETWEEN ENTRY AND IAF ALTITUDES
C GET WIND AND TEMP. AT HMID AND CONVERT CAS TO TAS
C CONVERT TO GROUND SPEED AND STORE WITH ALTITUDE AND ID.
C
HMID = 0.5*(ALT(IS) + HIAF)
CALL WETHER(HMID,0)
VATH = WINMAG*COS(ANGEF(IQUAD,INFIX)-WINANG)
CALL CASTAS (HMID,VCAS,VTAS)
VRTP(3,IS) = VTAS + VATH
HRTP(3,IS) = HMID
IDRTP(3,IS) = 0
C GET WIND AND TEMP AT IAF ALTITUDE
C ASSUME CAS AT LEAST 250 KNOTS ON DOP SEGMENT
C FIND TAS AND GROUND SPEED AT BOTTOM OF DESCENT SEGMENT
C STORE GROUND SPEED WITH ALTITUDE AND ID.
C
CALL WETHER(HIAF,0)
VATH = WINMAG*COS(ANGEF(IQUAD,INFIX)-WINANG)
IF (VCAS.LT.250.)
VCAS = 250.
CALL CASTAS (HIAF,VCAS,VTAS)
VRTP(4,IS) = VTAS + VATH
HRTP(4,IS) = HIAF
IDRTP(4,IS) = 0
C GET IAF TRACK ANGLE AND WIND
C DETERMINE IAF GROUND SPEED AND STORE WITH ALT. AND ID.
C
CALL WETHER(HIAF,0)
VATH = WINMAG*COS(ANGIAF(IQUAD)-WINANG)
VCAS = 250.
CALL CASTAS (HIAF,VCAS,VTAS)
VRTP(5,IS) = VTAS + VATH
HRTP(5,IS) = HIAF
IDRTP(5,IS) = 2
C SET EF TIME AND MAJOR SEGMENT DISTANCES IN RTPOUT
C
TRTP(1,IS)=TIAF
DRTP(1,IS) = 0.
DRTP(2,IS) = DLF
DRTP(5,IS) = DDP
RETURN

4.2.44 Subroutine TASCAS

4.2.44.1 Abstract

Subroutine TASCAS (H, VTAS, VCAS) returns the calibrated airspeed (VCAS), given the pressure altitude (H) and true airspeed(VTAS).

4.2.44.2 Data Interfaces

TASCAS Inputs

Through Calling Sequence:

H	Pressure altitude (in thousands of feet)
VTAS	True airspeed (in knots)

Through Common Statements:

/ATMO/	DENSRF	Atmospheric density ratio factor
	PRESR	Standard day pressure ratio

TASCAS Outputs

VCAS	Calibrated airspeed (in knots)

TASCAS Calls

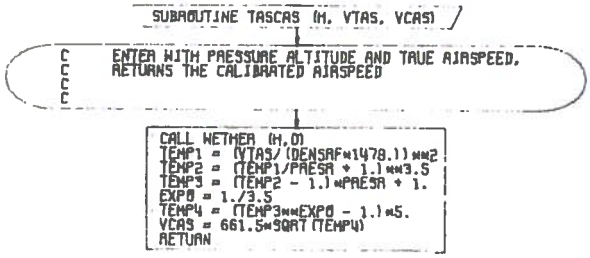
WETHER (H, 0)	WETHER in turn, calls ATMOSP, which provides DENSRF and PRESR
SQRT	FORTRAN library function

TASCAS Called by

STUFF1, STUFF2, STUFF3

4.2.44.3 TASCAS Variables

None



4.2.45 Function TBLU

4.2.45.1 Abstract

Function TBLU does linear interpretation on a monotonic function.

4.2.45.2 Data Interfaces

TBLU Inputs

N	The number of data points on the curve
X	An array containing the N-dependent values of the curve
Y	An array containing the N-independent values of the curve
YO	A Y value for which the corresponding X is desired

TBLU Outputs

TBLU	The X value corresponding to YO on the curve

TBLU Calls

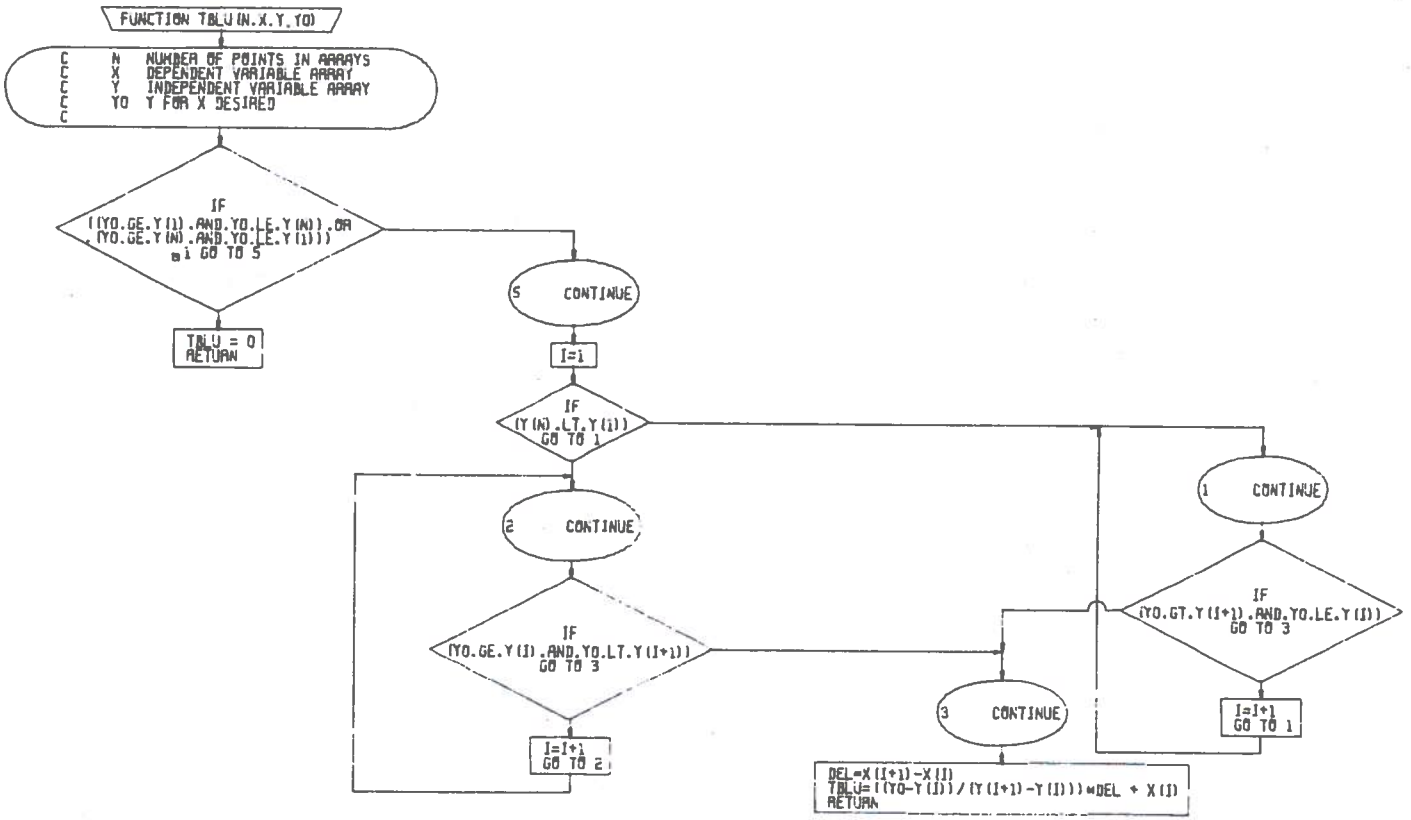
None

TBLU Called by

ERLANG, TRAFIC

4.2.45.3 TBLU Variables

None


FUNCTION TBLU(N,X,Y,YO)
C N NUMBER OF POINTS IN ARRAYS
C X DEPENDENT VARIABLE ARRAY
C Y INDEPENDENT VARIABLE ARRAY
C YO Y FOR X DESIRED
C
IF
((YO.GE.Y(1).AND.YO.LE.Y(N)).OR.
(YO.GE.Y(N).AND.YO.LE.Y(1)))
=1 GO TO 5
TBLU = 0
RETURN
5 CONTINUE
I=1
IF
(Y(N).LT.Y(I))
GO TO 1
2 CONTINUE
IF
(YO.GE.Y(I).AND.YO.LT.Y(I+1))
GO TO 3
I=I+1
GO TO 2
1 CONTINUE
IF
(YO.GT.Y(I+1).AND.YO.LE.Y(I))
GO TO 3
I=I+1
GO TO 1
3 CONTINUE
DEL=X(I+1)-X(I)
TBLU=((YO-Y(I))/(Y(I+1)-Y(I)))*DEL + X(I)
RETURN

4.2.46 Subroutine TFIND

4.2.46.1 Abstract

Subroutine TFIND (IS, SUMT) transforms an altitude-groundspeed profile into a transition time and completes the RTP from the EF to the IAF.

4.2.46.2 Data Interfaces

TFIND Inputs

Through Calling Sequence:

IS	Index of the airplane whose RTP is being computed

Through Common Statements:

/CNSTNT/	GRADI	
/RTPOUT/	HRTP (-, IS)	Altitudes previously generated
	VRTP (-, IS)	Velocities previously generated
	IDRTP (-, IS)	Waypoint identification array previously generated

TFIND Outputs

Through Calling Sequence:

SUMT	Time to transit from EF to IAF

Through Common Statements:

/RTPOUT/	DRTP (-, IS)	Cumulative waypoint distances between EF and IAF
	TRTP (-, IS)	Cumulative transition times from EF to IAF

TFIND Calls

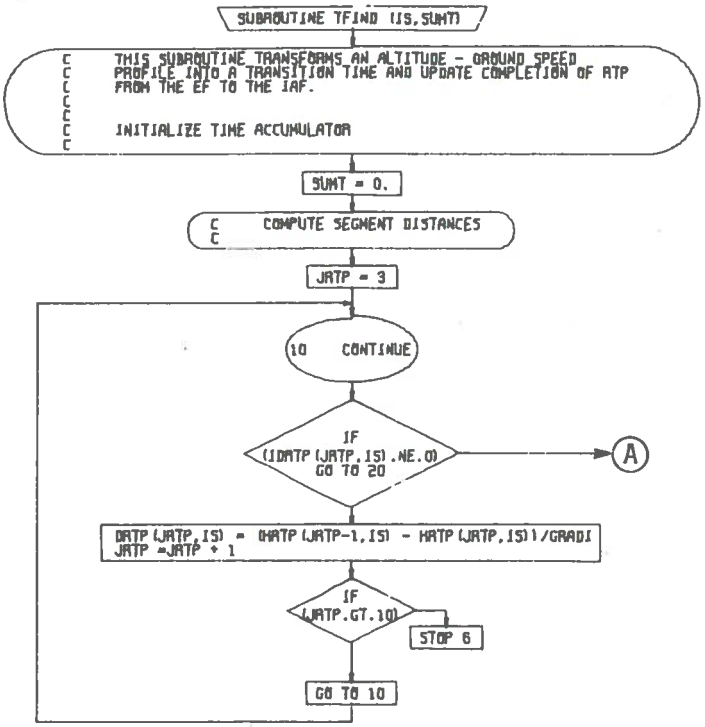
None

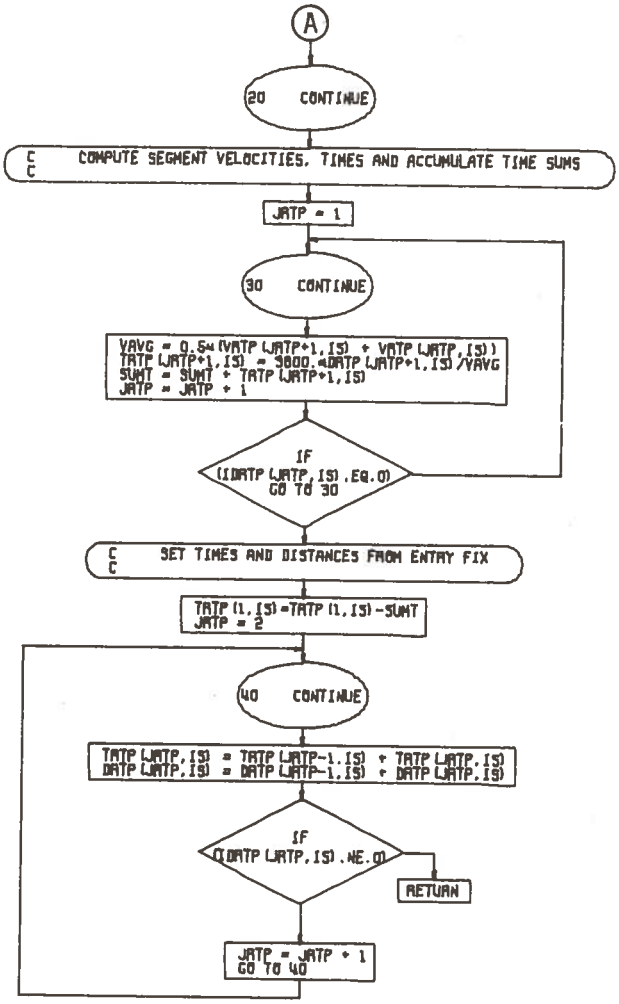
TFIND Called by

RTP

4.2.46.3 TFIND Variables

JRTP	Waypoint index
VAVG	Average velocity between two waypoints (in knots)


SUBROUTINE TFIND (IS,SUMT)
C THIS SUBROUTINE TRANSFORMS AN ALTITUDE - GROUND SPEED
C PROFILE INTO A TRANSITION TIME AND UPDATE COMPLETION OF RTP
C FROM THE EF TO THE IAF.
C
C
C INITIALIZE TIME ACCUMULATOR
C
SUMT = 0.
C COMPUTE SEGMENT DISTANCES
C
JRTP = 3
10 CONTINUE
IF
(IDRTP(JRTP,IS).NE.0)
GO TO 20
A
DRTP(JRTP,IS) = (HRTP(JRTP-1,IS) - HRTP(JRTP,IS))/GRADI
JRTP =JRTP + 1
IF
(JRTP.GT.10)
STOP 6
GO TO 10


A
20 CONTINUE
C COMPUTE SEGMENT VELOCITIES, TIMES AND ACCUMULATE TIME SUMS
C
JATP = 1
30 CONTINUE
VAVG = 0.5*(VATP(JATP+1,IS) + VATP(JATP,IS))
TATP(JATP+1,IS) = 3600.*DATP(JATP+1,IS)/VAVG
SUMT = SUMT + TATP(JATP+1,IS)
JATP = JATP + 1
IF (IDATP(JATP,IS).EQ.0) GO TO 30
C SET TIMES AND DISTANCES FROM ENTRY FIX
C
TATP(1,IS)=TATP(1,IS)-SUMT
JATP = 2
40 CONTINUE
TATP(JATP,IS) = TATP(JATP-1,IS) + TATP(JATP,IS)
DATP(JATP,IS) = DATP(JATP-1,IS) + DATP(JATP,IS)
IF (DATP(JATP,IS).NE.0)
RETURN
JATP = JATP + 1
GO TO 40

4.2.47 Subroutine TIMAA

4.2.47.1 Abstract

Subroutine TIMAA (T, K1, K) determines the minimum runway constraint time from the scheduling of arrival K1. The constraint value T represents the earliest time that arrival K can be scheduled subject to the ATC constraints.

4.2.47.2 Data Interfaces

TIMAA Inputs

Through Calling Sequence:

K	Sequence number of arrival airplane being scheduled
K1	Sequence number of arrival previously scheduled

Through Common Statements:

/TRAFOT/	VBUGG (I), ITYPEA (J)
/ATCIN/	DIST (I), TIME (I), IRWO (I), TARO (J)
/TRAFIN/	NTYPEA, NMTYPA (I)
/AERO1O/	VCG (I, J)
/GEOMOT/	DOM (1)

TIMAA Outputs

Through Calling Sequence:

T	Minimum runway constraint time

TIMAA Calls

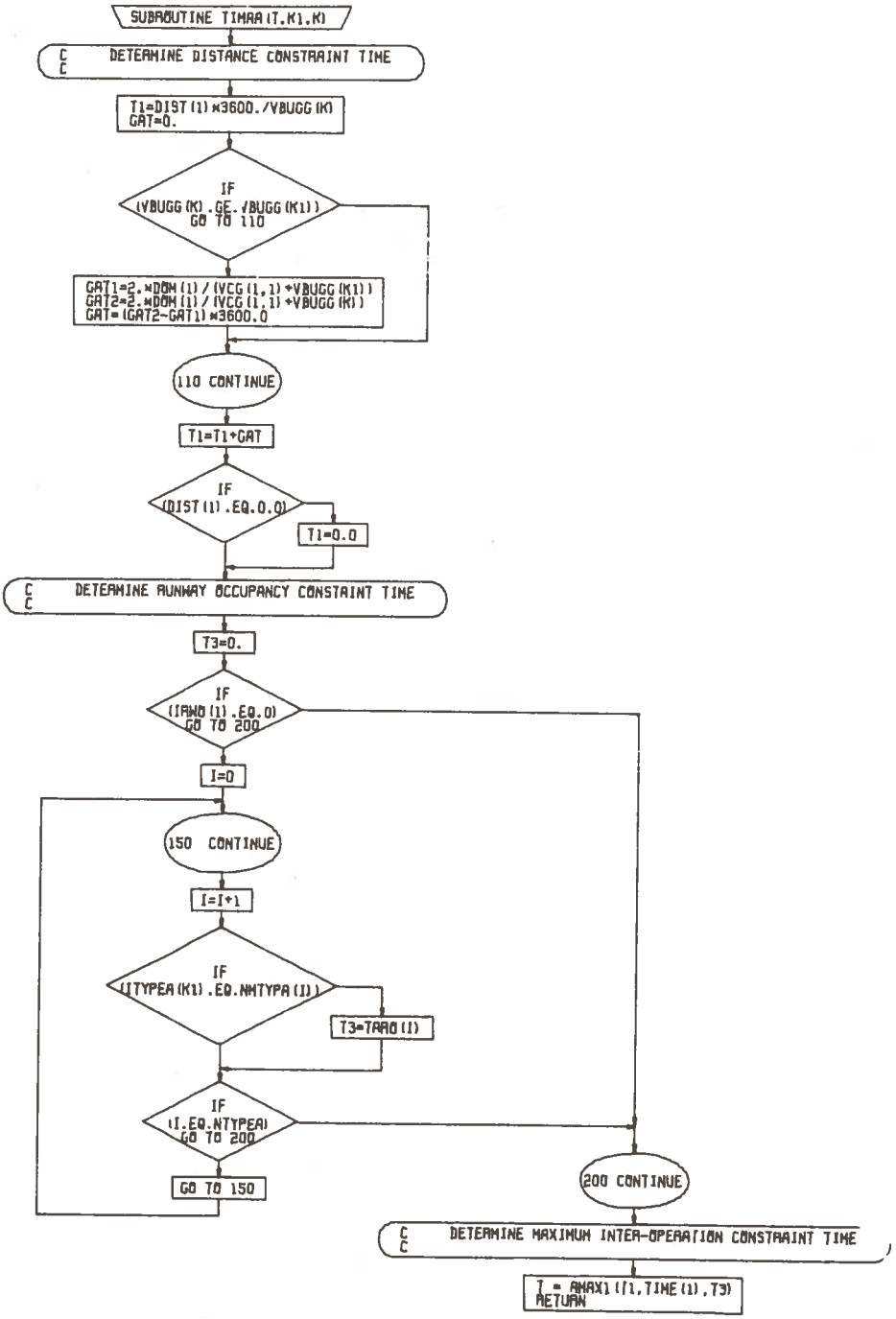
None

TIMAA Called by

SCHLD1, SCHLD2, SCHLD3

4.2.47.3 TIMAA Variables

GAT1	Outer marker to threshold time of arrival K1
GAT2	Outer marker to threshold time of arrival K
GAT	Getaway time to ensure that fast airplane does not overtake slow airplane on final
T3	Runway occupancy constraint time


SUBROUTINE TIMAA(T,K1,K)
C DETERMINE DISTANCE CONSTRAINT TIME
C
T1=DIST(1)*3600./VBUGG(K)
GAT=0.
IF (VBUGG(K).GE./BUGG(K1)) GO TO 110
GAT1=2.*DOM(1)/(VCG(1,1)+VBUGG(K1))
GAT2=2.*DOM(1)/(VCG(1,1)+VBUGG(K))
GAT=(GAT2-GAT1)*3600.0
110 CONTINUE
T1=T1+GAT
IF (DIST(1).EQ.0.0)
T1=0.0
C DETERMINE RUNWAY OCCUPANCY CONSTRAINT TIME
C
T3=0.
IF (IAWO(1).EQ.0) GO TO 200
I=0
150 CONTINUE
I=I+1
IF (ITYPEA(K1).EQ.NMTYPA(I))
T3=TAAO(I)
IF (I.EQ.NTYPEA) GO TO 200
GO TO 150
200 CONTINUE
C DETERMINE MAXIMUM INTER-OPERATION CONSTRAINT TIME
C
T = AMAX1(T1,TIME(1),T3)
RETURN

4.2.48 Subroutine TIMAD

4.2.48.1 Abstract

Subroutine TIMAD (T, K, J) determines the minimum runway constraint time from the scheduled landing time of arrival K. The value T represents the earliest time that departure J can be scheduled subject to the ATC constraints.

4.2.48.2 Data Interfaces

TIMAD Inputs

Through Calling Sequence:

K	Sequence number of arrival previously scheduled
J	Sequence number of departure being scheduled

Through Common Statements:

/TRAFOT/	ITYPEA (I)
/ATCIN/	TIME (I), IRWO (I), TARO (J)
/TRAFIN/	NTYPEA, NMTYPA (I)

TIMAD Outputs

Through Calling Sequence:

T	Minimum runway constraint time

TIMAD Calls

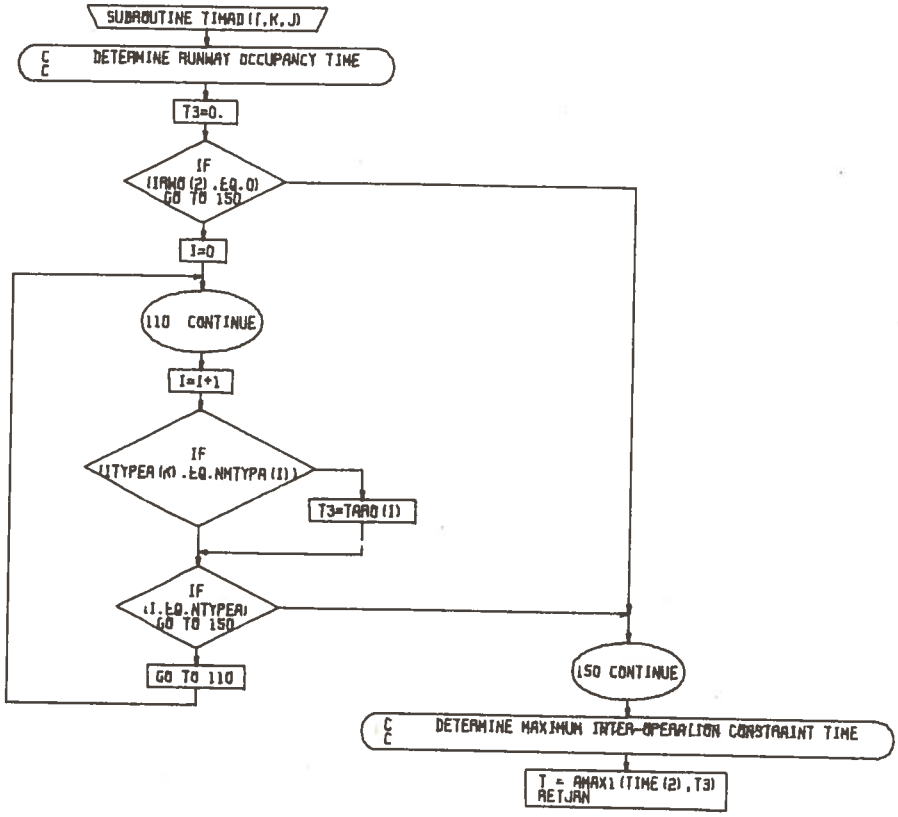
None

TIMAD Called by

SCHLD1, SCHLD2, SCHLD3

4.2.48.3 TIMAD Variables

T3	Runway occupancy constraint time


SUBROUTINE TIMAD(I,K,J)
C
C DETERMINE RUNWAY OCCUPANCY TIME
T3=0.
IF (IARRO(2).EQ.0) GO TO 150
I=0
110 CONTINUE
I=I+1
IF (ITYPEA(K).EQ.NMTYPA(I))
T3=TARO(I)
IF (I.EQ.NTYPEA) GO TO 150
GO TO 110
150 CONTINUE
C
C DETERMINE MAXIMUM INTER-OPERATION CONSTRAINT TIME
T = AMAX1(TIME(2),T3)
RETURN

4.2.49 Subroutine TIMDA

4.2.49.1 Abstract

Subroutine TIMDA (T, J, K) determines the minimum runway constraint time from the scheduled departure time of departure J. The value T represents the earliest time that arrival K can be scheduled subject to the ATC constraints.

4.2.49.2 Data Interfaces

TIMDA Inputs

Through Calling Sequence:

J	Sequence number of departure previously scheduled
K	Sequence number of arrival being scheduled

Through Common Statements:

/TRAFOT/	VBUGG (I), ITYPED (J)
/ATCIN/	DIST (I), TIME (I), IRWO (I), DROO (J)
/TRAFIN/	NTYPED, NMTYPD (J)
/AERO1O/	VCG (I, J)

TIMDA Outputs

Through Calling Sequence:

T	Minimum runway constraint time

TIMDA Calls

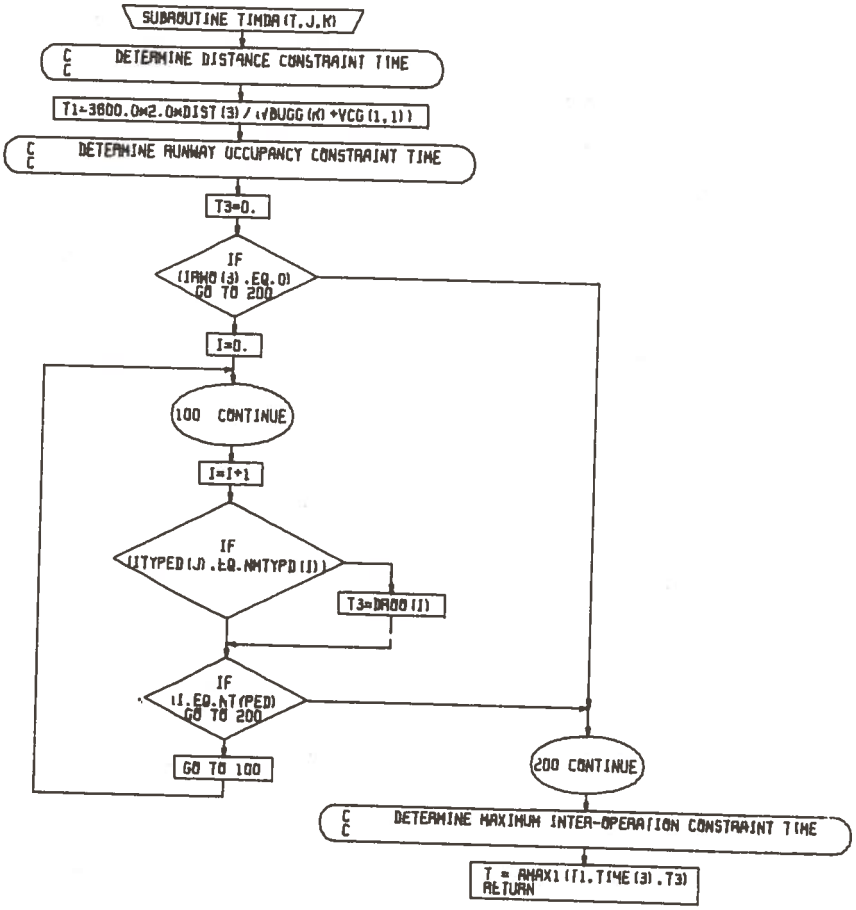
None

TIMDA Called by

SCHLD1, SCHLD2, SCHLD3

4.2.49.3 TIMDA Variables

T3	Runway occupancy constraint time


SUBROUTINE TIMDA(T,J,K)
C DETERMINE DISTANCE CONSTRAINT TIME
C
T1=3600.0*2.0*DIST(3)/(VBUGG(K)+VCG(1,1))
C DETERMINE RUNWAY OCCUPANCY CONSTRAINT TIME
C
T3=0.
IF (IAMO(3).EQ.0) GO TO 200
I=0.
100 CONTINUE
I=I+1
IF (ITYPED(J).EQ.NMTYPD(I))
T3=DADD(I)
IF (I.EQ.NTYPED) GO TO 200
GO TO 100
200 CONTINUE
C DETERMINE MAXIMUM INTER-OPERATION CONSTRAINT TIME
C
T = AMAX1(T1,TIME(3),T3)
RETURN

4.2.50 Subroutine TIMDD

4.2.50.1 Abstract

Subroutine TIMDD (T, J1, J) determines the minimum runway constraint time from the scheduling of departure J1. The constraint value T represents the earliest time that departure J can be scheduled subject to the ATC constraints.

4.2.50.2 Data Interfaces

TIMDD Inputs

Through Calling Sequence:

J1	Sequence number of last departure scheduled
J	Sequence number of next departure being scheduled

Through Common Statements:

/TRAFOT/	ITYPED (J)
/ATCIN/	DIST (I), TIME (I), IRWO (I), DROO (J)
/TRAFIN/	NTYPED, NMTYPD (J)
/CNSTNT/	ACCEL

TIMDD Outputs

Through Calling Sequence:

T	Minimum runway scheduling constraint time

TIMDD Calls

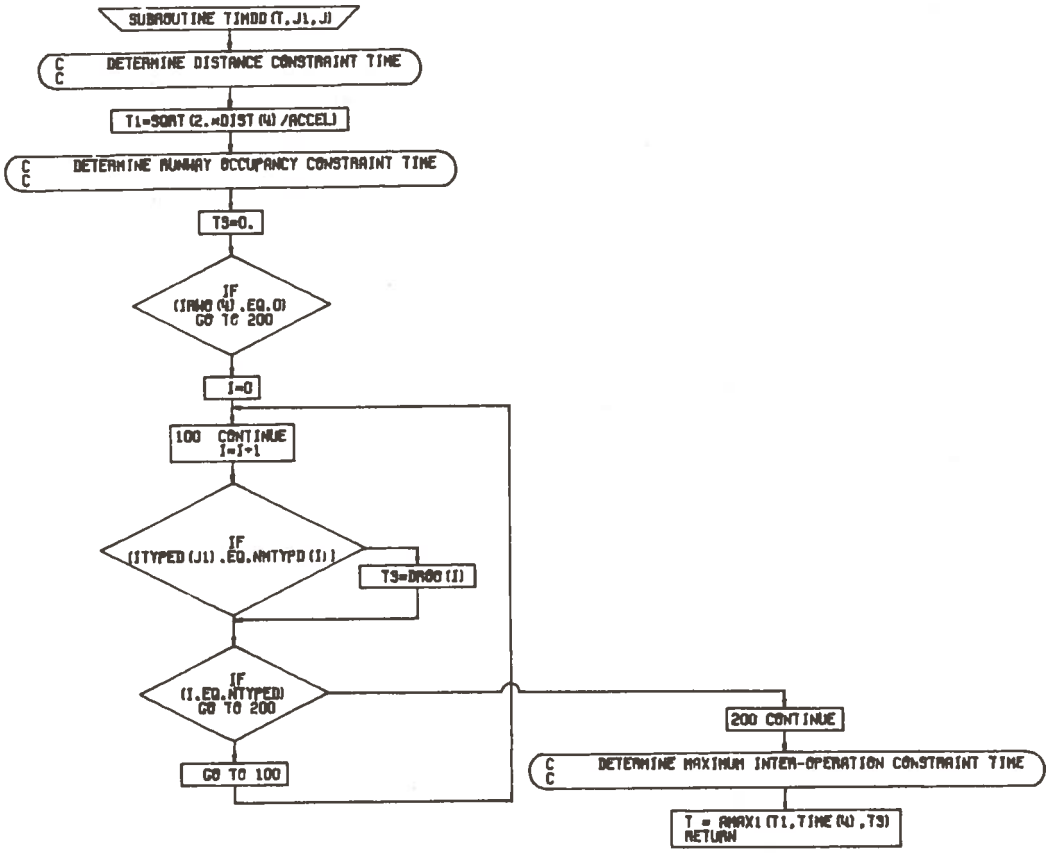
None

TIMDD Called by

SCHLD1, SCHLD2, SCHLD3

4.2.50.3 TIMDD Variables

T3	Runway occupancy constraint time


SUBROUTINE TIMDD(T,J1,J)
C
C DETERMINE DISTANCE CONSTRAINT TIME
T1=SQRT(2.*DIST(W)/ACCEL)
C
C DETERMINE RUNWAY OCCUPANCY CONSTRAINT TIME
T9=0.
IF
(IRWD(W).EQ.0)
GO TO 200
I=0
100 CONTINUE
I=I+1
IF
(ITYPED(J1).EQ.NWTYPD(I))
T9=DRGO(I)
IF
(I.EQ.NTYPED)
GO TO 200
GO TO 100
200 CONTINUE
C
C DETERMINE MAXIMUM INTER-OPERATION CONSTRAINT TIME
T = AMAX1(T1,TIME(W),T9)
RETURN

4.2.51 Subroutine TIMFND

4.2.51.1 Abstract

Subroutine TIMFND transforms an altitude-groundspeed profile into a transition time from the entry fix (EF) to the initial approach fix (IAF).

4.2.51.2 Data Interfaces

TIMFND Inputs

Through Calling Sequence:
IS

Through Common Statements:

/CNSTNT/	GRADI
/RTPOUT/	All variables

TIMFND Outputs

Through Calling Sequence:
SUMT

TIMFND Calls

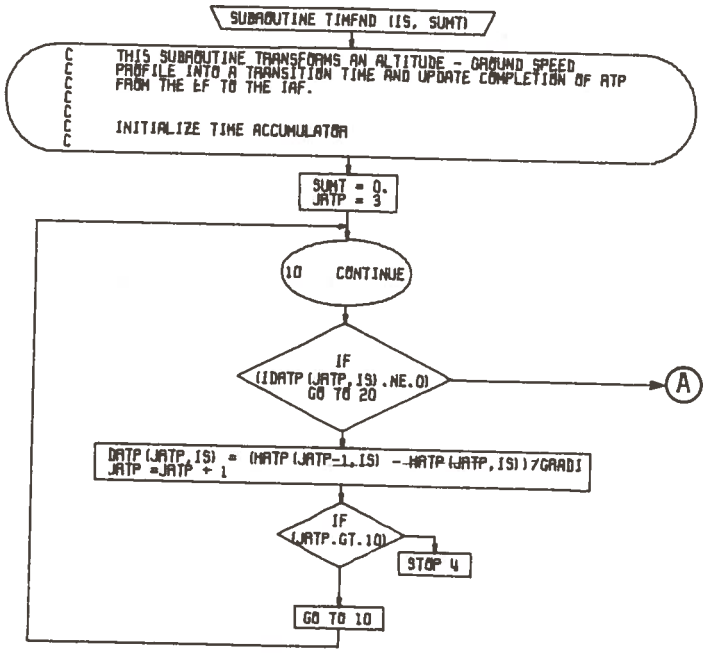
None

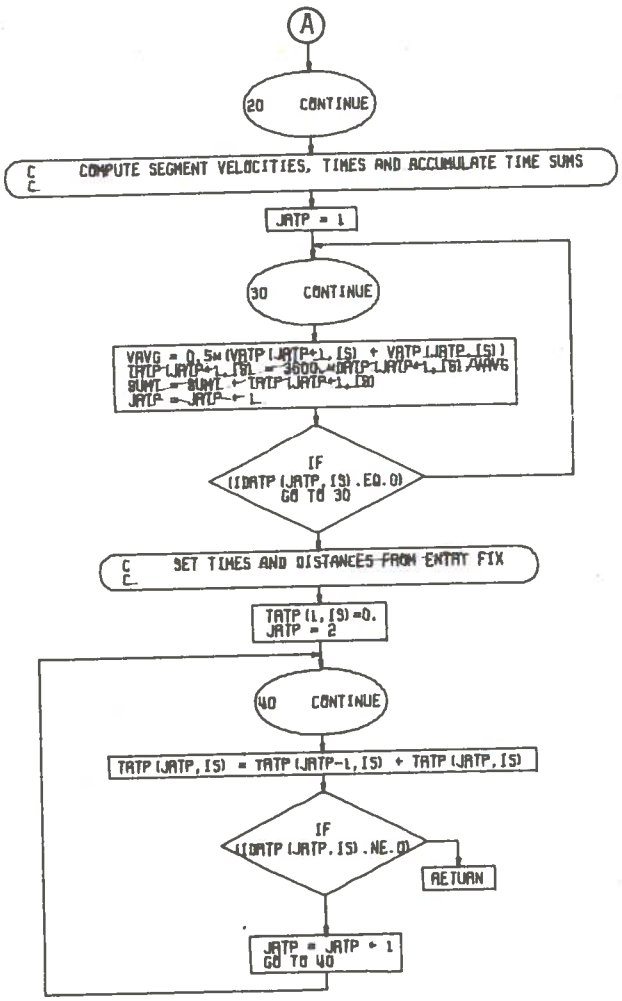
TIMFND Called by

FASLOW

4.2.51.3 TIMFND Variables

IS	Index of airplane under consideration
JRTP	Waypoint index used in the RTP arrays
VAVG	Average velocity between two waypoints (in knots)


SUBROUTINE TIMFND (IS, SUMT)
C THIS SUBROUTINE TRANSFORMS AN ALTITUDE - GROUND SPEED
C PROFILE INTO A TRANSITION TIME AND UPDATE COMPLETION OF ATP
C FROM THE EF TO THE IAF.
C
C INITIALIZE TIME ACCUMULATOR
SUMT = 0.
JATP = 3
10 CONTINUE
IF
(IDATP(JATP,IS).NE.0)
GO TO 20
A
DATP(JATP,IS) = (HATP(JATP-1,IS) - HATP(JATP,IS))/GRADI
JATP = JATP + 1
IF
(JATP.GT.10)
STOP 4
GO TO 10


A
20 CONTINUE
C COMPUTE SEGMENT VELOCITIES, TIMES AND ACCUMULATE TIME SUMS
C
JATP = 1
30 CONTINUE
VAVG = 0.5*(VATP(JATP+1,IS) + VATP(JATP,IS))
TATP(JATP+1,IS) = 3600.*DATP(JATP+1,IS)/VAVG
SUMT = SUMT + TATP(JATP+1,IS)
JATP = JATP + 1
IF
(IDATP(JATP,IS).EQ.0)
GO TO 30
C SET TIMES AND DISTANCES FROM ENTRY FIX
C
TATP(1,IS)=0.
JATP = 2
40 CONTINUE
TATP(JATP,IS) = TATP(JATP-1,IS) + TATP(JATP,IS)
IF
(IDATP(JATP,IS).NE.0)
RETURN
JATP = JATP + 1
GO TO 40

4.2.52 Subroutine TOMIAF

4.2.52.1 Abstract

Subroutine TOMIAF is called by AERO1. This subroutine is called for each path segment in each quadrant from the initial approach fix to the outer marker. TOMIAF returns groundspeed at each fix and path segment time.

4.2.52.2 Data Interfaces

TOMIAF Inputs

Through Calling Sequence:

HC	Altitude (in thousands of feet) at which other variables are to be computed
VCCAS	Calibrated airspeed (in knots)
VCG1	Groundspeed (in knots)

Through Common Statements:

/ATMO/	DENSRF
/OMIAF/	ANG, DIST
/WIND/	WINANG, WINMAG

TOMIAF Outputs

Through Calling Sequence:

VCTAS	True airspeed (in knots)
VCG	Wind-corrected groundspeed (in knots)
TC	Time duration (in seconds) for the flightpath segment

TOMIAF Calls

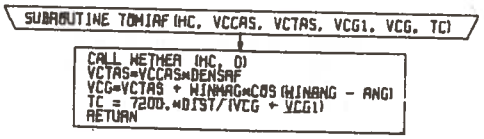
WETHER	Gives forecast wind and density ratio factor at the requested altitude (HC)
COS	FORTRAN library function

TOMIAF Called by

AERO1

4.2.52.3 TOMIAF Variables

None



4.2.53 Subroutine TRAFIC

4.2.53.1 Abstract

Subroutine TRAFIC is an airplane traffic simulator that generates the airplane arrival and departure operations. When SWTRAF is set to DISTRB, the arrival/departure traffic is generated according to the probability distributions input for arrival/departure time, airplane types, and the speed, weight, and altitude limits of each type. When SWTRAF is set to any other value, a list of prescheduled arrival/departure times, airplane types, and airplane weight and speed are entered with random variations generated according to distributions selected.

4.2.53.2 Data Interfaces

TRAFIC Inputs

Through Calling Sequence:

IFLITX	Print control parameter
ISAMP	Current sample number

Through Common Statements for Discrete Traffic Option:

/TRAFIN/	SWTRAF, CVREF, CWVREF, NCVREF, NMTYPA, TASIGM, TDSIGM, SPSIGM, NTYPEA
/TRAFOT/	NACA, NACD, TSECSD, ITYPEA, SPEED, TSECSA, GWA

Through Common Statements for Distribution Traffic Option:

/TRAFIN/	All variables except TASIGM, TDSIGM, SPSIGM

TRAFIC Outputs

Through Common Statements for Discrete Traffic Option:

/TRAFOT/	VREF, SPEED, TSECAA, TSECAD

Through Common Statements for Distribution Traffic Option:

/TRAFOT/	All variables except VBUGG, EELT, ELLT, KEY, SLT, SDT

TRAFIC Calls

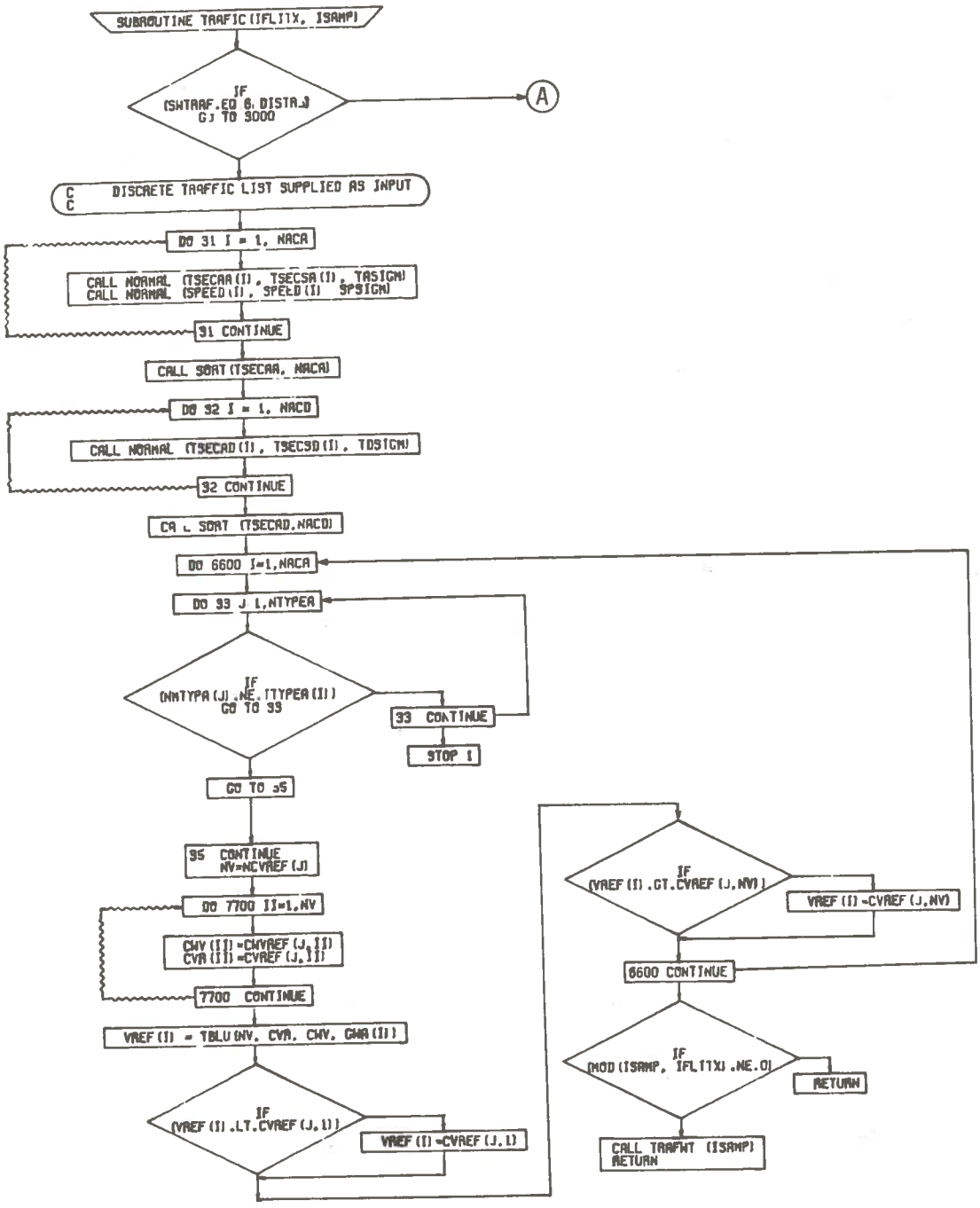
ERLANG	Subroutine to generate E(P1, P2) random variates
LIMITS	Subroutine to generate random variates of the specified probability distribution with preset upper/lower (or both) limits
NORMAL	Subroutine to generate N(P1, P2) random variates
SORT	Subroutine to sort an integer array
TBLU1	FORTRAN library function for interpretation
UNIFORM	Subroutine to generate U(P1, P2) random variates

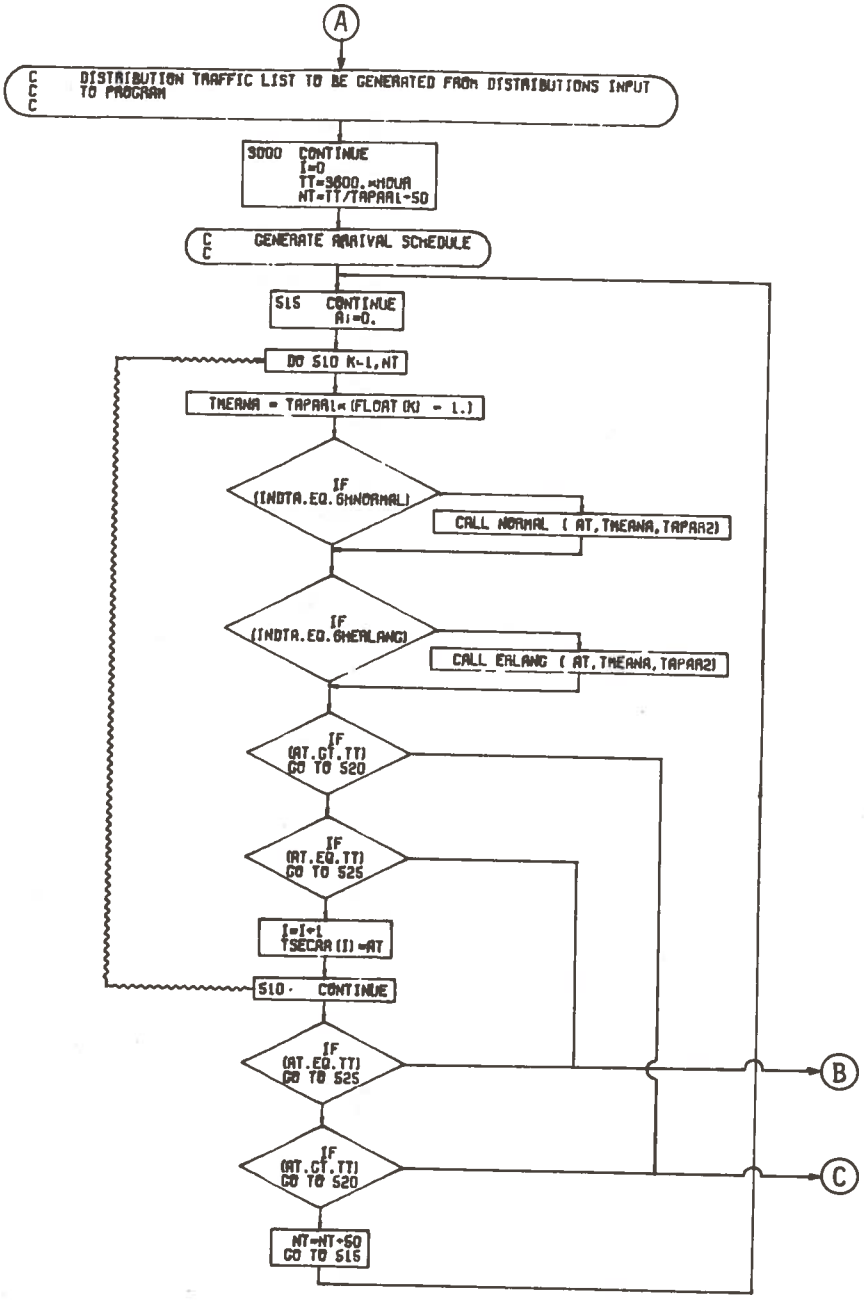
TRAFIC Called by

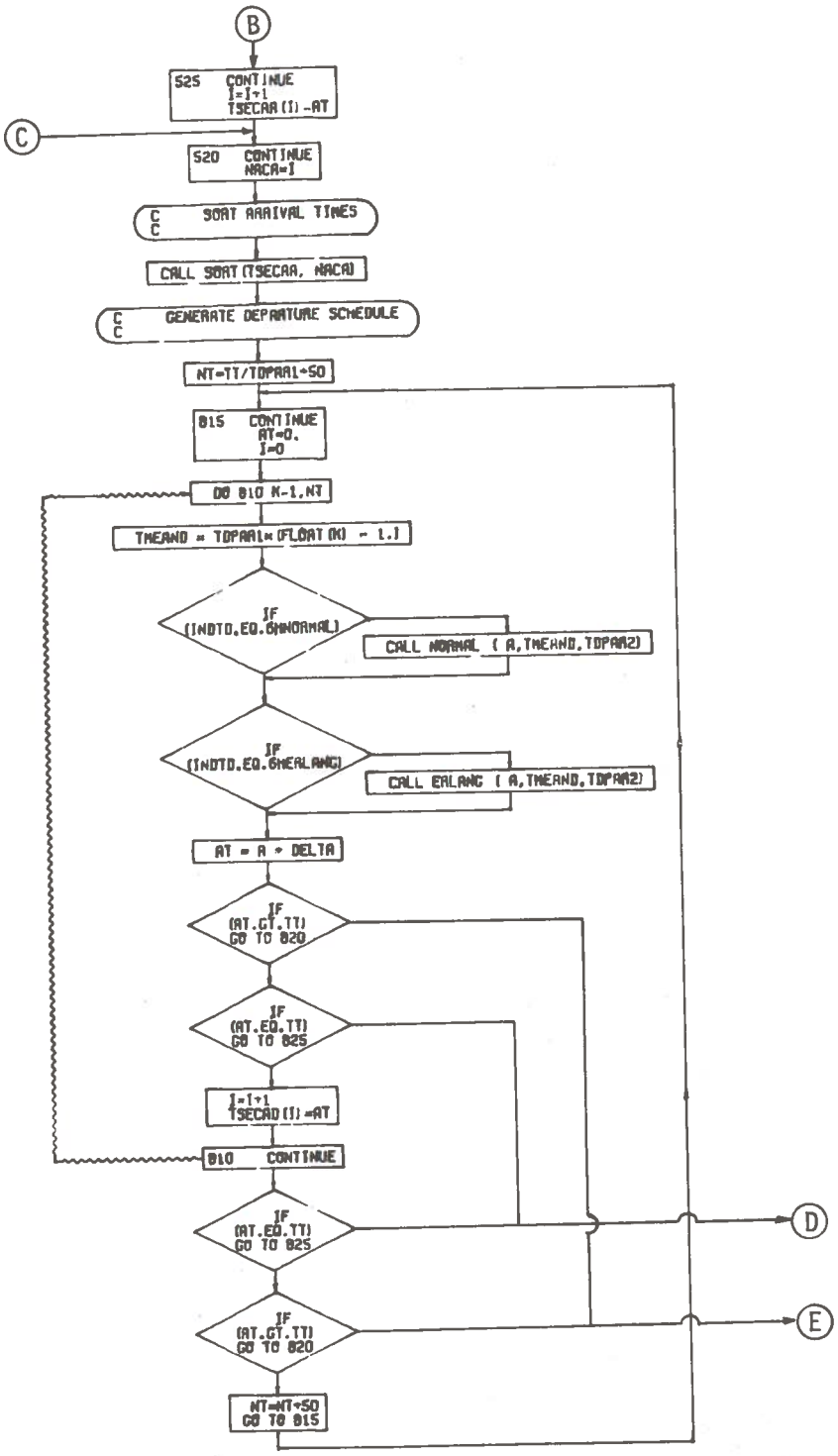
STRAD

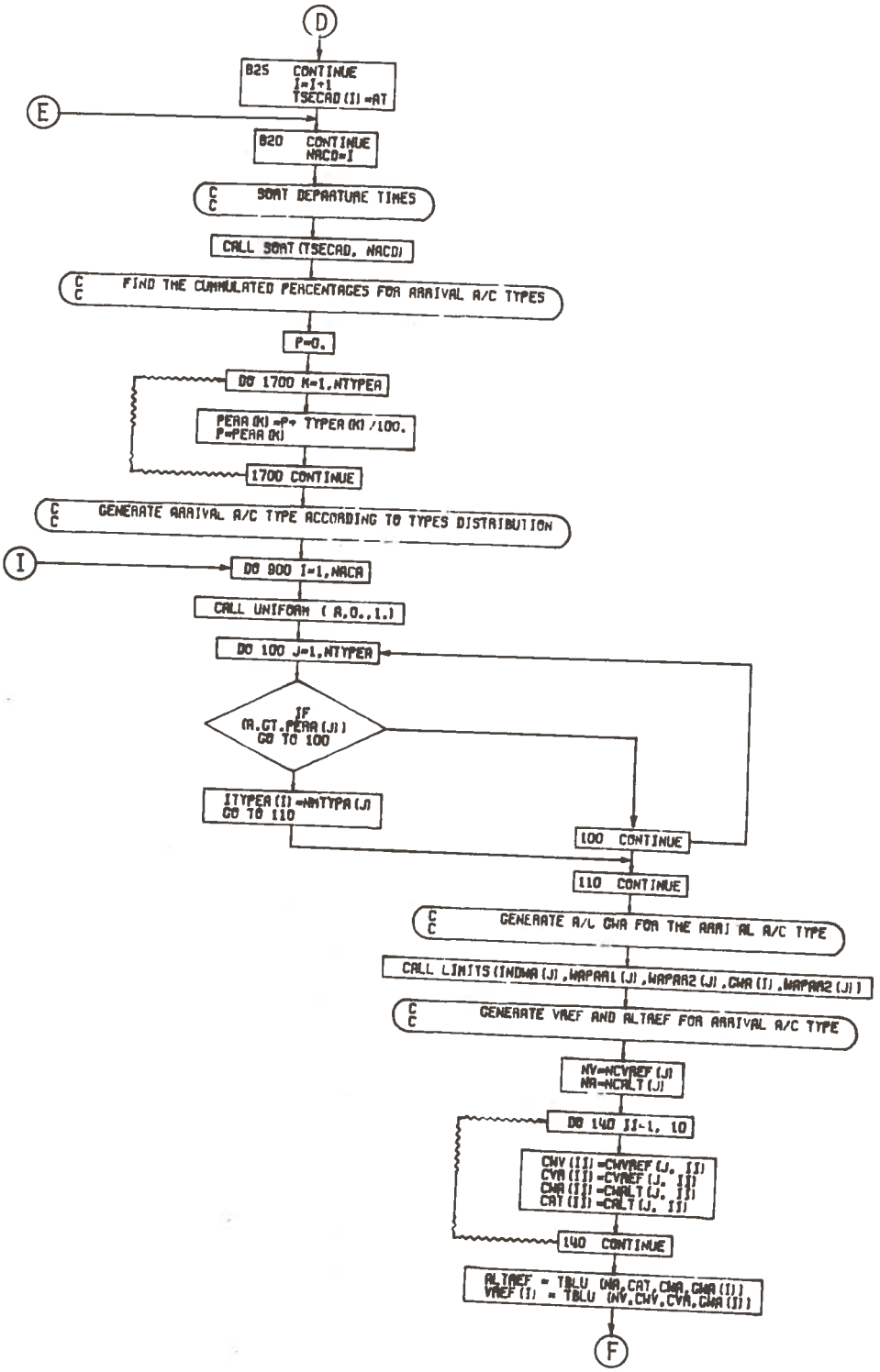
4.2.53.3 TRAFIC Variables

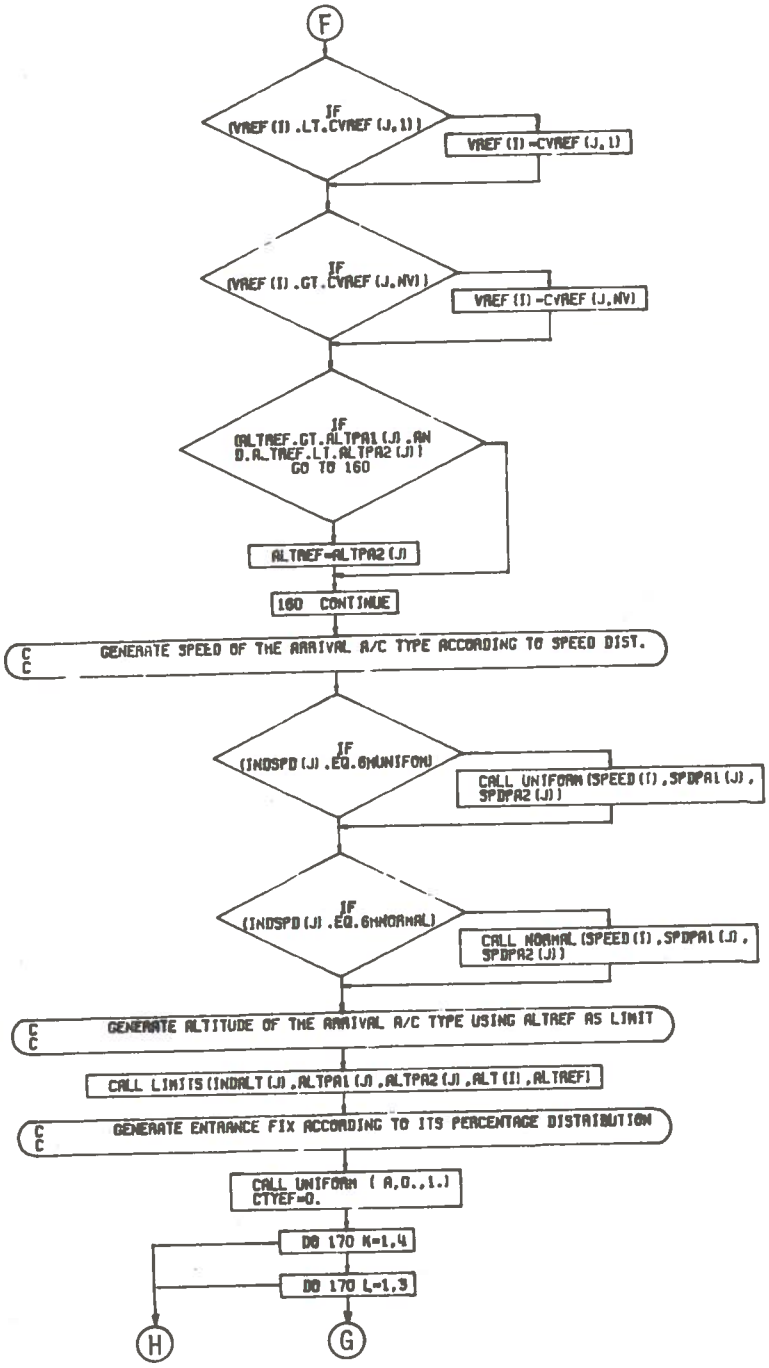
ALTREF	Limiting cruise altitude for particular airplane type
AT	Cumulative time in arrival/departure schedule
CTYEF	Cumulated percentage of a type of airplane for different entry points and all quadrants
DELTA	Difference in first arrival and first departure time
ITSEC	Integer form of time schedule for SORT routine input
NACAD	Total number of arrival and departure airplanes
NT	Maximum number of times to simulate the time schedule
PERA(K)	Cumulative percentage of the Kth type arrival airplane
PERD(K)	Cumulative percentage of the Kth type departure airplane
TT	Total time of scheduling (in seconds)
WTLA	Weight lower limit of a type of arrival airplane
WTLD	Weight lower limit of a type of departure airplane
WTUA	Weight upper limit of a type of arrival airplane
WTUD	Weight upper limit of a type of departure airplane

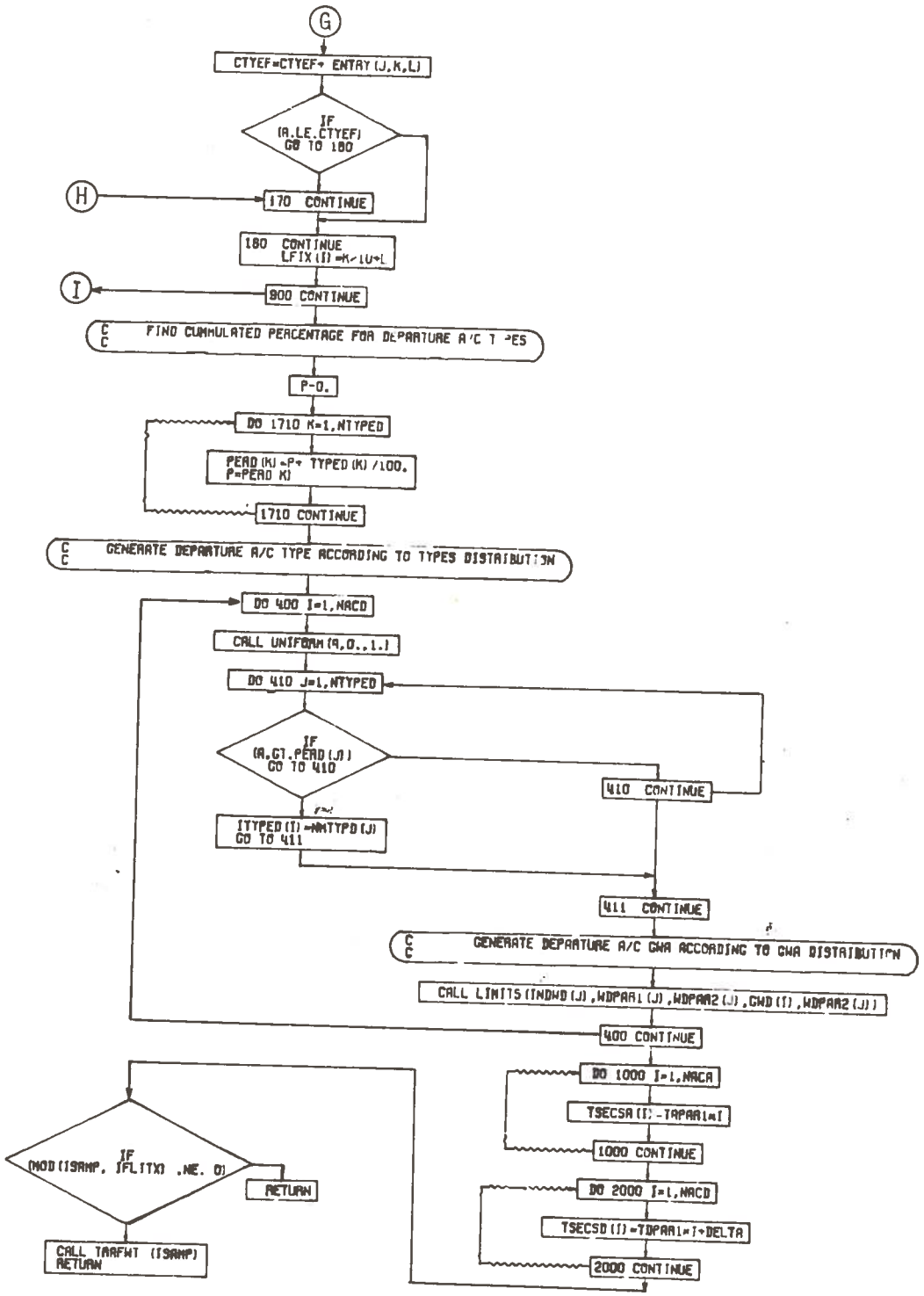

SUBROUTINE TRAFIC(IFLITX, ISAMP)
IF (SWTRAF.EQ.6.DISTR.) GO TO 9000
A
C DISCRETE TRAFFIC LIST SUPPLIED AS INPUT
C
DO 31 I = 1, NACA
CALL NORMAL (TSECAA(I), TSECSA(I), TASIGM)
CALL NORMAL (SPEED(I), SPEED(I) SPSIGM)
31 CONTINUE
CALL SORT(TSECAA, NACA)
DO 32 I = 1, NACD
CALL NORMAL (TSECAD(I), TSECSD(I), TDSIGM)
32 CONTINUE
CALL SORT (TSECAD,NACD)
DO 6600 I=1,NACA
DO 33 J 1,NTYPEA
IF (NNTYPA(J).NE.ITYPEA(I)) GO TO 33
33 CONTINUE
STOP 1
GO TO 35
35 CONTINUE
NV=NCVREF(J)
DO 7700 II=1,NV
CMV(II)=CMVREF(J,II)
CVA(II)=CVREF(J,II)
7700 CONTINUE
VREF(I) = TBLU(NV, CVA, CMV, GWA(I))
IF (VREF(I).LT.CVREF(J,1))
VREF(I)=CVREF(J,1)
IF (VREF(I).GT.CVREF(J,NV))
VREF(I)=CVREF(J,NV)
6600 CONTINUE
IF (MOD(ISAMP, IFLITX).NE.0)
RETURN
CALL TRAFWT (ISAMP)
RETURN


A
C DISTRIBUTION TRAFFIC LIST TO BE GENERATED FROM DISTRIBUTIONS INPUT TO PROGRAM
3000 CONTINUE
I=0
TT=3600.*HOUR
NT=TT/TAPAAL+50
C GENERATE ARRIVAL SCHEDULE
515 CONTINUE
AI=0.
DO 510 K=L,NT
TMEANA = TAPAAL*(FLOAT(K) - 1.)
IF (INDTA.EQ.6HNORMAL)
CALL NORMAL (AT,TMEANA,TAPAR2)
IF (INDTA.EQ.6HERLANG)
CALL ERLANG (AT,TMEANA,TAPAR2)
IF (AT.GT.TT) GO TO 520
IF (AT.EQ.TT) GO TO 525
I=I+1
TSECAA(I)=AT
510 CONTINUE
IF (AT.EQ.TT) GO TO 525
IF (AT.GT.TT) GO TO 520
NT=NT+50
GO TO 515
B
C


B
S25 CONTINUE
I=I+1
TSECAA(I)=AT
C
S20 CONTINUE
NACA=I
C SORT ARRIVAL TIMES
C
CALL SORT(TSECAA, NACA)
C GENERATE DEPARTURE SCHEDULE
C
NT=TT/TDPAR1+50
B15 CONTINUE
AT=0.
I=0
DO B10 K=1,NT
TMEAND = TDPAR1*(FLOAT(K) - 1.)
IF (INDTD.EQ.6HNORMAL)
CALL NORMAL (A,TMEAND,TDPAR2)
IF (INDTD.EQ.6HERLANG)
CALL ERLANG (A,TMEAND,TDPAR2)
AT = A + DELTA
IF (AT.GT.TT) GO TO B20
IF (AT.EQ.TT) GO TO B25
I=I+1
TSECAD(I)=AT
B10 CONTINUE
IF (AT.EQ.TT) GO TO B25
IF (AT.GT.TT) GO TO B20
NT=NT+50
GO TO B15
D
E


D
825 CONTINUE
I=I+1
TSECAD(I)=AT
E
820 CONTINUE
NACD=I
C SORT DEPARTURE TIMES
CALL SORT(TSECAD, NACD)
C FIND THE CUMMULATED PERCENTAGES FOR ARRIVAL A/C TYPES
P=0.
DO 1700 K=1,NTYPEA
PERA(K)=P+ TYPEA(K)/100.
P=PERA(K)
1700 CONTINUE
C GENERATE ARRIVAL A/C TYPE ACCORDING TO TYPES DISTRIBUTION
I
DO 900 I=1,NACA
CALL UNIFORM (A,0.,1.)
DO 100 J=1,NTYPEA
IF
(A.GT.PERA(J))
GO TO 100
ITYPEA(I)=NNTYPA(J)
GO TO 110
100 CONTINUE
110 CONTINUE
C GENERATE A/C CWA FOR THE ARRIVAL A/C TYPE
CALL LIMITS(INDWA(J),WAPAR1(J),WAPAR2(J),CWA(I),WAPAR2(J))
C GENERATE VREF AND ALTREF FOR ARRIVAL A/C TYPE
NV=NCVREF(J)
NA=NCALT(J)
DO 140 II=1, 10
CWV(II)=CWVREF(J, II)
CVA(II)=CVREF(J, II)
CWA(II)=CWALT(J, II)
CAT(II)=CALT(J, II)
140 CONTINUE
ALTREF = TBLU (NA,CAT,CWA,CWA(I))
VREF(I) = TBLU (NV,CWV,CVA,CWA(I))
F


F
IF (VREF(I).LT.CVREF(J,1))
VREF(I)=CVREF(J,1)
IF (VREF(I).GT.CVREF(J,NV))
VREF(I)=CVREF(J,NV)
IF (ALTREF.GT.ALTPA1(J).AND.ALTREF.LT.ALTPA2(J)) GO TO 160
ALTREF=ALTPA2(J)
160 CONTINUE
C GENERATE SPEED OF THE ARRIVAL A/C TYPE ACCORDING TO SPEED DIST.
IF (INDSPD(J).EQ.6HUNIFOM)
CALL UNIFORM(SPEED(I),SPDPA1(J),SPDPA2(J))
IF (INDSPD(J).EQ.6HNORMAL)
CALL NORMAL(SPEED(I),SPDPA1(J),SPDPA2(J))
C GENERATE ALTITUDE OF THE ARRIVAL A/C TYPE USING ALTREF AS LIMIT
CALL LIMITS(INDALT(J),ALTPA1(J),ALTPA2(J),ALT(I),ALTREF)
C GENERATE ENTRANCE FIX ACCORDING TO ITS PERCENTAGE DISTRIBUTION
CALL UNIFORM(A,0.,1.)
CTYEF=0.
DO 170 K=1,4
DO 170 L=1,3
H
G


G
CTYEF=CTYEF+ ENTRY(J,K,L)
IF (A.LE.CTYEF) GO TO 180
H
170 CONTINUE
180 CONTINUE
LFIX(I)=K+10*L
I
900 CONTINUE
C C FIND CUMMULATED PERCENTAGE FOR DEPARTURE A/C TYPES
P=0.
DO 1710 K=1,NTYPED
PERD(K)=P+TYPED(K)/100.
P=PERD(K)
1710 CONTINUE
C C GENERATE DEPARTURE A/C TYPE ACCORDING TO TYPES DISTRIBUTION
DO 400 I=1,NACD
CALL UNIFORM(R,0.,1.)
DO 410 J=1,NTYPED
IF (R.GT.PERD(J)) GO TO 410
410 CONTINUE
ITYPED(I)=NATYPD(J)
GO TO 411
411 CONTINUE
C C GENERATE DEPARTURE A/C GWA ACCORDING TO GWA DISTRIBUTION
CALL LIMITS(INDWD(J),WDPAR1(J),WDPAR2(J),GWD(I),WDPAR2(J))
400 CONTINUE
DO 1000 I=1,NACA
TSECSA(I)=TAPAR1+I
1000 CONTINUE
DO 2000 I=1,NACD
TSECSD(I)=TDPAR1+I+DELTA
2000 CONTINUE
IF (MOD(ISAMP,IFLITX).NE.0)
RETURN
CALL TARFWT (ISAMP)
RETURN

4.2.54 Subroutine TRAFWT

4.2.54.1 Abstract

Subroutine TRAFWT prints out the traffic list, which is broken into an arrival list and a departure list. The subroutine is called from TRAFIC for selected data samples.

4.2.54.2 Data Interface

TRAFWT Inputs

Through Calling Sequence:

ISAMP Sample number being printed

Through Common Statements:

/TRAFOT/ All variables except VBUGG, EELT, ELLT, KEY, SLT, SDT

TRAFWT Outputs

Through Printing:

ALT, IFIX, ITYPEA, SPEED, TSECSA, VREF, GWA, TSECAA, TSECSD, TSECAD, ITYPED, GWD

TRAFWT Calls

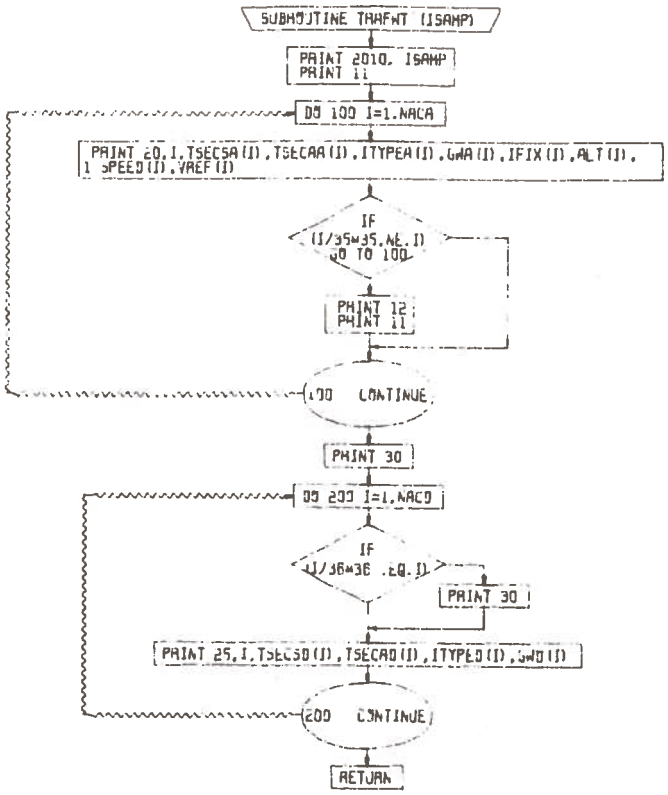
None

TRAFWT Called by

TRAFIC

4.2.54.3 TRAFWT Variables

None


SUBROUTINE TRAFNT (ISAMP)
PRINT 2010, ISAMP
PRINT 11
DO 100 I=1,NACA
PRINT 20,I,TSECSA(I),TSECAA(I),ITYPEA(I),GWA(I),IFIX(I),ALT(I),
1 SPEED(I),VREF(I)
IF
(I/35*35.NE.I)
GO TO 100
PRINT 12
PRINT 11
100 CONTINUE
PRINT 30
DO 200 I=1,NACD
IF
(I/36*36 .EQ.I)
PRINT 30
PRINT 25,I,TSECSD(I),TSECAD(I),ITYPED(I),GWD(I)
200 CONTINUE
RETURN

4.2.55 Subroutine TRKGOM

4.2.55.1 Abstract

Subroutine TRKGOM computes the distances between fix points and the flightpath track angles. The subroutine is called from GEOMTY once for each entry fix to threshold approach path and determines distances and angles working backward from the threshold out toward the entry fix.

4.2.55.2 Data Interfaces

TRKGOM Inputs

Through Common Statements:

/CNSTNT/	PI
/GEOMTX/	X, Y, RHO, NOFF, DOFF

TRKGOM Outputs

Through Common Statements:

/GEOMTX/	DSEG, THETA, DOFS

TRKGOM Calls

SQRT, ACOS

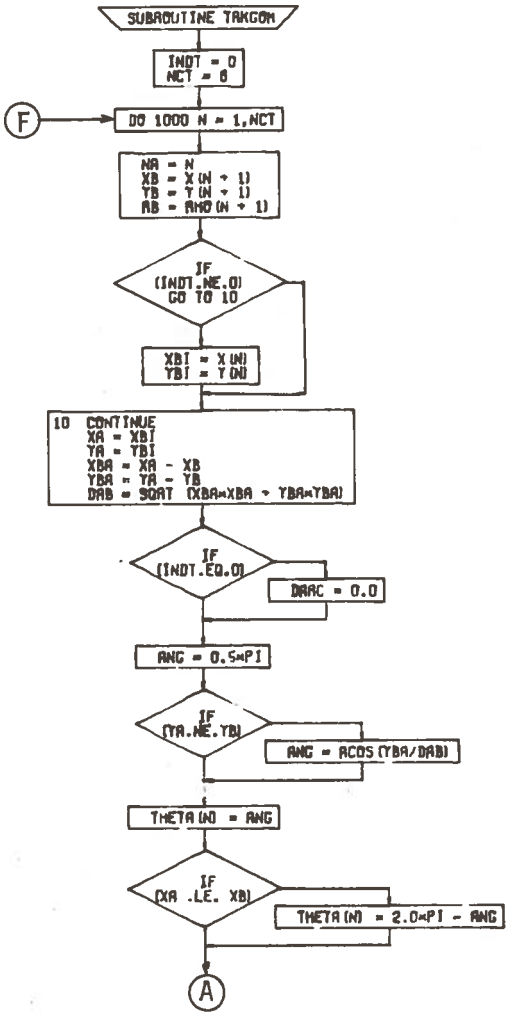
TRKGOM Called by

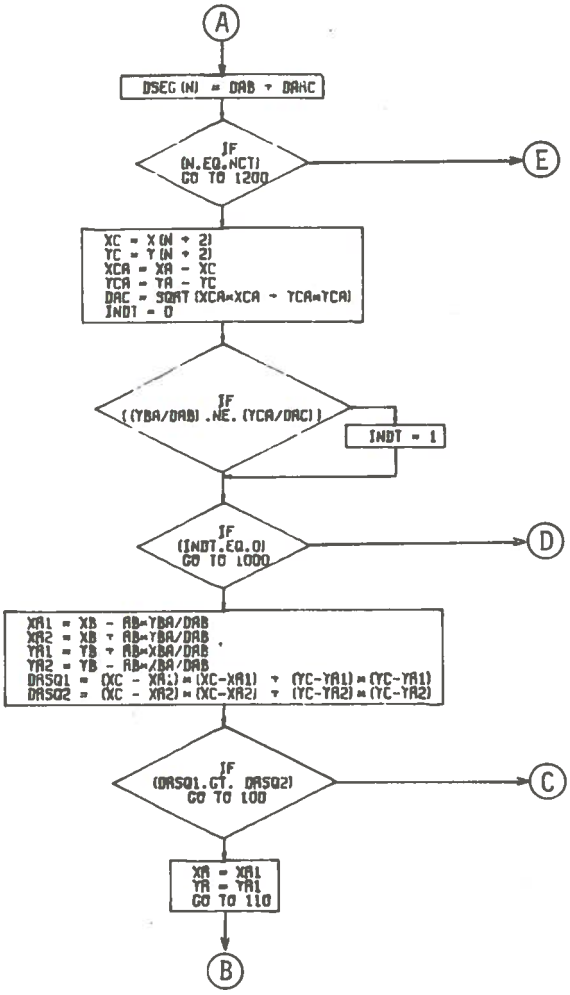
GEOMTY

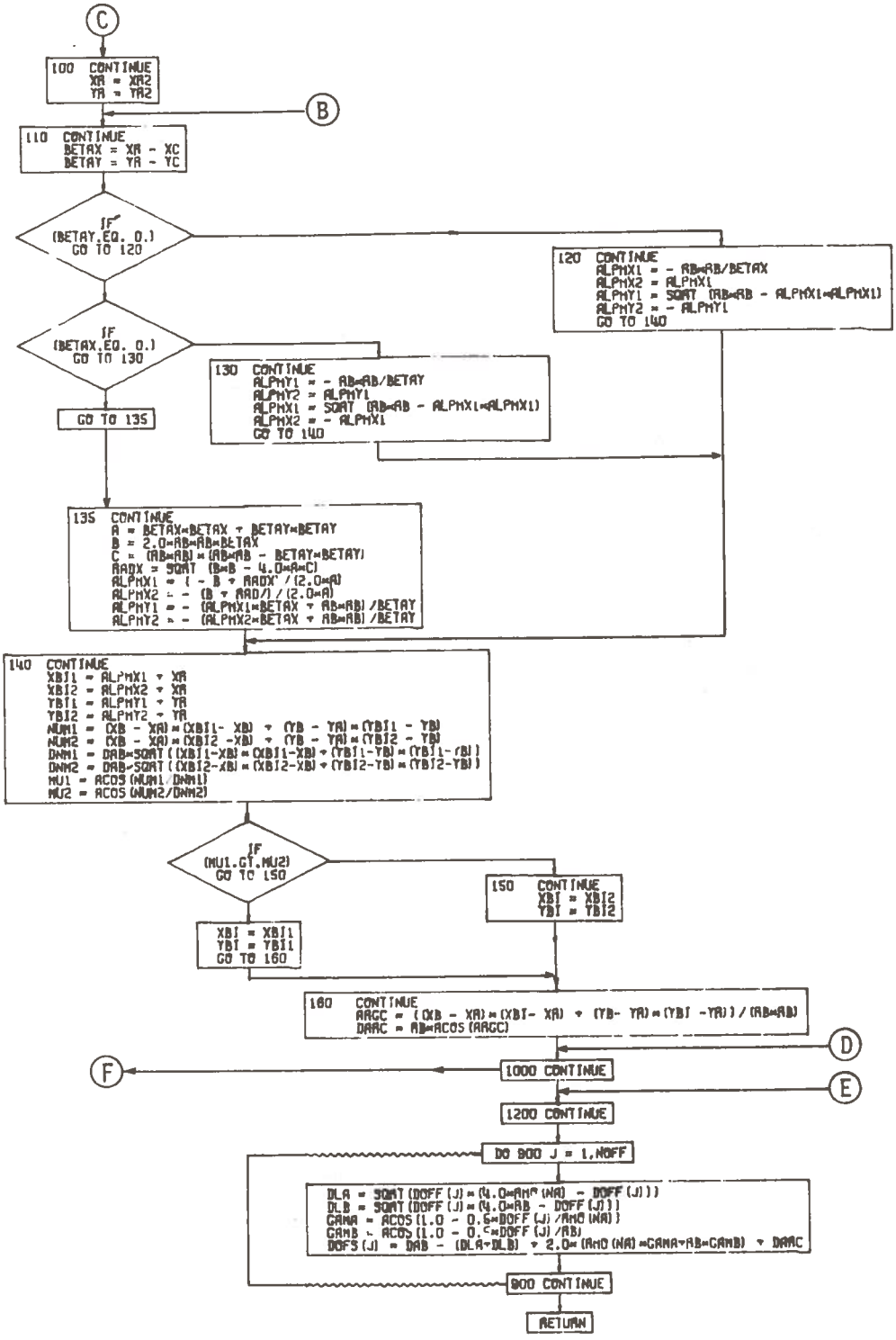
4.2.55.3 TRKGOM Variables

ANG	Path segment heading angle (in radians)
DAB	Distance (in nautical miles) along the flightpath (in the x,y plane) measured from the control fix to the turn entry point of the next control fix
DARC	Arc length of the turn circle flown by the airplane at a given control fix (in nautical miles)
INDT	Flag indicating whether three consecutive control fixes are colinear (INDT = 0 if fixes are colinear and = 1 otherwise)
NCT	Number of path segments between control fixes
RB	Turn radius (in nautical miles) at intermediate control fix
XA	The x-coordinate of the initial control fix under consideration
XL	The x-coordinate of the intermediate control fix under consideration
XBI	The x-coordinate of the turn entry point for the intermediate control fix

XC	The x-coordinate of the following control fix under consideration
XR	The x-coordinate of the center of the turn circle at the intermediate control fix
YA	The y-coordinate corresponding to XA
YB	The y-coordinate corresponding to XB
YBI	The y-coordinate corresponding to XBI
YC	The y-coordinate corresponding to XC
YR	The y-coordinate corresponding to XR




A
DSEG(N) = DAB * DAHC
IF
(N.EQ.NCT)
GO TO 1200
E
XC = X(N + 2)
YC = Y(N + 2)
XCA = XA - XC
YCA = YA - YC
DAC = SQRT(XCA*XCA + YCA*YCA)
INDT = 0
IF
((YBA/DAB) .NE. (YCA/DAC))
INDT = 1
IF
(INDT.EQ.0)
GO TO 1000
D
XA1 = XB - AB*YBA/DAB
XA2 = XB + AB*YBA/DAB
YA1 = YB + AB*XBA/DAB
YA2 = YB - AB*XBA/DAB
DASQ1 = (XC - XA1)*(XC-XA1) + (YC-YA1)*(YC-YA1)
DASQ2 = (XC - XA2)*(XC-XA2) + (YC-YA2)*(YC-YA2)
IF
(DASQ1.GT. DASQ2)
GO TO 100
C
XA = XA1
YA = YA1
GO TO 110
B


C
100 CONTINUE
XA = XA2
YA = YA2
B
110 CONTINUE
BETAX = XA - XC
BETAY = YA - YC
IF (BETAY.EQ. 0.) GO TO 120
120 CONTINUE
ALPHX1 = - AB*AB/BETAX
ALPHX2 = ALPHX1
ALPHY1 = SQRT (AB*AB - ALPHX1*ALPHX1)
ALPHY2 = - ALPHY1
GO TO 140
IF (BETAX.EQ. 0.) GO TO 130
130 CONTINUE
ALPHY1 = - AB*AB/BETAY
ALPHY2 = ALPHY1
ALPHX1 = SQRT (AB*AB - ALPHX1*ALPHX1)
ALPHX2 = - ALPHX1
GO TO 140
GO TO 135
135 CONTINUE
A = BETAX*BETAX + BETAY*BETAY
B = 2.0*AB*AB*BETAX
C = (AB*AB)*(AB*AB - BETAY*BETAY)
AADX = SQRT (B*B - 4.0*A*C)
ALPHX1 = (- B + AADX) / (2.0*A)
ALPHX2 = - (B + AADX) / (2.0*A)
ALPHY1 = (ALPHX1*BETAX + AB*AB) /BETAY
ALPHY2 = - (ALPHX2*BETAX + AB*AB) /BETAY
140 CONTINUE
XBI1 = ALPHX1 + XA
XBI2 = ALPHX2 + XA
YBI1 = ALPHY1 + YA
YBI2 = ALPHY2 + YA
ANM1 = (XB - XA)*(XBI1- XB) + (YB - YA)*(YBI1 - YB)
ANM2 = (XB - XA)*(XBI2 -XB) + (YB - YA)*(YBI2 - YB)
DAN1 = DAB*SQRT ((XBI1-XB)*(XBI1-XB)+(YBI1-YB)*(YBI1-YB))
DAN2 = DAB*SQRT ((XBI2-XB)*(XBI2-XB)+(YBI2-YB)*(YBI2-YB))
MU1 = ACOS (ANM1/DAN1)
MU2 = ACOS (ANM2/DAN2)
IF (MU1.GT.MU2) GO TO 150
150 CONTINUE
XBI = XBI2
YBI = YBI2
XBI = XBI1
YBI = YBI1
GO TO 160
160 CONTINUE
ARGC = (XB - XA)*(XBI- XA) + (YB- YA)*(YBI -YA) / (AB*AB)
DARC = AB*ACOS (ARGC)
D
F
1000 CONTINUE
E
1200 CONTINUE
DO 800 J = 1,NOFF
DLA = SQRT (DOFF (J)*(4.0*AM*(NA) - DOFF (J)))
DLB = SQRT (DOFF (J)*(4.0*AB - DOFF (J)))
GAMA = ACOS (1.0 - 0.5*DOFF (J) /AMO (NA))
GAMB = ACOS (1.0 - 0.5*DOFF (J) /AB)
DOFS (J) = DAB - (DLA+DLB) + 2.0*(AMO (NA)*GAMA+AB*GAMB) + DARC
800 CONTINUE
RETURN

4.2.56 Subroutine UNIFORM

4.2.56.1 Abstract

Subroutine UNIFORM generates uniformly distributed random numbers. One value is generated per call.

4.2.56.2 Data Interface

UNIFORM Inputs

Through Calling Sequence:

P1	Upper limit of the uniform distribution
P2	Lower limit of the uniform distribution
IU	Starter of the random-number generator
IX	Multiplier of the seed of the random-number generator

UNIFORM Outputs

Through Calling Sequence:

A	The uniformly distributed random variate generated

UNIFORM Calls

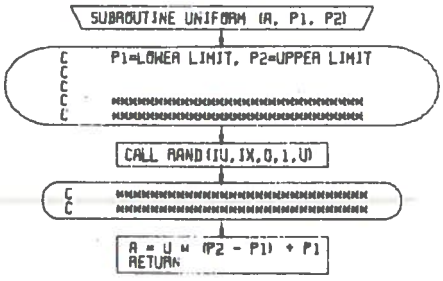
RAND	Boeing math library routine

UNIFORM Called by

TRAFIC, LIMITS

4.2.56.3 UNIFORM Variables

U	Uniformly distributed random number generated by RAND on the interval [0, 1]



4.2.57 Subroutine WETHER

4.2.57.1 Abstract

Subroutine WETHER provides a data bank containing wind, temperature, and pressure information stored as a function of altitude. Subroutine WETHER is called at an altitude and determines the north and east wind components (forecast or actual) and temperature (forecast or actual). WETHER, in turn, calls ATMOSP, which provides further meteorological data.

4.2.57.2 Data Interfaces

WETHER Inputs

Through Calling Sequence:

H	Altitude (in thousands of feet) for which the weather data are desired
ISW	Logic switch for actual or forecast weather. ISW = 0 gives forecast weather and ISW = 1 gives actual weather.

Through Common Statements:

/TEMPER/	GTEMP1, GTEMP2, GTEMP3, HTEMP0, HTEMP1, HTEMP2, TEMP1, TEMP2, TEROR1, TEROR2, TEROR3, TGROND
/WIND/	GWINE1, GWINE2, GWINE3, GWINN1, GWINN2, GWINN3, HWIND0, HWIND1, HWIND2, WERRE1, WERRE2, WERRE3, WERRN1, WERRN2, WERRN3, WINDE0, WINDE1, WINDE2, WINDN0, WINDN1, WINDN2
/CNSTNT/	PI

WETHER Outputs

Through Common Statements:

/TEMPER/	TEMP, TSTAND
/WIND/	WINANG, WINDE, WINDN, WINMAG

WETHER Calls

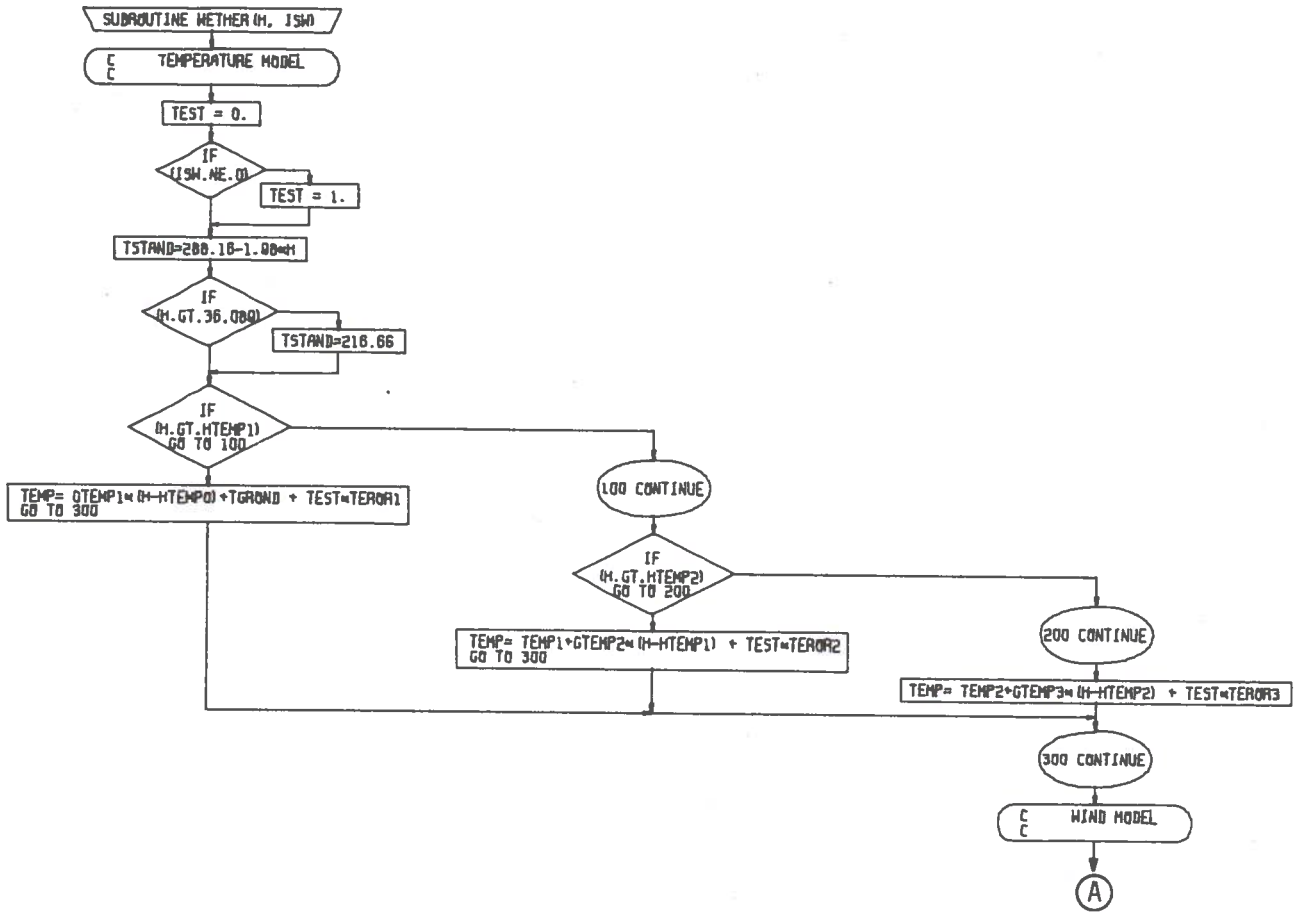
ATAN2	FORTRAN library function
SQRT	FORTRAN library function
ATMOSP	Computes density, pressure, and temperature ratios

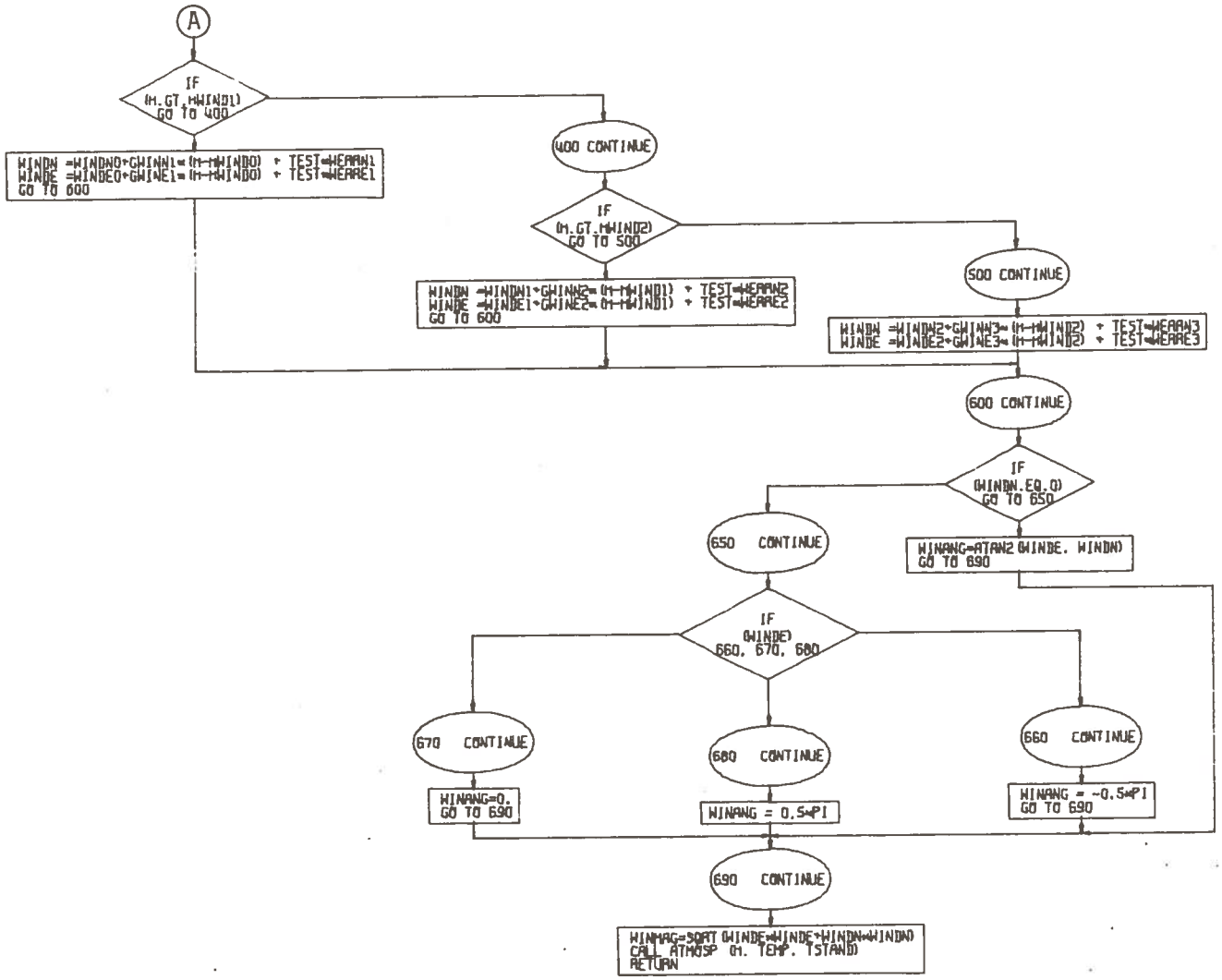
WETHER Called by

STRAD, AERO1, TOMIAF, AERO, REFBUG, SEQUEN, ERROR, TASCAS, CASTAS

4.2.57.3 WETHER Variables

None


SUBROUTINE WETHER(H, ISW)
C
C TEMPERATURE MODEL
TEST = 0.
IF
(ISW.NE.0)
TEST = 1.
TSTAND=288.16-1.98*H
IF
(H.GT.36.089)
TSTAND=216.66
IF
(H.GT.HTEMP1)
GO TO 100
TEMP= QTEMP1*(H-HTEMP0)+TGROND + TEST*TERGR1
GO TO 300
100 CONTINUE
IF
(H.GT.HTEMP2)
GO TO 200
TEMP= TEMP1+GTEMP2*(H-HTEMP1) + TEST*TERGR2
GO TO 300
200 CONTINUE
TEMP= TEMP2+GTEMP3*(H-HTEMP2) + TEST*TERGR3
300 CONTINUE
C
C WIND MODEL
A


A
IF (M.GT.MWIND1) GO TO 400
WINDN =WINDNO*GWINN1+(1.-GWINN1)*(M.-MWIND0) * TEST*HERRN1
WINDE =WINDEO*GWINE1+(1.-GWINE1)*(M.-MWIND0) * TEST*HERRE1
GO TO 600
400 CONTINUE
IF (M.GT.MWIND2) GO TO 500
WINDN =WINDN1*GWINN2+(1.-GWINN2)*(M.-MWIND1) * TEST*HERRN2
WINDE =WINDE1*GWINE2+(1.-GWINE2)*(M.-MWIND1) * TEST*HERRE2
GO TO 600
500 CONTINUE
WINDN =WINDN2*GWINN3+(1.-GWINN3)*(M.-MWIND2) * TEST*HERRN3
WINDE =WINDE2*GWINE3+(1.-GWINE3)*(M.-MWIND2) * TEST*HERRE3
600 CONTINUE
IF (WINDN.EQ.0) GO TO 650
650 CONTINUE
WINANG=ATAN2(WINDE, WINDN)
GO TO 690
IF (WINDE) 660, 670, 680
670 CONTINUE
WINANG=0.
GO TO 690
680 CONTINUE
WINANG = 0.5*PI
660 CONTINUE
WINANG = -0.5*PI
GO TO 690
690 CONTINUE
WINMAG=SQRT(WINDE*WINDE+WINDN*WINDN)
CALL ATMOSP(H, TEMP, TSTAND)
RETURN

4.2.58 Subroutine WTHTST

4.2.58.1 Abstract

For each 1000 feet of altitude, subroutine WTHTST prints out the forecast and actual weather input to the program.

4.2.58.2 Data Interfaces

WTHTST Inputs

Through Common Statements:

/TEMPER/	TEMP
/WIND/	WINDE, WINDN

WTHTST Outputs

Through Printing:

H, TFOCAS, TEMP, WSPED, WINDNF, WINDEF, WINDN, WINDE

WTHTST Calls

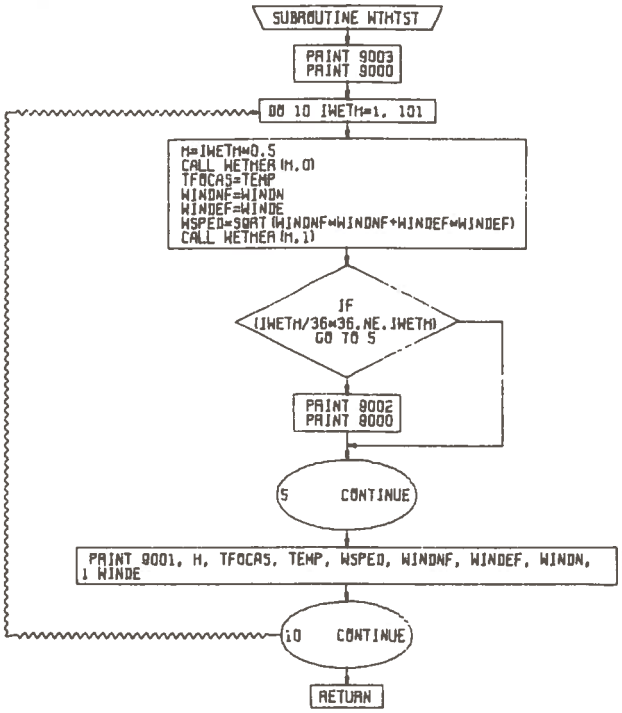
WETHER	
SQRT	FORTRAN library function

WTHTST Called by

STRAD

4.2.58.3 WTHTST Variables

H	Altitude (in thousands of feet)
TFOCAS	Forecast temperature (in degrees Kelvin) at altitude H
TEMP	Actual temperature (in degrees Kelvin) at altitude H
WSPED	Forecast windspeed (in knots) at altitude H
WINDNF	Forecast north wind component speed (in knots) at altitude H
WINDEF	Forecast east wind component speed (in knots) at altitude H
WINDN	Actual north wind component speed (in knots) at altitude H
WINDE	Actual east wind component speed (in knots) at altitude H


SUBROUTINE WTHTST
PRINT 9003
PRINT 9000
DO 10 IWETH=1, 101
H=IWETH*0.5
CALL WETHER(H,0)
TFOCAS=TEMP
WINDNF=WINDN
WINDEF=WINDE
WSPED=SQRT(WINDNF*WINDNF+WINDEF*WINDEF)
CALL WETHER(H,1)
IF
(IWETH/36*36.NE.IWETH)
GO TO 5
PRINT 9002
PRINT 9000
5 CONTINUE
PRINT 9001, H, TFOCAS, TEMP, WSPED, WINDNF, WINDEF, WINDN,
1 WINDE
10 CONTINUE
RETURN

4.2.59 Subroutine WTOUT

4.2.59.1 Abstract

Subroutine WTOUT formats and outputs the results of the statistical samples generated by the evaluation model. The output is divided into a summary (operation rates, safety, and delays) and tables of operational statistics (interoperation times, time and distance separations, path times, and delay times).

4.2.59.2 Data Interfaces

WTOUT Inputs

Through Common Statements:

/STATIN/	NDLY, NDSEP, NOPSTP, NTARR, NTSEP
/STATOT/	All variables except TOTAL

WTOUT Outputs

Through Printing:

IOFSET array (stored as data)	
/STATOT/	All variables except TOTAL

WTOUT Calls

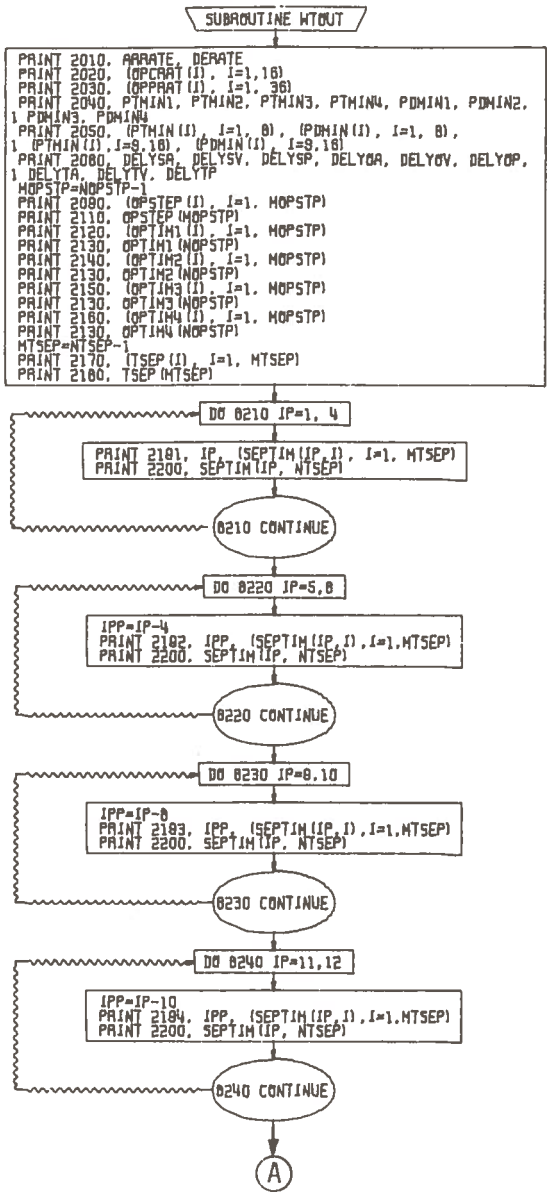
None

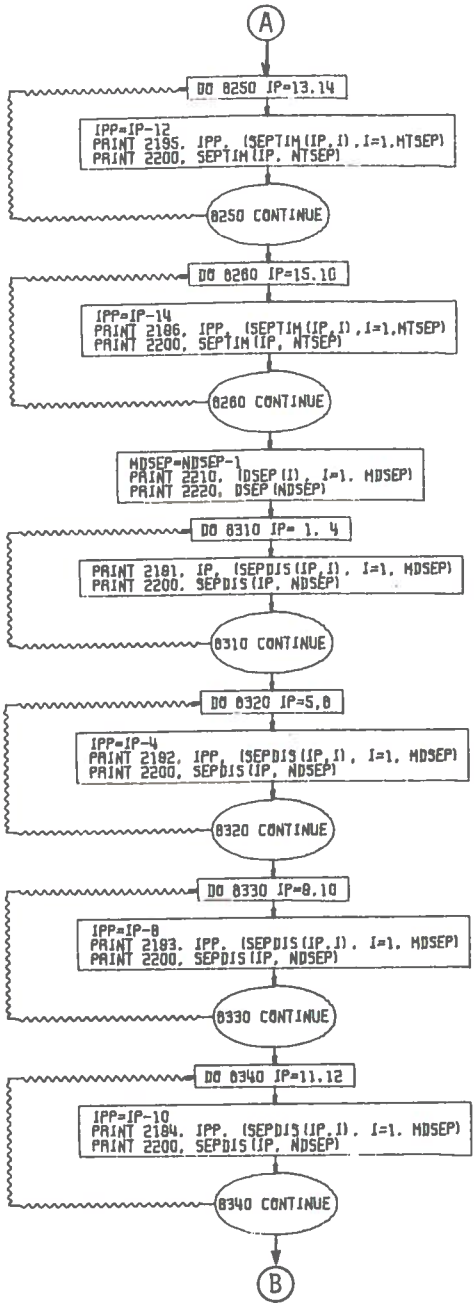
WTOUT Called by

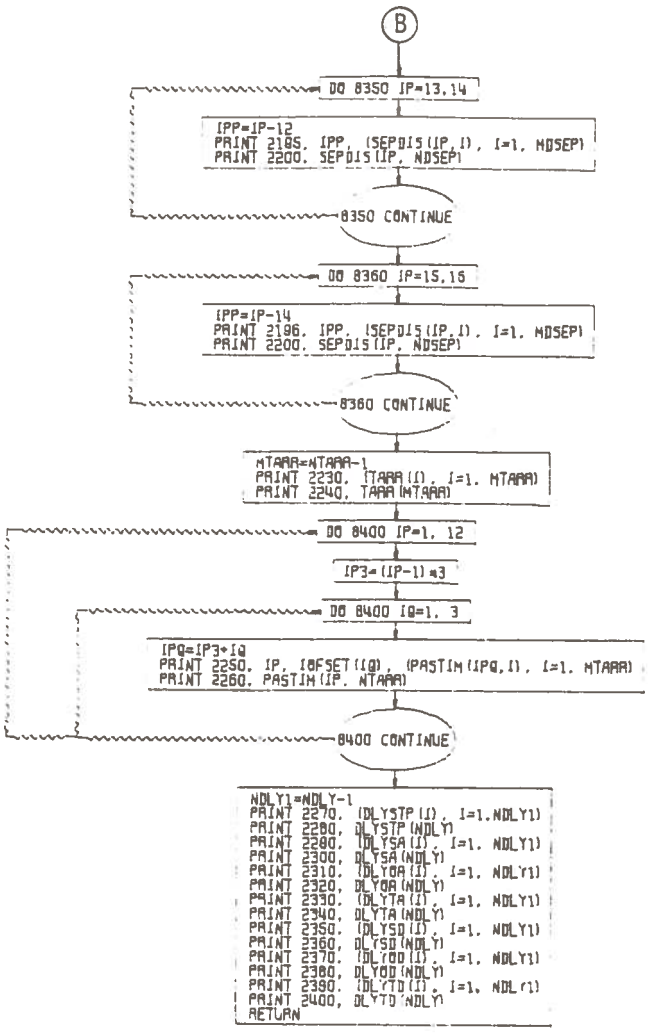
STRAD

4.2.59.3 WTOUT Variables

IOFSET (3)


SUBROUTINE WTOUT
PRINT 2010, AARATE, DERATE
PRINT 2020, (QPCRAT(I), I=1,16)
PRINT 2030, (QPPRAT(I), I=1, 36)
PRINT 2040, PTMIN1, PTMIN2, PTMIN3, PTMIN4, PDMIN1, PDMIN2,
1 PDMIN3, PDMIN4
PRINT 2050, (PTMIN(I), I=1, 8), (PDMIN(I), I=1, 8),
1 (PTMIN(I),I=9,16), (PDMIN(I), I=9,16)
PRINT 2080, DELYSA, DELYSV, DELYSP, DELYDA, DELYDV, DELYDP,
1 DELYTA, DELYTV, DELYTP
NOPSTP=NOPSTP-1
PRINT 2090, (QPSTEP(I), I=1, NOPSTP)
PRINT 2110, QPSTEP(NOPSTP)
PRINT 2120, (QPTIM1(I), I=1, NOPSTP)
PRINT 2130, QPTIM1(NOPSTP)
PRINT 2140, (QPTIM2(I), I=1, NOPSTP)
PRINT 2130, QPTIM2(NOPSTP)
PRINT 2150, (QPTIM3(I), I=1, NOPSTP)
PRINT 2130, QPTIM3(NOPSTP)
PRINT 2160, (QPTIM4(I), I=1, NOPSTP)
PRINT 2130, QPTIM4(NOPSTP)
NTSEP=NTSEP-1
PRINT 2170, (TSEP(I), I=1, NTSEP)
PRINT 2180, TSEP(NTSEP)
DO 8210 IP=1, 4
PRINT 2101, IP, (SEPTIM(IP,I), I=1, NTSEP)
PRINT 2200, SEPTIM(IP, NTSEP)
8210 CONTINUE
DO 8220 IP=5,8
IPP=IP-4
PRINT 2102, IPP, (SEPTIM(IP,I),I=1,NTSEP)
PRINT 2200, SEPTIM(IP, NTSEP)
8220 CONTINUE
DO 8230 IP=8,10
IPP=IP-8
PRINT 2103, IPP, (SEPTIM(IP,I),I=1,NTSEP)
PRINT 2200, SEPTIM(IP, NTSEP)
8230 CONTINUE
DO 8240 IP=11,12
IPP=IP-10
PRINT 2104, IPP, (SEPTIM(IP,I),I=1,NTSEP)
PRINT 2200, SEPTIM(IP, NTSEP)
8240 CONTINUE
A


A
DO 8250 IP=13,14
IPP=IP-12
PRINT 2195, IPP, (SEPTIM(IP,I),I=1,MTSEP)
PRINT 2200, SEPTIM(IP, MTSEP)
8250 CONTINUE
DO 8260 IP=15,16
IPP=IP-14
PRINT 2196, IPP, (SEPTIM(IP,I),I=1,MTSEP)
PRINT 2200, SEPTIM(IP, MTSEP)
8260 CONTINUE
MDSEP=NDSEP-1
PRINT 2210, (DSEP(I), I=1, MDSEP)
PRINT 2220, DSEP(NDSEP)
DO 8310 IP= 1, 4
PRINT 2181, IP, (SEPDIS(IP,I), I=1, MDSEP)
PRINT 2200, SEPDIS(IP, NDSEP)
8310 CONTINUE
DO 8320 IP=5,8
IPP=IP-4
PRINT 2182, IPP, (SEPDIS(IP,I), I=1, MDSEP)
PRINT 2200, SEPDIS(IP, NDSEP)
8320 CONTINUE
DO 8330 IP=8,10
IPP=IP-8
PRINT 2183, IPP, (SEPDIS(IP,I), I=1, MDSEP)
PRINT 2200, SEPDIS(IP, NDSEP)
8330 CONTINUE
DO 8340 IP=11,12
IPP=IP-10
PRINT 2184, IPP, (SEPDIS(IP,I), I=1, MDSEP)
PRINT 2200, SEPDIS(IP, NDSEP)
8340 CONTINUE
B


B
DO 8350 IP=13,14
IPP=IP-12
PRINT 2185, IPP, (SEPDIS(IP,I), I=1, MDSEP)
PRINT 2200, SEPDIS(IP, NDSEP)
8350 CONTINUE
DO 8360 IP=15,16
IPP=IP-14
PRINT 2196, IPP, (SEPDIS(IP,I), I=1, MDSEP)
PRINT 2200, SEPDIS(IP, NDSEP)
8360 CONTINUE
MTARR=MTARR-1
PRINT 2230, (TARR(I), I=1, MTARR)
PRINT 2240, TARR(MTARR)
DO 8400 IP=1, 12
IP3=(IP-1)*3
DO 8400 IQ=1, 3
IPQ=IP3+IQ
PRINT 2250, IP, IOFSET(IQ), (PASTIM(IPQ,I), I=1, MTARR)
PRINT 2260, PASTIM(IP, MTARR)
8400 CONTINUE
NDLY1=NDLY-1
PRINT 2270, (DLYSTP(I), I=1,NDLY1)
PRINT 2280, DLYSTP(NDLY)
PRINT 2290, (DLYSA(I), I=1, NDLY1)
PRINT 2300, DLYSA(NDLY)
PRINT 2310, (DLYOA(I), I=1, NDLY1)
PRINT 2320, DLYOA(NDLY)
PRINT 2330, (DLYTA(I), I=1, NDLY1)
PRINT 2340, DLYTA(NDLY)
PRINT 2350, (DLYSD(I), I=1, NDLY1)
PRINT 2360, DLYSD(NDLY)
PRINT 2370, (DLYOD(I), I=1, NDLY1)
PRINT 2380, DLYOD(NDLY)
PRINT 2390, (DLYTD(I), I=1, NDLY1)
PRINT 2400, DLYTD(NDLY)
RETURN

4.3 PROGRAM LABELED ERROR STOPS

The only nonlabeled stop in the STRAD program occurs after all data sets have been processed in the main program. All other stop conditions are noted (STOP1 through STOP7) when incurred. The STOP conditions are defined below.

STOP1 Occurs in subroutine TRAFIC when the discrete input option is used. If the type (ITYPEA) of the Ith arrival airplane cannot be found in the type list, the arrival cannot be processed.

STOP2 Occurs in subroutine LIMITS when the input distribution type is not recognized. The three acceptable distribution types are (spelling as recognized by the program) UNIFOM, NORMAL, and ERLANG.

STOP3 Occurs in subroutine AERO under the same conditions as STOP1.

STOP4 Occurs in subroutine TIMFND when more than seven waypoints are found between the entry fix and the intermediate approach fix. Generation of waypoints occurs in STUFF1, STUFF2, and STUFF3.

STOP5 Occurs in subroutine RTP when more than 14 waypoints and control fixes are found.

STOP6 Occurs in subroutine TFIND and denotes the same condition as STOP4.

STOP7 Occurs in subroutine ERROR when the seventh control fix (designating the runway threshold) cannot be found.

4.4 MACHINE DEPENDENCIES

4.4.1 Random-Number Generation

Random-number generation is machine dependent and is frequently installation dependent. Therefore, it will become necessary to convert the random-number generation scheme used.

RAND, the routine used on the Boeing Computer Services' CDC 6600 computers, requires two seeds to generate random numbers. These seeds are read into the program by subroutines RDDSCR and RDDSTR and are stored in COMMON/RANDOM/. The seed variables, IU and IX, are the fifth and sixth entries of the third data card read in for the discrete input data cases, and the fourth and fifth data entries of the third card for the distribution input data cases.

Table 4-5 lists the subroutines that use RAND and COMMON/RANDOM/. In the program listing, calls to RAND are boxed for ease of location.

TABLE 4-5 -MACHINE DEPENDENCIES

Subroutine names	RAND called	COMMON/RANDOM/ appears
ERLANG	X	X
ERROR	X	X
NORMAL	X	X
RDDSCR		X
RDDSTR		X
UNIFORM	X	X

The calling sequence of RAND is RAND(IU, IX, K, N, U), where IU and IX are the seeds from COMMON/RANDOM/. K is an indicator of the distribution type. K = 0 provides a uniform distribution on the interval (0, 1). K ≠ 0 provides a normal distribution with mean 1.0 and variance 1.0. N is the number of random numbers desired. U is an array of the N random numbers generated.

4.4.2 Program Storage Requirements

Program storage will vary from machine to machine depending on its design and system routine requirements. The data presented in this section reflect the storage requirements for a Control Data Corporation 6600 utilizing the KRONOS operating system. The CDC 6600 is a fixed-word-length machine with a 60-bit word. No double precision variables are used. The 60-bit word allows 10 BCD characters to be stored per word for those variables requiring alphanumeric data storage. No alphanumeric information is stored that requires more than 10 characters.

The STRAD program requires 46,592 decimal (133,000 octal) words, including all system software. The program less the system and library routines requires 42,877 decimal (123,575 octal) words of central memory. The FORTRAN library routines such as COS, SQRT, etc., require 487 decimal (747 octal) words. The remaining storage—3228 decimal (6234 octal) words—is required for system routines such as input/output software.

Common data storage used in STRAD requires 21,858 decimal (52,542 octal) words. Storage of instructions and input/output buffers requires 12,518 decimal (30,346 octal) words. Temporary and local variables, constants, and formats account for another 30,359 decimal (73,227 octal) words.

If any reduction in storage is required for the STRAD program, a decrease in the common data storage can be implemented by decreasing the maximum number of arrivals and departures (200 and 100, respectively) to be processed per data sample. Such large data commons as /TRAFOT/ and /RTPOUT/—with 3102 and 14,000 decimal words, respectively—could be decreased, depending on the maximum numbers permitted. Other savings could be made by compacting format statements, reducing other common variable maximum dimensions, or by resetting buffer sizes.

5.0 STRATEGIC ALGORITHM EVALUATION MODEL LISTING

A complete listing of the strategic algorithm evaluation model FORTRAN program as written for the CDC 6600 computer is given in table 5-1. The listing begins with the main program, STRAD, followed by the various program subroutines in alphabetical order.

TABLE 5-1.—PROGRAM LISTING

```
FUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE AERO (INDX,  ITHAC, IQUAD, INUMB, ISW)                00018250
      C                                                                          00018260
                                                                                 00018270
000010          COMMON /AEROIN/                                                  00018280
               1    CAERS(10),     HA(10,7),      HCRITN(10),    NAERO(10),     00018290
               2    VFAST1(10,7),  VFAST2(10,7),  VSLCW1(10,7),  VSLCW2(10,7),  00018300
               3    W1(10),        W2(10)                                       00018310
000010          COMMON /AEROOT/                                                  00018320
               1    HB(10),        HCRIT,         JMAX,          SPKNOT,        00018330
               2    VFAST(10),     VSLCW(10)                                    00018340
000010          COMMON /GEOMOT/                                                  00018350
               1    ANGEF(4,3),    ANGFAF(2),     ANGIAF(4),     ANGMF(2),      00018360
               2    ANGCM(2),      ANGTF(4),      DEF(4,3,5),    DFAF(2),       00018370
               3    DIAF(4),       DMF(2),        DCM(2),        DFATH(4,3,5),  00018380
               4    DTF(4),        DTH(2)                                       00018390
000010          COMMON /TEMPER/                                                  00018400
               1    GTEMP1,        GTEMP2,        GTEMP3,        HTEMPO,        00018410
               2    HTEMP1,        HTEMP2,        TEMP,          TEMP1,         00018420
               3    TEMP2,         TGROND,        TEROR1,        TEROR2,        00018430
               4    TEROR3,        TSTAND                                       00018440
000010          COMMON /TRAFIN/                                                  00018450
               1    ALTFA1(10),    ALTFA2(10),    CALT (10,10),  CVREF(10,10),  00018460
               2    CWALT(10,10),  CWVREF(10,10), ENTRY(10,4,3), HCUR,          00018470
               3    INDALT(10),    INDSPD(10),    INDTA,         INDTD,         00018480
               4    INDWA(10),     INDWD(10),     NCALT(10),     NCVREF(10),    00018490
               5    NMTYFA(10),    NMTYPD(10),    NTYPEA,        NTYPED,        00018500
               6    SPDFA1(10),    SPDFA2(10),    SPSIGM,        SWTRAF,        00018510
               7    TAFAR1,        TAFAR2,        TASIGM,        TDFAR1,        00018520
               8    TDFAR2,        TDSIGM,        TYPEA(10),     TYPED(10),     00018530
               9    WAFAR1(10),    WAFAR2(10),    WDFAR1(10),    WDFAR2(10)     00018540
000010          INTEGER            SWTRAF                                       00018550
000010          COMMON /TRAFOT/                                                  00018560
               1    ALT(200),      EELT(200),     ELLT(200),     GWA(200),      00018570
               2    GWD(100),      ITYPEA(200),   ITYPED(100),   KEY(200),      00018580
               3    LFIX(200),     NACA,          NACD,          SDT(100),      00018590
               4    SLT(200),      SPEED(200),    TSECAA(200),   TSECAD(100),   00018600
               5    TSECSA(200),   TSECSD(100),   VDUGG(200),    VREF(200)      00018610
000010          COMMON /WIND/                                                    00018620
               1    GWINE1,        GWINE2,        GWINE3,        GWINN1,        00018630
               2    GWINN2,        GWINN3,        HWIND0,        HWIND1,        00018640
               3    HWIND2,        WERRE1,        WERRE2,        WERRE3,        00018650
               4    WERRN1,        WERRN2,        WERRN3,        WINANG,        00018660
               5    WINDE,         WINDE0,        WINDE1,        WINDE2,        00018670
               6    WINDN,         WINDN0,        WINDN1,        WINDN2,        00018680
               7    WINMAG                                                      00018690
000010          DIMENSION          DVFAST(10),    DVSLCW(10)                    00018700
      C                                                                          00018710
000010          GW=GWA(ITHAC)                                                    00018720
000012          H=ALT(ITHAC)                                                     00018730
000013          INDX=0                                                           00018740
000014          DO 100 K=1, NTYPEA                                               00018750
000015             IF (ITYPEA(ITHAC).NE.NMTYPA(K)) GO TO 100                     00018760
000020                INDX=K
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

000020                GO TO 120                                          00018770
000021     100 CONTINUE                                                  00018780
000024         PRINT 2010, ITYPEA(ITHAC)                                 00018790
000034     2010 FORMAT(1HO, 34H *** CAUTION *** THE A/C TYPE OF (, A4,   00018800
                1       49H  HAS NO MATCH IN THE INPUT ARRIVAL A/C TYPE LIST/1HO)  00018810
000034         STOP 3                                                    00018820
               C                                                         00018830
               C     SET CRITICAL MACH TRANSITION ALTITUDE FOR THIS AIRCRAFT TYPE  00018840
               C     (AT INDX)                                           00018850
               C                                                         00018860
000036     120 CONTINUE                                                  00018870
000036         HCRIT = HCRITN(INDX)                                      00018880
               C                                                         00018890
               C     IF GROSS WEIGHT WITHIN BOUNDS                       00018900
               C                                                         00018910
000043         IF (GW.LT.W1(INDX)) 150, 160                              00018920
000050   150      CONTINUE                                               00018930
000050            GW = W1(INDX)                                          00018940
000052            GO TO 200                                              00018950
000052   160   CONTINUE                                                  00018960
000052         IF (GW.GT.W2(INDX)) GW = W2(INDX)                         00018970
000056   200   CONTINUE                                                  00018980
               C                                                         00018990
               C     IF ARRIVAL ALTITUDE WITHIN BOUNDS                   00019000
               C                                                         00019010
000056         IMAX=NAERO(INDX)                                          00019020
000060         IF (H - HA(INDX,IMAX)) 240, 250, 230                      00019030
000064   230         CONTINUE                                            00019040
000064               H = HA(INDX,IMAX)                                   00019050
000070               GO TO 250                                           00019060
000070   240      CONTINUE                                               00019070
000070            IF (H.LT.12.0) H = 12.0                                00019080
000074   250   CONTINUE                                                  00019090
000074         WR=(GW-W1(INDX))/(W2(INDX)-W1(INDX))                      00019100
000101         WR1=1.0-WR                                                00019110
               C                                                         00019120
               C     BUILD AERO ENVELOPE FOR THE GW AND H OF THIS A/C    00019130
               C                                                         00019140
000103         DO 260 I=1, IMAX                                          00019150
000104            DVSLOW(I)=VSLOW2(INDX,I)-VSLOW1(INDX,I)                00019160
000112            DVFAST(I)=VFAST1(INDX,I)-VFAST2(INDX,I)                00019170
000116     260 CONTINUE                                                  00019180
000121         I=1                                                       00019190
000122         J=1                                                       00019200
000123         JMAX=0                                                    00019210
000124     270 CONTINUE                                                  00019220
000124            IF (H-HA(INDX,I))400, 350, 300                         00019230
000131     300       CONTINUE                                            00019240
000131               HD(J)=HA(INDX,I)                                    00019250
000136               VSLOW(J)=VSLOW1(INDX,I)+WR*DVSLOW(I)                00019260
000143               VFAST(J)=VFAST2(INDX,I)+WR1*DVFAST(I)               00019270
000151               I=I+1                                               00019280
000152               J=J+1                                               00019290
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000153          GO TO 270                                                    00019300
000154      350 CONTINUE                                                     00019310
000154          HD(J)=H                                                      00019320
000156          JMAX=J                                                       00019330
000157          VSLCW(J)=VSLCW1(INDX,I)+WR*DVSLCW(I)                         00019340
000165          VFAST(J)=VFAST2(INDX,I)+WR1*DVFAST(I)                        00019350
000174          GO TO 450                                                    00019360
000174      400 CONTINUE                                                     00019370
000174          HR=(H-HA(INDX,I-1))/(HA(INDX,I)-HA(INDX,I-1))                00019380
000204          HR1=1.0-HR                                                   00019390
000206          HD(J)=H                                                      00019400
000210          JMAX=J                                                       00019410
000211          VS1=VSLCW1(INDX,I-1)+WR*DVSLCW (I-1)                         00019420
000216          VS2=VSLCW1(INDX,I)+WR*DVSLCW(I)                              00019430
000223          VSLCW(J)=VS1+HR*(VS2-VS1)                                    00019440
000227          VF1=VFAST2(INDX,I-1)+WR1*DVFAST(I-1)                         00019450
000235          VF2=VFAST2(INDX,I)+WR1*DVFAST(I)                             00019460
000242          VFAST(J)=VF2+HR1*(VF1-VF2)                                   00019470
              C                                                              00019480
              C     APPLY AERO ENVELOPE TEMPERATURE CORRECTION               00019490
              C     APPLY AERO ENVELOPE WIND CORRECTION                      00019500
              C                                                              00019510
000246      450 CONTINUE                                                     00019520
000246          DO 600 I=1, JMAX                                             00019530
000250             CALL WETHER(HD(I), ISW)                                   00019540
000252             WIND1=WINMAG*COS(ANGEF(IQUAD,INUMD)-WINANG)               00019550
000267             IF (HD(I).GE.12.5) GO TO 500                              00019560
000272                ANG2=0.5*(ANGEF(IQUAD,INUMD)+ANGIAF(IQUAD))            00019570
000277                WIND2=WINMAG*COS(ANG2-WINANG)                          00019580
000304                WIND3=WINMAG*COS(ANGIAF(IQUAD)-WINANG)                 00019590
000320                VSLCW(I) = AMAX1(WIND1,WIND2,WIND3) + VSLCW(I)*SQRT(TEMP/  00019600
            1                      TSTAND)                                 00019610
000334                VFAST(I) = AMIN1(WIND1,WIND2,WIND3) + VFAST(I)*SQRT(TEMP/  00019620
            1                      TSTAND)                                 00019630
000351                GO TO 600                                              00019640
000355      500    CONTINUE                                                  00019650
000355             VSLCW(I)=VSLCW(I)*SQRT(TEMP/TSTAND) + WIND1               00019660
000365             VFAST(I)=VFAST(I)*SQRT(TEMP/TSTAND) + WIND1               00019670
000374             IF (I.EQ.JMAX)    SPKNOT=38.975*SQRT(TEMP)*SPEED(ITHAC)+WIND1  00019680
000411      600 CONTINUE                                                     00019690
000414          RETURN                                                       00019700
000414          END                                                          00019710
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE AERO1                                                        00017070
      C                                                                                 00017080
000002          COMMON /AERO1CV/                                                        00017090
               1    HC(4,5),        TC(4,5),        TCSUM(4),       THPRES,          00017100
               2    VCCAS(5),       VCG(4,5),       VCTAS(4,5)                       00017110
000002          COMMON /ATMCV/                                                          00017120
               1    DENSR,          DENSRF,         DENSTY,         PRESR,           00017130
               2    PRESUR,         TEMR,           TEMRS                            00017140
000002          COMMON /GECMIN/                                                         00017150
               1    DPARAL,         EFX(4,3),       EFY(4,3),       EFZ(4,3),        00017160
               2    FAFX(2),        FAFY(2),        FAFZ(2),        IAFX(4),         00017170
               3    IAFY(4),        IAFZ(4),        MAX(2),         MAY(2),          00017180
               4    MAZ(2),         MFX(2),         MFY(2),         MFZ(2),          00017190
               5    NEF,            NPATH,          NQUAD,          NRNWY,           00017200
               6    CMX(2),         CMY(2),         CMZ(2),         RHOEF,           00017210
               7    RHOFAF,         RHOIAF,         RHOMF,          RHOOM,           00017220
               8    RHOTF,          TFX(4),         TFY(4),         TFZ(4),          00017230
               9    THX(2),         THY(2),         THZ(2)                           00017240
000002          REAL                IAFX,           IAFY,           IAFZ,            00017250
               1    MAX,            MAY,            MAZ,            MFX,             00017260
               2    MFY,            MFZ                                              00017270
000002          COMMON /GECMOT/                                                         00017280
               1    ANGEF(4,3),     ANGFAF(2),      ANGIAF(4),      ANGMF(2),        00017290
               2    ANGOM(2),       ANGTF(4),       DEF(4,3,5),     DFAF(2),         00017300
               3    DIAF(4),        DMF(2),         DOM(2),         DPATH(4,3,5),    00017310
               4    DTF(4),         DTH(2)                                           00017320
000002          COMMON /OMIAF/                                                          00017330
               1    ANG,            DIST                                             00017340
      C                                                                               00017350
000002          CALL WETHER (THZ(1), 0)                                                 00017360
000004          THPRES=PRESUR                                                           00017370
      C                                                                               00017380
      C     COMPUTE VCG AND OTHER VALUES AT OM                                        00017390
      C                                                                               00017400
000006          HC(1,1)=OMZ(1)                                                          00017410
000007          ANG=ANGOM(1)                                                            00017420
000011          DIST=0                                                                  00017430
000012          CALL TOMIAF(HC(1,1), VCCAS(1), VCTAS(1,1), 0, VCG(1,1), DUM)           00017440
000016          DO 100  I=2, NQUAD                                                      00017450
000020             HC(I,1)=HC(1,1)                                                      00017460
000022             VCTAS(I,1)=VCTAS(1,1)                                                00017470
000023             VCG(I,1)=VCG(1,1)                                                    00017480
000025    100 CONTINUE                                                                  00017490
      C                                                                               00017500
      C     COMPUTE TC AND OTHER VALUES AT FAF                                        00017510
      C                                                                               00017520
000027          HC(1,2)=FAFZ(1)                                                         00017530
000030          ANG=ANGFAF(1)                                                           00017540
000032          DIST=DFAF(1)                                                            00017550
000033          CALL TOMIAF(HC(1,2), VCCAS(2), VCTAS(1,2), VCG(1,1), VCG(1,2),          00017560
               1            TC(1,2))                                                   00017570
000037          DO 200  I=2, NQUAD                                                      00017580
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000041            HC(1,2)=HC(1,2)                                                    00017590
000043            VCTAS(1,2)=VCTAS(1,2)                                              00017600
000044            VCG(1,2)=VCG(1,2)                                                  00017610
000046            TC(1,2)=TC(1,2)                                                    00017620
000047      200 CONTINUE                                                             00017630
                C                                                                  00017640
                C     COMPUTE TC AND OTHER VALUES AT MF                            00017650
                C                                                                  00017660
000051          DO 300 I = 1, NQUAD                                                00017670
000053             INDX = 1                                                        00017680
000054             IF ((I.EQ.2).OR.(I.EQ.3)) INDX = 2                              00017690
000064             HC(I,3)=MFZ(INDX)                                               00017700
000067             ANG=ANGMF(INDX)                                                 00017710
000070             DIST=DMF(INDX)                                                  00017720
000072             CALL TCMIAF(HC(I,3), VCCAS(3), VCTAS(I,3), VCG(I,2), VCG(I,3), 00017730
               1                TC(I,3))                                          00017740
000076      300 CONTINUE                                                           00017750
                C                                                                  00017760
                C     COMPUTE TC AND OTHER VALUES AT TF                            00017770
                C                                                                  00017780
000101          DO 400 I=1, NQUAD                                                  00017790
000102             HC(I,4)=TFZ(I)                                                  00017800
000104             ANG=ANGTF(I)                                                    00017810
000105             DIST=DTF(I)                                                     00017820
000107             CALL TCMIAF(HC(I,4), VCCAS(4), VCTAS(I,4), VCG(I,3), VCG(I,4), 00017830
               1                TC(I,4))                                          00017840
000113      400 CONTINUE                                                           00017850
                C                                                                  00017860
                C     COMPUTE TC AND OTHER VALUES AT IAF                           00017870
                C                                                                  00017880
000116          DO 500 I=1, NQUAD                                                  00017890
000117             HC(I,5)=IAFZ(I)                                                 00017900
000121             ANG=ANGIAF(I)                                                   00017910
000122             DIST=DIAF(I)                                                    00017920
000124             CALL TCMIAF(HC(I,5), VCCAS(5), VCTAS(I,5), VCG(I,4), VCG(I,5), 00017930
               1                TC(I,5))                                          00017940
000130      500 CONTINUE                                                           00017950
000133          DO 600 I=1, NQUAD                                                  00017960
000134             TCSUM(I)=0                                                      00017970
000135             DO 600 J=2, NPATH                                               00017980
000137                TCSUM(I)=TCSUM(I)+TC(I,J)                                    00017990
000144      600 CONTINUE                                                           00018000
000151          RETURN                                                             00018010
000151          END                                                                00018020
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                                                                                00016790
                SUBROUTINE  ATMOSP (H,  TEMP, TSTAND)                           00016800
      C                                                                         00016810
000006          COMMON /ATMOV                                                   00016820
               1    DENSR,          DENSRF,         DENSTY,         PRESR,      00016830
               2    PRESUR,         TEMR,           TEMRS                       00016840
      C                                                                         00016850
      C     TEMRS IS THE TEMP. RATIO (STD. DAY)                                 00016860
      C     TEMR IS THE TEMPERATURE RATIO (STD. DAY)                            00016870
      C     PRESR IS THE PRESSURE RATIO (STD. DAY)                              00016880
      C     DENSR IS THE DENSITY RATIO                                          00016890
      C     DENSRF IS THE DENSITY RATIO FACTOR.                                 00016900
      C     PRESUR IS THE PRESSURE AT A GIVEN H AND TEMP. IN LBS./SQ. FT.       00016910
      C     DENSTY IS THE DENSITY AT A GIVEN H AND TEMP. IN LB.-SEC**2/FT**4    00016920
      C                                                                         00016930
000006          TEMRS = TSTAND/288.16                                           00016940
000007          TEMR = TEMP/288.16                                              00016950
000011          IF (H.GT.36.089) GO TO 10                                       00016960
000014             PRESR = TEMRS**5.256                                         00016970
000017             GO TO 20                                                     00016980
000020    10    CONTINUE                                                        00016990
000020             PRESR = .2234*EXP(-(H-36.0890)/20.806)                       00017000
000031    20    CONTINUE                                                        00017010
000031          DENSR = PRESR/TEMR                                              00017020
000033          DENSRF = 1./SQRT(DENSR)                                         00017030
000036          PRESUR = 2116.2*PRESR                                           00017040
000040          DENSTY = 0.002377*DENSR                                         00017050
000043          RETURN                                                          00017060
000043          END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                                                                        00000850
      BLOCK DATA                                                        00000860
C                                                                       00000870
C     BLOCK DATA CONTAINS THE PROGRAM CONSTANTS USED BY VARIOUS ROUTINES00000880
C                                                                       00000890
000002      COMMON /AERO1CV/                                            00000900
           1    HC(4,5),        TC(4,5),        TCSUM(4),       THPRES, 00000910
           2    VCCAS(5),       VCG(4,5),       VCTAS(4,5)              00000920
000002      COMMON /CNSTNT/                                             00000930
           1    ACCEL,          GRADI,          ITMAX,          PI,     00000940
           2    RTPTST                                                  00000950
000002      COMMON /RTPCV/                                              00000960
           1    DDP,            DLF,            HIAF,           I,      00000970
           2    INF1X,          IQUAD,          IS,             TIAF    00000980
C                                                                       00000990
000002      DATA                                                        00001000
           1    ACCEL,          DDP,            GRADI,          ITMAX/  00001010
           2    0.002,          10.,            0.25,           20.  /, 00001020
           3    PI,             RTPTST/                                 00001030
           4    3.1415927,      3.    /,        170.,           190.,   00001040
           5    VCCAS/          160.,                                   00001050
           6    220.,           250./                                   00001060
000002      END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE CASTAS (H, VCAS, VTAS)                                00032520
          C                                                                      00032530
          C     ENTER WITH ALTITUDE AND CALIBRATED AIRSPEED,                     00032540
          C     RETURN WITH TRUE AIRSPEED                                        00032550
          C                                                                      00032560
000006            COMMON /ATMOV                                                  00032570
                 1    DENSR,          DENSRF,         DENSTY,         PRESR,     00032580
                 2    PRESUR,         TEMR,           TEMRS                      00032590
          C                                                                      00032600
000006            CALL WETHER (H, 0)                                             00032610
000007            TEMP1 = ((VCAS/661.5)**2)*.2 + 1.                              00032620
000014            TEMP2 = (TEMP1**3.5 - 1.)/PRESR + 1.                           00032630
000022            EXPO  = 1./3.5                                                 00032640
000024            TEMP3 = (TEMP2**EXPO - 1.)*PRESR                               00032650
000031            VEAS  = 1479.*SQRT(TEMP3)                                      00032660
000034            VTAS  = VEAS*DENSRF                                            00032670
000040            RETURN                                                         00032680
000040            END                                                            00032690
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                                                                                00032700
                SUBROUTINE CNFLCTG (J1,J2,TEF2,TIAF2,IFLAG)                     00032710
      C                                                                         00032720
      C     THIS SUBROUTINE TESTS FOR A PASSING SITUATION REQUIRING             00032730
      C     THE USE OF PARALLEL PATHS                                           00032740
      C                                                                         00032750
000010          COMMON /RTPOUT/                                                 00032760
               1    DRTP(14,200),  HRTP(14,200),  IDRTP(14,200), TRTP(14,200), 00032770
               2    VRTP(14,200)                                               00032780
      C                                                                         00032790
      C     SET CONFLICT FLAG TO NO CONFLICT DETECTED                           00032800
      C     SEE IF THERE IS A PRIOR AIRCRAFT ON THE PATH, IF NONE RETURN        00032810
      C                                                                         00032820
000010          IFLAG = 0                                                       00032830
000010          IF (J1.EQ.0) RETURN                                             00032840
      C                                                                         00032850
      C     RETRIEVE INFORMATION ABOUT THE PRIOR AIRCRAFT ON THE PATH           00032860
      C                                                                         00032870
000012          DO 10 I=1,14                                                    00032880
000014             IF (IDRTP(I,J1).EQ.1) TEF1 = TRTP(I,J1)                      00032890
000023             IF (IDRTP(I,J1).EQ.2) TIAF1 = TRTP(I,J1)                     00032900
000032    10    CONTINUE                                                        00032910
      C                                                                         00032920
      C     SEE IF THE AIRCRAFT PASS.  IF SO SET THE CONFLICT FLAG              00032930
      C                                                                         00032940
000034          IF ((TEF1.GT.TEF2).AND.(TIAF1.LE.TIAF2)) IFLAG = 1              00032950
000047          RETURN                                                          00032960
000050          END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                      SUBROUTINE CNFLCTR (J1, J2, IFLAG,IFLITX,NSAMP)                  00032970
                                                                                       00032980
                C                                                                      00032990
                C     CNFLCTR TESTS THE RTP FOR AIRCRAFT J1 AND J2 FOR ALTITUDE        00033000
                C     DISTANCE AND TIME SEPARATION.  IF THE SEPARATION STANDARDS       00033010
                C     ARE VIOLATED, THERE IS A CONFLICT AND IFLAG IS SET TO 1.         00033020
                C                                                                      00033030
000010                COMMON /ATCIN/                                                   00033040
                     1    DIST(4),        DROO(10),       DRTPS,          HRTPS,      00033050
                     2    IRWO(4),        ISCSW,          TARO(10),       TIME(4),    00033060
                     3    TRTPS                                                      00033070
000010                COMMON /RTPOUT/                                                  00033080
                     1    DRTP(14,200),   HRTP(14,200),   IDRTP(14,200), TRTP(14,200),00033090
                     2    VRTP(14,200)                                               00033100
000010                DIMENSION           KEYR(28),       R1(28,4),       R2(28,4)   00033110
                C                                                                      00033120
                C     SET CONFLICT FLAG TO NO CONFLICT (I.E., EQUAL TO 0)              00033130
                C     IF J1 = 0, THERE IS NO PRIOR AIRCRAFT ON THE PATH, HENCE NO      00033140
                C     CONFLICT                                                         00033150
                C                                                                      00033160
000010                IFLAG = 0                                                        00033170
000010                IF (J1.LE.0) RETURN                                              00033180
                C                                                                      00033190
                C     FIND EF AND IAF TIMES FOR AIRCRAFT J1 AND J2                     00033200
                C                                                                      00033210
000012                DO 10 I=1,14                                                     00033220
000014                   IF (IDRTP(I,J2).EQ.1) TEF2 = TRTP(I,J2)                       00033230
000023                   IF (IDRTP(I,J2).EQ.1) IEF2 = I                                00033240
000030                   IF (IDRTP(I,J1).EQ.2) TIAF1 = TRTP(I,J1)                      00033250
000037                   IF (IDRTP(I,J1).EQ.2) IAF1 = I                                00033260
000044          10    CONTINUE                                                         00033270
                C                                                                      00033280
                C     ARE THE AIRCRAFT IN THE REGION BETWEEN THE EF AND THE IAF        00033290
                C     AT THE SAME TIME.                                                00033300
                C                                                                      00033310
000046                IF (TEF2.GE.TIAF1) RETURN                                        00033320
                C                                                                      00033330
                C     ZERO THE TEMPORARY RTP STORAGE                                   00033340
                C                                                                      00033350
000051                DO 20 I=1,28                                                     00033360
000053                DO 20 J = 1, 4                                                   00033370
000054                   R1(I,J) = 0.                                                  00033380
000057                   R2(I,J) = 0.                                                  00033390
000062          20    CONTINUE                                                         00033400
                C                                                                      00033410
                C     STORE THE PART OF THE RTP OF INTEREST IN THE TEMPORARY STORAGE   00033420
                C     FIND THE LEAST INDEX, IN TIME, FOR J1 GREATER THAN TEF2          00033430
                C                                                                      00033440
000066                DO 30 I=1,14                                                     00033450
000067                   IF (TRTP(I,J1).GE.TEF2) GO TO 40                              00033460
000073          30    CONTINUE                                                         00033470
000075                ISTOP = 1                                                        00033480
000076                GO TO 200
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000077   40    CONTINUE                                                    00033490
               C                                                          00033500
               C     IE1 CORRESPONDS IN THE J1 DATA TO THE EF TIME FOR J2 00033510
               C     IF IE1 EQUALS 1, THERE IS A PASSING SITUATION        00033520
               C                                                          00033530
000077         IE1 = I                                                    00033540
000101         IF (IE1.EQ.1) IFLAG = 1                                    00033550
000104         IF (IFLAG.EQ.1) RETURN                                     00033560
               C                                                          00033570
               C     FIND THE LARGEST INDEX IN TIME IN J2 DATA SO THAT IT IS 00033580
               C     LESS THAN TIAF1                                      00033590
               C                                                          00033600
000106         DO 50 I=1,14                                               00033610
000110            IF (TRTP(I,J2).LE.TIAF1) GO TO 50                       00033620
000114            I = I - 1                                               00033630
000116            GO TO 60                                                00033640
000116   50    CONTINUE                                                   00033650
000120         ISTOP = 2                                                  00033660
000121         GO TO 200                                                  00033670
000122   60    CONTINUE                                                   00033680
               C                                                          00033690
               C     IA2 CORRESPONDS FOR J2 TIME TO TIAF FOR AIRCRAFT J1  00033700
               C     STORE DATA IN R1 AND R2                              00033710
               C                                                          00033720
000122         IA2 = I                                                    00033730
000124         DO 70 J = IEF2, IA2                                        00033740
000126            I = J - IEF2 + 1                                        00033750
000130            R2(I,1) = TRTP(J,J2)                                    00033760
000134            R2(I,2) = DRTP(J,J2)                                    00033770
000137            R2(I,3) = HRTP(J,J2)                                    00033780
000142            R2(I,4) = VRTP(J,J2)                                    00033790
000145   70    CONTINUE                                                   00033800
000147         DO 80 I = IE1,IAF1                                         00033810
000151            J = I - IE1 + 1                                         00033820
000153            R1(J,1) = TRTP(I,J1)                                    00033830
000157            R1(J,2) = DRTP(I,J1)                                    00033840
000162            R1(J,3) = HRTP(I,J1)                                    00033850
000165            R1(J,4) = VRTP(I,J1)                                    00033860
000170   80    CONTINUE                                                   00033870
               C                                                          00033880
               C     FIND NUMBER OF ENTRIES IN EACH ARRAY AT THIS POINT   00033890
               C     FIND OUT IF NEED TO ADD A POINT TO R1 CORRESPONDING TO TEF2 00033900
               C                                                          00033910
000172         LNG1 = IAF1 - IE1 + 1                                      00033920
000174         LNG2 = IA2 - IEF2 + 1                                      00033930
000176         IF (TEF2.EQ.R1(1,1)) GO TO 90                              00033940
               C                                                          00033950
               C        NEED TO ADD THE POINT                             00033960
               C                                                          00033970
000200            DELT1 = TRTP(IE1,J1) - TRTP(IE1 - 1,J1)                 00033980
000206            DELT2 = R2(1,1) - TRTP(IE1 - 1,J1)                      00033990
000210            RATIO = DELT2/DELT1                                     00034000
000212            R1(LNG1+1,2) = (DRTP(IE1,J1) - DRTP(IE1-1,J1))*RATIO    00034010
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A         16.56.27. 73/12/21.

                1                    + DRTP(IE1-1,J1)                                    00034020
000221              R1(LNG1+1,3) = (HRTP(IE1,J1) - HRTP(IE1-1,J1))*RATIO                 00034030
                1                    + HRTP(IE1-1,J1)                                    00034040
000230              R1(LNG1+1,4) = (VRTP(IE1,J1) - VRTP(IE1-1,J1))*RATIO                 00034050
                1                    + VRTP(. 1-1,J1)                                    00034060
000240              R1(LNG1+1,1) = TEF2                                                  00034070
000242              LNG1 = LNG1 + 1                                                      00034080
000243    90      CONTINUE                                                             00034090
          C                                                                            00034100
          C       FIND OUT IF NEED TO ADD POINT TO R2 CORRESPONDING TO TIAF1           00034110
          C                                                                            00034120
000243            IF (TIAF1.EQ.R2(LNG2,1)) GO TO 100                                   00034130
          C                                                                            00034140
          C         NEED TO ADD THE POINT                                              00034150
          C                                                                            00034160
000246              DELT1 = TRTP(IA2 + 1,J2) - TRTP(IA2,J2)                            00034170
000254              DELT2 = TIAF1 - TRTP(IA2,J2)                                       00034180
000256              RATIO = DELT2/DELT2                                                00034190
000257              R2(LNG2+1,2) = (DRTP(IA2+1,J2) - DRTP(IA2,J2))*RATIO               00034200
                1                    + DRTP(IA2,J2)                                  00034210
000266              R2(LNG2+1,3) = (HRTP(IA2+1,J2) - HRTP(IA2,J2))*RATIO               00034220
                1                    + HRTP(IA2,J2)                                  00034230
000275              R2(LNG2+1,4) = (VRTP(IA2+1,J2) - VRTP(IA2,J2))*RATIO               00034240
                1                    + VRTP(IA2,J2)                                  00034250
000305              R2(LNG2+1,1) = TIAF1                                               00034260
000307              LNG2 = LNG2 + 1                                                    00034270
000310    100     CONTINUE                                                           00034280
          C                                                                          00034290
          C       START TO FILL ARRAYS FOR THE OTHER NON-CORRESPONDING POINTS        00034300
          C       SORT R1 AND R2 FIRST                                               00034310
          C                                                                          00034320
000310            CALL SHELL  (R1(1,1),KEYR,LNG1)                                    00034330
000313            CALL SHELLX (R1(1,2),KEYR,LNG1)                                    00034340
000316            CALL SHELLX (R1(1,3),KEYR,LNG1)                                    00034350
000321            CALL SHELLX (R1(1,4),KEYR,LNG1)                                    00034360
000324            CALL SHELL  (R2(1,1),KEYR,LNG2)                                    00034370
000327            CALL SHELLX (R2(1,2),KEYR,LNG2)                                    00034380
000332            CALL SHELLX (R2(1,3),KEYR,LNG2)                                    00034390
000335            CALL SHELLX (R2(1,4),KEYR,LNG2)                                    00034400
000340            LN1 = LNG1 - 1                                                     00034410
000342            LN2 = LNG2 - 1                                                     00034420
000344            DO 120 J = 1, LN2                                                  00034430
000350            DO 120 I = 2, LN1                                                  00034440
000351              IF (R1(I,1).EQ.R2(J,1)) GO TO 120                                00034450
000354              IF ((R2(J,1).GE.R1(I,1)).OR.(R2(J+1,1).LE.R1(I,1))) GO TO 120    00034460
          C                                                                          00034470
          C           NEED TO ADD A POINT IN R2, AS THERE IS A PT. IN R1 BETWEEN     00034480
          C           R2(J,1) AND R2(J+1,1)                                          00034490
          C                                                                          00034500
000365                R2(LNG2+1,1) = R1(I,1)                                         00034510
000367                DELT1 = R2(J+1,1) - R2(J,1)                                    00034520
000372                DELT2 = R1(I,1) -R2(J,1)                                       00034530
000375                RATIO = DELT2/DELT1                                            00034540
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000377                R2(LNG2+1,2) = R2(J,2) + RATIO*(R2(J+1,2) - R2(J,2))        00034550
000405                R2(LNG2+1,3) = R2(J,3) + RATIO*(R2(J+1,3) - R2(J,3))        00034560
000414                R2(LNG2+1,4) = R2(J,4) + RATIO*(R2(J+1,4) - R2(J,4))        00034570
000423                LNG2 = LNG2 + 1                                             00034580
000424    120    CONTINUE                                                         00034590
                 C                                                               00034600
                 C     INTERCHANGE ROLES OF R1 AND R2 TO FILL R1 ARRAY           00034610
                 C                                                               00034620
000431           CALL SHELL  (R1(1,1),KEYR,LNG1)                                 00034630
000434           CALL SHELLX (R1(1,2),KEYR,LNG1)                                 00034640
000437           CALL SHELLX (R1(1,3),KEYR,LNG1)                                 00034650
000442           CALL SHELLX (R1(1,4),KEYR,LNG1)                                 00034660
000445           CALL SHELL  (R2(1,1),KEYR,LNG2)                                 00034670
000450           CALL SHELLX (R2(1,2),KEYR,LNG2)                                 00034680
000453           CALL SHELLX (R2(1,3),KEYR,LNG2)                                 00034690
000456           CALL SHELLX (R2(1,4),KEYR,LNG2)                                 00034700
000461           LN1 = LNG1 - 1                                                  00034710
000463           LN2 = LNG2 - 1                                                  00034720
000465           DO 140 J = 1, LN1                                               00034730
000471           DO 140 I = 2, LN2                                               00034740
000472              IF (R2(I,1).EQ.R1(J,1)) GO TO 140                            00034750
000475              IF ((R1(J,1).GE.R2(I,1)).OR.(R1(J+1,1).LE.R2(I,1))) GO TO 140 00034760
                 C                                                               00034770
                 C        NEED TO ADD A POINT IN R1, AS THERE IS A PT. IN R2 BETWEEN 00034780
                 C        R1(J,1) AND R1(J+1,1)                                  00034790
                 C                                                               00034800
000506                    R1(LNG1+1,1) = R2(I,1)                                 00034810
000510                    DELT1 = R1(J+1,1) - R1(J,1)                            00034820
000513                    DELT2 = R2(I,1) - R1(J,1)                              00034830
000516                    RATIO = DELT2/DELT1                                    00034840
000520                    R1(LNG1+1,2) = R1(J,2) + RATIO*(R1(J+1,2) - R1(J,2))   00034850
000526                    R1(LNG1+1,3) = R1(J,3) + RATIO*(R1(J+1,3) - R1(J,3))   00034860
000535                    R1(LNG1+1,4) = R1(J,4) + RATIO*(R1(J+1,4) - R1(J,4))   00034870
000544                    LNG1 = LNG1 + 1                                        00034880
000545    140    CONTINUE                                                         00034890
                 C                                                               00034900
                 C     ALL POINTS ADDED TO R1 AND R2.  SORT BEFORE CHECKING IF ANY 00034910
                 C     SEPARATION VIOLATIONS ARE PRESENT.                        00034920
                 C                                                               00034930
000552           CALL SHELL  (R1(1,1),KEYR,LNG1)                                 00034940
000555           CALL SHELLX (R1(1,2),KEYR,LNG1)                                 00034950
000560           CALL SHELLX (R1(1,3),KEYR,LNG1)                                 00034960
000563           CALL SHELLX (R1(1,4),KEYR,LNG1)                                 00034970
000566           CALL SHELL  (R2(1,1),KEYR,LNG2)                                 00034980
000571           CALL SHELLX (R2(1,2),KEYR,LNG2)                                 00034990
000574           CALL SHELLX (R2(1,3),KEYR,LNG2)                                 00035000
000577           CALL SHELLX (R2(1,4),KEYR,LNG2)                                 00035010
                 C                                                               00035020
                 C     IF ARRAY LENGTHS ARE NOT THE SAME, THERE IS AN ERROR.     00035030
                 C                                                               00035040
000602           IF (LNG1.NE.LNG2) ISTOP = 3                                     00035050
000610           IF (LNG1.NE.LNG2) GO TO 200                                     00035060
                 C                                                               00035070
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

      C     GO THROUGH RTP AND LOOK FOR SEPARATION VIOLATIONS                  00035080
      C                                                                        00035090
000612        DO 190 I = 1, LNG1                                               00035100
      C                                                                        00035110
      C        CHECK DISTANCE                                                  00035120
      C        CHECK ALTITUDE                                                  00035130
      C        CHECK TIME                                                      00035140
      C                                                                        00035150
000613           IF (ABS(R1(I,2) - R2(I,2)).GT.DRTPS) GO TO 190                  00035160
000621           IF (ABS(R1(I,3) - R2(I,3)).GT.HRTPS) GO TO 190                  00035170
      C                                                                        00035180
      C        NOTE THAT THE DISTANCE CHECK IS ALREADY DONE AND WAS NOT        00035190
      C        VIOLATION FREE                                                  00035200
      C                                                                        00035210
000626           DELD = R1(I,2) - R2(I,2)                                        00035220
000630           DELT = 3600.*DELD/R2(I,4)                                       00035230
000633           IF (DELT.GT.TRTPS) GO TO 190                                    00035240
000636           IFLAG = 1                                                       00035250
000636           RETURN                                                          00035260
000637   190   CONTINUE                                                        00035270
000642         RETURN                                                          00035280
      C                                                                        00035290
      C     CNFLCTR ERROR RETURN                                               00035300
      C     PRINT ERROR MESSAGE.  RTP WILL PROCEED TO NEXT AIRCRAFT            00035310
      C                                                                        00035320
000642   200   CONTINUE                                                        00035330
000642         IFLITX = NSAMP                                                  00035340
000643         PRINT 9000, J1, J2, NSAMP, ISTOP                                00035350
000657   9000  FORMAT (1HO, 3X, 26HSUBROUTINE CNFLCTR CANNOT ,                 00035360
              1        16HTEST FOR CONFLICTS / 4X, 16HBETWEEN AIRCRAFT , I4,  00035370
              2        4H AND, I4, 33H (RUNWAY SEQUENCE NOS.) IN SAMPLE ,     00035380
              3        I4, 12H.  ERROR NO. , I4)                              00035390
000657         RETURN                                                          00035400
000660         END                                                             00035410
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION 00? 73 A          16.56.27. 73/12/21.

                SUBROUTINE DETAIL                                                    00054720
      C                                                                              00054730
000002          COMMON /FLGTXCV                                                      00054740
               1    DISSEP(16,17), DLY(6,17),      IHIST(5),      NAR,             00054750
               2    [illegible],6),  NDCR(16),     NDEC[illegible],6), NDEL(6),    00054760
               3    NDEP,          NHIST,          NOCF(16),      NP(36),          00054770
               4    NRD(4),        NRDCR(4),       NRECOD(5),     NRT(4),          00054780
               5    NRTCR(4),      NT(16),         NTCR(16),      OPTIM(4,17),     00054790
               6    SUM(6),        SUMSQ(6),       SUMTIM,        TIMSEP(16,17),   00054800
               7    TIPATH(36,17)                                                   00054810
000002          INTEGER            DISSEP,         DLY,           OPTIM,           00054820
               1    TIMSEP,        TIPATH                                          00054830
      C                                                                              00054840
000002          DIMENSION          ACTYPP(21),     AORDP(21),     ARRAY(66,20),    00054850
               1    IEFNOP(21),    INITMP(21),     ISEQNO(21),    PASNOP(21),      00054860
               1    TIMEAP(30,21), TIMESP(30,21)                                   00054870
000002          INTEGER            ACTYPP,         AORDP,         ARRAY,           00054880
               1    PASNOP,        TIMEAP,         TIMESP                          00054890
      C                                                                              00054900
000002          PRINT 2410                                                         00054910
000006          REWIND 1                                                           00054920
000010          DO 9000 IH=1, NHIST                                                00054930
000012             IF (IH.GT.1)  PRINT 2416                                        00054940
000020             PRINT 2415, IH, IHIST(IH)                                       00054950
      C                                                                          00054960
000030             ISEQ1= 1                                                        00054970
000031             IPAGE=0                                                         00054980
000032             NPLANE=NRECOD(IH)                                               00054990
000034  8000       CONTINUE                                                        00055000
000034             IPAGE=IPAGE+1                                                   00055010
000036             MOD12=IPAGE-12                                                  00055020
      C                                                                          00055030
      C          ISEQ2    SEQ. NO. OF LAST A/C WRITTEN ON THIS PAGE              00055040
      C          ISEQ     NO. OF COLUMNS WRITTEN ON THIS PAGE                    00055050
      C                                                                          00055060
000040             ISEQ2= MIND(MOD12, NPLANE)                                      00055070
000043             ISEQ=ISEQ2-MOD12+12                                             00055080
000045             DO 8100 J=1, ISEQ                                               00055090
000046                READ (1) (ARRAY(L,J), L=1, 66)                               00055100
000061  8100       CONTINUE                                                        00055110
000064             DO 8200 J=1, ISEQ                                               00055120
000065                INITMP(J)=ARRAY(2,J)                                         00055130
000070                ACTYPP(J)=ARRAY(3,J)                                         00055140
000073                AORDP (J)=ARRAY(4,J)                                         00055150
000076                IEFNOP(J)=ARRAY(5,J)                                         00055160
000101                PASNOP(J)=ARRAY(6,J)                                         00055170
000104  8200       CONTINUE                                                        00055180
000106             DO 8300 J=1, ISEQ                                               00055190
000110                DO 8300 K=1, 30                                              00055200
000111                   TIMEAP(K,J) = ARRAY(2*K+5,J)                              00055210
000117                   TIMESP(K,J) = ARRAY(2*K+6,J)                              00055220
000124  8300       CONTINUE                                                        00055230
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000130            DO 8510  I=1, ISEQ                                    00055240
000132               ISEQNO(I)=ISEQ1                                     00055250
000134               ISEQ1=ISEQ1+1                                       00055260
000135    8510    CONTINUE                                               00055270
000137            IF (IPAGE.GT.1) PRINT 2416                             00055280
000144            PRINT 2420, (ISEQNO(I), I=1, ISEQ)                     00055290
000157            PRINT 2430, (INITMP(I), I=1, ISEQ)                     00055300
000172            PRINT 2440, (ACTYPP(I), I=1, ISEQ)                     00055310
000205            PRINT 2450, (ACRDP (I), I=1, ISEQ)                     00055320
000220            PRINT 2460, (IEFNCP(I), I=1, ISEQ)                     00055330
000233            PRINT 2470, (PASNCP(I), I=1, ISEQ)                     00055340
000246            DO 8520 J=1, 12                                        00055350
000250               PRINT 2480, (TIMEAP(J,I), I=1, ISEQ)                00055360
000263               PRINT 2490, J, (TIMESP(J,I), I=1, ISEQ)             00055370
000301    8520    CONTINUE                                               00055380
000303            PRINT 2500, (ISEQNO(I), I=1, ISEQ)                     00055390
000316            DO 8530 J=13, 16                                       00055400
000320               PRINT 2480, (TIMEAP(J,I), I=1, ISEQ)                00055410
000333               JJ=J-12                                             00055420
000335               PRINT 2510, JJ, (TIMESP(J,I), I=1, ISEQ)            00055430
000353    8530    CONTINUE                                               00055440
000355               DO 8540 J=17, 20                                    00055450
000357               PRINT 2480, (TIMEAP(J,I), I=1, ISEQ)                00055460
000372               JJ=J-16                                             00055470
000374               PRINT 2520, JJ, (TIMESP(J,I), I=1, ISEQ)            00055480
000412    8540    CONTINUE                                               00055490
000414            DO 8550 J=21, 22                                       00055500
000416               JJ=J-20                                             00055510
000420               PRINT 2480, (TIMEAP(J,I), I=1, ISEQ)                00055520
000433               PRINT 2530, JJ, (TIMESP(J,I), I=1, ISEQ)            00055530
000451    8550    CONTINUE                                               00055540
000453            DO 8560 J=23, 24                                       00055550
000455               JJ=J-22                                             00055560
000457               PRINT 2480, (TIMEAP(J,I), I=1, ISEQ)                00055570
000472               PRINT 2540, JJ, (TIMESP(J,I), I=1, ISEQ)            00055580
000510    8560    CONTINUE                                               00055590
000512            DO 8570 J=25, 26                                       00055600
000514               JJ=J-24                                             00055610
000516               PRINT 2480, (TIMEAP(J,I), I=1, ISEQ)                00055620
000531               PRINT 2550, JJ, (TIMESP(J,I), I=1, ISEQ)            00055630
000547    8570    CONTINUE                                               00055640
000551            DO 8580 J=27, 28                                       00055650
000553               JJ=J-26                                             00055660
000555               PRINT 2480, (TIMEAP(J,I), I=1, ISEQ)                00055670
000570               PRINT 2560, JJ, (TIMESP(J,I), I=1, ISEQ)            00055680
000603    8580    CONTINUE                                               00055690
000610            IF (ISEQ2.NE.NPLANE)  GO TO 8000                       00055700
000612    9000 CONTINUE                                                  00055710
000615         RETURN                                                    00055720
                                                                        00055730
          C
000615    2420 FORMAT (1HO, 9H  SEQ NO ,   12(3X, I3))                  00055740
000615    2430 FORMAT (1HO,  9H INI TIME,  12I6)                        00055750
000615    2440 FORMAT (1HO, 9H A/C TYPE , 12(2X, A4))                   00055760
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000615    2450 FORMAT (1HO, 11H  A OR D   , 12(3X, A1, 2X))                    00055770
000615    2460 FORMAT (1HO, 11H E.F. NO   ,12(2X, I2, 2X))                     00055780
000615    2470 FORMAT (1HO, 11H F.F. NO   ,12(2X, I2, 2X))                     00055790
000615    2480 FORMAT (3HO  , 7HTIME A , 1216)                                 00055800
000615    2490 FORMAT (3X, 2HEF, I2, 3H S , 1216)                              00055810
000615    2500 FORMAT (1H1/1HO, 9H  SEQ NO , 12(3X, I3)/ 1H )                  00055820
000615    2510 FORMAT(3X, 3HIAF, I1, 1X, 1HS, 1X, 1216)                        00055830
000615    2520 FORMAT(3X, 3HTF , I1, 1X, 1HS, 1X, 1216)                        00055840
000615    2530 FORMAT(3X, 3HMF , I1, 1X, 1HS, 1X, 1216)                        00055850
000615    2540 FORMAT(3X, 3HFAF, I1, 1X, 1HS, 1X, 1216)                        00055860
000615    2550 FORMAT(3X, 3HCM , I1, 1X, 1HS, 1X, 1216)                        00055870
000615    2560 FORMAT(3X, 3HTH , I1, 1X, 1HS, 1X, 1216)                        00055880
000615   2410  FORMAT(1H1/ 6X, 34H***** DETAILED SCHEDULES AND TIME ,          00055890
             1       33HHISTORIES OF ALL ACTIVITIES ***** //)                 00055900
000615   2415  FORMAT(1H ,I2, 46H. TIME HISTORY FOR ARRIVALS AND DEPARTURES IN , 00055910
             1       23HSECONDS FOR SAMPLE NO. , I3 /1HO, 2X,10HA = ACTUAL, 2X, 00055920
             2       13HS = SCHEDULED, 2X, 14HEF = ENTRY FIX, 2X,11HFP = PARALL,00055930
             3        7HEL PATH, 2X,16HCF = CONTROL FIX )                     00055940
000615   2416  FORMAT(1H1/)                                                    00055950
000615         END                                                            00055960
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE ERLANG (S, YMU, YK)                                  00014540
        C                                                                       00014550
        C       THE ROUTINE COMPUTES   AN ERLANG DISTRIBUTION CURVE (CUMULATIVE OF00014560
        C       DENSITY CURVE) OF 100 DATA PTS WITH FIXED INCREMENTS. THE RANDOM 00014570
        C       NUMBER GENERATOR PRODUES AND UNIFORMLY DISTRIBUTED RANDOM NUMBER. 00014580
        C       THE CORRESPONDING ERLANG RANDOM NUMBER IS OBTAINED BY DOING LINEAL00014590
        C       INTERPRETATION ON THE DISTRIBUTION CURVE.                       00014600
        C                                                                       00014610
        C       S         RANDOM NUMBER RETURNED                                00014620
        C       YMU       MEAN  OF THE ERLANG DISTRIBUTION                      00014630
        C       YK        ORDER OF THE ERLANG DISTRIBUTION                      00014640
        C       V(I)      ARRAY OF EQUALLY SPACED NUMBERS BTWN 0.0 AND 1.0      00014650
        C       U(I)      CORRESPONDING ERLANG DISTRIBUTION VALUES              00014660
        C                                                                       00014670
000006          COMMON /RANDOM/                                                 00014680
               1    IU,              IX                                       00014690
000006          DIMENSION            FACTOR(36),    U(101),        V(101)       00014700
        C                                                                       00014710
000006          K=YK+0.1                                                        00014720
000010          XMUK=K/YMU                                                      00014730
000012          YMU3=3.0*YMU                                                    00014740
000013          IF (K.EQ.1) YMU3=5.0*YMU                                        00014750
000017          K1=K-1                                                          00014760
000021          DO 100 I=1, K1                                                  00014770
000022            FACTOR(I)=FACT(I)                                             00014780
000025  100     CONTINUE                                                        00014790
000031          DELTAV=0.01*YMU3                                                00014800
000033          V(2)=DELTAV                                                     00014810
000035          DO 300 I=2, 100                                                 00014820
000036            POWER=XMUK*V(I)                                               00014830
000040            UP=0                                                          00014840
000040            IF (K1.EQ.0) GO TO 210                                        00014850
000042              DO 200 J=1, K1                                              00014860
000043                UP=UP+(POWER**J)/FACTOR(J)                                00014870
000051  200         CONTINUE                                                    00014880
000054  210       CONTINUE                                                      00014890
000054            V(I+1)=V(I)+DELTAV                                            00014900
000057            U(I)=1.0-(1.0+UP)/EXP(POWER)                                  00014910
000066  300     CONTINUE                                                        00014920
000072          U(1)=0.0                                                        00014930
000072          V(1)=0.0                                                        00014940
000073          U(101)=1.0                                                      00014950
000074          V(101)=YMU*10.0                                                 00014960
000076          IF (K.EQ.1) V(101)=20.0*YMU                                     00014970
        C       ********************************                                00014980
        C       ********************************                                00014990
000102          CALL RAND (IU, IX, 0, 1, R)                                     00015000
        C       ********************************                                00015010
        C       ********************************                                00015020
000106          S  = TBLU(101, V, U, R)                                         00015030
000115          RETURN                                                          00015040
000116          END                                                             00015050
```

TABLE 5-1.--CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                                                                                00046840
                SUBROUTINE ERROR(KAOR,ICHK,INOD)                                00046850
C                                                                               00046860
000006          COMMON /AERCOT/                                                 00046870
               1    HD(10),          HCRIT,          JMAX,          SFKNOT,    00046880
               2    VFAST(10),       VSLOW(10)                                 00046890
000006          COMMON /AELCOV/                                                 00046900
               1    HC(4,5),         TC(4,5),        TCSUM(4),      THFRES,    00046910
               2    VCCAS(5),        VCG(4,5),       VCTAS(4,5)                00046920
000006          COMMON /AVICOM/                                                 00046930
               1    MURO,            MUT,            SIGRO,         SIGT       00046940
000006          REAL                 MURO,           MUT                       00046950
000006          COMMON /CNSTNT/                                                 00046960
               1    ACCEL,           GRADI,          ITMAX,         PI,        00046970
               2    RTFTST                                                     00046980
000006          COMMON /ERCOT/                                                  00046990
               1    ICFLG(7),        TFR(7),         TSCH(7),       VFR(7)     00047000
000006          COMMON /GECMIN/                                                 00047010
               1    DFARAL,          EFX(4,3),       EFY(4,3),      EFZ(4,3),  00047020
               2    FAFX(2),         FAFY(2),        FAFZ(2),       IAFX(4),   00047030
               3    IAFY(4),         IAFZ(4),        MAX(2),        MAY(2),    00047040
               4    MAZ(2),          MFX(2),         MFY(2),        MFZ(2),    00047050
               5    NEF,             NPATH,          NQUAD,         NRNWY,     00047060
               6    OMX(2),          OMY(2),         OMZ(2),        RHOEF,     00047070
               7    RHOFAF,          RHOIAF,         RHOMF,         RHOOM,     00047080
               8    RHOTF,           TFX(4),         TFY(4),        TFZ(4),    00047090
               9    THX(2),          THY(2),         THZ(2)                    00047100
000006          REAL                 IAFX,           IAFY,          IAFZ,      00047110
               1    MAX,             MAY,            MAZ,           MFX,       00047120
               2    MFY,             MFZ                                       00047130
000006          COMMON /GECMON/                                                 00047140
               1    ANGEF(4,3),      ANGFAF(2),      ANGIAF(4),     ANGMF(2),  00047150
               2    ANGOM(2),        ANGTF(4),       DEF(4,3,5),    DFAF(2),   00047160
               3    DIAF(4),         DMF(2),         DOM(2),        DPATH(4,3,5),00047170
               4    DTF(4),          DTH(2)                                    00047180
000006          COMMON /RANDOM/                                                 00047190
               1    IU,              IX                                        00047200
000006          COMMON /RTFOUT/                                                 00047210
               1    DRTF(14,200),    HRTF(14,200),   IDRTF(14,200), TRTF(14,200),00047220
               2    VRTF(14,200)                                               00047230
000006          COMMON /TRAFOT/                                                 00047240
               1    ALT(200),        EELT(200),      ELLT(200),     GWA(200),  00047250
               2    GWD(100),        ITYPEA(200),    ITYPED(100),   KEY(200),  00047260
               3    LFIX(200),       NACA,           NACD,          SDT(100),  00047270
               4    SLT(200),        SPEED(200),     TSECAA(200),   TSECAD(100),00047280
               5    TSECSA(200),     TSECSD(100),    VTAG(200),     VREF(200)  00047290
000006          COMMON /WIND/                                                   00047300
               1    GWIND1,          GWIND2,         GWIND3,        HWIND1,    00047310
               2    HWIND2,          WERRE1,         WERRE2,        WERRE3,    00047320
               3    WERRN1,          WERRN2,         WERRN3,        WINANG,    00047330
               4    WINDE,           WINDE0,         WINDE1,        WINDE2,    00047340
               5    WINDN,           WINDN0,         WINDN1,        WINDN2,    00047350
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                7-    WINMAG                                                  00047360
000006          DIMENSION           ACR(14),        ID(14),         PSI(7),   00047370
                1     T.US(14),     THETA(7),       TR(14),         TWPT(14), 00047380
                2     VR(14)                                                  00047390
      C                                                                       00047400
      C     LOCATE THE EF QUADRANT, NUMBER AND PARALLEL PATH                  00047410
      C                                                                       00047420
000006          GAMMA = 0.05                                                  00047430
000007          KX=KARR                                                       00047440
000010          DO 12 I=1,14                                                  00047450
000012             TWPT(I) = 0.                                               00047460
000013       12 CONTINUE                                                      00047470
000015          DO 9 I=1,14                                                   00047480
000016             TWPT(I)=TRTP(I,KX)/3600.0                                  00047490
000022             IF(IDRTF(I,KX).EQ.7) GO TO 14                              00047500
000026        9 CONTINUE                                                      00047510
000027          STOP 7                                                        00047520
000031       14 CONTINUE                                                      00047530
000031          I = LFIX(KX)/100                                              00047540
000035          J=LFIX(KX)/10-10*I                                            00047550
000041          JF=J                                                          00047560
000042          THETA(1)= ANGEF(I,J)                                          00047570
000045          THETA(2)=ANGIAF(I)                                            00047580
000046          THETA(3)=ANGTF(I)                                             00047590
000050          K=1                                                           00047600
000051          IF (I.EQ.2.OR.I.EQ.3)K=2                                      00047610
000061          THETA(4)= ANGMF(K)                                            00047620
000063          IF(NRNWY.NE.1)GO TO 100                                       00047630
000065             THETA(5)=ANGFAF(1)                                         00047640
000067             THETA(6)= ANGCM(1)                                         00047650
000070             GO TO 150                                                  00047660
000071       100 CONTINUE                                                     00047670
000071          K=1                                                           00047680
000072          IF (I.EQ.2.OR.I.EQ.3)K=2                                      00047690
000103          THETA(5)=ANGFAF(K)                                            00047700
000105          THETA(6)= ANGCM(K)                                            00047710
000107       150 CONTINUE                                                     00047720
000107          THETA(7)=THETA(6)                                             00047730
      C                                                                       00047740
      C     GENERATE RANDOM POSITIONAL ERROR                                  00047750
      C                                                                       00047760
      C     ********************************                                  00047770
      C     ********************************                                  00047780
000111          CALL RAND(IU,IX,1,1,RHO)                                      00047790
      C     ********************************                                  00047800
      C     ********************************                                  00047810
000114          RHO=RHO*SIGRO+MURO                                            00047820
      C     ********************************                                  00047830
      C     ********************************                                  00047840
000117          CALL RAND(IU,IX,0,1,PHI)                                      00047850
      C     ********************************                                  00047860
      C     ********************************                                  00047870
000123          PHI=2.0*PI*PHI                                                00047880
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000126          DO 200 M=1,7                                                    00047890
000131             PSI(M)=RHO*COG(THETA(N)-PHI)                                 00047900
000140       200 CONTINUE                                                       00047910
          C                                                                     00047920
          C     GET TABLE OF VFAST,VSLOW VS. ALTITUDE AND RTP PROFILE         00047930
          C     EVALUATE TRUE TIME SCHEDULE AND SPEE AT WAY POINT M AND M+1   00047940
          C     DESIGNATE THE M-TH WAY POINT AS A AND THE M+1-TH WAY POINT AS B 00047950
          C                                                                     00047960
000143          NFX=7                                                         00047970
000144          VR(1)=VRTF(1,KX)                                              00047980
000147          TR(1)=TWPT(1)                                                 00047990
000151          VFR(1)=VR(1)                                                  00048000
000152          TSCH(1)=TRTF(1,KX)                                            00048010
000155          TFR(1)=TSCH(1)                                                00048020
000156          ICNW=0                                                        00048030
000157          ICFLG(1)=0                                                    00048040
000160          M=0                                                           00048050
000161          IF (ICHK.NE.0) GO TO 250                                      00048060
000162             IF (MCD(IMCD,2).EQ.0) PRINT 2060                           00048070
000173             PRINT 2070                                                 00048080
000177    2060     FORMAT(1H1)                                               00048090
000177    2070     FORMAT(1HD,3X,3HWAY, 4X, 27HW A Y  P O I N T  T I M E S , 4X, 00048100
                1       38HW A Y  P O I N T  V E L O C I T I E S ,3X,1GHCONSTRAINT/00048110
                2       2X, 5HPOINT, 2X, 9HSCHEDULED, 2X, 8HAIRCRAFT, 4X, 8HACTUAL 00048120
                3       3X,9HSCHEDULED,2X,8HACTUAL,4X,7HMINIMUM,3X,7HMAXIMUM,     00048130
                4       4X, 9HVIOLATION / 2X, 5HINDEX, 13X, 13HS E C O N D S ,24X, 00048140
                5       9HK N O T S , 19X, 4HFLAG //)                           00048150
000177       250 CONTINUE                                                     00048160
000177          CALL AERO(DUMD,KARR,I,JF,1)                                   00048170
          C                                                                     00048180
          C     SHIFT INDEX ON ALTITIUDES AND SPEEDS FROM AERO OUTPUT         00048190
          C                                                                     00048200
000205          NFT= 3                                                        00048210
000206          DO 300 IUP=1,JMAX                                             00048220
000212             IDWN=JMAX-IUP+1                                            00048230
000214             HD(IDWN+NFT)=HD(IDWN)                                      00048240
000217             VFAST(IDWN+NFT)=VFAST(IDWN)                                00048250
000222             VSLOW(IDWN+NFT)=VSLOW(IDWN)                                00048260
000224       300 CONTINUE                                                     00048270
          C                                                                     00048280
          C     EXTEND AEROPERFORMANCE ENVELOPE TO GROUND                     00048290
          C                                                                     00048300
000227          JMAX=JMAX+NFT                                                 00048310
000230          HD(3)=IAFZ(I)-1.                                              00048320
000232          XLT=HD(3)                                                     00048330
000233          VELF=250.                                                     00048340
000235          VELS=VCCAS(1)                                                 00048350
000236          CALL CASTAS(XLT,VELF,VTAS1)                                   00048360
000241          CALL CASTAS(XLT,VELS,VTAS2)                                   00048370
000244          CALL WETHER(XLT,1)                                            00048380
000246          DAN=ANGIAF(I)-WINANG                                          00048390
000251          VEL1=VTAS1+WINMAG*COS(DAN)                                    00048400
000255          VEL2=VTAS2+WINMAG*COS(DAN)                                    00048410
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000262          VFAST(3)=VEL1                                          00048420
000264          VSLOW(3)=VEL2                                          00048430
000265          HD(2)=CHZ(1)                                           00048440
000267          XLT=HD(2)                                              00048450
000270          VELF=215.                                              00048460
000271          VELS=VCCAS(1)                                          00048470
000273          CALL CASTAS(XLT,VELF,VTAS1)                            00048480
000275          CALL CASTAS(XLT,VELS,VTAS2)                            00048490
000300          CALL WETHER(XLT,1)                                     00048500
000302          DAN=ANGCM(1)-WINANG                                    00048510
000304          VEL1=VTAS1+WINMAG*CCS(DAN)                             00048520
000311          VEL2=VTAS2+WINMAG*CCS(DAN)                             00048530
000316          VFAST(2)=VEL1                                          00048540
000320          VSLOW(2)=VEL2                                          00048550
000321          HD(1)=0.                                               00048560
000322          VFAST(1)=VDUGG(KX)+1.0                                 00048570
000324          VSLOW(1)=VDUGG(KX)                                     00048580
000326          DO 800 J=2,NFX                                         00048590
000331       80    CONTINUE                                          00048600
000331             M=M+1                                              00048610
000333             M2 = HRTP (M,KX)                                   00048620
000336             M3=HRTP(M+1,KX)                                    00048630
000341             ID(M)=IDRTP(M,KX)                                  00048640
             C     ***************************                       00048650
             C     ***************************                       00048660
000344             CALL RAND(IU,IX,1,1,TAD)                           00048670
             C     ***************************                       00048680
             C     ***************************                       00048690
000347             TAU  = TAD*SIGT+MUT                                00048700
000352             TAUS(M)=TAU                                        00048710
000354             TAU=TAU/3600.                                      00048720
000356             VA=VRTP(M,KX)                                      00048730
000361             TA=TWPT(M)                                         00048740
000362             TR(M)=TR(M)+TAU                                    00048750
000365             TRA=TR(M)                                          00048760
000366             TAS=TA*3600.                                       00048770
000370             TRAS=TRA*3600.                                     00048780
000371             VRA=VR(M)                                          00048790
000373             IF(J.EQ.NFX) GO TO 610                             00048800
000377                VB=VRTP(M+1,KX)                                 00048810
000402                TB=TWPT(M+1)                                    00048820
000403                VAV=(VA+VB)/2.0                                 00048830
000406                TAB=TB-TA                                       00048840
000410                IF(TAB.NE.0.) GO TO 120                         00048850
000411                   TR(M+1)=TR(M)                                00048860
000413                   GO TO 80                                     00048870
000414       120      CONTINUE                                        00048880
000414                TRAB=TB-TRA                                     00048890
000416                VAVR=VAV * TAB/TRAB                             00048900
000421                VRB=2.0 * VAVR-VRA                              00048910
000423                AR=(VRB-VRA)/TRAB                               00048920
000426                TRB=TRA+TRAB                                    00048930
000427                VR(M+1)=VRB                                     00048940
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

000431                TR(M+1)=TRD                                          00048950
000432                ACR(M)=AR                                            00048960
000434                DAD =DRTF(M+1,KX)-DRTF(M,KX)                         00048970
000442                IFLGD=0                                              00048980
000443                DO 440 K=1,JMAX                                      00048990
000444                   HK1=HD(K)                                         00049000
000446                   HK2=HD(K+1)                                       00049010
000447                   DVFAST=VFAST(K+1)-VFAST(K)                        00049020
000451                   DVSLO =VSLOW(K+1)-VSLOW(K)                        00049030
000453                   IF (H3.LE.HD(1)) GO TO 410                        00049040
000456                      IF(K.EQ.JMAX) GO TO 410                        00049050
000457                         IF(H3.GT.HK2)GO TO 440                      00049060
000462     410           CONTINUE                                          00049070
000462                   DH=H3-HK1                                         00049080
000464                   DHD=HK2-HK1                                       00049090
000466                   VMXD=VFAST(K)+ DVFAST * DH/DHD                    00049100
000472                   VMND=VSLOW(K)+DVSLO*DH/DHD                        00049110
000475                   GO TO 450                                         00049120
000476     440           CONTINUE                                          00049130
000501     450           CONTINUE                                          00049140
000501                   IF(VRD.LE.VMXD) GO TO 460                         00049150
000504                      VRD =VMXD                                      00049160
000504                      IFLGD=2                                        00049170
000505                      GO TO 480                                      00049180
000506     460           CONTINUE                                          00049190
000506                   IF(VRD.GE.VMND) GO TO 470                         00049200
000511                      VRD=VMND                                       00049210
000511                      IFLGD=1                                        00049220
000512                      GO TO 480                                      00049230
000513     470           CONTINUE                                          00049240
000513     480           CONTINUE                                          00049250
           C                                                              00049260
           C             CHECK REVISED VELOCITY  TO CONSTRAIN VRD TO RTP VELOCITY  00049270
           C                                                              00049280
000513                   VUP = VD *(1 + GAMMA)                             00049290
000517                   VDOWN = VD*(1.-GAMMA)                             00049300
000521                   IF(VRD.GT.VUP)IFLGD = 4                           00049310
000524                   IF(VRD.GT.VUP)VRD = VUP                           00049320
000527                   IF(VRD.LT.VDOWN) IFLGD = 3                        00049330
000533                   IF(VRD.LT.VDOWN) VRD = VDOWN                      00049340
000536                   IF(IFLGD.EQ.0) GO TO 490                          00049350
000537                      ICNW=ICNW+1                                    00049360
000541                      VAVR=(VRA+VRD)/2.0                             00049370
000544                      TAD=DAD/VAVR                                   00049380
000545                      TRD=TRA+TAD                                    00049390
000547                      AR=(VRD-VRA)/TAD                               00049400
000552                      VR(M+1)=VRD                                    00049410
000553                      TR(M+1)=TRD                                    00049420
000555                      ACR(M)=AR                                      00049430
000556                      TAS=TA*3600.                                   00049440
000560                      TDS=TD*3600.                                   00049450
000562                      TRAS=TRA*3600.                                 00049460
000563                      TRDS=TRD*3600.                                 00049470
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

000565     490       CONTINUE                                                    00049480
000565               IF(M.EQ.1) GO TO 720                                        00049490
000567                     IF(ID(M).EQ.J) GO TO 730                              00049500
000571     720       CONTINUE                                                    00049510
000571               IF (ICHK.NE.0) GO TO 725                                    00049520
000572                  PRINT 2080, M,TAS,TRAS,TA,VR(M),VMND,VMXD,IFLGD          00049530
000616    2080       FORMAT(1H ,3X,I2,2X,F8.2,3X,F8.2,14X,3(F7.2,3X),F7.2,8X,I2) 00049540
000616     725       CONTINUE                                                    00049550
000616               GO TO 80                                                    00049560
000621     730       CONTINUE                                                    00049570
000621               TSCH(J)=TRTP(M,KX)                                          00049580
000626               PZI=PSI(J)                                                  00049590
000627               IF(PZI.LT.0.) GO TO 500                                     00049600
000630                  AR=-ACR(M-1)                                             00049610
000632                  IF(M.EQ.1)AR=0.                                          00049620
000635                  IF(AR.EQ.0.) GO TO 520                                   00049630
000636                     RAD=SQRT(VRA*VRA+2.0*AR*PZI)                          00049640
000644                  DELT=(-VRA+RAD)/AR                                       00049650
000647                  DELT=-DELT                                               00049660
000650                  GO TO 560                                                00049670
000652     500       CONTINUE                                                    00049680
000652               PZI=ABS(PZI)                                                00049690
000654               AR=ACR(M)                                                   00049700
000656               IF(AR.EQ.0.) GO TO 520                                      00049710
000657                  RAD=(VRA*VRA+2.0*AR*PZI)                                 00049720
000662                  RAD=SQRT(RAD)                                            00049730
000665                  DELT=(-VRA+RAD)/AR                                       00049740
000670                  GO TO 560                                                00049750
000672     520       CONTINUE                                                    00049760
000672               PZI=-PZI                                                    00049770
000673               DELT=PZI/VRA                                                00049780
000675     560       CONTINUE                                                    00049790
000675               TFR(J)=TRA+DELT                                             00049800
000700               TFR(J)=TFR(J)*3600.                                         00049810
000702               TRS=TR(M)*3600.                                             00049820
000704               VFR(J)=VRA                                                  00049830
000706               ICFLG(J)=ICHW                                               00049840
000707               IF (ICHK.NE.0) GO TO 565                                    00049850
000710                  PRINT 2090, M,TAS,TRAS,TFR(J),VA,VR(M),VMND,VMXD,IFLGD   00049860
000736     565       CONTINUE                                                    00049870
000736               GO TO 600                                                   00049880
000741     610    CONTINUE                                                      00049890
000741            PZI=PSI(J)                                                    00049900
000743            AR=ACR(M-1)                                                   00049910
000745            IF(AR.EQ.0) GO TO 630                                         00049920
000746               IF(PZI.LT.0) GO TO 630                                     00049930
000747                  AR=-AR                                                  00049940
000750                  RAD=SQRT(VRA* VRA +2.0 * AR * PZI)                      00049950
000756                  DELT=(VRA-RAD)/AR                                       00049960
000761                  GO TO 650                                               00049970
000763     630    CONTINUE                                                      00049980
000763            PZI=-PZI                                                      00049990
000764            DELT=PZI/VRA                                                  00050000
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

000766     650     CONTINUE                                                    00050010
000766             TFR(J)=TRA+DELT                                             00050020
000771             TFR(J)=TFR(J)*3600.0                                        00050030
000773             TR(J)=TR(J)*3600.0                                          00050040
000775             VFR(J)=VRA                                                  00050050
000776             TSCH(J)=TRTP(M,K,J)                                         00050060
001002             ICFLG(J)=ICNW                                               00050070
001004             IF (ICHK.NE.0) GO TO 800                                    00050080
001005                PRINT 2090, M,TAS,TRAS,TFR(J),VA,VR(M),VMND,             00050090
                1     VMXB,IFLGB                                               00050100
001032    2090        FORMAT(1H ,3X,I2,2X,3(F8.2,3X),3(F7.2,3X),F7.2,8X,I2)    00050110
001032     800 CONTINUE                                                        00050120
001037         RETURN                                                          00050130
001037         END                                                             00050140
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                                                                  00015060
                FUNCTION FACT(N)                                  00015070
          C                                                       00015080
000003            FACT=1                                          00015090
000005            DO 10 I=1, N                                    00015100
000006               FACT=FACT*I                                  00015110
000010         10 CONTINUE                                        00015120
000012            RETURN                                          00015130
000013            END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE FASLOW (JQUAD, INUMD,II, SUMTF, SUMTS)                  00021270
              C                                                                    00021280
              C     THIS ROUTINE COMPUTES THE FAST AND SLOW TIME REQUIRED TO FLY FROM 00021290
              C     E.F. TO I.A.F. THE ROUTINE COMPUTES THE TIME BTWN EACH WAY PT PAIR00021300
              C     (START FROM I.A.F  END) AND THEN SUM THEM UP TO OBTAIN TOTAL TIME 00021310
              C                                                                    00021320
000010                COMMON /AERCOT/                                            00021330
                     1    HD(10),          HCRIT,          JMAX,          SPKNOT,  00021340
                     2    VFAST(10),       VSLOW(10)                             00021350
000010                COMMON /CNSTNT/                                            00021360
                     1    ACCEL,           GRADI,          ITMAX,         PI,      00021370
                     2    RTPTST                                                00021380
000010                COMMON /GEOMOT/                                            00021390
                     1    ANGEF(4,3),      ANGFAF(2),      ANGIAF(4),     ANGMF(2), 00021400
                     2    ANGCM(2),        ANGTF(4),       DEF(4,3,5),    DFAF(2),  00021410
                     3    DIAF(4),         DMF(2),         DCM(2),        DPATH(4,3,5), 00021420
                     4    DTF(4),          DTH(2)                                00021430
000010                COMMON /RTPCV/                                             00021440
                     1    DDP,             DLF,            HIAF,          I,       00021450
                     2    INFIX,           IQUAD,          IS,            TIAF     00021460
000010                COMMON /TRAFOT/                                            00021470
                     1    ALT(200),        EELT(200),      ELLT(200),     GWA(200), 00021480
                     2    GWD(100),        ITYPEA(200),    ITYPED(100),   KEY(200), 00021490
                     3    LFIX(200),       NACA,           NACD,          SDT(100), 00021500
                     4    SLT(200),        SPEED(200),     TSECAA(200),   TSECAD(100), 00021510
                     5    TSECSA(200),     TSECSD(100),    VDUGG(200),    VREF(200) 00021520
              C                                                                    00021530
              C     HMAX      MAX. HEIGHT AT E.F. IN THOUSAND FEET               00021540
              C     DGS       GLIDE SLOPE DISTANCE FOR THIS A/C IN N.MILES       00021550
              C     DLF       LEVEL FLIGHT DISTANCE BTWN E.F. AND HITTING GLIDE SLOPE 00021560
              C     DDP       DECELLERATION LEVEL FLIGHT PATH BTWN GLIDE SLOPE AND IAF 00021570
              C     INUMD     ENTRY FIX NUMBER OF ARRIVAL AIRCRAFT BEING CONSIDERED 00021580
              C                                                                    00021590
              C     COMPUTE MAX. ALLOWED ARRIVAL ALTITUDE AT E.F. W.R.T. GIVEN GEOMTRY00021600
              C     CONFIGURATION                                                00021610
              C                                                                    00021620
000010                IS = II                                                    00021630
000011                I  = II                                                    00021640
000012                INFIX = INUMD                                              00021650
000013                IQUAD = JQUAD                                              00021660
              C                                                                    00021670
              C     HIAF IS A *CUTE* CALCULATION. E.G., IAF ALT. FOR EF 1 IS 10K FT. 00021680
              C                                                                    00021690
000014                HIAF = 9. + FLOAT(INUMD)                                   00021700
000016                HMAX=GRADI*(DEF(JQUAD,INUMD,1)-DDP)+HIAF                   00021710
000023                H=ALT(II)                                                  00021720
000025                IF (H.GT.HMAX) H=HMAX                                      00021730
000030                DGS=(H-HIAF) /GRADI                                        00021740
000033                DLF=DEF(JQUAD,INUMD,1)-DGS-DDP                             00021750
              C                                                                    00021760
              C     THE FOLLOWING STEPS COMPUTE THE SLOW TIME BTWN E.F. AND I.A.F. VIA00021770
              C     THE SHORTEST PATH (PRIMARY PATH)                             00021780
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

          C     STUFF1 IS USED IF THE ENTRY FIX ALT. (H) IS .GT. THE CRITICAL ALT.00021790
          C                                                                      00021800
000037            IF (H.LE.HCRIT) CALL STUFF3 (VSLCW(JMAX))                    00021810
000050            IF (H.GT.HCRIT) CALL STUFF1 (H, VSLCW(JMAX))                 00021820
000062            CALL TIMEND (II, SUMTF)                                      00021830
          C                                                                      00021840
          C     THE FOLLOWING STEPS COMPUTE THE FAST TIME BTWN E.F. AND I.A.F. VIA00021850
          C     THE LONGEST PATH. STUFF2 IS USED IF H.GT.HCRIT.                00021860
          C                                                                      00021870
000064            DEFMAX=DEF(JQUAD, INUMD, 5)                                  00021880
000072            IF (DEFMAX.LT.DEF(JQUAD, INUMD, 4))DEFMAX=DEF(JQUAD, INUMD, 4) 00021890
000077            DLF=DEFMAX-DGS-DDP                                           00021900
000102            IF (H.LE.HCRIT) CALL STUFF3 (VFAST(JMAX))                    00021910
000113            IF (H.GT.HCRIT) CALL STUFF2 (VFAST(JMAX))                    00021920
000123            CALL TIMEND (II, SUMTF)                                      00021930
000125            RETURN                                                       00021940
000126            END                                                          00021950
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE FLIGTX(IFLITX, NSAMP)                                  00040930
      C                                                                           00040940
      C     THIS SUBROUTINE COMPUTES THE TIMES AND VELOCITIES AT VARIOUS          00040950
      C     FIX POINTS FOR BOTH ARRIVALS AND DEPARTURES - IT ALSO COMPU-          00040960
      C     TES TIME DIFFENCE AND DISTANCE DIFFERENCE PARAMETERS AND PRE-         00040970
      C     PARES THE DATA FOR OUTPUT                                             00040980
      C                                                                           00040990
      C     INTERNAL VARIABLES                                                    00041000
      C                                                                           00041010
      C     LOP       LAST OPERATION       = 0  NO OPERATION                    00041020
      C     NOP       NEXT OPERATION         1  ARRIVAL                         00041030
      C                                      2  DEPARTURE                       00041040
      C     TRW       TIME ON RUNWAY FOR NEXT OPERATION                         00041050
      C     TLRW      TIME ON RUNWAY FOR LAST OPERATION                         00041060
      C     CTOF      CURRENT TIME FOR AIRCRAFT  AT THE  ITH CONTROL FIX        00041070
      C     PTOF      TIME OF PREVIOUS AIRCRAFT  AT THE  ITH CONTROL FIX        00041080
      C     CVOF      CURRENT VELOCITY FOR AIRCRAFT AT CONTROL FIX I            00041090
      C     PVOF      VELOCITY FOR PREVIOUS AIRCRAFT AT CONTROL FIX I           00041100
      C     IARR      POINTER TO DETERMINE PLACE IN ARRIVAL LIST                00041110
      C     JDEP      POINTER TO DETERMINE PLACE IN DEPARTURE LIST              00041120
      C     KARR      POINTER TO DETERMINE PLACE IN THRESHOLD-ORDERED ARR LIST  00041130
      C     LL        DESIGNATES ENTRY FIX (1,2,3,4)                            00041140
      C     VDEPAV    AVERAGE VELOCITY FOR DEPARTURES(KNOTS)                    00041150
      C                                                                           00041160
000005          COMMON /ERORCT/                                                   00041170
               1    ICFLG(7),       TFR(7),         TSCH(7),        VFR(7)      00041180
000005          COMMON /FLGTXC/                                                   00041190
               1    DISSEP(16,17), DLY(6,17),      IHIST(5),       NAR,        00041200
               2    ND(16),         NDOR(16),       NDEOR(6),       NDEL(6),    00041210
               3    NDEP,           NHIST,          NOOF(36),       NP(36),     00041220
               4    NRD(4),         NRDOR(4),       NRECOD(5),      NRT(4),     00041230
               5    NRTOR(4),       NT(16),         NTOR(16),       OPTIM(4,17), 00041240
               6    SUM(6),         SUMSQ(6),       SUMTIM,         TIMSEP(16,17), 00041250
               7    TIPATH(36,17)                                                00041260
000005          INTEGER             DISSEP,         DLY,            OPTIM,      00041270
               1    TIMSEP,         TIPATH                                       00041280
000005          COMMON /GEOMIN/                                                   00041290
               1    DPARAL,         EFX(4,3),       EFY(4,3),       EFZ(4,3),   00041300
               2    FAFX(2),        FAFY(2),        FAFZ(2),        IAFX(4),    00041310
               3    IAFY(4),        IAFZ(4),        MAX(2),         MAY(2),     00041320
               4    MAZ(2),         MFX(2),         MFY(2),         MFZ(2),     00041330
               5    NEF,            NPATH,          NQUAD,          NRNWY,      00041340
               6    OMX(2),         OMY(2),         OMZ(2),         RHOEF,      00041350
               7    RHOFAF,         RHOIAF,         RHOMF,          RHOOM,      00041360
               8    RHOTF,          TFX(4),         TFY(4),         TFZ(4),     00041370
               9    THX(2),         THY(2),         THZ(2)                      00041380
000005          REAL                IAFX,           IAFY,           IAFZ,       00041390
               1    MAX,            MAY,            MAZ,            MFX,        00041400
               2    MFY,            MFZ                                         00041410
000005          COMMON /STATIN/                                                   00041420
               1    DCRIT,          DLYLEL,         DLYINT,         DMIN(5),    00041430
               2    DSLEL,          DSINT,          NDLY,           NDSEP,      00041440
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                3    NOPSTF,          NTARR,          NTSEP,          OPDEL,          00041450
                4    OPINT,           TARDEL,         TARINT,         TMIN(5),        00041460
                5    TSDEL,           TSINT                                           00041470
000005           COMMON /TRAFIN/                                                      00041480
                1    ALTPA1(10),      ALTPA2(10),     CALT (1' 10),   CVREF(10,10),   00041490
                2    CWALT(10,10),    CWVREF(10,10),  ENTRY(10,4,3),  HOUR,           00041500
                3    INDALT(10),      INDSPD(10),     INDTA,          INDTD,          00041510
                4    INDWA(10),       INDWD(10),      NCALT(10),      NCVREF(10),     00041520
                5    NMTYPA(10),      NMTYPD(10),     NTYPEA,         NTYPED,         00041530
                6    SPDPA1(10),      SPDPA2(10),     SPSIGM,         SWTRAF,         00041540
                7    TAPAR1,          TAPAR2,         TASIGM,         TDPAR1,         00041550
                8    TDPAR2,          TDSIGM,         TYPEA(10),      TYPED(10),      00041560
                9    WAPAR1(10),      WAPAR2(10),     WDPAR1(10),     WDPAR2(10)      00041570
000005           INTEGER              SWTRAF                                          00041580
000005           COMMON /TRAFOT/                                                      00041590
                1    ALT(200),        EELT(200),      ELLT(200),      GWA(200),       00041600
                2    GWD(100),        ITYPEA(200),    ITYPED(100),    KEY(200),       00041610
                3    LFIX(200),       NACA,           NACD,           SDT(100),       00041620
                4    SLT(200),        SPEED(200),     TSECAA(200),    TSECAD(100),    00041630
                5    TSECSA(200),     TSECSD(100),    VDUGG(200),     VREF(200)       00041640
000005           DIMENSION            CTCF(16),       CVCF(16),       HISTRY(66),     00041650
                1    IAD(2,2),        NTCF(16),       PTCF(16),       PVCF(16)        00041660
000005           INTEGER              HISTRY,         QUAD                            00041670
000005           DATA                 VDEPAV          /160./                          00041680
000005           DATA                 (IAD(I),I=1,4) /4, 3, 2, 1/                     00041690
          C                                                                          00041700
          C      TAPE 1 CONTAINS VELOCITIES AND TIMES AT EVERY FIX                   00041710
          C      POINT FOR ALL AIRCRAFT IN THE SAMPLE                                00041720
          C      INITIALIZE COUNTERS AND PARAMETERS                                  00041730
          C                                                                          00041740
000005            IMOD = 0                                                         00041750
000006            LOF=0                                                            00041760
000007            NOP = 0                                                          00041770
000010            TRW = 0                                                          00041780
000011            TLRW =0                                                          00041790
          C                                                                          00041800
          C      SET UP SAMPLE PRINT VARIABLE AND PRINT HEADINGS IF REQUIRED         00041810
          C                                                                          00041820
000012            ICHK = MOD(NSAMP,IFLITX)                                         00041830
000015            IF (ICHK.EQ.0) PRINT 9000                                        00041840
000024    9000    FORMAT(1H1//30X, 36H***** FLIGHT SIMULATION OUTPUT ***** //      00041850
                1        35X, 26HCONSTRAINT VIOLATION CODES //                    00041860
                2        15X, 49H1 - COMPUTED VELOCITY LESS THAN MINIMUM ALLOWABLE 00041870
                3        9H VELOCITY /                                            00041880
                4        15X,52H2 - COMPUTED VELOCITY GREATER THAN MINIMUM ALLOWABLE00041890
                5        9H VELOCITY /                                            00041900
                6        15X, 45H3 - COMPUTED VELOCITY LESS THAN 95 PERCENT OF    00041910
                7        19H SCHEDULED VELOCITY /                                 00041920
                8        15X, 46H4 - COMPUTED VELOCITY GREATER THAN 95 PERCENT OF 00041930
                9        19H SCHEDULED VELOCITY )                                 00041940
          C                                                                          00041950
          C      ZERO OUT VELOCITY AND TIME ARRAYS                                   00041960
          C                                                                          00041970
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000024          DO 10 I=1,16                                              00041980
000026             CTCF(I)=0.                                             00041990
000027             PTCF(I)=0.                                             00042000
000030             CVCF(I)=0.                                             00042010
000031             PVCF(I)=0.                                             00042020
000032             NTCF(I)=0                                              00042030
000033       10 CONTINUE                                                  00042040
             C                                                           00042050
             C     START LOOP FOR EVERY AIRCRAFT IN SAMPLE               00042060
             C     INITIALIZE COUNTERS                                   00042070
             C                                                           00042080
000035          IARR = 1                                                 00042090
000036          JDEP = 1                                                 00042100
000037          IWR=0                                                    00042110
000040          ISAM = NSAMP / IFLITX                                    00042120
000042          NHIST = ISAM                                             00042130
000043          IF(ISAM.GT.5) ICHK =1                                    00042140
000046          IF (ISAM.GT.5) NHIST=5                                   00042150
000051        5 CONTINUE                                                 00042160
000051          LOP = NOP                                                00042170
000053          KARR = KEY(IARR)                                         00042180
000055          IF(ICHK.NE.0)GO TO 7                                     00042190
000056             DO 9 I=1,66                                           00042200
000057                HISTRY(I)= 0                                       00042210
000060        9    CONTINUE                                             00042220
000062        7 CONTINUE                                                 00042230
000062          IF (JDEP.GT.NACD) GO TO 12                               00042240
             C                                                           00042250
             C        IS AN ARRIVAL OR DEPARTURE FIRST IN LIST           00042260
             C                                                           00042270
000066             IF (SLT(KARR).GT.SDT(JDEP)) GO TO 15                  00042280
             C                                                           00042290
000072       12 CONTINUE                                                00042300
             C                                                           00042310
             C     ARRIVAL IS FIRST                                      00042320
             C                                                           00042330
000072          NOP = 1                                                  00042340
000073          GO TO 6                                                  00042350
             C                                                           00042360
             C     DEPARTURE IS FIRST                                    00042370
             C                                                           00042380
000074       15 CONTINUE                                                00042390
000074          NOP = 2                                                  00042400
000075          TLRW=TRW                                                 00042410
000077          TRW=SDT(JDEP)                                            00042420
000100          IF(ICHK.NE.0)GO TO 130                                   00042430
000101          IH = IARR + JDEP-1                                       00042440
             C                                                           00042450
             C     SET UP DEPARTURE HISTORY VARIABLES                    00042460
             C                                                           00042470
000103          HISTRY(1) =  IH                                          00042480
000104          HISTRY(2)= IFIX(TSECAD(JDEP))                            00042490
000106          HISTRY(3) =  ITYPED(JDEP)                                00042500
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000110          HISTRY(4)= 1HD                                          00042510
000111          DO 145 LLI=5,66                                         00042520
000113             HISTRY(LLI)= 0                                       00042530
000114      145 CONTINUE                                                00042540
000116          WRITE(1)(HISTRY(I),I=1,66)                              00042550
000121          IWR = IWR + 1                                           00042560
000125          GO TO 130                                               00042570
            C                                                           00042580
            C     DETERMINE VELOCITIES AND TIMES AT CONTROL FIXES USED  00042590
            C     BY THIS AIRCRAFT                                      00042600
            C                                                           00042610
000127        6 CONTINUE                                                00042620
000127          IMOD = IMOD + 1                                         00042630
000131          CALL ERROR(KARR,ICHK,IMOD)                              00042640
000133          QUAD = LFIX(KARR)/100                                   00042650
000137          LL = LFIX(KARR)/100                                     00042660
000141          IF(LL.NE.1)GO TO 30                                     00042670
            C                                                           00042680
            C        FIRST QUADRANT (LL =1)                             00042690
            C        SET UP ARRAY INDEXES                               00042700
            C                                                           00042710
000144             IIAF = 1                                             00042720
000145             ITF  = 5                                             00042730
000146             IMF  = 9                                             00042740
000147       36    CONTINUE                                             00042750
000147             IFAF = 11                                            00042760
000150             ICM = 13                                             00042770
000151             ITH = 15                                             00042780
000152             GO TO 35                                             00042790
000153       30 CONTINUE                                                00042800
000153          IF(LL.NE.4)GO TO 31                                     00042810
            C                                                           00042820
            C        FOURTH QUADRANT INDEXES                            00042830
            C                                                           00042840
000155             IIAF = 4                                             00042850
000156             ITF = 8                                              00042860
000157             IMF = 9                                              00042870
000160             GO TO 36                                             00042880
000161       31 CONTINUE                                                00042890
000161          IF(LL.NE.2)GO TO 32                                     00042900
            C                                                           00042910
            C        SECOND QUADRANT INDEXES                            00042920
            C                                                           00042930
000163             IIAF =2                                              00042940
000164             ITF = 6                                              00042950
000165             IMF = 10                                             00042960
            C                                                           00042970
            C        CHECK FOR NUMBER OF RUNWAYS USED                   00042980
            C                                                           00042990
000166       38    CONTINUE                                             00043000
000166             IF(NRNWY .EQ.1) GO TO 36                             00043010
000170                IFAF = 12                                         00043020
000171                ICM = 14                                          00043030
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000172                ITH = 16                                          00043040
000173                GO TO 35                                          00043050
          C                                                             00043060
          C         THIRD QUADRANT INDEXES                              00043070
          C                                                             00043080
000174       32 CONTINUE                                                00043090
000174          IIAF = 3                                                00043100
000175          ITF = 7                                                 00043110
000176          IMF = 10                                                00043120
000177          GO TO 38                                                00043130
000200       35 CONTINUE                                                00043140
          C                                                             00043150
          C     CHECK INDEX OF EACH FIX POINT AND STORE IN APPROPIATE   00043160
          C     ARRAY AND LOCATION                                      00043170
          C                                                             00043180
000200          IF(ICHK.NE.0) GO TO 155                                 00043190
          C                                                             00043200
          C         SET UP HISTOGRAM ARRAY FOR ARRIVALS                 00043210
          C                                                             00043220
000201            HISTRY(1) = IARR + JDEP - 1                           00043230
000204            HISTRY(2)= IFIX(TSECAA(KARR))                         00043240
000206            HISTRY(3) = ITYPEA (KARR)                             00043250
000210            HISTRY(4)= 1HA                                        00043260
000211            HISTRY(5) = LFIX(KARR)/10                             00043270
000214            HISTRY(6) = LFIX(KARR)- 10 *(LFIX(KARR)/10)           00043280
000221            QUAD = LFIX(KARR)/100                                 00043290
000224            NUM  = (LFIX(KARR) - 100 * QUAD)/10                   00043300
000230            INDX = 3 *(QUAD-1) + NUM                              00043310
000232            IEFA = 2* (INDX-1) + 7                                00043320
000234            IEFB = IEFA + 1                                       00043330
000236            IAFA = 2* (IIAF -1)+ 31                               00043340
000240            IAFB = IAFA + 1                                       00043350
000241            ITFA = 2* (ITF- 5) + 39                               00043360
000244            ITFB = ITFA + 1                                       00043370
000246            IMFA = 2* (IMF- 9) + 47                               00043380
000251            IMFB = IMFA + 1                                       00043390
000252            IFAFA= 2* (IFAF-11) + 51                              00043400
000255            IFAFB= IFAFA + 1                                      00043410
000257            ICMA = 2* (ICM-13) + 55                               00043420
000262            ICMB = ICMA + 1                                       00043430
000263            ITHA = 2*(ITH-15) + 59                                00043440
000266            ITHB = ITHA + 1                                       00043450
000270            HISTRY(IEFA) = IFIX(TFR(1))                           00043460
000273            HISTRY(IAFA) = IFIX(TFR(2))                           00043470
000276            HISTRY(ITFA) = IFIX(TFR(3))                           00043480
000301            HISTRY(IMFA) = IFIX(TFR(4))                           00043490
000304            HISTRY(IFAFA)= IFIX(TFR(5))                           00043500
000307            HISTRY(ICMA) = IFIX(TFR(6))                           00043510
000312            HISTRY(ITHA) = IFIX(TFR(7))                           00043520
000315            HISTRY(IEFB) = IFIX(TSCH(1))                          00043530
000320            HISTRY(IAFB) = IFIX(TSCH(2))                          00043540
000323            HISTRY(ITFB) = IFIX(TSCH(3))                          00043550
000326            HISTRY(IMFB) = IFIX(TSCH(4))                          00043560
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

000331            HISTRY(IFAFD)= IFIX(TSCH(5))                    00043570
000334            HISTRY(ICMD) = IFIX(TSCH(6))                    00043580
000337            HISTRY(ITHD) = IFIX(TSCH(7))                    00043590
000342            WRITE(1)(HISTRY(I),I=1,66)                      00043600
000347            IWR = IWR + 1                                   00043610
000351       155 CONTINUE                                         00043620
000351          DO 40 I= 2,7                                      00043630
000354            IJ = I-1                                        00043640
000356            GO TO (41,42,43,44,45,46),IJ                    00043650
                C                                                 00043660
                C         IAF FIX                                 00043670
                C                                                 00043680
000370       41   CONTINUE                                        00043690
000370            II = IIAF                                       00043700
000372            GO TO 50                                        00043710
                C                                                 00043720
                C         TF FIX                                  00043730
                C                                                 00043740
000372       42   CONTINUE                                        00043750
000372            II = ITF                                        00043760
000374            GO TO 50                                        00043770
                C                                                 00043780
                C         MF FIX                                  00043790
                C                                                 00043800
000374       43   CONTINUE                                        00043810
000374            II = IMF                                        00043820
000376            GO TO 50                                        00043830
                C                                                 00043840
                C         FAF FIX                                 00043850
                C                                                 00043860
000376       44   CONTINUE                                        00043870
000376            II = IFAF                                       00043880
000400            GO TO 50                                        00043890
                C                                                 00043900
                C         OM FIX                                  00043910
                C                                                 00043920
000400       45   CONTINUE                                        00043930
000400            II = IOM                                        00043940
000402            GO TO 50                                        00043950
                C                                                 00043960
                C         TH FIX                                  00043970
                C                                                 00043980
000402       46   CONTINUE                                        00043990
000402            II = ITH                                        00044000
000404       50   CONTINUE                                        00044010
                C                                                 00044020
                C         STORE PREVIOUS AIRCRAFT DATA            00044030
                C                                                 00044040
000404            PTCF(II) = CTCF(II)                             00044050
000406            PVCF(II) = CVCF(II)                             00044060
                C                                                 00044070
                C         STORE PRESENT AIRCRAFT DATA             00044080
                C                                                 00044090
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000410              CTCF(II) = TFR(I)                                        00044100
000412              CVCF(II) = VFR(I)                                        00044110
000413              NCCF(II) = NCCF(II) + 1                                  00044120
000415              NTCF(II)=NTCF(II)+1                                      00044130
                C                                                            00044140
                C   CHECK IF PRESENT AIRCRAFT IS THE FIRST AIRCRAFT        00044150
                C   OF THE SAMPLE                                          00044160
                C                                                            00044170
000417              IF (NTCF(II).LE.1) GO TO 40                              00044180
                C                                                            00044190
                C      IS NOT THE FIRST AIRCRAFT                             00044200
                C      COMPUTE DIFFERENCE IN TIME AND DIFFERENCE IN DISTANCE 00044210
                C      TIME                                                  00044220
                C                                                            00044230
000420                 DT = CTCF(II) - PTCF(II)                             00044240
                C                                                            00044250
                C      DISTANCE                                              00044260
                C                                                            00044270
000422                 DD = DT * PVCF(II)/3600.                             00044280
                C                                                            00044290
                C      STORE TIME DIFFERENCE IN ARRAY TIMSEP                 00044300
                C                                                            00044310
000425                 XDT = DT                                             00044320
000426                 CALL HST(TSINT,TSDEL,XDT,NTSEP,N)                    00044330
000431                 TIMSEP(II,N) = TIMSEP(II,N) + 1                      00044340
                C                                                            00044350
                C      COMPARE TIME SEPARATION WITH MINIMUM TIME SEPARATION (TMIN) 00044360
                C                                                            00044370
000435                 IF(XDT.LT.TMIN(5))GO TO 54                           00044380
                C                                                            00044390
                C         STORE NON- VIOLATIONS                              00044400
                C                                                            00044410
000441                    NT(II) = NT(II) + 1                               00044420
000443                    GO TO 55                                          00044430
                C                                                            00044440
                C      STORE VIOLATIONS                                      00044450
                C                                                            00044460
000443          54     CONTINUE                                             00044470
000443                 NTCR(II) = NTCR(II) + 1                              00044480
                C                                                            00044490
                C      CHECK DISTANCE SEPARATION AND STORE                   00044500
                C                                                            00044510
000446          55     CONTINUE                                             00044520
000446                 XDD = DD                                             00044530
000450                 CALL HST(DSINT,DSDEL,XDD,NDSEP,N)                    00044540
                C                                                            00044550
                C      STORE DISTANCE SEPARATION                             00044560
                C                                                            00044570
000453                 DISSEP(II,N) = DISSEP(II,N) + 1                      00044580
                C                                                            00044590
                C      CHECK IF DIST SET HAS VIOLATED MIN IN AIR SEPARATION  00044600
                C                                                            00044610
000457                 IF(XDD.LT.DMIN(5))GO TO 62                           00044620
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

      C                                                                  00044630
      C              STORE IN-AIR DISTANCE NON-VIOLATIONS                00044640
      C                                                                  00044650
000463                ND(II) = ND(II) +1                                 00044660
000465                GO TO 65                                           00044670
      C                                                                  00044680
      C            STORE IN-AIR DISTANCE VIOLATIONS                     00044690
      C                                                                  00044700
000465    62          CONTINUE                                           00044710
000465                NDCR(II) = NDCR(II) + 1                            00044720
000470    65          CONTINUE                                           00044730
000470    40 CONTINUE                                                    00044740
      C                                                                  00044750
      C     SET UP RUNWAY TIMES                                          00044760
      C                                                                  00044770
000472        TLRW = TRW                                                 00044780
000474        TRW = TFR(7)                                               00044790
      C                                                                  00044800
      C     UPDATE ARRIVALS COUNTER                                      00044810
      C                                                                  00044820
000475        IARR = IARR + 1                                            00044830
      C                                                                  00044840
      C     DETERMINE WHICH OF THE 36 TRACKS WAS USED                    00044850
      C                                                                  00044860
000476        IX = LFIX(KARR) /100                                       00044870
000502        IY =(LFIX(KARR)- 100 * IX)/10                              00044880
000506        IZ =(LFIX(KARR)- 100 * IX - 10 * IY)/2 + 1                 00044890
000513        M=9*(IX-1) + 3*(IY-1) + IZ                                 00044900
      C                                                                  00044910
      C     STORE PATH USAGE                                             00044920
      C                                                                  00044930
000517        NP(M) = NP(M) + 1                                          00044940
      C                                                                  00044950
      C     COMPUTE TIME AIRCRAFT WAS IN THE PATH                        00044960
      C                                                                  00044970
000521        DTIME = TRW - TSECAA(KARR)                                 00044980
      C                                                                  00044990
      C     STORE IN APPROPIATE TABLE                                    00045000
      C                                                                  00045010
000523        CALL HST(TARINT,TARDEL,DTIME,NTARR,N)                      00045020
000526        TIPATH(M,N) = TIPATH(M,N) + 1                              00045030
000533        LFIX(KARR)=LFIX(KARR)/10                                   00045040
      C                                                                  00045050
      C     IAD IS AN OPTION ARRAY TO DETERMINE SEQUENSE OF OPERATIONS   00045060
      C     NOP = 1 , LOP = 1     IAD(1,1)= 4                          00045070
      C     NOP = 2 , LOP = 1     IAD(1,2)= 3                          00045080
      C     NOP = 1 , LOP = 2     IAD(2,1)= 2                          00045090
      C     NOP = 2 , LOP = 2     IAD(2,2) = 1                         00045100
      C     DETERMINE SEQUENCE                                           00045110
      C                                                                  00045120
000537    130 CONTINUE                                                   00045130
000537        IF(LOP.EQ.0)GO TO 80                                       00045140
000542           I= IAD(LOP,NOP)                                         00045150
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                C
                C                                                                00045160
                C       COMPUTE DIFFERENCE OF TIME ON RUNWAY FOR SUCCESIVE OPERATIONS  00045170
                C                                                                00045180
000544                  TIO = ABS(TRW - TLRW)                                    00045190
000547                  CALL HST(OPINT,OPDEL,TIO,NOSTP,N)                        00045200
                C                                                                00045210
                C       STORE IN OPTIM ARRAY                                     00045220
                C                                                                00045230
000553                  OPTIM(I,N) = OPTIM(I,N) + 1                              00045240
                C                                                                00045250
                C       CHECK IF DIFF TIME ON RUNWAY(TIO) VIOLATES MINIMUN TIME  00045260
                C                                                                00045270
000557                  IF(TIO.LT.TMIN(I))GO TO 90                               00045280
                C                                                                00045290
                C          STORE NON-VIOLATIONS                                  00045300
                C                                                                00045310
000563                     NRT(I) = NRT(I) + 1                                   00045320
000564                     GO TO 92                                              00045330
                C                                                                00045340
                C       STORE VIOLATIONS                                         00045350
                C                                                                00045360
000565        90    CONTINUE                                                     00045370
000565              NRTOR(I) = NRTOR(I) + 1                                      00045380
000570        92    CONTINUE                                                     00045390
                C                                                                00045400
                C       COMPUTE SUCCESIVE OPERATIONS DISTANCE SEPARATION         00045410
                C                                                                00045420
000570                  IF (I.EQ.3) GO TO 98                                     00045430
000572                     IF(I.EQ.1)GO TO 96                                    00045440
                C                                                                00045450
                C             DEP/ARR  AND  ARR/ARR                              00045460
                C                                                                00045470
000574                        DIO = TIO * VDUGG(KARR)                           00045480
000576                        DIC=DIO/3600.                                     00045490
000600                        GO TO 100                                         00045500
000600        96           CONTINUE                                            00045510
000600                     DIO = TIO * VDEPAV                                  00045520
000602                     DIC=DIO/3600.                                       00045530
000604                     GO TO 100                                           00045540
000604        98        CONTINUE                                               00045550
000604                  DIO = TIO * VDUGG(KARR)/2                              00045560
000610                  DIC=DIO/3600.                                          00045570
                C                                                                00045580
                C       CHECK IF DISTANCE SEPARATION VIOLATES MINIMUM            00045590
                C                                                                00045600
000612      100    CONTINUE                                                      00045610
000612              IF(DIO.LT.DMIN(I))GO TO 102                                  00045620
                C                                                                00045630
                C       COUNT NON-VIOLATIONS                                     00045640
                C                                                                00045650
000615                  NRD(I) = NRD(I) + 1                                      00045660
000617                  GO TO 80                                                 00045670
                C                                                                00045680
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

      C       COUNT VIOLATIONS                                          00045690
      C                                                                 00045700
000620  102   CONTINUE                                                  00045710
000620        NRDCR(I) = NRDCR(I) + 1                                   00045720
000623    80 CONTINUE                                                   00045730
      C                                                                 00045740
      C     SET UP VARIABLES TO COMPUTE POSSIBLE DELAY                  00045750
      C                                                                 00045760
000623        IF(NOP.EQ.1)GO TO 110                                     00045770
      C                                                                 00045780
      C       FOR DEPARTURES                                          00045790
      C                                                                 00045800
000625          SCDLY = SDT(JDEP) - TSECAD(JDEP)                        00045810
000630          CPDLY = 0                                               00045820
000631          NDEP = NDEP + 1                                         00045830
000632          INDX = 4                                                00045840
000633          JDEP = JDEP + 1                                         00045850
000633          TLDLY = SCDLY + CPDLY                                   00045860
000635          GO TO 112                                               00045870
      C                                                                 00045880
      C     FOR ARRIVALS                                              00045890
      C                                                                 00045900
000636  110 CONTINUE                                                    00045910
000636        SCDLY =  SLT(KARR) - EELT(KARR)                           00045920
000641        CPDLY =  TRW - SLT(KARR)                                  00045930
000643        NAR = NAR + 1                                             00045940
000644        INDX = 1                                                  00045950
000645        TLDLY=SCDLY+CPDLY                                         00045960
      C                                                                 00045970
      C     COMPUTE DELAY                                             00045980
      C                                                                 00045990
000647  112 CONTINUE                                                    00046000
000647        NNX = INDX + 2                                            00046010
000651        DO 115 I=INDX,NNX                                         00046020
000652          IF(I.EQ.2.OR.I.EQ.5) GO TO 117                          00046030
000661            IF(I.EQ.1.OR.I.EQ.4) GO TO 116                        00046040
000667              DELY = TLDLY                                        00046050
000670              GO TO 118                                           00046060
000671  116       CONTINUE                                              00046070
000671            DELY = SCDLY                                          00046080
000673            GO TO 118                                             00046090
000673  117     CONTINUE                                                00046100
000673          DELY = CPDLY                                            00046110
000675  118     CONTINUE                                                00046120
000675          CALL HST(DLYINT,DLYDEL,DELY,NDLY,N)                     00046130
      C                                                                 00046140
      C                                                                 00046150
      C       STORE DELAY                                             00046160
      C       DELAY TABLE CONSISTS OF                                 00046170
      C         DLY(1,I) ARRIVAL SCHEDULED DELAY                      00046180
      C         DLY(2,I) ARRIVAL OPERATIONAL DELAY                    00046190
      C         DLY(3,I) ARRIVAL TOTAL DELAY                          00046200
      C         DLY(4,I) DEPARTURE SCHEDULED DELAY                    00046210
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

C            DLY(5,1) DEPARTURE OPERATIONAL DELAY                    00046220
C            DLY(6,1) DEPARTURE TOTAL DELAY                          00046230
C         1  MEANS FIX                                               00046240
C                                                                    00046250
000701            DLY(I,N) = DLY(I,N) + 1                            00046260
C                                                                    00046270
C         SUM DELAY FOR ALL DATA SAMPLES                             00046280
C                                                                    00046290
000705            IF (DELY.LT.0) DELY=0.0                            00046300
000711            SUM(I)= SUM(I) + DELY                              00046310
C                                                                    00046320
C         SUM OF SQUARES OF DELAYS FOR ALL DATA SAMPLES              00046330
C                                                                    00046340
000714            SUMSQ(I) = SUMSQ(I) + DELY * DELY                  00046350
C                                                                    00046360
C         CHECK DELAY AGAINST MINIMUM TOLERABLE                      00046370
C                                                                    00046380
000716            IF(DELY.GT.DCRIT) GO TO 120                        00046390
C                                                                    00046400
C            NON - VIOLATIONS                                        00046410
C                                                                    00046420
000721               NDEL(I) = NDEL(I) + 1                           00046430
000722               GO TO 115                                       00046440
C                                                                    00046450
C         VIOLATIONS                                                 00046460
C                                                                    00046470
000723    120    CONTINUE                                            00046480
000723           NDECR(I) = NDECR(I) + 1                             00046490
000726    115 CONTINUE                                               00046500
000731        IF (IARR.LE.NACA) GO TO 5                              00046510
000733           IF (ICHK.NE.0) GO TO 123                            00046520
000734              IHIST(ISAM) = NSAMP                              00046530
000736              NRECCD(ISAM) = IWR                               00046540
00074D    123    CONTINUE                                            00046550
C                                                                    00046560
C         ARE ALL ARRIVALS IN SAMPLE CONSIDERED                      00046570
C                                                                    00046580
00074D            SUMTIM = SUMTIM + HOUR*3600                        00046590
000744            RETURN                                             00046600
000744         END                                                   00046610
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE GECMTY                                                00009100
      C                                                                          00009110
000002          COMMON /GECMIN/                                                  00009120
                1     DPARAL,         EFX(4,3),       EFY(4,3),       EFZ(4,3),  00009130
                2     FAFX(2),        FAFY(2),        FAFZ(2),        IAFX(4),   00009140
                3     IAFY(4),        IAFZ(4),        MAX(2),         MAY(2),    00009150
                4     MAZ(2),         MFX(2),         MFY(2),         MFZ(2),    00009160
                5     NEF,            NPATH,          NQUAD,          NRNWY,     00009170
                6     CMX(2),         CMY(2),         CMZ(2),         RHOEF,     00009180
                7     RHOFAF,         RHOIAF,         RHOMF,          RHOOM,     00009190
                8     RHOTF,          TFX(4),         TFY(4),         TFZ(4),    00009200
                9     THX(2),         THY(2),         THZ(2)                     00009210
000002          REAL                  IAFX,           IAFY,           IAFZ,      00009220
                1     MAX,            MAY,            MAZ,            MFX,       00009230
                2     MFY,            MFZ                                        00009240
000002          COMMON /GECMOT/                                                  00009250
                1     ANGEF(4,3),     ANGFAF(2),      ANGIAF(4),      ANGMF(2),  00009260
                2     ANGOM(2),       ANGTF(4),       DEF(4,3,5),     DFAF(2),   00009270
                3     DIAF(4),        DMF(2),         DOM(2),         DPATH(4,3,5), 00009280
                4     DTF(4),         DTH(2)                                     00009290
000002          COMMON /GECMTX/                                                  00009300
                1     DOFF(10),       DOFS(10),       DSEG(6),        NOFF,      00009310
                2     RHO(7),         THETA(6),       X(7),           Y(7)       00009320
      C                                                                          00009330
000002          NOFF = NPATH - 1                                                 00009340
000004          DOFF(1) = DPARAL                                                 00009350
000006          DOFF(2) = DPARAL                                                 00009360
000007          DOFF(3) = 2.*DPARAL                                              00009370
000010          DOFF(4) = DOFF(3)                                                00009380
000011          DO 800 I = 1,NQUAD                                               00009390
000013             DO 700 J = 1,NEF                                              00009400
000014                INDX = 1                                                   00009410
000015                IF ((NRNWY.EQ.2).AND.((I.EQ.2).OR.(I.EQ.3))) INDX = 2      00009420
000031                X(1) = THX(INDX)                                           00009430
000033                Y(1) = THY(INDX)                                           00009440
000035                X(2) = CMX(INDX)                                           00009450
000036                Y(2) = CMY(INDX)                                           00009460
000040                X(3) = FAFX(INDX)                                          00009470
000041                Y(3) = FAFY(INDX)                                          00009480
000043                RHO(1) = 0.                                                00009490
000044                RHO(2) = RHOOM                                             00009500
000045                RHO(3) = RHOFAF                                            00009510
000047                INDX = 1                                                   00009520
000050                IF ((I.EQ.2).OR.(I.EQ.3)) INDX = 2                         00009530
000060                X(4) = MFX(INDX)                                           00009540
000062                Y(4) = MFY(INDX)                                           00009550
000064                RHO(4) = RHOMF                                             00009560
000065                X(5) = TFX(I)                                              00009570
000067                Y(5) = TFY(I)                                              00009580
000071                RHO(5) = RHOTF                                             00009590
000072                X(6) = IAFX(I)                                             00009600
000074                Y(6) = IAFY(I)                                             00009610
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000075                RHO(6) = RHOIAF                                          00009620
000077                X(7) = EFX(I,J)                                          00009630
000101                Y(7) = EFY(I,J)                                          00009640
000103                RHO(7) = RHOEF                                           00009650
000105                CALL TRKGCM                                              00009660
000106                INDX = 1                                                 00009670
000107                IF ((NRNWY.EQ.2).AND.((I.EQ.2).OR.(I.EQ.3))) INDX = 2    00009680
000123                DTH(INDX) = 0.                                           00009690
000125                DCM(INDX) = DSEG(1)                                      00009700
000126                ANGCM(INDX) = THETA(1)                                   00009710
000130                DFAF(INDX) = DSEG(2)                                     00009720
000131                ANGFAF(INDX) = THETA(2)                                  00009730
000133                INDX = 1                                                 00009740
000134                IF ((I.EQ.2).OR.(I.EQ.3)) INDX = 2                       00009750
000144                DMF(INDX) = DSEG(3)                                      00009760
000146                ANGMF(INDX) = THETA(3)                                   00009770
000150                DTF(I) = DSEG(4)                                         00009780
000152                ANGTF(I) = THETA(4)                                      00009790
000153                DIAF(I) = DSEG(5)                                        00009800
000155                ANGIAF(I) = THETA(5)                                     00009810
000156                ANGEF(I,J) = THETA(6)                                    00009820
000161                DEF(I,J,1) = DSEG(6)                                     00009830
000164                DPATH(I,J,1) = DSEG(1) + DSEG(2)+DSEG(3)+DSEG(4)+DSEG(5)+ 00009840
     1                               DSEG(6)                                   00009850
000173                DO 600 K = 2,NPATH                                       00009860
000174                   DEF(I,J,K) = DCFS(K - 1)                              00009870
000201                   DPATH(I,J,K) = DSEG(1) + DSEG(2)+DSEG(3)+DSEG(4)+DSEG(5)+00009880
     1                                 DCFS(K-1)                               00009890
000213    600         CONTINUE                                                 00009900
000216    700       CONTINUE                                                   00009910
000220    800 CONTINUE                                                         00009920
000223        CALL GECMWT                                                      00009930
000224        RETURN                                                           00009940
000225        END                                                              00009950
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE GEOMWT                                                00011100
      C                                                                          00011110
000002          COMMON /GEOMIN/                                                  00011120
               1    DPARAL,         EFX(4,3),       EFY(4,3),       EFZ(4,3),   00011130
               2    FAFX(2),        FAFY(2),        FAFZ(2),        IAFX(4),    00011140
               3    IAFY(4),        IAFZ(4),        MAX(2),         MAY(2),     00011150
               4    MAZ(2),         MFX(2),         MFY(2),         MFZ(2),     00011160
               5    NEF,            NPATH,          NQUAD,          NRNWY,      00011170
               6    CMX(2),         CMY(2),         CMZ(2),         RHOEF,      00011180
               7    RHOFAF,         RHOIAF,         RHOMF,          RHOCM,      00011190
               8    RHOTF,          TFX(4),         TFY(4),         TFZ(4),     00011200
               9    THX(2),         THY(2),         THZ(2)                      00011210
000002          REAL                IAFX,           IAFY,           IAFZ,       00011220
               1    MAX,            MAY,            MAZ,            MFX,        00011230
               2    MFY,            MFZ                                         00011240
000002          COMMON /GEOMOT/                                                  00011250
               1    ANGEF(4,3),     ANGFAF(2),      ANGIAF(4),      ANGMF(2),   00011260
               2    ANGCM(2),       ANGTF(4),       DEF(4,3,5),     DFAF(2),    00011270
               3    DIAF(4),        DMF(2),         DCM(2),         DPATH(4,3,5), 00011280
               4    DTF(4),         DTH(2)                                      00011290
      C                                                                          00011300
000002          DO 30 K=1,NRNWY                                                  00011310
000004             DO 20 I=1, NQUAD                                              00011320
000005                PRINT 100,K                                                00011330
000012                PRINT 110,I                                                00011340
000020                IJ=1                                                       00011350
000021                IF(I.EQ.2.OR.I.EQ.3)IJ=2                                   00011360
000032                DO 10 J=1, NEF                                             00011370
000034                   DO 10 L=1, NPATH                                        00011380
000035                      PRINT 120,J,L,DEF(I,J,L),ANGEF(I,J),DPATH(I,J,L)     00011390
000062  10            CONTINUE                                                   00011400
000067                PRINT 130,DIAF(I),DTF(I),DMF(IJ),DFAF(K),DCM(K)            00011410
000105                PRINT 140,ANGIAF(I),ANGTF(I),ANGMF(IJ),ANGFAF(K),ANGCM(K)  00011420
000123  20         CONTINUE                                                      00011430
000126  30      CONTINUE                                                         00011440
000130          RETURN                                                           00011450
      C                                                                          00011460
000131  100     FORMAT(1H1///35X, 25H*****GEOMETRY OUTPUT***** //41X,10HRUNWAY NO.00011470
               1       ,I2//10X,  8HQUADRANT, 4X, 3HFIX, 2X, 8HPARALLEL, 5X,    00011480
               2        9HEF TO IAF, 4X,15HEF PRIMARY PATH, 2X,5HTOTAL /         00011490
               3       22X, 3HNO., 2X, 8HPATH NO., 5X, 8HDISTANCE, 8X,           00011500
               4       10HEXIT ANGLE, 3X, 8HDISTANCE /)                          00011510
000131  110     FORMAT(5X,I1O)                                                   00011520
000131  120     FORMAT(15X,I9,I7, 9X, F7.2, 9X, F7.2, 6X, F7.2)                  00011530
000131  130     FORMAT(/21X, 19HIAF TO TF  DISTANCE , F7.2, 9H N. MILES /        00011540
               1        21X, 19HTF  TO MF  DISTANCE , F7.2, 9H N. MILES /       00011550
               2        21X, 19HMF  TO FAF DISTANCE , F7.2, 9H N. MILES /       00011560
               3        21X, 19HFAF TO CM  DISTANCE , F7.2, 9H N. MILES /       00011570
               4        21X, 19HCM  TO TH  DISTANCE , F7.2, 9H N. MILES )       00011580
000131  140     FORMAT(/21X, 15HIAF EXIT VECTOR, 4X, F7.2,19H RADIANS FROM NORTH/ 00011590
               1        21X, 15HTF  EXIT VECTOR, 4X, F7.2,19H RADIANS FROM NORTH/ 00011600
               2        21X, 15HMF  EXIT VECTOR, 4X, F7.2,19H RADIANS FROM NORTH/ 00011610
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                3       21X, 15HFAF EXIT VECTOR, 4X, F7.2,19H RADIANS FROM NORTH/ 0001162D
                4       21X, 15HCM  EXIT VECTOR, 4X, F7.2,19H RADIANS FROM NORTH) 0001163D
                                                                                  0001164D
0001131          END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE HST(XINIT,XINC,XPOINT,NENT,INDX)                    00046620
          C                                                                    00046630
          C     THIS SUBROUTINE COMPUTE THE APPROPIATE INDEX FOR ENTRY .      00046640
          C     IN HISTOGRAM TABLES                                          00046650
          C     NENT      NUMBER OF SLOTS IN TABLE                           00046660
          C     XINIT     INITIAL TABLE VALUE                                00046670
          C     XINC      INCREMENT                                          00046680
          C     XPOINT    ENTRY VALUE                                        00046690
          C     INDX      INDEX IN TABLE                                     00046700
          C                                                                    00046710
000010            IF(XPOINT. LT. XINIT) GO TO 20                             00046720
000012               N= (XPOINT - XINIT) / XINC + 2                          00046730
000015               IF (N.GT.NENT) GO TO 10                                 00046740
000020                  INDX = N                                             00046750
000020                  RETURN                                               00046760
000021    10         CONTINUE                                               00046770
000021               INDX = NENT                                             00046780
000022               RETURN                                                  00046790
000023    20      CONTINUE                                                   00046800
000023            INDX = 1                                                   00046810
000024            RETURN                                                     00046820
000025            END                                                        00046830
```

TABLE 5-1.—CONTINUED

```
FTN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE LIMITS(IDIST,P1,P2,A,XCH)                              00013870
              C                                                                   00013880
              C     P1        LOWER LIMIT(UNIFORM)                                00013890
              C               MEAN        (NORMAL)                                00013900
              C     P2        UPPER       (UNIFORM)                               00013910
              C               SIGMA       (NORMAL)                                00013920
              C                                                                   00013930
              C     CHECK TYPE OF DISTRIBUTION DESIRED                            00013940
              C                                                                   00013950
000010                IF(IDIST.EQ.6HUNIFOM)GO TO 10                               00013960
000012                   IF(IDIST.EQ.6HNORMAL)GO TO 20                            00013970
000013                      IF(IDIST.EQ.6HERLANG)GO TO 30                         00013980
000015                         PRINT 100,IDIST                                    00013990
000022   100                   FORMAT(1HO//25HWRONG INPUT DATA IDIST =  ,A6)      00014000
000022                         STOP 2                                             00014010
000024   10           CONTINUE                                                    00014020
000024                IF(P1.LE.P2)GO TO 15                                        00014030
000031                   PRINT 105,P1,P2                                          00014040
000041   105             FORMAT(1HO,42HLOWER LIMIT GREATER THAN UPPER LIMIT.  L =  ,I5,  00014050
                     1   2X, 4HU =  ,I5,2X,15HLIMITS REVERSED )                  00014060
000041                TEMP = P1                                                   00014070
000045                   P1= P2                                                   00014080
000046                   P2= TEMP                                                 00014090
000047   15           CONTINUE                                                    00014100
000047                   CALL UNIFORM(A,P1,P2)                                    00014110
000051                   IF(A.GT.0.AND.A.LE.XCH) RETURN                           00014120
000065                GO TO 15                                                    00014130
000066   20           CONTINUE                                                    00014140
000066                   CALL NORMAL(A,P1,P2)                                     00014150
000071                   XLOW =  P1 - 3.* P2                                      00014160
000075                   XHIGH=  P1 + 3.* P2                                      00014170
000077                   IF(A.GT.XLOW.AND.A.LT.XHIGH) RETURN                      00014180
000110                GO TO 20                                                    00014190
000111   30           CONTINUE                                                    00014200
000111                CALL ERLANG(A,P1,P2)                                        00014210
000113                RETURN                                                      00014220
000114                END                                                         00014230
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION 00T 73 A          16.56.27. 73/12/21.

                SUBROUTINE NORMAL (A, F1, F2)                          00014390
          C                                                            00014400
          C     F1=MEAN, F2=STD                                        00014410
          C                                                            00014420
000006            COMMON /RANDOM/                                      00014430
                 1    IU,             IX                              00014440
          C                                                            00014450
          C     ******************************                         00014460
          C     ******************************                         00014470
000006            CALL RAND(IU,IX,1,1,U)                               00014480
          C     ******************************                         00014490
          C     ******************************                         00014500
000011            A = U * F2 + F1                                      00014510
000015            RETURN                                               00014520
000016            END                                                  00014530
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE RDAERO                                                                    00006550
                                                                                                     00006560
      C                                                                                              00006570
000002          COMMON /AERO1N/                                                                      00006580
               1    CADFRS(10),    HA(10,7),      HCRITN(10),    NAERO(10),                        00006590
               2    VFAST1(10,7),  VFAST2(10,7),  VSLOW1(10,7),  VSLOW2(10,7),                     00006600
               3    W1(10),        W2(10)                                                           00006610
000002          COMMON /TRAF1N/                                                                      00006620
               1    ALTPA1(10),    ALTPA2(10),    CALT (10,10),  CVREF(10,10),                     00006630
               2    CWALT(10,10),  CWVREF(10,10), ENTRY(10,4,3), HOUR,                             00006640
               3    INDALT(10),    INDSPD(10),    INDTA,         INDTD,                           00006650
               4    INDWA(10),     INDWD(10),     NCALT(10),     NCVREF(10),                      00006660
               5    NMTYPA(10),    NMTYPD(10),    NTYPEA,        NTYPED,                          00006670
               6    SPDPA1(10),    SPDPA2(10),    SPSIGM,        SWTRAF,                          00006680
               7    TAPAR1,        TAPAR2,        TASIGM,        TDPAR1,                          00006690
               8    TDPAR2,        TDSIGM,        TYPEA(10),     TYPED(10),                       00006700
               9    WAPAR1(10),    WAPAR2(10),    WDPAR1(10),    WDPAR2(10)                       00006710
000002          INTEGER            SWTRAF                                                           00006720
      C                                                                                              00006730
000002          READ 1110, (NAERO(I), I=1, NTYPEA)                                                   00006740
000015          PRINT 3520                                                                           00006750
000021          DO 1320 I=1, NTYPEA                                                                  00006760
000023             READ 1120, W1(I), W2(I), CADFRS(I), HCRITN(I)                                     00006770
000036             IF (MOD(I,3).EQ.0) PRINT 3510                                                     00006780
000046             PRINT 3530, I, NMTYPA(I), W1(I), W2(I), CADFRS(I)                                 00006790
000064             MAERO=NAERO(I)                                                                    00006800
000066             DO 1310 J=1, MAERO                                                                00006810
000070                READ 1120, HA(I,J), VSLOW1(I,J), VSLOW2(I,J), VFAST1(I,J),                     00006820
               1                VFAST2(I,J)                                                       00006830
000114                PRINT 3540, HA(I,J), VSLOW1(I,J), VSLOW2(I,J), VFAST1(I,J),                   00006840
               1                VFAST2(I,J)                                                       00006850
000137    1310 CONTINUE                                                                            00006860
000142    1320 CONTINUE                                                                            00006870
000144          RETURN                                                                             00006880
      C                                                                                              00006890
000145    1110 FORMAT(711D)                                                                        00006900
000145    1120 FORMAT(7F10.0)                                                                      00006910
000145    3510 FORMAT(1H1)                                                                         00006920
000145    3520 FORMAT(1H1/ 1HD, 33H  IV. AERO PERFORMANCE INPUT DATA)                              00006930
000145    3540 FORMAT(10X, F7.2, 7X, F7.2, 3(7X, F7.2))                                            00006940
000145    3530  FORMAT(1HD, 4X, I2, 3H.  , A4/1HD,                                                00006950
               1    9X,32HLIGHT LANDING WEIGHT (1000 LBS.), 14X          , F7.2 /                 00006960
               2   10X,37HMAX. GROSS LANDING WEIGHT (1000 LBS.), 9X      , F7.2 /                 00006970
               3   10X,40HMAX. CABIN PRESSURE DIFFERENTIAL (LBS/SQ. IN.) , F7.2 //                 00006980
               4       12X, 4HALT., 9X, 19HSLOW SPEED BOUNDARY, 9X,                                00006990
               5       19HHIGH SPEED BOUNDARY / 10X, 9HX1000 FT., 3X,                               00007000
               6       11HLIGHT GROSS, 2X, 14HMAX. GROSS WT., 1X,                                   00007010
               7       11HLIGHT GROSS, 2X, 14HMAX. GROSS WT., /                                     00007020
               8       25X, 5HKNOTS,9X, 5HKNOTS,9X, 5HKNOTS,9X, 5HKNOTS /)                          00007030
000145          END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE RDATC                                                        00007800
      C                                                                                 00007810
000002          COMMON /ATCIN/                                                          00007820
                1    DIST(4),        DRCO(10),       TRTFS,          HRTFS,           00007830
                2    IRWD(4),        ISCSW,          TARO(10),       TIME(4),         00007840
                3    TRTFS                                                           00007850
000002          COMMON /TRAFIN/                                                         00007860
                1    ALTFA1(10),     ALTFA2(10),     CALT (10,10),   CVREF(10,10),    00007870
                2    CWALT(10,10),   CWVREF(10,10),  ENTRY(10,4,3),  HCUR,           00007880
                3    INDALT(10),     INDSFD(10),     INDTA,          INDTD,           00007890
                4    INDWA(10),      INDWD(10),      NCALT(10),      NCVREF(10),      00007900
                5    NMTYPA(10),     NMTYFD(10),     NTYPEA,         NTYPED,          00007910
                6    SPDFA1(10),     SPDFA2(10),     SFSIGN,         SWTRAF,          00007920
                7    TAFAR1,         TAFAR2,         TASIGN,         TDFAR1,          00007930
                8    TDFAR2,         TDSIGN,         TYPEA(10),      TYPED(10),       00007940
                9    WAFAR1(10),     WAFAR2(10),     WDFAR1(10),     WDFAR2(10)       00007950
000002          INTEGER             SWTRAF                                           00007960
      C                                                                                 00007970
000002          READ 1110, ISCSW                                                        00007980
000010          READ 1120, (DIST(I), I = 1, 4)                                          00007990
000016          READ 1120, (TIME(I), I = 1, 4)                                          00008000
000024          READ 1110, (IRWD(I), I = 1, 4)                                          00008010
000032          READ 1120, (TARO(I),I=1,NTYPEA)                                         00008020
000045          READ 1120, (DRCO(I),I=1,NTYPED)                                         00008030
000060          READ 1120, DRTFS, TRTFS, HRTFS                                          00008040
      C                                                                                 00008050
000072    1110 FORMAT(7I10)                                                             00008060
000072    1120 FORMAT(7F10.0)                                                           00008070
      C                                                                                 00008080
000072          PRINT 3705, ISCSW, (DIST(I),I=1,4),(TIME(I),I=1,4),(IRWD(I),I=1,4)00008090
000106          IF (NTYPEA.NE.0) PRINT 3710, (NMTYPA(I),I=1,NTYPEA)                     00008100
000122          IF (NTYPEA.NE.0) PRINT 3711, (TARO(I),I=1,NTYPEA)                       00008110
000136          IF (NTYPED.NE.0) PRINT 3712, (NMTYFD(I),I=1,NTYPED)                     00008120
000152          IF (NTYPED.NE.0) PRINT 3713, (DRCO(I),I=1,NTYPED)                       00008130
000166          PRINT 3715, TRTFS, DRTFS, HRTFS                                         00008140
000200          RETURN                                                                  00008150
      C                                                                                 00008160
000201    3705 FORMAT(1H1// 31H    VI. ATC CONSTRAINT CRITERIA ///                      00008170
                1 6X, 37HSCHEDULER CONSTRAINT SWITCH IS SET TO , I2 //                  00008180
                2 6X,57H(SWITCH SET TO 1 IMPLIES ARRIVALS HAVE ABSOLUTE PRIORITY)/      00008190
                3 6X,57H(SWITCH SET TO 2 IMPLIES ARRIVALS HAVE PRIORITY          )/     00008200
                4 6X,57H(SWITCH SET TO 3 IMPLIES MIXED OPERATIONS HAVE PRIORITY )//00008210
                5 /6X, 19HSEPARATION CRITERIA , 7X, 21HA/A   A/D   D/A   D/D  //       00008220
                6 6X, 8HDISTANCE, 17X, 4(F4.1,2X), 6H. MILES /                          00008230
                7 6X, 4HTIME, 21X, 4(F4.0, 2X), 7HSECONDS /                             00008240
                8 6X, 16HRUNWAY OCCUPANCY, 9X, 4(I4, 2X) //                             00008250
                9 6X, 56H(NOTE -- VALUE IS 1 IF NO JOINT RUNWAY OCCUPANCY APPLIES /00008260
                A 6X, 16HAND 0 OTHERWISE) )                                             00008270
000251    3710 FORMAT(1H0/6X, 43HARRIVAL AIRCRAFT TYPE RUNWAY OCCUPANCY TIME //         00008280
                1 6X, 4HTYPE, 11X, 10(A4,1X) )                                          00008290
000261    3711 FORMAT(6X, 11HTIME (SECS), 2X, 10F5.0)                                   00008300
000261    3712 FORMAT(1H0/6X, 45HDEPARTURE AIRCRAFT TYPE RUNWAY OCCUPANCY TIME //00008310
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION 66T 73 A          16.56.27. 73/12/21.

                1 6X, 4HTYPE, 11X, 10(A4, 1X) )                                  00008320
                                                                                 00008330
000201      3713 FORMAT(6X, 11HTIME (SECS), 2X, 10F5.0 )                         00008340
000201      3715 FORMAT(1H0/6X, 43HMINIMAL TIME SEPARATION FOR RTP          = ,  00008350
                1              F5.1, 8H SECONDS/                                 00008360
                2        6X, 43HMINIMAL DISTANCE SEPARATION FOR RTP     = ,     00008370
                3              F5.1, 8H N.MILES/                                 00008380
                4        6X, 43HMINIMAL ALTITUDE SEPARATION FOR RTP     = ,     00008390
                5              F5.1, 7H 1000FT)                                  00008400
000201          END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                                                                        00008410
                SUBROUTINE RDAVNC                                       00008420
      C                                                                 00008430
000002          COMMON /AVICN/                                          00008440
               1    MURO,          MUT,          SIGRO,          SIGT   00008450
000002          REAL               MURO,         MUT                    00008460
      C                                                                 00008470
000002          READ 1120, MURO, SIGRO, MUT, SIGT                       00008480
000016    1120  FORMAT(7F10.0)                                          00008490
      C                                                                 00008500
000016          PRINT 3805, MURO, SIGRO                                 00008510
000026          PRINT 3810, MUT, SIGT                                   00008520
000036          RETURN                                                  00008530
      C                                                                 00008540
000037    3805  FORMAT(1HO/1HO/1HO, 33H   VII. AVIONICS PERFORMANCE DATA//  00008550
               1       6X, 43HMEAN VALUE OF NAVIGATION ERROR DIST.     = ,  00008560
               2             F5.2, 8H N.MILES/                         00008570
               3       6X, 43HSTANDARD DEVIATION OF NAV.ERROR DIST.    = ,  00008580
               4             F5.2, 8H N.MILES)                         00008590
000037    3810  FORMAT(6X, 43HMEAN VALUE OF CONTROL SYSTEM ERROR DIST. = ,  00008600
               1             F5.1, 8H SECONDS/                         00008610
               2       6X, 43HSTANDARD DEVIATION OF CONTROL ERROR DST. = ,  00008620
               3             F5.1, 8H SECONDS)                         00008630
000037          END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE RDDSCR
           C
000002          COMMON /RANDOM/
               1     IU,            IX
000002          COMMON /TRAFIN/
               1     ALTFA1(10),     ALTFA2(10),     CALT  (10,10),  CVREF(10,10),
               2     CWALT(10,10),   CWVREF(10,10),  ENTRY(10,4,3),  HCUR,
               3     INDALT(10),     INDSFD(10),     INDTA,          INDTD,
               4     INDWA(10),      INDWD(10),      NCALT(10),      NCVREF(10),
               5     NMTYFA(10),     NMTYFD(10),     NTYFEA,         NTYFED,
               6     SFDFA1(10),     SFDFA2(10),     SFSIGM,         SWTRAF,
               7     TAFAR1,         TAFAR2,         TASIGM,         TDFAR1,
               8     TDFAR2,         TDSIGM,         TYFEA(10),      TYFED(10),
               9     WAFAR1(10),     WAFAR2(10),     WDFAR1(10),     WDFAR2(10)
000002          INTEGER              SWTRAF
000002          COMMON /TRAFOT/
               1     ALT(200),       EELT(200),      ELLT(200),      GWA(200),
               2     GWD(100),       ITYFEA(200),    ITYFED(100),    KEY(200),
               3     LFIX(200),      NACA,           NACD,           SDT(100),
               4     SLT(200),       SPEED(200),     TSECAA(200),    TSECAD(100),
               5     TSECSA(200),    TSECSD(100),    VDUGG(200),     VREF(200)
           C
000002          READ 1110, NACA, NACD, NTYFEA, NTYFED, IU, IX
000022          READ 1140, (NMTYFA(I), I=1, NTYFEA)
000035          READ 1140, (NMTYF      I=1, NTYFED)
000050          READ 1110, (NCVRE( ), I=1, NTYFEA)
000063          READ 1210, (TSECSA(I), ITYFEA(I), GWA(I), ALT(I),
               1                SPEED(I), LFIX(I), I=1, NACA)
000110          READ 1215, (TSECSD(I), ITYFED(I), GWD(I), I=1, NACD)
000127          DO 1200 I = 1, NACD
000131             TSECSD(I) = TSECSD(I) + 1600.
000134     1200 CONTINUE
000136          READ 1120, TASIGM, TDSIGM, SFSIGM
000147          DO 1220  J=1, NTYFEA
000151             NV=NCVREF(J)
000153             READ 1120, (CVREF(J,I), I=1, NV)
000166             READ 1120, (CWVREF(J,I), I=1, NV)
000202     1220 CONTINUE
           C
000205     1110 FORMAT(7I10)
000205     1120 FORMAT(7F10.0)
000205     1140 FORMAT(7(A4, 6X))
000205     1210 FORMAT(F10.0, A4, 6X, 3F10.0, I10)
000205     1215 FORMAT(F10.0, A4, 6X, F10.0)
           C
000205          HCUR=(TSECSA(NACA)-TSECSA(1))/3600.
000210          NPLANE=NACA+NACD
000212          PRINT 3010, NPLANE, NTYFEA, TASIGM, TDSIGM, SFSIGM
000227          PRINT 3020
000233          DO 1230 J=1, NTYFEA
000235             NV=NCVREF(J)
000237             PRINT 3030, NMTYFA(J), (CVREF(J,I), I=1, NV)
```

TABLE 5-1.–CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000254            PRINT 3040, (CWVREF(J,I), I=1, NV)                          00003390
000270     1230 CONTINUE                                                      00003400
000273          RETURN                                                        00003410
           C                                                                  00003420
000273     3010  FORMAT(1HO// 47H    I. TRAFFIC GENERATOR INPUT DATA -- DISCRETE/  00003430
                1 1HO,6X,24HTOTAL NO. OF AIRCRAFT  = , I5 /                 00003440
                2      7X,24HNUMBER OF A/C TYPES   = , I5 /                 00003450
                3      7X,24HARRIVAL TIME STD DEV  = , F8.2, 8H SECONDS /   00003460
                4      7X,24HDEPARTURE TIME STD DEV = , F8.2, 8H SECONDS /  00003470
                5      7X,24HA/C SPEED STD DEV     = , F8.2, 5H MACH /)     00003480
000273     3020 FORMAT(7X, 47HDATA PTS OF VREF/WEIGHT CURVE FOR EACH A/C TYPE)  00003490
000273     3030 FORMAT(1HO, 10X,  A4, 4X, 12HVREF (KNOTS), 4X, 10F8.2)        00003500
000273     3040 FORMAT(19X, 16HWEIGHT (1000LBS), 10F8.2)                       00003510
000273          END                                                           00003520
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE REDSTR                                                    00003530
          C                                                                          00003540
000002            COMMON /RANDOM/                                                  00003550
                 1    IU,             IX                                          00003560
000004            COMMON /TRAFIN/                                                  00003570
                 1    ALTPA1(10),     ALTPA2(10),     CALT (10,10),   CVREF(10,10), 00003580
                 2    CWALT(10,10),   CWVREF(10,10),  ENTRY(10,4,3),  HOUR,       00003590
                 3    INDALT(10),     INDSPD(10),     INDTA,          INDTD,       00003600
                 4    INDWA(10),      INDWD(10),      NCALT(10),      NCVREF(10),  00003610
                 5    NMTYPA(10),     NMTYPD(10),     NTYPEA,         NTYPED,      00003620
                 6    SPDPA1(10),     SPDPA2(10),     SPSIGM,         SWTRAF,      00003630
                 7    TAPAR1,         TAPAR2,         TASIGM,         TDPAR1,      00003640
                 8    TDPAR2,         TDSIGM,         TYPEA(10),      TYPED(10),   00003650
                 9    WAPAR1(10),     WAPAR2(10),     WDPAR1(10),     WDPAR2(10)   00003660
000002            INTEGER             SWTRAF                                       00003670
000002            COMMON /TRAFOT/                                                  00003680
                 1    ALT(200),       EELT(200),      ELLT(200),      GWA(200),    00003690
                 2    GWD(100),       ITYPEA(200),    ITYPED(100),    KEY(200),    00003700
                 3    LFIX(200),      NACA,           NACD,           SDT(100),    00003710
                 4    SLT(200),       SPEED(200),     TSECAA(200),    TSECAD(100), 00003720
                 5    TSECSA(200),    TSECSD(100),    VDUGG(200),     VREF(200)    00003730
          C                                                                          00003740
          C     NENTRY   NO. OF ENTRY PTS (CONTROL FIX)                            00003750
          C                                                                          00003760
000002            READ 1110, NTYPEA, NTYPED, NENTRY, IU, IX                        00003770
000020            READ 1110, (NCVREF(I), I=1, NTYPEA)                              00003780
000033            READ 1110, (NCALT(I), I=1, NTYPEA)                               00003790
000046            READ 1140, (NMTYPA(I), I=1, NTYPEA)                              00003800
000061            READ 1140, (NMTYPD(I), I=1, NTYPED)                              00003810
000074            READ 1120, TAPAR1, TAPAR2, TDPAR1, TDPAR2, HOUR                  00003820
000112            READ 1120, (TYPEA(I), I=1, NTYPEA)                               00003830
000125            READ 1120, (TYPED(I), I=1, NTYPED)                               00003840
000140            READ 1130, INDTA, INDTD                                          00003850
000150            READ 1130, (INDWA(I), I=1,NTYPEA)                                00003860
000163            READ 1130, (INDWD(I), I=1,NTYPED)                                00003870
000176            READ 1130, (INDALT(I), I=1, NTYPEA)                              00003880
000211            READ 1130, (INDSPD(I), I=1, NTYPEA)                              00003890
000224            DO 1260 I=1, NTYPEA                                              00003900
000226               READ 1120,  WAPAR1(I), WAPAR2(I), ALTPA1(I), ALTPA2(I),       00003910
                 1              SPDPA1(I), SPDPA2(I)                              00003920
000245    1260 CONTINUE                                                           00003930
000250         DO 1265 I=1, NTYPED                                                00003940
000251            READ 1120,  WDPAR1(I), WDPAR2(I)                                00003950
000260    1265 CONTINUE                                                           00003960
000262         DO 1270 J=1, NTYPEA                                                00003970
000264            NV=NCVREF(J)                                                    00003980
000266            NA=NCALT(J)                                                     00003990
000267            READ 1120, (CVREF(J,I), I=1, NV)                                00004000
000303            READ 1120, (CWVREF(J,I), I=1, NV)                               00004010
000317            READ 1120, (CALT(J,I), I=1, NA)                                 00004020
000333            READ 1120, (CWALT(J,I), I=1, NA)                                00004030
000347            DO 1270 IQ=1, 4                                                 00004040
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000351                READ 1120, (ENTRY(J,IQ,IN), IN=1, 3)                    00004050
000365    1270 CONTINUE                                                       00004060
          C                                                                   00004070
000372    1110 FORMAT(7I10)                                                   00004080
000372    1120 FORMAT(7F10.0)                                                 00004090
000372    1130 FORMAT(7(A6,4X))                                               00004100
000372    1140 FORMAT(7(A4, 6X))                                              00004110
          C                                                                   00004120
000372         PRINT 9000, INDTA, INDTD                                       00004130
000401         PRINT 9001, TAPAR1, TAPAR2, TDPAR1, TDPAR2                     00004140
000415         PRINT 9002                                                     00004150
000421         PRINT 9003, ((NAMTYPA(I), INDWA(I), WAPAR1(I), WAPAR2(I),      00004160
              1             INDALT(I), ALTPA2(I), ALTPA1(I), INDSPD(I), SPDPA2(I),00004170
              2             SPDPA1(I), TYPEA(I)), I = 1, NTYPEA)             00004180
000460         PRINT 9004                                                     00004190
000464         PRINT 9005, ((NAMTYPD(I), INDWD(I), WDPAR1(I), WDPAR2(I), TYPED(I))00004200
              1             ,I = 1, NTYPED)                                  00004210
000507         PRINT 9006                                                     00004220
000513         DO 3230 J=1, NTYPEA                                            00004230
000515            NV=NCVREF(J)                                                00004240
000517            NA=NCALT(J)                                                 00004250
000520            PRINT 9007, NAMTYPA(J), (CVREF(J,I), I=1, NV)               00004260
000536            PRINT 9008, (CWREF(J,I), I=1, NV)                           00004270
000552            PRINT 9009, (CALT(J,I), I=1, NA)                            00004280
000566            PRINT 9010, (CWALT(J,I), I=1, NA)                           00004290
000602    3230 CONTINUE                                                       00004300
000605         RETURN                                                         00004310
          C                                                                   00004320
000605    9000 FORMAT(1H0,49H  I. TRAFFIC GENERATOR INPUT DATA -- DISTRIBUTION/ 00004330
              1      1H0,6X,46HARRIVAL INTER-OPERATION TIME DISTRIBUTION   = ,A6/00004340
              2       7X, 46HDEPARTURE INTER-OPERATION TIME DISTRIBUTION = , A6) 00004350
000605    9001 FORMAT(1H0,6X,27HMEAN ARRIVAL OPERATION TIME, 17X, 1H=,F7.1,   00004360
              1       6H SECONDS /                                           00004370
              2       7X, 33HSTD. DEVIATION OR ORDER PARAMETER , 11X, 1H=, F7.1, 00004380
              3       6H SECONDS /                                           00004390
              4       7X, 29HMEAN DEPARTURE OPERATION TIME, 15X, 1H=, F7.1,  00004400
              5       6H SECONDS /                                           00004410
              6       7X, 33HSTD. DEVIATION OR ORDER PARAMETER, 11X, 1H=, F7.1, 00004420
              7       6H SECONDS )                                           00004430
000605    9002 FORMAT(1H0/7X, 21HARRIVAL AIRCRAFT DATA//7X, 4HTYPE,5X,        00004440
              1       20HW E I G H T  D A T A ,5X, 24HA L T I T U D E  D A T A , 00004450
              2       6X, 16HS P E E D  D A T A , 7X, 10HPERCENT OF /14X,     00004460
              3       3(5HDIST., 2X, 7HMEAN OR, 3X, 8HS. D. OR, 2X), 2X,     00004470
              4       8HARRIVALS /14X,3(4HTYPE,3X,7HMAXIMUM,3X,7HMINIMUM,3X)/ 00004480
              5       20X, 2(19H(1000LBS) (1000LBS) ,6X), 2X, 2(4HMACH,6X)/) 00004490
000605    9003 FORMAT(7X,A4,2X,2(F5.1,2X,F6.1,4X,F6.1,3X),A6,3X,F4.2,6X,F4.2,8X, 00004500
              1       F5.1)                                                  00004510
000605    9004 FORMAT(1H0, 6X, 23HDEPARTURE AIRCRAFT DATA //7X, 4HTYPE, 5X,    00004520
              1       20HW E I G H T  D A T A , 6X, 10HPERCENT OF /14X,       00004530
              2       5HDIST., 2X, 31HMEAN OR   S. D. OR   DEPARTURES  /     00004540
              3       14X, 4HTYPE, 3X, 17HMAXIMUM   MINIMUM /20X,            00004550
              4       19H(1000LBS) (1000LBS) / )                             00004560
000605    9005 FORMAT(7X, A4, 2X, A6, 2X, 2(F6.1, 4X), 3X, F5.1)              00004570
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

0000C5      9006 FORMAT(1H1, 6X, 43HDATA FTS OF VREF/WEIGHT CURVE AND ALTITUDE/,   00004580
                1       29HWEIGHT CURVE OF EACH A/C TYPE)                         00004590
0000C5      9007 FORMAT(1HO, 13X, A4, 4X, 12HVREF (KNOTS), 5X, 10F8.2)            00004600
0000C5      9008 FORMAT(22X, 17HWEIGHT (1000LBS) ,10F8.2)                         00004610
0000C5      9009 FORMAT(1HO, 21X, 17HALTITUTE (1000FT), 10F8.2)                   00004620
0000C5      9010 FORMAT(22X, 17HWEIGHT (1000LBS) , 10F8.2)                        00004630
0000C5           END                                                              00004640
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                    SUBROUTINE RDGEOM                                                    00005650
              C                                                                          00005660
000002              COMMON /GEOMIN/                                                      00005670
                   1    DPARAL,          EFX(4,3),        EFY(4,3),        EFZ(4,3),   00005680
                   2    FAFX(2),         FAFY(2),         FAFZ(2),         IAFX(4),    00005690
                   3    IAFY(4),         IAFZ(4),         MAX(2),          MAY(2),     00005700
                   4    MAZ(2),          MFX(2),          MFY(2),          MFZ(2),     00005710
                   5    NEF,             NPATH,           NQUAD,           NRNWY,      00005720
                   6    CMX(2),          CMY(2),          CMZ(2),          RHOEF,      00005730
                   7    RHOFAF,          RHOIAF,          RHOMF,           RHOOM,      00005740
                   8    RHOTF,           TFX(4),          TFY(4),          TFZ(4),     00005750
                   9    THX(2),          THY(2),          THZ(2)                       00005760
000002              REAL                  IAFX,            IAFY,            IAFZ,       00005770
                   1    MAX,             MAY,             MAZ,             MFX,        00005780
                   2    MFY,             MFZ                                          00005790
              C                                                                          00005800
000002              READ 1110, NQUAD, NRNWY, NEF, NPATH                                  00005810
000016              DO 1290 I=1, NQUAD                                                   00005820
000020                 READ  1120, (EFX(I,J), J=1,NEF), IAFX(I), TFX(I)                  00005830
000037                 READ  1120, (EFY(I,J), J=1,NEF), IAFY(I), TFY(I)                  00005840
000057                 READ  1120, (EFZ(I,J), J=1,NEF), IAFZ(I), TFZ(I)                  00005850
000077   1290 CONTINUE                                                                   00005860
000102              READ 1120, MFX(1), MFY(1), MFZ(1), MFX(2), MFY(2), MFZ(2)            00005870
000121              DO 1300 I=1, NRNWY                                                   00005880
000123                 READ 1120, FAFX(I), CMX(I), THX(I), MAX(I)                        00005890
000136                 READ 1120, FAFY(I), CMY(I), THY(I), MAY(I)                        00005900
000152                 READ  1120,  FAFZ(I), CMZ(I), THZ(I), MAZ(I)                      00005910
000166   1300 CONTINUE                                                                   00005920
000171              READ  1120, RHOEF, RHOIAF, RHOTF, RHOMF, RHOFAF, RHOOM, DPARAL       00005930
              C                                                                          00005940
000212   1110 FORMAT(7I10)                                                             00005950
000212   1120 FORMAT(7F10.0)                                                           00005960
              C                                                                          00005970
000212              PRINT 3260, NRNWY, NQUAD, NEF, NPATH                                 00005980
000226              PRINT  3270                                                          00005990
000232              DO 3290 I=1, NQUAD                                                   00006000
000234                 DO 3290 J = 1, NEF                                                00006010
000235                    PRINT  3280, I, J, EFX(I,J), EFY(I,J), EFZ(I,J)                00006020
000256   3290 CONTINUE                                                                   00006030
000263              PRINT  3300                                                          00006040
000267              DO 3320 I=1, NQUAD                                                   00006050
000271                 PRINT  3310, I, IAFX(I), IAFY(I), IAFZ(I)                         00006060
000304   3320 CONTINUE                                                                   00006070
000307              PRINT 3330                                                           00006080
000312              DO 3350 I=1, NQUAD                                                   00006090
000314                 PRINT 3340, I, TFX(I), TFY(I), TFZ(I)                             00006100
000327   3350 CONTINUE                                                                   00006110
000332              PRINT 3360                                                           00006120
000335              PRINT 3370, MFX(1), MFY(1), MFZ(1), MFX(2), MFY(2), MFZ(2)           00006130
000355              PRINT 3390, FAFX(1), FAFY(1), FAFZ(1)                                00006140
000367              IF (NRNWY.GE.2) PRINT 3400, FAFX(2), FAFY(2), FAFZ(2)                00006150
000403              PRINT 3420, CMX(1), CMY(1), CMZ(1)                                   00006160
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000415          IF (NRNWY.GE.2) PRINT 3430, CMX(2), CMY(2), CMZ(2)                    00006170
000431          PRINT 3450, THX(1), THY(1), THZ(1)                                    00006180
000443          IF (NRNWY.GE.2) PRINT 3460, THX(2), THY(2), THZ(2)                    00006190
000457          PRINT 3480, MAX(1), MAY(1),MAZ(1)                                     00006200
000471          IF (NRNWY.GE.2) PRINT 3490, MAX(2), MAY(2), MAZ(2)                    00006210
000505          PRINT 3510, RHOEF, RHOMF, RHOIAF, RHOFAF, RHOTF, RHOCM, DPARAL        00006220
000527          RETURN                                                                00006230
          C                                                                           00006240
000530     3260 FORMAT (1H1// 38H  III. FLIGHT PATH/GEOMETRY INPUT DATA//             00006250
               1        7X, 33HNUMBER OF RUNWAYS                =, I3 /               00006260
               2        7X, 33HNUMBER OF ARRIVAL QUADRANTS      =, I3 /               00006270
               3        7X, 33HNUMBER OF ENTRY FIXES/QUADRANT   =, I3 /               00006280
               4        7X, 33HNUMBER OF PARALLEL OFFSET PATHS =, I3 /)               00006290
000530     3280  FORMAT(11X, 3HEF , 2I1, 3X, 3F11.3)                                 00006300
000530     3270  FORMAT(/7X,26HENTRY FIX (EF) COORDINATES/26X,1HX,10X,1HY,10X,1HZ/)   00006310
000530     3300  FORMAT(/7X,42HINTERMEDIATE ARRIVAL FIX (IAF) COORDINATES /)          00006320
000530     3310  FORMAT(11X, 4HIAF , I1, 3X, 3F11.3)                                 00006330
000530     3330  FORMAT(/7X,25HTURN FIX (TF) COORDINATES/)                           00006340
000530     3340  FORMAT(11X, 2HTF , I3, 3X, 3F11.3)                                  00006350
000530     3360  FORMAT(/7X,26HMERGE FIX (MF) COORDINATES/)                          00006360
000530     3370  FORMAT(11X, 5HMF  1, 3X, 3F11.3 / 11X, 5HMF  2, 3X, 3F11.3)         00006370
000530     3390  FORMAT(/7X,36HFINAL APPROACH FIX (FAF) COORDINATES //               00006380
               1       11X, 5HFAF 1, 3X, 3F11.3)                                     00006390
000530     3400  FORMAT(11X, 5HFAF 2, 3X, 3F11.3)                                    00006400
000530     3420  FORMAT(/7X,29HOUTER MARKER (OM) COORDINATES//11X,5HOM  1,3X,        00006410
               1       3F11.3)                                                       00006420
000530     3430  FORMAT(11X, 5HOM  2, 3X, 3F11.3)                                    00006430
000530     3450  FORMAT(1H1//7X,33HRUNWAY THRESHOLD (TH) COORDINATES//               00006440
               1       11X, 5HTH  1, 3X, 3F11.3)                                     00006450
000530     3460  FORMAT(11X, 5HTH  2, 3X, 3F11.3)                                    00006460
000530     3480  FORMAT(/7X,36HMISSED APPROACH FIX (MA) COORDINATES //               00006470
               1       11X, 5HMA  1, 3X, 3F11.3)                                     00006480
000530     3490  FORMAT(11X, 5HMA  2, 3X, 3F11.3)                                    00006490
000530     3510  FORMAT(1HO,6X,22HTURN RADII IN N. MILES//18X,3HEF ,F8.2,7X,3HMF ,   00006500
               1       F8.2/18X, 3HIAF, F8.2, 7X, 3HFAF, F8.2 /18X,3HTF , F8.2,     00006510
               2       7X, 3HOM , F8.2 //7X, 27HPARALLEL FLIGHT PATH OFFSET          00006520
               3       20H DISTANCE (N. MILES) , F8.2)                               00006530
000530          END                                                                   00006540
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE RDSTAT                                                00007040
          C                                                                      00007050
000002          COMMON /STATIN/                                                00007060
                1    DCRIT,          DLYDEL,         DLYINT,         DMIN(5),  00007070
                2    DSDEL,          DSINT,          NDLY,           NDSEP,    00007080
                3    NOFSTP,         NTARR,          NTSEP,          OPDEL,    00007090
                4    OPINT,          TARDEL,         TARINT,         TMIN(5),  00007100
                5    TSDEL,          TSINT                                     00007110
          C                                                                      00007120
000002          READ 1110, NDLY, NDSEP, NOFSTP, NTARR, NTSEP                   00007130
000020          READ 1120, DCRIT, DLYDEL, DLYINT, DSDEL, DSINT, OPDEL, OPINT,  00007140
                1          TARDEL,TARINT, TSDEL,  TSINT                        00007150
000052          READ 1120, (DMIN(I), I=1,5)                                    00007160
000060          READ 1120, (TMIN(I), I=1,5)                                    00007170
          C                                                                      00007180
000066     1110 FORMAT(7I10)                                                   00007190
000066     1120 FORMAT(7F10.0)                                                 00007200
          C                                                                      00007210
000066          PRINT 3605, (TMIN(I), I=1,5)                                   00007220
000074          PRINT 3610, (DMIN(I), I=1,5)                                   00007230
000102          PRINT 3615, DCRIT                                              00007240
000110          PRINT 3620, OPINT, OPDEL, NOFSTP                               00007250
000122          PRINT 3625, TSINT, TSDEL, NTSEP                                00007260
000134          PRINT 3630, DSINT, DSDEL, NDSEP                                00007270
000146          PRINT 3635, TARINT, TARDEL, NTARR                              00007280
000160          PRINT 3640, DLYINT,DLYDEL,NDLY                                 00007290
          C                                                                      00007300
          C     CONVERT OF INPUT TIMES TO SECONDS FOR INTERNAL PROGRAM USE      00007310
          C                                                                      00007320
000172          TSINT=TSINT*60.                                                00007330
000174          TSDEL=TSDEL*60.                                                00007340
000175          TARINT=TARINT*60.                                              00007350
000176          TARDEL=TARDEL*60.                                              00007360
000177          DLYINT=DLYINT*60.                                              00007370
000200          DLYDEL=DLYDEL*60.                                              00007380
000201          DCRIT=DCRIT*60.                                                00007390
000202          RETURN                                                         00007400
          C                                                                      00007410
000202     3605 FORMAT(1H1// 37H    V. STATISTICS OUTPUT CONTROL DATA//        00007420
                1       6X, 32H1. SUMMARY STATISTICS PARAMETERS //             00007430
                2       14X, 27HMINIMUM TIME SEPARATION FOR /                  00007440
                3       14X, 23HDEPARTURE/DEPARTURE =  , F5.1,8H SECONDS/      00007450
                4       14X, 23HDEPARTURE/ ARRIVAL  =  , F5.1,8H SECONDS/      00007460
                5       14X, 23HARRIVAL  /DEPARTURE =  , F5.1,8H SECONDS/      00007470
                6       14X, 23HARRIVAL  / ARRIVAL  =  , F5.1,8H SECONDS/      00007480
                7       14X, 6HIN-AIR , 14X, 3H=  , F5.1, 8H SECONDS)          00007490
000202     3610 FORMAT(1H / 14X, 31HMINIMUM DISTANCE SEPARATION FOR  /           00007500
                1       14X, 23HDEPARTURE/DEPARTURE =  , F5.1, 9H N. MILES/    00007510
                2       14X, 23HDEPARTURE/ ARRIVAL  =  , F5.1, 9H N. MILES/    00007520
                3       14X, 23HARRIVAL  /DEPARTURE =  , F5.1, 9H N. MILES/    00007530
                4       14X, 23HARRIVAL  / ARRIVAL  =  , F5.1, 9H N. MILES/    00007540
                5       14X, 6HIN-AIR , 14X, 3H=  , F5.2, 9H N. MILES)         00007550
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000202    3615 FORMAT(1H / 14X, 35HCRITICAL DELAY TIME TO RUNWAY   =  ,          00007560
                1          F5.2, 8H MINUTES)                                   00007570
000202    3620 FORMAT(1H /1H / 6X, 40H2. STATISTICS FREQUENCY TABLE PARAMETERS/  00007580
                1          1H / 14X, 32HRUNWAY INTEROPERATION TIME TABLE//     00007590
                2          25X, 24HINITIAL TABLE VALUE  =  , F5.1, 8H SECONDS/ 00007600
                3          25X, 24HTABLE TIME INCREMENT =  , F5 1, 8H SECONDS/ 00007610
                4          25X, 24HNO. OF TABLE ENTRIES =  , I5)               00007620
000202    3625 FORMAT(1H / 14X, 32HIN-AIR TIME SEPARATION TABLE    //           00007630
                1          25X, 24HINITIAL TABLE VALUE  =  , F5.2, 8H MINUTES/ 00007640
                2          25X, 24HTABLE TIME INCREMENT =  , F5.2, 8H MINUTES/ 00007650
                3          25X, 24HNO. OF TABLE ENTRIES =  , I5)               00007660
000202    3630 FORMAT(1H / 14X, 32HIN-AIR DISTANCE SEPARATION TABLE//           00007670
                1          25X, 24HINITIAL TABLE VALUE  =  , F5.2, 8H N.MILES/ 00007680
                2          25X, 24HTABLE DIST.INCREMENT =  , F5.2, 8H N.MILES/ 00007690
                3          25X, 24HNO. OF TABLE ENTRIES =  , I5)               00007700
000202    3635 FORMAT(1H / 14X, 32HPATH-TIME DURATION TABLE        //           00007710
                1          25X, 24HINITIAL TABLE VALUE  =  , F5.2, 8H MINUTES/ 00007720
                2          25X, 24HTABLE TIME INCREMENT =  , F5.2, 8H MINUTES/ 00007730
                3          25X, 24HNO. OF TABLE ENTRIES =  , I5)               00007740
000202    3640 FORMAT(1H / 14X, 32HDELAY TIME TABLE                 //          00007750
                1          25X, 24HINITIAL TABLE VALUE  =  , F5.2, 8H MINUTES/ 00007760
                2          25X, 24HTABLE TIME INCREMENT =  , F5.2, 8H MINUTES/ 00007770
                3          25X, 24HNO. OF TABLE ENTRIES =  , I5)               00007780
000202          END                                                            00007790
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE REWTHR                                                    00004650
      C                                                                              00004660
000002          COMMON /TEMPER/                                                      00004670
               1    GTEMP1,         GTEMP2,         GTEMP3,         HTEMPO,        00004680
               2    HTEMP1,         HTEMP2,         TEMP,           TEMP1,         00004690
               3    TEMP2,          TGROND,         TEROR1,         TEROR2,        00004700
               4    TEROR3,         TSTAND                                         00004710
000002          COMMON /WIND/                                                        00004720
               1    GWINE1,         GWINE2,         GWINE3,         GWINN1,        00004730
               2    GWINN2,         GWINN3,         HWINDO,         HWIND1,        00004740
               3    HWIND2,         WERRE1,         WERRE2,         WERRE3,        00004750
               4    WERRN1,         WERRN2,         WERRN3,         WINANG,        00004760
               5    WINDE,          WINDED,         WINDE1,         WINDE2,        00004770
               6    WINDN,          WINDND,         WINDN1,         WINDN2,        00004780
               7    WINMAG                                                         00004790
      C                                                                              00004800
000002          READ  1120, HTEMPO, HTEMP1, HTEMP2, GTEMP1, GTEMP2, GTEMP3,        00004810
               1                TEROR1, TEROR2, TEROR3, TGROND                     00004820
000032    1120 FORMAT(7F10.0)                                                        00004830
      C                                                                              00004840
000032          TEMZN1=HTEMP1-HTEMPO                                                 00004850
000034          TEMZN2=HTEMP2-HTEMP1                                                 00004860
000035          TEMP1=GTEMP1+TEMZN1+TGROND                                           00004870
000040          TEMP2=GTEMP2+TEMZN2+TEMP1                                            00004880
      C                                                                              00004890
000043          PRINT  3240, HTEMPO, HTEMP1, HTEMP2, GTEMP1                          00004900
000057          PRINT  3245, GTEMP2, GTEMP3, TEROR1, TEROR2, TEROR3, TGROND          00004910
000077    3240 FORMAT(1H1// 31H   II. WEATHER MODEL INPUT DATA //                    00004920
               1       7X, 50HBASE OF BOTTOM TEMPERATURE ZONE (GROUND LEVEL)   =,  00004930
               2            F8.2, 14H THOUSAND FEET/                              00004940
               3       7X, 50HCEILING OF BOTTOM TEMPERATURE ZONE               =,  00004950
               4            F8.2, 14H THOUSAND FEET/                              00004960
               5       7X, 50HCEILING OF MIDDLE TEMPERATURE ZONE               =,  00004970
               6            F8.2, 14H THOUSAND FEET/                              00004980
               7       7X, 50HTEMPERATURE GRADIENT IN BOTTOM ZONE               =,  00004990
               8            F8.2, 13H DEG-K/1000FT)                               00005000
000077    3245 FORMAT(7X, 50HTEMPERATURE GRADIENT IN MIDDLE ZONE                  =,  00005010
               A            F8.2, 13H DEG-K/1000FT/                               00005020
               B       7X, 50HTEMPERATURE GRADIENT IN UPPER ZONE                =,  00005030
               C            F8.2, 13H DEG-K/1000FT/                               00005040
               D       7X, 50HTEMPERATURE FORECAST ERROR IN BOTTOM ZONE         =,  00005050
               E            F8.2,  6H DEG-K/                                      00005060
               F       7X, 50HTEMPERATURE FORECAST ERROR IN MIDDLE ZONE          =,  00005070
               G            F8.2,  6H DEG-K/                                      00005080
               H       7X, 50HTEMPERATURE FORECAST ERROR IN UPPER  ZONE          =,  00005090
               I            F8.2,  6H DEG-K/                                      00005100
               J       7X, 50HFORECAST TEMPERATURE AT GROUND LEVEL              =,  00005110
               K            F8.2,  6H DEG-K)                                      00005120
      C                                                                              00005130
000077          READ 1120, HWINDO, HWIND1, HWIND2, GWINN1, GWINN2, GWINN3,         00005140
               1            GWINE1, GWINE2, GWINE3, WINDND, WINDED, WERRN1,       00005150
               2            WERRN2, WERRN3, WERRE1, WERRE2, WERRE3                00005160
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000145          WINZN1=HWIND1-HWIND0                                            00005170
000147          WINZN2=HWIND2-HWIND1                                            00005180
000150          WINDN1=GWINN1+WINZN1+WINDN0                                     00005190
000153          WINDN2=GWINN2+WINZN2+WINDN1                                     00005200
000156          WINDE1=GWINE1+WINZN1+WINDE0                                     00005210
000160          WINDE2=GWINE2+WINZN2+WINDE1                                     00005220
              C                                                                 00005230
000163          PRINT  3250, HWIND0, HWIND1, HWIND2, GWINN1, GWINN2, GWINN3,    00005240
               1                GWINE1, GWINE2, GWINE3                          00005250
000211          PRINT  3255, WINDN0, WINDE0, WERRN1, WERRN2, WERRN3, WERRE1,    00005260
               1                WERRE2, WERRE3                                  00005270
000235          RETURN                                                          00005280
              C                                                                 00005290
000236    3250 FORMAT(1HO, 6X, 39HBASE OF BOTTOM WIND ZONE (GROUND LEVEL), 10X,  00005300
               1       1H=,       F8.2, 14H THOUSAND FEET/                     00005310
               2       7X, 50HCEILING OF BOTTOM WIND ZONE                  =,  00005320
               3                  F8.2, 14H THOUSAND FEET/                     00005330
               4       7X, 50HCEILING OF MIDDLE WIND ZONE                  =,  00005340
               5                  F8.2, 14H THOUSAND FEET/                     00005350
               6       7X, 50HNORTH COMP. OF WIND GRADIENT IN BOTTOM ZONE  =,  00005360
               7                  F8.2, 6H KNOTS/                              00005370
               8       7X, 50HNORTH COMP. OF WIND GRADIENT IN MIDDLE ZONE  =,  00005380
               9                  F8.2, 6H KNOTS/                              00005390
               A       7X, 50HNORTH COMP. OF WIND GRADIENT IN UPPER  ZONE  =,  00005400
               B                  F8.2, 6H KNOTS/                              00005410
               C       7X, 50HEAST  COMP. OF WIND GRADIENT IN BOTTOM ZONE  =,  00005420
               D                  F8.2, 6H KNOTS/                              00005430
               E       7X, 50HEAST  COMP. OF WIND GRADIENT IN MIDDLE ZONE  =,  00005440
               F                  F8.2, 6H KNOTS/                              00005450
               G       7X, 50HEAST  COMP. OF WIND GRADIENT IN UPPER  ZONE  =,  00005460
               H                  F8.2, 6H KNOTS)                              00005470
000236    3255 FORMAT(7X, 50HNORTH COMP. OF FORECAST WIND AT GROUND LEVEL  =,  00005480
               J                  F8.2, 6H KNOTS/                              00005490
               K       7X, 50HEAST  COMP. OF FORECAST WIND AT GROUND LEVEL  =,  00005500
               L                  F8.2, 6H KNOTS/                              00005510
               M       7X, 50HFORECAST ERROR ON NORTH WIND COMP IN BOTTOM ZONE =, 00005520
               N                  F8.2, 6H KNOTS/                              00005530
               O       7X, 50HFORECAST ERROR ON NORTH WIND COMP IN MIDDLE ZONE =, 00005540
               P                  F8.2, 6H KNOTS/                              00005550
               Q       7X, 50HFORECAST ERROR ON NORTH WIND COMP IN UPPER  ZONE =, 00005560
               R                  F8.2, 6H KNOTS/                              00005570
               S       7X, 50HFORECAST ERROR ON EAST  WIND COMP IN BOTTOM ZONE =, 00005580
               T                  F8.2, 6H KNOTS/                              00005590
               U       7X, 50HFORECAST ERROR ON EAST  WIND COMP IN MIDDLE ZONE =, 00005600
               V                  F8.2, 6H KNOTS/                              00005610
               W       7X, 50HFORECAST ERROR ON EAST  WIND COMP IN UPPER  ZONE =, 00005620
               X                  F8.2, 6H KNOTS)                              00005630
000236          END                                                             00005640
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE READIN                                                    00002330
      C                                                                              00002340
000002          COMMON /TRAFIN/                                                      00002350
               1    ALTPA1(10),    ALTPA2(10),    CALT (10,10),  CVREF(10,10),     00002360
               2    CWALT(10,10),  CWVREF(10,10), ENTRY(10,4,3), HOUR,             00002370
               3    INDALT(10),    INDSPD(10),    INDTA,         INDTD,           00002380
               4    INDWA(10),     INDWD(10),     NCALT(10),     NCVREF(10),      00002390
               5    NMTYPA(10),    NMTYPD(10),    NTYPEA,        NTYPED,          00002400
               6    SPDPA1(10),    SPDPA2(10),    SPSIGM,        SWTRAF,          00002410
               7    TAPAR1,        TAPAR2,        TASIGM,        TDPAR1,          00002420
               8    TDPAR2,        TDSIGM,        TYPEA(10),     TYPED(10),       00002430
               9    WAPAR1(10),    WAPAR2(10),    WDPAR1(10),    WDPAR2(10)       00002440
000002          INTEGER            SWTRAF                                          00002450
      C                                                                          00002460
000002          READ  1130, SWTRAF                                               00002470
000010          PRINT  3000                                                      00002480
      C                                                                          00002490
000014    1130 FORMAT(A6)                                                        00002500
000014    3000 FORMAT (1H1//                                                     00002510
               1        10X, 49H***** EVALUATION MODEL INPUT VARIABLES *****)  00002520
      C                                                                          00002530
      C     READ DISTRIBUTION INPUT DATA FOR TRAFFIC GENERATOR                   00002540
      C                                                                          00002550
000014          IF (SWTRAF.EQ.6HDISTRB) CALL RDDSTR                              00002560
      C                                                                          00002570
      C     READ DISCRETE INPUT DATA FOR TRAFFIC GENERATOR SUBROUTINE TRAFIC     00002580
      C                                                                          00002590
000017          IF (SWTRAF.NE.6HDISTRD) CALL RDDSCR                              00002600
      C                                                                          00002610
      C     READ WEATHER DATA                                                    00002620
      C                                                                          00002630
000022          CALL RDWTHR                                                      00002640
      C                                                                          00002650
      C     READ FLIGHT PATH/GEOMETRY INPUT DATA                                 00002660
      C                                                                          00002670
000023          CALL RDGEOM                                                      00002680
      C                                                                          00002690
      C     READ AERO-PERFORMANCE INPUT DATA FOR ARRIVAL A/C TYPES               00002700
      C                                                                          00002710
000024          CALL RDAERO                                                      00002720
      C                                                                          00002730
      C     READ INPUT DATA FOR EVALUATION STATISTICS TABULATION                 00002740
      C                                                                          00002750
000025          CALL RDSTAT                                                      00002760
      C                                                                          00002770
      C     READ ATC SAFETY SEPARATION PARAMETERS                                00002780
      C                                                                          00002790
000026          CALL RDATC                                                       00002800
      C                                                                          00002810
      C     READ AVIONICS PERFORMANCE PARAMETERS                                 00002820
      C                                                                          00002830
000027          CALL RDAVNC                                                      00002840
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000030          RETURN                                  00002850
000031          END                                     00002860
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE REFBUG                                                00019720
          C                                                                      00019730
          C     THIS ROUTINE TAKES THE FLIGHT PATH ANGLE CORRESPONDING           00019740
          C     TO THE FINAL APPROACH COURSE, FINDS THE ALONG TRACK              00019750
          C     WIND COMPONENT, FINDS VBUG, AND FINALLY DETERMINES               00019760
          C     THE GROUND SPEED EQUIVALENT OF THE BUG AIRSPEED,                 00019770
          C     WHICH IS PROPERLY CORRECTED FOR THE TEMPERATURE FORCAST.          00019780
          C                                                                      00019790
000002            COMMON /ATMCV                                                00019800
                 1    DENSR,          DENSRF,         DENSTY,         PRESR,     00019810
                 2    PRESUR,         TEMR,           TEMRS                      00019820
000002            COMMON /GECMIN/                                              00019830
                 1    DPARAL,         EFX(4,3),       EFY(4,3),       EFZ(4,3),  00019840
                 2    FAFX(2),        FAFY(2),        FAFZ(2),        IAFX(4),   00019850
                 3    IAFY(4),        IAFZ(4),        MAX(2),         MAY(2),    00019860
                 4    MAZ(2),         MFX(2),         MFY(2),         MFZ(2),    00019870
                 5    NEF,            NPATH,          NQUAD,          NRNWY,     00019880
                 6    CMX(2),         CMY(2),         CMZ(2),         RHOEF,     00019890
                 7    RHOFAF,         RHOIAF,         RHOMF,          RHOOM,     00019900
                 8    RHOTF,          TFX(4),         TFY(4),         TFZ(4),    00019910
                 9    THX(2),         THY(2),         THZ(2)                     00019920
000002            REAL                IAFX,           IAFY,           IAFZ,      00019930
                 1    MAX,            MAY,            MAZ,            MFX,       00019940
                 2    MFY,            MFZ                                        00019950
000002            COMMON /GECMCT/                                              00019960
                 1    ANGEF(4,3),     ANGFAF(2),      ANGIAF(4),      ANGMF(2),  00019970
                 2    ANGOM(2),       ANGTF(4),       DEF(4,3,5),     DFAF(2),   00019980
                 3    DIAF(4),        DMF(2),         DOM(2),         DPATH(4,3,5), 00019990
                 4    DTF(4),         DTH(2)                                     00020000
000002            COMMON /TRAFCT/                                              00020010
                 1    ALT(200),       EELT(200),      ELLT(200),      GWA(200),  00020020
                 2    GWD(100),       ITYPEA(200),    ITYPED(100),    KEY(200),  00020030
                 3    LFIX(200),      NACA,           NACD,           SDT(100),  00020040
                 4    SLT(200),       SPEED(200),     TSECAA(200),    TSECAD(100), 00020050
                 5    TSECSA(200),    TSECSD(100),    VBUGG(200),     VREF(200)  00020060
000002            COMMON /WIND/                                                00020070
                 1    GWINE1,         GWINE2,         GWINE3,         GWINN1,    00020080
                 2    GWINN2,         GWINN3,         HWIND0,         HWIND1,    00020090
                 3    HWIND2,         WERRE1,         WERRE2,         WERRE3,    00020100
                 4    WERRN1,         WERRN2,         WERRN3,         WINANG,    00020110
                 5    WINDE,          WINDE0,         WINDE1,         WINDE2,    00020120
                 6    WINDN,          WINDN0,         WINDN1,         WINDN2,    00020130
                 7    WINMAG                                                     00020140
          C                                                                      00020150
          C     GET DENSITY RATIO FACTOR AT RUNWAY ALTITUDE                      00020160
          C     GET ALONG TRACK WIND AT RUNWAY ALTITUDE                          00020170
          C                                                                      00020180
000002            CALL WETHER(CMZ(1),0)                                        00020190
000004            VATW = WINMAG*COS(ANGOM(1)-WINANG)                           00020200
          C                                                                      00020210
          C     GET APPROACH AIRSPEED                                          00020220
          C                                                                      00020230
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000012          IF (VATW) 10, 30, 50                                    00020240
000013    10    CONTINUE                                                00020250
000013             DO 20 I = 1, NACA                                    00020260
000015             VDUGG(I) = (VREF(I) - 0.5*VATW)*DENSRF + VATW        00020270
000022    20       CONTINUE                                            00020280
000024             RETURN                                               00020290
000024    30    CONTINUE                                                00020300
000024             DO 40 I = 1, NACA                                    00020310
000026             VDUGG(I) = VREF(I)*DENSRF                            00020320
000030    40       CONTINUE                                            00020330
000032             RETURN                                               00020340
000033    50    CONTINUE                                                00020350
000033             DO 60 I = 1, NACA                                    00020360
000035             VDUGG(I) = VREF(I)*DENSRF + VATW                     00020370
000040    60       CONTINUE                                            00020380
000042             RETURN                                               00020390
000043          END                                                     00020400
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE RTP (IFLITX, NSAMP)                                  00027350
          C                                                                     00027360
          C     THIS SUBROUTINE DETERMINES THE CONFLICT FREE PATH,            00027370
          C     VELOCITIES, AND TIMES TO BE MADE GOOD BY EACH                 00027380
          C     AIRPLANE IN THE SEQUENCE LIST                                 00027390
          C                                                                     00027400
000005            COMMON /AERCOT/                                             00027410
                 1    HB(10),          HCRIT,          JMAX,          SFKNOT,  00027420
                 2    VFAST(10),       VSLOW(10)                               00027430
000005            COMMON /AERC1C/                                             00027440
                 1    HC(4,5),         TC(4,5),        TCSUM(4),      THPRES,  00027450
                 2    VCCAS(5),        VCG(4,5),       VCTAS(4,5)              00027460
000005            COMMON /CNSTNT/                                             00027470
                 1    ACCEL,           GRADI,          ITMAX,         PI,      00027480
                 2    RTPTST                                                  00027490
000005            COMMON /GECMIN/                                             00027500
                 1    DPARAL,          EFX(4,3),       EFY(4,3),      EFZ(4,3),00027510
                 2    FAFX(2),         FAFY(2),        FAFZ(2),       IAFX(4), 00027520
                 3    IAFY(4),         IAFZ(4),        MAX(2),        MAY(2),  00027530
                 4    MAZ(2),          MFX(2),         MFY(2),        MFZ(2),  00027540
                 5    NEF,             NPATH,          NQUAD,         NRNWY,   00027550
                 6    CMX(2),          CMY(2),         CMZ(2),        RHOEF,   00027560
                 7    RHOFAF,          RHOIAF,         RHOMF,         RHOOM,   00027570
                 8    RHOTF,           TFX(4),         TFY(4),        TFZ(4),  00027580
                 9    THX(2),          THY(2),         THZ(2)                  00027590
000005            REAL                 IAFX,           IAFY,          IAFZ,    00027600
                 1    MAX,             MAY,            MAZ,           MFX,     00027610
                 2    MFY,             MFZ                                     00027620
000005            COMMON /GECMOT/                                             00027630
                 1    ANGEF(4,3),      ANGFAF(2),      ANGIAF(4),     ANGMF(2),00027640
                 2    ANGOM(2),        ANGTF(4),       DEF(4,3,5),    DFAF(2), 00027650
                 3    DIAF(4),         DMF(2),         DOM(2),        DPATH(4,3,5),00027660
                 4    DTF(4),          DTH(2)                                  00027670
000005            COMMON /RTPCV/                                              00027680
                 1    DDP,             DLF,            HIAF,          I,       00027690
                 2    INFIX,           IQUAD,          IS,            TIAF     00027700
000005            COMMON /RTPOUT/                                             00027710
                 1    DRTP(14,200),    HRTP(14,200),   IDRTP(14,200), TRTP(14,200),00027720
                 2    VRTP(14,200)                                            00027730
000005            COMMON /TRAFOT/                                             00027740
                 1    ALT(200),        EELT(200),      ELLT(200),     GWA(200),00027750
                 2    GWB(100),        ITYPEA(200),    ITYPEB(100),   KEY(200),00027760
                 3    LFIX(200),       NACA,           NACB,          SDT(100),00027770
                 4    SLT(200),        SPEED(200),     TSECAA(200),   TSECAB(100),00027780
                 5    TSECSA(200),     TSECSB(100),    VDUGG(200),    VREF(200)00027790
000005            DIMENSION            LACF(4,3,5)                             00027800
          C                                                                     00027810
000005            DO 10 I = 1, 4                                              00027820
000006               DO 10 J = 1, 3                                           00027830
000007                  DO 10 K = 1, 5                                        00027840
000010                     LACF(I,J,K) = 0.                                   00027850
000015    10    CONTINUE                                                      00027860
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A         16.56.27. 73/12/21.

                C                                                              00027870
                C     INITIALIZE TO FIRST AIRPLANE, AND GET SEQUENCE POINTER    00027880
                C                                                              00027890
000023          DO 100 I = 1, NACA                                             00027900
                C                                                              00027920
                C        DETERMINE EF NUMBER AND QUADRANT.  STORED IN LFIX AS A 00027930
                C        TWO DIGIT NO., THE FIRST DIGIT IS THE QUAD (1,2,3, OR 4) 00027940
                C        AND THE SECOND DIGIT IS THE EF NO. (ILFIX = 1,2, OR 3) 00027950
                C                                                              00027960
000024                   IS = KEY(I)                                           00027910
000026                   ILFIX = LFIX(IS)                                      00027970
000030                   IQUAD = ILFIX/10                                      00027980
000033                   INFIX = ILFIX - IQUAD*10                              00027990
                C                                                              00028000
                C        COMPUTE THE IAF TO TH TIME REQUIRED                   00028010
                C                                                              00028020
000035                   VAVGG = 0.5*(VCG(IQUAD,1) + VDUGG(IS))                00028030
000040                   TCMTH = 3600.*DCM(1)/VAVGG                            00028040
000043                   TIAFTH = TCSUM(IQUAD) + TCMTH                         00028050
                C                                                              00028060
                C        DETERMINE EF TIME FOR THIS AIRPLANE                   00028070
                C        COMPUTE TIME AT IAF                                   00028080
                C        COMPUTE TRANSITION TIME FROM EF TO IAF                00028090
                C        INITIALIZE PATH COUNTER                               00028100
                C                                                              00028110
000045                   TEF = TSECAA(IS)                                      00028120
000047                   TIAF = SLT(IS) - TIAFTH                               00028130
000051                   TEFIAF = TIAF - TEF                                   00028140
000053                   IPATH = 1                                             00028150
                C                                                              00028160
                C        TEST FOR A GROSS CONFLICT WITH PRIOR AIRCRAFT ON PATH 00028170
                C                                                              00028180
000054          60       CONTINUE                                              00028190
000054                   CALL CNFLCTG(LACP(IQUAD,INFIX,IPATH),IS,TEF,TIAF,IFLAG) 00028200
000064                   IF (IFLAG.EQ.0) GO TO 50                              00028210
                C                                                              00028220
                C           CONFLICT EXISTS, GO TO THE NEXT PATH               00028230
                C                                                              00028240
000067                      IPATH = IPATH + 1                                  00028250
000070                      IF (IPATH.LE.NPATH) GO TO 60                       00028260
000072                         IPATH = NPATH                                   00028270
000073                         PRINT 8000, I, IS                               00028280
000102          8000           FORMAT (33HO----NO MORE PATHS AT CNFLCTG---- /  00028290
               1                5X,4HI =  ,I4,7H  IS =  ,I4)                   00028300
000102          50       CONTINUE                                              00028310
                C                                                              00028320
                C        NO CONFLICT, COMPUTE ALT. AT IAF                      00028330
                C        HIAF IS A *CUTE* CALCULATION--E.G., IAF ALT FOR FIX 1 IS 00028340
                C        10 K FEET.                                            00028350
                C                                                              00028360
000102                   HIAF = 9. + INFIX                                     00028370
                C                                                              00028380
                C        COMPUTE DESCENT PATH LENGTH                           00028390
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

         C        GET AERO PERFORMANCE TABLE                                    00028400
         C                                                                      00028410
000105             DGS = (ALT(IS) - HIAF)/GRADI                                00028420
000110             DLF = DEF(IQUAD,INFIX,IPATH) - DGS - DDF                    00028430
000117             CALL AERO (DUMY,IS,IQUAD,INFIX,0)                           00028440
         C                                                                      00028450
         C        DETERMINE RTP BY ITERATION                                    00028460
         C        SET ITERATION COUNTER TO ZERO                                 00028470
         C                                                                      00028480
000122             ITCNTR=0                                                    00028490
         C                                                                      00028500
         C        TEST IF ALT IS .LE. TO THE CRITICAL ALTITUDE                  00028510
         C        IF SO USE STRATEGY 3                                          00028520
         C                                                                      00028530
000123             IF (ALT(IS).LE.HCRIT) GO TO 93                              00028540
         C                                                                      00028550
         C        MAKE A CONSTANT MACH/CONSTANT CAS LET DOWN AT ENTRY MACH      00028560
         C                                                                      00028570
000130             VNEW = SPKNOT                                               00028580
000131             CALL STUFF2(VNEW)                                           00028590
000133             CALL TFIND(IS,SUMT)                                         00028600
         C                                                                      00028610
         C        TEST RTP CALCULATED                                           00028620
         C                                                                      00028630
000135             IF (SUMT-TEFIAF) 195,95,295                                 00028640
         C                                                                      00028650
         C        AIRPLANE GOING TOO FAST TO MEET TIME AT IAF                   00028660
         C                                                                      00028670
000141     195     CONTINUE                                                    00028680
000141             V2=VNEW                                                     00028690
000143             T2=SUMT                                                     00028700
         C                                                                      00028710
         C        SLOW TO LOW SPEED BOUNDARY AND MAKE MACH/CAS LETDOWN          00028720
         C                                                                      00028730
000144             VNEW=VSLOW(JMAX)                                            00028740
000146             CALL STUFF2(VNEW)                                           00028750
000147             CALL TFIND(IS,SUMT)                                         00028760
         C                                                                      00028770
         C        TEST LETDOWN FOR TIME NEEDED                                  00028780
         C                                                                      00028790
000151             IF (SUMT-TEFIAF) 197,95,196                                 00028800
         C                                                                      00028810
         C        GOING TOO SLOW -- USE STRATEGY 2                              00028820
         C                                                                      00028830
000155     196     CONTINUE                                                    00028840
000155             V1=VNEW                                                     00028850
000157             T1=SUMT                                                     00028860
000160             GO TO 96                                                    00028870
         C                                                                      00028880
         C        GOING TOO FAST, MAKE A CAS LETDOWN                            00028890
         C                                                                      00028900
000161     197     CONTINUE                                                    00028910
000161             H2=HCRIT                                                    00028920
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000163            T2=SUMT                                                    00028930
000164            HNEW=ALT(IS)                                               00028940
000166            CALL STUFF1 (HNEW, VSLOW(JMAX))                            00028950
000171            CALL TFIND (IS, SUMT)                                      00028960
000173            IF (SUMT-TEFIAF) 198,95,199                                00028970
000177     198    CONTINUE                                                   00028980
000177            IF(ABS(SUMT-TEFIAF).LE. RTPTST)GO TO 95                    00028990
          C                                                                  00029000
          C       GOING TOO FAST AT LOW SPEED BOUNDARY -- NO RTP FOUND       00029010
          C                                                                  00029020
000204            PRINT 9000, I, IS                                          00029030
000213    9000    FORMAT (34HO---NO RTP FOUND (ERROR NO. 2)---- /5X,4HI =  ,  00029040
                1   I5,6H IS =  , I5)                                        00029050
000213            IFLITX = NSAMP                                             00029060
000215            GO TO 95                                                   00029070
          C                                                                  00029080
          C       GOING TOO SLOW -- USE STRATEGY 1                           00029090
          C                                                                  00029100
000216     199    CONTINUE                                                   00029110
000216            H1=ALT(IS)                                                 00029120
000220            T1=SUMT                                                    00029130
000222            GO TO 97                                                   00029140
          C                                                                  00029150
          C       AIRPLANE GOING TOO FAST ON RTP GENERATED FOR CONSTANT MACH/CAS 00029160
          C       LETDOWN                                                    00029170
          C                                                                  00029180
000222     295    CONTINUE                                                   00029190
000222            V1=VNEW                                                    00029200
000224            T2=SUMT                                                    00029210
          C                                                                  00029220
          C       SPEED TOO HIGH SPEED BOUNDARY -- MAKE A MACH/CAS LETDOWN   00029230
          C                                                                  00029240
000225            VNEW = VFAST(JMAX)                                         00029250
000227            CALL STUFF2(VNEW)                                          00029260
000230            CALL TFIND(IS,SUMT)                                        00029270
000232            IF (SUMT-TEFIAF) 297,95,296                                00029280
          C                                                                  00029290
          C       GOING TOO SLOW AT HIGH SPEED BOUNDARY                      00029300
          C                                                                  00029310
000236     296    CONTINUE                                                   00029320
000236            PRINT 9001, I, IS                                          00029330
000246    9001    FORMAT (35HO----NO RTP FOUND (ERROR NO. 1)---- /5X,4HI =  ,  00029340
                1   I5,7H  IS =  ,I5)                                        00029350
000246            IFLITX = NSAMP                                             00029360
000250            GO TO 95                                                   00029370
          C                                                                  00029380
          C       GOING TOO FAST AT HIGH SPEED BOUNDARY -- SLOW DOWN         00029390
          C                                                                  00029400
000251     297    CONTINUE                                                   00029410
000251            V2=VNEW                                                    00029420
000253            T2=SUMT                                                    00029430
          C                                                                  00029440
          C       USE STRATEGY 2                                             00029450
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

      C                                                                  00029460
000254                GO TO 96                                         00029470
      C                                                                  00029480
      C                                                                  00029490
      C       STRATEGY 1                                               00029500
      C                                                                  00029510
      C                                                                  00029520
000255    97  CONTINUE                                                 00029530
      C                                                                  00029540
      C       GET ALT. BREAK POINT FOR TRIAL AND MAKE A LETDOWN        00029550
      C                                                                  00029560
000255                HNEW=0.5*(H1+H2)                                 00029570
000260                CALL STUFF1(HNEW,VSLOW(JMAX))                    00029580
000263                CALL TFIND(IS,SUMT)                              00029590
      C                                                                  00029600
      C       WAS THE LETDOWN WITHIN TOLERANCE                         00029610
      C                                                                  00029620
000265                IF (ABS(SUMT-TEFIAF).LT.RTPIST) GO TO 95         00029630
      C                                                                  00029640
      C       NO                                                       00029650
      C                                                                  00029660
000273                IF (SUMT.GT.TEFIAF) GO TO 971                    00029670
      C                                                                  00029680
      C       GOING TOO FAST                                           00029690
      C                                                                  00029700
000276                H2=HNEW                                          00029710
000277                T2=SUMT                                          00029720
000300                GO TO 972                                        00029730
      C                                                                  00029740
      C       GOING TOO SLOW                                           00029750
      C                                                                  00029760
000300    971 CONTINUE                                                 00029770
000300                H1=HNEW                                          00029780
000302                T1=SUMT                                          00029790
000303    972 CONTINUE                                                 00029800
000303                ITCNTR=ITCNTR+1                                  00029810
      C                                                                  00029820
      C       HAVE TOO MANY ITERATIONS OCCURRED                        00029830
      C                                                                  00029840
000305                IF (ITCNTR.LE.ITMAX) GO TO 97                    00029850
000307                PRINT 9002, I, IS                                00029860
000317   9002 FORMAT (40HO----TOO MANY ITERATIONS, STRATEGY 1---- /,5X, 00029870
          1    4HI =  , I5, 6H IS =  , I5)                            00029880
000317                IFLITX = NSAME                                   00029890
000321                GO TO 95                                         00029900
      C                                                                  00029910
      C                                                                  00029920
      C       STRATEGY 2                                               00029930
      C                                                                  00029940
      C                                                                  00029950
000322    96  CONTINUE                                                 00029960
000322                VNEW=0.5*(V1+V2)                                 00029970
      C                                                                  00029980
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                C         CALCULATE DATA FOR A LETDOWN                                    00029990
                C                                                                         00030000
000325                    CALL STUFF2(VNEW)                                               00030010
000327                    CALL TFIND(IS,SUMT)                                             00030020
                C                                                                         00030030
                C         WAS LETDOWN WITHIN TOLERANCE                                    00030040
                C                                                                         00030050
000331                    IF (ABS(SUMT-TEFIAF).LT.RTPTST) GO TO 95                        00030060
                C                                                                         00030070
                C         NO                                                              00030080
                C                                                                         00030090
000337                    IF (SUMT.GT.TEFIAF) GO TO 961                                   00030100
                C                                                                         00030110
                C         GOING TOO FAST                                                  00030120
                C                                                                         00030130
000342                    V2=VNEW                                                         00030140
000343                    T2=SUMT                                                         00030150
000344                    GO TO 962                                                       00030160
                C                                                                         00030170
                C         GOING TOO SLOW                                                  00030180
                C                                                                         00030190
000344      961   CONTINUE                                                              00030200
000344            V1=VNEW                                                               00030210
000346            T1=SUMT                                                               00030220
000347      962   CONTINUE                                                              00030230
000347            ITCNTR=ITCNTR+1                                                       00030240
                C                                                                         00030250
                C         TOO MANY ITERATIONS                                             00030260
                C                                                                         00030270
000351            IF (ITCNTR.LE.ITMAX) GO TO 96                                         00030280
000353            PRINT 9003, I, IS                                                     00030290
000363     9003   FORMAT (41HO----TOO MANY ITERATIONS - STRATEGY 2---- /,5X,             00030300
                1    4HI =  ,I5,6H IS =  ,I5)                                         00030310
000363            IFLITX = NSAMP                                                        00030320
000365            GO TO 95                                                              00030330
                C                                                                         00030340
                C                                                                         00030350
                C         STRATEGY 3                                                      00030360
                C                                                                         00030370
                C                                                                         00030380
                C         MAKE A CAS LETDOWN AT THE  LOW SPEED BOUNDARY                   00030390
                C                                                                         00030400
000366       93   CONTINUE                                                              00030410
000366            VNEW=VSLOW(JMAX)                                                      00030420
000370            CALL STUFF3(VNEW)                                                     00030430
000372            CALL TFIND(IS,SUMT)                                                   00030440
000374            IF (SUMT-TEFIAF) 931,95,932                                           00030450
                C                                                                         00030460
                C         GOING TOO FAST                                                  00030470
                C                                                                         00030480
000400       931  CONTINUE                                                              00030490
000400            PRINT 9004, I, IS                                                     00030500
000410     9004   FORMAT (35HO----NO RTP FOUND (ERROR NO. 4)---- /5X, 4HI =  ,          00030510
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION 66T 73 A          16.56.27. 73/12/21.

     1    15,7H  IS =  ,I5)                                          00030520
000410            IFLITX = NSAMP                                     00030530
000412            GO TO 95                                           00030540
000413      932   CONTINUE                                           00030550
            C                                                        00030560
            C     GOING TOO SLOW                                     00030570
            C                                                        00030580
000413            V1=VNEW                                            00030590
000415            T1=SUMT                                            00030600
            C                                                        00030610
            C     MAKE A CAS LETDOWN AT THE HIGH SPEED BOUNDARY      00030620
            C                                                        00030630
000416            VNEW=VFAST(JMAX)                                   00030640
000420            CALL STUFF3(VNEW)                                  00030650
000421            CALL TFIND(IS,SUMT)                                00030660
            C                                                        00030670
            C     LET DOWN WITHIN TOLERANCE                          00030680
            C                                                        00030690
000423            IF (ABS(SUMT-TEFIAF).LT.RTPTST) GO TO 95           00030700
            C                                                        00030710
            C     NO                                                 00030720
            C                                                        00030730
000431            IF (SUMT.LE.TEFIAF) GO TO 933                      00030740
            C                                                        00030750
            C     GOING TOO SLOW                                     00030760
            C                                                        00030770
000433            PRINT 9005, I, IS                                  00030780
000442     9005   FORMAT(35HO----NO RTP FOUND (ERROR NO. 3)---- /5X, 4HI = ,  00030790
         1    I5,7H  IS =  ,I5)                                     00030800
000442            IFLITX = NSAMP                                     00030810
000444            GO TO 95                                           00030820
            C                                                        00030830
            C     GOING TOO FAST - COMPUTE A NEW LETDOWN             00030840
            C                                                        00030850
000445      933   CONTINUE                                           00030860
000445            V2=VNEW                                            00030870
000447            T2=SUMT                                            00030880
000450       92   CONTINUE                                           00030890
000450            VNEW=0.5*(V1+V2)                                   00030900
000453            CALL STUFF3(VNEW)                                  00030910
000455            CALL TFIND(IS, SUMT)                               00030920
            C                                                        00030930
            C     LET DOWN WITHIN TOLERANCE                          00030940
            C                                                        00030950
000457            IF (ABS(SUMT-TEFIAF).LT.RTPTST) GO TO 95           00030960
            C                                                        00030970
            C     NO                                                 00030980
            C                                                        00030990
000465            IF (SUMT.GT.TEFIAF) GO TO 921                      00031000
            C                                                        00031010
            C     TOO FAST                                           00031020
            C                                                        00031030
000470            V2=VNEW                                            00031040
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000471                T2=SUMT                                                  00031050
000472                GO TO 922                                                00031060
            C                                                                  00031070
            C         TOO SLOW                                                 00031080
            C                                                                  00031090
000472     921    CONTINUE                                                     00031100
000472            V1=VNEW                                                      00031110
000474            T1=SUMT                                                      00031120
000475     922    CONTINUE                                                     00031130
000475            ITCNTR=ITCNTR+1                                              00031140
            C                                                                  00031150
            C         HAVE TOO MANY ITERATIONS OCCURRED                        00031160
            C                                                                  00031170
000477            IF (ITCNTR.LE.ITMAX) GO TO 92                                00031180
000501            PRINT 9006, I, IS                                            00031190
000511    9006    FORMAT(41HO----TOO MANY ITERATIONS - STRATEGY 3---- /,5X,    00031200
                1    4HI =  ,I5,7H  IS =  ,I5)                               00031210
000511            IFLITX = NSAMP                                               00031220
            C                                                                  00031230
            C                                                                  00031240
            C         COLLECT RTP                                              00031250
            C                                                                  00031260
            C                                                                  00031270
000513      95    CONTINUE                                                     00031280
000513            DO 500 JRTP=1,14                                             00031290
000515               IF (IDRTP(JRTP,IS).EQ.2) GO TO 51                         00031300
000521     500    CONTINUE                                                     00031310
000523      52    CONTINUE                                                     00031320
000523            PRINT 9007                                                   00031330
000527    9007    FORMAT (41HO----RTP ARRAYS DIMENSIONED TOO SMALL---- )       00031340
000527            STOP 5                                                       00031350
000531      51    CONTINUE                                                     00031360
000531            JRTP=JRTP+1                                                  00031370
000533            IF (JRTP.EQ.15) GO TO 52                                     00031380
000536            DRTP(JRTP,IS)=DRTP(JRTP-1,IS)+DIAF(IQUAD)                    00031390
000544            TRTP(JRTP,IS)=TRTP(JRTP-1,IS)+TC(IQUAD,5)                    00031400
000550            VRTP(JRTP,IS)=VCG(IQUAD,4)                                   00031410
000552            HRTP(JRTP,IS)=HC(IQUAD,4)                                    00031420
000554            IDRTP(JRTP,IS)= 3                                            00031430
000556            JRTP=JRTP+1                                                  00031440
000557            IF (JRTP.EQ.15) GO TO 52                                     00031450
000561            DRTP(JRTP,IS)=DRTP(JRTP-1,IS)+DTF(IQUAD)                     00031460
000564            TRTP(JRTP,IS)=TRTP(JRTP-1,IS)+TC(IQUAD,4)                    00031470
000570            VRTP(JRTP,IS)=VCG(IQUAD,3)                                   00031480
000571            HRTP(JRTP,IS)=HC(IQUAD,3)                                    00031490
000573            IDRTP(JRTP,IS)= 4                                            00031500
000575            JRTP=JRTP+1                                                  00031510
000577            IF (JRTP.EQ.15) GO TO 52                                     00031520
000601            ISIDE=1                                                      00031530
000602            IF (IQUAD.EQ.2) ISIDE=2                                      00031540
000605            IF (IQUAD.EQ.3) ISIDE=2                                      00031550
000610            DRTP(JRTP,IS)=DRTP(JRTP-1,IS)+DMF(ISIDE)                     00031560
000615            TRTP(JRTP,IS)=TRTP(JRTP-1,IS) + TC(IQUAD,3)                  00031570
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000621            VRTP(JRTP,IS)=VCG(IQUAD,2)                                    00031580
000623            HRTP(JRTP,IS)=HC(IQUAD,2)                                     00031590
000625            IDRTP(JRTP,IS)= 5                                             00031600
000627            JRTP=JRTP+1                                                   00031610
000630            IF (JRTP.EQ.15) GO TO 52                                      00031620
000632            IRNWY=1                                                       00031630
000633            IF((ISIDE.EQ.2).AND.(IRNWY.EQ.2))IRNWY=2                      00031640
000643            DRTP(JRTP,IS) = DRTP(JRTP-1,IS)+DFAF(IRNWY)                   00031650
000650            TRTP(JRTP,IS)=TRTP(JRTP-1,IS)+TC(IQUAD,2)                     00031660
000655            VRTP(JRTP,IS)=VCG(IQUAD,1)                                    00031670
000657            HRTP(JRTP,IS)=HC(IQUAD,1)                                     00031680
000661            IDRTP(JRTP,IS)= 6                                             00031690
000663            JRTP=JRTP+1                                                   00031700
000664            IF (JRTP.EQ.15) GO TO 52                                      00031710
000666            DRTP(JRTP,IS) = DRTP(JRTP-1,IS)+DCM(IRNWY)                    00031720
000672            TRTP(JRTP,IS)=SLT(IS)                                         00031730
000674            VRTP(JRTP,IS)=VBUGG(IS)                                       00031740
000676            HRTP(JRTP,IS)=THZ(IRNWY)                                      00031750
000701            IDRTP(JRTP,IS)= 7                                             00031760
          C                                                                     00031770
          C       CHECK RTP GENERATED FOR CONFLICTS WITH PRECEDING AIRCRAFT      00031780
          C                                                                     00031790
000702            CALL CNFLCTR(LACP(IQUAD,INFIX,IPATH),IS,IFLAG,IFLITX,NSAMP)   00031800
000712            IF (IFLAG.EQ.0) GO TO 90                                      00031810
          C                                                                     00031820
          C          IF A CONFLICT EXISTS TRY A NEW PATH                        00031830
          C                                                                     00031840
000715               IPATH = IPATH + 1                                          00031850
000716               IF (IPATH.LE.NPATH) GO TO 60                               00031860
000720               IPATH = NPATH                                              00031870
000721               PRINT 8001                                                 00031880
000724    8001       FORMAT (41HO----UNRESOLVABLE CONFLICT AT CNFLCTR---- )     00031890
000724    90      CONTINUE                                                      00031900
          C                                                                     00031910
          C       SET THIS AIRPLANE TO LAST ASSIGNED ON THIS PATH               00031920
          C       ADD PARALLEL NUMBER TO LFIX DATA. (NOW A THREE DIGIT NO.)     00031930
          C                                                                     00031940
000724            LACP(IQUAD,INFIX,IPATH)=IS                                    00031950
000732            LFIX(IS) = ILFIX*10 + IPATH                                   00031960
000734    100   CONTINUE                                                        00031970
          C                                                                     00031980
          C     PRINT RTPS THAT HAVE BEEN GENERATED, IF REQUIRED                00031990
          C                                                                     00032000
000740        ICHK = MOD(NSAMP,IFLITX)                                          00032010
000744        IF (ICHK.NE.0) RETURN                                             00032020
000745        PRINT 9008                                                        00032030
000751        DO 1000 I = 1, NACA                                               00032040
000754           IS = KEY(I)                                                    00032050
000756           DO 1001 J = 1, 14                                              00032060
000757              IF (IDRTP(J,IS).EQ.7) GO TO 1002                            00032070
000763    1001    CONTINUE                                                      00032080
000765    1002    CONTINUE                                                      00032090
000765            IF (MOD(I,3).LE.0) PRINT 9009                                 00032100
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

001000            PRINT 9011, I, IS, LFIX(IS), ((IDRTP(K,IS), TRTP(K,IS),      00032110
                 1              VRTP(K,IS), HRTP(K,IS), DRTP(K,IS)),  K = 1, J)  00032120
001037     1000 CONTINUE                                                        00032130
001043          RETURN                                                          00032140
                C                                                               00032150
001014     9008  FORMAT(1H1/, 22X, 31H***** ROUTE TIME PROFILES ***** //1HO,5X,  00032160
                1       23HRUNWAY ENTRY FIX QUAD.,,3X, 3HWAY, 4X, 3HWAY, 4X, 3HWAY,  00032170
                2       4X, 3HWAY, 4X, 4HCUM. / 6X,24HSEQUENCE ARRIVAL   FIX ,,3X,  00032180
                3       3HPT., 4X, 3HPT., 4X, 3HPT., 4X, 3HPT., 3X, 7HWAY PT. / 7X,  00032190
                4       22HORDER    ORDER    PATH, 4X, 4HTYPE, 3X, 4HTIME, 2X,     00032200
                5       5HSPEED, 3X, 4HALT., 3X, 5HDIST. / 22X, 1GHDESIGNATOR, 7X,  00032210
                6       6H(SECS), 7H(KNOTS), 8H(1000FT), 1X, 6H(N.M.) //)       00032220
                                                                                00032230
001044     9009  FORMAT(1H1//1HO,6X,                                            00032240
                1       23HRUNWAY ENTRY FIX QUAD.,,3X, 3HWAY, 4X, 3HWAY, 4X, 3HWAY,  00032250
                2       4X, 3HWAY, 4X, 4HCUM. / 6X,24HSEQUENCE ARRIVAL   FIX ,,3X,  00032260
                3       3HPT., 4X, 3HPT., 4X, 3HPT., 4X, 3HPT., 3X, 7HWAY PT. / 7X,  00032270
                4       22HORDER    ORDER    PATH, 4X, 4HTYPE, 3X, 4HTIME, 2X,     00032280
                5       5HSPEED, 3X, 4HALT., 3X, 5HDIST. / 22X, 1GHDESIGNATOR, 7X,  00032290
                6       6H(SECS), 7H(KNOTS), 8H(1000FT), 1X, 6H(N.M.) //)       00032300
001044     9011 FORMAT (6X,I4,6X,I4,5X,I4,4X,I2,2X,F6.0,2X,F6.1,2X,F5.1,2X,F6.1/  00032310
                1       13(33X,I2,2X,F6.0,2X,F6.1,2X,F5.1,2X,F6.1/))            00032320
                C                                                               00032330
001044          END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE SCHLD1(IFLITX,ISAMP)                                  00022410
      C                                                                          00022420
000005          COMMON /TRAFOT/                                                  00022430
                1    ALT(200),      EELT(200),     ELLT(200),     GWA(200),     00022440
                2    GWD(100),      ITYPEA(200),   ITYPED(100),   KEY(200),     00022450
                3    LFIX(200),     NACA,          NACD,          SDT(100),     00022460
                4    SLT(200),      SPEED(200),    TSECAA(200),   TSECAD(100),  00022470
                5    TSECSA(200),   TSECSD(100),   VDUGG(200),    VREF(200)     00022480
000005          DIMENSION           DLYA(200),     DLYD(100),     DLYEF(200)    00022490
000005          REAL                NASLOT,        NDSLOT                       00022500
      C                                                                          00022510
      C     INITIALIZE PARAMETERS                                                00022520
      C                                                                          00022530
000005          DO 110 I=1,200                                                   00022540
000006             DLYA(I)=0.                                                    00022550
000007             DLYEF(I)=0.                                                   00022560
000010        110 CONTINUE                                                       00022570
000012          DO 120 I=1,100                                                   00022580
000013             DLYD(I)=0.                                                    00022590
000014        120 CONTINUE                                                       00022600
000016          I=1                                                              00022610
000016          K=KEY(I)                                                         00022620
000020          NASLOT=0.                                                        00022630
      C                                                                          00022640
      C     SCHEDULE ARRIVALS WITH ABSOLUTE PRIORITY OVER DEPARTURES             00022650
      C                                                                          00022660
000022        150 CONTINUE                                                       00022670
000022             SLT(K)=NASLOT                                                 00022680
000024             IF (EELT(K).GT.NASLOT) SLT(K)=EELT(K)                         00022690
000030             DLYA(K)=SLT(K)-EELT(K)                                        00022700
000033             THOLD=SLT(K)-ELLT(K)                                          00022710
000035             IF (THOLD.GT.0.0) DLYEF(K)=THOLD                              00022720
000040             TSECAA(K)=TSECAA(K)+DLYEF(K)                                  00022730
      C                                                                          00022740
      C          STEP TO NEXT ARRIVAL                                          00022750
      C                                                                          00022760
000043             I=I+1                                                         00022770
000044             IF (I.GT.NACA) GO TO 300                                      00022780
000050                K1=K                                                       00022790
000050                K=KEY(I)                                                   00022800
000052                CALL TIMAA(T,K1,K)                                         00022810
000054                NASLOT=SLT(K1)+T                                           00022820
000057          GO TO 150                                                        00022830
      C                                                                          00022840
      C     SCHEDULE DEPARTURES BASED ON OPPORTUNITY SLOTS                       00022850
      C                                                                          00022860
000061        300 CONTINUE                                                       00022870
000061          J=1                                                              00022880
000062          NDSLOT=0.                                                        00022890
000063          I=1                                                              00022900
000064          K=KEY(I)                                                         00022910
000066        350 CONTINUE                                                       00022920
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000066                SDT(J)=NDSLOT                                          00022930
000070                IF (TSECAD(J).GT.NDSLOT) SDT(J)=TSECAD(J)              00022940
000074          300   CONTINUE                                               00022950
000074                IF (SDT(J).LT.SLT(K)) GO TO 370                        00022960
000100          365      CONTINUE                                            00022970
000100                   CALL TIMAD(T,K,J)                                   00022980
000103                   NDSLOT=SLT(K)+T                                     00022990
000105                   I=I+1                                               00023000
000107                   IF (I.GT.NACA) GO TO 500                            00023010
000114                      K=KEY(I)                                         00023020
000115                      GO TO 350                                        00023030
000115           370 CONTINUE                                               00023040
000115               CALL TIMDA(T,J,K)                                      00023050
000120               IF ((SLT(K)-SDT(J)).LT.T) GO TO 365                    00023060
000126               DLYD(J)=SDT(J)-TSECAD(J)                               00023070
000130               J=J+1                                                  00023080
000131               IF (J.GT.NACD) GO TO 500                               00023090
000135                  CALL TIMDD(T,J-1,J)                                 00023100
000140                  NDSLOT=SDT(J-1)+T                                   00023110
000143                  GO TO 350                                           00023120
000145           500 CONTINUE                                               00023130
000145               IF (MOD(ISAMP,IFLITX).NE.0) RETURN                     00023140
000151               CALL SCHWT(DLYEF,DLYA,DLYD)                            00023150
000154               RETURN                                                 00023160
000155               END                                                    00023170
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE SCHLD2(IFLITX,ISAMF)                                  00023180
      C                                                                          00023190
000005          COMMON /TRAFOT/                                                  00023200
               1     ALT(200),       EELT(200),      ELLT(200),      GWA(200),   00023210
               2     GWD(100),       ITYPEA(200),    ITYPED(100),    KEY(200),   00023220
               3     LFIX(200),      NACA,           NACD,           SDT(100),   00023230
               4     SLT(200),       SPEED(200),     TSECAA(200),    TSECAD(100),00023240
               5     TSECSA(200),    TSECSD(100),    VBUGG(200),     VREF(200)   00023250
000005          DIMENSION             DLYA(200),      DLYD(100),      DLYEF(200)  00023260
000005          REAL                  NASLOT,         NDSLOT                      00023270
      C                                                                          00023280
      C     INITIALIZE PARAMETERS                                                00023290
      C                                                                          00023300
000005          DO 110 I=1,200                                                   00023310
000006             DLYA(I)=0.                                                    00023320
000007             DLYEF(I)=0.                                                   00023330
000010    110 CONTINUE                                                         00023340
000012          DO 120 I=1,100                                                   00023350
000013             DLYD(I)=0.                                                    00023360
000014    120 CONTINUE                                                         00023370
000016          I=1                                                              00023380
000016          K=KEY(I)                                                         00023390
000020          NASLOT=0.                                                        00023400
      C                                                                          00023410
      C     SCHEDULE ARRIVAL LIST LEAVING DEPARTURE HOLES                        00023420
      C                                                                          00023430
000022    150 CONTINUE                                                         00023440
000022             SLT(K)=NASLOT                                                 00023450
000024             IF (EELT(K).GT.NASLOT) SLT(K)=EELT(K)                         00023460
000030             DLYA(K)=SLT(K)-EELT(K)                                        00023470
000033             THOLD=SLT(K)-ELLT(K)                                          00023480
000035             IF (THOLD.GT.0.0) DLYEF(K)=THOLD                              00023490
000040             TSECAA(K)=TSECAA(K)+DLYEF(K)                                  00023500
      C                                                                          00023510
      C        STEP TO THE NEXT ARRIVAL                                          00023520
      C                                                                          00023530
000043             I=I+1                                                         00023540
000044             IF (I.GT.NACA) GO TO 300                                      00023550
000050                K1=K                                                       00023560
000050                K=KEY(I)                                                   00023570
000052                J=0                                                        00023580
000052                TMAX=0.                                                    00023590
000054    160         CONTINUE                                                   00023600
000054                   J=J+1                                                   00023610
000056                   IF (J.GT.NACD) GO TO 170                                00023620
000061                      CALL TIMAD(T1,K1,J)                                  00023630
000063                      CALL TIMDA(T2,J,K)                                   00023640
000066                      IF ((T1+T2).GT.TMAX) TMAX=T1+T2                      00023650
000075                GO TO 160                                                  00023660
000076    170         CONTINUE                                                   00023670
000076                TMAX = TMAX + 1.                                           00023680
000100                CALL TIMAA(T3,K1,K)                                        00023690
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000103                IF (T3.GT.TMAX) TMAX = T3                          00023700
000110                NASLOT=SLT(K1)+TMAX                                00023710
000113             GO TO 150                                             00023720
           C                                                             00023730
           C        SCHEDULE DEPARTURES BASED ON OPPORTUNITY SLOTS       00023740
           C                                                             00023750
000113     300 CONTINUE                                                  00023760
000113         J=1                                                       00023770
000114         NDSLOT=0.                                                 00023780
000115         I=1                                                       00023790
000116         K=KEY(I)                                                  00023800
000120     350 CONTINUE                                                  00023810
000120             SDT(J)=NDSLOT                                         00023820
000122             IF (TSECAD(J).GT.NDSLOT) SDT(J)=TSECAD(J)             00023830
000126     360     CONTINUE                                              00023840
000126             IF (SDT(J).LT.SLT(K)) GO TO 370                       00023850
000132     365        CONTINUE                                           00023860
000132                CALL TIMAD(T,K,J)                                  00023870
000135                NDSLOT=SLT(K)+T                                    00023880
000137                I=I+1                                              00023890
000141                IF (I.GT.NACA) GO TO 500                           00023900
000146                   K=KEY(I)                                        00023910
000147                   GO TO 350                                       00023920
000147     370 CONTINUE                                                  00023930
000147         CALL TIMDA(T,J,K)                                         00023940
000152         IF ((SLT(K)-SDT(J)).LT.T) GO TO 365                       00023950
000160         DLYD(J)=SDT(J)-TSECAD(J)                                  00023960
000162         J=J+1                                                     00023970
000163         IF (J.GT.NACD) GO TO 500                                  00023980
000167            CALL TIMDD(T,J-1,J)                                    00023990
000172            NDSLOT=SDT(J-1)+T                                      00024000
000175            GO TO 350                                              00024010
000177     500 CONTINUE                                                  00024020
000177         IF (MOD(ISAMP,IFLITX).NE.0) RETURN                        00024030
000203         CALL SCHWT(DLYEF,DLYA,DLYD)                               00024040
000206         RETURN                                                    00024050
000207         END                                                       00024060
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE SCHLD3(IFLITX,ISAMF)                                 00024070
      C                                                                         00024080
000005          COMMON /TRAFOT/                                                 00024090
               1    ALT(200),       FELT(200),      ELLT(200),      GWA(200),   00024100
               2    GWD(100),       ITYPEA(200),    ITYPED(100),    KEY(200),   00024110
               3    LFIX(200),      NACA,           NACD,           SDT(100),   00024120
               4    SLT(200),       SPEED(200),     TSECAA(200),    TSECAD(100),00024130
               5    TSECSA(200),    TSECSD(100),    VDUGG(200),     VREF(200)   00024140
000005          DIMENSION           DLYA(200),      DLYD(100),      DLYEF(200)  00024150
000005          REAL                NASLOT,         NDSLOT                      00024160
      C                                                                         00024170
      C     INITIALIZE PARAMETERS                                               00024180
      C                                                                         00024190
000005          DO 110 I=1,200                                                  00024200
000006             DLYA(I)=0.                                                   00024210
000007             DLYEF(I)=0.                                                  00024220
000010       110 CONTINUE                                                       00024230
000012          DO 120 I=1,100                                                  00024240
000013             DLYD(I)=0.                                                   00024250
000014       120 CONTINUE                                                       00024260
000016          I=1                                                             00024270
000016          K=KEY(I)                                                        00024280
000020          J=1                                                             00024290
000021          NASLOT=0.                                                       00024300
000022          NDSLOT=0.                                                       00024310
000024       150 CONTINUE                                                       00024320
000024             IF (TSECSD(J).LT.EELT(K)) GO TO 200                          00024330
000030                GO TO 300                                                 00024340
      C                                                                         00024350
      C             CONSIDER NEXT DEPARTURE FOR SCHEDULING                      00024360
      C                                                                         00024370
000030       200    CONTINUE                                                    00024380
000030             SDT(J)=TSECSD(J)                                             00024390
000032             IF (SDT(J).LT.NDSLOT) SDT(J)=NDSLOT                          00024400
000036             DLYD(J)=SDT(J)-TSECSD(J)                                     00024410
000041             J1=J                                                         00024420
000041             J=J+1                                                        00024430
000043             IF ((J.GT.NACD).AND.(I.GT.NACA)) GO TO 500                   00024440
000054             IF (J.GT.NACD) GO TO 250                                     00024450
000056                CALL TIMDD(T,J1,J)                                        00024460
000060                NDSLOT=SDT(J1)+T                                          00024470
000062                IF (I.GT.NACA) GO TO 200                                  00024480
000070                   CALL TIMDA(T,J1,K)                                     00024490
000072                   NASLOT=SDT(J1)+T                                       00024500
000074                   IF (EELT(K).LT.NDSLOT) GO TO 300                       00024510
000101                   GO TO 150                                              00024520
000102       250 CONTINUE                                                       00024530
000102          CALL TIMDA(T,J1,K)                                              00024540
000105          NASLOT=SDT(J1)+T                                                00024550
      C                                                                         00024560
      C     CONSIDER NEXT ARRIVAL FOR SCHEDULING                                00024570
      C                                                                         00024580
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000110     300 CONTINUE                                                  00024590
000110         SLT(K)=EELT(K)                                            00024600
000112         IF (SLT(K).LT.NASLOT) SLT(K)=NASLOT                       00024610
000120         DLYA(K)=SLT(K)-EELT(K)                                    00024620
000124         THOLD=SLT(K)-ELLT(K)                                      00024630
000125         IF (THOLD.GT.0.0) DLYEF(K)=THOLD                          00024640
000130         TSECAA(K)=TSECAA(K)+THOLD                                 00024650
000132         K1=K                                                      00024660
000133         I=I+1                                                     00024670
000135         IF((J.GT.NACD).AND.(I.GT.NACA)) GO TO 500                 00024680
000146            IF (J.GT.NACD) GO TO 350                               00024690
000150               CALL TIMAD(T,K1,J)                                  00024700
000152               NDSLOT=SLT(K1)+T                                    00024710
000154               IF (I.GT.NACA) GO TO 200                            00024720
000162                  K=KEY(I)                                         00024730
000163                  CALL TIMAA(T,K1,K)                               00024740
000165                  NASLOT=SLT(K1)+T                                 00024750
000167                  IF (TSECSD(J).LT.NASLOT) GO TO 200               00024760
000174                  GO TO 150                                        00024770
000175     350     CONTINUE                                              00024780
000175             K=KEY(I)                                              00024790
000177             CALL TIMAA(T,K1,K)                                    00024800
000202             NASLOT=SLT(K1)+T                                      00024810
000205             GO TO 300                                             00024820
000207     500 CONTINUE                                                  00024830
000207         IF (MOD(ISAMP,IFLITX).NE.0) RETURN                        00024840
000213         CALL SCHWT(DLYEF,DLYA,DLYD)                               00024850
000216         RETURN                                                    00024860
000217         END                                                       00024870
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE SCHWT(AHCLDT, DLAY, DLAYD)                                   00026890
      C                                                                                 00026900
000006          COMMON /TRAFOT/                                                         00026910
               1    ALT(200),       EELT(200),      ELLT(200),      GWA(200),         00026920
               2    GWD(100),       ITYPEA(200),    ITYPED(100),    KEY(200),         00026930
               3    LFIX(200),      NACA,           NACD,           SDT(100),         00026940
               4    SLT(200),       SPEED(200),     TSECAA(200),    TSECAD(100),      00026950
               5    TSECSA(200),    TSECSD(100),    VBUGG(200),     VREF(200)         00026960
000006          DIMENSION           AHCLDT(1),      DLAY(1),        DLAYD(1)          00026970
      C                                                                                 00026980
000006          PRINT 2000                                                            00026990
000011          PRINT 2001                                                            00027000
000015          DO 10 I=1,NACA                                                        00027010
000021             PRINT 2100,I,KEY(I),EELT(I),ELLT(I),AHCLDT(I),DLAY(I),SLT(I)       00027020
000060             IF (I/34*34.NE.I) GO TO 10                                         00027030
000067                PRINT 2002                                                      00027040
000073                PRINT 2001                                                      00027050
000077    10    CONTINUE                                                            00027060
000104          IF (NACD.LT.1) RETURN                                                 00027070
000107          PRINT 2010                                                            00027080
000113          PRINT 2011                                                            00027090
000117          DO 20 I = 1, NACD                                                     00027100
000123             PRINT 2200, I, SDT(I), DLAYD(I)                                    00027110
000141             IF (I/41*41.NE.I) GO TO 20                                         00027120
000151                PRINT 2002                                                      00027130
000154                PRINT 2011                                                      00027140
000160    20    CONTINUE                                                            00027150
000165          RETURN                                                                00027160
      C                                                                             00027170
000165    2000  FORMAT(1H1, 35X, 33H*****ARRIVAL RUNWAY SCHEDULE***** )              00027180
000165    2001  FORMAT(1HO, 10X, 5HENTRY,5X, 6HRUNWAY, 6X, 9HESTIMATED, 7X,          00027190
               1       9HESTIMATED, 15X, 17HS C H E D U L E D / 12X,3HFIX, 5X,      00027200
               2       8HSEQUENCE, 6X,, 8HEARLIEST, 8X, 6HLATEST, 11X, 4HHOLD,      00027210
               3       10X, 5HDELAY, 8X, 7HLANDING / 10X,7HARRIVAL, 17X,            00027220
               4       7HLANDING, 9X, 7HLANDING, 10X, 4HTIME, 10X, 4HTIME, 11X,     00027230
               5       4HTIME / 11X, 5HORDER, 20X, 4HTIME, 11X, 4HTIME, 11X,        00027240
               6       6H(SECS), 9X, 6H(SECS), 8X, 6H(SECS) / 35X, 6H(SECS),        00027250
               7       9X, 6H(SECS) /)                                              00027260
000165    2002  FORMAT(1H1)                                                          00027270
000165    2010  FORMAT(1H1, 10X, 35H*****DEPARTURE RUNWAY SCHEDULE***** //)          00027280
000165    2011  FORMAT(1HO, 9X, 9HDEPARTURE, 5X, 9HSCHEDULED / 12X, 5HORDER, 7X,     00027290
               1       9HDEPARTURE, 8X, 5HDELAY / 26X, 4HTIME, 11X, 4HTIME / 25X,   00027300
               2       6H(SECS) //)                                                 00027310
000165    2100  FORMAT(5X,2I10,5(5X,F10.0))                                         00027320
000165    2200  FORMAT(5X,I10,2(5X,F10.0))                                          00027330
000165          END                                                                   00027340
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE SEQUEN(IFLITX, ISAMP)                                        00020410
          C                                                                             00020420
000005          COMMON /AEROIN/                                                       00020430
                1    CABRS(10),     HA(10,7),      HCRITN(10),    NAERO(10),         00020440
                2    VFAST1(10,7),  VFAST2(10,7),  VSLOW1(10,7),  VSLOW2(10,7),      00020450
                3    W1(10),        W2(10)                                           00020460
000005          COMMON /AERO1O/                                                       00020470
                1    HC(4,5),       TC(4,5),       TCSUM(4),      THPRES,            00020480
                2    VCCAS(5),      VCG(4,5),      VCTAS(4,5)                        00020490
000005          COMMON /ATMO/                                                         00020500
                1    DENSR,         DENSRF,        DENSTY,        PRESR,             00020510
                2    PRESUR,        TEMR,          TEMRS                             00020520
000005          COMMON /GEOMIN/                                                       00020530
                1    DPARAL,        EFX(4,3),      EFY(4,3),      EFZ(4,3),          00020540
                2    FAFX(2),       FAFY(2),       FAFZ(2),       IAFX(4),           00020550
                3    IAFY(4),       IAFZ(4),       MAX(2),        MAY(2),            00020560
                4    MAZ(2),        MFX(2),        MFY(2),        MFZ(2),            00020570
                5    NEF,           NPATH,         NQUAD,         NRNWY,             00020580
                6    OMX(2),        OMY(2),        OMZ(2),        RHOEF,             00020590
                7    RHOFAF,        RHOIAF,        RHOMF,         RHOOM,             00020600
                8    RHOTF,         TFX(4),        TFY(4),        TFZ(4),            00020610
                9    THX(2),        THY(2),        THZ(2)                            00020620
000005          REAL                IAFX,          IAFY,          IAFZ,              00020630
                1    MAX,           MAY,           MAZ,           MFX,               00020640
                2    MFY,           MFZ                                              00020650
000005          COMMON /GEOMOT/                                                       00020660
                1    ANGEF(4,3),    ANGFAF(2),     ANGIAF(4),     ANGMF(2),          00020670
                2    ANGOM(2),      ANGTF(4),      DEF(4,3,5),    DFAF(2),           00020680
                3    DIAF(4),       DMF(2),        DOM(2),        DPATH(4,3,5),      00020690
                4    DTF(4),        DTH(2)                                           00020700
000005          COMMON /TRAFOT/                                                       00020710
                1    ALT(200),      EELT(200),     ELLT(200),     GWA(200),          00020720
                2    GWD(100),      ITYPEA(200),   ITYPED(100),   KEY(200),          00020730
                3    LFIX(200),     NACA,          NACD,          SDT(100),          00020740
                4    SLT(200),      SPEED(200),    TSECAA(200),   TSECAD(100),       00020750
                5    TSECSA(200),   TSECSD(100),   VDUGG(200),    VREF(200)          00020760
          C                                                                             00020770
000005          CALL REFDUG                                                           00020780
000006          DO 500 I=1, NACA                                                      00020790
000011             IQUAD=LFIX(I)/10                                                   00020800
000014             INUMB=LFIX(I)-IQUAD*10                                             00020810
000017             DIST = DOM(1)                                                      00020820
000021             IF ((NRNWY.NE.1).AND.(IQUAD.EQ.2 .OR. IQUAD.EQ.3)) DIST=DOM(2)     00020830
          C                                                                         00020840
          C          T1       TIME REQUIRED FROM THRESHOLD TO OUTER MARKER PLUS TIMEDO020850
          C          FROM IAF TO OM.                                                00020860
          C          T3       TIME REQUIRED FROM EF TO IAF                          00020870
          C                                                                         00020880
000035             T1 =  7200.*DIST/(VDUGG(I) + VCG(1,1)) + TCSUM(IQUAD)              00020890
000043             CALL AERO (INDX, 1, IQUAD, INUMB, 0)                               00020900
000046             CALL FASLOW (IQUAD, INUMB, 1, T3F, T3S)                            00020910
000052             TFAST=T1+T3F                                                       00020920
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000054            TSLOW=T1+T3S                                                  00020930
          C                                                                     00020940
          C       COMPUTE CABIN RE-PRESSURIZATION TIME                          00020950
          C                                                                     00020960
000056            CALL WETHER (ALT(I), D)                                       00020970
000061            TREPRS=(THPRES-PRESUR+CABPRS(INDX)*144.0)/22.9                00020980
000067            EELT(I) = TSECAA(I) + TFAST                                   00020990
000072            IF (TFAST*60.0.LE.TREPRS) EELT(I) = TSECAA(I) + TREPRS*60.0   00021000
000101            ELLT(I)=TSECAA(I)+TSLOW                                       00021010
000104      500 CONTINUE                                                        00021020
          C                                                                     00021030
          C     SORT EELT(I) ARRAY TO ESTABLISH SEQUENCING LIST KEY(I)          00021040
          C                                                                     00021050
000106          KEY(1)=1                                                      00021060
000107          VMIN=EELT(1)                                                  00021070
000111          IK=1                                                          00021080
000112      600 CONTINUE                                                      00021090
000112            DO 700 I=1, NACA                                            00021100
          C                                                                     00021120
          C           SKIP THE ELEMENTS WHICH ARE ALREADY SEQUENCED           00021130
          C                                                                     00021140
000114              IK1=IK-1                                                  00021110
000116              DO 650 J=1, IK1                                           00021150
000120                 IF (I.EQ.KEY(J)) GO TO 700                             00021160
000122      650       CONTINUE                                                00021170
000124                 IF (VMIN.LT.EELT(I)) GO TO 700                         00021180
000127                    VMIN=EELT(I)                                        00021190
000130                    KEY(IK)=I                                           00021200
000132      700    CONTINUE                                                   00021210
000135             IF (IK.EQ.NACA) RETURN                                     00021220
000137             IK=IK+1                                                    00021230
000141             VMIN=1.0E20                                                00021240
000142          GO TO 600                                                     00021250
000143          END                                                           00021260
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE SETN(NX)                                                    00001070
          C                                                                            00001080
          C     SETN SETS VARIABLES TO NX OR FLOAT(NX) BEFORE EACH DATA                00001090
          C     SET IS RUN.                                                            00001100
          C                                                                            00001110
000003            COMMON /FLGTXCV/                                                   0000.120
                 1    DISSEP(16,17), DLY(6,17),      IHIST(5),      NAR,             00001130
                 2    ND(16),        NDCR(16),       NDECR(6),      NDEL(6),         00001140
                 3    NDEP,          NHIST,          NOCF(16),      NP(36),          00001150
                 4    NRD(4),        NRDCR(4),       NRECOD(5),     NRT(4),          00001160
                 5    NRTCR(4),      NT(16),         NTCR(16),      OPTIM(4,17),     00001170
                 6    SUM(6),        SUMSQ(6),       SUMTIM,        TIMSEP(16,17),   00001180
                 7    TIPATH(36,17)                                                  00001190
000003            INTEGER            DISSEP,         DLY,           OPTIM,           00001200
                 1    TIMSEP,        TIPATH                                         00001210
000003            COMMON /STATOT/                                                    00001220
                 1    ARRATE,        DELYOA,         DELYOP,        DELYOV,          00001230
                 2    DELYSA,        DELYSP,         DELYSV,        DELYTA,          00001240
                 3    DELYTP,        DELYTV,         DERATE,        DLYOA(17),       00001250
                 4    DLYOD(17),     DLYSA(17),      DLYSD(17),     DLYSTP(17),      00001260
                 5    DLYTA(17),     DLYTD(17),      DSEP(17),      OPORAT(16),      00001270
                 6    OPPRAT(36),    OPSTEP(17),     OPTIM1(17),    OPTIM2(17),      00001280
                 7    OPTIM3(17),    OPTIM4(17),     PASTIM(36,17), PDMIN(16),       00001290
                 8    PDMIN1,        PDMIN2,         PDMIN3,        PDMIN4,          00001300
                 9    PTMIN(16),     PTMIN1,         PTMIN2,        PTMIN3,          00001310
                 A    PTMIN4,        SEPDIS(16,17), SEPTIM(16,17), TARR(17),        00001320
                 B    TOTAL(36),     TSEP(17)                                       00001330
000003            INTEGER            DLYSTP,         OPSTEP,        TARR             00001340
          C                                                                          00001350
          C     SET COMMON/FLGTXCV TO NX                                             00001360
          C                                                                          00001370
000003            XN = FLOAT(NX)                                                     00001380
000004            SUMTIM = XN                                                        00001390
000005            NAR = NX                                                           00001400
000006            NDEP = NX                                                          00001410
000007            NHIST = NX                                                         00001420
000010            DO 10 I=1,4                                                        00001430
000011            NRTCR(I) = NX                                                      00001440
000013            NRT(I) = NX                                                        00001450
000014            NRDCR(I) = NX                                                      00001460
000015            NRD(I)= NX                                                         00001470
000016            DO 10 J=1,17                                                       00001480
000017               OPTIM(I,J) = NX                                                 00001490
000023         10 CONTINUE                                                         00001500
000026            DO 15 I=1,16                                                       00001510
000030               NOCF(I) = NX                                                    00001520
000032               NTCR(I) = NX                                                    00001530
000033               NT(I) = NX                                                      00001540
000034               NDCR(I) = NX                                                    00001550
000035               ND(I) = NX                                                      00001560
000036               DO 15 J=1,17                                                    00001570
000037                  TIMSEP(I,J) = NX                                             00001580
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000042                DISSEP(I,J) = NX                                  00001590
000044          15 CONTINUE                                             00001600
000050             DO 20 I=1,36                                         00001610
000051                NP(I) = NX                                        00001620
000053                DO 20 J=1,17                                      00001630
000054                   TIPATH(I,J)= NX                                00001640
000061          20 CONTINUE                                             00001650
000064             DO 25 I=1,6                                          00001660
000066                SUM(I) = XN                                       00001670
000070                SUMSQ(I) = XN                                     00001680
000071                NDECR(I) = NX                                     00001690
000072                NDEL(I) = NX                                      00001700
000073                DO 25 J=1,17                                      00001710
000075                   DLY(I,J) = NX                                  00001720
000102          25 CONTINUE                                             00001730
000105             DO 30 I=1,5                                          00001740
000107                IHIST(I) =NX                                      00001750
000111                NRECCD(I) = NX                                    00001760
000112          30 CONTINUE                                             00001770
                C                                                       00001780
                C     SET COMMON/STATOT/ TO NX                          00001790
                C                                                       00001800
000113             DO 12 I=1,17                                         00001810
000115                DLYCA(I) = XN                                     00001820
000117                DLYCD(I) = XN                                     00001830
000120                DLYSA(I) = XN                                     00001840
000121                DLYSD(I) = XN                                     00001850
000122                DLYSTP(I) = NX                                    00001860
000123                DLYTA(I) = XN                                     00001870
000124                DLYTD(I) = XN                                     00001880
000125                DSEP(I) = XN                                      00001890
000126                OPSTEP(I) = NX                                    00001900
000127                OPTIM1(I) = XN                                    00001910
000130                OPTIM2(I) = XN                                    00001920
000131                OPTIM3(I) = XN                                    00001930
000132                OPTIM4(I) = XN                                    00001940
000133                TSEP(I) = XN                                      00001950
000134                TARR(I) = NX                                      00001960
000135                DO 12 J=1,16                                      00001970
000137                   SEPDIS(J,I) = XN                               00001980
000142                   SEPTIM(J,I) = XN                               00001990
000144          12 CONTINUE                                             00002000
000150             DO 17 I=1,36                                         00002010
000151                OPPRAT(I) = XN                                    00002020
000153                TOTAL(I) = XN                                     00002030
000154                DO 17 J=1,17                                      00002040
000155                   PASTIM(I,J) = XN                               00002050
000162          17 CONTINUE                                             00002060
000165             DO 22 I=1,16                                         00002070
000167                PDMIN(I) = XN                                     00002080
000171                PTMIN(I) = XN                                     00002090
000172                OPCRAT(I) = XN                                    00002100
000173          22 CONTINUE                                             00002110
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000174          ARRATE = XN                                    00002120
000175          DERATE = XN                                    00002130
000176          PTMIN1 = XN                                    00002140
000177          PTMIN2 = XN                                    00002150
000200          PTMIN3 = XN                                    00002160
000201          PTMIN4 = XN                                    00002170
000202          PDMIN1 = XN                                    00002180
000203          PDMIN2 = XN                                    00002190
000204          PDMIN3 = XN                                    00002200
000205          PDMIN4 = XN                                    00002210
000206          DELYSA = XN                                    00002220
000207          DELYSV = XN                                    00002230
000210          DELYSP = XN                                    00002240
000211          DELYOA = XN                                    00002250
000212          DELYOV = XN                                    00002260
000213          DELYOP = XN                                    00002270
000214          DELYTA = XN                                    00002280
000215          DELYTV = XN                                    00002290
000216          DELYTP = XN                                    00002300
000217          RETURN                                         00002310
000217          END                                            00002320
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000425            DRTP(JRTP,IS)=DRTP(JRTP+1,IS)-                    00038050
000432            TKTP(JRTP,IS)=TKTP(JRTP+1,IS)                     00038060
000434            VRTP(JRTP,IS)=VRTP(JRTP+1,IS)                     00038070
000437            HRTP(JRTP,IS)=HRTP(JRTP+1,IS)                     00038080
000441            IDRTP(JRTP,IS)=IDRTP(JRTP+1,IS)                   0003809C
000444       210 CONTINUE                                           00038100
000445          RETURN                                              00038110
000446          END                                                 00038120
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                    SUBROUTINE STUFF2(VNEW)                                          00038130
               C                                                                     00038140
               C    THIS SUBROUTINE SETS UP THE RTPOUT ARRAY WITH A PROPER SET       00038150
               C    OF VARIABLES TO ALLOW AN RTP INTEGRATION BY SUBROUTINE TFIND.    00038160
               C    STUFF2 IS USED WHEN AN AIRPLANE ARRIVES ABOVE THE CRITICAL       00038170
               C    ALTITUDE AND THE DESCENT WILL BE MADE VIA CONSTANT               00038180
               C    MACH TO THE CRITICAL ALTITUDE, AND THEN VIA A CONSTANT           00038190
               C    CALIBRATED AIRSPEED.                                             00038200
               C                                                                     00038210
               C    THIS SUBROUTINE CALLS THE FOLLOWING SUBROUTINES                  00038220
               C        TASCAS                                                       00038230
               C        CASTAS                                                       00038240
               C        SQRT                                                         00038250
               C                                                                     00038260
000003              COMMON /AEROOT/                                                  00038270
                   1    HD(10),          HCRIT,          JMAX,          SPKNOT,     00038280
                   2    VFAST(10),       VSLOW(10)                                 00038290
000003              COMMON /GEOMOT/                                                  00038300
                   1    ANGEF(4,3),      ANGFAF(2),      ANGIAF(4),     ANGMF(2),   00038310
                   2    ANGOM(2),        ANGTF(4),       DEF(4,3,5),    DFAF(2),    00038320
                   3    DIAF(4),         DMF(2),         DOM(2),        DPATH(4,3,5), 00038330
                   4    DTF(4),          DTH(2)                                    00038340
000003              COMMON /RTFC/                                                    00038350
                   1    DDP,             DLF,            HIAF,          I,          00038360
                   2    INFIX,           IQUAD,          IS,            TIAF        00038370
000003              COMMON /RTPOUT/                                                  00038380
                   1    DRTP(14,200),  HRTP(14,200),  IDRTP(14,200), TRTP(14,200), 00038390
                   2    VRTP(14,200)                                              00038400
000003              COMMON /TEMPER/                                                  00038410
                   1    GTEMP1,          GTEMP2,         GTEMP3,        HTEMP0,     00038420
                   2    HTEMP1,          HTEMP2,         TEMP,          TEMP1,      00038430
                   3    TEMP2,           TGROND,         TEROR1,        TEROR2,     00038440
                   4    TEROR3,          TSTAND                                    00038450
000003              COMMON /TRAFOT/                                                  00038460
                   1    ALT(200),        EELT(200),      ELLT(200),     GWA(200),   00038470
                   2    GWD(100),        ITYPEA(200),    ITYPED(100),   KEY(200),   00038480
                   3    LFIX(200),       NACA,           NACD,          SDT(100),   00038490
                   4    SLT(200),        SPEED(200),     TSECAA(200),   TSECAD(100), 00038500
                   5    TSECSA(200),     TSECSD(100),    VBUGG(200),    VREF(200)   00038510
000003              COMMON /WIND/                                                    00038520
                   1    GWINE1,          GWINE2,         GWINE3,        GWINN1,     00038530
                   2    GWINN2,          GWINN3,         HWIND0,        HWIND1,     00038540
                   3    HWIND2,          WERRE1,         WERRE2,        WERRE3,     00038550
                   4    WERRN1,          WERRN2,         WERRN3,        WINANG,     00038560
                   5    WINDE,           WINDEO,         WINDE1,        WINDE2,     00038570
                   6    WINDN,           WINDNO,         WINDN1,        WINDN2,     00038580
                   7    WINMAG                                                    00038590
000003              REAL                 MACH                                     00038600
               C                                                                     00038610
               C    STORE ENTRY GROUND SPEED,  ALTITUDE AND ID                       00038620
               C                                                                     00038630
000003              VRTP(1,IS) = SPKNOT                                              00038640
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION( OCT 73 A          16.56.27. 73/12/21.

000007          HRTP(1,IS)=ALT(IS)                                      00038650
000011          IDRTP(1,IS)=1                                           00038660
                C                                                       00038670
                C     CONVERT THE START OF DESCENT G/S TO TAS, MACH AND CAS  00038680
                C     STORE G/S WITH ALT AND ID                         00038690
                C                                                       00038700
000014          CALL WETHER(ALT(IS),0)                                  00038710
000017          VATW = WINMAG*COS(ANGEF(IQUAD,INFIX) - WINANG)          00038720
000027          MACH = (VNEW - VATW)/(38.975*SQRT(TEMP))                00038730
000035          VRTP(2,IS)=VNEW                                         00038740
000040          HRTP(2,IS)=ALT(IS)                                      00038750
000043          IDRTP(2,IS)=0                                           00038760
                C                                                       00038770
                C     COMPUTE UPPER MIDPOINT ALTITUDE                   00038780
                C     COMPUTE WIND AND TEMPERATURE AT HMIDUP            00038790
                C     CONVERT MACH TO GROUND SPEED AND STORE WITH ALT. AND ID  00038800
                C                                                       00038810
000046          HMIDUP=(ALT(IS)+HCRIT)/2.                               00038820
000051          CALL WETHER(HMIDUP,0)                                   00038830
000053          VATW = WINMAG*COS(ANGEF(IQUAD,INFIX)-WINANG)            00038840
000063          VRTP(3,IS)=38.975*SQRT(TEMP)*MACH + VATW                00038850
000072          HRTP(3,IS)=HMIDUP                                       00038860
000075          IDRTP(3,IS)=0                                           00038870
                C                                                       00038880
                C     COMPUTE WIND AND TEMPERATURE AT HCRIT             00038890
                C     CONVERT MACH TO TAS,CAS AND G/S AT HCRIT          00038900
                C     STORE G/S WITH ALTITUDE AND ID                    00038910
                C                                                       00038920
000100          CALL WETHER(HCRIT,0)                                    00038930
000102          VATW = WINMAG*COS(ANGEF(IQUAD,INFIX)-WINANG)            00038940
000112          VTAS = 38.975*SQRT(TEMP)*MACH                           00038950
000116          CALL TASCAS(HCRIT,VTAS,VCAS)                            00038960
000120          VRTP(4,IS)=VTAS + VATW                                  00038970
000125          HRTP(4,IS)=HCRIT                                        00038980
000127          IDRTP(4,IS) = 0                                         00038990
                C                                                       00039000
                C     COMPUTE LOWER MIDPOINT ALTITUDE                   00039010
                C     COMPUTE WIND AND TEMPERATURE AT HMIDLO            00039020
                C     CONVERT CAS TO TAS AND G/S AT HMIDLO              00039030
                C     STORE G/S WITH ALT AND ID                         00039040
                C                                                       00039050
000132          HMIDLO=(HCRIT+HIAF)/2.                                  00039060
000134          CALL WETHER(HMIDLO,0)                                   00039070
000136          VATW = WINMAG*COS(ANGEF(IQUAD,INFIX)-WINANG)            00039080
000146          CALL CASTAS (HMIDLO,VCAS,VTAS)                          00039090
000150          VRTP(5,IS)=VTAS + VATW                                  00039100
000155          HRTP(5,IS)=HMIDLO                                       00039110
000157          IDRTP(5,IS)=0                                           00039120
                C                                                       00039130
                C     ASSURE CAS AT LEAST 250 KNOTS ON ENTRY TO DDP SEGMENT  00039140
                C     COMPUTE WIND AND TEMPERATURE AT ENTRY TO DDP SEGMENT  00039150
                C     COMPUTE TAS AND G/S AT ENTRY TO DDP SEGMENT       00039160
                C     STORE G/S WITH ALT AND ID                         00039170
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                                                                        00039180
      C                                                                 00039190
000162        IF (VCAS.LT.250.) VCAS=250.                               00039200
000166        CALL WETHER(HIAF,0)                                       00039210
000170        VATW = WINMAG*COS(ANGEF(IQUAD,INFIX)-WINANG)              00039220
000200        CALL CASTAS (HIAF,VCAS,VTAS)                              00039230
000202        VRTP(6,IS)=VTAS + VATW                                    00039240
000207        HRTP(6,IS)=HIAF                                           00039250
000211        IDRTP(6,IS)=0                                             00039260
      C                                                                 00039270
      C     GET IAF TRACK ANGLE AND WIND                                00039280
      C     DETERMINE IAF G/S AND STORE WITH ALT AND ID                 00039290
      C                                                                 00039300
000214        CALL WETHER(HIAF,0)                                       00039310
000216        VATW = WINMAG*COS(ANGIAF(IQUAD)-WINANG)                   00039320
000224        VCAS=250.                                                 00039330
000225        CALL CASTAS (HIAF,VCAS,VTAS)                              00039340
000230        HRTP(7,IS)=HIAF                                           00039350
000234        IDRTP(7,IS)=2                                             00039360
      C                                                                 00039370
      C     SET EF TIME AND MAJOR SEGMENT DISTANCES IN RTPOUT           00039380
      C                                                                 00039390
000236        VRTP(7,IS)=VTAS + VATW                                    00039400
000242        TRTP(1,IS)=TIAF                                           00039410
000245        DRTP(1,IS)=0.                                             00039420
000250        DRTP(2,IS)=DLF                                            00039430
000252        DRTP(7,IS)=DEP                                            00039440
000255        RETURN                                                    00039450
000256        END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE STUFF3 (VNEW)                                        00039460
      C                                                                         00039470
      C     THIS SUBROUTINE SETS UP THE RTPOUT ARRAY WITH A PROPER SET        00039480
      C     OF VARIABLES TO ALLOW AN RTP INTEGRATION BY SUBROUTINE TFIND.     00039490
      C     STUFF3 IS USED WHEN AN AIRPLANE ARRIVES BELOW THE CRITICAL        00039500
      C     ALTITUDE AND THE DESCENT WILL BE VIA CONSTANT CALIBRATED          00039510
      C     AIRSPEED.                                                         00039520
      C                                                                         00039530
000003          COMMON /AEROOT/                                               00039540
               1    HB(10),          HCRIT,          JMAX,           SPKNOT,  00039550
               2    VFAST(10),       VSLOW(10)                                00039560
000003          COMMON /GEOMOT/                                               00039570
               1    ANGEF(4,3),      ANGFAF(2),      ANGIAF(4),      ANGMF(2),00039580
               2    ANGOM(2),        ANGTF(4),       DEF(4,3,5),     DFAF(2), 00039590
               3    DIAF(4),         DMF(2),         DOM(2),         DPATH(4,3,5),00039600
               4    DTF(4),          DTH(2)                                   00039610
000003          COMMON /RTPO/                                                 00039620
               1    DDP,             DLF,            HIAF,           I,       00039630
               2    INFIX,           IQUAD,          IS,             TIAF     00039640
000003          COMMON /RTPOUT/                                               00039650
               1    DRTP(14,200),    HRTP(14,200),   IDRTP(14,200),  TRTP(14,200),00039660
               2    VRTP(14,200)                                             00039670
000003          COMMON /TRAFOT/                                               00039680
               1    ALT(200),        EELT(200),      ELLT(200),      GWA(200),00039690
               2    GWD(100),        ITYPEA(200),    ITYPED(100),    KEY(200),00039700
               3    LFIX(200),       NACA,           NACD,           SDT(100),00039710
               4    SLT(200),        SPEED(200),     TSECAA(200),    TSECAD(100),00039720
               5    TSECSA(200),     TSECSD(100),    VBUGG(200),     VREF(200)00039730
000003          COMMON /WIND/                                                 00039740
               1    GWINE1,          GWINE2,         GWINE3,         GWINN1,  00039750
               2    GWINN2,          GWINN3,         HWIND0,         HWIND1,  00039760
               3    HWIND2,          WERRE1,         WERRE2,         WERRE3,  00039770
               4    WERRN1,          WERRN2,         WERRN3,         WINANG,  00039780
               5    WINDE,           WINDED,         WINDE1,         WINDE2,  00039790
               6    WINDN,           WINDND,         WINDN1,         WINDN2,  00039800
               7    WINMAG                                                   00039810
      C                                                                         00039820
      C     STORE ENTRY GROUND SPEED, ALTITUDE AND ID                         00039830
      C                                                                         00039840
000003          VRTP(1,IS) = SPKNOT                                           00039850
000007          HRTP(1,IS) = ALT(IS)                                          00039860
000011          IDRTP(1,IS) = 1                                               00039870
      C                                                                         00039880
      C     CONVERT THE START OF DESCENT GROUND SPEED TO TAS AND CAS          00039890
      C     STORE START OF DESCENT GROUND SPEED, ALT., AND ID.                00039900
      C                                                                         00039910
000014          CALL WETHER(ALT(IS),0)                                        00039920
000017          VATW = WINMAG*COS(ANGEF(IQUAD,INFIX) -  WINANG)               00039930
000027          VTAS = VNEW  - VATW                                           00039940
000031          CALL TASCAS (ALT(IS),VTAS,VCAS)                               00039950
000035          VRTP(2,IS) = VNEW                                             00039960
000041          HRTP(2,IS) = ALT(IS)                                          00039970
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000044          IDRTP(2,IS) = 0                                          00039980
                C                                                        00039990
                C     COMPUTE MIDPOINT ALTITUDE BETWEEN ENTRY AND IAF ALTITUDES  00040000
                C     GET WIND AND TEMP. AT HMID AND CONVERT CAS TO TAS  00040010
                C     CONVERT TO GROUND SPEED AND STORE WITH ALTITUDE AND ID  00040020
                C                                                        00040030
000047          HMID = 0.5*(ALT(IS) + HIAF)                              00040040
000052          CALL WETHER(HMID,0)                                      00040050
000054          VATW = WINMAG*COS(ANGEF(IQUAD,INFIX)-WINANG)             00040060
000064          CALL CASTAS (HMID,VCAS,VTAS)                             00040070
000066          VRTP(3,IS) = VTAS + VATW                                 00040080
000073          HRTP(3,IS) = HMID                                        00040090
000075          IDRTP(3,IS) = 0                                          00040100
                C                                                        00040110
                C     GET WIND AND TEMP AT IAF ALTITUDE                  00040120
                C     ASSURE CAS AT LEAST 250 KNOTS ON DDP SEGMENT       00040130
                C     FIND TAS AND GROUND SPEED AT BOTTOM OF DESCENT SEGMENT  00040140
                C     STORE GROUND SPEED WITH ALTITUDE AND ID.           00040150
                C                                                        00040160
000100          CALL WETHER(HIAF,0)                                      00040170
000102          VATW = WINMAG*COS(ANGEF(IQUAD,INFIX)-WINANG)             00040180
000112          IF (VCAS.LT.250.) VCAS = 250.                            00040190
000116          CALL CASTAS (HIAF,VCAS,VTAS)                             00040200
000121          VRTP(4,IS) = VTAS + VATW                                 00040210
000126          HRTP(4,IS) = HIAF                                        00040220
000130          IDRTP(4,IS) = 0                                          00040230
                C                                                        00040240
                C     GET IAF TRACK ANGLE AND WIND                       00040250
                C     DETERMINE IAF GROUND SPEED AND STORE WITH ALT. AND ID.  00040260
                C                                                        00040270
000133          CALL WETHER(HIAF,0)                                      00040280
000135          VATW = WINMAG*COS(ANGIAF(IQUAD)-WINANG)                  00040290
000143          VCAS = 250.                                              00040300
000144          CALL CASTAS (HIAF,VCAS,VTAS)                             00040310
000147          VRTP(5,IS) = VTAS + VATW                                 00040320
000154          HRTP(5,IS) = HIAF                                        00040330
000156          IDRTP(5,IS) = 2                                          00040340
                C                                                        00040350
                C     SET EF TIME AND MAJOR SEGMENT DISTANCES IN RTPOUT  00040360
                C                                                        00040370
000161          TRTP(1,IS)=TIAF                                          00040380
000164          DRTP(1,IS) = 0.                                          00040390
000167          DRTP(2,IS) = DLF                                         00040400
000171          DRTP(5,IS) = DDP                                         00040410
000174          RETURN                                                   00040420
000175          END                                                      00040430
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE TASCAS (H, VTAS, VCAS)                                  00032340
          C                                                                        00032350
          C     ENTER WITH PRESSURE ALTITUDE AND TRUE AIRSPEED,                    00032360
          C     RETURNS THE CALIBRATED AIRSPEED                                    00032370
          C                                                                        00032380
000006            COMMON /ATMO/                                                  00032390
                1    DENSR,          DENSRF,         DENSTY,         PRESR,      00032400
                2    PRESUR,         TEMR,           TEMRS                       00032410
          C                                                                        00032420
000006            CALL WETHER (H,0)                                              00032430
000007            TEMP1 = (VTAS/(DENSRF*1479.))**2                               00032440
000013            TEMP2 = (TEMP1/PRESR + 1.)**3.5                                00032450
000021            TEMP3 = (TEMP2 - 1.)*PRESR + 1.                                00032460
000024            EXPO  = 1./3.5                                                 00032470
000026            TEMP4 = (TEMP3**EXPO - 1.)*5.                                  00032480
000033            VCAS  = 661.5*SQRT(TEMP4)                                      00032490
000040            RETURN                                                         00032500
000041            END                                                            00032510
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                      FUNCTION TBLU(N,X,Y,YO)                                  00015140
                C                                                              00015150
000007                DIMENSION X(1),Y(1)                                      00015160
                C                                                              00015170
                C     N   NUMBER OF POINTS IN ARRAYS                           00015180
                C     X   DEPENDENT VARIABLE ARRAY                             00015190
                C     Y   INDEPENDENT VARIABLE ARRAY                           00015200
                C     YO  Y FOR X DESIRED                                      00015210
                C                                                              00015220
000007                IF((YO.GE.Y(1).AND.YO.LE.Y(N)).OR.(YO.GE.Y(N).AND.YO.LE.Y(1)))  00015230
                     1   GO TO 5                                               00015240
000031                   TBLU = 0                                              00015250
000032                   RETURN                                                00015260
000033          5     CONTINUE                                                 00015270
000033                I=1                                                      00015280
000034                IF(Y(N).LT.Y(1)) GO TO 1                                 00015290
000037          2        CONTINUE                                              00015300
000037                      IF(YO.GE.Y(I).AND.YO.LT.Y(I+1)) GO TO 3            00015310
000051                         I=I+1                                           00015320
000052                   GO TO 2                                               00015330
000053          1     CONTINUE                                                 00015340
000053                   IF(YO.GT.Y(I+1).AND.YO.LE.Y(I)) GO TO 3               00015350
000065                      I=I+1                                              00015360
000066                GO TO 1                                                  00015370
000067          3     CONTINUE                                                 00015380
000067                DEL=X(I+1)-X(I)                                          00015390
000072                TBLU=((YO-Y(I))/(Y(I+1)-Y(I)))*DEL + X(I)                00015400
000100                RETURN                                                   00015410
000101                END                                                      00015420
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                      SUBROUTINE TFIND (IS,SUMT)                                                    00040440
                C                                                                                   00040450
                C     THIS SUBROUTINE TRANSFORMS AN ALTITUDE - GROUND SPEED                         00040460
                C     PROFILE INTO A TRANSITION TIME  ID UPDATE COMPLETION OF RTP                   00040470
                C     FROM THE EF TO THE IAF.                                                       00040480
                C                                                                                   00040490
000005                COMMON /CNSTNT/                                                               00040500
                     1    ACCEL,          GRADI,          ITMAX,          PI,                     00040510
                     2    RTPTST                                                                   00040520
000005                COMMON /RTPOUT/                                                               00040530
                     1    DRTP(14,200),  HRTP(14,200),  IDRTP(14,200), TRTP(14,200),              00040540
                     2    VRTP(14,200)                                                             00040550
                C                                                                                   00040560
                C     INITIALIZE TIME ACCUMULATOR                                                   00040570
                C                                                                                   00040580
000005                SUMT = 0.                                                                     00040590
                C                                                                                   00040600
                C     COMPUTE SEGMENT DISTANCES                                                     00040610
                C                                                                                   00040620
000005                JRTP = 3                                                                      00040630
000006           10   CONTINUE                                                                      00040640
000006                IF (IDRTP(JRTP,IS).NE.0) GO TO 20                                             00040650
000011                   DRTP(JRTP,IS) = (HRTP(JRTP-1,IS) - HRTP(JRTP,IS))/GRADI                    00040660
000020                   JRTP =JRTP + 1                                                             00040670
000021                   IF (JRTP.GT.10) STOP 6                                                     00040680
000025                   GO TO 10                                                                   00040690
000026           20   CONTINUE                                                                      00040700
                C                                                                                   00040710
                C     COMPUTE SEGMENT VELOCITIES, TIMES AND ACCUMULATE TIME SUMS                    00040720
                C                                                                                   00040730
000026                JRTP = 1                                                                      00040740
000027           30   CONTINUE                                                                      00040750
000027                   VAVG = 0.5*(VRTP(JRTP+1,IS) + VRTP(JRTP,IS))                               00040760
000036                   TRTP(JRTP+1,IS) = 3600.*DRTP(JRTP+1,IS)/VAVG                               00040770
000042                   SUMT = SUMT + TRTP(JRTP+1,IS)                                              00040780
000044                   JRTP = JRTP + 1                                                            00040790
000045                   IF (IDRTP(JRTP,IS).EQ.0) GO TO 30                                          00040800
                C                                                                                   00040810
                C     SET TIMES AND DISTANCES FROM ENTRY FIX                                        00040820
                C                                                                                   00040830
000046                TRTP(1,IS)=TRTP(1,IS)-SUMT                                                    00040840
000050                JRTP = 2                                                                      00040850
000051           40   CONTINUE                                                                      00040860
000051                   TRTP(JRTP,IS) = TRTP(JRTP-1,IS) + TRTP(JRTP,IS)                            00040870
000057                   DRTP(JRTP,IS) = DRTP(JRTP-1,IS) + DRTP(JRTP,IS)                            00040880
000061                   IF (IDRTP(JRTP,IS).NE.0) RETURN                                            00040890
000063                   JRTP = JRTP + 1                                                            00040900
000065                   GO TO 40                                                                   00040910
000065                END                                                                           00040920
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE TIMAA(T,K1,K)                                                                00026280
                                                                                                        00026290
          C                                                                                             00026300
000006          COMMON /AERO1O/                                                                       00026310
               1    HC(4,5),        TC(4,5),        TCSUM(4),       THPRES,                         00026320
               2    VCCAS(5),       VCG(4,5),       VCTAS(4,5)                                      00026330
000006          COMMON /ATCIN/                                                                        00026340
               1    DIST(4),        DRCO(10),       DRTPS,          HRTPS,                          00026350
               2    IRWD(4),        ISCSW,          TARO(10),       TIME(4),                        00026360
               3    TRTPS                                                                           00026370
000006          COMMON /GEOMCT/                                                                       00026380
               1    ANGEF(4,3),     ANGFAF(2),      ANGIAF(4),      ANGMF(2),                       00026390
               2    ANGCM(2),       ANGTF(4),       DEF(4,3,5),     DFAF(2),                        00026400
               3    DIAF(4),        DMF(2),         DOM(2),         DPATH(4,3,5),                   00026410
               4    DTF(4),         DTH(2)                                                          00026420
000006          COMMON /TRAFIN/                                                                       00026430
               1    ALTPA1(10),     ALTPA2(10),     CALT (10,10),   CVREF(10,10),                   00026440
               2    CWALT(10,10),   CWVREF(10,10),  ENTRY(10,4,3),  HCUR,                           00026450
               3    INDALT(10),     INDSPD(10),     INDTA,          INDTD,                          00026460
               4    INDWA(10),      INDWD(10),      NCALT(10),      NCVREF(10),                     00026470
               5    NMTYPA(10),     NMTYPD(10),     NTYPEA,         NTYPED,                         00026480
               6    SPDPA1(10),     SPDPA2(10),     SPSIGM,         SWTRAF,                         00026490
               7    TAPAR1,         TAPAR2,         TASIGM,         TDPAR1,                         00026500
               8    TDPAR2,         TDSIGM,         TYPEA(10),      TYPED(10),                      00026510
               9    WAPAR1(10),     WAPAR2(10),     WDPAR1(10),     WDPAR2(10)                      00026520
000006          INTEGER             SWTRAF                                                          00026530
000006          COMMON /TRAFOT/                                                                       00026540
               1    ALT(200),       EELT(200),      ELLT(200),      GWA(200),                       00026550
               2    GWD(100),       ITYPEA(200),    ITYPED(100),    KEY(200),                       00026560
               3    LFIX(200),      NACA,           NACD,           SDT(100),                       00026570
               4    SLT(200),       SPEED(200),     TSECAA(200),    TSECAD(100),                    00026580
               5    TSECSA(200),    TSECSD(100),    VDUGG(200),     VREF(200)                       00026590
          C                                                                                             00026600
          C     DETERMINE DISTANCE CONSTRAINT TIME                                                    00026610
          C                                                                                             00026620
000006          T1=DIST(1)*3600./VDUGG(K)                                                             00026630
000011          GAT=0.                                                                                00026640
000012          IF (VDUGG(K).GE.VDUGG(K1)) GO TO 110                                                  00026650
000015             GAT1=2.*DOM(1)/(VCG(1,1)+VDUGG(K1))                                                00026660
000020             GAT2=2.*DOM(1)/(VCG(1,1)+VDUGG(K))                                                 00026670
000024             GAT=(GAT2-GAT1)*3600.0                                                             00026680
000027     110 CONTINUE                                                                               00026690
000027          T1=T1+GAT                                                                             00026700
000031          IF (DIST(1).EQ.0.0) T1=0.0                                                            00026710
          C                                                                                             00026720
          C     DETERMINE RUNWAY OCCUPANCY CONSTAINT TIME                                             00026730
          C                                                                                             00026740
000033          T3=0.                                                                                 00026750
000034          IF (IRWD(1).EQ.0) GO TO 200                                                           00026760
000035             I=0                                                                                00026770
000036     150     CONTINUE                                                                           00026780
000036                I=I+1                                                                           00026790
000040                IF (ITYPEA(K1).EQ.NMTYPA(I)) T3=TARO(I)
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000044                  IF (I.EQ.NTYPEA) GO TO 200                    00026800
000046                GO TO 150                                       00026810
000047     200 CONTINUE                                               00026820
           C                                                          00026830
           C     DETERMINE MAXIMUM INTER-OPERATION CONSTRAINT TIME    00026840
           C                                                          00026850
000047           T = AMAX1(T1,TIME(I),T3)                            00026860
000055           RETURN                                              00026870
000055           END                                                 00026880
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                                                                                                                00025870
                SUBROUTINE TIMAD(T,K,J)                                                                         00025880
      C                                                                                                         00025890
000006          COMMON /ATC1N/                                                                                  00025900
               1    DIST(4),          DRCO(10),         DRTFS             HRTFS,                              00025910
               2    IFWD(4),          ISCSW,            TARO(10),         TIME(4),                            00025920
               3    TRTFS                                                                                      00025930
000006          COMMON /TRAFIN/                                                                                 00025940
               1    ALTPA1(10),       ALTPA2(10),       CALT (10,10),     CVREF(10,10),                       00025950
               2    CWALT(10,10),     CWVREF(10,10),    ENTRY(10,4,3),    HOUR,                               00025960
               3    INDALT(10),       INDSFD(10),       INDTA,            INDTD,                              00025970
               4    INDWA(10),        INDWD(10),        NCALT(10),        NCVREF(10),                         00025980
               5    NMTYPA(10),       NMTYPD(10),       NTYPEA,           NTYPED,                             00025990
               6    SFDPA1(10),       SFDPA2(10),       SPSIGM,           SWTRAF,                             00026000
               7    TAPAR1,           TAPAR2,           TASIGM,           TDPAR1,                             00026010
               8    TDPAR2,           TDSIGM,           TYPEA(10),        TYPED(10),                          00026020
               9    WAPAR1(10),       WAPAR2(10),       WDPAR1(10),       WDPAR2(10)                          00026030
000006          INTEGER               SWTRAF                                                                    00026040
000006          COMMON /TRAFOT/                                                                                 00026050
               1    ALT(200),         EELT(200),        ELLT(200),        GWA(200),                           00026060
               2    GWD(100),         ITYPEA(200),      ITYPED(100),      KEY(200),                           00026070
               3    LFIX(200),        NACA,             NACD,             SDT(100),                           00026080
               4    SLT(200),         SPEED(200),       TSECAA(200),      TSECAD(100),                        00026090
               5    TSECSA(200),      TSECSD(100),      VBUGG(200),       VREF(200)                           00026100
      C                                                                                                         00026110
      C     DETERMINE RUNWAY OCCUPANCY TIME                                                                   00026120
      C                                                                                                         00026130
000006          T3=0.                                                                                           00026140
000007          IF (IRWD(2).EQ.0) GO TO 150                                                                     00026150
000010             I=0                                                                                          00026160
000011     110     CONTINUE                                                                                     00026170
000011                I=I+1                                                                                     00026180
000013                IF (ITYPEA(K).EQ.NMTYPA(I)) T3=TARO(I)                                                    00026190
000017                IF (I.EQ.NTYPEA) GO TO 150                                                                00026200
000021             GO TO 110                                                                                    00026210
000022     150 CONTINUE                                                                                         00026220
      C                                                                                                         00026230
      C     DETERMINE MAXIMUM INTER-OPERATION CONSTRAINT TIME                                                 00026240
      C                                                                                                         00026250
000022          T = AMAX1(TIME(2),T3)                                                                           00026260
000026          RETURN                                                                                          00026270
000026          END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE TIMDA(T,J,K)                                          00025390
          C                                                                      00025400
000006            COMMON /AERO1CV                                                00025410
                 1     HC(4,5),        TC(4,5),        TCSUM(4),       THRES,    00025420
                 2     VCCAS(5),       VCG(4,5),       VCTAS(4,5)                00025430
000006            COMMON /ATCIN/                                                 00025440
                 1     DIST(4),        DRCO(10),       DRTPS,          HRTPS,    00025450
                 2     IRWO(4),        ISCSW,          TARO(10),       TIME(4),  00025460
                 3     TRTPS                                                     00025470
000006            COMMON /TRAFIN/                                                00025480
                 1     ALTPA1(10),     ALTPA2(10),     CALT (10,10),   CVREF(10,10),  00025490
                 2     CWALT(10,10),   CWVREF(10,10),  ENTRY(10,4,3),  HCUR,     00025500
                 3     INDALT(10),     INDSPD(10),     INDTA,          INDTD,    00025510
                 4     INDWA(10),      INDWD(10),      NCALT(10),      NCVREF(10),  00025520
                 5     NMTYPA(10),     NMTYPD(10),     NTYPEA,         NTYPED,   00025530
                 6     SPDPA1(10),     SPDPA2(10),     SPSIGM,         SWTRAF,   00025540
                 7     TAPAR1,         TAPAR2,         TASIGM,         TDPAR1,   00025550
                 8     TDPAR2,         TDSIGM,         TYPEA(10),      TYPED(10),  00025560
                 9     WAPAR1(10),     WAPAR2(10),     WDPAR1(10),     WDPAR2(10)  00025570
000006             INTEGER             SWTRAF                                    00025580
000006             COMMON /TRAFOT/                                               00025590
                 1     ALT(200),       EELT(200),      ELLT(200),      GWA(200),  00025600
                 2     GWD(100),       ITYPEA(200),    ITYPED(100),    KEY(200),  00025610
                 3     LFIX(200),      NACA,           NACD,           SDT(100),  00025620
                 4     SLT(200),       SPEED(200),     TSECAA(200),    TSECAD(100),  00025630
                 5     TSECSA(200),    TSECSD(100),    VBUGG(200),     VREF(200)  00025640
          C                                                                      00025650
          C     DETERMINE DISTANCE CONSTRAINT TIME                               00025660
          C                                                                      00025670
000006            T1=3600.0*2.0*DIST(3)/(VBUGG(K)+VCG(1,1))                      00025680
          C                                                                      00025690
          C     DETERMINE RUNWAY OCCUPANCY CONSTRAINT TIME                       00025700
          C                                                                      00025710
000012            T3=0.                                                          00025720
000013            IF (IRWO(3).EQ.0) GO TO 200                                    00025730
000014               I=0.                                                        00025740
000015       100     CONTINUE                                                    00025750
000015                  I=I+1                                                    00025760
000017                  IF (ITYPED(J).EQ.NMTYPD(I)) T3=DRCO(I)                   00025770
000023                  IF (I.EQ.NTYPED) GO TO 200                               00025780
000025               GO TO 100                                                   00025790
000026       200 CONTINUE                                                        00025800
          C                                                                      00025810
          C     DETERMINE MAXIMUM INTER-OPERATION CONSTRAINT TIME                00025820
          C                                                                      00025830
000026            T = AMAX1(T1,TIME(3),T3)                                       00025840
000034            RETURN                                                         00025850
000034            END                                                            00025860
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE TIMED(T,J1,J)                                        00024880
          C                                                                     00024890
000000            COMMON /AERO1CV                                               00024900
                 1    HC(4,5),        TC(4,5),        TCSUM(4),       THRES,    00024910
                 2    VCCAS(5),       VCG(4,5),       VC1AS(4,5)                00024920
000006            COMMON /ATCIN/                                                00024930
                 1    DIST(4),        DRCO(10),       DRTFS,          HRTFS,    00024940
                 2    IRWO(4),        ISCSW,          TARO(10),       TIME(4),  00024950
                 3    TRTPS                                                     00024960
000006            COMMON /CNSTNT/                                               00024970
                 1    ACCEL,          GRAD1,          ITMAX,          PI,       00024980
                 2    RTFTST                                                    00024990
000006            COMMON /TRAFIN/                                               00025000
                 1    ALTPA1(10),     ALTPA2(10),     CALT (10,10),   CVREF(10,10),  00025010
                 2    CWALT(10,10),   CWVREF(10,10),  ENTRY(10,4,3),  HCUR,     00025020
                 3    INDALT(10),     INDSPD(10),     INDTA,          INDTD,    00025030
                 4    INDWA(10),      INDWD(10),      NCALT(10),      NCVREF(10),  00025040
                 5    NMTYPA(10),     NMTYPD(10),     NTYPEA,         NTYPED,   00025050
                 6    SPDPA1(10),     SPDPA2(10),     SPSIGM,         SWTRAF,   00025060
                 7    TAPAR1,         TAPAR2,         TASIGM,         TDPAR1,   00025070
                 8    TDPAR2,         TDSIGM,         TYPEA(10),      TYPED(10),  00025080
                 9    WAPAR1(10),     WAPAR2(10),     WDPAR1(10),     WDPAR2(10)  00025090
000006            INTEGER             SWTRAF                                    00025100
000006            COMMON /TRAFOT/                                               00025110
                 1    ALT(200),       EELT(200),      ELLT(200),      GWA(200),  00025120
                 2    GWD(100),       ITYPEA(200),    ITYPED(100),    KEY(200),  00025130
                 3    LFIX(200),      NACA,           NACD,           SDT(100),  00025140
                 4    SLT(200),       SPEED(200),     TSECAA(200),    TSECAD(100),  00025150
                 5    TSECSA(200),    TSECSD(100),    VDUGG(200),     VREF(200)  00025160
          C                                                                     00025170
          C     DETERMINE DISTANCE CONSTRAINT TIME                            00025180
          C                                                                     00025190
000006            T1=SQRT(2.*DIST(4)/ACCEL)                                     00025200
          C                                                                     00025210
          C     DETERMINE RUNWAY OCCUPANCY CONSTRAINT TIME                    00025220
          C                                                                     00025230
000013            T3=0.                                                         00025240
000014            IF (IRWO(4).EQ.0) GO TO 200                                   00025250
000017               I=0                                                        00025260
000020       100     CONTINUE                                                   00025270
000020                  I=I+1                                                   00025280
000022                  IF (ITYPED(J1).EQ.NMTYPD(I)) T3=DRCO(I)                 00025290
000026                  IF (I.EQ.NTYPED) GO TO 200                              00025300
000030               GO TO 100                                                  00025310
000031       200 CONTINUE                                                       00025320
          C                                                                     00025330
          C     DETERMINE MAXIMUM INTER-OPERATION CONSTRAINT TIME             00025340
          C                                                                     00025350
000031            T = AMAX1(T1,TIME(4),T3)                                      00025360
000037            RETURN                                                        00025370
000037            END                                                           00025380
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A        16.56.27. 73/12/21.

                      SUBROUTINE TIMFND (IS, SUMT)                                    00021960
                C                                                                     00021970
                C     THIS SUBROUTINE TRANSFORMS AN ALTITUDE - GROUND SPEED           00021980
                C     PROFILE INTO A TRANSITION TIME                                  00021990
                C     FROM THE EF TO THE IAF.                                         00022000
                C                                                                     00022010
000005                COMMON /CNSTNT/                                                 00022020
                     1    ACCEL,          GRADI,          ITMAX,          PI,        00022030
                     2    RTPTST                                                     00022040
000005                COMMON /RTPCUT/                                                 00022050
                     1    DRTP(14,200),  HRTP(14,200),  IDRTP(14,200), TRTP(14,200), 00022060
                     2    VRTP(14,200)                                               00022070
                C                                                                     00022080
                C     INITIALIZE TIME ACCUMULATOR                                     00022090
                C                                                                     00022100
000005                SUMT = 0.                                                       00022110
000005                JRTP = 3                                                        00022120
000006           10   CONTINUE                                                        00022130
000006                IF (IDRTP(JRTP,IS).NE.0) GO TO 20                               00022140
000011                   DRTP(JRTP,IS) = (HRTP(JRTP-1,IS) - HRTP(JRTP,IS))/GRADI      00022150
000020                   JRTP =JRTP + 1                                               00022160
000021                   IF (JRTP.GT.10) STOP 4                                       00022170
000025                   GO TO 10                                                     00022180
000026           20   CONTINUE                                                        00022190
                C                                                                     00022200
                C     COMPUTE SEGMENT VELOCITIES, TIMES AND ACCUMULATE TIME SUMS      00022210
                C                                                                     00022220
000026                JRTP = 1                                                        00022230
000027           30   CONTINUE                                                        00022240
000027                   VAVG = 0.5*(VRTP(JRTP+1,IS) + VRTP(JRTP,IS))                 00022250
000036                   TRTP(JRTP+1,IS) = 3600.*DRTP(JRTP+1,IS)/VAVG                 00022260
000042                   SUMT = SUMT + TRTP(JRTP+1,IS)                                00022270
000044                   JRTP = JRTP + 1                                              00022280
000045                   IF (IDRTP(JRTP,IS).EQ.0) GO TO 30                            00022290
                C                                                                     00022300
                C     SET TIMES AND DISTANCES FROM ENTRY FIX                          00022310
                C                                                                     00022320
000046                TRTP(1,IS)=0.                                                   00022330
000050                JRTP = 2                                                        00022340
000050           40   CONTINUE                                                        00022350
000050                   TRTP(JRTP,IS) = TRTP(JRTP-1,IS) + TRTP(JRTP,IS)              00022360
000056                   IF (IDRTP(JRTP,IS).NE.0) RETURN                              00022370
000057                   JRTP = JRTP + 1                                              00022380
000061                   GO TO 40                                                     00022390
000061                END                                                           00022400
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE TCMIAF(HC, VCCAS, VCTAS, VCG1, VCG, TC)                00018030
                                                                                  00018040
      C                                                                           00018050
000011          COMMON /ATMCV                                                     00018060
               1    DENSR,          DENSRF,         DENSTY,         PRESR,       00018070
               2    PRESUR,         TEMR,           TEMRS                        00018080
000011          COMMON /CMIAF/                                                    00018090
               1    ANG,            DIST                                         00018100
000011          COMMON /WIND/                                                     00018110
               1    GWINE1,         GWINE2,         GWINE3,         GWINN1,      00018120
               2    GWINN2,         GWINN3,         HWINDG,         HWIND1,      00018130
               3    HWIND2,         WERRE1,         WERRE2,         WERRE3,      00018140
               4    WERRN1,         WERRN2,         WERRN3,         WINANG,      00018150
               5    WINDE,          WINDED,         WINDE1,         WINDE2,      00018160
               6    WINDN,          WINDND,         WINDN1,         WINDN2,      00018170
               7    WINMAG                                                       00018180
      C                                                                           00018190
000011          CALL WETHER (HC, 0)                                               00018200
000012          VCTAS=VCCAS*DENSRF                                                00018210
000017          VCG=VCTAS + WINMAG*CCG(WINANG - ANG)                              00018220
000031          TC = 7200.*DIST/(VCG + VCG1)                                      00018230
000034          RETURN                                                            00018240
000035          END
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE TRAFIC(IFLITX, ISAMP)                                  00011650
      C                                                                           00011660
000005          COMMON /TRAFIN/                                                   00011670
               1    ALTPA1(10),    ALTPA2(10),    CALT (10,10),  CVREF(10,10),  00011680
               2    CWALT(10,10),  CWVREF(10,10), ENTRY(10,3,3), HOUR,          00011690
               3    INDALT(10),    INDSPD(10),    INDTA,         INDTD,         00011700
               4    INDWA(10),     INDWD(10),     NCALT(10),     NCVREF(10),    00011710
               5    NMTYPA(10),    NMTYPD(10),    NTYPEA,        NTYPED,        00011720
               6    SPDPA1(10),    SPDPA2(10),    SPSIGM,        SWTRAF,        00011730
               7    TAPAR1,        TAPAR2,        TASIGM,        TDPAR1,        00011740
               8    TDPAR2,        TDSIGM,        TYPEA(10),     TYPED(10),     00011750
               9    WAPAR1(10),    WAPAR2(10),    WDPAR1(10),    WDPAR2(10)     00011760
000005          INTEGER            SWTRAF                                         00011770
000005          COMMON /TRAFOT/                                                   00011780
               1    ALT(200),      EELT(200),     ELLT(200),     GWA(200),      00011790
               2    GWD(100),      ITYPEA(200),   ITYPED(100),   KEY(200),      00011800
               3    LFIX(200),     NACA,          NACD,          SDT(100),      00011810
               4    SLT(200),      SPEED(200),    TSECAA(200),   TSECAD(100),   00011820
               5    TSECSA(200),   TSECSD(100),   VDUGG(200),    VREF(200)      00011830
000005          DIMENSION          CAT(10),       CVR(10),       CWA(10),       00011840
               1    CWV(10),       FERA(10),      FERD(10),      WTLA(10),      00011850
               2    WTLD(10),      WTUA(10),      WTUD(10)                      00011860
000005          DATA               DELTA          /20./                          00011870
      C                                                                           00011880
000005          IF(SWTRAF.EQ.6HDISTRB)GO TO 3000                                  00011890
      C                                                                           00011900
      C     DISCRETE TRAFFIC LIST SUPPLIED AS INPUT                             00011910
      C                                                                           00011920
000007          DO 31 I = 1, NACA                                                 00011930
000010             CALL NORMAL (TSECAA(I), TSECSA(I), TASIGM)                     00011940
000013             CALL NORMAL (SPEED(I), SPEED(I), SPSIGM)                       00011950
000017       31 CONTINUE                                                        00011960
000023          CALL SORT(TSECAA, NACA)                                           00011970
000025          DO 32 I = 1, NACD                                                 00011980
000030             CALL NORMAL (TSECAD(I), TSECSD(I), TDSIGM)                     00011990
000033       32 CONTINUE                                                        00012000
000037          CALL SORT (TSECAD,NACD)                                           00012010
000041          DO 6600 I=1,NACA                                                  00012020
000044             DO 33 J=1,NTYPEA                                               00012030
000045                IF (NMTYPA(J).NE.ITYPEA(I)) GO TO 33                        00012040
000050                GO TO 35                                                    00012050
000050       33    CONTINUE                                                     00012060
000053             STOP 1                                                        00012070
000055       35    CONTINUE                                                     00012080
000055             NV=NCVREF(J)                                                  00012090
000057             DO 7700 II=1,NV                                               00012100
000061                CWV(II)=CWVREF(J,II)                                       00012110
000065                CVR(II)=CVREF(J,II)                                        00012120
000070    7700    CONTINUE                                                     00012130
000072             VREF(I) = TDLU(NV, CVR, CWV, GWA(I))                          00012140
000100             IF(VREF(I).LT.CVREF(J,1))VREF(I)=CVREF(J,1)                   00012150
000106             IF(VREF(I).GT.CVREF(J,NV))VREF(I)=CVREF(J,NV)                 00012160
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000116    0000 CONTINUE                                                    00012170
000121         IF (MOD(ISAMP, IFLITX).NE.0) RETURN                         00012180
000125         CALL TRAFWT (ISAMP)                                         00012190
000127         RETURN                                                      00012200
               C                                                           00012210
               C     DISTRIBUTION TRAFFIC LIST TO BE GENERATED FROM DISTRIBUTIONS INPUT00012220
               C     TO PROGRAM                                            00012230
               C                                                           00012240
000130    3000 CONTINUE                                                    00012250
000130         I=0                                                         00012260
000131         TT=3600.*HOUR                                               00012270
000133         NT=TT/TAPAR1+50                                             00012280
               C                                                           00012290
               C     GENERATE ARRIVAL SCHEDULE                             00012300
               C                                                           00012310
000137    515  CONTINUE                                                    00012320
000137           AT=0.                                                     00012330
000140           DO 510 K=1,NT                                             00012340
000143              TMEANA = TAPAR1*(FLOAT(K) - 1.)                        00012350
000147              IF(INDTA.EQ.6HNORMAL)CALL NORMAL ( AT,TMEANA,TAPAR2)   00012360
000155              IF(INDTA.EQ.6HERLANG)CALL ERLANG ( AT,TMEANA,TAPAR2)   00012370
000164              IF(AT.GT.TT)GO TO 520                                  00012380
000170                 IF(AT.EQ.TT)GO TO 525                               00012390
000171                    I=I+1                                            00012400
000172                    TSECAA(I)=AT                                     00012410
000174    510        CONTINUE                                              00012420
000176           IF(AT.EQ.TT)GO TO 525                                     00012430
000200              IF(AT.GT.TT)GO TO 520                                  00012440
000203                 NT=NT+50                                            00012450
000204         GO TO 515                                                   00012460
               C                                                           00012470
000204    525  CONTINUE                                                    00012480
000204         I=I+1                                                       00012490
000206         TSECAA(I)=AT                                                00012500
000210    520  CONTINUE                                                    00012510
000210         NACA=I                                                      00012520
               C                                                           00012530
               C     SORT ARRIVAL TIMES                                    00012540
               C                                                           00012550
000212         CALL SORT(TSECAA, NACA)                                     00012560
               C                                                           00012570
               C     GENERATE DEPARTURE SCHEDULE                           00012580
               C                                                           00012590
000214         NT=TT/TDPAR1+50                                             00012600
000220    815  CONTINUE                                                    00012610
000220           AT=0.                                                     00012620
000221           I=0                                                       00012630
000222           DO 810 K=1,NT                                             00012640
000225              TMEAND = TDPAR1*(FLOAT(K) - 1.)                        00012650
000231              IF(INDTD.EQ.6HNORMAL)CALL NORMAL ( A,TMEAND,TDPAR2)    00012660
000237              IF(INDTD.EQ.6HERLANG)CALL ERLANG ( A,TMEAND,TDPAR2)    00012670
000246              AT = A + DELTA                                         00012680
000250              IF(AT.GT.TT)GO TO 820                                  00012690
```

TABLE 5-1.—CONTINUED

```
FUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000254                    IF(AT.EQ.TT)GO TO 825                          00012700
000255                       I=I+1                                       00012710
000256                       TSECAD(I)=AT                                00012720
000260   810        CONTINUE                                            00012730
000262              IF(AT.EQ.TT)GO TO 825                               00012740
000263                 IF(AT.GT.TT)GO TO 820                            00012750
000266                    NT=NT+50                                      00012760
000267           GO TO 815                                             00012770
         C                                                             00012780
000267   825    CONTINUE                                              00012790
000267          I=I+1                                                 00012800
000271          TSECAD(I)=AT                                          00012810
000273   820    CONTINUE                                              00012820
000273          NACD=I                                                00012830
         C                                                             00012840
         C      SORT DEPARTURE TIMES                                   00012850
         C                                                             00012860
000275          CALL SORT(TSECAD, NACD)                               00012870
         C                                                             00012880
         C      FIND THE CUMULATED PERCENTAGES FOR ARRIVAL A/C TYPES   00012890
         C                                                             00012900
000277          P=0.                                                  00012910
000300          DO 1700 K=1,NTYPEA                                    00012920
000303             PERA(K)=P+ TYPEA(K)/100.                           00012930
000307             P=PERA(K)                                          00012940
000310    1700 CONTINUE                                               00012950
         C                                                             00012960
         C      GENERATE ARRIVAL A/C TYPE ACCORDING TO TYPES DISTRIBUTION 00012970
         C                                                             00012980
000312          DO 900 I=1,NACA                                       00012990
000313             CALL UNIFORM ( A,0.,1.)                            00013000
000315             DO 100 J=1,NTYPEA                                  00013010
000320                IF(A.GT.PERA(J))GO TO 100                       00013020
000324                   ITYPEA(I)=NMTYPA(J)                          00013030
000326                   GO TO 110                                    00013040
000326     100     CONTINUE                                           00013050
000331     110     CONTINUE                                           00013060
         C                                                             00013070
         C       GENERATE A/C GWA FOR THE ARRIVAL A/C TYPE           00013080
         C                                                             00013090
000331             CALL LIMITS(INDWA(J),WAPAR1(J),WAPAR2(J),GWA(I),WAPAR2(J)) 00013100
         C                                                             00013110
         C       GENERATE VREF AND ALTREF FOR ARRIVAL A/C TYPE       00013120
         C                                                             00013130
000340             NV=NCVREF(J)                                       00013140
000342             NA=NCALT(J)                                        00013150
000344             DO 140 II=1, 10                                    00013160
000347                CWV(II)=CWVREF(J, II)                           00013170
000353                CVR(II)=CVREF(J, II)                            00013180
000356                CWA(II)=CWALT(J, II)                            00013190
000361                CAT(II)=CALT(J, II)                             00013200
000364     140     CONTINUE                                           00013210
000365             ALTREF = TBLU (NA,CAT,CWA,GWA(I))                  00013220
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000372            VREF(I) = TBLU (NV,CWV,CVR,GWA(I))                          00013230
000400            IF(VREF(I).LT.CVREF(J,1))VREF(I)=CVREF(J,1)                 00013240
000406            IF(VREF(I).GT.CVREF(J,NV))VREF(I)=CVREF(J,NV)               00013250
000416            IF(ALTREF.GT.ALTPA1(J).AND.ALTREF.LT.ALTPA2(J))GO TO 160    00013260
000430            ALTREF=ALTPA2(J)                                            00013270
000431      160   CONTINUE                                                    00013280
          C                                                                   00013290
          C         GENERATE SPEED OF THE ARRIVAL A/C TYPE ACCORDING TO SPEED DIST.00013300
          C                                                                   00013310
000431              IF(INDSPD(J).EQ.6HUNIFORM)CALL UNIFORM(SPEED(I),SPDPA1(J),  00013320
         1              SPDPA2(J))                                           00013330
000442              IF(INDSPD(J).EQ.6HNORMAL) CALL NORMAL(SPEED(I),SPDPA1(J),   00013340
         1              SPDPA2(J))                                           00013350
          C                                                                   00013360
          C         GENERATE ALTITUDE OF THE ARRIVAL A/C TYPE USING ALTREF AS LIMIT00013370
          C                                                                   00013380
000453              CALL LIMITS(INDALT(J),ALTPA1(J),ALTPA2(J),ALT(I),ALTREF)    00013390
          C                                                                   00013400
          C         GENERATE ENTRANCE FIX ACCORDING TO ITS PERCENTAGE DISTRIBUTION 00013410
          C                                                                   00013420
000461              CALL UNIFORM ( A,0.,1.)                                     00013430
000464              CTYEF=0.                                                    00013440
000465              DO 170 K=1,4                                                00013450
000470                 DO 170 L=1,3                                             00013460
000471                    CTYEF=CTYEF+  ENTRY(J,K,L)                            00013470
000477                    IF(A.LE.CTYEF)GO TO 180                               00013480
000501      170   CONTINUE                                                    00013490
000505      180   CONTINUE                                                    00013500
000505              LFIX(I)=K*10+L                                              00013510
000511      900 CONTINUE                                                      00013520
          C                                                                   00013530
          C     FIND CUMMULATED PERCENTAGE FOR DEPARTURE A/C TYPES            00013540
          C                                                                   00013550
000513        P=0.                                                            00013560
000514        DO 1710 K=1,NTYPED                                              00013570
000515           PERD(K)=P+ TYPED(K)/100.                                     00013580
000521           P=PERD(K)                                                    00013590
000522     1710 CONTINUE                                                      00013600
          C                                                                   00013610
          C     GENERATE DEPARTURE A/C TYPE ACCORDING TO TYPES DISTRIBUTION   00013620
          C                                                                   00013630
000524        DO 400 I=1,NACD                                                 00013640
000525           CALL UNIFORM(A,0.,1.)                                        00013650
000527           DO 410 J=1,NTYPED                                            00013660
000532              IF(A.GT.PERD(J))GO TO 410                                 00013670
000536                 ITYPED(I)=NMTYPD(J)                                    00013680
000540                 GO TO 411                                              00013690
000540      410   CONTINUE                                                    00013700
000543      411   CONTINUE                                                    00013710
          C                                                                   00013720
          C         GENERATE DEPARTURE A/C GWA ACCORDING TO GWA DISTRIBUTION  00013730
          C                                                                   00013740
000543              CALL LIMITS(INDWD(J),WDPAR1(J),WDPAR2(J),GWD(I),WDPAR2(J))  00013750
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000552     400 CONTINUE                                                  00013760
000556         DO 1000 I=1,NACA                                          00013770
000560            TSECSA(I)=TAPAR1*I                                     00013780
000562    1000 CONTINUE                                                 00013790
000565         DO 2000 I=1,NACB                                          00013800
000566            TSECSB(I)=TBPAR1*I+DELTA                               00013810
000571    2000 CONTINUE                                                 00013820
000574         IF (MOD(ISAMP, IFLITX) .NE. 0)  RETURN                    00013830
000600         CALL TRAFWT (ISAMP)                                       00013840
000602         RETURN                                                    00013850
000603         END                                                       00013860
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE TRAFWT (ISAMP)                                                    00015690
      C                                                                                      00015700
000003          COMMON /TRAFOT/                                                              00015710
               1    ALT(200),      EELT(200),     ELLT(200),     GWA(200),                 00015720
               2    GWD(100),      ITYPEA(200),   ITYPED(100),   KEY(200),                 00015730
               3    LFIX(200),     NACA,          NACD,          SDT(100),                 00015740
               4    SLT(200),      SPEED(200),    TSECAA(200),   TSECAD(100),              00015750
               5    TSECSA(200),   TSECSD(100),   VBUGG(200),    VREF(200)                 00015760
      C                                                                                      00015770
000003          PRINT 2010, ISAMP                                                            00015780
000011          PRINT 11                                                                     00015790
000015          DO 100 I=1,NACA                                                              00015800
000020             PRINT 20,I,TSECSA(I),TSECAA(I),ITYPEA(I),GWA(I),LFIX(I),ALT(I),           00015810
               1          SPEED(I),VREF(I)                                                  00015820
000045             IF (I/35*35.NE.I) GO TO 100                                               00015830
000054                PRINT 12                                                               00015840
000057                PRINT 11                                                               00015850
000063    100   CONTINUE                                                                     00015860
000067          PRINT 30                                                                     00015870
000072          DO 200 I=1,NACD                                                              00015880
000075             IF(I/36*36 .EQ.I)PRINT 30                                                 00015890
000106             PRINT 25,I,TSECSD(I),TSECAD(I),ITYPED(I),GWD(I)                           00015900
000124    200   CONTINUE                                                                     00015910
000130          RETURN                                                                       00015920
      C                                                                                      00015930
000130    20    FORMAT(I8,2F10.2,3X,A6,F10.2,4X,I2,2F10.2, 2X, F10.2)                        00015940
000130    25    FORMAT(I8,4X,2(F10.2,4X),A6,F10.2)                                           00015950
000130    2010  FORMAT(1H1, 25X, 34H*****TRAFFIC LIST OF SAMPLE NUMBER,I4,5H*****)          00015960
000130    12    FORMAT(1H1)                                                                  00015970
000130    11    FORMAT(1HO,35X, 12HARRIVAL LIST /1HO, 5X, 2HEF, 3X, 7HNOMINAL,              00015980
               1       4X, 5HENTRY, 4X, 4HTYPE, 6X, 6HWEIGHT, 1X, 7HARRIVAL, 2X,           00015990
               2       8HALTITUDE, 1X, 9HENTRY FIX, 1X, 11HANTICIPATED / 4X,               00016000
               3       7HARRIVAL, 1X, 5HENTRY , 5X, 4HTIME, 15X, 5HX1000,                  00016010
               4       3X, 5HENTRY, 4X, 5HX1000, 5X, 5HSPEED, 4X,                          00016020
               5       10HREF. SPEED /5X, 5HORDER, 1X, 5H SECS , 5X, 5H SECS ,             00016030
               6       16X,3HLBS,5X,3HFIX,6X,4HFEET,6X,4HMACH,6X,5HKNOTS /)                00016040
000130    30    FORMAT(1H1/25X, 14HDEPARTURE LIST //3X, 9HDEPARTURE, 3X,7HNOMINAL,          00016050
               1       6X, 9HDEPARTURE,3X,4HTYPE, 6X, 6HWEIGHT /5X,5HORDER, 4X,            00016060
               2       9HDEPARTURE, 7X, 5HREADY,13X,1GH(LBSX1000)/16X, 4HTIME,              00016070
               3       10X, 4HTIME/15X,6H(SECS), 8X, 6H(SECS)/)                             00016080
000130          END                                                                          00016090
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE TRKGCM                                                00009960
                                                                                 00009970
          C                                                                      00009980
000002            COMMON /CNSTNT/                                                00009990
                 1    ACCEL,          GRAD1,          ITMAX,          PI,       00010000
                 2    RTPTST                                                     00010010
000002            COMMON /GECMTX/                                                00010020
                 1    DOFF(10),       DOFS(10),       DSEG(6),        NOFF,     00010030
                 2    RHO(7),         THETA(6),       X(7),           Y(7)      00010040
000002            REAL                MU1,            MU2,            NUM1,     00010050
                 1    NUM2                                                       00010060
          C                                                                      00010070
000002            INDT = 0                                                       00010080
000003            NCT = 6                                                        00010090
000004            DO 1000 N = 1,NCT                                              00010100
000006               NA = N                                                      00010110
000007               XB = X(N + 1)                                               00010120
000010               YB = Y(N + 1)                                               00010130
000012               RB = RHO(N + 1)                                             00010140
000013               IF (INDT.NE.0) GO TO 10                                     00010150
000015                  XB1 = X(N)                                               00010160
000016                  YB1 = Y(N)                                               00010170
000020         10    CONTINUE                                                    00010180
000020               XA = XB1                                                    00010190
000021               YA = YB1                                                    00010200
000023               XBA = XA - XB                                               00010210
000025               YBA = YA - YB                                               00010220
000027               DAB = SQRT (XBA*XBA + YBA*YBA)                              00010230
000033               IF(INDT.EQ.0) DARC = 0.0                                    00010240
000036               ANG = 0.5*PI                                                00010250
000040               IF (YA.NE.YB) ANG = ACOS(YBA/DAB)                           00010260
000047               THETA(N) = ANG                                              00010270
000051               IF (XA .LE. XB) THETA(N) = 2.0*PI - ANG                     00010280
000057               DSEG(N) = DAB + DARC                                        00010290
000062               IF(N.EQ.NCT) GO TO 1200                                     00010300
000064                  XC = X(N + 2)                                            00010310
000065                  YC = Y(N + 2)                                            00010320
000066                  XCA = XA - XC                                            00010330
000070                  YCA = YA - YC                                            00010340
000072                  DAC = SQRT(XCA*XCA + YCA*YCA)                            00010350
000077                  INDT = 0                                                 00010360
000100                  IF ((YBA/DAB).NE.(YCA/DAC)) INDT = 1                     00010370
000106                  IF (INDT.EQ.0) GO TO 1000                                00010380
000107                     XR1 = XB - RB*YBA/DAB                                 00010390
000112                     XR2 = XB + RB*YBA/DAB                                 00010400
000114                     YR1 = YB + RB*XBA/DAB                                 00010410
000117                     YR2 = YB - RB*XBA/DAB                                 00010420
000121                     DRSQ1 = (XC - XR1)*(XC-XR1) + (YC-YR1)*(YC-YR1)       00010430
000125                     DRSQ2 = (XC - XR2)*(XC-XR2) + (YC-YR2)*(YC-YR2)       00010440
000132                     IF (DRSQ1.GT. DRSQ2) GO TO  100                       00010450
000136                        XR = XR1                                           00010460
000136                        YR = YR1                                           00010470
000140                        GO TO 110                                          00010480
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CC1 73 A          16.56.27. 73/12/21.

000141      100          CONTINUE                                                  00010480
000141                   XR = XR2                                                  00010490
000142                   YR = YR2                                                  00010500
000144      110          CONTINUE                                                  00010510
000144                   DETAX = XR - XC                                           00010520
000146                   DETAY = YR - YC                                           00010530
000150                   IF (DETAY.EQ. 0.) GO TO 120                               00010540
000151                      IF (DETAX.EQ. 0.) GO TO 130                            00010550
000152                      GO TO 135                                              00010560
000153      120          CONTINUE                                                  00010570
000153                   ALPHX1 = - RD*RD/DETAX                                    00010580
000155                   ALPHX2 = ALPHX1                                           00010590
000157                   ALPHY1 = SQRT (RD*RD - ALPHX1*ALPHX1)                     00010600
000162                   ALPHY2 = - ALPHY1                                         00010610
000164                   GO TO 140                                                 00010620
000164      130          CONTINUE                                                  00010630
000164                   ALPHY1 = - RD*RD/DETAY                                    00010640
000166                   ALPHY2 = ALPHY1                                           00010650
000170                   ALPHX1 = SQRT (RD*RD - ALPHX1*ALPHX1)                     00010660
000174                   ALPHX2 = - ALPHX1                                         00010670
000176                   GO TO 140                                                 00010680
000176      135          CONTINUE                                                  00010690
000176                   A = DETAX*DETAX + DETAY*DETAY                             00010700
000200                   B = 2.0*RD*RD*DETAX                                       00010710
000203                   C = (RD*RD)*(RD*RD - DETAY*DETAY)                         00010720
000205                   RADX = SQRT (D**B - 4.0*A*C)                              00010730
000213                   ALPHX1 = ( - B + RADX)/(2.0*A)                            00010740
000217                   ALPHX2 = - (B + RADX)/(2.0*A)                             00010750
000222                   ALPHY1 = - (ALPHX1*DETAX + RD*RD)/DETAY                   00010760
000227                   ALPHY2 = - (ALPHX2*DETAX + RD*RD)/DETAY                   00010770
000232      140          CONTINUE                                                  00010780
000232                   XBI1 = ALPHX1 + XR                                        00010790
000234                   XBI2 = ALPHX2 + XR                                        00010800
000235                   YBI1 = ALPHY1 + YR                                        00010810
000237                   YBI2 = ALPHY2 + YR                                        00010820
000241                   NUM1 = (XB - XA)*(XBI1- XB) + (YB - YA)*(YBI1 - YB)       00010830
000253                   NUM2 = (XB - XA)*(XBI2 -XB) + (YB - YA)*(YBI2 - YB)       00010840
000265                   DNM1 = DAD*SQRT((XBI1-XB)*(XBI1-XB)+(YBI1-YB)*(YBI1-YB))  00010850
000275                   DNM2 = DAD*SQRT((XBI2-XB)*(XBI2-XB)+(YBI2-YB)*(YBI2-YB))  00010860
000304                   MU1 = ACOS(NUM1/DNM1)                                     00010870
000310                   MU2 = ACOS(NUM2/DNM2)                                     00010880
000314                   IF(MU1.GT.MU2) GO TO 150                                  00010890
000320                      XBI = XBI1                                             00010900
000320                      YBI = YBI1                                             00010910
000322                      GO TO 160                                              00010920
000323      150          CONTINUE                                                  00010930
000323                   XBI = XBI2                                                00010940
000324                   YBI = YBI2                                                00010950
000326      160          CONTINUE                                                  00010960
000326                   ARGC = ((XB - XR)*(XBI- XR) + (YB- YR)*(YBI -YR))/(RD*RD) 00010970
000341                   DARC = RD*ACOS(ARGC)                                      00010980
000345      1000 CONTINUE                                                          00010990
000350      1200 CONTINUE                                                          00011000
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000350          DO 900 J = 1,NOFF                                                    00011010
000352             DLA = SQRT(DCFF(J)*(4.D*RHO(NA) - DCFF(J)))                       00011020
000360             DLB = SQRT(DCFF(J)*(4.D*RB - DCFF(J)))                            00011030
000367             GAMA = ACOS(1.D - 0.5*DCFF(J)/RHO(NA))                            00011040
000375             GAMB = ACOS(1.D - 0.5*DCFF(J)/RB)                                 00011050
000406             DCFS(J) = DAB - (DLA+DLB) + 2.D*(RHO(NA)*GAMA+RB*GAMB) + DARC     00011060
000421      900 CONTINUE                                                             00011070
000423          RETURN                                                               00011080
000424          END                                                                  00011090
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                      SUBROUTINE UNIFORM (A, P1, P2)                    00014240
                C                                                       00014250
                C     P1=LOWER LIMIT, P2=UPPER LIMIT                    00014260
                C                                                       00014270
000006                COMMON /RANDOM/                                   00014280
                     1    IU,            IX                           00014290
                C                                                       00014300
                C     ********************************                  00014310
                C     ********************************                  00014320
000006                CALL RAND(IU,IX,0,1,U)                            00014330
                C     ********************************                  00014340
                C     ********************************                  00014350
000011                A = U * (P2 - P1) + P1                            00014360
000016                RETURN                                            00014370
000017                END                                               00014380
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE WETHER(H, ISW)                                       00016100
          C                                                                     00016110
000005            COMMON /CNSTNT/                                               00016120
                 1    ACCEL,          GRAD1,          ITMAX,          PI,      00016130
                 2    RTD:ST                                                   00016140
000005            COMMON /TEMPER/                                               00016150
                 1    GTEMP1,         GTEMP2,         GTEMP3,         HTEMP0,  00016160
                 2    HTEMP1,         HTEMP2,         TEMP,           TEMP1,   00016170
                 3    TEMP2,          TGROND,         TERCR1,         TERCR2,  00016180
                 4    TERCR3,         TSTAND                                   00016190
000005            COMMON /WIND/                                                 00016200
                 1    GWINE1,         GWINE2,         GWINE3,         GWINN1,  00016210
                 2    GWINN2,         GWINN3,         HWIND0,         HWIND1,  00016220
                 3    HWIND2,         WERRE1,         WERRE2,         WERRE3,  00016230
                 4    WERRN1,         WERRN2,         WERRN3,         WINANG,  00016240
                 5    WINDE,          WINDE0,         WINDE1,         WINDE2,  00016250
                 6    WINDN,          WINDN0,         WINDN1,         WINDN2,  00016260
                 7    WINMAG                                                   00016270
          C                                                                     00016280
          C     TEMPERATURE MODEL                                             00016290
          C                                                                     00016300
000005            TEST = 0.                                                     00016310
000005            IF (ISW.NE.0) TEST = 1.                                       00016320
000010            TSTAND=288.16-1.98*H                                          00016330
000013            IF (H.GT.36.089) TSTAND=216.66                                00016340
000017            IF (H.GT.HTEMP1)  GO TO 100                                   00016350
000023               TEMP= GTEMP1*(H-HTEMP0)+TGROND + TEST*TERCR1               00016360
000030               GO TO 300                                                  00016370
000031       100 CONTINUE                                                       00016380
000031            IF (H.GT.HTEMP2)  GO TO 200                                   00016390
000035               TEMP= TEMP1+GTEMP2*(H-HTEMP1) + TEST*TERCR2                00016400
000043               GO TO 300                                                  00016410
000043       200 CONTINUE                                                       00016420
000043            TEMP= TEMP2+GTEMP3*(H-HTEMP2) + TEST*TERCR3                   00016430
000052       300 CONTINUE                                                       00016440
          C                                                                     00016450
          C     WIND MODEL                                                    00016460
          C                                                                     00016470
000052            IF (H.GT.HWIND1)  GO TO 400                                   00016480
000056               WINDN =WINDN0+GWINN1*(H-HWIND0) + TEST*WERRN1              00016490
000063               WINDE =WINDE0+GWINE1*(H-HWIND0) + TEST*WERRE1              00016500
000072               GO TO 600                                                  00016510
000072       400 CONTINUE                                                       00016520
000072            IF (H.GT.HWIND2)  GO TO 500                                   00016530
000076               WINDN =WINDN1+GWINN2*(H-HWIND1) + TEST*WERRN2              00016540
000103               WINDE =WINDE1+GWINE2*(H-HWIND1) + TEST*WERRE2              00016550
000112               GO TO 600                                                  00016560
000112       500 CONTINUE                                                       00016570
000112            WINDN =WINDN2+GWINN3*(H-HWIND2) + TEST*WERRN3                 00016580
000120            WINDE =WINDE2+GWINE3*(H-HWIND2) + TEST*WERRE3                 00016590
000127       600 CONTINUE                                                       00016600
000127            IF (WINDN.EQ.0) GO TO 650                                     00016610
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000130              WINANG=ATAN2(WINDE, WINDN)                    00016620
000133              GO TO 690                                     00016630
000135   650    CONTINUE                                          00016640
000135              IF (WINDE) 660, 670, 680                      00016650
000137   660    CONTINUE                                          00016660
000137              WINANG = -0.5*PI                              00016670
000141              GO TO 690                                     00016680
000142   670    CONTINUE                                          00016690
000142              WINANG=0.                                     00016700
000143              GO TO 690                                     00016710
000144   680    CONTINUE                                          00016720
000144              WINANG = 0.5*PI                               00016730
000146   690    CONTINUE                                          00016740
000146              WINMAG=SQRT(WINDE*WINDE+WINDN*WINDN)          00016750
000153              CALL ATMOSP (H, TEMP, TSTAND)                 00016760
000156              RETURN                                        00016770
000157              END                                           00016780
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE WTHTST                                                00008640
      C                                                                          00008650
000002          COMMON /TEMPER/                                                  00008660
               1    GTEMP1,         GTEMP2,         GTEMP3,         HTEMP0,     00008670
               2    HTEMP1,         HTEMP2,         TEMP,           TEMP1,      00008680
               3    TEMP2,          TGROND,         TERGR1,         TERGR2,     00008690
               4    TERGR3,         TSTAND                                      00008700
000002          COMMON /WIND/                                                    00008710
               1    GWINE1,         GWINE2,         GWINE3,         GWINN1,     00008720
               2    GWINN2,         GWINN3,         HWIND0,         HWIND1,     00008730
               3    HWIND2,         WERRE1,         WERRE2,         WERRE3,     00008740
               4    WERRN1,         WERRN2,         WERRN3,         WINANG,     00008750
               5    WINDE,          WINDE0,         WINDE1,         WINDE2,     00008760
               6    WINDN,          WINDN0,         WINDN1,         WINDN2,     00008770
               7    WINMAG                                                      00008780
      C                                                                          00008790
000002          PRINT 9003                                                       00008800
000006          PRINT 9000                                                       00008810
000012          DO 10 IWETH=1, 101                                               00008820
000014             H=IWETH*0.5                                                   00008830
000016             CALL WETHER(H,0)                                              00008840
000020             TFCCAS=TEMP                                                   00008850
000021             WINDNF=WINDN                                                  00008860
000023             WINDEF=WINDE                                                  00008870
000024             WSPED=SQRT(WINDNF*WINDNF+WINDEF*WINDEF)                       00008880
000032             CALL WETHER(H,1)                                              00008890
000034             IF (IWETH/36*36.NE.IWETH) GO TO 5                             00008900
000041                PRINT 9002                                                 00008910
000045                PRINT 9000                                                 00008920
000051    5        CONTINUE                                                      00008930
000051             PRINT 9001, H, TFCCAS, TEMP,   WSPED, WINDNF, WINDEF, WINDN,  00008940
               1              WINDE                                             00008950
000075    10    CONTINUE                                                       00008960
000077          RETURN                                                         00008970
      C                                                                        00008980
000100    9000  FORMAT(1HD,3X, 8HALTITUDE , 3X, 8HFORECAST , 7X, 6HACTUAL , 5X, 00008990
               1       8HFORECAST , 5X, 8HFORECAST , 5X, 8HFORECAST , 2(7X,    00009000
               2       6HACTUAL )/                                             00009010
               3       14X, 2(11HTEMPERATURE , 2X), 10HWIND SPEED , 3X, 2(     00009020
               4       11HWIND N-COMP , 2X, 11HWIND E-COMP , 2X)/              00009030
               5       4X, 8H(1000FT) ,  4X, 7H(DEG-K) , 6X, 7H(DEG-K) , 6X,   00009040
               6       5(7H(KNOTS) , 6X)/1HD)                                  00009050
000100     9001 FORMAT(5X,F6.2, 5X, 2(F6.1, 7X), 5(F6.1,7X))                    00009060
000100    9002  FORMAT (1H1)                                                    00009070
000100    9003  FORMAT(1H1/42X, 28H*****WEATHER MODEL TEST***** )               00009080
000100          END                                                            00009090
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

                SUBROUTINE WTOUT                                                    00052480
      C                                                                             00052490
000002          COMMON /STATIN/                                                     00052500
               1    DCRIT,          DLYDEL,         DLYINT,         DMIN(5),      00052510
               2    DSDEL,          DSINT,          NDLY,           NDSEP,        00052520
               3    NOPSTP,         NTARR,          NTSEP,          OPDEL,        00052530
               4    OPINT,          TARDEL,         TARINT,         TMIN(5),      00052540
               5    TSDEL,          TSINT                                         00052550
000002          COMMON /STATOT/                                                     00052560
               1    ARRATE,         DELYOA,         DELYOP,         DELYOV,       00052570
               2    DELYSA,         DELYSP,         DELYSV,         DELYTA,       00052580
               3    DELYTP,         DELYTV,         DERATE,         DLYOA(17),    00052590
               4    DLYOD(17),      DLYSA(17),      DLYSD(17),      DLYSTP(17),   00052600
               5    DLYTA(17),      DLYTD(17),      DSEP(17),       OPCRAT(16),   00052610
               6    OPFRAT(36),     OPSTEP(17),     OPTIM1(17),     OPTIM2(17),   00052620
               7    OPTIM3(17),     OPTIM4(17),     PASTIM(36,17),  PDMIN(16),    00052630
               8    PDMIN1,         PDMIN2,         PDMIN3,         PDMIN4,       00052640
               9    PTMIN(16),      PTMIN1,         PTMIN2,         PTMIN3,       00052650
               A    PTMIN4,         SEPDIS(16,17),  SEPTIM(16,17),  TARR(17),     00052660
               B    TOTAL(36),      TSEP(17)                                      00052670
000002          INTEGER             DLYSTP,         OPSTEP,         TARR          00052680
      C                                                                             00052690
000002          INTEGER             ICFSET(3)                                     00052700
000002          DATA                ICFSET          /1HA, 1HD, 1HC/               00052710
      C                                                                             00052720
000002          PRINT 2010, ARRATE, DERATE                                        00052730
000012          PRINT  2020, (OPCRAT(I), I=1,16)                                  00052740
000020          PRINT  2030, (OPFRAT(I), I=1, 36)                                 00052750
000026          PRINT  2040, PTMIN1, PTMIN2, PTMIN3, PTMIN4, PDMIN1, PDMIN2,      00052760
               1                 PDMIN3, PDMIN4                                   00052770
000052          PRINT  2050, (PTMIN(I), I=1, 8), (PDMIN(I), I=1, 8),              00052780
               1                (PTMIN(I),I=9,16), (PDMIN(I), I=9,16)            00052790
000066          PRINT 2080, DELYSA, DELYSV, DELYSP, DELYOA, DELYOV, DELYOP,       00052800
               1           DELYTA, DELYTV, DELYTP                                00052810
000114          NOPSTP=NOPSTP-1                                                   00052820
000116          PRINT 2090, (OPSTEP(I), I=1, NOPSTP)                              00052830
000131          PRINT 2110, OPSTEP(NOPSTP)                                        00052840
000137          PRINT 2120, (OPTIM1(I), I=1, NOPSTP)                              00052850
000152          PRINT 2130, OPTIM1(NOPSTP)                                        00052860
000160          PRINT 2140, (OPTIM2(I), I=1, NOPSTP)                              00052870
000173          PRINT 2130, OPTIM2(NOPSTP)                                        00052880
000201          PRINT 2150, (OPTIM3(I), I=1, NOPSTP)                              00052890
000214          PRINT 2130, OPTIM3(NOPSTP)                                        00052900
000222          PRINT 2160, (OPTIM4(I), I=1, NOPSTP)                              00052910
000235          PRINT 2130, OPTIM4(NOPSTP)                                        00052920
000243          NTSEP=NTSEP-1                                                     00052930
000245          PRINT 2170, (TSEP(I), I=1, NTSEP)                                 00052940
000260          PRINT 2180, TSEP(NTSEP)                                           00052950
000266          DO 8210 IP=1, 4                                                   00052960
000270             PRINT 2191, IP, (SEPTIM(IP,I), I=1, NTSEP)                     00052970
000305             PRINT 2200, SEPTIM(IP, NTSEP)                                  00052980
000315    8210 CONTINUE                                                           00052990
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

000317          DO 8220 IP=5,8                                          00053000
000321             IPP=IP-4                                             00053010
000323             PRINT 2192, IPP, (SEPTIM(IP,I),I=1,MTSEP)            00053020
000340             PRINT 2200, SEPTIM(IP, NTSEP)                        00053030
000350     8220 CONTINUE                                                00053040
000352          DO 8230 IP=9,10                                         00053050
000354             IPP=IP-8                                             00053060
000356             PRINT 2193, IPP, (SEPTIM(IP,I),I=1,MTSEP)            00053070
000373             PRINT 2200, SEPTIM(IP, NTSEP)                        00053080
000403     8230 CONTINUE                                                00053090
000405          DO 8240 IP=11,12                                        00053100
000407             IPP=IP-10                                            00053110
000411             PRINT 2194, IPP, (SEPTIM(IP,I),I=1,MTSEP)            00053120
000426             PRINT 2200, SEPTIM(IP, NTSEP)                        00053130
000436     8240 CONTINUE                                                00053140
000440          DO 8250 IP=13,14                                        00053150
000442             IPP=IP-12                                            00053160
000444             PRINT 2195, IPP, (SEPTIM(IP,I),I=1,MTSEP)            00053170
000461             PRINT 2200, SEPTIM(IP, NTSEP)                        00053180
000471     8250 CONTINUE                                                00053190
000473          DO 8260 IP=15,16                                        00053200
000475             IPP=IP-14                                            00053210
000477             PRINT 2196, IPP, (SEPTIM(IP,I),I=1,MTSEP)            00053220
000514             PRINT 2200, SEPTIM(IP, NTSEP)                        00053230
000524     8260 CONTINUE                                                00053240
000526          MDSEP=NDSEP-1                                           00053250
000530          PRINT 2210, (DSEP(I), I=1, MDSEP)                       00053260
000543          PRINT 2220, DSEP(NDSEP)                                 00053270
000551          DO 8310 IP= 1, 4                                        00053280
000553             PRINT 2191, IP, (SEPDIS(IP,I), I=1, MDSEP)           00053290
000570             PRINT 2200, SEPDIS(IP, NDSEP)                        00053300
000600     8310 CONTINUE                                                00053310
000602          DO 8320 IP=5,8                                          00053320
000604             IPP=IP-4                                             00053330
000606             PRINT 2192, IPP, (SEPDIS(IP,I), I=1, MDSEP)          00053340
000623             PRINT 2200, SEPDIS(IP, NDSEP)                        00053350
000633     8320 CONTINUE                                                00053360
000635          DO 8330 IP=9,10                                         00053370
000637             IPP=IP-8                                             00053380
000641             PRINT 2193, IPP, (SEPDIS(IP,I), I=1, MDSEP)          00053390
000656             PRINT 2200, SEPDIS(IP, NDSEP)                        00053400
000666     8330 CONTINUE                                                00053410
000670          DO 8340 IP=11,12                                        00053420
000672             IPP=IP-10                                            00053430
000674             PRINT 2194, IPP, (SEPDIS(IP,I), I=1, MDSEP)          00053440
000711             PRINT 2200, SEPDIS(IP, NDSEP)                        00053450
000721     8340 CONTINUE                                                00053460
000723          DO 8350 IP=13,14                                        00053470
000725             IPP=IP-12                                            00053480
000727             PRINT 2195, IPP, (SEPDIS(IP,I), I=1, MDSEP)          00053490
000744             PRINT 2200, SEPDIS(IP, NDSEP)                        00053500
000754     8350 CONTINUE                                                00053510
000756          DO 8360 IP=15,16                                        00053520
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION CCT 73 A          16.56.27. 73/12/21.

000760            IPP=IP-14                                                    00053530
000762            PRINT 2196, IPP, (SEPDIS(IP,I), I=1, MDSEP)                  00053540
000777            PRINT 2200, SEPDIS(IP, MDSEP)                                00053550
001007     8360 CONTINUE                                                       00053560
001011          MTARR=NTARR-1                                                  00053570
001013          PRINT 2230, (TARR(I), I=1, MTARR)                              00053580
001026          PRINT 2240, TARR(MTARR)                                        00053590
001034          DO 8400 IP=1, 12                                               00053600
001036             IP3=(IP-1)*3                                                00053610
001040             DO 8400 IQ=1, 3                                             00053620
001042                IPQ=IP3+IQ                                               00053630
001044                PRINT 2250, IP, ICFSET(IQ), (FASTIM(IPQ,I), I=1, MTARR)  00053640
001064                PRINT 2260, FASTIM(IP, NTARR)                            00053650
001074     8400 CONTINUE                                                       00053660
001100          NDLY1=NDLY-1                                                   00053670
001102          PRINT 2270, (DLYSTF(I), I=1,NDLY1)                             00053680
001115          PRINT 2280, DLYSTF(NDLY)                                       00053690
001123          PRINT 2290, (DLYSA(I), I=1, NDLY1)                             00053700
001136          PRINT 2300, DLYSA(NDLY)                                        00053710
001144          PRINT 2310, (DLYCA(I), I=1, NDLY1)                             00053720
001157          PRINT 2320, DLYCA(NDLY)                                        00053730
001165          PRINT 2330, (DLYTA(I), I=1, NDLY1)                             00053740
001200          PRINT 2340,  DLYTA(NDLY)                                       00053750
001206          PRINT 2350, (DLYSD(I), I=1, NDLY1)                             00053760
001221          PRINT 2360, DLYSD(NDLY)                                        00053770
001227          PRINT 2370, (DLYCD(I), I=1, NDLY1)                             00053780
001242          PRINT 2380, DLYCD(NDLY)                                        00053790
001250          PRINT 2390, (DLYTD(I), I=1, NDLY1)                             00053800
001263          PRINT 2400, DLYTD(NDLY)                                        00053810
001271          RETURN                                                         00053820
                C                                                            00053830
001272     2010 FORMAT(1H1/1HO, 20X, 32H****** OUTPUT  SUMMARY ******/         00053840
               1       1HO/1HO,  9X, 18HI. TRAFFIC SUMMARY/ 1HO/              00053850
               2       1HO, 11X, 16H1. ARRIVALS/HOUR, 6X, 1H=, 4X, F7.2/      00053860
               3       1HO, 11X, 18H2. DEPARTURES/HOUR, 4X, 1H=, 4X, F7.2/    00053870
               4       1HO, 11X, 43H3. OPERATIONS/HOUR THROUGH EACH CONTROL FIX ) 00053880
001272     2030 FORMAT(1HO// 12X, 39H4. NO. OF OPERATIONS/HOUR THROUGH EACH ,   00053890
               1       22H(PARALLEL) FLIGHT PATH//                            00053900
               2        52HO  PATH      PO1A   PO1B   PO1C   PO2A   PO2B   PO2C , 00053910
               3        21H   PO3A   PO3B   PO3C / 9H   NO. OF / 7H   OPS., 3X, 00053920
               4        9F7.2 // 45HO  PATH      PO4A   PO4B   PO4C   PO5A   PO5B,00053930
               5        28H   PO5C   PO6A   PO6B   PO6C / 9H   NO. OF / 7H   OPS.,00053940
               6        3X, 9F7.2 // 41HO  PATH      PO7A   PO7B   PO7C   PO8A  ,00053950
               7        32HPO8B   PO8C   PO9A   PO9B   PO9C / 9H   NO. OF /     00053960
               8        7H   OPS., 3X, 9F7.2 // 27HO  PATH      P10A   P10B   ,  00053970
               9        46HP10C   P11A   P11B   P11C   P12A   P12B   P12C /      00053980
               A        9H   NO. OF / 7H   OPS., 3X, 9F7.2 )                   00053990
001272     2080 FORMAT (1H1/1HO/1HO, 9X, 19HIII. DELAY  SUMMARY/                 00054000
               1       1HO, 11X, 5HDELAY, 11X, 4HMEAN, 8X, 8HVARIANCE, 8X,    00054010
               2       18HP(D.GT.D-CRITICAL)/ 12X, 4HTYPE, 9X, 9H(MINUTES),   00054020
               3       5X, 9H(MINUTES)/ 1HO, 8X, 9HSCHEDULED, 8X,             00054030
               4       F6.2, 8X, F6.2, 14X, F6.4/                             00054040
               5       1HO, 7X, 11HOPERATIONAL, 7X, F6.2, 8X, F6.2, 14X, F6.4/ 00054050
```

TABLE 5-1.—CONTINUED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

              6       1HO, 10X, 5HTOTAL, 10X, F6.2, 8X, F6.2, 14X, F6.4)        00054060
001272    2210 FORMAT (1H1/1HO, 2X, 29H3. IN-AIR SEPARATION DISTANCE/ 1HO,31X,   00054070
              1       41HDISTANCE  SEPARATION  IN  NAUTICAL  MILES/ 3X,         00054080
              2       8HLOCATION, 5X, 5HUNDER , 76X, 4HOVER /17X, 16(F4.1, 1X)) 00054090
001272    2220 FORMAT (1H+, 96X,  F4.1)                                         00054100
001272    2230 FORMAT (1H1, 4 , 25H4. TIME DISTRIBUTIONS . .,                   00054110
              1       29H ARRIVALS  (IN TERMINAL AREA)//28X,                    00054120
              2       37HARRIVAL  EXECUTION  TIME  IN  MINUTES/ 6X, 8HLOCATION ,00054130
              3       2X, 5HUNDER , 76X, 4HOVER/15X, 1615)                     00054140
001272    2240 FORMAT (1H+, 96X, I3)                                            00054150
001272    2250 FORMAT (6X, 4HPATH, I2, A1, 3X, 16F5.2)                          00054160
001272    2260 FORMAT (1H+, 95X, F5.2)                                          00054170
001272    2270 FORMAT (1H1/1HO/1HO, 2X, 22H5. DELAY DISTRIBUTIONS/ 1HO,         00054180
              1       50X, 18HDELAY  IN  MINUTES/ 19X,                          00054190
              2       7H UNDER , 76X, 4HOVER /21X, 16(I3,2X))                   00054200
001272    2280 FORMAT (1H+, 101X,I3// 4X, 1HA)                                  00054210
001272    2290 FORMAT (4X, 13HR   SCHEDULED ,3X, 16F5.2)                        00054220
001272    2300 FORMAT (1H+, 100X, F5.2/ 4X, 1HR)                                00054230
001272    2310 FORMAT (4X, 15HI   OPERATIONAL, 1X, 16F5.2)                      00054240
001272    2320 FORMAT (1H+,  100X, F5.2/ 4X, 1HV)                               00054250
001272    2330 FORMAT (4X, 9HA   TOTAL, 7X, 16F5.2)                             00054260
001272    2340 FORMAT(1H+, 100X, F5.2/4X, 1HL//4X, 1HD/4X, 1HE)                 00054270
001272    2350 FORMAT (4X, 13HP   SCHEDULED, 3X, 16F5.2)                        00054280
001272    2360 FORMAT (1H+, 100X, F5.2/ 4X, 1HA)                                00054290
001272    2370 FORMAT (4X, 15HR   OPERATIONAL, 1X, 16F5.2)                      00054300
001272    2380 FORMAT (1H+, 100X, F5.2/ 4X, 1HT)                                00054310
001272    2390 FORMAT (4X, 9HU   TOTAL, 7X, 16F5.2)                             00054320
001272    2400 FORMAT (1H+, 100X, F5.2/ 4X, 1HR/ 4X, 1HE )                      00054330
001272    2020  FORMAT(1HO/1HO, 21X, 6HI.A.F., 23X, 4HT.F. //3X, 4HC.F., 7X, 1H1, 00054340
              2       6X, 1H2, 6X, 1H3, 6X, 1H4, 6X, 1H1, 6X, 1H2, 6X, 1H3, 6X, 00054350
              3       1H4 / 8H  NO. OF/3X, 4HOPS., 3X,  8F7.2 //1HO, 15X,       00054360
              4       4HM.F.,9X, 6HF.A.F., 9X, 4HO.M., 10X, 4HT.H. //3X,4HC.F., 00054370
              5       7X, 1H1, 6X, 1H2, 6X, 1H1, 6X, 1H2, 6X, 1H1, 6X, 1H2, 6X, 00054380
              6       1H1, 6X, 1H2 / 8H  NO. OF /3X, 4HOPS., 3X, 8F7.2)          00054390
001272    2040  FORMAT(1H1//1HO, 9X, 18HII. SAFETY SUMMARY/1HO//4X, 10H OPERATION,00054400
              1       5H TYPE, 8X, 3HD/D, 7X, 3HD/A, 7X, 3HA/D, 7X, 3HA/A //8X,  00054410
              2       8HLOCATION,6X,4(10H    RUNWAY )//3X,15H   P(T.LT.TMIN),4X,  00054420
              3       4F10.5 //3X, 15H   F(D.LT.DMIN), 4X, 4F10.5)               00054430
001272    2050  FORMAT(1HO/1HO//8X, 8HLOCATION, 5X, 23H  IAF1   IAF2   IAF3   ,  00054440
              1       31HIAF4   TF1    TF2    TF3    TF4 //6X,12HP(T.LT.TMIN),3X,00054450
              2       8F7.4 //6X,12HP(D.LT.DMIN), 3X, 8F7.4 /1HO/8X, 8HLOCATION, 00054460
              3       7X, 49HMF1    MF2    FAF1   FAF2   CM1    CM2    TH1    , 00054470
              4       3HTH2 //6X,12HP(T.LT.TMIN),3X,8F7.4 //6X,12HP(D.LT.DMIN), 00054480
              5       3X, 8F7.4)                                                 00054490
001272    2090  FORMAT(1H1/1HO/34X,32H****** OPERATION STATISTICS ******/1HO//2X,  00054500
              1       44H1. RUNWAY INTER-OPERATION TIME DISTRIBUTIONS /1HO, 34X, 00054510
              2       31HINTER-OPERATION TIME IN SECONDS /10H OPERATION, 1X,     00054520
              3       5HUNDER,84X, 4HOVER / 4X, 4HTYPE,2X,I3, 14(3X, I3))       00054530
001272    2110  FORMAT(1H+,99X, I3)                                             00054540
001272    2120  FORMAT(1HO, 3X, 3HD/D, 1X, 15F6.3)                              00054550
001272    2130  FORMAT(1H+, 97X, F6.3)                                          00054560
001272    2140  FORMAT(1HO, 3X, 3HD/A, 1X, 15F6.3)                              00054570
001272    2150  FORMAT(1HO, 3X, 3HA/D, 1X, 15F6.3)                              00054580
```

TABLE 5-1.—CONCLUDED

```
RUN  VERSION OCT 73 A          16.56.27. 73/12/21.

001272   2160  FORMAT(1HO, 3X, 3HA/A, 1X, 15F6.3)                                          00054590
001272   2170  FORMAT(1H1/1HO, 2X, 25H2. IN AIR SEPARATION TIME // 41X, 5HTIME ,           00054600
               1       21HSEPARATION IN MINUTES / 3X,8HLOCATION,5X,5HUNDER,76X,          00054610
               2       4HOVER / 16X, 16(1X, F4.1))                                       00054620
001272   2180  FORMAT(1H+, 96X, F4.1)                                                      00054630
001272   2191  FORMAT(1HO, 4X, 3HAF , I1,  7X, 16F5.2)                                    00054640
001272   2192  FORMAT(1HO, 4X, 3HTF , I1,  7X, 16F5.2)                                    00054650
001272   2193  FORMAT(1HO, 4X, 3HMF , I1,  7X, 16F5.2)                                    00054660
001272   2194  FORMAT(1HO, 4X, 3HFAF, I1,  7X, 16F5.2)                                    00054670
001272   2195  FORMAT(1HO, 4X, 3HOM , I1,  7X, 16F5.2)                                    00054680
001272   2196  FORMAT(1HO, 4X, 3HTH , I1,  7X, 16F5.2)                                    00054690
001272   2200  FORMAT(1H+, 95X, F5.2)                                                      00054700
001272         END                                                                         00054710
```