

NL

END
DATE
FILMED
7 80
DTIC

L. E. #

B.S.

USAFETAC/TN-80/002

ADA 085486



WIND FACTOR SIMULATION MODEL

User's Manual

Roger C. Whiton, Maj, USAF
Patrick L. Herod, Capt, USAF

April 1980

DTIC
JUN 16 1980

A

Approved For Public Release; Distribution Unlimited

UNITED STATES AIR FORCE
AIR WEATHER SERVICE (MAC)

USAF ENVIRONMENTAL
TECHNICAL APPLICATIONS CENTER

SCOTT AIR FORCE BASE, ILLINOIS 62225

80 6 13 036

DDC FILE COPY.

REVIEW AND APPROVAL STATEMENT

USAFETAC/TN-80/002, Wind Factor Simulation Model: User's Manual, April 1980, is approved for public release. There is no objection to unlimited distribution of this report to the public at large, or by the Defense Technical Information Center (DTIC) to the National Technical Information Service (NTIS).

This technical note has been reviewed and is approved for publication.


GERALD J. DITTMER, Maj, USAF
Chief, Aerospace Sciences Branch
Reviewing Officer

FOR THE COMMANDER


WALTER S. BURGMANN
Scientific and Technical
Information Officer (STINFO)

12 MAY 1980

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER USAFETAC/TN-86/002	2. GOVT ACCESSION NO. AD-A085486	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle)		5. TYPE OF REPORT & PERIOD COVERED Final Report
6. WIND FACTOR SIMULATION MODEL: User's Manual.		7. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Roger C./Whiton Maj, USAF Patrick L./Herod Capt, USAF		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS USAF Environmental Technical Applications Center Scott AFB, Illinois 62225		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS USAF Environmental Technical Applications Center Scott AFB, Illinois 62225		12. REPORT DATE April 1980
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 12/66		13. NUMBER OF PAGES 66
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Computer applications; computer programs; computer program documentation; computerized simulation; equivalent headwind technique; great circle; great circle distance; great circle equation; great circle heading; great circle navigation; manual; meteorology; models; navigation; simulation; subroutines; user needs; weather; wind; wind factor; wind factor simulation.		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) User instructions and a concise description are provided for a Wind Factor Simulation Model (WFSM). The WFSM is a fast, economical module designed to reside as a collection of subroutines within the user's larger simulation model. The WFSM, upon call by the user, produces mean overall climatological wind factors for great circle routes between arbitrary points 'A' and 'B' (specified by latitude and longitude) anywhere on the globe. The WFSM produces wind factors in any of three modes (calm wind case, 90-percent worst case, and the mean wind case), for either of two altitudes (25,000 ft and 35,000 ft) for any (Cont'd)		

DD FORM 1 JAN 73 1473

400945 Jm

iii

UNCLASSIFIED

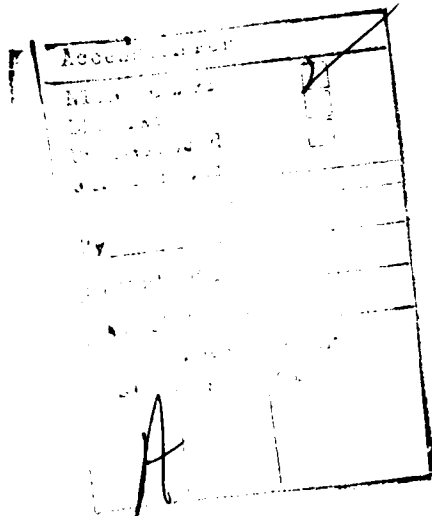
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

20. ABSTRACT (Cont'd)

→ of four seasons of the year. In addition, the model can provide great circle distance between points "A" and "B." From this information and known airspeed, the user can calculate ground speed and adjusted flying time between "A" and "B." Software solves the equation of a great circle. Program listing and flow chart are included.



iv UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

USER'S MANUAL

TABLE OF CONTENTS

	PAGE
SECTION 1.0 GENERAL	1
1.1 Purpose of the User's Manual.	1
1.2 Project References.	1
1.2.1 Project Request	1
1.2.2 Documentation on the Project.	1
1.2.3 Documentation Concerning Related Projects	1
1.2.4 Standards or Reference Documentation.	1
1.2.4.1 Documentation Standards and Specifications.	1
1.2.4.2 Programming Conventions	1
1.2.4.3 DOD or Federal Standards.	1
1.3 Terms and Abbreviations	1
1.4 Security and Privacy.	2
SECTION 2.0 SYSTEM SUMMARY.	3
2.1 System Application.	3
2.2 System Operation.	3
2.3 System Configuration.	4
2.4 System Organization	4
2.5 Performance	6
2.5.1 Capabilities.	6
2.5.2 Assumptions	6
2.5.3 Limitations	6
2.5.4 Processing Time	6
2.5.5 Flexibility	6
2.5.6 Error Detection	7
2.6 Data Base and Grid System	7
2.6.1 Data Base	7
2.6.2 Grid System	9
2.7 General Description of Inputs, Processing, Outputs.	10
2.7.1 Input	10
2.7.2 Processing.	10
2.7.3 Outputs	11
SECTION 3.0 STAFF FUNCTIONS RELATED TO TECHNICAL OPERATIONS	12
3.1 Initiation Procedures	12
3.2 Staff Input Requirements.	12
3.2.1 Input Formats and Composition Rules	13
3.3 Output Requirements	13
3.4 Utilization of System Outputs	13
3.5 Recovery and Error Correction Procedures.	14
APPENDIX A WFSM Flow Chart	15
APPENDIX B WFSM Program Listing.	21
ENRWNO.	21
GRTCIR.	38
SPHGLO.	50
DISTAN.	53
HDG	55
BRLNG	57
BRLAT	59

LIST OF FIGURES

Figure 1	System Operation.	3
Figure 2	Hierarchy of Subroutine Calls	5
Figure 3	WFSM as Server-Model within User's Simulation Model	5
Figure 4	Grid System for WFSM.	10

LIST OF TABLES

Table 1	WFSM Subprograms.	4
Table 2	Error Detection Features of WFSM.	8
Table 3	Structure of Wind Data File	9
Table 4	Wind Data File Record Elements.	9
Table 5	Input Arguments to Subroutine ENRWND.	13

SECTION 1.0 GENERAL

1.1 Purpose of the User's Manual. The objective of the user's manual (UM) for the Wind Factor Simulation Model (WFSM), USAFETAC Project 1923, is to provide the information necessary to use the model effectively. This user's manual is supplemented by USAFETAC/TN-80/001, Wind Factor Simulation Model, April 1980, which provides an in-depth treatment of the model's science and mathematics. This manual refers to USAFETAC/TN-80/001 frequently. Readers may wish to have the technical note available as supplementary reading.

1.2 Project References. The Wind Factor Simulation Model (WFSM) is embodied in a constellation of computer subroutines that calculate simulated wind factors for a larger simulation model in which the WFSM resides. Originally, the WFSM was written to serve the Military Airlift Command's (MAC) airlift system simulation called COLOSSUS. COLOSSUS is MAC's attempt to test by means of computer simulation its ability to respond to a contingency at any place or any time. MAC perceived a need to enhance the realism of COLOSSUS by adding simulated weather. The formal request for this weather support is dated 24 July 1978. This request called for three weather modules: (a) terminal weather for takeoff/departure; (b) enroute refueling weather; and (c) enroute wind factors. Efforts to meet requirement (a) began as early as 1977. This UM documents software developed to meet requirement (c).

1.2.1 Project Request. 7WW/DON letter to USAFETAC/DO, Request for Weather Data, 24 Jul 1978.

1.2.2 Documentation on the Project. USAFETAC/TN-80/001, Wind Factor Simulation Model: Model Description, April 1980.

1.2.3 Documentation Concerning Related Projects. None.

1.2.4 Standards or Reference Documentation.

1.2.4.1 Documentation Standards and Specifications. Documentation is in accordance with DOD Standard 7935.1-S, Automated Data Systems Documentation Standards, 13 September 1977, and AFM 171-100, 300-12, and 300-15.

1.2.4.2 Programming Conventions. American National Standard X3.9-1966 FORTRAN programming convention has been adhered to except in the following cases:

a. Extensive use of the Honeywell 6000-Series FORTRAN Execution Error Monitor (FEXEM) has been made to assist in debugging.

b. Continuation statements are handled with an ampersand in column 1 rather than a nonblank character in column 6. This permits running the WFSM in Honeywell TSS VFORTRAN as well as in CARDIN.

1.2.4.3 DOD or Federal Standards. Documentation provided in the UM is in accordance with DOD Standard 7935.1-S, 13 September 1977.

1.3 Terms and Abbreviations:

Model: A representation, description, or imitation of a system or physical process (such as the atmosphere) in another medium (such as a computer). A model is a simplified, generalized conceptualization of complex reality, usually based on a set of simplifying assumptions needed to obtain tractable solutions. The model is so constructed as to behave similarly to its prototype system or physical process in some sense considered critical to the problem at hand. The model abstracts or preserves suitably chosen "essential" properties of the system or process being modeled.

Simulation: A numerical technique by which systems or processes are modeled digitally in order to study the behavior of the process or system being simulated, usually as a function of time. Simulation has been described as attaining the essence without the reality. Its purpose is to permit drawing conclusions about the real-world system or process through use of the simulator as a tool for study. All simulations are models, but not all models are simulators.

Environmental Simulation: A selectively realistic synthesis of aerospace behavior consistent in space and time. A technique, often involving mathematical and probabilistic models, of describing or analyzing the environment of the effects of the environment on the system. An environmental simulation model can stand alone or can operate within a larger simulation model such as COLOSSUS.

COLOSSUS: Name given to MAC's attempt to simulate their airlift forces' capability to respond to a contingency anywhere at any time.

Wind Factor: The difference between the ground speed of an aircraft and its true airspeed.

Ninety Percent Worst: A wind factor whose value is exceeded 90 percent of the time, i.e., there is a 10-percent risk that one will encounter a wind factor more negative than the 90-percent worst wind factor.

Great Circle Distance (GCD): The distance between two points computed along a great circle of the earth. A great circle of the earth is any circle around the surface of the earth whose center coincides with the center of the earth. All longitude lines are great circles. The only latitude line that is a great circle is the equator.

Newton's Iterative Method: A mathematical technique applied in subroutine GRTCIR to solve a transcendental equation (see USAFETAC/TN-80/001).

Sawyer's Equivalent Headwind Technique: The mathematical technique used in the WFSM to compute wind factors. For details see USAFETAC/TN-80/001 and AWS-TR-77-267, Guide for Applied Climatology, November 1977.

1.4 Security and Privacy. This UM and the Wind Factor Simulation Model (WFSM) for which it is written are unclassified and can be released to the public. No privacy restrictions are associated with the use of this system to include input, output, data base or programs.

SECTION 2.0 SYSTEM SUMMARY

2.1 System Application. The Wind Factor Simulation Model (WFSM) was designed and developed to be incorporated in the Military Airlift Command's simulation of airlift forces. This advanced computer simulation effort is called COLOSSUS.

Designers of the COLOSSUS simulation determined that three weather elements most seriously impact airlift system operations: terminal weather for departure/recovery, enroute visibility for inflight refueling, and enroute winds. The WFSM was developed in response to the third COLOSSUS requirement. Wind factors generated by the WFSM enhance the effectiveness of the COLOSSUS simulation by enabling calculation of realistic flight times based on climatological winds.

The WFSM provides to COLOSSUS the ground speed of an aircraft on a simulated flight from point "A" (takeoff) to point "B" (landing), two points anywhere on the globe. It does this by calculating an overall enroute mean wind factor between these points by the Sawyer equivalent headwind technique and then applying the result to a simple algebraic expression for ground speed. For details on how this is accomplished, see USAFETAC/TN-80/001.

There are at least two other applications for the WFSM. First, it can be used, with appropriate modifications, as an efficient method of generating wind and great circle navigation information. It is efficient in terms of computer run time and core storage. Its efficiency is especially evident in comparisons between the current operational wind factor models and the WFSM. In addition, this module, again with required changes, can be used in other simulators. Simulation efforts are increasing in number and complexity. It follows that a requirement for a module incorporating weather information should be included in these simulation efforts. The WFSM can be a partial fulfillment of that requirement.

2.2 System Operation. Before the WFSM is first invoked, a user-supplied initializing module must read the wind data base from a sequential file (card, tape, or disk; but as implemented on MAC's Honeywell Series-6000 computer, a disk file) into the computer's memory.

After the data base has been loaded into the computer's memory, the user at any time may invoke the WFSM by calling its main subroutine, ENRWMD. In that call, the user tells ENRWMD the location of point "A" (takeoff) and point "B" (landing), the aircraft altitude and airspeed, and the date/time for which the wind factor is requested. The WFSM then responds by stepping the aircraft along a great circle route between "A" and "B", calculates a route-mean wind factor, adds that wind factor to the given airspeed, and returns route-mean ground speed to the user. The process is illustrated in Figure 1.

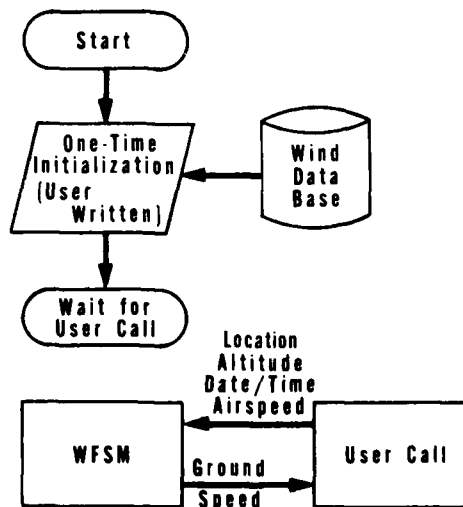


FIGURE 1. System Operation.

2.3 System Configuration. The WFSM was developed for use on the MAC Honeywell 6000-Series computers at Scott AFB, Illinois. In particular, the WFSM executes on a Honeywell 6080 computer. During program design and coding, the goal of intercomputer compatibility was for the most part adhered to. With exceptions noted in paragraph 1.2.4.2, the WFSM should compile and execute on any computer with a FORTRAN compiler.

The wind data base for the WFSM resides in the computer's core storage during execution and on a sequential file (card, tape, or disk) before execution. A user-supplied initializing module external to the WFSM must read that data base from the file into the computer's memory before WFSM is first invoked. As implemented on the MAC Honeywell 6080, the wind data base resides on a permanent sequential disk file.

Configuration necessary for a computer to use WFSM typically requires a FORTRAN compiler, linking loader, disk storage, and FORTRAN disk input/output capability.

2.4 System Organization. The WFSM is coded as a constellation of seven FORTRAN subprograms with the functions described in Table 1. The principal subroutine is ENRWND, with which the user's program communicates directly via arguments of the FORTRAN CALL statement. Subroutine ENRWND, in turn, calls other subprograms, some of which call still other subprograms. The hierarchy of subroutine calls is shown in Figure 2.

TABLE 1. WFSM SUBPROGRAMS.

<u>Subprogram</u>	<u>Purpose</u>
ENRWND	Main enroute wind factor subprogram, called directly by user to compute ground speed from given airspeed in any of three modes for two flight levels and four seasons of the year.
DISTAN	Called by ENRWND and HDG or directly by user to compute great circle distance between any two points on the globe.
SPHGLO	Conversion of spherical coordinates to global latitude and longitude or vice-versa. Called by GRTCIR.
GRTCIR	Solution of great circle equation for latitude given longitude, or for longitude given latitude. Called by ENRWND.
HDG	Calculation of initial heading along a great circle route flown from a given origin to a given destination. Called by ENRWND and GRTCIR.
BPLNG	For a spherical grid system whose longitude boundaries are spaced at 30° intervals, finds the longitude grid values bracketing a given arbitrary longitude. Called by ENRWND.
RPLAT	For a spherical grid system whose latitude boundaries are spaced at 15° intervals, finds the latitude grid values bracketing a given arbitrary latitude. Called by ENRWND.

Detailed explanation of the processing done by each subprogram is available in USAFETAC/TN-80/001.

As shown in Figure 3, the WFSM is designed to operate as a miniature, subservient simulation within the user's overall simulation model. As such, WFSM is supplied as

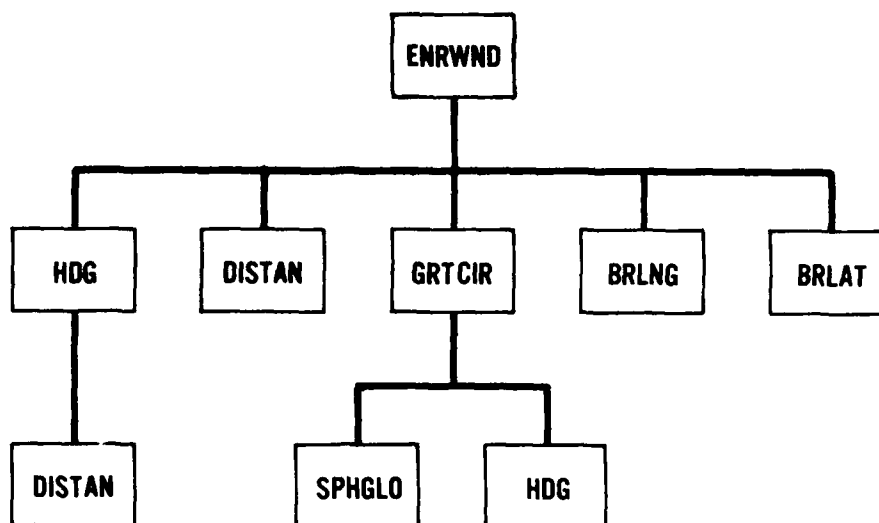


FIGURE 2. Hierarchy of Subroutine Calls.

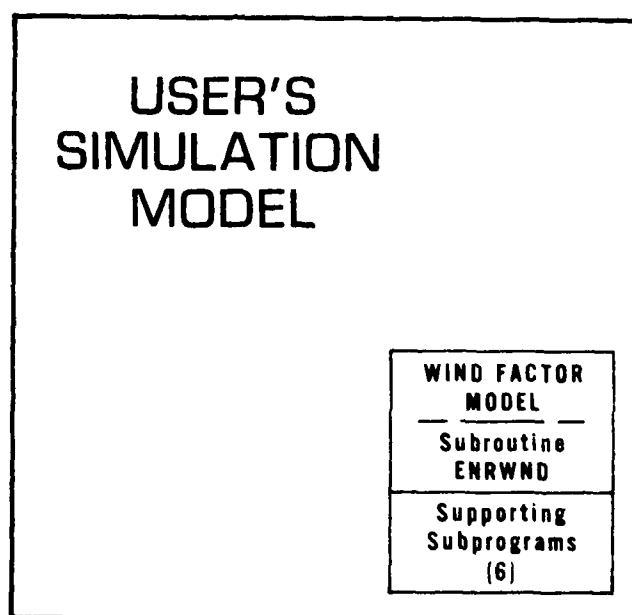


FIGURE 3. WFSM as Server-Model within User's Simulation Model.

a constellation of FORTRAN subroutines. Even the main WFSM module, called ENRWND, is a subroutine. All communication between WFSM and the encompassing overall simulation is done through the arguments of the user's CALL statement and corresponding arguments of the SUBROUTINE ENRWND statement.

2.5 Performance.

2.5.1 Capabilities. The WFSM is capable of producing mean overall climatological wind factors for great circle routes. It does so in any of three modes: calm wind case, 90-percent worst wind case, and the mean wind case. The wind factor can be produced for two flight levels and for four seasons. In addition, the model, through its subroutine DISTAN, can provide the great circle distance (GCD) between any two points over the globe. Since it produces both wind factors and GCD, it is capable of calculating both the ground speed and the adjusted flying time between points "A" and "B" if given the airspeed.

2.5.2 Assumptions. The WFSM assumes that all routes are great circle routes or can be subdivided into several legs each of which is a great circle segment. It is further assumed that climb winds and descent winds play a negligible role in determining the route-mean wind factor. In the model as presently written, the simulated aircraft is always "at altitude." Since an arithmetic rather than a harmonic mean is used in computing the route-mean wind factor, it is assumed that the ground speed is less than or equal to one third of the airspeed.

2.5.3 Limitations. Presently, the WFSM cannot produce a "simulated" wind having a day-to-day variability. Hence, within any given season and for a particular altitude, a particular route will always experience the same wind factor, regardless of the passage of time. Requests for the simulated wind default to the mean wind.

Furthermore, the present model is incapable of producing a "forecast" wind factor. Requests for a forecast wind default to the mean wind.

Temperature, aircraft performance, fuel consumption and other elements considered by typical flight planning models are not included in the present simulation, which deals only with wind.

Operations north of 75°N or south of 60°S are not permitted.

The coarseness of the present 15° latitude by 30° longitude grid precludes use of the model for calculation of the operationally realistic wind factors unless changes are made. Simple modifications to the existing data base and grid system, along with corresponding changes in the software, can remove this limitation.

Specifically, the following restrictions must be adhered to:

- a. No route should be flown directly over either pole.
- b. Neither pole should serve as a point "A" or "B."
- c. Circumferential or round-robin flights in which "A" and "B" coincide will be aborted. Such flights should be broken into smaller segments.
- d. Routes or segments of routes flown directly north or directly south along a longitude line will also abort.
- e. At present, the model allows flights only at altitudes of 25,000 feet and 35,000 feet.
- f. All departure and destination points must lie within the latitude range from 75°N to 60°S.

Restrictions e and f were imposed in order to reduce the core storage requirements of the model.

2.5.4 Processing Time. Generating a wind factor for a simulated aircraft flying a long route (4,725 nautical miles) requires approximately 0.1-0.5 seconds of central processing unit (CPU) time on a Honeywell 6080 general purpose computer. The CPU time depends on the number of legs in the simulated flight. Accordingly, this CPU time, referring as it does to a very long route, represents almost a worst case time estimate.

2.5.5 Flexibility. A number of the limitations of the WFSM can be overcome by selective modification of the model, the data base, or both. Since the WFSM was

built for economy in run time and storage, such modifications will have the side effect of increasing run time and/or storage. In particular, the resolution of the WFSM grid can easily be improved by changing delta-latitudes and delta-longitudes in the model, altering the grid numbering scheme, and adding to the data base. In addition, the number of vertical levels and time periods can be increased. With more effort, WFSM capabilities can be expanded to include a "simulated" (i.e., variable) wind by writing Part V of subroutine ENRWIND, which is presently set to default to the mean wind.

2.5.6 Error Detection. User inputs to the WFSM are scanned for legal values. In iterative calculations, iteration counters are used to flag computations that are diverging or failing to converge. The Honeywell 6000-Series FORTRAN Execution Error Monitor (FXEM) is invoked to display to the user any errors so identified. The FXEM error message, including error number and a plain text diagnostic, are displayed on SYSOUT. Error number 61 has been used for all WFSM errors. Whenever FXEM error 61 appears, the user knows that a problem associated with the WFSM has been detected. Each FXEM error has a disposition. Some permit the program to continue executing, while others produce a program abort. Presently, FXEM error 61 causes program abort. Users can alter this disposition by means of initializing call to the FXEM module. Errors and problems detected and treated by the WFSM are summarized in Table 2.

2.6 Data Base and Grid System

2.6.1 Data Base. Basic data for the WFSM are the USAFETAC Single Integrated Operational Plan (SIOP) winds. With a period of record extending from January 1972 to December 1976, the SIOP winds contain mean u-component, v-component, and vector standard deviation, tabulated on a 5-degree offset grid. This grid is much finer than that used by the WFSM.

To prepare a data base for the WFSM's 150 latitude by 300 longitude grid system, simple averaging is used. The u-component, the v-component, and the vector standard deviation are averaged separately. The average is performed by extracting from the SIOP winds three values latitudinally and six values longitudinally. The resulting 18 values are summed and the sum divided by 18 to obtain the mean data values for each grid sector. In this way, mean u-component, v-component, and vector standard deviation are obtained for all 108 grid sectors. Such a data base is constructed for January (winter), April (spring), July (summer), and October (fall) for altitudes 25,000 feet (taken from 400-mb winds) and 35,000 feet (taken from 250-mb winds). Using an in-house USAFETAC program WIND, the

$$4 \text{ season} \times 2 \text{ altitude} = 8 \text{ sets}$$

of 108-sector wind data are converted from u, v, and vector standard deviation to direction (beta angle, described in Appendix R, USAFETAC/TM-80/001), speed, and variance. After conversion, the winds are stored sequentially in a data file in the order described in Table 3. Each sector of the data file requires one record of disk storage containing the three elements of information shown in Table 4.

There are

$$4 \text{ seasons} \times 2 \text{ altitudes} \times 108 \text{ sectors} = 864 \text{ records}$$

of wind data in the data file. To be used by the WFSM, the data base must be loaded from the data file into the arrays,

DIR(870)	Wind Direction, Beta Angle, Radians
SPD(870)	Wind Speed, Knots
VAR(870)	Wind Variance, Knots ²

which are located in the COMMON block WEA and require 2.6K words of computer core storage. The COMMON block must be loaded with wind data before the WFSM is first executed. Most often, an initializing routine external to the WFSM is used for this purpose. That routine merely reads the 864 wind records in order and stores them,

TABLE 2. ERROR DETECTION FEATURES OF WFSM.

<u>Subroutine</u>	<u>Error Description/MESSAGE</u>	<u>FXEM Display</u>	<u>Disposition</u>
ENRWND	Wind option less than 0 or greater than 3 ILLEGAL WIND OPTION	Yes	Abort
ENRWND	Latitude or longitude out- side legal range ILLEGAL LAT/LON	Yes	Abort
ENRWND	Julian date greater than 366 or less than 1 ILLEGAL JULIAN DATE	Yes	Abort
ENRWND	Altitude index less than 1 or greater than 2 ILLEGAL ALTITUDE	Yes	Abort
ENRWND	Subroutine GRTCIR twice consecutively failed to converge when solving for longitude of Candidate Point #2 GRTCIR FAILED TWICE	Yes	Abort
ENRWND	Number of legs of the simu- lated flight exceeds a variable maximum number currently set to 50 RUNAWAY ROUTE	Yes	Abort
GRTCIR	Subroutine function vector less than 1 or greater than 5 ILLEGAL ICONV	Yes	Abort
GRTCIP	Direct north-south flight is attempted SOLUTION NOT UNIQUE IN THETA	Yes	Abort
GRTCIR	Newton's iterative method fails to converge, as indi- cated by an iteration count exceeding a variable maximum currently set at 7. Because of the way in which GRTCIR is used by ENRWND, failure to converge is sometimes normal behavior. In these cases no error message is displayed and the program continues. When, the GRTCIR function flag, ICONV, is set to 2, an error message SOLUTION DID NOT CONVERGE is displayed, and the pro- gram aborts.	No Yes	Continue Abort
SPHGLO	Erroneous function flag ICONV. Error code ICONV=10 returned	No	Continue
SPHGLO	Out-of-bounds latitude. Error code ICONV=11 returned	No	Continue

SPHGLO	Out-of-bounds longitude. Error code ICONV=12 returned	No	Continue
SPHGLO	Out-of-bounds θ Error code ICONV=13 returned	No	Continue
SPHGLO	Out-of-bounds ϕ Error code ICONV=14 returned	No	Continue

Table 3. STRUCTURE OF WIND DATA FILE.

<u>Season</u>	<u>Flight Level</u>	<u>Record Numbers</u>
Winter	250	1 - 108
	350	109 - 216
Spring	250	217 - 324
	350	325 - 432
Summer	250	433 - 540
	350	541 - 648
Fall	250	649 - 756
	350	757 - 864

Table 4. WIND DATA FILE RECORD ELEMENTS.

<u>Element</u>	<u>Units</u>	<u>Columns</u>	<u>FORTRAN Format</u>
Wind Direction (Beta Angle)	radians	1-15	F15.5
Wind Speed	knots	16-30	F15.5
Variance	knots ²	31-45	F15.5

without rearrangement, in the COMMON block WEA. Thereafter, the wind data base can be accessed in terms of a record number NRND as follows:

DIR(NRND) SPD(NRND) VAR(NRND)

The record number NRND is computed as follows:

$$\begin{aligned} \text{NRND} = & (\text{Season Index} - 1) \times (\text{Total Number Altitudes}) \times \\ & (\text{Total Number Grid Sectors}) \\ & + (\text{Altitude Index} - 1) \times (\text{Total Number Grid Sectors}) \\ & + (\text{Sector Number}) \end{aligned}$$

2.6.2 Grid System. The WFSM is required to be economical in terms of computer core storage and run time; yet the model must provide a global wind factor capability to the user. These conflicting requirements dictated the use of a coarse grid. The original idea, motivated by the latitude/longitude orientation of the winds used as input data, was to have a grid system in global (latitude/longitude) coordinates. The grid would have a spacing of 15° latitude by 30° longitude and would cover the globe, requiring 144 grid sectors (12 sectors from pole to pole times 12 sectors around the equator). Closer investigation of the latitudes of the COLLOSSUS terminals and routes indicated that grid sectors above 75°N and below 60°S were unnecessary. Since three quantities (wind direction, speed, and variance) are stored for each grid sector for four seasons and two flight levels, approximately 0.87K words of core storage could be saved by reducing the number of grid sectors from 144 to 108.

The grid system chosen for implementation is in global (latitude/longitude) coordinates with a resolution of 15° latitude by 30° longitude. The grid system extends from 75°N to 60°S latitude (9 rows) and globally in longitude (12 columns). Hence, there are 108 grid sectors. Numbering of the grid sectors is shown in Figure 4. It is important to note that column 1 is keyed on 30°W , not 0° .

		Column # →													
		1	2	3	4	5	6	7	8	9	10	11	12		
Row # ↓	1	1	2	3	4	5	6	7	8	9	10	11	12	75°N	
	2	13	14	15	16	17	18	19	20	21	22	23	24	60°N	
	3	25	26	27	28	29	30	31	32	33	34	35	36	30°N	
	4	37	38	39	40	41	42	43	44	45	46	47	48		
	5	49	50	51	52	53	54	55	56	57	58	59	60	0°	
	6	61	62	63	64	65	66	67	68	69	70	71	72		
	7	73	74	75	76	77	78	79	80	81	82	83	84	30°S	
	8	85	86	87	88	89	90	91	92	93	94	95	96		
	9	97	98	99	100	101	102	103	104	105	106	107	108	60°S	
		0°			90°E			180°			90°W				

FIGURE 4. Grid System for WFSM.

A grid system such as this, involving undefined regions near the poles, prevents flights over the poles (which would be complicated by the singularity in coordinate systems at the poles). The coarseness of the present grid system precludes use of this model for calculation of operational wind factors. A coarse grid is suitable for simulation, however. The grid could be made finer if improved meteorological resolution is desired.

2.7 General Descriptions of Inputs, Processing, Outputs.

2.7.1 Input. A user employing the WFSM to calculate a route-mean wind factor and ground speed must stipulate the global (latitude/longitude) coordinates of point "A" (takeoff) and point "B" (landing), the Julian base date of the wind factor request (used to determine season of the year), aircraft altitude (25,000 ft or 35,000 ft), wind option (calm, mean, or 90-percent worst), and airspeed.

To accommodate future growth, the user must also stipulate Greenwich mean base time of the wind factor request and forecast hours ahead of the wind factor requested. Because the WFSM does not as yet have a capability to generate forecast wind factors, these two inputs are ignored and may have any value.

2.7.2 Processing. The keystone subprogram of the WFSM is ENRWND, which accomplishes most of the aircraft navigation except solution of the equation of a great circle. It also accomplishes all of the meteorology involved in calculation of the wind factor (see Chapter 3, USAFETAC/TN-80/001).

The ENRWND subprogram navigates a simulated aircraft over a great circle route from point "A" to point "B" in a global (latitude/longitude-oriented) grid system whose resolution is 15° latitude by 30° longitude. The subroutine determines which grid sector the aircraft is in and the length of the aircraft's path through the sector. Then the routine consults the wind data base for the particular grid sector, flight level, and season to obtain sector-mean wind direction, speed, and variance. Cross-track and along-track components of the wind are computed. These are weighted by the length of the aircraft's track through the sector and are accumulated. At the

end of the simulated flight, the accumulators are divided by the total great circle distance for the flight, producing a distance-weighted, route-mean cross-track wind component, an along-track component and a variance. From these a route-mean wind factor is computed using Sawyer's equivalent headwind technique (see Chapter 3, USAFETAC/TN-80/001). If the 90-percent worst wind option has been selected, that wind factor is statistically adjusted to the 10-percent risk value. The wind factor is then added to the airspeed to obtain the route-mean airspeed. If the calm wind option is requested, all these computations are bypassed, and a zero wind factor is used. Presently, requests for the simulated wind option will default to the mean wind. In the mean wind case, the wind factor is developed directly from a mean wind data base. In the 90-percent worst case, this mean wind factor is statistically adjusted to a value such that 10 percent of the flights over that route will experience a worse wind factor (10-percent risk).

2.7.3 Outputs. The output from the WFSM (subroutine ENRWND) is the variable GSPEED, a real quantity representing the aircraft's ground speed in knots.

SECTION 3.0 STAFF FUNCTIONS RELATED TO TECHNICAL OPERATIONS

3.1 Initiation Procedures. The Wind Factor Simulation Model (WFSM) operates within a larger model, such as MAC's COLOSSUS. The larger model invokes the WFSM by issuing calls to its subroutines, principally subroutine ENRWND. Before the first call to WFSM, the WFSM must have been initialized by reading wind data from the data base into the COMMON block WEA. Users must provide this special initializing routine, as the WFSM code assumes wind data have already been read into core. A suitable routine is shown below:

```
COMMON /WEA/ DIR(870), SPD(870), VAR(870)
...
DO iii NREC = 1,864
  READ (fc,iii) DIR(NREC), SPD(NREC), VAR(NREC)
iii FORMAT (3F15.5)
iii CONTINUE
...
```

In this example, iii and jjj are integer FORTRAN statement numbers. The file code for the wind data base is fc.

Job control language (JCL) or its equivalent is needed to equate the logical file fc with a catalogued file name on the particular computer system used. Assuming that the wind data base is on a sequential disk file called PWIND.SQ within the subcatalogue MACRO in catalogue MACRO, then the Honeywell 6000-Series JCL is

```
$      PRMFL    fc,R,S,MACRO/MACRO/PWIND.SQ
```

where fc is the file code selected above. The \$ PRMFL JCL must appear behind the \$ EXECUTE record in a Honeywell job. Structure and format of the data base contained in the file PWIND.SQ is outlined in Table 3.

3.2 Staff Input Requirements. The WFSM communicates with the user's simulation model only through the argument list of the user's call to subroutine ENRWND.*

Whenever a wind factor is desired in the user's model, the user should place a call to ENRWND, supplying the values for input arguments. Subroutine ENRWND then takes control, calculates a route-mean wind factor and uses it to prepare the output argument, namely the ground speed of the aircraft. The user's call to subroutine ENRWND should be of the form,

```
CALL ENRWND (FRMLAT, FRMLNG, TOLAT, TOLNG, JULDAT,
            GMT, FCHRS, IALT, IOPTN, ASPEED, GSPEED)
```

Inputs to the WFSM are in the form of input arguments to subroutine ENRWND. These input arguments include the global (latitude/longitude) coordinates of point "A" (takeoff) and point "R" (landing). The user specifies which one of the four seasons he needs by stipulating a Julian base date (JULDAT) from 1 to 366. The user requests calm, mean, or 90-percent worst winds by specifying a wind option (IOPTN) from 0 to 3. The user then specifies the altitude of the aircraft by setting the flag IALT to 1 for 25,000 feet or 2 for 35,000 feet, and airspeed in knots in the input variable ASPEED.

*The exception is that user's may call subroutine DISTAN directly if they need a great circle distance, or function HDG if they want an initial great circle heading.

To accomodate future growth, the user must also stipulate GMT, the Greenwich mean base time (0.0 - 24.0) of the wind factor request, and FCHRS, a forecast hours ahead of the wind factor required. Presently, these inputs are ignored and may have any value.

A description of these input arguments is contained in Table 5.

TABLE 5. INPUT ARGUMENTS TO SUBROUTINE ENRWND.

<u>Variable Name</u>	<u>Data Type</u>	<u>Definition</u>	<u>Units/Values</u>
FRMLAT	Real	Latitude of point "A" (Takeoff)	Decimal degrees (75.00 to -60.00)
FRMLNG	Real	Longitude of point "A" (Takeoff)	Decimal degrees (180.00 to -180.00)
TOLAT	Real	Latitude of point "B" (Landing)	Decimal degrees (75.00 to -60.00)
TOLNG	Real	Longitude of point "B" (Landing)	Decimal degrees (180.00 to -180.00)
JULDAT	Integer	Julian base date of wind factor request	1 - 366
GMT	Real	Greenwich base time of wind factor request	0.0 - 24.0 hrs
FCHRS	Real	Forecast hours ahead	$\geq$ 0.0 hrs
IALT	Integer	Altitude index	1 = 25,000 ft 2 = 35,000 ft
IOPTN	Integer	Wind option or mode	0 = Calm wind 1 = Simulated wind (Defaults to 3) 2 = 90-percent worst wind 3 = Mean wind
ASPEED	Real	Airspeed	Knots

3.2.1 Input Formats and Composition Rules. Input arguments for subroutine ENRWND are presented in Table 5. Special format rules are as follows:

a. Latitudes and longitudes must be in decimal degrees, not the degrees-minutes-seconds system, e.g., 36.50 rather than 36° 30'.

b. South latitudes and east longitudes must be negative.

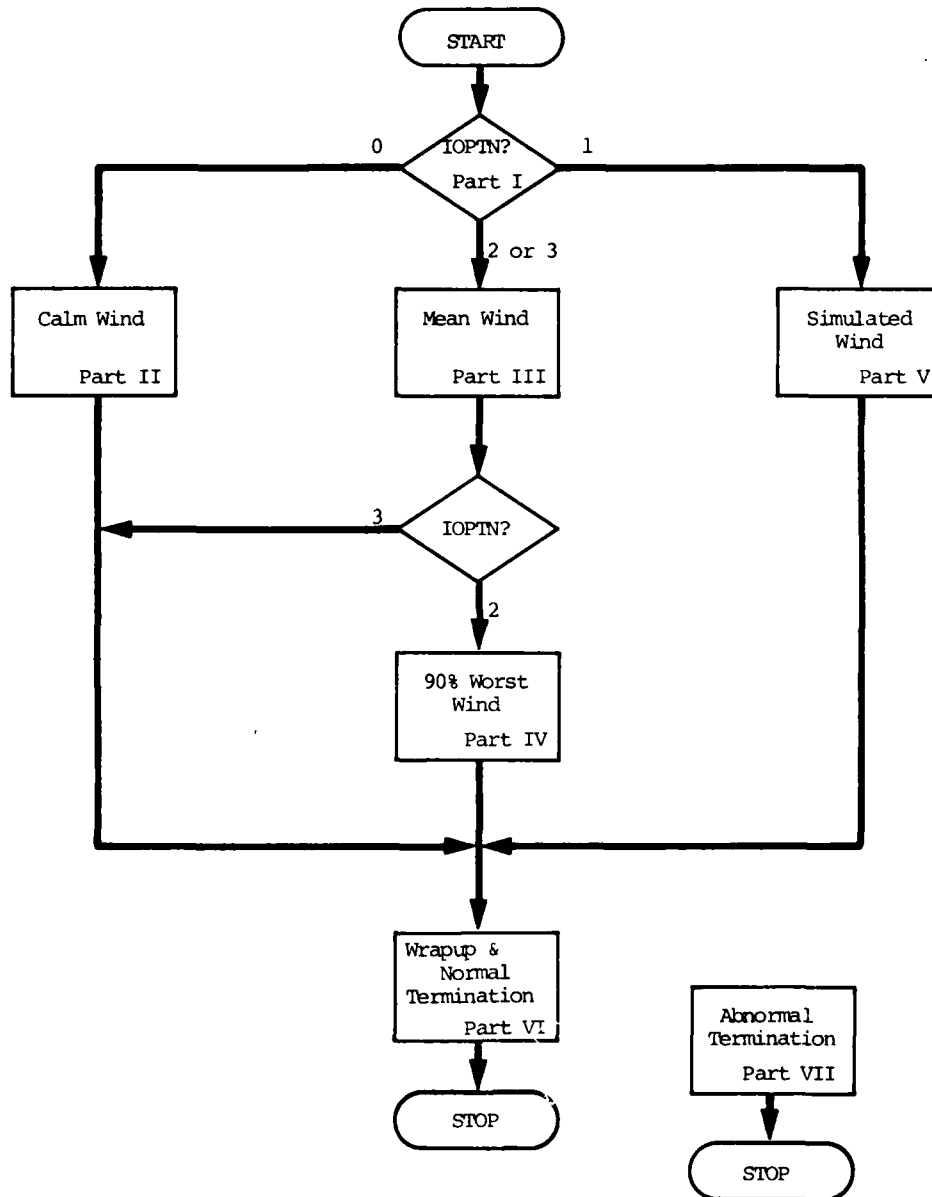
3.3 Output Requirements. Output argument for subroutine ENRWND is the variable GSPEED, a real quantity representing the aircraft's ground speed in knots. In addition, users may call subroutine DISTAN directly for a great circle distance, or function HDG for an initial great circle heading.

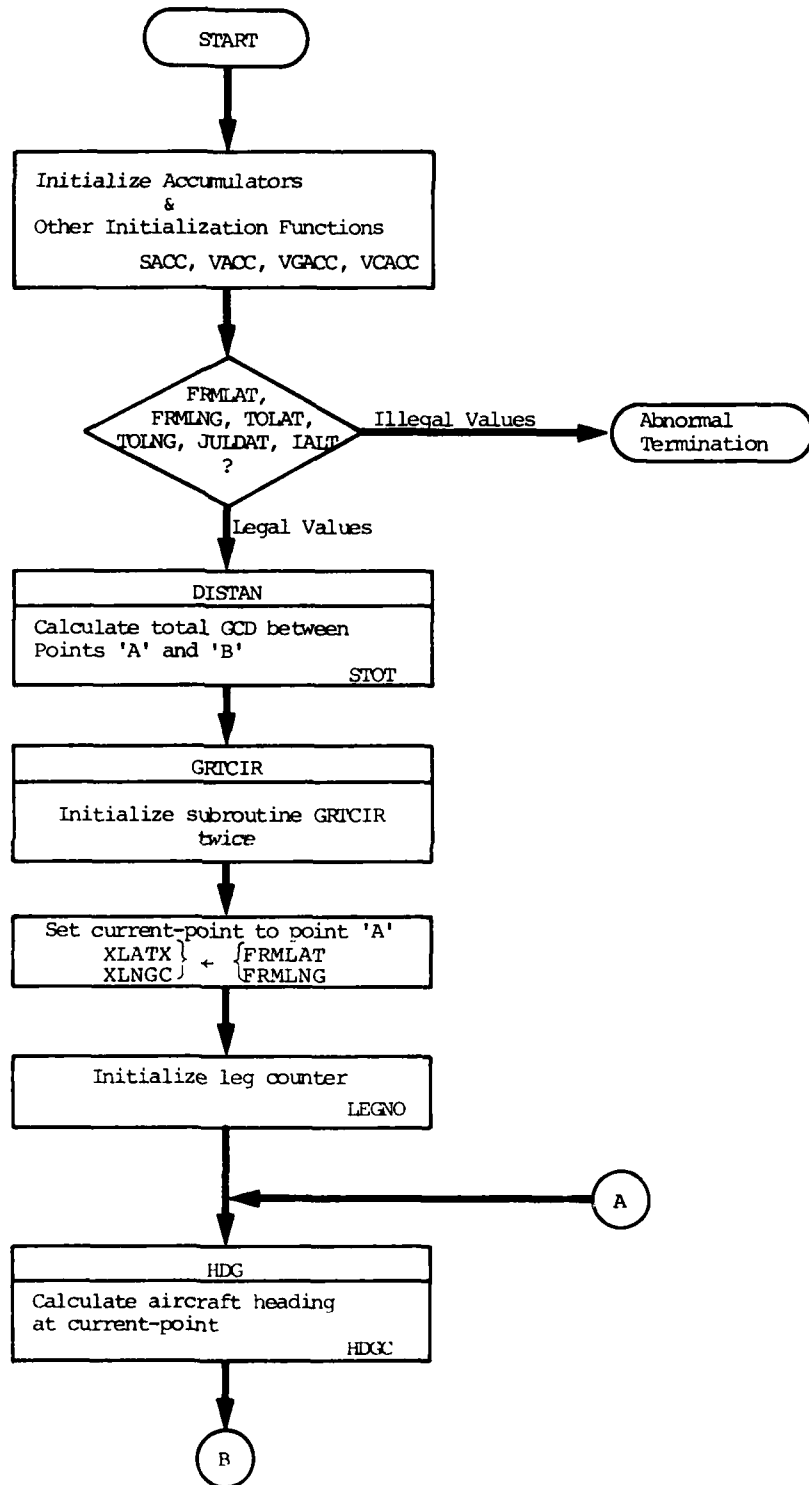
3.4 Utilization of System Outputs. The calling program, e.g., a simulation such as MAC's COLLOSSUS, uses the WFSM to obtain the ground speed of an aircraft flying from point "A" to point "B". In order to provide the ground speed, the WFSM must calculate a wind factor. Most of the work in this module occurs in finding that wind factor. Once this is known, it is a simple matter to calculate the ground speed (see section 2.1 above and USAFETAC/TN-80/001). The ground speed information is important to the user's simulation because from it the adjusted flying time can be calculated. The user's simulation then has a timing factor to use for mission planning, war gaming, judging contingency response capabilities, and other applications.

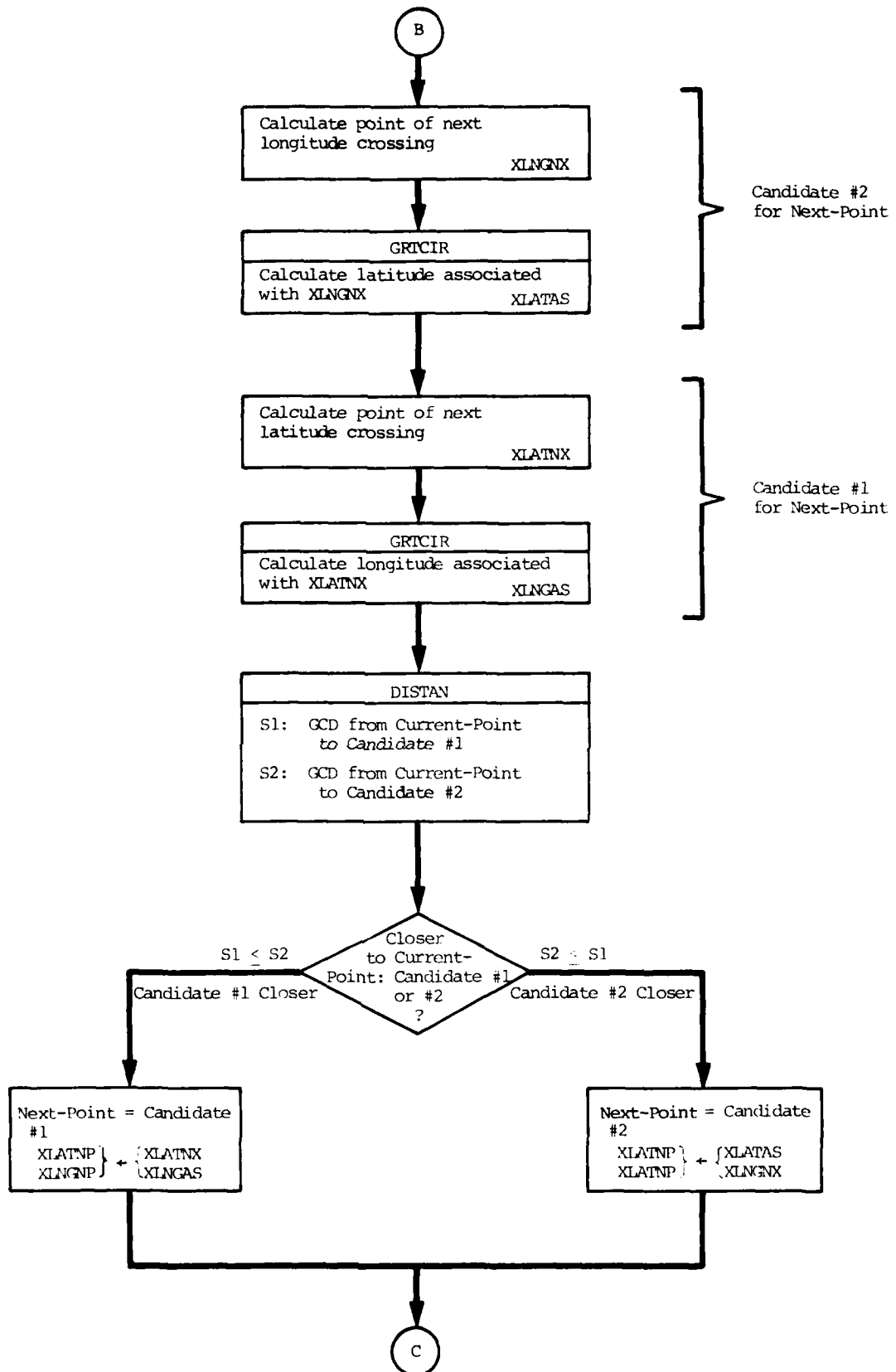
3.5 Recovery and Error Correction Procedures. To assist in debugging efforts, considerable use of the Honeywell Series 6000 FORTRAN Execution Error Monitor (FXEM) has been made. Whenever FXEM error 61 appears, there is a problem in the wind factor routines. A plain text diagnostic, indicating the nature of the problem, will appear on SYSOUT. Presently, FXEM error 61 is set to abort execution of the main program. By a call to one of the FXEM initializing routines, the user may set the disposition of FXEM error 61 to "continue" rather than "abort." No explicit restart features have been incorporated into the WFSM. Such procedures would not be feasible in an iterative, space-stepped solution.

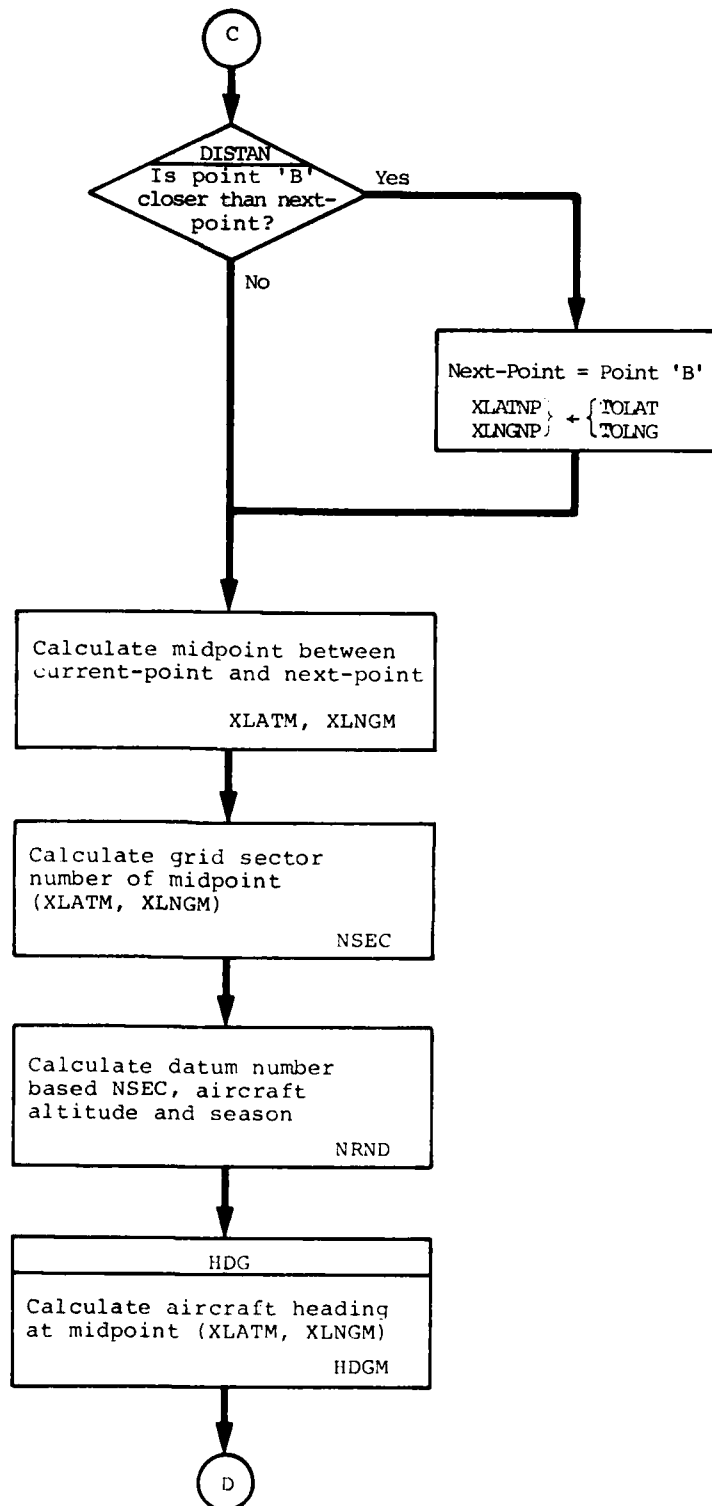
Appendix A

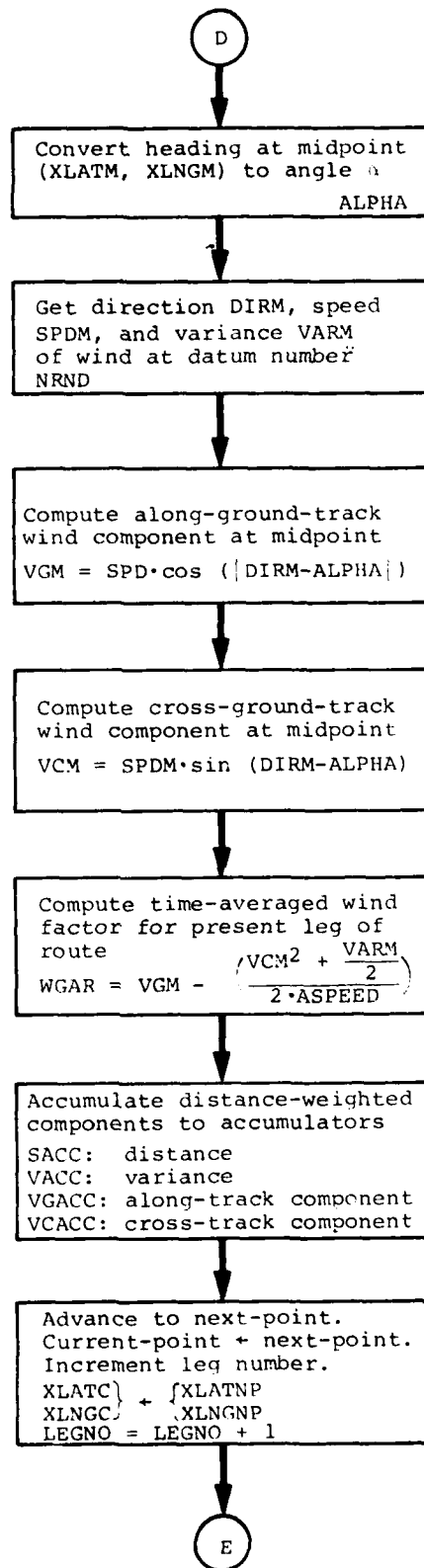
WFSM FLOW CHART

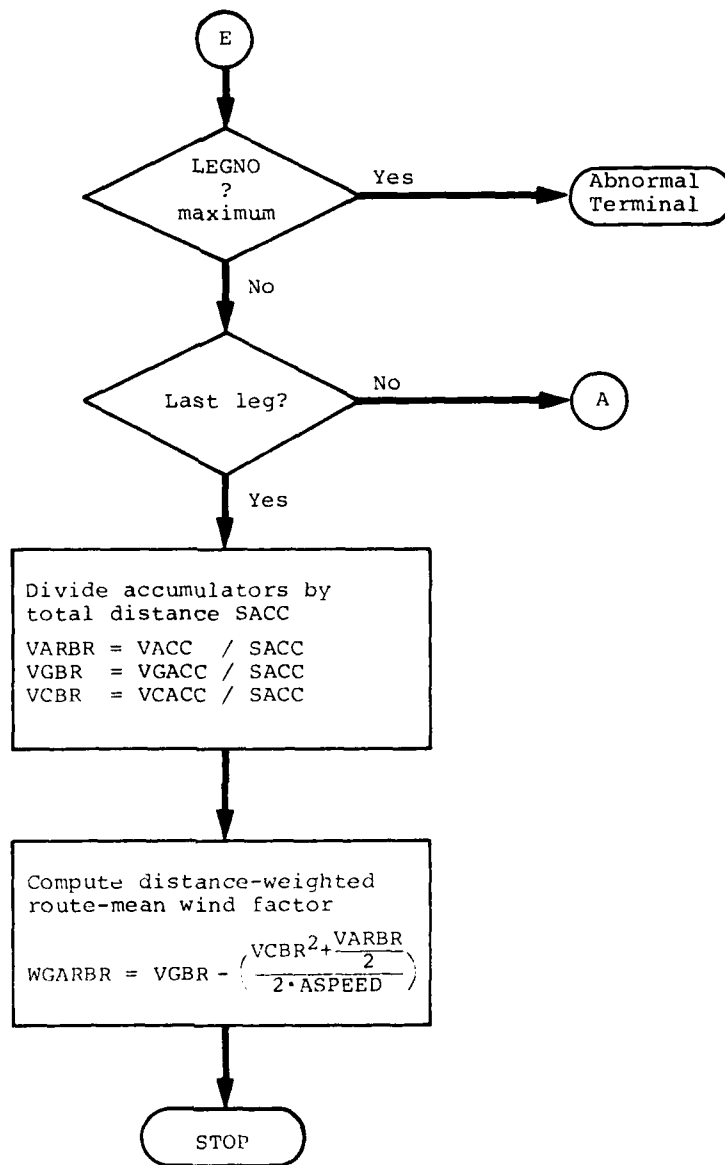












Appendix B

WFSM PROGRAM LISTING

```

1  CENRWND ENROUTE WIND/R. C. WHITON/27 FEB 1979
2  C
3      SUBROUTINE ENRWND (FRMLAT, FRMLNG, TOLAT, TOLNG, JULDAT, GMT,
4      &      FCHRS, IALT, IOPTN, ASPEED, GSPEED)
5  C
6  C*****
7  C*
8  C*  PROGRAM ID-      ENRWND
9  C*  MET ANALYST-    MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
10 C*  SYS ANALYST-    MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
11 C*  PROGRAMMER-     MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
12 C*
13 C*  CREATED ON-     27 FEB 1979          PROJECT- 192301
14 C*
15 C*  DESCRIPTION-    THIS SUBROUTINE SUBPROGRAM CALCULATES CLIMATOLOGI-
16 C*                  CAL WIND FACTORS BY SAWYER'S EQUIVALENT HEADWIND
17 C*                  TECHNIQUE FOR REASONABLY ARBITRARY GREAT CIRCLE
18 C*                  ROUTES AT SPECIFIED CONSTANT ALTITUDES FOR ANY OF
19 C*                  FOUR SEASONS OF THE YEAR IN ANY OF FOUR MODES OR
20 C*                  WIND OPTIONS (CALM WIND, SIMULATED WIND, MEAN
21 C*                  WIND, AND 90% WORST WIND). CLIMB AND DESCENT
22 C*                  WINDS ARE NOT PROVIDED FOR. TEMPERATURES AND
23 C*                  AIRCRAFT PERFORMANCE/FUEL CONSUMPTION ARE NOT PRO-
24 C*                  VIDED FOR.
25 C*
26 C*                  THE SUBPROGRAM NAVIGATES A SIMULATED AIRCRAFT OVER
27 C*                  A GREAT CIRCLE ROUTE FROM POINT 'A' (FRMLAT, FRMLNG)
28 C*                  TO POINT 'B' (TOLAT, TOLNG) IN A GLOBAL (LATITUDE/
29 C*                  LONGITUDE ORIENTED) GRID SYSTEM WHOSE RESOLUTION IS
30 C*                  15 DEGREES LATITUDE BY 30 DEGREES LONGITUDE. THE
31 C*                  SUBROUTINE DETERMINES WHAT GRID SECTOR THE AIRCRAFT
32 C*                  IS IN AND THE LENGTH OF THE AIRCRAFT'S PATH THROUGH
33 C*                  THE SECTOR. THEN THE ROUTINE CONSULTS THE WIND DATA
34 C*                  BASE FOR THE PARTICULAR GRID SECTOR, FLIGHT LEVEL
35 C*                  AND SEASON TO OBTAIN SECTOR-MEAN WIND DIRECTION,
36 C*                  SPEED AND VARIANCE. CROSS-TRACK AND ALONG-TRACK
37 C*                  COMPONENTS OF THE WIND ARE COMPUTED. THESE ARE
38 C*                  WEIGHTED BY THE LENGTH OF THE AIRCRAFT'S TRACK
39 C*                  THROUGH THE SECTOR. AT THE END OF THE SIMULATED
40 C*                  FLIGHT, A DISTANCE-WEIGHTED, ROUTE-MEAN CROSS-TRACK
41 C*                  COMPONENT, ALONG-TRACK COMPONENT AND VARIANCE IS
42 C*                  COMPUTED. FROM THESE, A ROUTE-MEAN WIND FACTOR IS
43 C*                  COMPUTED AND ADDED TO THE AIR SPEED TO OBTAIN
44 C*                  GROUND SPEED. LATITUDE/LONGITUDE OF POINTS 'A'
45 C*                  AND 'B,' THE ALTITUDE OF THE AIRCRAFT, THE JULIAN
46 C*                  DATE AND GMT OF THE FLIGHT, AND THE AIRCRAFT'S
47 C*                  AIR SPEED ARE GIVEN. THE GROUND SPEED IS CALCULATED
48 C*                  BASED ON WIND FACTOR. WIND FACTOR CALCULATIONS
49 C*                  CAN BE MADE IN ANY OF FOUR MODES--CALM WIND,
50 C*                  SIMULATED WIND, 90% WORST WIND, AND MEAN WIND.
51 C*                  IN THE CALM WIND CASE, THE ROUTE-MEAN WIND FACTOR
52 C*                  IS SIMPLY SET TO ZERO, WITH NO NAVIGATION OF THE
53 C*                  AIRCRAFT OR ACCUMULATION OF DISTANCE-WEIGHTED
54 C*                  WIND COMPONENTS BEING NECESSARY. THE SIMULATED
55 C*                  WIND CASE IS RESERVED FOR FUTURE DEVELOPMENT, IN
56 C*                  WHICH A VARIABLE WIND MAY BE IMPLEMENTED SUCH
57 C*                  THAT THE WIND FACTOR NEED NOT BE IDENTICALLY THE
58 C*                  SAME EVERY TIME THE ROUTE IS FLOWN. PRESENTLY,
59 C*                  THE SIMULATED WIND DEFAULTS TO THE MEAN WIND.
60 C*                  IN THE MEAN WIND CASE, THE MEAN WIND IS DEVELOPED

```

61	C*	
62	C*	
63	C*	
64	C*	
65	C*	DIRECTLY FROM THE MEAN WIND DATA BASE. IN THE 90% WORST CASE, THIS MEAN WIND FACTOR IS STATISTICALLY ADJUSTED TO A VALUE SUCH THAT ONLY 10% OF FLIGHTS OVER THAT ROUTE WILL EXPERIENCE A WORSE WIND FACTOR (10% RISK).
66	C*	
67	C*	
68	C*	THE USER DICTATES WHETHER HE WANTS CALM, SIMULATED, MEAN OR 90% WORST WINDS BY SPECIFYING A WIND OPTION (IOPTN) FROM 0 TO 3 (SEE TABLE IN PART I OF SUBPROGRAM).
69	C*	THE USER SPECIFIES WHICH OF THE FOUR SEASONS OF THE YEAR HE WANTS WINDS FOR BY STIPULATING A JULIAN DATE (JULDAT) FROM 1 TO 366. THE TABLE FOR CONVERSION OF JULIAN DATE TO SEASON OF THE YEAR IS PROVIDED IN PART III OF THE SUBPROGRAM.
70	C*	THE USER SPECIFIES THE ALTITUDE OF THE AIRCRAFT BY SETTING THE FLAG IALT TO 1 FOR 25,000 FT AND TO 2 FOR 35,000 FT.
71	C*	THE USER SPECIFIES HIS AIRSPEED IN KNOTS IN THE INPUT VARIABLE ASPEED AND RECEIVES CALCULATED GROUND SPEED IN KNOTS IN THE OUTPUT VARIABLE GSPEED.
72	C*	
73	C*	
74	C*	ARCHITECTURE OF THE SUBPROGRAM IS SUCH THAT PART II HANDLES THE CALM WIND CASE, PART III THE MEAN WIND, PART IV THE 90% WORST WIND, AND PART V THE SIMULATED WIND. PART I ACCOMPLISHES BRANCHING TO PARTS II-V.
75	C*	PART VI HANDLES WRAPUP AND NORMAL TERMINATION. PART VII HANDLES ABNORMAL TERMINATION.
76	C*	
77	C*	
78	C*	
79	C*	
80	C*	
81	C*	
82	C*	
83	C*	
84	C*	
85	C*	
86	C*	
87	C*	
88	C*	
89	C*	METHOD-
90	C*	THE METHODS USED BREAK NEATLY INTO TWO CATEGORIES, (1) GEODESY AND NAVIGATION, AND (2) METEOROLOGY.
91	C*	
92	C*	(1) GEODESY AND NAVIGATION...
93	C*	
94	C*	
95	C*	THE AIRCRAFT IS NAVIGATED ALONG A GREAT CIRCLE ROUTE BETWEEN POINTS 'A' AND 'B' BY MOVING FROM A POINT CALLED THE 'CURRENT-POINT' (XLATC, XLNGC) TO A POINT CALLED THE 'NEXT-POINT' (XLATNP, XLNGNP). MOST OF THE COMPUTATION OF THE NEXT-POINT IS DONE IN GLOBAL (LATITUDE/LONGITUDE) COORDINATES.
96	C*	
97	C*	
98	C*	
99	C*	
100	C*	
101	C*	
102	C*	FOR ANY GIVEN CURRENT-POINT, THE NEXT-POINT IS EITHER THE POINT WHERE THE GREAT CIRCLE ROUTE CROSSES A LATITUDE GRID LINE (0, 15, 30, ... DEG) OR THE POINT WHERE THE ROUTE CROSSES A LONGITUDE GRID LINE (0, 30, 60, ... DEG), WHICHEVER IS CLOSER TO THE CURRENT-POINT.
103	C*	
104	C*	
105	C*	
106	C*	
107	C*	
108	C*	
109	C*	AT THE CURRENT-POINT, THE AIRCRAFT'S HEADING CAN BE CALCULATED VIA THE HEADING FORMULA (FUNCTION HDG). KNOWLEDGE OF THE LOCATION OF THE CURRENT-POINT AND THE AIRCRAFT'S HEADING AT THAT POINT DICTATES THE NEXT LATITUDE CROSSING AND THE NEXT LONGITUDE CROSSING. CANDIDATE #1 FOR NEXT-POINT IS THE POINT OF NEXT LATITUDE CROSSING XLATNX (AND ASSOCIATED LONGITUDE XLNGAS). CANDIDATE #2 IS THE POINT OF NEXT LONGITUDE CROSSING XLNGNX (AND ASSOCIATED LATITUDE XLATAS).
110	C*	
111	C*	
112	C*	
113	C*	
114	C*	
115	C*	
116	C*	
117	C*	
118	C*	
119	C*	
120	C*	CALCULATION OF LONGITUDE XLNGAS ASSOCIATED WITH A GIVEN LATITUDE XLATNX IS DONE ITERATIVELY BY SUBROUTINE GRTCIR, WHICH SOLVES THE EQUATION OF A GREAT CIRCLE. NEWTON'S ITERATIVE METHOD WITH AN EXACT SPHERICAL TRIANGLE FIRST GUESS IS USED. VIRTUALLY ALL OF THE COMPLEXITY OF SUBROUTINES GRTCIR AND ENRWD IS ATTRIBUTABLE IN ONE WAY OR ANOTHER TO THIS PROBLEM OF SOLVING THE EQUATION OF A GREAT CIRCLE FOR LONGITUDE, GIVEN THE LATITUDE.
121	C*	
122	C*	
123	C*	
124	C*	
125	C*	
126	C*	
127	C*	
128	C*	

```

129 C*
130 C*
131 C*
132 C*
133 C*
134 C*
135 C*
136 C*
137 C*
138 C*
139 C*
140 C*
141 C*
142 C*
143 C*
144 C*
145 C*
146 C*
147 C*
148 C*
149 C*
150 C*
151 C*
152 C*
153 C*
154 C*
155 C*
156 C*
157 C*
158 C*
159 C*
160 C*
161 C*
162 C*
163 C*
164 C*
165 C*
166 C*
167 C*
168 C*
169 C*
170 C*
171 C*
172 C*
173 C*
174 C*
175 C*
176 C*
177 C*
178 C*
179 C*
180 C*
181 C*
182 C*
183 C*
184 C*
185 C*
186 C*
187 C*
188 C*
189 C*
190 C*
191 C*
192 C*
193 C*
194 C*
195 C*
196 C*

```

CALCULATION OF LATITUDE XLATAS ASSOCIATED WITH A GIVEN LONGITUDE XLNGX IS DONE DETERMINISTICALLY BY SUBROUTINE GRTCLR. NO DIFFICULTY IS ENCOUNTERED HERE.

CANDIDATES FOR NEXT-POINT ARE THEN AS FOLLOWS...

```

    CANDIDATE #1 ..... (XLATNX, XLNGAS)
    CANDIDATE #2 ..... (XLATAS, XLNGNX)

```

WITH CANDIDATES 1 AND 2 IDENTIFIED, SUBROUTINE DISTAN IS THEN USED TO CALCULATE GREAT CIRCLE DISTANCES AS FOLLOWS...

```

    S1 ..... GCD FROM CURRENT-POINT TO CAND #1
    S2 ..... GCD FROM CURRENT-POINT TO CAND #2

```

THE CANDIDATE WITH THE SHORTER S-DISTANCE IS SELECTED AS THE NEXT-POINT.

THE CURRENT-POINT CAN THEN BE THOUGHT OF AS THE POINT WHERE THE AIRCRAFT ENTERS THE GRID SECTOR, AND THE NEXT-POINT AS THE POINT WHERE THE AIRCRAFT DEPARTS THE SECTOR. THE DISTANCE SNP BETWEEN CURRENT-POINT AND NEXT-POINT IS THE DISTANCE BY WHICH THE WIND COMPONENTS AND VARIANCE ARE WEIGHTED IN COMPUTING THE ROUTE-MEAN WIND FACTOR.

2. METEOROLOGY...

WITH CURRENT-POINT AND NEXT-POINT ESTABLISHED, AN APPROXIMATE MIDPOINT (XLATM, XLNGM) BETWEEN THEM IS ESTABLISHED. THE NUMBER NSEC INDICATING WHICH OF THE 108 GRID SECTORS THROUGH THE AIRCRAFT IS PASSING THROUGH IS THEN CALCULATED, BASED ON (XLATM, XLNGM). SEE DESCRIPTION OF THE GRID SYSTEM BELOW. SINCE FOUR SEASONS AND TWO FLIGHT LEVELS, AS WELL AS 108 GRID SECTORS, ARE PROVIDED FOR, A DATUM-NUMBER NRND IS OBTAINED FROM THE GRID SECTOR NUMBER NSEC (1-108), THE ALTITUDE INDEX IALT (1-2) AND THE SEASON INDEX ISEASN (1-4) AS FOLLOWS...

```

    NRND = (ISEASN-1) * (TOTAL-NUMBER ALTITUDES) *
           (TOTAL-NUMBER-GRID-SECTORS)
           + (IALT-1) * (TOTAL-NUMBER-GRID-SECTORS)
           + NSEC

```

WITH THE DATUM-NUMBER NRND CALCULATED, THE DATA BASE (SEE DESCRIPTION BELOW) IS CONSULTED TO OBTAIN SECTOR-MEAN WIND DIRECTION DIRM (RADIAN, IN BETA-ANGLE FORM, AS DESCRIBED BELOW), SPEED SPDM (KNOTS) AND VARIANCE VARM (KNOTS**2)...

```

    DIRM = DIR(NRND)
    SPDM = SPD(NRND)
    VARM = VAR(NRND)

```

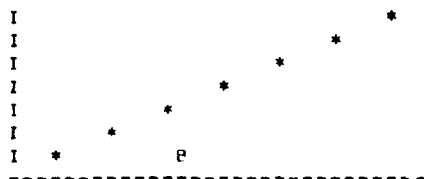
THE AIRCRAFT HEADING AT (XLATM, XLNGM) IS COMPUTED BY MEANS OF FUNCTION HDG. THE HEADING HDGM IS COMPUTED IN BETA-ANGLE FORMAT (SEE BELOW) AND IN THAT FORMAT IS REPRESENTED BY THE ANGLE ALPHA.

ALPHA, THE WIND DIRECTION DIRM, AND THE WIND SPEED SPDM ARE USED TO CALCULATE THE ALONG-GROUND-TRACK WIND COMPONENT VGM AND THE CROSS-GROUND-TRACK WIND COMPONENT VCM...

```

197 C*      VGM = SPDM * COS(ABS(DIRM - ALPHA))
198 C*      VCM = SPDM * SIN(   DIRM - ALPHA )
199 C*
200 C*      A TIME-AVERAGED WIND FACTOR FOR THE PRESENT LEG IS
201 C*      THEN COMPUTED USING SAWYER'S EQUIVALENT HEADWIND
202 C*      FORMULA...
203 C*
204 C*      WBAR = VGM - ((VCM**2 + VARM/2.0) /
205 C*      (2.0 * ASPEED))
206 C*
207 C*      BUT NO FURTHER USE IS MADE OF WBAR.
208 C*
209 C*      THE QUANTITIES VGM, VCM AND VARM ARE MULTIPLIED BY
210 C*      THE SECTOR- OR LEG-LENGTH SNP, AND THE PRODUCT IS
211 C*      ACCUMULATED IN VGACC, VCACC, AND VACC.
212 C*
213 C*      THEN THE NEXT-POINT BECOMES THE CURRENT-POINT, A NEW
214 C*      NEXT-POINT IS CALCULATED, AND THE PROCESS IS RE-
215 C*      PEATED FOR THE NEXT LEG OF THE SIMULATED FLIGHT.
216 C*
217 C*      AFTER THE LAST LEG OF THE FLIGHT HAS BEEN PROCESSED,
218 C*      THE ACCUMULATORS VGACC, VCACC AND VACC ARE DIVIDED
219 C*      BY THE TOTAL ROUTE LENGTH. THE RESULTING DISTANCE-
220 C*      WEIGHTED ALONG-TRACK, CROSS-TRACK AND VARIANCE QUAN-
221 C*      TITIES ARE THEN USED IN THE SAWYER EQUIVALENT HEAD-
222 C*      WIND FORMULA GIVEN ABOVE TO OBTAIN THE DISTANCE-
223 C*      WEIGHTED ROUTE-MEAN WIND FACTOR WBARBR.
224 C*
225 C*      IF THE 90% WORST WIND FACTOR HAS BEEN SELECTED,
226 C*      WBARBR IS STATISTICALLY ADJUSTED TO THE 90% WORST
227 C*      VALUE BY SAWYER'S TECHNIQUE, IMPLEMENTED IN PART IV
228 C*      OF SUBROUTINE ENRWND.
229 C*
230 C*      AS PART OF THE WRAPUP ACTIONS IN PART VI OF SUBROU-
231 C*      TINE ENRWND, THE ROUTE-MEAN WIND FACTOR WBARBR IS
232 C*      ADDED TO THE AIR SPEED ASPEED TO GET A GROUND SPEED
233 C*      GSPEED FOR RETURN TO THE PROGRAM CALLING SUBROUTINE
234 C*      ENRWND.
235 C*
236 C*      THE METEOROLOGICAL TECHNIQUE USED IS REFERRED TO AS
237 C*      SAWYER'S EQUIVALENT HEADWIND TECHNIQUE AND IS DOCU-
238 C*      MENTED IN AWS-TR-77-267, GUIDE FOR APPLIED CLIMA-
239 C*      TOLOGY.
240 C*
241 C*      BETA-ANGLE FORMAT...
242 C*
243 C*      THE WIND DIRECTION DD CAN BE EXPRESSED IN THE
244 C*      FORM OF A BETA-ANGLE B...
245 C*
246 C*      I
247 C*      I
248 C*      I
249 C*      I
250 C*      I
251 C*      I
252 C*      I *
253 C*      I
254 C*      -----
255 C*
256 C*      THE BETA ANGLE DESCRIBES THE DIRECTION TOWARD
257 C*      WHICH THE WIND IS BLOWING IN TERMS OF RADIANS
258 C*      COUNTERCLOCKWISE FROM EASTWARD. CONVERSIONS
259 C*      ARE AS FOLLOWS...

```



		DIRECTION TOWARD WHICH	BETA-ANGLE (DEGREES)	BETA-ANGLE (RADIAN)
259	C*			
260	C*			
261	C*			
262	C*			
263	C*	300.0	-30.0	-0.524
264	C*	270.0	0.0	0.000
265	C*	240.0	30.0	0.524
266	C*	180.0	90.0	1.571
267	C*			
268	C*			
269	C*	GRID		
270	C*	SYSTEM-		
271	C*	THE GRID SYSTEM IS IN GLOBAL (LATITUDE/LONGITUDE)		
272	C*	COORDINATES. WITH A RESOLUTION OF (15 DEG LAT X		
273	C*	30 DEG LONG). THE GRID SYSTEM EXTENDS FROM 75 DEG		
274	C*	N TO 60 DEG S (9 ROWS) AND GLOBALLY IN LONGITUDE		
275	C*	(12 COLUMNS). HENCE, THERE ARE 108 GRID SECTORS.		
276	C*	COLUMN 1 EXTENDS FROM 30 DEG W TO 0 DEG (GREEN-		
277	C*	WICH). COLUMN 2 FROM 0 DEG TO 30 DEG E, ETC. ROW		
278	C*	1 EXTENDS FROM 75 DEG N TO 60 DEG N, ROW 2 FROM		
279	C*	60 DEG N TO 45 DEG N, ETC. SUCH A GRID SYSTEM IS		
280	C*	TOO COARSE FOR CALCULATION OF OPERATIONAL WIND		
281	C*	FACTORS BUT IS SUITABLE FOR PURPOSES SUCH AS		
282	C*	SIMULATION.		
283	C*	DATA		
284	C*	BASE-		
285	C*	THE SIOP WINDS WERE USED TO CREATE THE DATA BASE		
286	C*	EMPLOYED BY SUBROUTINE ENRWIND. SIOP WINDS HAVE		
287	C*	A (5 DEG X 5 DEG) RESOLUTION. ALL SIOP WINDS		
288	C*	WITHIN EACH OF THE PRESENT (15 DEG X 30 DEG) GRID		
289	C*	SYSTEM WERE EQUALLY WEIGHTED AND AVERAGED TO		
290	C*	OBTAIN A SINGLE WIND FOR EACH OF THE 108 GRID		
291	C*	SECTORS. TWO FLIGHT LEVELS (25,000 FT AND		
292	C*	35,000 FT) AND FOUR SEASONS. THERE ARE THUS A		
293	C*	TOTAL OF 864 WINDS IN THE DATA BASE. EACH HAS A		
294	C*	WIND DIRECTION DIR(NRND) (RADIAN), BETA-ANGLE		
295	C*	FORMAT), SPEED SPD(NRND) (KNOTS), AND VARIANCE		
296	C*	VAR(NRND) (KNOTS**2). THESE WINDS ARE STORED		
297	C*	SEQUENTIALLY IN THE FOLLOWING ORDER...		
298	C*	WINTER SEASON		
299	C*	FL 250		
300	C*	108 SECTORS		
301	C*	FL 350		
302	C*	108 SECTORS		
303	C*	SPRING SEASON		
304	C*	FL 250		
305	C*	108 SECTORS		
306	C*	FL 350		
307	C*	108 SECTORS		
308	C*			
309	C*	ETC.		
310	C*			
311	C*	THE WINDS ARE IN THE ARRAYS DIR(870), SPD(870)		
312	C*	AND VAR(870). THESE ARRAYS ARE MEMBERS OF THE		
313	C*	LABELLED COMMON BLOCK WEA...		
314	C*			
315	C*	COMMON /WEA/ DIR(870), SPD(870), VAR(870)		
316	C*			
317	C*	AN INITIALIZING ROUTINE IS NEEDED TO READ THE		
318	C*	WIND DATA FROM AN 864-RECORD FILE ON DISK OR		
319	C*	TAPE TO THE WIND ARRAYS. THIS INITIALIZATION		
320	C*	MUST BE EXECUTED BEFORE FIRST CALL TO SUBROUTINE		
321	C*	ENRWIND.		
322	C*			
323	C*	LIMITATIONS-		
324	C*	PRESENTLY, THE WIND FACTOR SIMULATION MODEL CAN-		
325	C*	NOT PRODUCE A 'SIMULATED' WIND HAVING A DAY-TO-		
326	C*	DAY VARIABILITY. HENCE, WITHIN ANY GIVEN SEASON		
		AND FOR ANY GIVEN ALTITUDE, A PARTICULAR ROUTE		

327	C*	
328	C*	
329	C*	
330	C*	WILL ALWAYS EXPERIENCE THE SAME WIND FACTOR, REGARDLESS OF THE PASSAGE OF TIME. REQUESTS FOR THE SIMULATED WIND WILL DEFAULT TO THE MEAN WIND.
331	C*	
332	C*	
333	C*	FURTHERMORE, THE PRESENT MODEL IS INCAPABLE OF PRODUCING A 'FORECAST' WIND FACTOR. REQUESTS FOR A FORECAST WIND WILL DEFAULT TO THE MEAN WIND.
334	C*	
335	C*	
336	C*	
337	C*	TEMPERATURE, AIRCRAFT PERFORMANCE, FUEL CONSUMPTION, AND OTHER FACTORS CONSIDERED BY TYPICAL FLIGHT PLANNING MODELS ARE NOT INCLUDED IN THE PRESENT MODEL, WHICH DEALS ONLY WITH WIND.
338	C*	
339	C*	
340	C*	
341	C*	
342	C*	OPERATIONS NORTH OF 75 DEG N OR SOUTH OF 60 DEG S ARE NOT PERMITTED.
343	C*	
344	C*	
345	C*	
346	C*	THE COARSENESS OF THE PRESENT 15 DEG LATITUDE BY 30 DEG LONGITUDE GRID SYSTEM PRECLUDES USE OF THE MODEL FOR CALCULATION OF OPERATIONALLY REALISTIC WIND FACTORS UNLESS CHANGES ARE MADE. RATHER SIMPLE MODIFICATIONS TO THE EXISTING DATA BASE AND GRID SYSTEM, ALONG WITH CORRESPONDING CHANGES TO THE SOFTWARE, CAN ELIMINATE THIS LIMITATION.
347	C*	
348	C*	
349	C*	
350	C*	
351	C*	
352	C*	
353	C*	SPECIFICALLY, THE FOLLOWING RESTRICTIONS MUST BE ADHERED TO...
354	C*	
355	C*	
356	C*	A. NO ROUTE SHOULD BE FLOWN DIRECTLY OVER EITHER POLE.
357	C*	
358	C*	
359	C*	B. NEITHER POLE SHOULD SERVE AS A POINT 'A' (TAKEOFF) OR A POINT 'B' (LANDING).
360	C*	
361	C*	
362	C*	C. CIRCUMFERENTIAL OR ROUND ROBIN FLIGHTS IN WHICH POINT 'A' AND 'B' COINCIDE WILL BE ABORTED. BREAK SUCH FLIGHTS INTO SMALLER SEGMENTS.
363	C*	
364	C*	
365	C*	
366	C*	
367	C*	D. SEMI-CIRCUMFERENTIAL FLIGHTS IN WHICH POINT 'B' IS EXACTLY OPPOSITE POINT 'A' THROUGH THE CENTER OF THE EARTH WILL ALSO ABORT. BREAK SUCH FLIGHTS INTO SMALLER SEGMENTS.
368	C*	
369	C*	
370	C*	
371	C*	
372	C*	
373	C*	E. ROUTES OR SEGMENTS OF ROUTES FLOWN DIRECTLY NORTH OR DIRECTLY SOUTH ALONG A LONGITUDE LINE WILL ALSO ABORT.
374	C*	
375	C*	
376	C*	F. AT PRESENT, THE MODEL ALLOWS FLIGHTS ONLY AT ALTITUDES 25,000 FT AND 35,000 FT.
377	C*	
378	C*	
379	C*	
380	C*	G. ALL DEPARTURE AND DESTINATION POINTS MUST LIE WITHIN THE LATITUDINAL RANGE FROM 75 DEG N TO 60 DEG S.
381	C*	
382	C*	
383	C*	
384	C*	RESTRICTIONS F AND G WERE IMPOSED IN ORDER TO REDUCE THE CORE STORAGE REQUIREMENTS OF THE MODEL.
385	C*	
386	C*	REFERENCES-
387	C*	1. WHITON, R. C., AND P. L. HEROD, 1980: WIND FACTOR SIMULATION MODEL: MODEL DESCRIPTION, USAFETAC/TN-80/001.
388	C*	
389	C*	
390	C*	2. HEROD, P. L., AND R. C. WHITON, 1980: WIND FACTOR SIMULATION MODEL: USER'S MANUAL, USAFETAC/TN-80/002.
391	C*	
392	C*	

```

393 C*
394 C* INPUT- FRMLAT = LATITUDE OF POINT 'A' (TAKEOFF) IN
395 C* DECIMAL DEGREES, NEGATIVE SOUTH
396 C* FRMLNG = LONGITUDE OF POINT 'A' (TAKEOFF) IN
397 C* DECIMAL DEGREES, NEGATIVE EAST
398 C* TOLAT = LATITUDE OF POINT 'B' (LANDING) IN
399 C* DECIMAL DEGREES, NEGATIVE SOUTH
400 C* TOLNG = LONGITUDE OF POINT 'B' (LANDING) IN
401 C* DECIMAL DEGREES, NEGATIVE EAST
402 C* JULDAT = JULIAN BASE DATE OF WIND FACTOR RE-
403 C* QUEST, INTEGER, VALUES 1 - 366
404 C* GMT = GREENWICH BASE TIME OF WIND FACTOR
405 C* REQUEST, VALUE 0.0 - 24.0 HRS
406 C* FCHRS = FORECAST HOURS AHEAD, VALUE GREATER
407 C* THAN OR EQUAL TO 0.0 HRS
408 C* IALT = ALTITUDE INDEX, 1 FOR 25,000 FT AND
409 C* 2 FOR 35,000 FT
410 C* IOPTN = WIND OPTION OR MODE, INTEGER, 0 FOR
411 C* CALM WIND, 2 FOR 90% WORST WIND, AND
412 C* 3 FOR MEAN WIND. IN THE FUTURE, 1
413 C* WILL BE IMPLEMENTED AS THE SIMULATED
414 C* WIND BUT PRESENTLY DEFAULTS TO THE
415 C* MEAN WIND
416 C* ASPEED = ROUTE-MEAN AIRCRAFT AIR SPEED IN KNOTS
417 C*
418 C* OUTPUT-
419 C* GSPEED = ROUTE-MEAN AIRCRAFT GROUND SPEED IN
420 C* KNOTS
421 C*
422 C* SYSTEM SUB-
423 C* PROGRAMS
424 C* USED- SORT, SIN, COS, FXEM
425 C*
426 C* USER SUB-
427 C* PROGRAMS
428 C* USED- GRTCIR, DISTAN, SPHGLO, HDG, RRLAT, BRLNG
429 C*
430 C* ESTIMATED
431 C* CPU TIME- A LONG ROUTE CONSISTING OF 8 LEGS AND REQUIRING
432 C* AN EQUATOR CROSSING TAKES 0.1 - 0.5 SEC CPU TIME
433 C* ON A HONEYWELL 6080 COMPUTER.
434 C*
435 C* STORAGE
436 C* REQUIREMENTS- PROCEDURE PLUS DATA OCCUPY 1130 WORDS OF CORE
437 C* STORAGE. COMMON BLOCK /WEA/ ADDS 2610 WORDS.
438 C* TOTAL CORE IS 3740 WORDS FOR THIS SUBROUTINE.
439 C*
440 C* PROGRAM
441 C* UPDATES- NONE
442 C*
443 C*
444 C* COMMON /WEA/ DIR(870), SPD(870), VAR(870)
445 C* COMMON /DBG/ NDEBUG
446 C*
447 C* DESCRIPTION OF SPHERICAL GRID. LATITUDE GRID LINES ARE SPACED
448 C* EVERY DELTA DEGREES OF LATITUDE. LONGITUDE GRID LINES ARE
449 C* SPACED EVERY DELTA DEGREES OF LONGITUDE.
450 C*
451 C* DATA DELTA/15.0/, DELTLO/30.0/
452 C*
453 C* ANGLE CONVERSION FACTORS. MULTIPLIER DTOR CONVERTS DEGREES TO
454 C* RADIANS. RTOD CONVERTS RADIANS TO DEGREES.
455 C*
456 C* DATA DTOR/0.01745329/, RTOD/57.295780/
457 C*
458 C* LEGMAX IS MAXIMUM PERMITTED NUMBER OF LEGS IN ROUTE, USED TO
459 C* DETECT RUNAWAY ROUTES.
460 C*

```

```

461 DATA LEGMAX/50/
462 C
463 C *****
464 C * PART I *
465 C * BRANCH *
466 C *****
467 C
468 C BRANCH TO APPROPRIATE PART OF SUBROUTINE, BASED ON VALUE OF IOPTN
469 C FLAG.
470 C
471 C IOPTN FUNCTION SUBROUTINE PARTS
472 C
473 C 0 CALM WIND PART II
474 C 1 SIMULATED WIND PART V
475 C 2 90% WORST WIND PARTS III & IV
476 C 3 MEAN WIND PART III
477 C
478 C NOTE THAT PREPARATION OF THE 90% WORST WIND REQUIRES HAVING PRE-
479 C VIQUSLY ESTABLISHED THE MEAN WIND.
480 C
481 C NOTE THAT IN THIS VERSION OF THE SUBROUTINE, THE SIMULATED WIND
482 C OPTION IS NOT OPERATIONAL. CALLS FOR THE SIMULATED WIND WILL DE-
483 C FAULT TO THE MEAN WIND.
484 C
485 C IF (IOPTN .LT. 0 .OR. IOPTN .GT. 3) GO TO 7020
486 C IOPTNP = IOPTN + 1
487 C GO TO (2000, 5000, 3000, 3000), IOPTNP
488 C
489 C *****
490 C * PART II *
491 C * CALM WIND *
492 C *****
493 C
494 C IN THE CALM WIND CASE, THE WIND FACTOR IS ZERO, AND GROUND SPEED
495 C IS EQUAL TO AIR SPEED IN KNOTS. ROUTE-AVERAGED WIND FACTOR WBARBR
496 C IS ZFR0.
497 C
498 C 2000 WBARBR = 0.0
499 C
500 C GO TO WRAPUP AND TERMINATION.
501 C
502 C GO TO 6000
503 C
504 C *****
505 C * PART III *
506 C * MEAN WIND *
507 C *****
508 C
509 C INITIALIZE ACCUMULATORS FOR DISTANCE (SACC), VARIANCE (VACC),
510 C ALONG-TRACK WIND COMPONENT (VGACC), AND CROSS-TRACK WIND
511 C COMPONENT (VCACC).
512 C
513 C 3000 SACC = 0.0
514 C VACC = 0.0
515 C VGACC = 0.0
516 C VCACC = 0.0
517 C
518 C INITIALIZE LAST-LEG FLAG LASTLG. WHEN THE CURRENT LEG OF THE
519 C FLIGHT IS THE LAST LEG, LASTLG = 1.
520 C
521 C LASTLG = 0
522 C
523 C TEST POSITION OF POINT 'A' (FRMLAT,FRMLNG) AND POINT 'B' (TOLAT,
524 C TOLNG) FOR LEGAL RANGE OF VALUES IN DEGREES.
525 C
526 C IF (FRMLAT .LT. -90.0 .OR. FRMLAT .GT. 90.0) GO TO 7030
527 C IF (TOLAT .LT. -90.0 .OR. TOLAT .GT. 90.0) GO TO 7030
528 C IF (FRMLNG .LT. -180.0 .OR. FRMLNG .GT. 180.0) GO TO 7030

```

```

529      IF (TOLNG .LT. -180.0 .OR. TOLNG .GT. 180.0) GO TO 7030
530      C
531      C      INITIALIZE INTERMEDIATE POINT COORDINATES.
532      C
533      XLATC = 0.0
534      XLNGC = 0.0
535      THETAC = 0.0
536      PHIC = 0.0
537      C
538      C      INTERPRET THE JULIAN DATE JULDAT IN TERMS OF SEASON OF THE YEAR,
539      C      INDICATED BY THE SEASON INDEX ISEASN. GIVE AN ERROR CONDITION
540      C      FOR ILLEGAL JULDAT VALUES.
541      C
542      C      JULIAN DATES      SEASON      SEASON INDEX
543      C      JULDAT      OF YEAR      ISEASN
544      C
545      C      1 - 59      WINTER      1
546      C      60 - 151      SPRING      2
547      C      152 - 243      SUMMER      3
548      C      244 - 334      FALL      4
549      C      335 - 366      WINTER      1
550      C
551      IF (JULDAT .GT. 366) GO TO 7000
552      IF (JULDAT .GE. 335) GO TO 3065
553      IF (JULDAT .GE. 244) GO TO 3050
554      IF (JULDAT .GE. 152) GO TO 3055
555      IF (JULDAT .GE. 60) GO TO 3060
556      IF (JULDAT .GE. 1) GO TO 3065
557      GO TO 7000
558      3050 ISEASN = 4
559      GO TO 3070
560      3055 ISEASN = 3
561      GO TO 3070
562      3060 ISEASN = 2
563      GO TO 3070
564      3065 ISEASN = 1
565      GO TO 3070
566      C
567      C      CHECK ALTITUDE OPTION IALT FOR LEGAL RANGE.
568      C
569      C      IALT      AIRCRAFT ALTITUDE
570      C
571      C      1      25,000 FT
572      C      2      35,000 FT
573      C
574      3070 IF (IALT .LT. 1 .OR. IALT .GT. 2) GO TO 7010
575      C
576      C      CALCULATE TOTAL GREAT CIRCLE DISTANCE STOT IN NAUTICAL MILES BE-
577      C      TWEEN POINTS 'A' AND 'B.'
578      C
579      STOT = 0.0
580      CALL DISTAN (FRMLAT, FRMLNG, TOLAT, TOLNG, STOT)
581      C
582      C      INITIALIZE GREAT CIRCLE SUBROUTINE. NOTE THAT GRTCIR IS
583      C      CALLED TWICE--ONCE FOR ICONV = 1 AND ONCE FOR ICONV = 2.
584      C
585      ICONV = 1
586      IGUFSS = 0
587      XLNGG = 0.0
588      XEAST = 0.0
589      XNORTH = 0.0
590      XLATC = 0.0
591      XLNGAP = 0.0
592      SAPRIM = 0.0
593      3075 CALL GRTCIR (ICONV, FRMLAT, FRMLNG, TOLAT, TOLNG, XLATC, XLNGC,
594      &      THETAA, PHIA, THETAB, PHIB, QTX, QTY, QTZ, IGUESS, XLNGG,
595      &      XEAST, XNORTH, XLNGAP, SAPRIM, XLATC, THETAC, PHIC)
596      IF (ICONV .EQ. 2) GO TO 3080

```

```

597      ICONV = 2
598      GO TO 3075
599      C
600      C      SET THE LOCATION (XLATC,XLNGC) OF THE CURRENT-POINT TO POINT
601      C      'A' (FRMLAT,FRMLNG). THIS ROUTINE OPERATES WITH TWO PRINCIPAL
602      C      POINTERS, NAMELY THE CURRENT-POINT (XLATC,XLNGC) AND THE NEXT-
603      C      POINT (XLATNP,XLNGNP). COMPUTATIONS IN THIS SUBROUTINE ARE
604      C      PRIMARILY IN TERMS OF LATITUDE AND LONGITUDE. SOME LIMITED
605      C      USE OF COLATITUDE (THETA) AND AZIMUTH (PHI) IS MADE BY
606      C      SUBROUTINE GRTCIR, SO THESE QUANTITIES (THETAC, PHIC) ARE
607      C      MAINTAINED FOR THE CURRENT-POINT (XLATC, XLNGC).
608      C
609      3080 XLATC = FRMLAT
610      XLNGC = FRMLNG
611      ICONVX = 1
612      CALL SPHGLO (ICONVX, XLATC, XLNGC, THETAC, PHIC)
613      C
614      C      INITIALIZE THE LEG COUNTER LEGNO, WHICH IS UNITY FOR THE FIRST
615      C      LEG OF THE FLIGHT.
616      C
617      LEGNO = 1
618      C      IF (NDEBUG .GT. 0) WRITE (6,B000)
619      C8000 FORMAT (I10, T8, 'CURR PT', T21, 'NEXT PT', T34, 'DISTANCE', T44,
620      C      'SEC-', T50, 'ACFT', T55, '-----WIND-----'/IX,
621      C      'LG', T7, 'LAT',
622      C      T14, 'LNG', T20, 'LAT', T27, 'LNG', T33, 'NAUT MILES', T44,
623      C      'TOR', T51, 'HDG', T58, 'DIR', T64, 'SPD', T70, 'VAR', T74,
624      C      'FACTOR'//)
625      C
626      C      BEGIN NAVIGATION LOOP.
627      C
628      C      BASED ON A GREAT CIRCLE BETWEEN THE CURRENT-POINT AND POINT 'B,'
629      C      CALCULATE THE AIRCRAFT'S HEADING HDGC AT THE CURRENT-POINT.
630      C      HEADING IS THE DIRECTION TOWARD WHICH IN DEGREES, 0.0 FOR NORTH-
631      C      WARD, 90.0 FOR EASTWARD, 180.0 FOR SOUTHWARD, AND 270.0 FOR
632      C      WESTWARD.
633      C
634      3100 HDGC = HDG (XLATC, XLNGC, TOLAT, TOLNG)
635      C
636      C      BASED ON THE HEADING HDGC AT THE CURRENT-POINT, DETERMINE THE
637      C      EASTWARD DIRECTION INDICATOR XEAST (POSITIVE FOR EASTWARD
638      C      COMPONENT OF AIRCRAFT MOTION) AND NORTHWARD DIRECTION INDICATOR
639      C      XNORTH (POSITIVE FOR NORTHWARD COMPONENT OF AIRCRAFT
640      C      MOTION). SPECIAL CASES OF CARDINAL HEADINGS ARE HANDLED BY
641      C      STATEMENTS 3110 THROUGH 3137, WHILE GENERAL CASES ARE HANDLED
642      C      BY 3140 THROUGH 3167.
643      C
644      3110 IF (HDGC .EQ. 0.0 .OR. HDGC .EQ. 360.0) GO TO 3120
645      IF (HDGC .EQ. 90.0) GO TO 3125
646      IF (HDGC .EQ. 180.0) GO TO 3130
647      IF (HDGC .EQ. 270.0) GO TO 3135
648      GO TO 3140
649      3120 XEAST = 0.0
650      XNORTH = 1.0
651      GO TO 3170
652      3125 XEAST = 1.0
653      XNORTH = 0.0
654      GO TO 3170
655      3130 XEAST = 0.0
656      XNORTH = -1.0
657      GO TO 3170
658      3135 XEAST = -1.0
659      XNORTH = 0.0
660      3137 GO TO 3170
661      3140 IF (HDGC .LT. 90.0) GO TO 3150
662      IF (HDGC .LT. 180.0) GO TO 3155
663      IF (HDGC .LT. 270.0) GO TO 3160
664      GO TO 3165

```

```

665      3150 XEAST = 1.0
666          XNORTH = 1.0
667          GO TO 3170
668      3155 XEAST = 1.0
669          XNORTH = -1.0
670          GO TO 3170
671      3160 XEAST = -1.0
672          XNORTH = -1.0
673          GO TO 3170
674      3165 XEAST = -1.0
675      3167 XNORTH = 1.0
676      3170 CONTINUE
677      C      IF (NDEBUG .GT. 1) WRITE (6,8010) XEAST, XNORTH
678      C8010 FORMAT (1H , T4, 'XEAST,XNORTH = ', T36, 2F10.4)
679      C
680      C      BASED ON EASTWARD DIRECTION INDICATOR, CALCULATE THE POINT OF NEXT
681      C      LONGITUDE CROSSING (XLNGNX) AND ASSOCIATED LATITUDE (XLATAS) DE-
682      C      TERMINED FROM THE EQUATION OF A GREAT CIRCLE. THIS IS CANDIDATE
683      C      POINT #1. STATEMENTS 3200-3299. DIFFERENT PROCEDURE IS USED,
684      C      DEPENDING ON WHETHER THIS IS THE FIRST LEG OF THE FLIGHT OR
685      C      WHETHER (ON SUBSEQUENT LEGS) THE CURRENT-POINT WAS CANDIDATE #1
686      C      (NCAND = 1) OR CANDIDATE #2 (NCAND = 2).
687      C
688      3200 IF (LEGNO .GT. 1) GO TO 3240
689      C
690      C      PROCEDURE FOR FIRST LEG OF FLIGHT. DETERMINE BRACKETING LONGI-
691      C      TUDE VALUES. MOVE EAST (DECREASING LONGITUDE) IF XEAST .GT.
692      C      0.0. SET XLNGNX.
693      C
694      3210 CALL BRLNG (XLNGC, XLNGLO, XLNGHI)
695      C      IF (NDEBUG .GT. 1) WRITE (6,8020) XLNGLO, XLNGHI
696      C8020 FORMAT (1H , T4, 'XLNGLO,XLNGHI = ', T36, 2F10.4)
697      C      IF (XEAST) 3220,3220,3230
698      3220 XLNGNX = XLNGHI
699          GO TO 3290
700      3230 XLNGNX = XLNGLO
701          GO TO 3290
702      C
703      C      PROCEDURE FOR SUCCEEDING LEGS DEPENDS ON WHETHER CANDIDATE #1 OR
704      C      CANDIDATE #2 WAS CHOSEN LAST TIME.
705      C
706      3240 GO TO (3250,3270), NCAND
707      C
708      C      IF CURRENT-POINT WAS A CANDIDATE #1 (EXACT LONGITUDE), THEN NEXT
709      C      LONGITUDE CROSSING IS THAT LONGITUDE PLUS OR MINUS DELTA-LONGI-
710      C      TUDE (PLUS FOR WESTWARD, MINUS FOR EASTWARD).
711      C
712      3250 XLNGNX = -(XEAST * DELTLO) + XLNGC
713      C
714      C      ADJUST LONGITUDE FOR DATELINE FOLD.
715      C
716      C      IF (XLNGNX .GT. 180.0) GO TO 3260
717      C      IF (XLNGNX .LT. -180.0) GO TO 3265
718      C      GO TO 3290
719      3260 XLNGNX = XLNGNX - 360.0
720          GO TO 3290
721      3265 XLNGNX = XLNGNX + 360.0
722          GO TO 3290
723      C
724      C      IF CURRENT-POINT WAS A CANDIDATE #2 (INEXACT LONGITUDE), THEN
725      C      NEXT LONGITUDE CROSSING MUST BE FOUND BY THE BRACKETING METHOD
726      C      USED FOR LEGNO = 1.
727      C
728      3270 GO TO 3210
729      C
730      C      SET ASSOCIATED LATITUDE (XLATAS).
731      C
732      3290 ICONV = 3

```

```

733      CALL GRTCIR (ICONV, FRMLAT, FRMLNG, TOLAT, TOLNG, XLATAS, XLNGNX,
734      &      THETAA, PHIA, THETAB, PHIB, QTX, QTY, QTZ, IGUESS, XLNGG,
735      &      XEAST, XNORTH, XLNGAP, SAPRIM, XLATC, THETAC, PHIX)
736      C      IF (NDEBUG .GT. 1) WRITE (6,8030) XLATAS, XLNGNX
737      C8030 FORMAT (1H, T4, 'XLATAS,XLNGNX = ', T36, 2F10.4)
738      C
739      C      BASED ON NORTHWARD DIRECTION INDICATOR, CALCULATE THE POINT OF
740      C      NEXT LATITUDE CROSSING (XLATNX) AND ASSOCIATED LONGITUDE
741      C      (XLNGAS) DETERMINED FROM THE EQUATION OF A GREAT CIRCLE. THIS
742      C      IS CANDIDATE POINT #2. STATEMENTS 3300-3399. DIFFERENT PRO-
743      C      CEDURE IS USED, DEPENDING WHETHER THIS IS FIRST LEG OF FLIGHT
744      C      OR WHETHER (ON SUBSEQUENT LEGS) THE CURRENT-POINT WAS A
745      C      CANDIDATE #1 (NCAND = 1) OR A CANDIDATE #2 (NCAND = 2).
746      C
747      C      FIRST SET COUNTER OF 'GRTCIR' CONVERGENCE FAILURES TO ZERO.
748      C      THIS WILL LATER BE USED TO REVERSE THE LATITUDE SEARCH.
749      C
750      3300 NFAIL = 0
751      C
752      C      DISCRIMINATE FIRST FROM SUBSEQUENT LEGS.
753      C
754      3310 IF (LEGN0 .GT. 1) GO TO 3340
755      C
756      C      PROCEDURE FOR FIRST LEG. DETERMINE BRACKETING LATITUDE VALUES.
757      C      MOVE NORTH (INCREASING LATITUDE) IF XNORTH .GT. 0.0. SET
758      C      XLATNX.
759      C
760      3315 CALL BRLAT (XLATC, XLATLO, XLATHI)
761      C      IF (NDEBUG .GT. 1) WRITE (6,8040) XLATLO, XLATHI
762      C8040 FORMAT (1H, T4, 'XLATLO,XLATHI = ', T36, 2F10.4)
763      C      IF (XNORTH) 3320,3320,3330
764      3320 XLATNX = XLATLO
765      C      GO TO 3390
766      3330 XLATNX = XLATHI
767      C      GO TO 3390
768      C
769      C      PROCEDURE FOR SUCCEEDING LEGS DEPENDS ON WHETHER CANDIDATE #1 OR
770      C      CANDIDATE #2 WAS CHOSEN LAST TIME.
771      C
772      3340 GO TO (3350,3370), NCAND
773      C
774      C      IF CURRENT-POINT WAS A CANDIDATE #1 (INEXACT LATITUDE), THEN NEXT
775      C      LATITUDE CROSSING MUST BE FOUND BY THE BRACKETING METHOD USED FOR
776      C      LEGN0 = 1.
777      C
778      3350 GO TO 3315
779      C
780      C      IF CURRENT-POINT WAS A CANDIDATE #2 (EXACT LATITUDE), THEN NEXT
781      C      LATITUDE CROSSING IS THAT LATITUDE PLUS OR MINUS DELTA-LATITUDE
782      C      (PLUS FOR NORTHWARD, MINUS FOR EASTWARD).
783      C
784      3370 XLATNX = (XNORTH * DELTA) + XLATC
785      C      IF (ABS(XLATNX) .LE. 0.001) XLATNX = 0.0
786      C
787      C      ADJUST LATITUDE FOR POLAR SINGULARITY.
788      C
789      C      IF (XLATNX .GT. 90.0) GO TO 3375
790      C      IF (XLATNX .LT. -90.0) GO TO 3380
791      C      GO TO 3390
792      3375 XLATNX = 180.0 - XLATNX
793      C      GO TO 3390
794      3380 XLATNX = -180.0 - XLATNX
795      C      GO TO 3390
796      C
797      C      ATTEMPT TO SET ASSOCIATED LONGITUDE (XLNGAS).
798      C
799      3390 ICONV = 5
800      C      IF (NDEBUG .GT. 1) WRITE (6,8050) XLATNX

```

```

801      C8050 FORMAT (1H , T4, 'XLATNX          = ', T36, F10.4)
802      CALL GRTCIR (ICONV, FRMLAT, FRMLNG, TOLAT, TOLNG, XLATNX,
803      &          XLNGAS, THETAA, PHIA, THETAB, PHIB, QTX, QTY, QTZ,
804      &          IGUESS, XLNGG, XEAST, XNORTH, XLNGAP, SAPRIM, XLATC, THETAX,
805      &          PHIC)
806      C
807      C      RESCUE PHIC FOR LATER USE.
808      C
809      ICONVX = 1
810      CALL SPHGLO (ICONVX, XLATC, XLNGC, THETAC, PHIC)
811      C      IF (NDEBUG .GT. 1) WRITE (6,8060) ICONV
812      C8060 FORMAT (1H , T4, 'ICONV          = ', T36, I10)
813      C      IF (NDEBUG .GT. 1) WRITE (6,8070) XLNGG
814      C8070 FORMAT (1H , T4, 'FIRST GUESS LONGITUDE = ', T36, F10.4)
815      C      IF (NDEBUG .GT. 1) WRITE (6,8090) XLATNX, XLNGAS
816      C8080 FORMAT (1H , T4, 'XLATNX,XLNGAS = ', T36, 2F10.4)
817      C
818      C      IF 'GRTCIR' SUBROUTINE FAILS TO CONVERGE IN LONGITUDE, INCREMENT
819      C      FAILURE COUNTER BY ONE, REVERSE LATITUDE SEARCH DIRECTION, AND
820      C      TRY AGAIN TO DETERMINE THE NEXT LATITUDE CROSSING (XLATNX) AND
821      C      ASSOCIATED LONGITUDE. OTHERWISE, CONTINUE. TWO CONSECUTIVE
822      C      FAILURES PRODUCE AN ERROR ABORT.
823      C
824      IF (ICONV .NE. 11) GO TO 3400
825      NFAIL = NFAIL + 1
826      IF (NFAIL .GT. 1) GO TO 7040
827      XNORTH = -XNORTH
828      C      IF (NDEBUG .GT. 1) WRITE (6,8090) XNORTH
829      C8090 FORMAT (1H , T4, 'REVERSED XNORTH = ', T36, F10.4)
830      GO TO 3310
831      C
832      C      CALCULATE GREAT CIRCLE DISTANCE (GCD) S1 FROM CURRENT-POINT
833      C      (XLATC,XLNGC) TO CANDIDATE #1 (XLATAS,XLNGNX). CALCULATE
834      C      GCD S2 FROM CURRENT-POINT TO CANDIDATE #2 (XLATNX,XLNGAS).
835      C
836      3400 CALL DISTAN (XLATC, XLNGC, XLATAS, XLNGNX, S1)
837      CALL DISTAN (XLATC, XLNGC, XLATNX, XLNGAS, S2)
838      C
839      C      WHICHEVER OF THE TWO POINTS, CANDIDATES 1 OR 2, IS CLOSEST TO THE
840      C      CURRENT-POINT IS SELECTED AS THE NEXT-POINT (XLATNP,XLNGNP)
841      C      ALONG THE PATH. DECISION IS MADE ON THE BASIS OF GCD S1 FOR CAN-
842      C      DIDATE POINT #1 AND S2 FOR CANDIDATE POINT #2.
843      C
844      IF (S2 - S1) 3420,3420,3430
845      C
846      C      CASE OF S1 .GE. S2 ... CHOOSE CANDIDATE #2.
847      C
848      3420 NCAND = 2
849      XLATN = XLATNP
850      XLNGN = XLNGNP
851      XLATNP = XLATNX
852      XLNGNP = XLNGAS
853      SNP = S2
854      GO TO 3450
855      C
856      C      CASE OF S2 .GE. S1 ... CHOOSE CANDIDATE #1.
857      C
858      3430 NCAND = 1
859      XLATN = XLATNP
860      XLNGN = XLNGNP
861      XLATNP = XLATAS
862      XLNGNP = XLNGNX
863      SNP = S1
864      C
865      C      NOTE... (XLATN,XLNGN) IS CROSSING POINT BEFORE NEXT-POINT.
866      C
867      C      CALCULATE GREAT CIRCLE DISTANCE (GCD) SNEXT IN NAUTICAL MILES
868      C      FROM POINT 'A' (FRMLAT,FRMLNG) TO THE NEXT-POINT (XLATNP,

```

```

869 C      XLNGNP) JUST SELECTED. IF THAT DISTANCE (SNEXT) IS GREATER
870 C      THAN THE TOTAL GCD (STOT) BETWEEN 'A' AND 'B,' THEN THE NEXT-
871 C      POINT IS BEYOND THE TERMINATION-POINT 'B.' IN SUCH A CASE,
872 C      THE COORDINATES OF 'B' MUST BE USED AS THE LAST-POINT
873 C      (XLATNP,XLNGNP). DISTANCE SNP MUST BE ADJUSTED, AND LAST-LEG
874 C      FLAG MUST BE TURNED ON.
875 C
876 C      3450 SNEXT = 0.0
877 C      IF (NDEBUG .GT. 1) WRITE (6,8100) NCAND, XLATNP, XLNGNP
878 C8100 FORMAT (1H, T4, 'NCAND,XLATNP,XLNGNP = ', T36, I10, 2F10.4)
879 C      CALL DISTAN (FRMLAT, FRMLNG, XLATNP, XLNGNP, SNEXT)
880 C      IF (STOT - SNEXT) 3460,3500,3500
881 C
882 C      PROCEDURE FOR SNEXT .GT. STOT.
883 C
884 C      3460 LASTLG = 1
885 C      XLATNP = TOLAT
886 C      XLNGNP = TOLNG
887 C      CALL DISTAN (XLATN, XLNGN, TOLAT, TOLNG, SNP)
888 C
889 C      FIND APPROXIMATE MIDPOINT BETWEEN CURRENT-POINT (XLATC,XLNGC)
890 C      AND NEXT-POINT (XLATNP,XLNGNP). THIS MIDPOINT IS LATER USED
891 C      TO DETERMINE GRID SECTOR NUMBER NSEC AND AS POSITION FOR
892 C      COMPUTATION OF AIRCRAFT HEADING HDGM.
893 C
894 C      3500 XLATM = 0.5 * (XLATC + XLATNP)
895 C      XLNGM = 0.5 * (XLNGC + XLNGNP)
896 C
897 C      FIND ROW NUMBER NROW (LATITUDE COUNTER) AND COLUMN NUMBER NCOL
898 C      (LONGITUDE COUNTER) OF MIDPOINT. THEN USE NROW AND NCOL TO
899 C      COMPUTE SECTOR NUMBER NSEC IN WHICH THE MIDPOINT (XLATM,
900 C      XLNGM) LIES. IF POINT LIES OUTSIDE PLUS OR MINUS 75.0 DEGREES
901 C      LATITUDE LIMITS OF THE GRID SYSTEM, BOGUS THE POINT INTO THE
902 C      SYSTEM.
903 C
904 C      XLATDM = XLATM
905 C      IF (XLATDM .GT. 75.0) XLATDM = 75.0
906 C      IF (XLATDM .LT. -60.0) XLATDM = -60.0
907 C      ANGLE = 75.0 - XLATDM
908 C      NROW = (ANGLE / DELTLA) + 1
909 C      ANGLE = -XLNGM + 30.0
910 C      IF (XLNGM .GT. 0.0) ANGLE = 390.0 - XLNGM
911 C      NCOL = (ANGLE / DELTLO) + 1
912 C      IF (NCOL .GT. 12) NCOL = NCOL - 12
913 C      NSEC = ((NROW - 1) * 12) + 1 + (NCOL - 1)
914 C
915 C      CALCULATE DATUM NUMBER NRND BASED ON SECTOR NUMBER NSEC
916 C      (1-108), AIRCRAFT ALTITUDE IALT (1-2) AND SEASON INDEX
917 C      ISEASN (1-4).
918 C
919 C      NRND = (ISEASN - 1) * 2 * 108 + (IALT - 1) * 108 + NSEC
920 C
921 C      CALCULATE AIRCRAFT HEADING HDGM (DIRECTION TOWARD WHICH IN
922 C      DEGREES) AT THE MIDPOINT 'M.'
923 C
924 C      HDGM = HDG (XLATM, XLNGM, TOLAT, TOLNG)
925 C
926 C      CONVERT HEADING IN DEGREES TO ANGLE ALPHA IN RADIANS. ALPHA
927 C      IS THE DIRECTION TOWARD WHICH THE AIRCRAFT'S GROUND TRACK IS
928 C      INCLINED, IN RADIANS COUNTERCLOCKWISE FROM EASTWARD.
929 C
930 C      ALPHA = 90.0 - HDGM
931 C      IF (ABS(HDGM) - 180.0) 3560,3560,3550
932 C      3550 ALPHA = 360.0 + ALPHA
933 C      3560 ALPHA = DTOR * ALPHA
934 C
935 C      RETRIEVE DIRECTION DIRM AND SPEED SPDM TOWARD WHICH THE WIND IS
936 C      BLOWING AT THE MIDPOINT POSITION (XLATM,XLNGM). DIRECTION IS

```

```

937 C      IN RADIANS COUNTERCLOCKWISE FROM EASTWARD (E.G., WIND FROM 225
938 C      DEGREES HAS DIRM = +PI/4 = 0.78540). SPEED IS IN KNOTS. ALSO
939 C      RETRIEVE VECTOR STANDARD VARIANCE VARM (SQUARE OF VECTOR STAN-
940 C      DARD DEVIATION) IN UNITS OF KNOTS**2.
941 C
942 C      DIRM = DIR(NRND)
943 C      SPDM = SPD(NRND)
944 C      VARM = VAR(NRND)
945 C
946 C      COMPUTE ALONG-GROUND-TRACK WIND COMPONENT VGM AND CROSS-GROUND-
947 C      TRACK WIND COMPONENT VCM IN KNOTS.
948 C
949 C      GAMMA = DIRM - ALPHA
950 C      GAMMAA = ABS(GAMMA)
951 C      VGM = SPDM * COS(GAMMAA)
952 C      VCM = SPDM * SIN(GAMMA )
953 C
954 C      COMPUTE TIME-AVERAGED WIND FACTOR WBAR IN KNOTS FOR THE
955 C      PRESENT LEG.
956 C
957 C      WBAR = VGM - ((VCM*VCM + (VARM/2.0)) / (2.0*ASPEED))
958 C      IF (NDEBUG .GT. 0) WRITE (6,8110) LEGNO, XLATC, XLNGC, XLATNP,
959 C      XLNGNP, SNP, NSEC, HDGM, DIRM, SPDM, VARM, WBAR
960 C8110 FORMAT (1H,12,2(F6.1,F7.1),T33,F8.2,T42,I4,1X,2F7.3,F6.1,
961 C      F6.0, F7.1)
962 C
963 C      ACCUMULATE TOTAL DISTANCE TO THE ACCUMULATOR SACC, DISTANCE-
964 C      WEIGHTED TOTAL VARIANCE TO VACC, DISTANCE-WEIGHTED TOTAL
965 C      ALONG-TRACK COMPONENT TO VGACC, AND DISTANCE-WEIGHTED
966 C      TOTAL CROSS-TRACK COMPONENT TO VCACC.
967 C
968 C      SACC = SACC + SNP
969 C      VACC = VACC + (SNP * VARM)
970 C      VGACC = VGACC + (SNP * VGM )
971 C      VCACC = VCACC + (SNP * VCM )
972 C
973 C      ADVANCE TO NEXT-POINT. OLD NEXT-POINT (XLATNP,XLNGNP) BECOMES
974 C      CURRENT-POINT (XLATC,XLNGC). UPDATE (THETAC, PHIC) COR-
975 C      RESPONDING TO (XLATC, XLNGC). INCREMENT LEG NUMBER COUNTER.
976 C      GIVE ERROR CONDITION FOR RUNAWAY ROUTE (EXCESSIVE NUMBER OF
977 C      LEGS).
978 C
979 C      XLATC = XLATNP
980 C      XLNGC = XLNGNP
981 C      ICONVX = 1
982 C      CALL SPHGLO (ICONVX, XLATC, XLNGC, THETAC, PHIC)
983 C      LEGNO = LEGNO + 1
984 C      IF (LEGNO .GT. LEGMAX) GO TO 7150
985 C
986 C      IF CURRENT-POINT (XLATC,XLNGC) IS POINT 'B' (TOLAT,TOLNG), THIS
987 C      IS THE LAST LEG OF THE FLIGHT, AND SUBROUTINE BRANCHES TO WRAP-
988 C      UP. OTHERWISE, ROUTINE GOES BACK FOR THE NEXT LEG.
989 C
990 C      XLADIF = ABS(XLATC - TOLAT)
991 C      XLNDIF = ABS(XLNGC - TOLNG)
992 C      IF (XLADIF .LT. 0.0001 .AND. XLNDIF .LT. 0.0001) LASTLG = 1
993 C      IF (LASTLG .EQ. 0) GO TO 3100
994 C
995 C      COMPUTE FINAL ROUTE-AVERAGED WIND FACTOR WBARBR. DIVIDE DISTANCE-
996 C      WEIGHTED ACCUMULATORS BY TOTAL DISTANCE TO GET ROUTE-MEAN VARI-
997 C      ANCE VARBR, ROUTE-MEAN ALONG-TRACK COMPONENT VGBR, AND ROUTE-
998 C      MEAN CROSS-TRACK COMPONENT VCBR. THEN CALCULATE THE ROUTE-MEAN
999 C      WIND FACTOR WBARBR.
1000 C
1001 C      VARBR = VACC / SACC
1002 C      VGBR = VGACC / SACC
1003 C      VCBR = VCACC / SACC
1004 C      WBARBR = VGBR - ((VCBR*VCBR + (VARBR / 2.0)) / (2.0 * ASPEED))

```

```

1005 C IF (NDEBUG .GT. 0) WRITE (6,8120) STOT, VGBR, VCBR, VARR, WBARBR
1006 C8120 FORMAT (1H0, 'GREAT CIRCLE DISTANCE', T30, F15.5//1X,
1007 C 'ROUTE-AVERAGE', T16, 'ALONG-TRACK', T30, F15.5/
1008 C T16, 'CROSS-TRACK', T30, F15.5/T16, 'VARIANCE', T30, F15.5/
1009 C T16, 'WIND FACTOR', T30, F15.5)
1010 C
1011 C IF IOPTN = 2, BRANCH TO PART IV OF THE SUBROUTINE TO ADJUST THE
1012 C WIND FACTOR TO THE 90% WORST CASE. OTHERWISE, BRANCH TO PART
1013 C VI FOR WRAPUP AND NORMAL TERMINATION.
1014 C
1015 C GO TO (6000, 6000, 4000, 6000), IOPTNP
1016 C
1017 C *****
1018 C * PART IV *
1019 C * 90% WORST WIND *
1020 C *****
1021 C
1022 C 90% WORST WIND FACTOR CALCULATION EMPLOYS THE SAWYER METHOD
1023 C DESCRIBED IN AIR WEATHER SERVICE TECHNICAL REPORT 77-267,
1024 C GUIDE TO APPLIED CLIMATOLOGY, PAGE 6-8. THE NEW VARIABLES
1025 C USED IN THIS CALCULATION ARE AS FOLLOWS...
1026 C
1027 C XKFACT = FACTOR TO CONVERT MEAN STANDARD VECTOR DEVIATION
1028 C OF WINDS OVER A ROUTE (STVDVW) TO STANDARD DEVIATION
1029 C OF THE MEAN WIND FACTOR (STDVWF)
1030 C
1031 C STVDVW = MEAN STANDARD VECTOR DEVIATION OF WINDS OVER A
1032 C ROUTE
1033 C
1034 C STDVWF = STANDARD DEVIATION OF THE MEAN WIND FACTOR
1035 C
1036 C WBARBR = 90% WORST WIND FACTOR (I.E., 10% RISK)
1037 C
1038 C CALCULATE K-FACTOR OF SAWYER. THE RELATION USED BELOW IS A
1039 C GEOMETRIC CURVE FIT TO SAWYER'S DATA REPORTED IN ANSH 77-
1040 C 267. THE GEOMETRIC CURVE FIT WAS CONSTRUCTED USING HEWLETT-
1041 C PACKARD 65 STAT PAC #2, PAGE 22. THE FUNCTION GIVES
1042 C K (XKFACT) AS A FUNCTION OF ROUTE LENGTH (STOT).
1043 C
1044 C 4000 XKFACT = 0.70585066 * (0.99983056**STOT)
1045 C
1046 C CALCULATE MEAN STANDARD VECTOR DEVIATION OF WIND OVER ROUTE.
1047 C
1048 C STVDVW = SQRT(VARBR)
1049 C
1050 C CALCULATE STANDARD DEVIATION OF THE MEAN WIND FACTOR.
1051 C
1052 C STDVWF = STVDVW * XKFACT
1053 C
1054 C CALCULATE 90% WORST WIND FACTOR.
1055 C
1056 C WBARBR = WBARBR - (1.28 * STDVWF)
1057 C
1058 C WRAPUP AND NORMAL TERMINATION.
1059 C
1060 C GO TO 6000
1061 C
1062 C *****
1063 C * PART V *
1064 C * SIMULATED WIND *
1065 C *****
1066 C
1067 C SIMULATED WIND PRESENTLY DEFAULTS TO THE MEAN WIND CASE.
1068 C
1069 C 5000 GO TO 3000
1070 C

```

```

1071 C *****
1072 C * PART VI *
1073 C * WRAPUP & NORMAL TERMINATION *
1074 C *****
1075 C
1076 C WIND FACTOR IS GROUND SPEED MINUS AIR SPEED. USE THIS RELATION
1077 C TO COMPUTE GROUND SPEED (GSPEED) IN KNOTS FROM GIVEN AIR SPEED
1078 C (ASPEED) IN KNOTS AND ROUTE-AVERAGED WIND FACTOR (WBARRR) IN
1079 C KNOTS.
1080 C
1081 C 6000 GSPEED = ASPEED + WBARRR
1082 C
1083 C NORMAL TERMINATION.
1084 C
1085 C RETURN
1086 C
1087 C *****
1088 C * PART VII *
1089 C * ABNORMAL TERMINATION *
1090 C *****
1091 C
1092 C 7000 GSPEED = ASPEED
1093 C WRITE (6,8130) JULDAT, GMT, IALT, FRMLAT, FRMLNG, TOLAT,
1094 C & TOLNG, IOPTN
1095 C 8130 FORMAT (1H0, '****WIND FACTOR ERROR DIAGNOSTICS****'/1X,
1096 C & 'JULIAN DATE = ', T15, I10, T35, 'GMT = ', T41, F10.2, T53,
1097 C & 'ALT OPTION = ', T66, I10/1X,
1098 C & 'FROM-LAT = ', F10.2, T30, 'FROM-LNG = ', T41, F10.2/1X,
1099 C & 'TO-LAT = ', F10.2, T32, 'TO-LNG = ', F10.2, T52,
1100 C & 'WIND OPTION = ', I10//)
1101 C CALL FXEM (61, 'SUBROUTINE ENRWND...ILLEGAL JULIAN DATE', 10)
1102 C RETURN
1103 C 7010 GSPEED = ASPEED
1104 C WRITE (6,8130) JULDAT, GMT, IALT, FRMLAT, FRMLNG, TOLAT,
1105 C & TOLNG, IOPTN
1106 C CALL FXEM (61, 'SUBROUTINE ENRWND...ILLEGAL ALTITUDE', 9)
1107 C RETURN
1108 C 7020 GSPEED = ASPEED
1109 C WRITE (6,8130) JULDAT, GMT, IALT, FRMLAT, FRMLNG, TOLAT,
1110 C & TOLNG, IOPTN
1111 C CALL FXEM (61, 'SUBROUTINE ENRWND...ILLEGAL WIND OPTION', 10)
1112 C RETURN
1113 C 7030 GSPEED = ASPEED
1114 C WRITE (6,8130) JULDAT, GMT, IALT, FRMLAT, FRMLNG, TOLAT,
1115 C & TOLNG, IOPTN
1116 C CALL FXEM (61, 'SUBROUTINE ENRWND...ILLEGAL LAT/LON', 9)
1117 C RETURN
1118 C 7040 GSPEED = ASPEED
1119 C WRITE (6,8130) JULDAT, GMT, IALT, FRMLAT, FRMLNG, TOLAT,
1120 C & TOLNG, IOPTN
1121 C CALL FXEM (61, 'SUBROUTINE ENRWND...GRT CIR FAILED TWICE', 10)
1122 C 7050 GSPEED = ASPEED
1123 C WRITE (6,8130) JULDAT, GMT, IALT, FRMLAT, FRMLNG, TOLAT,
1124 C & TOLNG, IOPTN
1125 C CALL FXEM (61, 'SUBROUTINE ENRWND...RUNAWAY ROUTE', 9)
1126 C RETURN
1127 C END

```

```

1 CGRTCIR EQ GRT CIRCLE/R. C. WHITON/05 FEB 1979
2 C
3 SUBROUTINE GRTCIR (ICONV, FRMLAT, FRMLNG, TOLAT, TOLNG, XLAT,
4 & XLNG, THETAA, PHIA, THETAB, PHIB, QTX, QTY, QTZ, IGUESS, XLNGG,
5 & XEAST, XNORTH, XLNGAP, SAPRIM, XLATC, THETA, PHI)
6 C
7 C*****
8 C*
9 C* PROGRAM ID- GRTCIR
10 C* MET ANALYST- MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
11 C* SYS ANALYST- MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
12 C* PROGRAMMER- MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
13 C*
14 C* CREATED ON- 05 FEB 1979 PROJECT- 192301
15 C*
16 C* DESCRIPTION- THIS SUBROUTINE SUBPROGRAM SOLVES THE EQUATION OF
17 C* THE GREAT CIRCLE DEFINED BY POINTS 'A' ((FRMLAT,
18 C* FRMLNG) OR (THETAA,PHIA)) AND 'B' ((TOLAT,TOLNG)
19 C* OR (THETAB,PHIB)) AT AN INTERMEDIATE POINT
20 C* ((XLAT,XLNG) OR (THETA,PHI)). DEPENDING ON THE
21 C* VALUE OF THE FLAG ICONV, THE PROGRAM PERFORMS ONE
22 C* OF THE FOLLOWING FUNCTIONS...
23 C*
24 C* ICONV FUNCTION
25 C*
26 C* 1 GIVEN INPUT LATITUDE AND LONGITUDE OF POINTS
27 C* 'A' (FRMLAT,FRMLNG) AND 'B' (TOLAT,TOLNG),
28 C* CALCULATE SPHERICAL COORDINATES (THETAA,
29 C* PHIA) AND (THETAB,PHIB) OF THE POINTS BY
30 C* INVOKING SUBROUTINE SPHGLO. THEN COMPUTE
31 C* THE CROSS PRODUCT VECTOR  $QT = QA \times QB$  OF
32 C* 'A' AND 'B.'
33 C*
34 C* 2 INITIALIZE THE EXACT SPHERICAL TRIANGLE
35 C* METHOD OF OBTAINING FIRST GUESS LONGITUDE.
36 C* FIRST, DETERMINE THE LONGITUDE XLNGAP OF
37 C* THE POINT A-PRIME AT WHICH THE GREAT CIR-
38 C* CLE ROUTE CROSSES THE EQUATOR. THEN ES-
39 C* TABLISH THE INTERIOR ANGLE A-PRIME OF THE
40 C* SPHERICAL TRIANGLE (A-PRIME,C-PRIME,B).
41 C* THE SINE OF THAT ANGLE IS SAPRIM.
42 C*
43 C* 3 GIVEN INPUT INTERMEDIATE LONGITUDE XLNG,
44 C* CALCULATE INTERMEDIATE PHI, INTERMEDIATE
45 C* THETA AND INTERMEDIATE LATITUDE XLAT.
46 C*
47 C* 4 GIVEN INPUT INTERMEDIATE PHI, CALCULATE IN-
48 C* TERMEDIATE THETA AND INTERMEDIATE LATITUDE
49 C* XLAT.
50 C*
51 C* 5 GIVEN INPUT INTERMEDIATE LATITUDE XLAT,
52 C* CALCULATE INTERMEDIATE THETA, INTERMEDIATE
53 C* PHI AND INTERMEDIATE LONGITUDE XLNG.
54 C*
55 C* 6 GIVEN INPUT INTERMEDIATE THETA, CALCULATE
56 C* INTERMEDIATE PHI AND INTERMEDIATE LONGI-
57 C* TUDE XLNG.
58 C*
59 C* THIS SUBROUTINE MUST BE INVOKED ONCE FOR FUNCTION
60 C* ICONV = 1 AND ONCE FOR ICONV = 2 BEFORE BEING
61 C* USED FOR FUNCTION ICONV = 3 THROUGH 6. THUS
62 C* THERE MUST BE 'INITIAL' CALLS TO THIS ROUTINE
63 C* USING ICONV = 1 AND 2 BEFORE CALLS USING
64 C* ICONV = 3 THROUGH 6.
65 C*
66 C* METHOD- THE METHOD USED EMERGES FROM THE VECTOR CALCULUS
67 C* IN A RECTANGULAR (X,Y,Z) COORDINATE SYSTEM. QA IS
68 C* THE POSITION VECTOR OF POINT 'A,' WHILE QB IS THAT

```

```

69      C*      OF POINT 'B.' THE POSITION VECTOR OF THE INTER-
70      C*      MEDIATE POINT IS 0. THE PLANE OF THE GREAT CIRCLE
71      C*      IS DEFINED BY THE CROSS PRODUCT  $QT = QA \times QB$ ,
72      C*      WHICH IS DIRECTED NORMAL TO THE PLANE. THE EQUA-
73      C*      TION OF THE GREAT CIRCLE IS DEFINED BY THE RELATION,
74      C*
75      C*       $Q \cdot (QA \times QB) = 0$ 
76      C*
77      C*      WHERE THE PERIOD INDICATES A DOT PRODUCT. THIS IS
78      C*      EQUIVALENT TO SAYING THAT THE COMPONENT OF Q IN THE
79      C*      QT-DIRECTION IS ZERO. EXPANSION OF THIS PRODUCT IN
80      C*      RECTANGULAR COORDINATES LEADS TO THE RELATION,
81      C*
82      C*       $QTX * SIN(THETA) * COS(PHI) + QTY * SIN(THETA) * SIN(PHI)$ 
83      C*       $+ QTZ * COS(THETA) = 0$ 
84      C*
85      C*      WHERE QTX, QTY AND QTZ ARE THE X, Y AND Z-COMPONENTS
86      C*      OF THE CROSS PRODUCT QT, DIVIDED BY  $R^2$ , THE
87      C*      SQUARE OF THE RADIUS OF THE EARTH. SOLVING THIS
88      C*      EQUATION FOR THETA, THE COLATITUDE, LEADS TO A
89      C*      DIRECTLY SOLVABLE EQUATION,
90      C*
91      C*       $THETA = ATAN(-QTZ / (QTX * COS(PHI) + QTY * SIN(PHI)))$ 
92      C*
93      C*      ON THE OTHER HAND, SOLVING FOR PHI LEADS TO A
94      C*      TRANSCENDENTAL EQUATION OF THE FORM,
95      C*
96      C*       $QTX * COS(PHI) + QTY * SIN(PHI) + QTZ / TAN(THETA) = 0$ 
97      C*
98      C*      THIS LATTER EQUATION IS SOLVED BY NEWTON'S ITERA-
99      C*      TIVE TECHNIQUE FOR NON-LINEAR EQUATIONS. THE
100     C*      CRITICAL FIRST GUESS FOR PHI IS PROVIDED BY ONE
101     C*      OF THREE METHODS. DEPENDING ON USER DESIRES AND
102     C*      WHETHER THE POINT A-PRIM HAS ALREADY BEEN ESTAB-
103     C*      LISHED, ANY ONE OF THE THREE FIRST GUESS METHODS
104     C*      CAN BE USED. THE METHODS ARE...
105     C*
106     C*      (1) USER-SPECIFIED FIRST GUESS
107     C*
108     C*      (2) APPROXIMATELY PLANAR TRIANGLE
109     C*
110     C*      (3) EXACT SPHERICAL TRIANGLE.
111     C*
112     C*      NORMALLY THE THIRD METHOD IS USED, BUT METHOD (2)
113     C*      MUST BE USED AT LEAST ONCE TO BREAK GROUND FOR
114     C*      (3). METHOD (2) IS ONLY APPROXIMATE AND LOSES
115     C*      ACCURACY THE LONGER THE DIS-
116     C*      TANCE BETWEEN 'A' AND 'B.' THIS CAN BE A MAJOR
117     C*      LIMITING FACTOR, BECAUSE NEWTON'S TECHNIQUE REQUIRES
118     C*      A REASONABLY CLOSE FIRST GUESS FOR PHI.
119     C*
120     C*      LIMITATIONS-
121     C*      1. THE APPROXIMATELY PLANAR TRIANGLE
122     C*      METHOD USED AS FIRST GUESS FOR NEWTON'S ITERATIVE
123     C*      TECHNIQUE APPEARS TO BREAK DOWN WHEN ABSOLUTE VAL-
124     C*      OF THE DELTA-LONGITUDE OR DELTA-LATITUDE BETWEEN
125     C*      POINTS 'A' AND 'B' EXCEEDS ABOUT 40-50 DEGREES, PAR-
126     C*      TICULARLY WHEN THE TWO POINTS ARE IN DIFFERENT QUAD-
127     C*      RANTS OF THE GLOBE. UNDER THESE CIRCUMSTANCES, NEW-
128     C*      TON'S METHOD CONVERGES TO A FALSE ROOT OR FAILS TO
129     C*      CONVERGE ENTIRELY, AND IT BECOMES IMPOSSIBLE TO
130     C*      SOLVE THE EQUATION OF A GREAT CIRCLE FOR THE AZIMUTH
131     C*      ANGLE PHI. POINTS 'A' AND 'B' ALONG THE GREAT
132     C*      CIRCLE MUST THUS BE KEPT REASONABLY CLOSE TOGETHER.
133     C*      IF POINT 'B' IS TOO FAR REMOVED FROM 'A,' THEN USE
134     C*      THIS SUBROUTINE DETERMINISTICALLY (ICONV = 1 AND 2
135     C*      FOLLOWED BY ICONV = 3 OR 4) TO CALCULATE THETA
136     C*      (XLAT) FOR AN INTERMEDIATE PHI (XLNG) NOT MORE THAN
137     C*      30 DEGREES FROM PHIA. THEN USE THAT POINT AS THE

```

```

137      C*      NEW POINT B IN ALL FURTHER GREAT CIRCLE COMPUTA-
138      C*      TIONS. THIS CAN BE DONE BECAUSE THE 'INTERMEDIATE'
139      C*      POINT NEED NOT BE BOUNDED BY 'A' AND 'B.'
140      C*
141      C*      2. IN THE CASE OF A GREAT CIRCLE THAT IS ALSO
142      C*      A LONGITUDE CIRCLE, THE IMPLICIT FUNCTION
143      C*      F(THETA,PHI) = 0.0 DOES NOT RETURN A UNIQUE
144      C*      SOLUTION IN THETA FOR A SPECIFIED PHI. IF THE
145      C*      GREAT CIRCLE IS ALONG A LONGITUDE, ANY VALUE
146      C*      OF THETA IS A SOLUTION TO THE FUNCTION. THIS
147      C*      SUBROUTINE CANNOT BE USED TO CALCULATE THETA
148      C*      FROM PHI IF FRMLNG = TOLNG. AN ERROR MESSAGE
149      C*      IS DISPLAYED ON SYSOUT AND ERROR FLAG ICONV=12
150      C*      IS SET IF THIS IS ATTEMPTED.
151      C*
152      C*      INPUT-      FRMLAT = LATITUDE OF POINT 'A' (ORIGIN) IN DEG
153      C*
154      C*      FRMLNG = LONGITUDE OF POINT 'A' (ORIGIN) IN DEG
155      C*
156      C*      TOLAT = LATITUDE OF POINT 'B' (DESTINATION) IN DEG
157      C*
158      C*      TOLNG = LONGITUDE OF POINT 'B' (DESTINATION) IN DEG
159      C*
160      C*      THETAA = COLATITUDE ANGLE IN RADIAN OF POINT 'A,'
161      C*      0.0 TO PI RADIAN SOUTHWARD FROM NORTH,
162      C*      POSITIVE
163      C*
164      C*      PHIA = AZIMUTH ANGLE IN RADIAN COUNTERCLOCKWISE
165      C*      FROM GREENWICH LOOKING DOWNWARD AT THE
166      C*      NORTH POLE. PLUS OR MINUS 0.0 TO PI RADIAN
167      C*      (NEGATIVE MEANS CLOCKWISE)
168      C*
169      C*      THETAB = COLATITUDE IN RADIAN OF POINT 'B'
170      C*
171      C*      PHIB = AZIMUTH IN RADIAN OF POINT 'B'
172      C*
173      C*      I GUESS = FLAG CONTROLLING WHETHER FIRST GUESS IS
174      C*      SPECIFIED FOR LONGITUDE IN VARIABLE
175      C*      XLNGG. IF I GUESS = 1, XLNGG IS INTER-
176      C*      PRETED AS FIRST GUESS LONGITUDE. IF
177      C*      I GUESS = 0, XLNGG IS IGNORED ON INPUT
178      C*      BUT IS FILLED WITH THE PROGRAM-GENERATED
179      C*      FIRST GUESS VALUE ON OUTPUT.
180      C*
181      C*      XEAST = EASTWARD DIRECTION INDICATOR, POSITIVE
182      C*      FOR AIRCRAFT MOTION TOWARD THE EAST.
183      C*
184      C*      XNORTH = NORTHWARD DIRECTION INDICATOR, POSITIVE
185      C*      FOR AIRCRAFT MOTION TOWARD THE NORTH.
186      C*
187      C*      XLATC = LATITUDE OF CURRENT-POINT, I.E., THE
188      C*      POINT FROM WHICH THE PROGRAM IS MARCHING
189      C*      IN OBTAINING XLAT, THE LATITUDE OF THE
190      C*      SO CALLED NEXT-POINT.
191      C*
192      C*      INPUT &
193      C*      OUTPUT-      ICONV = FLAG CONTROLLING FUNCTION TO BE PERFORMED
194      C*      BY THIS SUBROUTINE. SEE TABLE IN 'METHOD'
195      C*      ABOVE. WHEN RETURNED AS OUTPUT, ICONV = 1
196      C*      THROUGH 6 FOR NORMAL TERMINATION OR A NUM-
197      C*      BER GREATER THAN OR EQUAL TO 10 FOR ABNOR-
198      C*      MAL TERMINATION. ABNORMAL TERMINATION
199      C*      CODES FOR ICONV ARE...
200      C*
201      C*      10- ERRONEOUS ICONV SPECIFIED
202      C*      11- NEWTON'S ITERATIVE TECHNIQUE FOR
203      C*      SOLUTION OF NON-LINEAR EQUATIONS
204      C*      DID NOT CONVERGE IN NMAX ITERA-

```

```

205      C*
206      C*
207      C*
208      C*
209      C*
210      C*
211      C*
212      C*
213      C*
214      C*
215      C*
216      C*
217      C*
218      C*
219      C*
220      C*
221      C*
222      C*
223      C*
224      C*
225      C*
226      C*
227      C*
228      C*
229      C*
230      C*
231      C*
232      C*
233      C*
234      C*
235      C*
236      C*
237      C*
238      C*
239      C*
240      C*
241      C*
242      C*
243      C*
244      C*
245      C*
246      C*
247      C*
248      C*
249      C*
250      C*
251      C*
252      C*
253      C*
254      C*
255      C*
256      C*
257      C*
258      C*
259      C*
260      C
261      C
262      C
263      C
264      C
265      C
266      C
267      C
268      C
269      C
270      C
271      C
272      C

```

TIONS
 12- ATTEMPT WAS MADE TO CALCULATE
 THETA FROM PHI FOR A FLIGHT
 ALONG A LONGITUDE CIRCLE

XLAT = LATITUDE OF INTERMEDIATE POINT IN DEG
 XLNG = LONGITUDE OF INTERMEDIATE POINT IN DEG
 QTX = X-COMPONENT OF CROSS PRODUCT QT DIVIDED BY
 R**2. COMPUTED IN FUNCTION ICONV = 1 AND
 REUSED IN SUBSEQUENT CALLS
 QTY = Y-COMPONENT OF ABOVE
 QTZ = Z-COMPONENT OF ABOVE
 XLNGG = FIRST GUESS LONGITUDE FOR ICONV = 4
 OR 5. IF VALUE IS NOT SPECIFIED AS
 INPUT, THE PROGRAM-GENERATED FIRST
 GUESS WILL BE PROVIDED AS OUTPUT
 XLNGAP = LONGITUDE IN DEGREES OF THE POINT A-PRIME
 AT WHICH THE GREAT CIRCLE CROSSES THE
 EQUATOR
 SAPRIM = SINE OF INTERIOR ANGLE A-PRIME OF THE
 EXACT SPHERICAL TRIANGLE
 THETA = COLATITUDE OF INTERMEDIATE POINT IN RADIANS
 PHI = AZIMUTH OF INTERMEDIATE POINT IN RADIANS
 NOTE- ALL SOUTH LATITUDES AND EAST LONGITUDES ARE
 REPRESENTED AS NEGATIVE NUMBERS.

SYSTEM SUB-
 PROGRAMS
 USED- SIN, ARSIN, COS, ARCOS, TAN, ATAN, EXEM
 USER SUB-
 PROGRAMS
 USED- SPHGLO, HDG
 ESTIMATED
 CPU TIME- TO BE SUPPLIED.
 STORAGE
 REQUIREMENTS- PROCEDURE PLUS DATA OCCUPY 1042 WORDS OF STORAGE.
 PROGRAM
 UPDATES- NONE

 EPSILON IS THE CRITERION FOR CONVERGENCE OF NEWTON'S ITERATIVE
 TECHNIQUE AND HAS UNITS OF RADIANS. PI IS THE GEOMETRIC PI.
 NMAX IS THE MAXIMUM NUMBER OF ITERATIONS PERMITTED FOR THE NEW-
 TON TECHNIQUE. NTRIFG IS A FLAG INDICATING WHETHER THE
 DIRECTION (TRIDIR) OF THE SPHERICAL TRIANGLE IN FIRST
 GUESS METHOD #3 HAS BEEN REVERSED DUE TO AN EQUATOR
 CROSSING. NORMALLY ZERO, NTRIFG = 1 INDICATES A REVERSAL.
 COMMON /DBG/ NDEBUG
 DIMENSION APRMD(2), XLNGGG(2), PHIGG(2)
 DATA EPS/0.0001/, PI/3.1415927/, NTRIFG/0/, NMAX/7/

```

273 C      BRANCH TO APPROPRIATE SECTION OF THE SUBROUTINE BASED ON THE FUNC-
274 C      TION REQUESTED VIA THE ICONV FLAG.  THE ERROR CONDITION ICONV=10
275 C      IS RETURNED FOR ILLEGAL ICONV VALUES.  IF FLIGHT IS ALONG
276 C      A LONGITUDE CIRCLE AND ICONV REQUESTS CALCULATION OF THETA,
277 C      AN ERROR CONDITION ICONV=12 IS SET AND AN ERROR MESSAGE IS
278 C      DISPLAYED ON SYSOUT.  ICONVS IS A SAVED VALUE OF THE ORIGINAL
279 C      ICONV.
280 C
281 C      ICONVS = ICONV
282 C      IF (ICONV .GE. 1 .AND. ICONV .LE. 5) GO TO 50
283 C      ICONV = 10
284 C      GO TO 1000
285 C      50 IF (ICONV .NE. 2 .AND. ICONV .NE. 3) GO TO 60
286 C      IF (ABS(FRMLNG - TOLNG) .GT. 0.0001) GO TO 60
287 C      ICONV = 12
288 C      GO TO 1020
289 C      60 GO TO (100, 200, 300, 400, 500, 600), ICONV
290 C
291 C
292 C      *****
293 C      *          PART I          *
294 C      *      FUNCTION ICONV = 1      *
295 C      *      CROSS PRODUCT COMPUTATION      *
296 C      *****
297 C
298 C      CONVERT (LAT,LNG) OF 'A' AND 'B' TO (THETAA,PHIA) AND (THETAB,
299 C      PHIB).
300 C
301 C      100 ICONVX = 1
302 C      CALL SPHGLO (ICONVX, FRMLAT, FRMLNG, THETAA, PHIA)
303 C      CALL SPHGLO (ICONVX, TOLAT, TOLNG, THETAB, PHIB)
304 C
305 C      PRELIMINARIES FOR COMPONENTS OF POSITION VECTORS QA AND QB.
306 C
307 C      SINTHA = SIN(THETAA)
308 C      COSTHA = COS(THETAA)
309 C      SINPHA = SIN(PHIA )
310 C      COSPHA = COS(PHIA )
311 C      SINTHB = SIN(THETAB)
312 C      COSTHB = COS(THETAB)
313 C      SINPHB = SIN(PHIB )
314 C      COSPHB = COS(PHIB )
315 C
316 C      RECTANGULAR COMPONENTS OF POSITION VECTORS QA AND QB, DIVIDED
317 C      BY THE RADIUS OF THE EARTH R.
318 C
319 C      QAX = SINTHA * COSPHA
320 C      QAY = SINTHA * SINPHA
321 C      QAZ = COSTHA
322 C      QBX = SINTHB * COSPHB
323 C      QBY = SINTHB * SINPHB
324 C      QBZ = COSTHB
325 C
326 C      COMPUTE COMPONENTS OF THE CROSS PRODUCT VECTOR QT = QA X QB,
327 C      DIVIDED BY THE RADIUS OF THE EARTH SQUARED, R**2.
328 C
329 C      QTX = QAY * QBZ - QAZ * QBY
330 C      QTY = QAZ * QBX - QAX * QBZ
331 C      QTZ = QAX * QBY - QAY * QBX
332 C
333 C      INITIALIZE TRIANGLE FLAG, USED TO REVERSE THE DIRECTION
334 C      TRIDIR OF THE SPHERICAL TRIANGLE IN THE FIRST GUESS PRO-
335 C      CEDURE WHENEVER THE ROUTE CROSSES THE EQUATOR.
336 C
337 C      NTRIFG = 0
338 C
339 C      END OF FUNCTION ICONV = 1.  RETURN TO MAIN PROGRAM.
340 C

```

```

341      RETURN
342      C
343      C
344      C      *****
345      C      *      PART II      *
346      C      *      FUNCTION ICONV = 2      *
347      C      *      EQ CROSSING PT & ANGLE      *
348      C      *      *****
349      C
350      C      DETERMINE WHETHER POINT 'A' OR POINT 'B' IS TO SERVE AS POINT
351      C      'E.' WHICHEVER IS CLOSEST TO THE EQUATOR WILL SERVE AS 'E.'
352      C      THE OTHER WILL SERVE AS 'F.' NSIGN IS A FLAG INDICATING THE
353      C      ROUTE INVOLVES AN EQUATOR CROSSING. NORMALLY 1, NSIGN TAKES
354      C      ON THE VALUE 2 FOR ROUTES CROSSING THE EQUATOR. NCASE IS
355      C      A FLAG THAT HAS THE VALUE 12 WHEN POINT 'A' SERVES AS POINT
356      C      'E' AND THE VALUE 34 WHEN POINT 'B' SERVES AS POINT 'E.'
357      C
358      200 NSIGN = 1
359      IF (FRMLAT .LT. 0.0 .AND. TOLAT .GT. 0.0) NSIGN = 2
360      IF (FRMLAT .GT. 0.0 .AND. TOLAT .LT. 0.0) NSIGN = 2
361      XLATE = FRMLAT
362      XLNGE = FRMLNG
363      PHIE = PHIA
364      XLATF = TOLAT
365      XLNGF = TOLNG
366      PHIF = PHIB
367      NCASE = 12
368      IF ( ABS(XLATE) .LT. ABS(XLATF) ) GO TO 205
369      XLATE = TOLAT
370      XLNGE = TOLNG
371      PHIE = PHIB
372      XLATF = FRMLAT
373      XLNGF = FRMLNG
374      PHIF = PHIA
375      NCASE = 34
376      205 CONTINUE
377      C      IF (NDEBUG .GT. 1) WRITE (6,6000) NSIGN, NCASE
378      C6000 FORMAT (1H , T4, 'NSIGN,NCASE = ', T36, 2I10)
379      C
380      C      ESTABLISH INITIAL HEADING OF AIRCRAFT AT POINT 'E' ON THE GREAT
381      C      CIRCLE. CONVERT HEADING AT 'E' IN DEGREES TO ANGLE 'E' IN
382      C      RADIANS. TAKE ITS TANGENT TANE FOR LATER USE IN THE FIRST
383      C      GUESS PROCEDURE #2.
384      C
385      HDGE = HDG (XLATE, XLNGE, XLATF, XLNGF)
386      IF (HDGE .GE. 270.0) GO TO 212
387      IF (HDGE .GE. 180.0) GO TO 214
388      IF (HDGE .GE. 90.0) GO TO 216
389      GO TO 218
390      212 E = HDGE - 270.0
391      GO TO 220
392      214 E = 270.0 - HDGE
393      GO TO 220
394      216 E = HDGE - 90.0
395      GO TO 220
396      218 E = 90.0 - HDGE
397      GO TO 220
398      220 TANE = TAN (E * (PI/180.0))
399      C      IF (NDEBUG .GT. 1) WRITE (6,6010) HDGE, E
400      C6010 FORMAT (1H , T4, 'HEADING AT POINT E = ', T36, F10.4,
401      C      ' DEG'/1X, T4, T4, 'ANGLE E = ', T36, F10.4, ' DEG')
402      C
403      C      SET LATITUDE TO ZERO AND DETERMINE LONGITUDE WHERE GREAT CIRCLE
404      C      CROSSES EQUATOR. THIS IS POINT A-PRIME, WHOSE LONGITUDE IS
405      C      XLNGAP IN DEGREES.
406      C
407      XLAT = 0.0
408      GO TO 500

```

```

409      230 XLNGAP = XLNG
410      C      IF (NDEBUG .GT. 1) WRITE (6,6020) XLNGAP
411      C6020 FORMAT (1H , T4, 'LONGITUDE OF EQUATOR CROSSING = ', T36, F10.4,
412      C      ' DEG')
413      C
414      C      IF LONGITUDE XLNGAP OF EQUATOR CROSSING POINT 'A-PRIME' IS
415      C      WEST OF LONGITUDE OF POINT 'A' (FRMLNG), THE SPHERICAL
416      C      TRIANGLE OPENS EASTWARD (TRIDIR = -1.0). OTHERWISE, IT
417      C      OPENS WESTWARD (TRIDIR = 1.0).
418      C
419      XLGAPP = XLNGAP
420      FRMLNN = FRMLNG
421      IF (XLGAPP .LT. -90.0) XLGAPP = XLGAPP + 360.0
422      IF (FRMLNN .LT. -90.0) FRMLNN = FRMLNN + 360.0
423      IF (FRMLNN - XLGAPP ) 240,245,245
424      240 TRIDIR = -1.0
425      GO TO 250
426      245 TRIDIR = 1.0
427      250 CONTINUE
428      C      IF (NDEBUG .GT. 1) WRITE (6,6030) TRIDIR
429      C6030 FORMAT (1H , T4, 'TRIDIR = ', T36, F10.4)
430      C
431      C      ESTABLISH INITIAL HEADING OF THE AIRCRAFT AT POINT A-PRIME ON
432      C      CIRCLE. CONVERT HEADING AT A-PRIME TO ANGLE A-PRIME. CONVERT
433      C      ANGLE A-PRIME TO RADIANS AND TAKE ITS SINE FOR LATER USE IN
434      C      FIRST GUESS PROCEDURE #3.
435      C
436      260 HDGAP = HDG (XLAT, XLNGAP, TOLAT, TOLNG)
437      IF (HDGAP .GE. 270.0) GO TO 270
438      IF (HDGAP .GE. 180.0) GO TO 275
439      IF (HDGAP .GE. 90.0) GO TO 280
440      GO TO 285
441      270 APRIMD = HDGAP - 270.0
442      GO TO 290
443      275 APRIMD = 270.0 - HDGAP
444      GO TO 290
445      280 APRIMD = HDGAP - 90.0
446      GO TO 290
447      285 APRIMD = 90.0 - HDGAP
448      GO TO 290
449      290 SAPRIM = SIN (APRIMD * (PI/180.0))
450      C      IF (NDEBUG .GT. 1) WRITE (6,6040) APRIMD
451      C6040 FORMAT (1H , T4, 'APRIMD = ', T36, F10.4, ' DEG')
452      C
453      C      END OF FUNCTION ICONV = 2. RETURN TO MAIN PROGRAM.
454      C
455      RETURN
456      C
457      C
458      C      *****
459      C      *          PART III          *
460      C      *          FUNCTION ICONV = 3          *
461      C      *          XLNG TO XLAT          *
462      C      *          *****
463      C
464      C      CONVERT INTERMEDIATE LONGITUDE XLNG TO INTERMEDIATE PHI.
465      C
466      300 ICONVX = 1
467      CALL SPHGLO (ICONVX, 0.0, XLNG, THETAX, PHI)
468      C
469      C
470      C      *****
471      C      *          PART IV          *
472      C      *          FUNCTION ICONV = 4          *
473      C      *          PHI TO XLAT          *
474      C      *          *****
475      C
476      C      CALCULATE THETA FROM PHI.

```

```

477 C
478 400 THETA = ATAN(-QTX / (QTX * COS(PHI) + QTY * SIN(PHI) ) )
479 IF (THETA) 410,900,900
480 410 THETA = THETA + PI
481 GO TO 900
482 C
483 C
484 C *****
485 C * PART V *
486 C * FUNCTION ICONV = 5 *
487 C * XLAT TO XLNG *
488 C *****
489 C
490 C SAVE AZIMUTH ANGLE OF THE CURRENT POINT IN PHIC FOR LATER USE IN
491 C CHECKING FOR MONOTONIC PROGRESS IN LONGITUDE.
492 C
493 500 PHIC = PHI
494 C
495 C CONVERT INTERMEDIATE LATITUDE XLAT TO INTERMEDIATE THETA.
496 C
497 ICONVX = 1
498 CALL SPHGLO (ICONVX, XLAT, 0.0, THETA, PHIX)
499 C
500 C
501 C *****
502 C * PART VI *
503 C * FUNCTION ICONV = 6 *
504 C * THETA TO XLNG *
505 C *****
506 C
507 C CALCULATE PHI FROM THETA (NEWTON'S ITERATIVE TECHNIQUE WITH AN
508 C APPROPRIATE FIRST GUESS).
509 C
510 C THREE METHODS CAN BE USED AS A FIRST GUESS FOR THE AZIMUTH
511 C ANGLE PHI. THE FIRST AND LEAST SOPHISTICATED METHOD IS
512 C FOR THE USER TO SPECIFY A FIRST GUESS BY SETTING INPUT
513 C I GUESS = 1 AND BY INSERTING A FIRST GUESS LONGITUDE IN
514 C XLNGG IN HIS CALL TO THE GRTCIR SUBROUTINE. THIS METHOD
515 C IS IMPLEMENTED IN STATEMENTS 605-609.
516 C
517 C THE SECOND FIRST GUESS METHOD IS THAT OF APPROXIMATELY
518 C PLANAR TRIANGLES. THIS METHOD HAS IMPORTANT LIMIT-
519 C TATIONS ON ITS RELIABILITY AND SO IS USED ONLY TO FIND THE
520 C EQUATORIAL LONGITUDE OF THE GREAT CIRCLE (FUNCTION ICONV =
521 C 2). THE APPROXIMATELY PLANAR TRIANGLE METHOD
522 C IS IMPLEMENTED IN STATEMENTS 610-624.
523 C
524 C THE THIRD AND MOST SOPHISTICATED FIRST GUESS METHOD IS THAT
525 C OF EXACT SPHERICAL TRIANGLES. HIS METHOD REQUIRES
526 C KNOWLEDGE OF THE EQUATORIAL CING POINT AND ANGLE.
527 C THE EXACT SPHERICAL TRIANGLE METHOD GUESAS METHOD IS
528 C IMPLEMENTED IN STATEMENTS 625-700.
529 C 625-700.
530 C
531 600 IF (IGUESS .EQ. 1) GO TO 605
532 IF (ICONV .EQ. 2) GO TO 610
533 GO TO 625
534 C
535 C *****FIRST GUESS METHOD #1*****
536 C
537 C SPECIFIED FIRST GUESS. STATEMENTS 605-609.
538 C
539 C
540 605 ICONVX = 1
541 DUMLAT = 0.0
542 DUMTHE = 0.0
543 CALL SPHGLO (ICONVX, DUMLAT, XLNGG, DUMTHE, PHIG)
544 PHIN = PHIG

```

```

545      ICONVX = 2
546      CALL SPHGLO (ICONVX, DUMLAT, XLNGG, DUMTHE, PHIG)
547      GO TO 750
548      C
549      C
550      C      *****FIRST GUESS METHOD #2*****
551      C
552      C      APPROXIMATELY PLANAR TRIANGLE FIRST GUESS. STATE-
553      C      MENTS 610-624.
554      C
555      610 IF (NSIGN .EQ. 1) GO TO 612
556      IF (NSIGN .EQ. 2) GO TO 616
557      612 PHIN = PHIE + ABS( XLATE * (PI/180.0) * TANE )
558      PHIG = PHIN
559      GO TO 620
560      616 PHIN = PHIE + ABS( XLATE * (PI/180.0) / TANE )
561      PHIG = PHIN
562      GO TO 620
563      620 ICONVX = 2
564      CALL SPHGLO (ICONVX, DUMLAT, XLNGG, DUMTHE, PHIG)
565      C      IF (NDEBUG .GT. 1) WRITE (6,6050) PHIE, XLATE, PHIN
566      C6050 FORMAT (1H, T4, 'PHIE,XLATE,PHIN = ', T36, 3F10.4,
567      C      ' RAD,DEG,RAD')
568      C      IF (NDEBUG .GT. 1) WRITE (6,6060) XLNGG, PHIG
569      C6060 FORMAT (1H, T4, 'FIRST GUESS LONGITUDE & AZIMUTH = ',
570      C      T36, 2F10.4, ' DEG,RAD')
571      624 GO TO 750
572      C
573      C
574      C      *****FIRST GUESS METHOD #3*****
575      C
576      C      EXACT SPHERICAL TRIANGLE FIRST GUESS. STATEMENTS 625-700.
577      C
578      C      LATITUDE XLAT OF THE INTERMEDIATE POINT GIVES THE ANGULAR
579      C      DISTANCE DE-PRIME IN RADIAN.
580      C
581      625 DEPRIM = XLAT * (PI/180.0)
582      C      IF (NDEBUG .GT. 1) WRITE (6,6070) DEPRIM
583      C6070 FORMAT (1H, T4, 'DEPRIM = ', T36, F10.4, ' RAD')
584      C
585      C      LAW OF SINES FOR SPHERICAL TRIANGLES GIVES SOLUTION FOR ANGULAR
586      C      DISTANCE APRMD(1) AND APRMD(2) IN RADIAN. APRMD(2) IS
587      C      (180.0-APRMD(1)) AND HAS THE SAME SINE AS APRMD(1). THIS PRO-
588      C      CEDURE IS NECESSARY BECAUSE THE ARC SINE FUNCTION RETURNS
589      C      ONLY THE PRINCIPAL ANGLE.
590      C
591      SRATIO = SIN(DEPRIM) / SAPRIM
592      C
593      C      IF THE ABSOLUTE VALUE OF SRATIO EXCEEDS THE LEGAL RANGE, THEN
594      C      THE SOLUTION IN LONGITUDE IS INDETERMINATE FOR A GIVEN LATI-
595      C      TITUDE. IN SUCH A CASE, THE METHOD MAY BE SAID TO HAVE FAILED
596      C      TO CONVERGE, AND ERROR (ICONV = 11) IS ASSIGNED.
597      C
598      IF ( ABS(SRATIO) .LE. 1.0) GO TO 626
599      ICONV = 11
600      GO TO 1010
601      626 APRMD(1) = ARSIN ( SRATIO )
602      APRMD(2) = PI - APRMD(1)
603      C      IF (NDEBUG .GT. 1) WRITE (6,6080) APRMD(1), APRMD(2)
604      C6080 FORMAT (1H, T4, 'APRMD(1),APRMD(2) = ', T36, 2F10.4, ' RAD')
605      C
606      C      INVERSE OF FORMULA FOR GREAT CIRCLE DISTANCE (SEE SUBROUTINE
607      C      'DISTAN') GIVES FIRST GUESSES XLNGGG(I), I = 1,2, FOR LATITUDE
608      C      OF INTERMEDIATE POINT IN DEGREES. THE FACT THAT THERE ARE
609      C      TWO XLNGGG VALUES EMERGES FROM THE FACT THAT THERE ARE TWO
610      C      APRMD VALUES. SIGN MULTIPLIER TRIDIR (-1.0 OR 1.0) CAUSES
611      C      ADDITION OF LONGITUDE ELEMENT FOR THE CASE WHERE POINT
612      C      'A-PRIME' IS EAST OF POINT 'A' (SPHERICAL TRIANGLE OPENS

```

```

613 C WESTWARD). OTHERWISE, LONGITUDE ELEMENT IS SUBTRACTED.
614 C
615 C SPECIAL CODE IS NEEDED FOR THE CASE OF AN EQUATOR CROSSING.
616 C THE CODE IMMEDIATELY FOLLOWING AND EXTENDING THROUGH STATE-
617 C MENT 629 ACTS TO REVERSE THE DIRECTION OF THE SPHERICAL
618 C TRIANGLE AT THE POINT WHERE AN AIRCRAFT PATH CROSSES THE
619 C EQUATOR SOUTHWARD FROM THE NORTHERN HEMISPHERE OR NORTHWARD
620 C FROM THE SOUTHERN HEMISPHERE. ONLY IF NSIGN EQUALS 2 WILL
621 C AN EQUATOR CROSSING OCCUR. AN EQUATOR CROSSING IS DECLARED
622 C WHEN THE CURRENT POINT XLATC IS 0.0 AND THE NEXT-POINT IS
623 C NEGATIVE (FOR A SOUTHWARD CROSSING) OR POSITIVE (FOR A
624 C NORTHWARD CROSSING). WHETHER A FLIGHT IS PROCEEDING NORTH-
625 C WARD OR SOUTHWARD IS INDICATED BY THE XNORTH FLAG.
626 C
627 IF (NTRIFG .EQ. 1) GO TO 629
628 IF (NSIGN .EQ. 1) GO TO 629
629 IF (XNORTH) 627,628,628
630 627 IF (XLATC .NE. 0.0) GO TO 629
631 IF (XLAT .GE. 0.0) GO TO 629
632 TRIDIR = -TRIDIR
633 NTRIFG = 1
634 GO TO 629
635 628 IF (XLATC .NE. 0.0) GO TO 629
636 IF (XLAT .LE. 0.0) GO TO 629
637 TRIDIR = -TRIDIR
638 NTRIFG = 1
639 629 CONTINUE
640 C IF (NDERUG .GT. 1) WRITE (6,6090) TRIDIR
641 C6090 FORMAT (1H , T4, 'TRIDIR = ', T36, F10.4)
642 C
643 DO 640 I = 1,2
644 630 XLNGGG(I) = XLNGAP + TRIDIR * ( (180.0/PI) *
645 X ARCOS ( COS(APRMD(I)) / COS( XLAT * (PI/180.0) ) ) )
646 C
647 C ADJUST COMPUTED LONGITUDE XLNGG FOR DATELINE FOLD.
648 C
649 IF (XLNGGG(I) .GT. 180.0) GO TO 632
650 IF (XLNGGG(I) .LT. -180.0) GO TO 634
651 GO TO 636
652 632 XLNGGG(I) = XLNGGG(I) - 360.0
653 GO TO 636
654 634 XLNGGG(I) = XLNGGG(I) + 360.0
655 GO TO 636
656 C
657 C CONVERT LONGITUDES XLNGGG(I) IN DEGREES TO AZIMUTHS PHIGG(I) IN
658 C RADIANS.
659 C
660 636 ICONVX = 1
661 CALL SPHGLD (ICONVX, DUMLAT, XLNGGG(I), DUMTHE, PHIGG(I))
662 C IF (NDERUG .GT. 1) WRITE (6,6100) XLNGGG(I)
663 C6100 FORMAT (1H , T4, 'XLNGGG(I) = ', T36, F10.4, ' DEG')
664 640 CONTINUE
665 C
666 C INITIALLY, WHICHEVER LONGITUDE XLNGGG(I) HAS AN ASSOCIATED
667 C AZIMUTH PHIGG(I) ABSOLUTELY CLOSER TO PHIC, THE AZIMUTH
668 C ANGLE OF THE CURRENT POINT, IS SELECTED AS THE FIRST GUESS
669 C LONGITUDE XLNGG IN DEGREES WITH CORRESPONDING AZIMUTH PHIG
670 C (POINTER IS SAVED IN IGOOD). IF XEAST IS GREATER THAN
671 C ZERO (LESS THAN ZERO), PHIG MUST BE GREATER THAN (LESS
672 C THAN) PHIC. IF THIS IS NOT THE CASE, THE ALTERNATE LONGI-
673 C TUDE SOLUTION IS USED AS FIRST GUESS (POINTERS IGOOD AND
674 C IBAD ARE SWAPPED).
675 C
676 C IF (NDERUG .GT. 1) WRITE (6,6110) PHIGG, PHIC, XEAST
677 C6110 FORMAT (1H , T4, 'PHIG(1)-AND-(2),PHIC,XEAST = ', T36, 4F10.4)
678 IGOOD = 1
679 IBAD = 2
680 IF ( (ABS(PHIGG(IBAD) - PHIC)) .GT. (ABS(PHIGG(IGOOD) - PHIC)) )

```

```

081      &      GO TO 645
082      ITEMP = IG00D
083      IG00D = IBAD
084      I3AD = ITEMP
085      645 IF (PHIGG(IG00D) - PHIC) 650,652,652
086      650 IF (XEAST) 660,654,654
087      652 IF (XEAST) 654,654,660
088      654 ITEMP = IG00D
089      IG00D = IBAD
090      IBAD = ITEMP
091      660 CONTINUE
092      C      IF (NDERUG .GT. 1) WRITE (6,6120) XLNGGG(IG00D), PHIGG(IG00D),
093      C      XLNGGG(IBAD), PHIGG(IBAD)
094      6120 FORMAT (1H , T4, 'XLNGGG,PHIGG FOR IG00D = ', T36, 2F10.4/1X,
095      C      T4, 'XLNGGG,PHIGG FOR IBAD = ', T36, 2F10.4)
096      C
097      C      SET FIRST GUESS LONGITUDE AND PHI.
098      C
099      XLNGG = XLNGGG(IG00D)
100      PHIG = PHIGG (IG00D)
101      C
102      C      STORE PHIG IN PHIN.
103      C
104      690 PHIN = PHIG
105      700 GO TO 750
106      C
107      C
108      C      *****NEWTON'S ITERATIVE TECHNIQUE*****
109      C
110      C      IMPROVE BY NEWTON'S METHOD.
111      C
112      C      INITIALIZE ITERATION COUNT AND DENOMINATOR.
113      C
114      750 N = 0
115      NP1 = N + 1
116      TANTHE = TAN(THETA)
117      C
118      C      FUNCTION.
119      C
120      775 FPHI = QTX * COS(PHIN) + QTY * SIN(PHIN) + QTZ / TANTHE
121      C
122      C      FIRST DERIVATIVE OF FUNCTION.
123      C
124      FPPHI = -QTX * SIN(PHIN) + QTY * COS(PHIN)
125      C
126      C      IMPROVED PHI.
127      C
128      PHINP1 = PHIN - (FPHI / FPPHI)
129      C
130      C      TEST FOR CONVERGENCE. STOP WITH ERROR CONDITION ICONV = 11 IF
131      C      METHOD FAILS TO CONVERGE IN NMAX ITERATIONS.
132      C
133      DELPHI = ABS(PHINP1 - PHIN)
134      IF (DELPHI - EPS) 850,850,800
135      800 PHIN = PHINP1
136      N = N + 1
137      NP1 = N + 1
138      IF (N .LT. NMAX) GO TO 775
139      ICONV = 11
140      GO TO 1010
141      850 PHI = PHINP1
142      C

```

```

743      C
744      C
745      C
746      C
747      C
748      C
749      C
750      C
751      900 ICONVX = 2
752      CALL SPHGLO (ICONVX, XLAT, XLNG, THETA, PHI)
753      IF (ICONV .EQ. 2) GO TO 230
754      C
755      C
756      C
757      C
758      C
759      C
760      C
761      RETURN
762      C
763      C
764      C
765      C
766      C
767      C
768      C
769      1000 WRITE (6,6130) FRMLAT, FRMLNG, TOLAT, TOLNG
770      6130 FORMAT (1H0, '****WIND FACTOR ERROR DIAGNOSTICS****'/1X,
771      &      'FROM-LAT   = ', F10.2, T30, 'FROM-LNG = ', T41, F10.2//1X,
772      &      'TO-LAT    = ', F10.2, T32, 'TO-LNG  = ', T41, F10.2//)
773      CALL FXEM (61, 'SUBROUTINE GRTCIR...ILLEGAL ICONV', 9)
774      GO TO 1050
775      1010 CONTINUE
776      GO TO 1050
777      1020 WRITE (6,6130) FRMLAT, FRMLNG, TOLAT, TOLNG
778      CALL FXEM (61, 'SUBROUTINE GRTCIR...SOLUTION NOT UNIQUE IN THETA',
779      &      12)
780      1050 IF (ICONVS .EQ. 2 .OR. ICONVS .EQ. 3) GO TO 1060
781      IF (ICONVS .EQ. 4 .OR. ICONVS .EQ. 5) GO TO 1070
782      1060 XLAT = 0.0
783      THETA = 0.0
784      RETURN
785      1070 XLNG = 0.0
786      PHI = 0.0
787      RETURN
788      END

```

```

1  C$PHGLO CONV SPH<>GLOBAL/R. C. WHITON/05 FEB 1979
2  C
3      SUBROUTINE SPHGL0 (ICONV, XLAT, XLON, THETA, PHI)
4  C
5  C*****
6  C*
7  C*  PROGRAM ID-      SPHGL0
8  C*  MET ANALYST-    MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
9  C*  SYS ANALYST-    MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
10 C*  PROGRAMMER-     MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
11 C*
12 C*  CREATED ON-      05 FEB 1979          PROJECT- 192301
13 C*
14 C*  DESCRIPTION-     THIS SUBROUTINE SUBPROGRAM CONVERTS GLOBAL COOR-
15 C*                   DINATES (LAT,LON) TO MATHEMATICAL SPHERICAL COOR-
16 C*                   DINATES (THETA,PHI), OR VICE-VERSA, DEPENDING ON
17 C*                   THE VALUE OF THE ICONV FLAG (ICONV=1 FOR CONVER-
18 C*                   SION OF (LAT,LON) TO (THETA,PHI), AND ICONV=2 FOR
19 C*                   CONVERSION OF (THETA,PHI) TO (LAT,LON).
20 C*
21 C*  METHOD-           FOR CONVERSION OF (LAT,LON) TO (THETA,PHI)...
22 C*
23 C*                   THETA = (PI/2.0) - XLAT
24 C*                   PHI   = -XLON          FOR XLON .LE. 0.0
25 C*                   PHI   = (2.0*PI) - XLON FOR XLON .GT. 0.0
26 C*                   IF (PHI .GT. PI) PHI = PHI - (2.0*PI)
27 C*
28 C*                   FOR CONVERSION OF (THETA,PHI) TO (LAT,LON)...
29 C*
30 C*                   XLAT = (PI/2.0) - THETA
31 C*                   XLON = -PHI          FOR PHI .LT. PI
32 C*                   XLON = (2.0*PI) - PHI FOR PHI .GE. PI
33 C*
34 C*                   FOR ABS(PHI) .LT. 2.0*PI
35 C*
36 C*                   NOTE THAT SOUTH LATITUDES AND EAST LONGITUDES ARE
37 C*                   HANDLED AS NEGATIVE NUMBERS AND MUST BE SO SPECI-
38 C*                   FIED ON INPUT/OUTPUT FROM THE SUBROUTINE. LATI-
39 C*                   TUDES AND LONGITUDES ARE IN DEGREES, WHILE THETA
40 C*                   AND PHI ARE IN RADIANs. THE AZIMUTH ANGLE PHI IS
41 C*                   THE ANGLE COUNTERCLOCKWISE FROM GREENWICH. THE
42 C*                   ABSOLUTE VALUE RANGES FROM 0.0 TO 2.0*PI RADIANs.
43 C*                   NEGATIVE VALUES INDICATE CLOCKWISE. THE COLATI-
44 C*                   TUDE ANGLE THETA IS THE ANGLE SOUTHWARD FROM THE
45 C*                   NORTH POLE AND RANGES FROM 0.0 TO PI RADIANs.
46 C*
47 C*  INPUT &          ICONV = FLAG CONTROLLING WHETHER CONVERSION IS
48 C*  OUTPUT-          FROM (LAT,LON) TO (THETA,PHI) (ICONV=1)
49 C*                   OR FROM (THETA,PHI) TO (LAT,LON)
50 C*                   (ICONV=2). WHEN RETURNED AS OUTPUT,
51 C*                   ICONV = 1 OR 2 FOR NORMAL TERMINA-
52 C*                   TION OR A NUMBER GREATER THAN OR
53 C*                   OR EQUAL TO 10 FOR ABNORMAL TERMINA-
54 C*                   TION. ABNORMAL TERMINATION CODES
55 C*                   FOR ICONV ARE...
56 C*
57 C*                   10- ERRONEOUS ICONV SPECIFIED
58 C*                   11- OUT-OF-BOUNDS XLAT
59 C*                   12- OUT-OF-BOUNDS XLON
60 C*                   13- OUT-OF-BOUNDS THETA
61 C*                   14- OUT-OF-BOUNDS PHI
62 C*
63 C*                   XLAT = LATITUDE IN DEGREES, 0.0 TO 90.0,
64 C*                   POSITIVE FOR NORTH LATITUDES AND NEGA-
65 C*                   TIVE FOR SOUTH LATITUDES.
66 C*
67 C*                   XLON = LONGITUDE IN DEGREES, 0.0 TO 180.0, POSI-
68 C*                   TIVE FOR WEST LONGITUDES AND NEGATIVE FOR

```

```

69      C*                                     EAST LONGITUDES.
70      C*
71      C*          THETA  = COLATITUDE ANGLE IN RADIANS, 0.0 TO PI
72      C*                      RADIANS SOUTHWARD FROM NORTH. POSITIVE.
73      C*
74      C*          PHI    = AZIMUTH ANGLE IN RADIANS, 0.0 TO PI
75      C*                      RADIANS, COUNTERCLOCKWISE FROM GREENWICH
76      C*                      LOOKING DOWN AT THE NORTH POLE. NEGATIVE
77      C*                      VALUES INDICATE CLOCKWISE.
78      C*
79      C*  SYSTEM SUB-
80      C*  PROGRAMS
81      C*  USED-          NONE
82      C*
83      C*  USER SUB-
84      C*  PROGRAMS
85      C*  USED-          NONE
86      C*
87      C*  ESTIMATED
88      C*  CPU TIME-      TO BE SUPPLIED.
89      C*
90      C*  STORAGE
91      C*  REQUIREMENTS-  PROCEDURE PLUS DATA OCCUPY 154 WORDS OF
92      C*                      CORE STORAGE.
93      C*
94      C*  PROGRAM
95      C*  UPDATES-      NONE
96      C*
97      C*****
98      C
99      C      DATA TWOPI/6.2831853/,    PI/3.1415927/,  PID2/1.5707963/,
100     &      DTOR/0.01745329/,  RTGD/57.295780/
101     C
102     C      DEPENDING ON THE VALUE OF THE ICONV FLAG, CONVERT (LAT,LON)
103     C      TO (THETA,PHI) (ICONV=1) OR CONVERT (THETA,PHI) TO
104     C      (LAT,LON) (ICONV=2). FOR INPUT ICONV OTHER THAN 1 OR 2,
105     C      AN ERROR CONDITION (ICONV=10) IS RETURNED.
106     C
107     C      IF (ICONV .EQ. 1 .OR. ICONV .EQ. 2) GO TO 100
108     C      ICONV = 10
109     C      GO TO 1000
110     C      100 GO TO (200, 400), ICONV
111     C
112     C      CONVERT (LAT,LON) TO (THETA,PHI). RETURN ERROR CONDITION
113     C      ICONV=11 FOR OUT-OF-BOUNDS XLAT AND ICONV=12 FOR OUT-OF-
114     C      BOUNDS XLON.
115     C
116     C      200 IF (XLAT .LE. 90.0 .AND. XLAT .GE. -90.0) GO TO 220
117     C      ICONV = 11
118     C      GO TO 1000
119     C      220 IF (XLON .LE. 180.0 .AND. XLON .GE. -180.0) GO TO 250
120     C      ICONV = 12
121     C      GO TO 1000
122     C      250 THETA = PID2 - (XLAT * DTOR)
123     C      IF (XLON) 260,260,275
124     C      260 PHI = -(XLON * DTOR)
125     C      IF (PHI .GT. PI) PHI = PHI - TWOPI
126     C      GO TO 800
127     C      275 PHI = TWOPI - (XLON * DTOR)
128     C      IF (PHI .GT. PI) PHI = PHI - TWOPI
129     C      GO TO 800
130     C
131     C      CONVERT (THETA,PHI) TO (LAT,LON). RETURN ERROR CONDITION
132     C      ICONV=13 FOR OUT-OF-BOUNDS THETA AND ICONV=14 FOR OUT-OF-
133     C      BOUNDS PHI. ADJUST COMPUTED LONGITUDE FOR DATELINE FOLD.
134     C
135     C      400 IF (ABS(THETA) .LE. PI) GO TO 420
136     C      ICONV = 13

```

```

137      GO TO 1000
138      420 IF (ABS(PHI) .LE. TWOPI) GO TO 450
139          ICONV = 14
140          GO TO 1000
141      450 XLAT = RTOD * (PID2 - THETA)
142          IF (PHI .LT. PI) GO TO 475
143          XLON = RTOD * (TWOPI - PHI)
144          GO TO 500
145      475 XLON = -(PHI * RTOD)
146      500 IF (XLON .GT. 180.0) XLON = XLON - 360.0
147          IF (XLON .LT. -180.0) XLON = XLON + 360.0
148      C
149      C      NORMAL TERMINATION.
150      C
151      800 RETURN
152      C
153      C      ABNORMAL TERMINATION.
154      C
155      1000 XLAT = 0.0
156          XLON = 0.0
157          THETA = 0.0
158          PHI = 0.0
159          RETURN
160          END

```

```

1  CDISTAN GRT CIRCLE DIST/R. C. WHITON/26 OCT 1978
2  C
3      SUBROUTINE DISTAN (FRMLAT, FRMLNG, TOLAT, TOLNG, GCD)
4  C
5  C*****
6  C*
7  C* PROGRAM ID- DISTAN
8  C* MET ANALYST- MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
9  C* SYS ANALYST- MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
10 C* PROGRAMMER- MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
11 C*
12 C* CREATED ON- 26 OCT 1978 PROJECT- 192301
13 C*
14 C* DESCRIPTION- THIS SUBROUTINE SUBPROGRAM COMPUTES GREAT CIRCLE
15 C* DISTANCE OVER SPHERE OF THE EARTH BETWEEN TWO
16 C* POINTS 'A' AND 'B' WHOSE (LATITUDE, LONGITUDE) IS
17 C* SPECIFIED IN INPUT ARGUMENTS.
18 C*
19 C* METHOD- THE CONVENTIONAL FORMULA FOR THE GREAT CIRCLE ARC
20 C* GCD IN RADIANS IS USED (REF 2), NAMELY,
21 C*
22 C*  $GCD(RADIANS) = \arccos(\sin(FRMLAT) * \sin(TOLAT) +$ 
23 C*  $\cos(FRMLAT) * \cos(TOLAT) * \cos(TOLNG - FRMLNG))$ 
24 C*
25 C* WHERE INDEPENDENT VARIABLES ARE AS DESCRIBED IN
26 C* THE INPUT ARGUMENTS BELOW. LATER, GCD IN RADIANS
27 C* OF ARC IS CONVERTED TO GCD IN NAUTICAL MILES (NM)
28 C* OF CIRCULAR DISTANCE BY MULTIPLYING BY THE
29 C* RADIUS OF THE EARTH, 3440 NM. GCD IS RETURNED TO
30 C* THE MAIN PROGRAM IN NM.
31 C*
32 C* LIMITATIONS- 1. (LAT, LON) OF DESTINATION MUST NOT BE SAME
33 C* AS (LAT, LON) OF ORIGIN.
34 C*
35 C* REFERENCES- 1. CONRAD AND POLLACK, 1950- METHODS IN CLIMA-
36 C* TOLOGY
37 C* 2. HEWLETT-PACKARD, 1975- HP-65 NAVIGATION PAC 1
38 C*
39 C* INPUT- FRMLAT = LATITUDE OF POINT 'A' (ORIGIN) IN DEG
40 C* FRMLNG = LONGITUDE OF POINT 'A' (ORIGIN) IN DEG
41 C* TOLAT = LATITUDE OF POINT 'B' (DESTINATION) IN
42 C* DEG
43 C* TOLNG = LONGITUDE OF POINT 'B' (DESTINATION) IN
44 C* DEG
45 C*
46 C* NOTE- ALL SOUTH LATITUDES AND EAST LONGITUDES MUST
47 C* BE SUPPLIED AS NEGATIVE NUMBERS.
48 C*
49 C* OUTPUT- GCD = GREAT CIRCLE DISTANCE IN NM OF ARC OVER
50 C* THE SURFACE OF THE EARTH
51 C*
52 C* SYSTEM SUB-
53 C* PROGRAMS
54 C* USED- SIN, COS, ARCCOS
55 C*
56 C* USER SUB-
57 C* PROGRAMS
58 C* USED- NONE
59 C*
60 C* ESTIMATED
61 C* CPU TIME- ON A DEC-10 GENERAL PURPOSE COMPUTER, THIS SUB-
62 C* ROUTINE REQUIRES 2.3 CPU MILLISECONDS PER CALL.
63 C*
64 C* STORAGE
65 C* REQUIREMENTS- PROCEDURE PLUS DATA OCCUPY 94 WORDS OF
66 C* STORAGE.

```

```

67 C*
68 C* PROGRAM
69 C* UPDATES- NONE
70 C*
71 C*****
72 C
73 C   CONSTANTS. DTOR CONVERTS DEGREES TO RADIANS. A IS THE RADIUS OF
74 C   THE EARTH IN NAUTICAL MILES (NM).
75 C
76 C   DATA DTOR/0.01745329/, A/3440.0/
77 C
78 C   COMPUTE GCD IN RADIANS OF ARC. MULTIPLICATION BY DTOR CONVERTS
79 C   DEGREES TO RADIANS.
80 C
81 C   SIN1 = SIN(FRMLAT * DTOR)
82 C   SIN2 = SIN(TOLAT * DTOR)
83 C   COS1 = COS(FRMLAT * DTOR)
84 C   COS2 = COS(TOLAT * DTOR)
85 C   COS3 = COS( (TOLNG * DTOR) - (FRMLNG * DTOR) )
86 C   GCR = SIN1 * SIN2 + COS1 * COS2 * COS3
87 C   GCD = ARCOS(GCR)
88 C
89 C   CONVERT GCD FROM RADIANS OF ARC TO CIRCULAR DISTANCE BY MULTIPLY-
90 C   ING BY THE RADIUS OF THE EARTH A, 3440 NM.
91 C
92 C   GCD = GCD * A
93 C
94 C   TERMINATION.
95 C
96 C   RETURN
97 C   END

```

```

1  CHDG      INITIAL HEADING/P.L. HEROD/12 FEB 1979
2  C
3  FUNCTION HDG (XLAT1, XLNG1, XLAT2, XLNG2)
4  C
5  C*****
6  C*
7  C*  PROGRAM ID-      HDG
8  C*  MET ANALYST-    CAPT PATRICK L. HEROD, USAFETAC/DNS, EXT 5412
9  C*  SYS ANALYST-    MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
10 C*  PROGRAMMER-     CAPT PATRICK L. HEROD, USAFETAC/DNS, EXT 5412
11 C*
12 C*  CREATED ON-      12 FEB 1979          PROJECT-    192301
13 C*
14 C*  DESCRIPTION-     THIS FUNCTION SUBPROGRAM FINDS THE INITIAL HEADING
15 C*                   (HDG) ALONG A GREAT CIRCLE COURSE FLOWN FROM POINT
16 C*                   'A' (XLAT1,XLNG1) TO POINT 'B' (XLAT2,XLNG2). IT
17 C*                   WAS DEVELOPED FROM THE HEWLETT-PACKARD 65 (HP-65)
18 C*                   AVIATION PAC 1, GREAT CIRCLE NAVIGATION, APRIL
19 C*                   1975.
20 C*
21 C*  METHOD-           THIS FUNCTION SUBPROGRAM FIRST USES INPUT LATITUDES
22 C*                   AND LONGITUDES FOR POINTS 'A' AND 'B' AND CONVERTS
23 C*                   THEM FROM DEGREES TO RADIAN. IT THEN USES THE
24 C*                   HEADING FORMULA FOUND ON PAGE 54 OF THE HP-65 AVIA-
25 C*                   TION PAC 1 FOR CALCULATION OF THE INITIAL HEADING
26 C*                   FROM POINT 'A.' SINCE THE GREAT CIRCLE DISTANCE
27 C*                   (GCD) IS IN THIS HEADING FORMULA, THE FUNCTION SUB-
28 C*                   PROGRAM MAKES A CALL TO SUBROUTINE DISTAN, WHICH
29 C*                   RETURNS THE GCD IN NAUTICAL MILES (NM) FROM POINT
30 C*                   'A' TO POINT 'B.' FUNCTION HDG CONVERTS GCD IN
31 C*                   NM TO GCD IN RADIAN. THE GCD IN RADIAN IS USED IN
32 C*                   THE HDG FORMULA. HDG IS COMPUTED IN RADIAN.
33 C*                   THERE IS A FINAL CONVERSION PROCESS WHICH ENABLES
34 C*                   THIS FUNCTION TO OUTPUT THE INITIAL HEADING IN
35 C*                   DEGREES.
36 C*
37 C*  LIMITATIONS-     TRUNCATION AND ROUND OFF ERRORS OCCUR WHEN POINT 'A'
38 C*                   AND POINT 'B' ARE VERY CLOSE TOGETHER (1 MILE OR
39 C*                   LESS). INPUT DATA IS IN DECIMAL DEGREES, NOT DE-
40 C*                   GREES, MINUTES AND SECONDS. NORTH LATITUDES AND
41 C*                   WEST LONGITUDES ARE POSITIVE NUMBERS. SOUTH LATI-
42 C*                   TUDES AND EAST LONGITUDES ARE NEGATIVE NUMBERS.
43 C*
44 C*  REFERENCE-        HEWLETT-PACKARD, 1975- AVIATION PAC 1,
45 C*                   GREAT CIRCLE NAVIGATION
46 C*
47 C*  INPUT-            XLAT1  = LATITUDE OF POINT 'A' (ORIGIN) IN DECIMAL
48 C*                   DEGREES
49 C*                   XLNG1  = LONGITUDE OF POINT 'A' (ORIGIN) IN DECIMAL
50 C*                   DEGREES
51 C*                   XLAT2  = LATITUDE OF POINT 'B' (DESTINATION) IN
52 C*                   DECIMAL DEGREES
53 C*                   XLNG2  = LONGITUDE OF POINT 'B' (DESTINATION) IN
54 C*                   DECIMAL DEGREES
55 C*
56 C*  OUTPUT-           HDG    = INITIAL COURSE HEADING ALONG GREAT CIRCLE
57 C*                   FROM POINT 'A' TO POINT 'B' IN DECIMAL
58 C*                   DEGREES TOWARD WHICH
59 C*
60 C*  SYSTEM SUB-
61 C*  PROGRAMS
62 C*  USED-             SIN, COS, ARCOS
63 C*
64 C*  USER SUB-
65 C*  PROGRAMS
66 C*  USED-             DISTAN
67 C*
68 C*  ESTIMATED

```

```

69      C*   CPU TIME-           TO BE SUPPLIED.
70      C*
71      C*   STORAGE
72      C*   REQUIREMENTS-      PROCEDURE AND DATA OCCUPY 100 WORDS OF CORE
73      C*
74      C*
75      C*   PROGRAM
76      C*   UPDATES-           NONE
77      C*
78      C*****
79      C
80      DATA A/3440.0/, PI/3.1415927/, DTOR/0.01745329/
81      C
82      C   CONVERT LATITUDE AND LONGITUDE FROM DEGREES TO RADIANS.
83      C
84      XLAT1R = XLAT1 * DTOR
85      XLNG1R = XLNG1 * DTOR
86      XLAT2R = XLAT2 * DTOR
87      XLNG2R = XLNG2 * DTOR
88      C
89      C   CALCULATE GCD IN NAUTICAL MILES (GCDNM) FROM (XLAT1,XLNG1) TO
90      C   (XLAT2,XLNG2).
91      C
92      CALL DISTAN (XLAT1, XLNG1, XLAT2, XLNG2, GCDNM)
93      C
94      C   CONVERT GCD IN NAUTICAL MILES (GCDNM) TO GCD IN RADIANS (GCDR).
95      C
96      GCDR = GCDNM / A
97      C
98      C   CALCULATE HEADING IN RADIANS (HDGR) FROM HEWLETT-PACKARD 65
99      C
100     HDGR = ARCOS(((SIN(XLAT2R) - (COS(GCDR) * SIN(XLAT1R))) /
101     &      (SIN(GCDR) * COS(XLAT1R)))
102     IF (SIN((XLNG1 - XLNG2) * DTOR) .LT. 0.0) HDGR = 2.0 * PI - HDGR
103     C
104     C   CONVERT HEADING IN RADIANS (HDGR) TO HEADING IN NAUTICAL MILES
105     C   (HDG).
106     C
107     HDG = HDGR / DTOR
108     C
109     C   NORMAL TERMINATION.
110     C
111     RETURN
112     END

```

```

1  CBRNG BRACKET LONGITUDE/P. L. HEROD/17 JAN 1979
2  C
3      SUBROUTINE BRLNG (XLNG, XLNGLO, XLNGHI)
4  C
5  C*****
6  C*
7  C* PROGRAM ID- BRLNG
8  C* MET ANALYST- CAPT PATRICK L. HEROD, USAFETAC/DNS, EXT 5412
9  C* SYS ANALYST- MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
10 C* PROGRAMMER- CAPT PATRICK L. HEROD, USAFETAC/DNS, EXT 5412
11 C*
12 C* CREATED ON- 17 JAN 1979 PROJECT 192301
13 C*
14 C* DESCRIPTION- THIS SUBROUTINE SUBPROGRAM FINDS THE LONGITUDINAL
15 C* VALUES THAT BRACKET A GIVEN LONGITUDE. THE LONGITUDINAL BRACKET VALUES ARE IN INCREMENTS OF 30.0
16 C* I.E., A GIVEN LONGITUDE OF 132.35 HAS BRACKET VALUES
17 C* OF 120.0 AND 150.0.
18 C*
19 C*
20 C* METHOD- THE METHOD USED TO COMPUTE THE BRACKET LONGITUDE
21 C* VALUES IS A SIMPLE ALGEBRAIC ALGORITHM INVOKING
22 C* INTEGER ALGEBRA. SEPARATE EQUATIONS ARE REQUIRED
23 C* DEPENDING ON WHETHER WEST (POS) OR EAST (NEG)
24 C* LONGITUDES ARE USED.
25 C*
26 C* LIMITATIONS- ANY LONGITUDE VALUE FROM 0.00 TO 179.99 CAN BE
27 C* USED AS INPUT BUT THE BRACKET LONGITUDE VALUES
28 C* WILL ALWAYS BE 30.0 TIMES SOME INTEGER NUMBER
29 C* (0.0, 30.0, 60.0, ETC.).
30 C*
31 C*
32 C* IN THE CASE WHERE THE INPUT VALUE IS A BRACKET
33 C* LONGITUDE AND A POSITIVE NUMBER, THE RETURNED
34 C* LONGITUDE BRACKET VALUES WILL BE THE INPUT
35 C* VALUE PLUS 30.0. IF THE INPUT VALUE IS A BRACKET
36 C* VALUE AND A NEGATIVE NUMBER, THE RETURNED LONGITUDE
37 C* BRACKET VALUES WILL BE THE INPUT VALUE AND THE
38 C* INPUT VALUE MINUS 30.0. IF PLUS/MINUS 180.0 IS THE
39 C* INPUT VALUE, THE RETURNED LONGITUDE BRACKET VALUES
40 C* ARE FICTITIOUS (PLUS/MINUS 180.0 AND PLUS/MINUS
41 C* 210.0).
42 C* INPUT- XLNG = LONGITUDE FOR WHICH BRACKETING LONGITUDE
43 C* VALUES ARE DESIRED.
44 C*
45 C* OUTPUT- XLNGLO = LOWER BRACKETING LONGITUDE VALUE
46 C*
47 C* XLNGHI = UPPER BRACKETING LONGITUDE VALUE
48 C*
49 C* SYSTEM SUB-
50 C* PROGRAMS
51 C* USED- IFIX
52 C*
53 C* USER SUB-
54 C* PROGRAMS
55 C* USED- NONE
56 C*
57 C* ESTIMATED
58 C* CPU TIME- TO BE SUPPLIED
59 C*
60 C* STORAGE
61 C* REQUIREMENTS- PROCEDURE PLUS DATA OCCUPY 60 WORDS OF STORAGE.
62 C*
63 C* PROGRAM
64 C* UPDATES- NONE
65 C*
66 C*****
67 C
68 DATA DELTLO/30.0/

```

```

69      C
70      C      BRANCH TO POSITIVE OR NEGATIVE LONGITUDE COMPUTATION.
71      C
72      C      IF (XLNG) 110,105,105
73      C
74      C      LOWER LONGITUDE LIMIT FOR POSITIVE LONGITUDE.
75      C
76      C      105 XLNGLO = IFIX (XLNG/DELTLO) * DELTLO
77      C
78      C      UPPER LONGITUDE LIMIT FOR POSITIVE LONGITUDE.
79      C
80      C      XLNGHI = XLNGLO + DELTLO
81      C      GO TO 120
82      C
83      C      LOWER LONGITUDE LIMIT FOR NEGATIVE LONGITUDE.
84      C
85      C      110 XLNGLO = (IFIX(XLNG/DELTLO) - 1) * DELTLO
86      C
87      C      UPPER LONGITUDE LIMIT FOR NEGATIVE LONGITUDE.
88      C
89      C      XLNGHI = XLNGLO + DELTLO
90      C
91      C      TERMINATION.
92      C
93      C      120 RETURN
94      C      END

```

```

1  CBRLAT  BRACKET LATITUDE/P. L. HEROD/02 JAN 1979
2  C
3      SUBROUTINE BRLAT (XLAT, XLATLO, XLATHI)
4  C
5  C*****
6  C*
7  C*  PROGRAM ID-      BRLAT
8  C*  MET ANALYST-    CAPT PATRICK L. HEROD, USAFETAC/DNS, EXT 5412
9  C*  SYS ANALYST-    MAJ ROGER C. WHITON, USAFETAC/DNS, EXT 5412
10 C*  PROGRAMMER-     CAPT PATRICK L. HEROD, USAFETAC/DNS, EXT 5412
11 C*
12 C*  CREATED ON-      02 JAN 1979          PROJECT 192301
13 C*
14 C*  DESCRIPTION-     THIS SUBROUTINE SUBPROGRAM FINDS THE LATITUDINAL
15 C*                   VALUES IN TENTHS OF DEGREES THAT BRACKET A GIVEN
16 C*                   LATITUDE. THE LATITUDE BRACKET VALUES ARE IN
17 C*                   INCREMENTS OF 15.0, I.E., A GIVEN LATITUDE OF 27.46
18 C*                   HAS BRACKET VALUES OF 15.0 AND 30.0.
19 C*
20 C*  METHOD-           THE METHOD USED TO COMPUTE THE BRACKET LATITUDE
21 C*                   VALUES IS A SIMPLE ALGEBRAIC ALGORITHM INVOKING
22 C*                   INTEGER ALGEBRA. SEPARATE EQUATIONS ARE REQUIRED
23 C*                   DEPENDING ON WHETHER NORTH (POS) OR SOUTH (NEG)
24 C*                   LATITUDES ARE USED.
25 C*
26 C*  LIMITATIONS-     ANY LATITUDE VALUE FROM 0.00 TO PLUS/MINUS 89.99
27 C*                   CAN BE USED AS INPUT BUT THE BRACKET LATITUDE
28 C*                   VALUES WILL BE 15.0 TIMES SOME INTEGER NUMBER
29 C*                   (0.0, 15.0, 30.0, 45.0, ETC.). IN THE CASE WHERE
30 C*                   THE INPUT VALUE IS A BRACKET VALUE AND A POS-
31 C*                   ITIVE NUMBER, THE RETURNED LATITUDE BRACKET
32 C*                   VALUES WILL BE THE INPUT VALUE AND THE INPUT
33 C*                   VALUE PLUS 15.0. IF THE INPUT VALUE IS A BRACKET
34 C*                   VALUE AND A NEGATIVE NUMBER, THE RETURNED LATITUDE
35 C*                   BRACKET VALUES WILL BE THE INPUT VALUE AND THE
36 C*                   INPUT VALUE MINUS 15.0. IF PLUS/MINUS 90.0 IS THE
37 C*                   INPUT VALUE, THE RETURNED LATITUDE BRACKET
38 C*                   VALUES ARE FICTITIOUS (PLUS/MINUS 90.0 AND
39 C*                   PLUS/MINUS 105.0).
40 C*
41 C*  INPUT-            XLAT = LATITUDE FOR WHICH BRACKETING LATITUDE
42 C*                   VALUES ARE DESIRED.
43 C*
44 C*  OUTPUT-           XLATLO = LOWER BRACKETING LATITUDE VALUE
45 C*
46 C*                   XLATHI = UPPER BRACKETING LATITUDE VALUE
47 C*
48 C*  SYSTEM SUB-
49 C*  PROGRAMS
50 C*  USED-             IFIX
51 C*
52 C*  USER SUB-
53 C*  PROGRAMS
54 C*  USED-             NONE
55 C*
56 C*  ESTIMATED
57 C*  CPU TIME-         TO BE SUPPLIED
58 C*
59 C*  STORAGE
60 C*
61 C*  REQUIREMENTS-     PROCEDURE PLUS DATA OCCUPY 58 WORDS OF STORAGE.
62 C*
63 C*  PROGRAM
64 C*  UPDATES-          NONE
65 C*
66 C*****
67 C
68      DATA DELTIA/15.0/

```

```

69      C
70      C      BRANCH TO POSITIVE OR NEGATIVE LATITUDE COMPUTATION.
71      C
72      C      IF (XLAT) 110,105,105
73      C
74      C      LOWER LATITUDE LIMIT FOR POSITIVE LATITUDE.
75      C
76      105 XLATLO = IFIX (XLAT/DELTAL) * DELTAL
77      C
78      C      UPPER LATITUDE LIMIT FOR POSITIVE LATITUDE.
79      C
80      XLATHI = XLATLO + DELTAL
81      GO TO 120
82      C
83      C      UPPER LATITUDE LIMIT FOR NEGATIVE LATITUDE.
84      C
85      110 XLATHI = IFIX (XLAT/DELTAL) * DELTAL
86      C
87      C      LOWER LATITUDE LIMIT FOR NEGATIVE LATITUDE.
88      C
89      XLATLO = XLATHI - DELTAL
90      C
91      C      TERMINATION.
92      C
93      120 RETURN
94      END

```

