NASA CR- 139150

VOLUME 2

Part 1

(NASA-CR-139150) GEODYN PROGRAMMERS GUIDE, VOLUME 2, PART 1 Final Report (Wolf Research and Development Corp.)
367 p CSCL 22A
N75-10974
Unclas
G3/13 03462

GEODYN PROGRAMMER'S GUIDE

Contract No.: NAS 5-11735 - MOD 65
PCN 550-W-72416

Prepared By:

N.E. Mullins T.V. Martin
C.C. Goad N.L. Boulware
N.C. Dao M.M. Chin

Wolf Research and Development Corporation
Riverdale, Maryland

For

Goddard Space Flight Center
Greenbelt, Maryland

October 1972

393

TABLE OF CONTENTS

INTRODUCTION

The Programmer's Guide to GEODYN contains the programming details associated with the GEODYN program. This is the second of four volumes which completely document the GEODYN System and is to be regarded as a programmer's supplement to Volume 1, the Systems Description.

The GEODYN program is the heart of the GEODYN Orbit and Geodetic Parameter Estimation System, as it is responsible for estimating the orbit and geodetic parameters for the System.

SECTION 1.0

INTRODUCTION TO THE GEODYN PROGRAM

The major component of the GEODYN System is the program GEODYN. The GEODYN program estimates orbit and geodetic parameters for the System. It possesses the capability to estimate that set of orbital elements, station positions, measurement biases, and a set of force model parameters such that the orbital tracking data from multiple arcs of multiple satellites best fit the entire set of estimated parameters.

GEODYN currently consists of 113 different program segments, including the main program, subroutines, functions, and block data routines. All are in G or H level FORTRAN and are currently operational on GSFC's IBM 360/95 and IBM 360/91.

Due to the large number of block data routines in GEODYN, each has been assigned a unique name. These are, of course, not FORTRAN names, but are rather a device to enable meaningful discussion.

SECTION 2.0

GEODYN ENVIRONMENTAL REQUIREMENTS

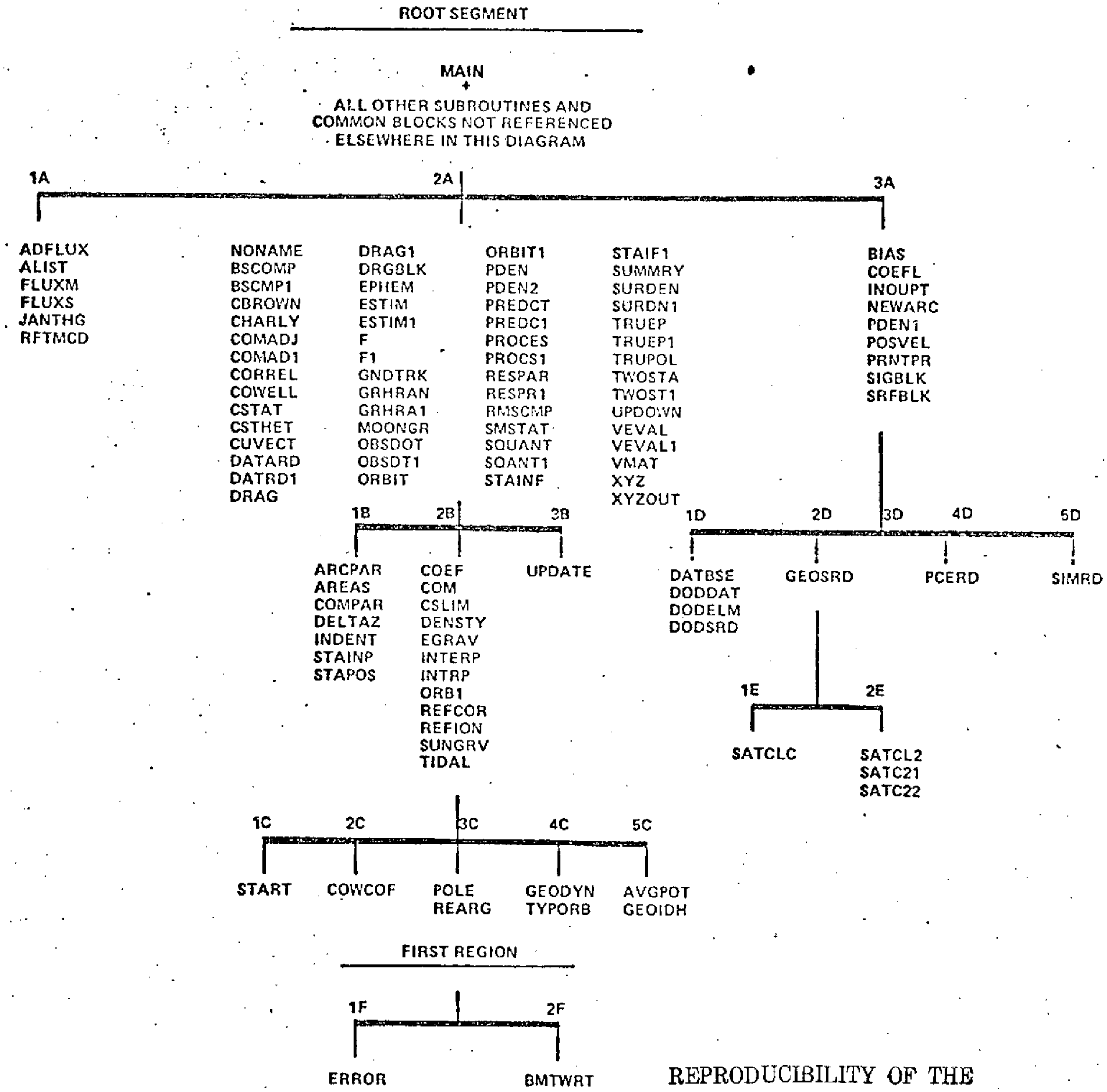
Normal operation of the GEODYN program requires a large scale IBM 360 Computer with a minimum of 400K bytes of users accessable core, one 2314 direct access disk unit, two 9-track tape drives, one high speed card reader, and one high speed printer. For some applications, GEODYN can operate with only one 9-track tape drive.

The current GEODYN program is operational under version 19 of the IBM 360 Operating System on the GSFC IBM 360/95 and version 20 on the GSFC IBM 360/91.

For compilation, GEODYN requires an IBM FORTRAN IV Level G compiler and an IBM FORTRAN IV Level H compiler. A non-GSFC user should ensure that sufficient space is allocated for the desired compiler at SYSGEN time to accomodate the required table space. More efficient operation of GEODYN may be obtained by use of a Level H compiler with level 2 optimization for all subroutines which are not affected by compiler size restrictions.

SECTION 3.0

DIAGRAM OF OVERLAY STRUCTURE



REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

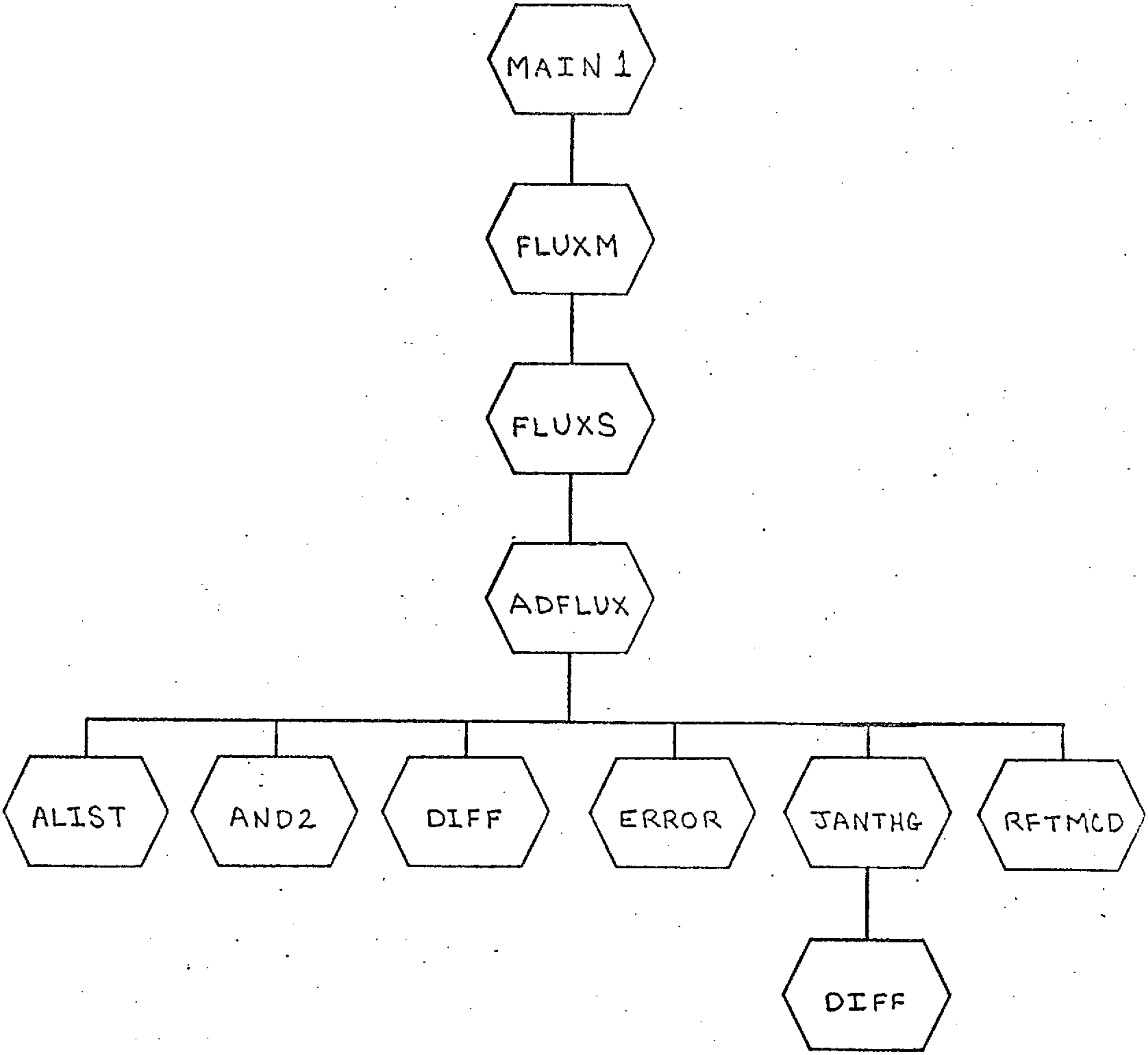
SECTION 4.0

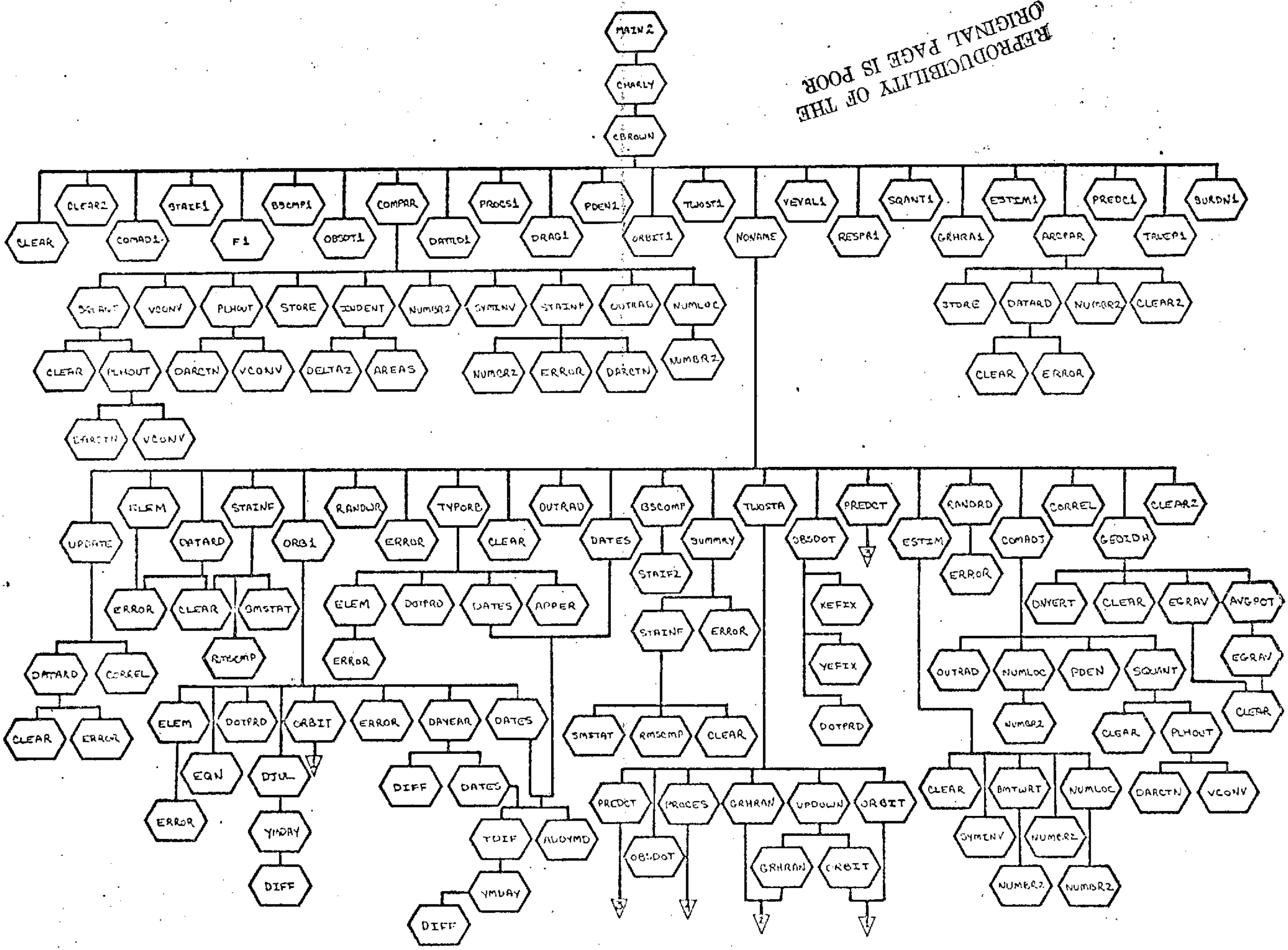
DIAGRAMS OF SUBROUTINE STRUCTURE

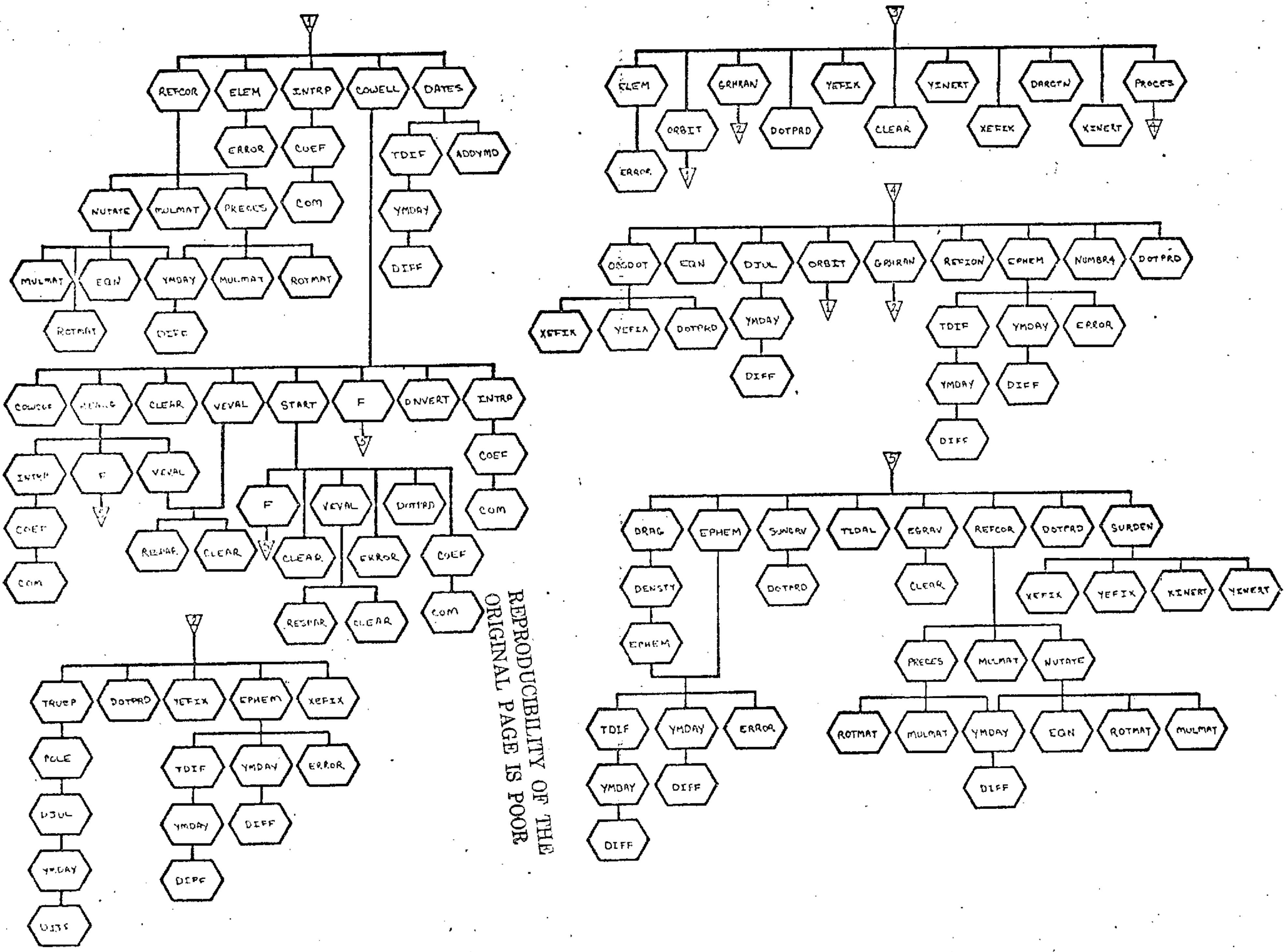
On the following pages appear diagrams depicting the subroutine call structure of the GEODYN program. These diagrams show all possible chains of subroutine calls.

MAIN is the executive routine of the GEODYN program. For ease in explaining the subroutine structure MAIN has been broken down into three parts, MAIN 1, MAIN 2, and MAIN 3.

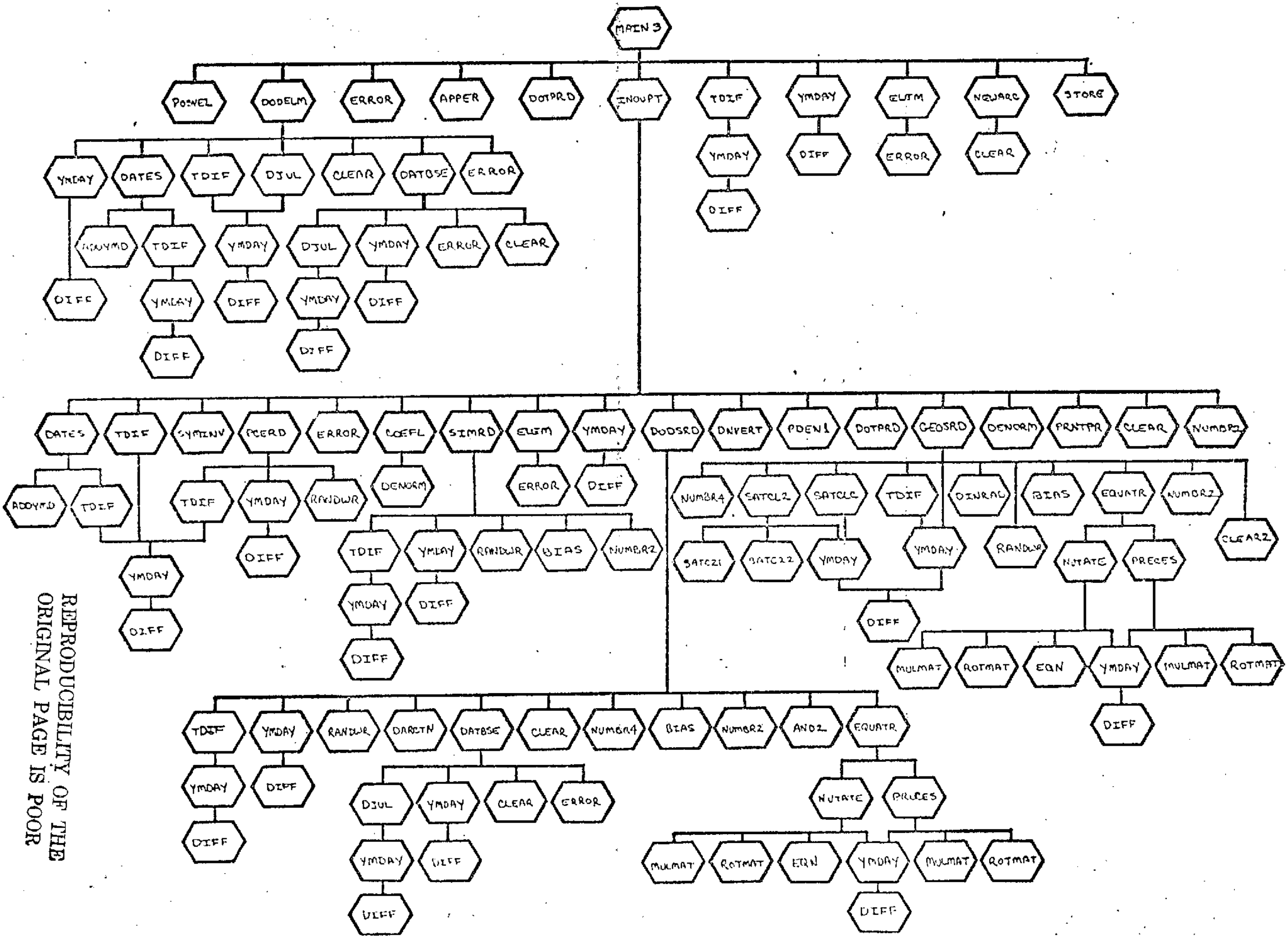
The logic of the diagrams flows down, right or left, but never up. Subroutines which have no extensions from the bottom are on the lowest level. Subroutines may appear at more than one place in the diagrams and, therefore, following a branch backwards does not necessarily determine all subroutines which might call any specific subroutine.


MAIN 1
FLUXM
FLUXS
ADFLUX
ALIST
AND2
DIFF
ERROR
JANTHG
RFTMCD
DIFF


MAIN2
CHARLY
CBROWN
CLEAR
CLEAR2
COMAD1
STAIF1
F1
BSCMP1
OBSDT1
COMPAR
DATMD1
PROCS1
DRAG1
PDEN1
ORBIT1
TWOST1
NONAME
VEVAL1
RESPR1
SQANT1
GRHRA1
ESTIM1
ARCPAR
PREDC1
TALEP1
SURDN1
SOLAVT
VCONV
PLHOUT
STORE
INDENT
NUMBR2
SYMINV
STAINP
OUTRAD
NUMLOC
CLEAR
PLHOUT
DARCTN
VCONV
DELTAZ
AREAS
NUMBR2
ERROR
DARCTN
NUMBR2
DARCTN
VCONV
STORE
DATARD
NUMBR2
CLEAR2
CLEAR
ERROR
UPDATE
ELEM
DATARD
STAINF
ORB1
RANDWR
ERROR
TYPORB
CLEAR
OUTRAD
DATES
BSCOMP
SUMMRY
TWOSTA
OBSDOT
PREDCT
ESTIM
RANDRD
COMADJ
CORREL
GEOIDH
CLEAR2
ERROR
CLEAR
OMSTAT
RMSCMP
ELEM
DOTPRD
DATES
APPER
ERROR
STAIF2
STAINF
ERROR
XEFIX
YEFIX
DOTPRD
3
ERROR
DWVERT
CLEAR
EGRAV
AVGPOT
EGRAV
CLEAR
DATARD
CORREL
CLEAR
ERROR
ELEM
DOTPRD
ORBIT
ERROR
DAYEAR
DATES
SMSTAT
RMSCMP
CLEAR
OUTRAD
NUMLOC
PDEN
SQUANT
NUMBR2
CLEAR
PLHOUT
1
EQN
DJUL
DIFF
DATES
ERROR
YMDAY
DIFF
TDIF
AUDYMD
YMDAY
DIFF
PREDCT
PROCES
GRHRAN
UPDOWN
ORBIT
OBSDOT
GRHRAN
ORBIT
3
1
2
1
CLEAR
BMTWRT
NUMLOC
SYMINV
NUMBR2
NUMBR2
NUMBR2
DARCTN
VCONV



REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR



REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

SECTION 5.0
SUMMARY OF SUBPROGRAMS USED BY GEODYN

MAIN	Reads and calls subroutines to read GEODYN input cards, determines array sizes for variable storage allocation and acts as a driver for all segments of GEODYN.
NONAME	Serves as a driver for orbit generator and data reduction operations and outputs residuals, ephemeris, and adjustments.
ADDYMD	Adds or subtracts an integral number of days from a date in the form YYMMDD giving a new date in the same one-word form.
ADFLUX	Reads flux cards from GEODYN input deck and adds solar and magnetic flux to stored tables. Writes on scratch file flux information for each arc and the Greenwich mean sidereal time on Jan. 0.0 of the reference year. Counts the number of arcs in the run.
ALIST	Lists the GEODYN input card deck.
ALPMRC	Block data storage of alpha-numeric information used by GEODYN print formats.
AND2	Two byte integer 'AND' function.
APPER	Computes apogee and perigee heights of a satellite.

ARCPAR	Loads individual arc parameters into variable storage arrays.
AREAS	Computes the ellipsoidal surface area lying between two fixed latitudes and two fixed longitudes.
AVGPOT	Computes the average gravitational potential of the Earth using only even zonal harmonic coefficients.
BIAS	Extracts bias start and stop times from data and counts biases.
BMTWRT	Writes out the B-matrix.
BSCOMP	Computes electronic biases and corrects normal equations for extraction of the electronic biases.
CBROWN	Passes variable storage arrays to the subroutines using them.
CHARLY	Allocates core for variable storage arrays.
CLEAR	Clears an array of four-byte integers.
CLEAR2	Clears an array of two-byte integers.
COEF	Computes interpolation coefficients.
COEFL	Lists non-zero gravity model coefficients of the spherical harmonic expansion of the geopotential used by GEODYN.
COM	Computes binomial coefficients.
COMADJ	Prints adjustments to common parameters.
COMPAR	Loads common parameters into variable storage arrays.

CONSTS	Block data storage of input/output file numbers, integrator stepsize information, conversion constants, and Earth parameters.
CORREL	Computes and prints correlation coefficients from above diagonal of normal matrix stored in vector form.
COWCOF	Assigns integrator coefficient values (orders 5-15).
COWELL	Integrates satellite equations of motion and force model partial derivatives to desired time.
DATARD	Stores, updates and retrieves individual arc parameter information.
DATBSE	Retrieves observation data from DODS Data Base.
DATES	Converts days elapsed from Jan. 0.0 of the arc reference year applying the transformation from the A.1 time system to the UTC time system, into a three word date of the form YYMMDD, HHMM, SEC.
DAYEAR	Converts a date from the year and number of days from Jan. 0.0 of that year to integral days and fraction of a day in integral seconds. Outputs the date in the form YYMMDD.
DELTAZ	Computes the z-coordinate of a point of given latitude on the ellipsoid.
DENSTY	Computes atmospheric density based on the Jacchia-Nicolet model (height and temperature dependent).
DIFF	Calculates the difference between any two time points in the 20th century. Input date is two words in the form YYMMDD and HHMMSS; output date is integral days and seconds of a day.

DINRAD	Converts angles expressed in arc measurements or time measurements to radians.
DNVERT	Double-precision matrix inversion using Gauss-Jordan method of condensation with partial (column) pivoting. No restrictions on dimension of matrix.
DODELM	Retrieves satellite orbital starting elements from DODS Data Base.
DODSRD	Reads observation data tapes in DODS format and and partially preprocesses the observations.
DPFCT	Computes arctangents, denormalization factor for geopotential coefficients, Julian dates, and dot products. Computes Earth-fixed x and y from inertial x and y, and inertial x and y from Earth-fixed x and y.
DRAG	Computes accleration in rectangular coordinates of a satellite due to aerodynamic drag forces.
EGRAV	Calculates acceleration in rectangular coordinates on a satellite due to geopotential forces (spherical harmonic terms to maximum degree and order 30).
ELEM	Converts inertial position and velocity vector to osculating orbital elements.
EPHEM	Reads and interpolates lunar, solar, and planetary ephemerides and the nutation in right ascension.

EQN	Computes nutation in longitude, obliquity, and right ascension, and the true obliquity of date.
EQUATR	Rotates a vector from the mean or true equator and equinox of one epoch to the mean or true equator and equinox of another epoch.
ERROR	Prints specific error messages when the run is abnormally terminated.
ESTIM	Estimates correction vector to state vector using the method of Bayesian least-squares.
F	Evaluates the satellite acceleration vector and force model derivatives.
FLUXM	Blockdata storage of magnetic flux data through January 1972.
FLUXS	Blockdata storage of solar flux data through February, 1972.
FMODEL	Blockdata storage of the coefficients of the spherical harmonic expansion of the geopotential.
GEODYN	Blockdata storage of date and source tape number of this version of GEODYN.
GEOIDH	Positions surface density locations on geoid surface and computes matrix of constraint for density adjustment.
GEOSRD	Reads observation data in GEOS format and partially preprocesses the observations.

GRHRAN	Computes the right ascension of Greenwich and the satellite vectors used in computing measurement partials.
INDENT	Computes Cartesian coordinates and areas of surface density blocks.
INOUPT	Reads GEODYN input cards. Outputs run and arc descriptions. Calls subroutines to read data tapes.
INTRP	Interpolation routine.
JANTHG	Selects the Greenwich mean sidereal time on Jan. 0.0 of the reference year for each arc and selects flux data for each arc from block-data storage. Computes average solar flux values for each arc.
MULMAT	Multiplies three 3x3 matrices.
NEWARC	Initializes switches and constants for each arc.
NUMBR2	Searches the entries of an array of 2-byte integers and compares them with an input number or bit configuration. The index number or location of the entry matched is returned. If no match is found, zero is returned.
NUMBR4	Searches the entries of an array of 4-byte integers and compares them with an input number or bit configuration. The index number or location of the entry matched is returned. If no match is found, zero is returned.

NUMLOC	Searches the entries of an array and compares them with an input number or bit configuration. Index numbers or locations of the entries which match and the number of such matching entries are returned.
NUTATE	Generates nutation angles to transform a vector from true equator and equinox to mean equator and equinox.
OBSDOT	Calculates time derivatives of requested observation types. (Observation types available: 1 - right ascension and declination; 2 - range; 3 and 4 - range rate; 5 - ℓ and m direction cosines; 6 - X and Y angles; 7 - azimuth and elevation).
ORBIT	Returns satellite state (position and velocity) and force model partials at the called time.
ORB1	Generates a satellite ephemeris tape in ORB1 tape format.
OUTRAD	Converts radians to degrees, minutes and seconds or to hours, minutes and seconds.
PCERD	Reads PCE format data.
PDEN	Prints adjusted surface densities.
PDEN1	Prints input surface densities.

PLHOUT	Converts tracking station location and variance-covariance matrix in geocentric rectangular coordinates to geodetic latitude, longitude and height. Computes partial derivatives of the geodetic coordinates with respect to the geocentric coordinates.
POLE	Selects from a table, for a given input date, the coordinates of the true pole.
POSVEL	Converts osculating orbital elements to inertial position and velocity vectors.
PRECES	Generates the matrix for precession from mean equator and equinox of one epoch to mean equator and equinox of another epoch.
PREDCT	Computes measurements, residuals and measurement partials.
PROCES	Completes the preprocessing of observation measurements.
PRNTPR	Prints the requested observation preprocessing.
RANDOM	Unblocks and reads observations from a random access file. Blocks and writes observations on a random access file.
REARG	Rearranges the back value arrays when the integration step is changed in variable step mode.
REFCOR	Precesses and nutates a vector between the true equator and the equinox of a reference time to the true equator and equinox of date.

REFION	Dummy ionospheric refraction subroutine.
RESPAR	Calculates the partials of geopotential resonance coefficients requesting adjustment.
RFTMCD	Checks an 80-character alphanumeric string to determine if the first 6 characters are numerals and the remaining 74 characters are blank.
RMSCMP	Computes RMS, random normal deviate, and mean from summed information.
ROTMAT	Generates a rotation matrix from an angle and axis of rotation.
SATCLC	Applies satellite clock corrections to GEOS 1 optical data.
SATCL2	Applies satellite clock corrections to GEOS 2 optical data.
SATC21	Part 2 of satellite clock corrections for GEOS 2 (SATCL2).
SATC22	Part 3 of satellite clock corrections for GEOS 2 (SATCL2).
SIMRD	Reads simulated data tapes.
SMSTAT	Sums statistics.

SQUANT	Converts station positions to geocentric rectangular coordinates on the first call. On subsequent calls, converts station positions to latitude, longitude and height. Computes partials of the rectangular coordinates with respect to latitude, longitude and height.
STAINF	Computes statistical information at the end of each inner iteration for each arc and at the end of each outer iteration for all arcs. Corrects statistical information for electronic bias compensation.
STAINP	Reads input station positions. Determines whether position was input in rectangular of spherical coordinates, converting to spherical whenever necessary and stores those station positions to be used in arc.
STAPOS	Blockdata storage of station positions.
START	Starts the integration process using interpolator formulas and iterating until desired accuracy is obtained.
STORE	Stores common and arc information on disk.
SUMMRY	Prints arc statistical summary.
SUNGRV	Computes the acceleration in rectangular coordinates on a body in orbit about a central body due to the gravitational attraction of a disturbing body. (Includes the moon, sun, Venus, Mars, Jupiter and Saturn.)

SURDEN	Computes the gradient of the potential due to surface densities, and the partials of the gradients with respect to the surface densities for those densities to be adjusted.
SYMINV	Symmetric matrix inversion.
TDIF	Computes time differences between systems A.1, UTC, UT1 and UT2.
TIDAL	Computes acceleration due to solid Earth tidal bulges caused by lunar and solar gravitational effects on the Earth.
TRUEP	Rotates tracking station coordinates to account for polar wandering.
TWOSTA	Computes measurements and partials for VLBI and average range rate data.
TYPORB	Prints arc summary page.
UPDATE	Updates arc adjusted parameters to compensate for adjustments to common parameters.
UPDOWN	Computes uplink and downlink transit time for average range rate data or the two downlink transit times for VLBI data.
VCONV	Converts variance-covariance from one system to another.
VEVAL	Computes the variational partials of force model parameters.
YMDAY	Computes for a given date the number of days from Jan 0.0 of the reference year for the arc.

SECTION 6.0

SUBROUTINE CROSS REFERENCE CHART

A cross reference chart of the subroutines used by the GEODYN program is given on the following two pages. The calling routines are listed down the side of the page and the called routines are across the top.

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

CALLED ROUTINES

CALLING ROUTINES

	NONAME	ADDYMD	ADFLUX	ALIST	AND2	APPFR	ARCPAR	AREAS	AVGPOT	BIAS	BMTWRT	BSCOMP	BSTRPT	CBROWN	CHARLY	CLEAR	CLEAR2	COEF	COEFL	COM	COMADJ	COMADT	COMPAR	CORREL	COWCOF	COWELL	DARCIN	DATARD	DATRD1	DATBSF	DATES	DAYEAR	DELTAZ	DENSRM	DENSTY	DIFF	DIRRAD	DJDL	DLVLRT	DODELM	DODSRD	DOTPRD	DPZG	DRAG1	EGRAV	ELEM	EPHEM	EQN	EQUATR	ERROR	ESTIM	ESTIM1		F1	FLUXM	FLUXS	GEOIDH	GEOSRD	GRHRAN	GRHRAT	INDENT	INOUPT	INTRP	JANTHG	MATMAT	NEWARC	NUMDIF
MAIN						○									○																																																				
NONAME												○				○	○				○			○				○			○									○		○				○				○					○							○					
ADFLUX				○	○																																									○				○	○						○										
ARCPAR																	○											○								○														○														○			
AVGPOT																																																																			○
BMTWRT																																													○																						
BSCOMP																																																																			○
CBROWN	○						○						○			○	○					○	○						○															○								○		○						○							
CHARLY														○																																																					
COEF																				○																																															
COEFL																																		○																																	
COMADJ																																																																			
COMPAR																																																													○						○
COWELL																○									○														○													○											○				
DATARD																○																																		○																	
DATBSE																○																						○												○																	
DATES		○																																																																	
DAYEAR																															○					○																															
DENSTY																																																																			
DODELM																○														○	○							○												○																	
DODSRD					○					○						○											○			○																			○	○																	○
DPFCT*																																																																			
DRAG																																			○																																
EGRAV																○																																																			
ELEM																																																		○																	
EPHEM																																																		○																	
EQUATR																																																		○																	
TIM											○					○																																																			○
																																										○	○		○		○																					
FLUXM																																																								○											
FLUXS			○																																																																
GEOIDH									○							○																							○																												
GEOSRD										○							○																				○								○				○																		○
GRHRAN																																										○					○																				
INDENT								○																									○																																		
INOUPT																○			○												○								○		○	○				○				○								○									○
INTRP																		○																																																	
JANTHG																																				○																															
NEWARC																○																																																			
NUMLOC																																																																			
NUTATE																																																																			○
OBSDOT																																										○						○																	○		
ORB1																															○	○						○				○				○		○		○																	
ORBIT																										○					○															○																					
PCERD																																																															○				
PLHOUT																																																																			
POLE																																						○																													
PRECES																																																																			
PREDCT																○											○															○				○													○						○		
PROCES																																						○				○					○	○											○								
RANDOM*																																																		○																	
REARG																																																																			
REFCOR																																																				○											○				
SATCL2																																																																	○		
SATCLC																																																																			
SIMRD										○																																																									
SQUANT																○																																																		○	
STAINF*																○																																																			
STAINP																											○																																								
START																○		○																								○								○		○															○
SUMMRY																																																		○																	
SUNGRV																																																		○																	
SUHDEN																																										○																									
TDIF																																																																			
TRUEP																																																																			
TWOSTA																																																																			
PORB						○																									○											○				○													○								
DATE																								○				○																																							
DOAN																																																											○								
VEVAL																○																																																			
YMDAY																																																																			

*The starred routines have more than one entry point.

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

CALLED ROUTINES

ING ROUTINES

	NUMLOC	NUTATE	OBSDOT	OBSDT1	ORB1	ORBIT	ORBIT1	ORTRAD	PCERD	PDEN	PDEN1	PDEN2	PLHOUT	POLE	POSVEL	PRECES	PREDCT	PREDC1	PRINTPR	PROCES	PROCS1	RANDND	RANDWS	REARG	REFCOR	REFION	RESPAR	RESPRT	RETRGD	RWSCMP	SOLMAT	SATC71	SATCZ2	SATCLC	SATCL2	SIMRD	SQANT1	SQUANT	STAIF1	STAIF2	STAINF	SMSTAT	STAINP	START	STORE	SUMMRY	SUNGRV	SURDEN	SURDN1	SYNTHV	TDIF	TIDAL	TRUEP	TRUEEL	TWOSTA	TWOST1	TYPORB	UPDATE	UPDOWN	VCONV	VEVAL	VEVAL1	VFIX	VGRRT	VFIY	VINFIT	YMDAY
MAIN															○																														○						○																
NONAME			○		○			○									○					○	○																		○					○									○		○	○									
ADFLUX																													○																																						
ARCPAR																																													○																						
AVGPOT																																																																		
BMTWRT																																																																		
BSCOMP																																								○																											
CBROWN				○			○					○									○							○									○		○										○					○		○						○					
CHARLY																																																																		
COEF																																																																		
COEFL																																																																		
COMADJ	○							○		○																												○																													
COMPAR	○							○					○																									○					○		○					○										○							
COWELL																							○																				○																	○						
DATARD																																																																		
DATBSE																																																																		
DATES																																																			○																
DAYEAR																																																																		
DENSTY																																																																		
DODELM																																																			○																
DODSRD																						○																													○																
DPFCT*																																																																		
DRAG																																																																		
EGRAV																																																																		
ELEM																																																																		
EPHEM																																																			○																
EQUATR		○														○																																																			
ESTIM	○																																																	○																	
F																								○																						○	○				○															
FLUXM																																																																		
FLUXS																																																																		
GEOIDH																																																																		
GEOSRD																						○												○	○															○																
GRHRAN																																																				○										○		○		
INDENT																																																																		
INOUPT									○		○								○																														○	○																
INTRP																																																																		
JANTHG																																																																		
NEWARC																																																																		
NUMLOC																																																																		
NUTATE																															○																																				
OBSDOT																																																																		
ORB1						○																																																								○		○		
ORBIT																								○																																									
PCERD																						○																													○																
PLHOUT																																																												○							
POLE																																																																		
PRECES																															○																																				
PREDCT						○														○																																																
PROCES			○			○																																																														
RANDOM*																									○																																										
REARG																																																													○						
REFCOR		○														○																																																			
SATCLZ																																○	○																																			
SATCLC																																																																		
SIMRD																						○																													○																
SQUANT												○																																																						
STAINF*																														○												○																									
STAINP																																																																		
START																																																													○						
SUMMRY																																									○																										
SUNGRV																																																																		
SURDEN																																																																		
TDIF																																																																		
TRUEP													○																																																					
TWOSTA			○			○											○			○																																							○								
TYPORB																																																																		
UPDATE																																																																		
UPDOWN						○																																																													
VEVAL																																																																		
YMDAY																																																																		

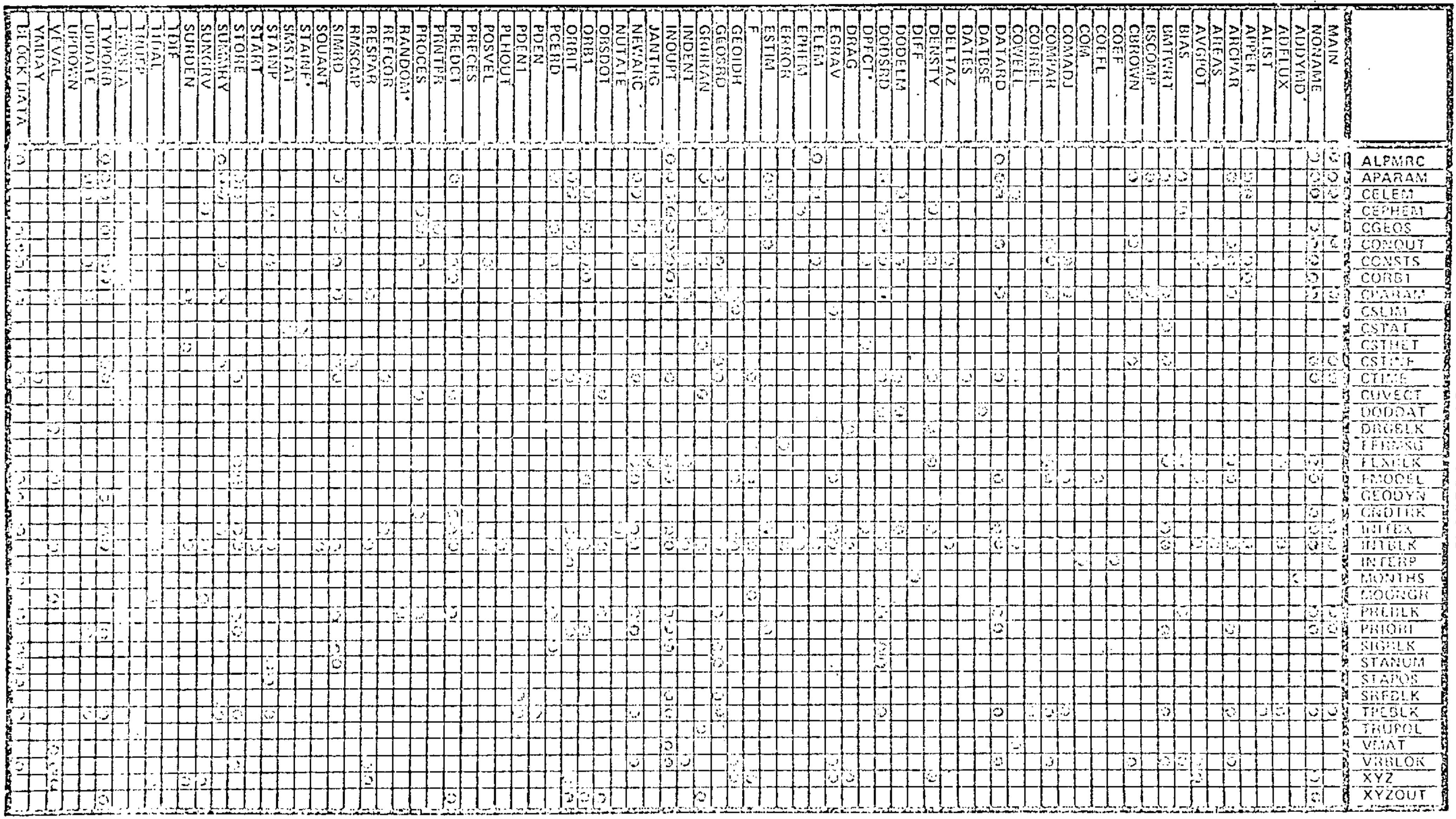
*The starred routines have more than one entry point.

SECTION 7.0

COMMON BLOCK CROSS REFERENCE CHART

On the following page is a cross-reference chart showing the common blocks used in the GEODYN program and which routines use them. The subroutines are listed down the side of the page and the common blocks are across the top.

SUBROUTINES



BLOCK DATA, YMDAY, VEVAL, UPDORB, UPDATE, TYPORB, TPOSTA, TIDEP, TIDAL, TDIF, SURDEN, SUNGRV, SUMMRY, STORE, START, STAINP, SMSTAT, STAINF*, SQUANT, SIMRD, RMSCMP, RESPAR, REFCOR, RANDOM*, PROCES, PRNTPR, PREDCT, PRECES, POSVEL, PLHOUT, PDENT, PDEN, PCERD, ORBIT, ORB1, OBSDOT, NUTATE, NEWARC, JANTHG, INOUPT, INDENT, GGRRAN, GEOSRD, GEOIDH, F, ESTIM, ERROR, EPHEM, ELEM, EGRAV, DRAG, DPFCT*, DODSRD, DODELM, DIFF, DENSTY, DELTAZ, DATES, DATBSE, DATARD, COVELL, CONNEL, COMPAR, COMADJ, COM, COEFL, COEF, CBROWN, BSCOMP, BMTVRT, BIAS, AVGPOT, AREAS, ARCPAR, APPER, ALIST, ADFLUX, ADDYMD*, NONAME, MAIN

COMMON BLOCKS

ALPMRC, APARAM, CELEM, CEPHEM, CGEOS, CONOUT, CONSTS, CORB1, CPARAM, CSLIM, CSTAT, CSTHET, CSTINF, CTIME, CUVECT, DODDAT, DRGBLK, EERMSG, FLXBLK, FMODEL, GEODYN, GRDTHR, INTEX, INTBLK, INTERP, MONTHS, MOONGR, PREBLK, PRIORI, SIGBLK, STANUM, STAPOS, SREBLK, TPEBLK, TRUPOL, VMAT, VRBLOK, XYZ, XYZOUT

7.0-2

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

8.0 PROGRAM DESCRIPTIONS

SECTION 8.0

PROGRAM DESCRIPTIONS

The functions of GEODYN and all of its subroutines are described within this section.

The listings of the GEODYN program and each of its subroutines have been thoroughly described with internal comment cards. All COMMON storage variables used by each subprogram are described in Section 9.0.

Flowcharts

This document uses the flowcharting systems developed for the NONAME documentation. This system utilizes only six basic flowcharting symbols. The symbols and their uses are as follows:

1)

a)

b)

c)

a) Entry to subroutine without multiple entry points.

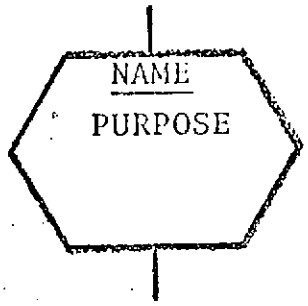
b) Entry to subroutine with multiple entry points.

c) Return.

2)

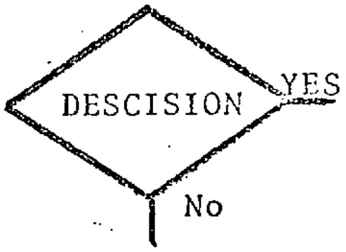
Description of operation being performed

3)



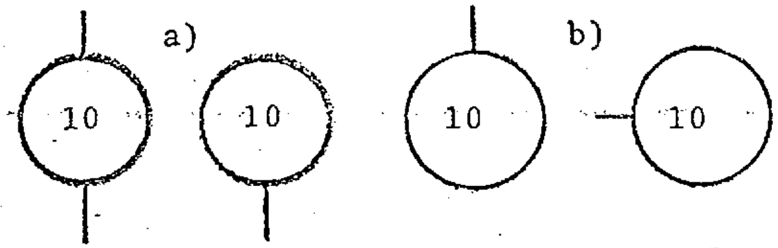
Subroutine or function call where:
NAME - Subroutine called
PURPOSE - Description of the purpose for the call.

4)



Decision

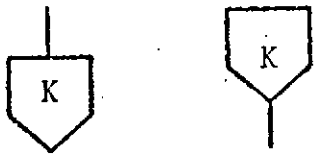
5)



a) Statement number.

b) Transfer to statement number.

6)



Off page connector.

In this new flowcharting system logic lines flow down, right, or left but never up. The only way for logic to travel opposite to the flow is to transfer to a statement number which appears earlier in the flow of logic.

At times, statement numbers are not available for back transfer. In these cased, an alphabetic name or character may be inserted in the line of logic at the point to which return of logic flow is desired. The name or character inserted must appear within symbol 5.

This flowcharting system has been designed to correlate as highly as possible with the program listings. The statement numbers used are the same as those used by the FORTRAN program and the description boxes contain the same information as the comment cards appearing in the program.

Programming Technique

This section is included as a further aid to programmers working with the GEODYN program and describes in general terms some of the techniques used in the programming of GEODYN.

Array and scalar names in GEODYN have been chosen in such a manner as to be meaningful. The following are some examples of naming in GEODYN.

NAME	SUBROUTINE	MEANING
PRDMIN	NONAME	Satellite period in minutes.
NARCS	NONAME	Number of arcs in run.
LOUTER	NONAME	Last outer iteration switch.
RATIO1	NONAME	Ratio to sigma for first measurement.

NAME	SUBROUTINE	MEANING
ORBELP	NONAME	Keplerian orbital parameters from previous iteration.
THETG	PROCES	Greenwich Hour Angle in radians.
CLATG	SQUANT	Cosine of station geodetic latitude.
NODEGF	STAINF	Number of degrees of freedom.
TWOPI	TDIF	Two times π.
MJSTOP	DATBSE	Modified Julian date of stop time for data selection.
VAR	COWELL	Switch for variable step integration.

The GEODYN program has been broken into many subroutines in order to optimize its use of core storage and also its efficiency. Many of the smaller subroutines are conversion routines which are called by many different segments of the program. Large portions of the GEODYN program are input/output subroutines which have been designed to give the user maximum ease in setting up the program.

GEODYN also uses a very large amount of COMMON storage. COMMON is used for five primary reasons by GEODYN.

1) To pass information between subroutines.

2) To store program constants.

3) To set switches for subprogram reinitialization.

4) To overlay storage requirements.

5) To load block data.

GEODYN uses two additional core saving techniques:

- Linkage Overlay
- Variable Core Allocation

Linkage Overlay

GEODYN uses a complex overlay structure to reduce the effective program size by nearly 300K bytes. A diagram of the GEODYN overlay structure is provided in Section 3.0.

In reference to the Diagram of the Overlay Structure, the overlay segments are utilized as follows:

SEGMENT	FREQUENCY OF USE
ROOT	Always in core.
1A	Once per job.
3A	Once per job.
1D, 2D, 3D, 4D	Maximum of one of these per arc.
1E, 2E	Maximum of one of these per arc.
2A	Once per job.
1B	Once per job.
2B	Once per outer iteration.
3B	Once per outer iteration.

SEGMENT	FREQUENCY OF USE
1C	Once per inner iteration.
2C	Once per inner iteration.
3C	Once per inner iteration.
4C	Once per outer iteration per arc.
5C	Once per outer iteration.
1F	Once or twice per arc.
2F	Once per arc maximum.

Variable Core Allocation

GEODYN counts the input parameters and allocates the minimum array sizes required to process each job.

As a result of this:

- Orbit generator arcs may require as little as 330K, and
- Data reduction arcs as little as 350K or as much as the entire capabilities of the GSFC IBM 360/95, depending on the user input requests.

On the following pages of this section appear the descriptions, program listings and flow charts for the GEODYN program.

The programmer should note that many of the GEODYN subroutines have more than one entry point. In general the purpose of these multiple entry points is to pass to these subroutines the starting locations of those arrays which are variable storage and to also set the dimension of these arrays.

MAIN

DESCRIPTION

- Calls program initialization subroutines.
- Calls FLUXM to list input cards, read flux cards and set up flux arrays.
- Reads run title, reference time, epoch and element cards.
- Calls INOUPT to read and print a description of the common and arc parameters.
- Calls STORE to store common and arc parameter information.
- Calls DODELM to obtain elements from DODS data base when necessary.
- Calls epoch element conversion routines when necessary.
- Calls APPER to calculate apogee and perigee heights.
- Calls CHARLY to allocate core and begin processing.
- Calls ERROR to print error messages when necessary.

NAME MAIN

PURPOSE

1) TO READ AND TO CALL SUBROUTINES TO READ GEODYN INPUT CARDS

2) TO DETERMINE ARRAY SIZES FOR VARIABLE STORAGE ALLOCATION

3) DRIVER FOR ALL SEGMENTS OF GEODYN

SUBROUTINES USED	TDIF	NEWARC	POSVEL	FLUXM	YMDAY
	ERROR	DODELM	APPER	INOUPT	DOTPRD
	CHARLY	STORE	ELEM		
COMMON BLOCKS	INITBK	INTBLK	TPEBLK	CTIME	PREBLK
	ALPMRC	APARAM	CPARAM	CSTINF	CELEM
	PRIOR1	CONOUT			

INPUT FILES FLTP - FLUX DATA FILE
INTP - GEODYN INPUT CARDS

OUTPUT FILES PRINTER

```
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ICORE(30)
      DATA NCORE/21/
      REAL APLM,CONVRG,OUTCON,RMSALL,AITIME,EDITN,RFINDX,SIGCHG,
     .   VARCOV,RMSTOT,TDIF
      LOGICAL*1 VHFCHN,PREPRO
      LOGICAL CMPGPR,DELEM,LITRES,INITAL,TOREFT,ORBTSW,MISLOG,VARSTP,
     .   HLVDSW,HYPER
      INTEGER ADDR,SRAD,ARCNO,ESTSTA,OUTP,DATP,XYZTP,RVTP,PLOTP,SCRA,
     .   SCRC,FLTP,GRDTP,ESTSTA,RECNO,ADDR,ORDER
      INTEGER*2 IPREPR,INDPRE,IMTYPE,ISTNO,CULL,MTYPE,NMEAS,PRETYP,
     .   CHANEL,JBASE
      DOUBLE PRECISION LOVE
      DIMENSION IHYPER(2)
      COMMON/INITBK/IEPYMD,IEPHM,EPSEC,IYREF,INNMAX,INNMIN,CONVRG,
     .   ORBEL(6,2),EDITN,INSLPR,IDSAT(2),ORBTSW,IXYZSW(11),MISLOG(9)
      COMMON/INTBLK/IHDOT(50),ORDER(2,2),ASAT(4),VARSTP(2),HLVDSW(2),
     .   NEQN(2),ADDR(2,3),LOVE(4)
      COMMON/TPEBLK/INTP,OUTP,DATP,XYZTP,KEPTAP,RVTP,PLOTP,IOBS,SCRA,
     .   SCRC,FLTP,GRDTP
      COMMON/CTIME/DATAEP,DAYREF,DSTART,DAYSTP,DAYINT,DORBIT,DAYEND,
     .   DRATE,DORB1,DORB1E,DRBRT,IYEEG
      COMMON/PREBLK/DAYSTA,OBSO1,OBSO2,SIG1,SIG2,SRFNDX,ISTA,MTYPE,
     .   NMEAS,PRETYP(2),CHANEL,VHFCHN,PREPRO,RECNO
      COMMON/ALPMRC/ALPHA(54),HYPER
      COMMON/APARAM/INPAR,INPAR1,NBIAS,ESTSTA,NSAT,NGPARC,NOREC1,
     .   NPARAM,NBIASM,MPAR
      COMMON/CPARAM/NSTA,NMAST,NSTEST,MAXPAR,MBIAS,NGPC1,NGPC2,
     .   NGPCOM,NCSEST,CMPGPR,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,
     .   INNRSW,NCONST,NOCONS
      COMMON/CSTINF/JBASE(556),LBASE
```

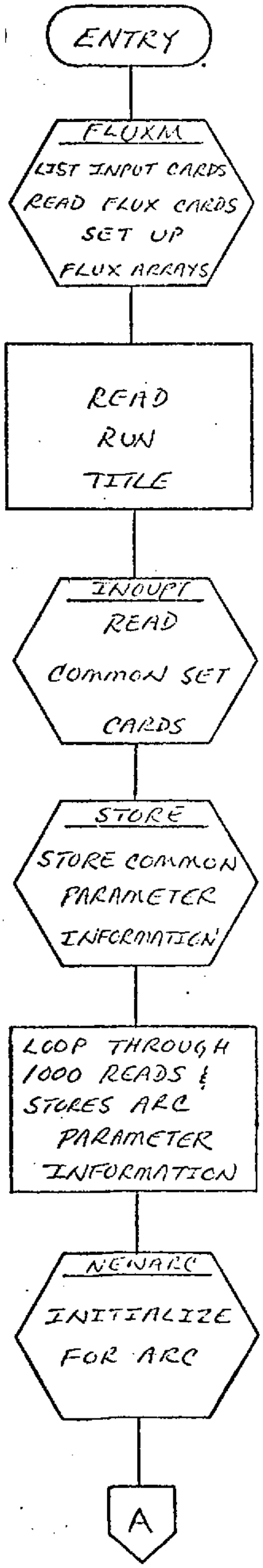
```
      COMMON/CELEM/ELEMST(6,2),ORBELA(6,2),XNU,EC,RMSTOT
      COMMON/PRIORI/ELEMIN(6,2),VARCOV(6,6,2),TITLE(30),DRAG(18)
      COMMON/CONOUT/RMSALL,OUTCON,MINOUT,MAXOUT,LITRES,MAXSAT,MAX2IN,
     .  NSTART,NEQNMX,IVAR,IORDER,NARCS,NSTARD,LSTART(6)
      DATA INTCOR/0/,NEBIAS/0/
      AITIME(DAYREF)=TDIF(4,3,DAYREF)/8.64E4
C READ INPUT CARDS, READ FLUX CARDS & SET UP FLUX ARRAYS
      CALL FLUXM(NARCS)
C READ RUN TITLE
      READ(INTF,10000) TITLE
      NSAT=2
      LBASE=0
C READ COMMON SET CARDS
      CALL INOUPT(NARCS,0)
C STORE COMMON PARAMETER INFORMATION
      CALL STORE(.FALSE.,.TRUE.)
      NGPCOM=NCSEST
      NTIDEN=NTIDST+NDENST-NCONST
      OUSUPR=INSUPR
      RMSALL=RMSTOT
      NGPC2=0
      MAXSAT=1
      MAXPAR=0
C LOOP THROUGH 1000 READS & STORES ARC PARAMETER INFORMATION
      DO 1000 ARCNO=1,NARCS
C INITIALIZE FOR ARC
      CALL NEWARC
      READ(INTP,10000) TITLE
      READ(INTF,10001) IYREF
      IYBEG=IYREF/10000
      DAYREF=YMDAY(IYREF,0,0.D0)
      DAYREF=DAYREF+AITIME(DAYREF)
C READ EPOCH CARD
      READ(INTF,10001) IEPYMD,IEPHM,EPSEC,INNMAX,INNMIN,CONVRG,IGEOS,
     .    IYMD,IHM,SEC,IYMDD,IHMD,SECD,NSAT,IELM,J,ISATID
      DELEM=J.GT.0
      IF(DELEM) NSAT=1
      NSAT=MAX0(1,MIN0(NSAT,2))
      MAXSAT=MAX0(MAXSAT,NSAT)
      MTYPE=IELM
      IF(IYMDD.GT.0) GO TO 100
      IYMDD=IEPYMD
      IHMD=IEPHM
      SECD=EPSEC
  100 DATAEP=YMDAY(IYMDD,IHMD,SECD)
      DATAEP=DATAEP+AITIME(DATAEP)
      INNMIN=MAX0(1,INNMIN)
      INNMAX=MAX0(INNMAX,1)
      IF(CONVRG.LE.0.0) CONVRG=0.02
      DAYSTP=9999.0D0
      IF(IYMD.EQ.0) GO TO 120
      DAYSTP=YMDAY(IYMD,IHM,SEC)
      DAYSTP=DAYSTP+AITIME(DAYSTP)
  120 DSTART=YMDAY(IEPYMD,IEPHM,EPSEC)
      DSTART=DSTART+AITIME(DSTART)
C OBTAIN ELEMENTS FROM DODS DATA BASE IF REQUESTED
```

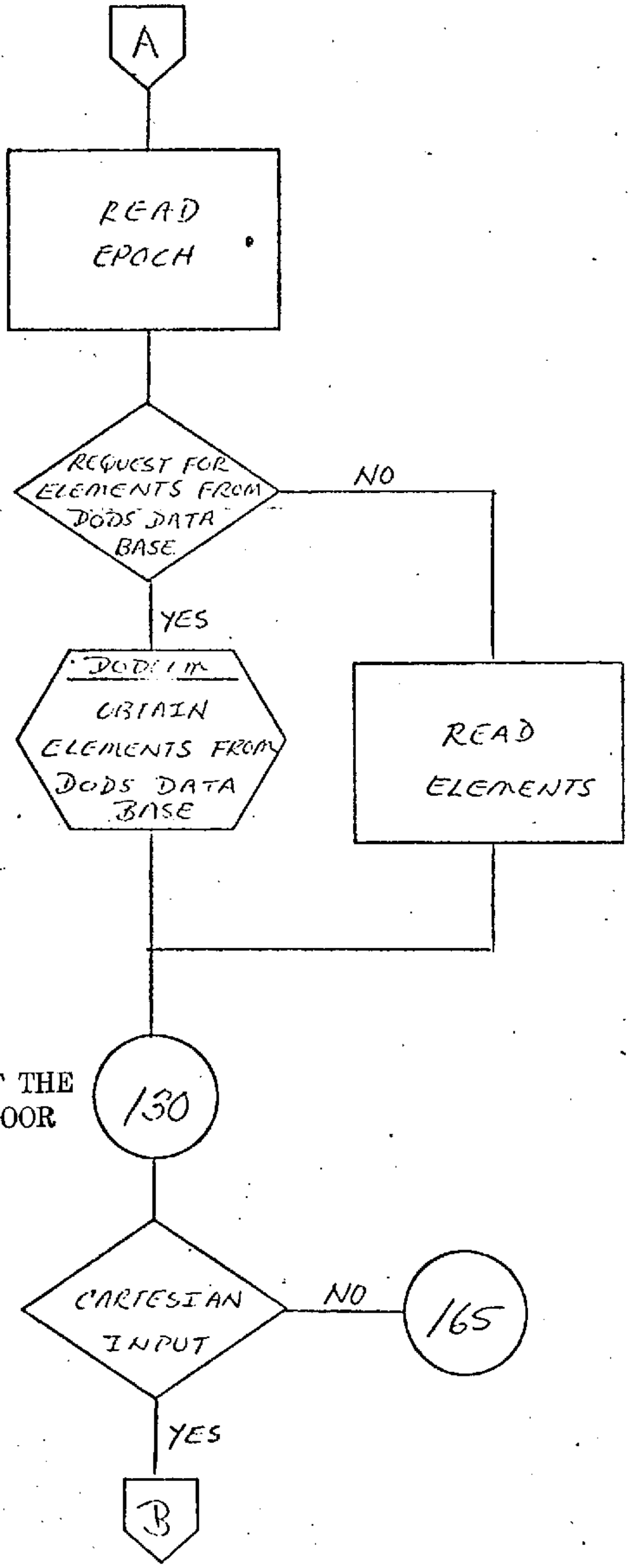
```
      IF(DELEM) CALL DODELM(IELM,ISATID,DSTART)
      HYPER=.FALSE.
      IHYPER(1)=0
C READ ELEMENTS
      IF(.NOT.DELEM) READ(INTP,10002) ((ELEMST(J,I),J=1,3),IHYPER(I),
     *   (ELEMST(J,I),J=4,6),I=1,NSAT)
      DO 200 ISAT=1,NSAT
      HYPER=HYPER.OR.IHYPER(ISAT).GT.0
      IF(IHYPER(ISAT)-1) 130,140,165
  130 IF(DOTPRD(ELEMST(4,ISAT),ELEMST(4,ISAT)).LT.DFLOAT(3*350**2).AND.
     *   DABS(ELEMST(2,ISAT)).LT.1.0D0)  GO TO 165
C CALCULATE KEPLER ELEMENTS IF INPUT IS CARTESIAN
  140 CALL ELEM(ELEMST(1,ISAT),ORBELA(1,ISAT),1,.TRUE.,TITLE)
      GO TO 190
165   DO 175 I=1,6
175   ORBELA(I,ISAT)=ELEMST(I,ISAT)
C CALCULATE CARTESIAN ELEMENTS IF INPUT IS KEPLER
      CALL POSVEL(ELEMST(1,ISAT),ORBELA(1,ISAT),1)
  190 DO 200 I=1,6
      ELEMIN(I,ISAT)=ELEMST(I,ISAT)
200   ORBEL(I,ISAT)=ORBELA(I,ISAT)
      IF(.NOT.HYPER) GO TO 220
      PRINT 10003,ARCNO
      CALL ERROR(10,LOVE)
C CALCULATE APOGEE & PERIGEE HEIGHTS
  220 CALL APPLE
      INSUPR=OLSUPR
C READ OPTION CARDS (AND DATA TAPE FOR D.C. RUNS)
      CALL INPUT(ICDOS,ARCNO)
      NEBIAS=MAX0(NEBIAS,NBIASE)
      NGPC1=900-NGPARC
      NGPC2=MAX0(NGPC2,NGPC1)
C STORE INFORMATION FOR ARC PARAMETERS
      CALL STORE(.FALSE.,.FALSE.)
      MBIAS=MAX0(MBIAS,NBIAS)
      NEQNMX=MAX0(NEQNMX,INPARI+NGPC1)
      KORE=0
      DO 400 I=1,NSAT
      K=0
      IF(.NOT.ORBTSW) K=6+NEQN(I)+NGPC1+NGPCOM+NTIDEN
      IVAR=1
      IF(VARSTP(I)) IVAR=2
  400 KORE=KORE+5*IVAR*((ORDER(I,1)-1)+K*(ORDER(I,2)-1))-(IVAR-1)*(1+K)
      INTCOR=MAX0(INTCOR,KORE)
 1000 MAXPAR=MAX0(MAXPAR,NBIAS+INPARI+NGPC1)
      MAXPAR=MAXPAR+NCSEST+NMAST*3+NTIDEN
      NEQNMX=NEQNMX+NTIDEN+NCSEST+1
      MBIAS=MBIAS+MOD(MBIAS,2)
      NGPC2=NGPC2+MOD(NGPC2,2)
      NCSEST=NCSEST+NGPC2
      NGPC1=NGPC2+1+MBIAS+3*MAXSAT
      NGPC2=NCSEST+MBIAS+3*MAXSAT
C CALCULATE SIZE OF VARIABLE STORAGE ARRAYS
C
C USTRT,USEND,BIASO,BIASSG,MSTAND,BTYPE,BIASN)
      ICORE(1)=MBIAS*7+NCSEST*4+13*MAXSAT+(MBIAS+2*MAXSAT+NCSEST+1)/2
```

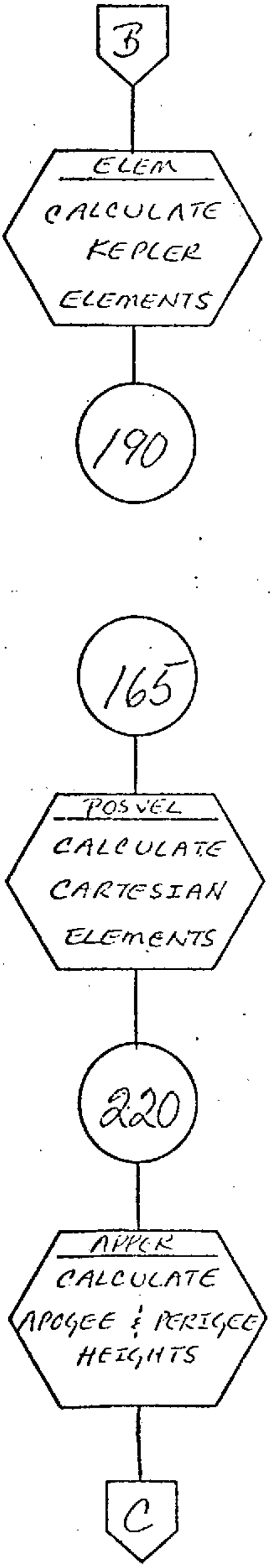
```
C BBIAS
      ICORE(2)=2*(MBIAS+3*MAXSAT+NCSEST+NDEN+NTIDST)
C ASUM
      ICORE(3)=6*4*(NSTA+MAXSAT)*MAXSAT
C NSUM
      ICORE(4)=3*3*(NSTA+MAXSAT)*MAXSAT
C RLAT,RLON,H,NAME,RLATO,RLONO,HO,THPRIM
      ICORE(5)=2*(8*NSTA+4+3*NSTEST)+2*(NSTA+1)*2
C ISTANO,ESTANO,LOC
      ICORE(6)=NSTEST+(NSTA+2)/2
C ENV
      ICORE(7)=2*3*(NSTA+1)*3
C SUM1
      MPAR2=MAXPAR*(MAXPAR+1)
      ICORE(8)=MAX0(MPAR2,72)
C GRPAR
      ICORE(9)=3*(NCSEST+3*MAXSAT+NTIDEN)*2
C INDXCS,STANOS
      ICORE(10)=(3*(NCSEST+NSTEST)+1)/2
C SUM2(DDELTA),ITL,DELTA
      ICORE(11)=2*MAXPAR*3
C XYZSIG,PLHSIG,STASIG,PLHNOM,XYZNOM
      ICORE(12)=9*NSTEST*3+6*NSTEST*2
C SUMS,FCT
      ICORE(13)=INTCOF+12*NEONMX*MAXSAT
C STAXYZ,STAXYZC
      ICORE(14)=2*(NSTA+1+NSTEST)*3
C PMPXO
      ICORE(15)=2*(NEONMX+5)*2
C XI
      ICORE(16)=2*NEONMX*6*MAXSAT
C PARNOS
      ICORE(17)=(NEONMX+9)/2
C AREA,CENTER,ECENTR,DENSO,DSIG
      ICORE(18)=8*4*NDEN*2+2*NDENST*2+2*(NTIDEN+NCONST)*2
C COEFFICIENTS OF CONSTRAINT EQNS FOR DENSITIES
      ICORE(19)=2*NDEN  *NCONST
C CSUM,MSUM,LSUM
      KBASE=0
      IF(LBASE.EQ.0) GO TO 444
      DO 333 I=1,NSTA
      J=0
      DO 222 L=1,LBASE
      IF(JBASE(L).EQ.I) J=J+1
  222 CONTINUE
  333 KBASE=MAX0(KBASE,J)
  444 LBASE=KBASE
      ICORE(20)=(NSTA+MAXSAT)*MAXSAT*LBASE*(16+6+1)
C BTWA,BTWD,BTWE,BTIME,BESTNO,BETYPE,NUMBER
      NEBIAS=NEBIAS+MOD(NEBIAS,2)
      NBIASE=NEBIAS
      ICORE(21)=2*(4*MAXPAR+13)*NEBIAS
      DO 300 I=1,NCORE
  300 ICORE(I)=ICORE(I)*4
      REWIND INTP
C ALLOCATE CORE & BEGIN PROCESSING
```

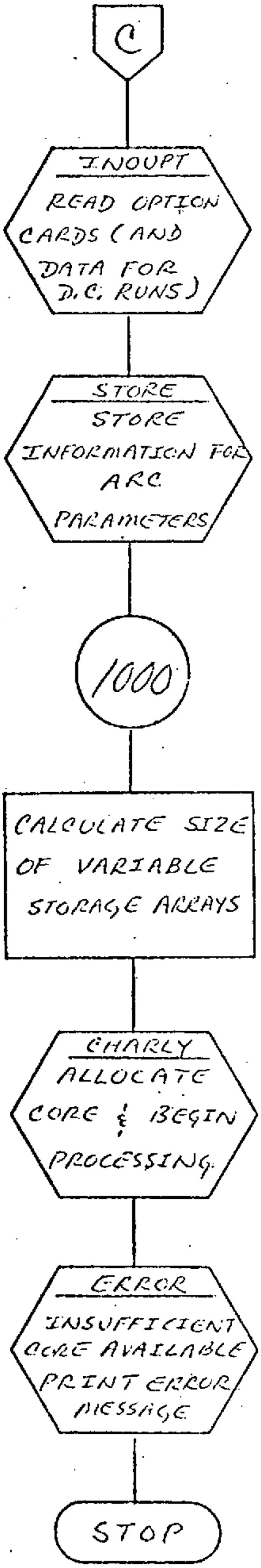
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      CALL CHARLY(NCORE,ICORE,&1050)
      STOP 1
 1050 KORE=0
      DO 1060 I=1,NCORE
 1060 KORE=KORE+ICORE(I)
      KORE=KORE/1024+320
      PRINT 1070,KORE
C INSUFFICIENT CORE AVAILABLE PRINT ERROR MESSAGE
      CALL ERROR(10,DATAER)
 1070 FORMAT(1H1,20X,'EXECUTION TERMINATING DUE TO INSUFFICIENT MAIN ',
     .   'CORE STORAGE.'/1H0,15X,'THE USER SHOULD SPECIFY REGION=',
     .   I5,'K PLUS ADDITIONAL STORAGE'/21X,'FOR ALL EXCESSIVELY ',
     .   'LARGE I/O BUFFERS USED.'/)
      STOP 10
10000 FORMAT(10A8)
10001 FORMAT(I6,I4,F7.4,I2,I1,2PF2.0,I1,2(I6,I4,0PF7.4),I1,I6,I1,I7)
10002 FORMAT(3D24.16,I1/3D24.16)
10003 FORMAT(1H1,20X,'WARNING:'/1H0,15X,'INPUT ON THE POSITION VECTOR ',
     .   'CARD IN ARC',I3,/21X,'INDICATES THAT KEPLER ELEMENTS ',
     .   'MAY HAVE AN ECCENTRICITY GREATER THAN 1.'//)
      END
```


ENTRY
FLUXM
LIST INPUT CARDS
READ FLUX CARDS
SET UP
FLUX ARRAYS
READ
RUN
TITLE
INOUPT
READ
COMMON SET
CARDS
STORE
STORE COMMON
PARAMETER
INFORMATION
LOOP THROUGH
1000 READS &
STORES ARC
PARAMETER
INFORMATION
NEWARC
INITIALIZE
FOR ARC
A


A
READ
EPOCH
REQUEST FOR
ELEMENTS FROM
DODS DATA
BASE
NO
YES
DODELM
OBTAIN
ELEMENTS FROM
DODS DATA
BASE
READ
ELEMENTS
130
CARTESIAN
INPUT
NO
165
YES
B


B
ELEM
CALCULATE KEPLER ELEMENTS
190
165
POSVEL
CALCULATE CARTESIAN ELEMENTS
220
APPER
CALCULATE APOGEE & PERIGEE HEIGHTS
C


C
INOUPT
READ OPTION CARDS (AND DATA FOR D.C. RUNS)
STORE
STORE INFORMATION FOR ARC PARAMETERS
1000
CALCULATE SIZE OF VARIABLE STORAGE ARRAYS
CHARLY
ALLOCATE CORE & BEGIN PROCESSING
ERROR
INSUFFICIENT CORE AVAILABLE PRINT ERROR MESSAGE
STOP

NONAME

DESCRIPTION

- Functions as a driver for orbit determination and data reduction operations.
- Reads input tape for restart when necessary.
- Calls GEOIDH to compute surface density geoid heights.
- Calls ESTIM to initialize the least squares estimator.
- Calls TWOSTA to compute partials and calculated observations for VLBI and average range rate data when necessary.
- Calls PREDCT to compute the orbit for orbit generation runs or calculated observations for measurements other than VLBI and average range rate data.
- Sets parameter numbers for adjusted stations, biases and timing biases.
- Writes a binary residual tape, groundtrack tape, and R-V tape and an optional ORB1 tape.
- Calls SUMMRY to print arc residual summary.
- Calculates and prints adjusted elements.
- Calls TYPORB to print arc summary page.

- Punches adjusted elements of drag and solar radiation pressure.
- Calls DATARD to store updated arc parameters.
- Prints adjusted parameter variance-covariance matrix.
- Calls ESTIM to calculate adjusted common parameter values.
- Prints adjusted arc force model parameters.
- Corrects and prints adjusted biases.
- Prints residual summary for all arcs.
- Calls ESTIM to estimate geopotential parameters.
- Computes estimated station correlations.
- Prints estimated station summary.
- Prints geopotential coefficient adjustment information.
- Calls UPDATE to compute effects of adjusted geopotential parameters on adjusted individual arc parameters.
- Calls subroutine to print correlations on adjusted parameters.

NAME NONAME

PURPOSE
1) TO DRIVE ORBIT GENERATOR AND DATA REDUCTION OPERATIONS
2) TO OUTPUT RESIDUALS, EPHEMERIS & ADJUSTMENTS

CALLING SEQUENCE

```
CALL NONAME(NPARM,PMPX0,BSNOS,BIAS0,BBIAS,BIASSG,
            BSTRT,BSEND,BSTANO,BYTPE,PARNOS,
            DDELTA,TTL,DELTA,STANOS,ISTANC,NAME,
            SUM1,INDXCS,GPVAL,GPVAL0,GPSIG,GPNO,
            AREA,CENTER,DENCON,BESTNO,BETYPE)
```

SYMBOL	TYPE	DESCRIPTION
NPARM	I	INPUT - MAXIMUM NUMBER OF PARAMETERS PER MEASUREMENTS
PMPX0 (NPARM,1)	DP	INPUT - MEASUREMENT PARTIALS WITH RESPECT TO EPOCH PARAMETERS
BSNOS (1)	I*2	INPUT - LOCATION IN NORMAL MATRIX FOR BIASES
BIAS0 (1)	DP	INPUT & OUTPUT - APRIORI BIAS ESTIMATES
BBIAS (1)	DP	INPUT - CURRENT BEST BIAS VALUES
BIASSG (1)	DP	INPUT & OUTPUT - STANDARD DEVIATIONS OF BIASES
BSTRT (1)	DP	INPUT & OUTPUT - BIAS START TIMES IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
BSEND (1)	DP	INPUT & OUTPUT - BIAS STOP TIMES IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
BSTANO (1)	I*2	INPUT & OUTPUT - BIAS STATION NUMBERS
BYTPE (1)	I*2	INPUT & OUTPUT - BIAS TYPES
PARNOS (1)	I*2	OUTPUT - PARAMETER NUMBERS
ODELTA (1)	DP	INPUT - SCRATCH VECTOR
TTL (1)	DP	INPUT - ADJUSTED PARAMETER TITLE ARRAY
DELTA (1)	DP	INPUT & OUTPUT - CORRECTION VECTOR FOR ADJUSTED PARAMETERS

STANOS (3,1)	I*2	INPUT - LOCATIONS IN NORMAL MATRIX OF INFORMATION PERTAINING TO ADJUSTED STATION COORDINATES
ISTANO (1)	I*2	INPUT - STATION NUMBERS
NAME (1)	DP	INPUT - STATION NAMES
SUM1 (1)	DP	INPUT - NORMAL MATRIX
INDXCS (3,1)	I*2	INPUT - INDICES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
GPVAL (1)	DP	INPUT - CURRENT BEST VALUES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
GPVALO (1)	DP	INPUT - APRIORI ESTIMATE FOR ADJUSTED GEOPOTENTIAL COEFFICIENTS
GPSIG (1)	DP	INPUT - SIGMAS FOR ADJUSTED GEOPOTENTIAL PARAMETER
GPNO (1)	I*2	INPUT - LOCATIONS OF GEOPOTENTIAL PARAMETERS IN NORMAL MATRIX
AREA (1)	DP	INPUT & OUTPUT - SURFACE DENSITY SUB-BLOCK AREAS
CENTER (1)	DP	OUTPUT - THE GEOCENTRIC COORDINATES OF THE SUB-BLOCK CENTERS
DENCON (1)	DP	INPUT & OUTPUT - COEFFICIENTS RELATING CONSTRAINED AND UNCONSTRAINED SURFACE DENSITIES
BESTNO (1)	I*2	INPUT - ELECTRONIC BIAS STATION NUMBERS
BETYPE (1)	I*2	INPUT - ELECTRONIC BIAS MEASUREMENT TYPES

SUBROUTINES USED	DATARD	CLEAR2	PREDCT	OUTRAD	ORB1
	ESTIM	RANDWR	ERROR	COMADJ	STAINF
	RANDRD	OBSDOT	SUMMRY	UPDATE	GEOIDH
	DATES	BSCCMP	TYPOR3	CLEAR	TWOSTA
	ELEM	CORREL			
COMMON BLOCKS	APARAM	CONOUT	CPARAM	ALPMRC	INTBLK
	INITBK	TPEBLK	CONSTS	CTIME	XYZ
	XYZOUT	PREBLK	FLXBLK	PRIORI	CSTINF
	FMODEL	CGEOS	CELEM	GNDTRK	CORB1

INPUT FILES INSTRT - INPUT RESTART TAPE NUMBER

```
                        FLTP - FLUX DATA FILE

   OUTPUT FILES         OUTP - PRINTER
                        PLOTP - BINARY RESIDUAL TAPE
                        XYZTP - PRINTER
                        RVTP - BINARY SATELLITE EPHEMERIS TAPE
                        GRDTP - GROUNDTRACK TAPE
```

```
      SUBROUTINE NONAME(NPARM,PMPX0,BSNOS,BIAS0,BBIAS,BIASSG,BSTRT,
     •   BSEND,BSTAND,BYTPE,PARNOS,DDELTA,TTL,DELTA,STANOS,
     •   ISTAND,NAME,SUM1,INDXCS,GPVAL,GPVAL0,GPSIG,GPNO,AREA,CENTER,
     •   DENCON,BESTNO,BETYPE)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL*1 VHFCHN,PREPRO
      LOGICAL TOREFT,VARSTP,ORBTSW,XYZFSW,XYZLSW,TOREF0,PLTLSW,NOPRNT,
     •   PLOTSW,INITAL,DATASW,SATSW,FITER,LITER,DRAGSW,LITRES,LINNER,
     •   LOUTER,NCRRAT,CMPGPR,GRDSW,GRDFSW,SUBSAT,KEPLER,HLVDSW,PARTGP,
     •   BMAT,SIMDAT,PCESIM,PBMAT,TRKSW,STARTA,STARTR,INNRSW,SATSAT
      LOGICAL HYPER
      INTEGER*2 MTYPE,NMEAS,PRETYP,CHANEL,BSTAND,BYTPE,PARNOS,BSNOS,
     •   STANOS,ISTAND,INDXCS,GPNO,ISAT,BETYPE,BESTNO
      INTEGER OUTP,RVTP,DATP,XYZTP,ARCNO,ADDR,ADDRD,SRAD,BSNDX,BINDX1,
     •   BINDX2,TINDX,PLOTP,FLTP,OUTER,JJSUPR,ESTSTA,PARLIM,RECNO,
     •   SCRA,SCRC,GRDTP,RECNO1,BMATNU,ORDER,STARTA,STARTO,OUTSTR
      REAL A1,A2,FNO,AXYZ,BSUM,RMS0,EDITN,EAST1,EAST2,RMSNO,WTMEAN,
     •   ASTRSK,CONVRG,RDMEAN,RMSPRV,SGPRNT,VARCOV,TITLE,OUTCON,RMSALL,
     •   RMSTOT,WTRND,RMSLST,RMSWTO,TYPRMS,PI,TWOPI,RAD,RSEC,BRSID1,
     •   BRSID2,BRSEC,NOEDIT
      DOUBLE PRECISION MODEL,NAME,ITNMS,LHAT,MSAT,MBODY,LOVE
      DIMENSION BSNOS(1),BIAS0(1),BBIAS(1),BIASSG(1),BSTRT(1),
     •   BSEND(1),BSTAND(1),BYTPE(1),PARNOS(1),PMPX0(NPARM,1),
     •   DDELTA(1),TTL(1),DELTA(1),ITAPE(2),SDELEM(6),RMSPOS(2),
     •   IDEG(4),MIN(4),SECA(4),APRIOR(2),ADJUST(2),SD(3),LHAT(3),
     •   ORBELP(6,2),ORBDIF(6),STANOS(3,1),BSNDX(3),ISTAND(1),
     •   NAME(1),SUM1(1),AXYZ(6),STPSZ1(4),ICS(3),INDXCS(3,1),DRAG(3),
     •   GPNO(1),GPVAL(1),GPVAL0(1),GPSIG(1),BRSID1(3),BRSID2(3)
      DIMENSION PVRMS(2,2),AREA(1),CENTER(1),DENCON(1),BESTNO(1),
     •   BETYPE(1)
      COMMON/APARAM/INPAR,INPAR1,NBIAS,ESTSTA,NSAT,NGPARC,RECNO1,NPARAM,
     •   NEBIAS,MAXPAR
      COMMON/CLNOUT/RMSALL,OUTCON,MINOUT,MAXOUT,LITRES,MAXSAT,MAX2IN,
     •   NSTART,NEONMX,IVAS,IORDER,NARCS,NSTARD,STARTR,STARTW,STARTA,
     •   STARTC,INSTRT,OUTSTP
      COMMON/CHARAM/NSTA,NMAST,NSTEST,NDIM,NBIAS,NGPC1,NGPC2,NGPCOM,
     •   NCBEST,CMPGPS,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,INNRSW,
     •   NCONST,NDCONS
      COMMON/ALFMRC/ITNMS(5),TIMING,BLANK,ATYPE(31),UNITS(15),ELCUT,
     •   HYPER
      COMMON/INTELK/THDOT1,THDOT2,THDT2S,GM,AE,AESQ,FLAT,FSQ32,FFSQ32,
     •   GM3(3),H(2),BDOT(2),FO(2),AFGM(2),APLM(2),RPRESS,INITAL,NORPAT,
     •   THETG0,MBODY(5),STEPSZ(4),HLVERR(2),RELERR(2),CTOL(2),RTOL(2),
     •   STPLOW(2),STPUP(2),ORDER(2,2),ASAT(2),MSAT(2),VARSTP(2),
     •   HLVDSW(3),NORM(2),APOR(2,3),LOVE(3),TOREFT,NBODY
```

```
      COMMON/INITBK/IEPYMD,IEPHM,EPSEC,IYREF,INNMAX,INNMIN,CONVRG,
     .   ORBEL(6,2),EDITN,INSUPR,IDSAT(2),ORBTSW,XYZFSW,XYZLSW,PLTLSW,
     .   GRDFSW,KEPLER,SUBSAT,PARTGP,PBMAT,BMATNO,SIMDAT,PCESIM,
     .   MISLOG(9)
      COMMON/TPEBLK/INTP,OUTP,DATP,XYZTP,KPLRTP,RVTP,PLOTP,IOBS,SCRA,
     .   SCRC,FLTP,GRDTP
      COMMON/CCNSTS/DPI,DTWOPI,DRAD,DRSEC
      COMMON/CTIME/DATAEP,DAYREF,DSTART,DAYSTP,DAYINT,DORBIT,DAYEND,
     .   DRATE,DORB1,DORBIE,ORBRT,IYBEG
      COMMON/XYZ/AEINFM(9),KSAT,IFORCE(2)
      COMMON/XYZOUT/XYZEND(6,2),ORGPAR(6,2)
      COMMON/PREBLK/DAYSTA,OBSO1,OBSO2,SIG1,SIG2,SRFNDX,ISTA,MTYPE,
     .   NMEAS,ISAT,PRETYP,CHANEL,VHFCHN,PREPRO,RECNO
      COMMON/FLXBLK/FLUX1(405),FLUX2(405),FLUX3(405),FLUX4(405),
     .   FLUX5(405)
      COMMON/PRIORI/ELEMIN(6,2),VARCOV(6,6,2),TITLE(60),DRAGSG(2,3),
     .   DRAGO(2,3),CD(2,3)
      COMMON/CSTINF/MEASNO(4),NOBS(4),RDMEAN(4),RMSD(4),RND(4),
     .   MEASWT(4),WTMEAN(4),RMSWTO(4),WTRND(4),TYPRMS(30),NOTYPE(2,30),
     .   BSUM(6,12),RMSNO(30),NCALL(30),NOWTDB,LBASE
      COMMON/FMODEL/INDEX1, NDEX2,INDEX3,INDEX4,CS(30,33),MODEL(6)
      COMMON/CGEOS/ISATID(2),IPREPR(453)
      COMMON/CELEM/ELEMST(6,2),ORBELA(6,2),IGIS(4),RMSTOT
      COMMON/GNDTRK/SATLAT(2),SATLON(2),SATH(2),ELEV(2),SATSW
      COMMON/CORB1/RANDOT(2),PERDOT(2),PERHT(2),APHT(2),PRD(2)
      REAL*8 PREV(2)/'PREVIO','US'/,CHANGE/'DELTA'/
      EQUIVALENCE (TINDX,BSNDX(1)),(BINDX1,BSNDX(2)),(BINDX2,BSNDX(3)),
     .   (SIG,SIG1),(KKSAT,SIG2),(PI,DPI),(TWOPI,DTWOPI),
     .   (RAD,DRAD),(RSEC,DRSEC),(PRD(1),TRKSW)
      DATA ASTRSK/1H*/,NOEDIT/1HE/
      DATA HERTZ/5HHERTZ/
      DATA APRIOR,ADJUST,SD/
     .      6HA PRIO,2HRI,5HADJUST,2HED,6HSTANDA,6HRD DEV,6HIATION/
      DATA AXYZ/2H X,2H Y,2H Z,4HXDOT,4HYDOT,4HZDOT/
      DATA PCE/4H PCE/
      DATA ARCNO,OUTER,NPAGE/3*1/
      DATA ICS/1HC,1HS/,DRAG/4HDRAG,8HDRAG OUT,5HSOLRAD/
      DATA BMAT/.FALSE./
      DATA NPAGEX/1/
      INDXNO(I)=NDIM*(I-1)-(I*(I-1))/2
10000 FORMAT(20A4)
      NCSMAX=NCSEST
      ARCNO=STARTA
      OUTER=STARTO
      IF(STARTR) MAXOUT=STARTO
      LOUTER=MAXOUT.LE.OUTER
      IF(.NOT.STARTR) GO TO 200
      IF(STARTA.LT.2) GO TO 200
C IF RESTARTING READ INPUT TAPE
      NI=STARTA-1
      DO 190 N=1,NI
      CALL DATARD(N,.TRUE.,.FALSE.,.FALSE.)
      NPARAM=INPARI+NBIAS
      IF(N.GT.1) READ(INSTRT)
      READ(INSTRT) LL
      L=NDIM*(NDIM+1)/2
```

```
      DO 125 I1=1,L,225
      I2=MIN0(I1+224,L)
  125 READ(INSTRT) (SUM1(I),I=I1,I2)
      DO 150 I1=1,NDIM,225
      I2=MIN0(I1+224,NDIM)
  150 READ(INSTRT) (DDELTA(I),I=I1,I2)
      DO 175 I1=1,LL,225
      I2=MIN0(I1+224,LL)
  175 WRITE(SCRC) (SUM1(I),I=I1,I2)
  190 CALL DATARD(N,.FALSE.,.TRUE.,.FALSE.)
      REWIND INSTRT
      CALL STAINF(1,0,P,P,P,P,P,P)
C START OF OUTER ITERATION LOOP
  200 RECNO=0
      IF(NGPCOM.LE.0) GO TO 250
C SET COMMON ADJUSTED GEOPOTENTIAL COEFFICIENTS TO CURRENT BEST VALUE
      NCS1=NCSMAX-NGPCOM+1
      DO 225 I=NCS1,NCSMAX
      J=INDXCS(1,I)
      N=INDXCS(2,I)
      M=INDXCS(3,I)
      C1=GPVAL(I)*GPSIG(I)
      IF(J.EQ.1) CS(N,M+1)=C1
  225 IF(J.EQ.2) CS(31-N,33-M)=C1
  250 CONTINUE
      IF(NDEN.LE.0) GO TO 300
C COMPUTE SURFACE DENSITY GEOID HEIGHTS
      CALL GEOIDH(AREA,CENTER,DENCON)
C READ INFORMATION FOR NEXT ARC
  300 CALL DATARD(ARCNO,.TRUE.,.FALSE.,.FALSE.)
      NSAT3=3*NSAT
      ITAPE(1)=OUTR
      ITAPE(2)=XYZTP
      IF(OUTER.NE.1) GO TO 320
      MAX2IN=MAX0(1,INNMAX/10)
      IF(INNMAX.NE.10) INNMAX=MAX0(1,MOD(INNMAX,10))
  320 CALL CLEAR(PSNOX,3,1)
      MINDEX=ESTSTA
C READ FLUX DATA FOR ARC
      READ(FLTP) THETGO,FLUX1
      READ(FLTP) FLUX2
      READ(FLTP) FLUX3
      READ(FLTP) FLUX4
      READ(FLTP) FLUX5
      IF(ORBTSW) GO TO 380
      CALL CLEAR2(PARNOS,NPARM,1)
      DO 375 I=1,INPARI
  375 PARNOS(I)=I
  380 INPAR=INPARI
      NPARAM=INPARI+NBIAS
      LINNER=INNMAX.EQ.1
      INNER=1
      DO 400 I=1,6
  400 STPSZ1(I)=STEPSZ(I)
      IF(NGPARC.LE.0) GO TO 420
C SET ARC ADJUSTED GEOPOTENTIAL COEFFICIENTS TO CURRENT BEST VALUES
```

```
      DO 410 I=1,NGPARC
      J=INDXCS(1,I)
      N=INDXCS(2,I)
      M=INDXCS(3,I)
      C1=GPVAL(I)*GPSIG(I)
      IF(J.EQ.1) CS(N,M+1)=C1
  410 IF(J.EQ.2) CS(31-N,33-M)=C1
  420 EARLIM=NSTART-1
C START OF INNER ITERATION LOOP
  650 IF(ORBTSW) GO TO 660
C SET SWITCHES FOR DATA REDUCTION
      LIM1=NSTART-1
      IF(INNER.EQ.1.AND.ARCNO.EQ.1) LIM1=NDIM
      ESTSTA=NMAST
      LIM2=0
      IF(LINNER) LIM2=1
      NOPRNT=INSUPR.EQ.4
      DAYORB=DSTART+DRATE
      IF(DATAEP.LT.DSTART) DAYORB=DATAEP
      BMAT=BMAT.OR.BMATNO.GT.0
      J=INDXNO(NDIM)+NDIM
      IF(BMATNO.GT.0) CALL CLEAR(SUM1,J,2)
C INITIALIZE LEAST SQUARES ESTIMATOR
      CALL ESTIM(1,P,P,P)
      GO TO 670
C SET START TIME FOR ORBIT GENERATOR
  660 DAYORB=DORBIT
      ISW= 0
  670 NOB=MINDEX-1
C INITIALIZE INTEGRATOR STEP SIZES FOR NEW INNER ITERATION
      DO 675 I=1,4
  675 STEPSZ(I)=STPSZ1(I)
      RECNO=RECNO1
C SET COUNTERS & LOGICAL SWITCHES FOR NEW INNER ITERATION
      PREPRO=.FALSE.
      MINDEX=1
      DAY1=DATAEP
      INNRSW=.FALSE.
      LINES=0
      LINESW=0
      FITER=INNER.EQ.1.AND.OUTER.EQ.1
      LITER=LINNER.AND.LOUTER
      NOPRNT=.NOT.(NOPRNT.OR.(LITER.AND.INSUPR.GT.1).OR.(FITER.AND.
     .        MOD(INSUPR,2).EQ.1))
      SATSW=(FITER.AND.XYZFSW).OR.(LITER.AND.XYZLSW)
      GROSW=FITER.AND.GROFSW
      SUBSAT= SATSW.OR.GROSW
      PLOTSW=PLTLSW.AND.LITER
      TRKSW=PLOTSW
      CMPGPR=NGPARC.GT.0
      NOBEST=NGPARC
      IF(.NOT.LINNER) GO TO 690
      INNRSW=NTIDEN.GT.0
      INPAR=INPAR+NGPCOM+NTIDEN
      IF(NGPCOM.EQ.0) GO TO 679
      CMPGPR=.TRUE.
```

```
      NCSEST=NCSMAX
  673 I1=NGPCOM+NTIDEN
      IF(I1.LE.0) GO TO 690
      DO 680 I=1,I1
  680 PARNOS(INPARI+I)=NSTART+I-1
C SET NUMBER OF FORCE MODEL EQUATIONS TO BE INTEGRATED
  690 NEQN(1)=1
      NEQN(2)=1
      IFORCE(1)=0
      IFORCE(2)=0
      IF(ORBTSW.OR.(LITER.AND.LITRES)) GO TO 692
      NEQN(1)=7+NGPARC+MAX0(ADDR(1,1),ADDR(1,2),ADDR(1,3))
      IFORCE(2)=NEQN(1)-7-NGPARC
      K=MAX0(ADDR(2,1),ADDR(2,2),ADDR(2,3))
      IF (K.GT.0) K=K-IFORCE(2)
      NEQN(2)=7+NGPARC+K
      IF(.NOT.LINNER) GO TO 693
      NEQN(1)=NEQN(1)+NGPCOM+NTIDEN
      NEQN(2)=NEQN(2)+NGPCOM+NTIDEN
      GO TO 693
  692 INPAR=0
  693 INITAL=.TRUE.
C SET EDIT CRITERION & START TIME
      EDIT=RMSTOT*EDITN
      RMSLST=RMSTOT
      DAY0=DSTART
      DATASW=.FALSE.
      ISTA=1
      IF(INNER.EQ.1.AND.ARCNO.EQ.1) ISTA=0
C INITIALIZE STATISTICS FOR INNER ITERATION
      IF(.NOT.ORBTSW) CALL STAINF(1,ISTA,P,P,P,P,P,P)
  700 IF(ORBTSW) GO TO 930
      IF(DATASW)GO TO 900
  725 RECNO=RECNO+1
C READ OBSERVATION
      CALL RANDFD
      IF(MTYPE.GT.0) GO TO 775
730   DAYSTA=99.D0
      GO TO 900
  775 DT=DAYSTA-DAY1
      IF(DT.GT.-.001D0)GO TO 900
      IF(.NOT.FITER) GO TO 725
C PRINT MESSAGE FOR OBSERVATION OUT OF TIME ORDER
      CALL DATES(DAYSTA,IYMD,IHM,SEC)
      WRITE(IOUTF,10202) ISTANO(ISTA),IYMD,IHM,SEC
      GO TO 725
C DETERMINE EARLIEST TIME FOR INTEGRATION END
 900  IF(DAYSTA-DAYORB) 975,950,925
  925 ISW=0
  930 DATASW=.TRUE.
      DAY1=DAYORB
      DAYORB=DAYORB+DRATE
      GO TO 1000
 950  DAYORB=DAYORB+DRATE
 975  DAY1=DAYSTA
      DATASW=.FALSE.
```

```
      ISN=ISTA
C DETERMINE IF INTEGRATION COMPLETE
1000  IF(DAYSTA.GT.999.D0.AND.DAY1.GT.DAYEND)GO TO 2000
      IF(MTYPE.LE.26) GO TO 1100
C COMPUTE PARTIALS & CALCULATED OBSERVATION FOR VLBI & AVG RANGE RATE
C DATA
      CALL TWOSTA(ISN,DAY1,RESID1,DATASW)
      GO TO 1150
C COMPUTE ORBIT FOR ORBIT GENERATOR RUNS OR CALCULATED OBSERVATION FOR
C MEASUREMENTS OTHER THAN VLBI & AVG RANGE RATE
 1100 CALL PREDCT(ISN,DAY1,RESID1,RESID2,DATASW)
C CALCULATE UTC DATE & TIME OF OBSERVATION OR EMPHEMERIS POINT
 1150 CALL DATES(DAY1,IYMD,IHM,SEC)
      IF(ORBTSW.OR.DATASW)GO TO 1600
      NEDIT1=0
      NEDIT2=0
      IF(MTYPE.GT.14.AND.MTYPE.LT.27) GO TO 1450
      IF(.NOT.LINNER) GO TO 1450
      IF(NSTEST.LE.0) GO TO 1450
C SET PARAMETER NUMBERS FOR ADJUSTED STATIONS
      DO 1410 I=1,3
      PARNOS(NPARM-3+I)=0
      PARNOS(NPARM-6+I)=0
      IF(ISTA.LE.NSTEST) PARNOS(NPARM-6+I)=STANOS(I,ISTA)
      IF(MTYPE.LT.27) GO TO 1410
      IF(CHANEL.LE.NSTEST) PARNOS(NPARM-3+I)=STANOS(I,CHANEL)
 1410 CONTINUE
 1450 IF (.NOT.FITER) GO TO 1460
      DAYSTA=DAY1
C REWRITE CORRECTED MEASUREMENTS ON C.A. FILE ON FIRST INNER ITERATION
C OF FIRST OUTER ONLY
      CALL RANDWR
      IF(.NOT.(SIMDAT.AND.FITER)) GO TO 1460
C IF SIMULATED DATA REQUESTED OUTPUT, WRITE CALCULATED OBSERVATION ON
C UNIT 17 ON FIRST INNER OF FIRST OUTER ONLY
      IF(MTYPE.GT.14) RESID2=0.0D0
      OBSIM1=OBS01-RESID1
      OBSIM2=OBS02
      ISTA2=CHANEL
      IF(MTYPE.GT.26) CHANEL=ISTANO(CHANEL)
      IF(MTYPE.LT.14) OBSIM2=OBS02-RESID2
      SATSAT=.FALSE.
      IF(MTYPE.NE.2.AND.MTYPE.NE.3) GO TO 1455
      SATSAT=KKSAT.GT.0.AND.KKSAT.LE.NSAT
      IF(.NOT.SATSAT) GO TO 1455
      J=KKSAT
      KKSAT=ISATID(KKSAT)
      OBSIM2=OBS02
 1455 K=ISTANO(ISTA)
      WRITE(17) IYMD,IHM,SEC,OBSIM1,OBSIM2,SIG1,SIG2,K,MTYPE,NMEAS,
     .   ISATID(ISAT),SRENDX,CHANEL
      CHANEL=ISTA2
      IF(SATSAT) KKSAT=J
C CHECK FOR ELEVATION CUTOFF
 1460 IF(ELEV(ISAT).GE.ELCUT) GO TO 1465
      SIG1=0.0D0
```

```
      IF(NMEAS.GT.1) SIG2=0.0D0
 1465 IF(MTYPE.LT.27) GO TO 1470
      IF(RANOUT(1).LT.ELCUT) SIG1=0.0D0
 1470 IF(NOPRNT) GO TO 1501
      IF(LINES.EQ.0) GO TO 1485
      IF(MOD(LINES,42).NE.0) GO TO 1500
      WRITE(OUTP,10101)
      NPAGE=NPAGE+1
 1485 WRITE(OUTP,10100) ARCNO,INNER,OUTER,NPAGE
 1500 IF(MOD(LINES,6).EQ.0) WRITE(OUTP,10102)
      LINES=LINES+1
 1501 IF(NEBIAS.LE.0) GO TO 7000
C SET INDICATORS FOR ELECTRONIC BIAS EXTRACTION
      DO 5000 I=1,NEBIAS
      IF(ISTA.NE.BESTNO(I)) GO TO 6000
      IF(MTYPE.EQ.BETYPE(I)) GO TO 5000
      IF(MTYPE.GT.7) GO TO 6000
      IF(MTYPE+7.NE.BETYPE(I)) GO TO 6000
      NEDIT2=I
      GO TO 6000
 5000 NEDIT1=I
 6000 CONTINUE
 7000 IF(NBIAS.EQ.0) GO TO 1526
C SET INDICATORS FOR BIASES & TIMING BIASES
      DO 1502 I=1,3
 1502 BSNDX(I)=0
      IF(MTYPE.GT.14.AND.MTYPE.LT.27) GO TO 1526
      DO 1520 I=1,NBIAS
      IF(ISTA.NE.BSTANO(I)) GO TO 1520
      I1=BYTPE(I)
      IF(I1.EQ.0) GO TO 1508
      IF(MTYPE.GT.20.AND.MTYPE.EQ.I1) GO TO 1503
      IF(MTYPE.NE.(I1-(I1/8)*7)) GO TO 1520
 1503 IF(DAYSTA.LT.BSTRT(I)) GO TO 1520
      IF(DAYSTA.GT.BSEND(I)) GO TO 1520
      I1=INPARI+I
      IF(BYTPE(I)-MTYPE)1512,1513,1514
 1512 TINDX=I1
      MEASTP=MTYPE
      IF(MTYPE.LE.14) TPAR1=DESDOT(MEASTP,ISTA,TPAR2)
      GO TO 1520
 1513 BINDX1=I1
      GO TO 1520
 1514 BINDX2=I1
 1520 CONTINUE
      IF(MTYPE.LE.14) GO TO 1525
      DO 1524 I=1,NBIAS
      IF(CHANEL.NE.BSTANO(I)) GO TO 1524
      IF(BYTPE(I).NE.0) GO TO 1524
      IF(DAYSTA.LT.BSTRT(I)) GO TO 1524
      IF(DAYSTA.GT.BSEND(I)) GO TO 1524
      J=I+NSAT2
      BINDX2=INPARI+I
      PARNOS(NPARM-6)=BINDX2
      RESID1=RESID1-PMPXO(NPARM-3,1)*BBIAS(J)
      TPAR1=PMPXO(NPARM-5,1)
```

```
      GO TO 1525
 1524 CONTINUE
C SET PARAMETER NUMBERS FOR BIASES & TIMING BIASES
 1525 PARNOS(NPARM-7)=BINDX2
      PARNOS(NPARM-6)=TINDX
      PMPX0(NPARM-6,1)=TPAR1
C CORRECT RESIDUALS FOR CURRENT VALUE OF BIASES & TIMING BIASES
      IF(BINDX1.GT.0) RESID1=RESID1-BBIAS(BINDX1-INPAR1+NSAT3)
      IF(TINDX.GT.0) RESID1=RESID1-TPAR1*BBIAS(TINDX-INPAR1+NSAT3)
      IF(NMEAS.EQ.1) GO TO 1526
      PMPX0(NPARM-6,2)=TPAR2
      IF(BINDX2.GT.0) RESID2=RESID2-BBIAS(BINDX2-INPAR1+NSAT3)
      IF(TINDX.GT.0) RESID2=RESID2-TPAR2*BBIAS(TINDX-INPAR1+NSAT3)
C COMPUTE RATIOS TO SIGMA & DETERMINE IF MEASUREMENT EDITED
 1526 RATIO1=0.D0
      EAST1=BLANK
      IF(SIG1.NE.0.D0) RATIO1=DABS(RESID1/SIG1)
      IF(NEDIT1.EQ.0.AND.RATIO1.GT.EDIT) SIG1=-SIG1
      IF(NEDIT1.GT.0) EAST1=NCEDIT
      IF(SIG1.LT.0.D0)EAST1=ASTRSK
      IF(SIG1.EQ.0.D0) NEDIT1=0
      IF(NMEAS.EQ.1) GO TO 1529
      RATIO2=0.D0
      EAST2=BLANK
      IF(SIG2.NE.0.D0) RATIO2=DABS(RESID2/SIG2)
      IF(NEDIT2.EQ.0.AND.RATIO2.GT.EDIT) SIG2=-SIG2
      IF(NEDIT2.GT.0) EAST2=NCEDIT
      IF(SIG2.LT.0.D0)EAST2=ASTRSK
      IF(SIG2.EQ.0.D0) NEDIT2=0
      J=MTYPE+7
C SUM RESIDUAL FOR SECOND MEASUREMENT INTO STATISTICS
      CALL STAINF(2,ISTA,J,SIG2,RESID2,P,ISAT,P)
      MINDEX=MINDEX+1
C SUM RESIDUAL & PARTIALS FOR SECOND MEASUREMENT INTO ESTIMATION
      IF(SIG2.GT.0.D0.AND.NEQN(ISAT).GT.1) CALL ESTIM(2,SIG2,RESID2,
     *   PMPX0(1,2))
C REMOVE ELECTRONIC BIAS COMPONENT FROM STATISTICS & ESTIMATOR FOR
C SECOND MEASUREMENT
      IF(NEDIT2.GT.0) CALL BSCOMP(NEDIT2,RESID2,SIG2,DAY1,PMPX0(1,2),
     *   LINNER)
      IF(MTYPE.EQ.4.OR.MTYPE.EQ.5) GO TO 1529
      OBS02=OBS02/RAD
      RESID2=RESID2/RSEC
 1529 J=MTYPE
      IF(NBIAS.NE.0) PARNOS(NPARM-7)=BINDX1
      ISTA2=CHANEL
C SUM RESIDUAL FOR FIRST MEASUREMENT INTO STATISTICS
      CALL STAINF(2,ISTA,J,SIG1,RESID1,P,ISAT,ISTA2)
      MINDEX=MINDEX+1
C SUM RESIDUAL & PARTIALS FOR FIRST MEASUREMENT INTO ESTIMATION
      IF(SIG1.GT.0.D0.AND.NEQN(ISAT).GT.1) CALL ESTIM(2,SIG1,RESID1,
     *   PMPX0)
C REMOVE ELECTRONIC BIAS COMPONENT FROM STATISTICS & ESTIMATOR FOR
C FIRST MEASUREMENT
      IF(NEDIT1.GT.0) CALL BSCOMP(NEDIT1,RESID1,SIG1,DAY1,PMPX0,LINNER)
      IF(NOPRNT) GO TO 1600
```

```
      SATSAT=.FALSE.
      GO TO (1540,1580,1585,1535,1530,1540,1540),MTYPE
      K=MTYPE-14
      GO TO (1580,1580,1580,1585,1585,1535,1580,1200,1250,1250,1250,
     .   1250,1220,1240,1585,1585),K
C CONVERT UNITS OF RESIDUAL & OBSERVATION OF PAIRED MEASUREMENT FOR
C OUTPUT ONLY
1530  OBSO1=OBSC1*1000.D0
      OBSO2=OBSC2*1000.D0
      RESID1=RESID1*1000.D0
      RESID2=RESID2*1000.D0
      GO TO 1550
1535  OBSO1=OBSC1*1.0D-3
      RESID2=RESID2*100.D0
      GO TO 1550
 1540 OBSO1=OBSC1/RAD
      RESID1=RESID1/RSEC
 1550 IM=MINDEX-2
C PRINT RESIDUAL & OBSERVATION FOR PAIRED MEASUREMENT
      WRITE(OUTP,10103) IYMD,IHM,SEC,ISAT,NAME(ISTA),ATYPE(MTYPE),OBSO1,
     .   RESID1,RATIO1,EAST1,ATYPE(MTYPE+7),OBSO2,RESID2,RATIO2,EAST2,
     .   ELEV(ISAT),IM
      IF(.NOT.PARTGF) GO TO 1600
      IF(SIG1.GT.0.0D0) WRITE(OUTP,46000) PMPXO
      IF(SIG2.GT.0.0D0) WRITE(OUTP,46000) (PMPXO(J,2),J=1,NPARM)
      GO TO 1600
C CONVERT UNITS OF RESIDUAL & OBSERVATION OF UNPAIRED MEASUREMENT FOR
C OUTPUT ONLY
 1200 OBSO1=OBSC1*1.0D6
      RESID1=RESID1*1.0D6
      GO TO 1590
 1220 OBSO1=OBSC1*1.0D6
      RESID1=RESID1*1.0D9
      GO TO 1590
 1240 OBSO1=OBSC1*1.0D+3
      RESID1=RESID1*1.0D+6
      GO TO 1590
 1250 OBSO1=OBSC1/DRAD
      RESID1=RESID1/DRSEC
      GO TO 1590
1580  OBSO1=OBSC1*1.0D-3
      GO TO 1588
1585  RESID1=RESID1*100.D0
 1588 SATSAT=KKSAT.GT.0.AND.KKSAT.LE.NSAT
 1590 IM=MINDEX-1
      SNAME=PCE
      IF(ISTA.NE.0) SNAME=NAME(ISTA)
C PRINT RESIDUAL & OBSERVATION FOR UNPAIRED MEASUREMENT
      WRITE(OUTP,10104) IYMD,IHM,SEC,ISAT,SNAME,ATYPE(MTYPE),OBSO1,
     .   RESID1,RATIO1,EAST1,ELEV(ISAT),IM
      IF(MTYPE.GT.26) GO TO 1592
      IF(SATSAT) WRITE(OUTP,47000) KKSAT
      GO TO 1598
 1592 K=MTYPE-26
      GO TO (1594,1594,1596,1596),K
 1594 WRITE(OUTP,47100) SNAME,NAME(CHANEL),RANDOT(1)
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      GO TO 1598
 1596 J=K-1
      WRITE(OUTP,47200) J,NAME(CHANEL),RANDOT(1)
 1599 IF(PARTOF.AND.SIG1.GT.0.0D0) WRITE(OUTP,46000) PMPXO
 1600 DAY0=DAY1
      IF(.NOT.PLOTSW) GO TO 1610
      IF(DATASW) GO TO 1620
C WRITE BINARY RESIDUAL TAPE
      INET=0
      IF(ISTA.GT.0) INET=ISTANO(ISTA)/1000
      J=MTYPE
      RATE1=OBSDOT(J,ISTA,RATE2)
      BRSEC=SEC
      IF(SIG1.LE.0.0D0) RATIO1=0.0D0
      IF(SIG2.LE.0.0D0) RATIO2=0.0D0
      BRSID1(1)=RESID1
      BRSID1(2)=RATIO1
      BRSID1(3)=RATE1
      BRSID2(1)=RESID2
      BRSID2(2)=RATIO2
      BRSID2(3)=RATE2
      WRITE(PLOTP) IYMD,IHM,BRSEC,NAME(ISTA),ATYPE(MTYPE),OBSO1,BRSID1,
     .   OBSO2,BRSID2,ELEV(ISAT),INET,RANDOT,PEROUT
 1610 CONTINUE
      IF(.NOT.GRDSW) GO TO 1620
C WRITE GROUND TRACK TAPE
      WRITE(GRDTP,10125) IYMD,IHM,SEC,NAME(ISTA),SATLAT(ISAT),
     .   SATLON(ISAT),SATH(ISAT)
 1620 IF(.NOT.SATSW) GO TO 700
      BRSEC=SEC
C WRITE RV TAPE
      IF(RVTP.GT.0) WRITE(RVTP) DAY1,IYMD,IHM,BRSEC,(XYZEND(I,1),I=1,6),
     .   SATLAT(1),SATLON(1),SATH(1)
      IF(LINESX.GT.0)GO TO 1650
      IF(XYZTP.NE.OUTP) WRITE(XYZTP,10302) TITLE
      IF(KEPLER) WRITE(KPLRTP,10302) TITLE
 1650 IF(MOD(LINESX,48).NE.0) GO TO 1675
      WRITE(XYZTP,10300) ARCNO,INNER,OUTER,NPAGEX
      IF(KEPLER) WRITE(KPLRTP,44480) ARCNO,INNER,NPAGEX
      NPAGEX=NPAGEX+1
 1675 IF(MOD(LINESX,6).EQ.0) WRITE(XYZTP,10102)
C WRITE CARTESIAN EPHEMERIS
      WRITE(XYZTP,10301) IYMD,IHM,SEC,(XYZEND(J,1),J=1,6),SATLAT(1),
     .   SATLON(1),SATH(1)
      IF(.NOT.(CHETSW.AND.SIMDAT.AND.PCESIM)) GO TO 1630
      DO 1625 I=1,6
      I1=I+12
C WRITE CARTESIAN PCE SIMULATED DATA ON UNIT 17 IN ORBIT GENERATOR
C MODE ONLY
 1625 WRITE(17,10400) I1,IYMD,IHM,SEC,XYZEND(I,1)
 1630 IF(NSAT.LT.2) GO TO 1670
      DO 1665 I=2,NSAT
      LINESX=LINESX+1
 1665 WRITE(XYZTP,10304) (XYZEND(J,I),J=1,6),SATLAT(I),SATLON(I),SATH(I)
 1670 LINESX=LINESX+1
      IF(.NOT.KEPLER) GO TO 700
```

```
C WRITE KEPLER EPHEMERIS
      IF(MOD(LINESX,6).EQ.NSAT) WRITE(KPLRTP,10102)
      DO 1650 J=1,NSAT
      CALL ELEM(XYZEND(1,J),AEINPM,2,.FALSE.,P)
      DO 1600 I=3,6
      I2=I-2
 1600 CALL OUTRAD(AEINPM(I),IDEG(I2),MIN(I2),SECA(I2),1)
      IF(J.NE.1) GO TO 1685
      WRITE(KPLRTP,44485) IYMD,IHM,SEC,(AEINPM(I),I=1,2),(IDEG(I),
     .   MIN(I),SECA(I),I=1,4)
      IF(PCESIM.OR..NOT.(ORBTSW.AND.SIMDAT)) GO TO 1690
C WRITE KEPLER PCE  SIMULATED DATA ON UNIT 17 IN ORBIT GENERATOR MODE
C ONLY
      DO 1640 I=1,6
      IF(I.GT.2) AEINPM(I)=AEINPM(I)/DRAD
      I1=20+I
 1640 WRITE(17,10600) I1,IYMD,IHM,SEC,AEINPM(I)
      GO TO 1650
 1685 WRITE(KPLRTP,44485) (AEINPM(I),I=1,2),(IDEG(I),MIN(I),SECA(I),
     .   I=1,4)
 1650 CONTINUE
      GO TO 700
C SUMMARIZE INFORMATION DERIVED FROM ITERATION
 2000 IF(ORBTSW)GO TO 3000
      IF(MINDEX.EQ.1.AND.CORR1.GT.0.D0)GO TO 3000
      IF(MINDEX.EQ.1) CALL ERROR(4,CARDID)
      IF(NEBIAS.GT.0) CALL BSCOMP(0,RESID1,SIG1,DAY1,RMPX0,LINNER)
      NOATUB=0
      IF(.NOT.NOPRNT) WRITE(OUTP,10101)
C SUMMARIZE STATISTICS FOR ITERATION
      CALL SUMMRY(ARCNO,INNER,OUTER,LINNER,EDIT,NAME)
      IF(NEON(1).EQ.1) GO TO 3000
C LIST EPOCH AND LAST ELEMENT SET
      DO 2440 I=1,12
      ORBELP(I,1)=ORBELA(I,1)
 2440 ORBELA(I,1)=ELEMST(I,1)
      CALL ESTIM(3,R,P,P)
      DO 2475 L=1,NSAT
      IOUT=1
      IF(SATSW)IOUT=2
      DO 2300 I=1,IOUT
      IUNIT=ITAPE(I)
      WRITE(IUNIT,10105) L,ARCNO,INNER,OUTER,IEPYMD,IEPHM,EPSEC
      WRITE(IUNIT,10103) APRIOR,BLANK,(ELEMIN(J,L),J=1,6)
 2300 WRITE(IUNIT,10103) PREV,BLANK,(ORBELA(J,L),J=1,6)
C SAVE OLD KEPLERIAN ELEMENTS AND COMPUTE STANDARD DEVIATIONS
      RMSPOS(1)=0.D0
      RMSPOS(2)=0.D0
      I1=6*(L-1)
      I1=I1*(NDIM-1)-I1*(I1-1)/2
      I6=6*(L-1)
      DO 2400 I=1,6
      I1=I+I6
      I2=I1+I1
      I3=1+(I-1)/3
      RMSPOS(I3)=RMSPOS(I3)+SUM1(I2)
```

```
      IF(SUM1(I2).LT.0.D00) WRITE(OUTP,45000) II
      SDELEM(I)=DSQRT(DABS(SUM1(I2)))
 2400 I1=I1+NDIM-II
      RMSPOS(1)=DSQRT(DABS(RMSPOS(1)))
      RMSPOS(2)=DSQRT(DABS(RMSPOS(2)))
      PVRMS(1,L)=RMSPOS(1)
      PVRMS(2,L)=RMSPOS(2)
C CALCULATE ADJUSTED ELEMENTS IN THE FORM OF KEPLERIAN ELEMENTS
      CALL ELEM(ELEMST(1,L),ORBELA(1,L),1,.FALSE.,P)
      DO 2470 I=1,6
 2470 ORBDIF(I)=ORBELA(I,L)-ORBELP(I,L)
C LIST CORRECTED ELEMENTS AND UNCERTAINTIES
      WRITE(OUTP,10109) ADJUST,BLANK,(ELEMST(J,L),J=1,6)
      J1=(L-1)*6+1
      J2=J1+5
      WRITE(OUTP,10109) CHANGE,BLANK,BLANK,(DELTA(J),J=J1,J2)
      WRITE(OUTP,10109) SD,SDELEM
      WRITE(OUTP,10110) RMSPOS
      WRITE(OUTP,10115)
      WRITE(OUTP,10116) APRIOR,BLANK,(ORBEL(J,L),J=1,6),PREV,BLANK,
     .   (ORBELP(J,L),J=1,6),ADJUST,BLANK,(ORBELA(J,L),J=1,6),CHANGE,
     .   BLANK,BLANK,ORBDIF
 2475 WRITE(OUTP,10303) L,ARCNO,(ELEMST(J,L),J=1,6)
      IF(INPAR1.EQ.(6*NSAT)) GO TO 2503
C UPDATE & PRINT ADJUSTED ARC FORCE MODEL PARAMETERS
      WRITE(OUTP,10501) ARCNO,INNER,OUTER
      I1=0
      DO 2640 I=1,3
      DO 2640 L=1,NSAT
      IF(ADJF(L,I).LE.0) GO TO 2640
      I1=I1+1
      IF(MOD(I1,45).EQ.1) WRITE(OUTP,10520)
      IF(L.EQ.1) WRITE(OUTP,10102)
      J=ADDR(L,I)+6*NSAT
      J1=INDXNC(J)+J
      IF(SUM1(J1).LT.0.D00) WRITE(OUTP,45010) L,DRAG(I)
      SIG1=DSQRT(DABS(SUM1(J1)))
      CD(L,I)=BBIAS(I+3*(L-1))
      WRITE(OUTP,10505) L,DRAG(I),DRAG0(L,I),CD(L,I),DRAGSG(L,I),SIG1
 2640 CONTINUE
      IF(I1.GT.0) I1=5*NSAT
      IF(NGPARC.LE.0) GO TO 2503
      K1=0
      DO 2660 I=1,NGPARC
      I1=I1+1
      IF(MOD(I1,5).NE.1) GO TO 2650
      IF(K1.NE.0.AND.MOD(I1,45).NE.1) GO TO 2645
      K1=1
      WRITE(OUTP,10530)
 2645 WRITE(OUTP,10102)
 2650 J=INDXCS(1,I)
      N=INDXCS(2,I)
      M=INDXCS(3,I)
      C1=GPVAL(I)*GPSIG(I)
      C2=GPVALC(I)*GPSIG(I)
      IF(J.EQ.1) CS(N,M+1)=C1
```

```
      IF(J.EQ.2) CS(31-N,33-M)=C1
      K=GPNO(I)
      K=INDXNO(K)+K
      SIG1=DSQRT(SUM1(K))
      SIG2=UPSIG(I)*SIG1
 2650 WRITE(OUTP,10510) ICS(J),N,M,C2,C1,SIG1,SIG2
C CORRECT & PRINT ADJUSTED BIASES
 2503 IF(NBIAS.EQ.0) GO TO 2540
      WRITE(OUTP,10440) ARCNO,INNER,OUTER
      LINES=0
      DO 2504 I=1,NBIAS
      IB=I+3
      IF(MOD(LINES,5).EQ.0)  WRITE(OUTP,10102)
      LINES=LINES+1
      I1=INPARI+I
      I1=INDXNO(I1)+I1
      IF(SUM1(I1).LT.0.0D0) WRITE(OUTP,45030)
      SIG1=DSQRT(DABS(SUM1(I1)))
      CALL DATES(BSTRT(I),IYMD1,IHM1,SEC)
      CALL DATES(BSEND(I),IYMD2,IHM2,SEC)
      C1=TIMING
      I1=BYTPE(I)
      IF(I1.NE.0) C1=ATYPE(I1)
      J=BSTANO(I)
      I2=BYTPE(I)+1
      IF(I1.LT.15) UNIT=UNITS(I2)
      IF(I1.EQ.27) UNIT=UNITS(1)
      IF(I1.EQ.28) UNIT=HERTZ
      IF(I1.GT.29) UNIT=UNITS(4)
 2504 WRITE(OUTP,10450) NAME(J),ISTANO(J),C1,BIAS0(I),BBIAS(IB),
     *    UNIT,BIASSG(IB),SIG1,IYMD1,IHM1,IYMD2,IHM2
C INCREMENT ITERATION COUNTER
 2540 IF(LINNER) GO TO 3000
      IF(FITER) GO TO 2580
C CHECK FOR INNER ITERATION CONERGENCE
      IF(ABS(RMSLST-RMSTOT)/RMSLST.LE.CONVRG) LINNER=.TRUE.
      IF(.NOT.LINNER) GO TO 2590
      WRITE(OUTP,46500) ARCNO,INNER,OUTER,CONVRG
      CALL ERROR(10,CARDID)
 2590 INNER=INNER+1
      LINNER=(LINNER.AND.INNER.GE.INNMIN).OR.INNER.GE.INNMAX
      GO TO 650
 3000 IF(.NOT.CRBTSW) GO TO 3010
      IF(RVTP.EQ.0) GO TO 3010
C WRITE LAST RECORD FOR RV TAPE
      BASEC=EPSEC
      DAYSTA=999.D0
      WRITE(RVTP) DAYSTA,IEPYMD,IEPHM,BESEC,(ELEMIN(I,1),I=1,5)
     *    ,SATLAT(1),SATLON(1),SATH(1)
      END FILE RVTP
 3010 IF(SATSW) WRITE(XYZTP,10102)
      IF(SIMLAT.AND.OUTER.EQ.1) ENDFILE 17
C PRINT ARC SUMMARY PAGE
      CALL TYPCAR(DAYO,MINDEX,ARCNO,OUTER,RVRMS)
      IF(INPAR.EQ.0) GO TO 3020
      IF(.NOT.LOUTER) GO TO 4000
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      IF(.NOT.FLOTSW) GO TO 3016                                        1NON 840
      WRITE(FLOTF) (BLANK,J=1,16)                                       1NON 841
      END FILE FLOTF                                                    1NON 842
C PUNCH ADJUSTED ELEMENTS OF DRAG & SOLRAD                               1NON 843
 3016 PUNCH 44500,ARCNO,INNER,OUTER,IYREF,IEPYMD,IEPHM,EPSEC,           1NON 844
     . ((ELEMST(I,L),I=1,6),L=1,NSAT)                                   1NON 845
      DO 3019 L=1,NSAT                                                  1NON 846
      IF(BO(L).LE.0.0D0) GO TO 3019                                     1NON 847
      IF(CD(L,1).GT.0.0D0) PUNCH 44510,DRAG(1),L,CD(L,1),DRAGSG(L,1),   1NON 848
     .   CD(L,2),DRAGSG(L,2)                                            1NON 849
      IF(CD(L,3).GT.0.0D0) PUNCH 44510,DRAG(3),L,CD(L,3),DRAGSG(L,3)    1NON 850
      PUNCH 44530,L,ASAT(L),MSAT(L)                                     1NON 851
 3019 CONTINUE                                                          1NON 852
 3020 CONTINUE                                                          1NON 853
 4000 ESTSTA=MINDEX                                                     1NON 854
      IF(OUTER.EQ.1) INNMAX=MAX2IN                                      1NON 855
C STORE UPDATED ARC PARAMETERS                                          1NON 856
      CALL DATARD(ARCNO,.FALSE.,.TRUE.,.TRUE.)                          1NON 857
      IF(ORBTSW) GO TO 4055                                             1NON 858
C PRINT ADJUSTED PARAMETER VARIANCE-COVARIANCE MATRIX                   1NON 859
      WRITE(OUTP,48000)                                                 1NON 860
      K1=-6                                                             1NON 861
      J1=0                                                              1NON 862
      DO 4050 L=1,NSAT                                                  1NON 863
      K1=K1+6                                                           1NON 864
      L1=K1+1                                                           1NON 865
      L2=K1+6                                                           1NON 866
      WRITE(OUTP,10106) L,ARCNO,(TTL(J),J=L1,L2)                        1NON 867
      DO 4050 J=L1,L2                                                   1NON 868
      I1=J1+J                                                           1NON 869
      I2=J1+6+K1                                                        1NON 870
      WRITE(OUTP,10111) TTL(J),(SUM1(K),K=I1,I2)                        1NON 871
 4050 J1=J1+NDIM-J                                                      1NON 872
      IF(NSAT.GT.1) WRITE(OUTP,48000)                                   1NON 873
      CALL CORREL(SUM1,NPARAM,NDIM,INNER,TTL)                           1NON 874
 4055 IF(DORB1.LT.0.D0.OR..NOT.LITER)GO TO 4060                         1NON 875
      IF(DAYO-DORB1.GT.0.0010D0)INITAL=.TRUE.                           1NON 876
      TOREFO=ORBRT.GT.1.0D0                                             1NON 877
      IF(TOREFO) ORBRT=ORBRT-1.0D0                                      1NON 878
      TOREFT=TOREFO                                                     1NON 879
      IF(IDSAT(1).GT.0) ISATID(1)=IDSAT(1)                              1NON 880
      MISLOG(3)=0                                                       1NON 881
C WRITE OPTIONAL ORB1 TAPE ON LAST INNER ITERATION OF LAST OUTER        1NON 882
C ITERATION                                                             1NON 883
 4056 CALL ORB1(ORBRT)                                                  1NON 884
 4060 ARCNO=ARCNO+1                                                     1NON 885
      IF(NGPARC.LE.0) GO TO 4090                                        1NON 886
C REPLACE ARC GEOPOTENTIAL COEFFICIENTS WITH APRIORI VALUES             1NON 887
      DO 4080 I=1,NGPARC                                                1NON 888
      J=INDXCS(1,I)                                                     1NON 889
      N=INDXCS(2,I)                                                     1NON 890
      M=INDXCS(3,I)                                                     1NON 891
      C1=GPVALO(I)*GPSIG(I)                                             1NON 892
      IF(J.EQ.1) CS(N,M+1)=C1                                           1NON 893
 4080 IF(J.EQ.2) CS(31-N,33-M)=C1                                       1NON 894
 4090 IF(ARCNO.GT.NARCS) GO TO 4100                                     1NON 895
```

```
      GO TO 300
C STOP IF NO COMMON PARAMETERS ADJUSTED OR IF B-MATRIX OUTPUT
 4100 IF(NSTEST.EQ.0.AND.NGPCOM.EQ.0.AND.NTIDEN.EQ.0) STOP 1
      IF(BMAT) STOP 4
      ENDFILE SCRC
      REWIND SCRC
      RMSPRV=RMSALL
      NMALL=3*NMAST+NGPCOM+NTIDEN
C OBTAIN STATISTICAL SUMMARY FOR ALL ARCS
      CALL STAINF(4,0,NMALL,RMSALL,P,P,P,P)
C PRINT RESIDUAL SUMMARY FOR ALL ARCS
      WRITE(OUTP,20111) OUTER
      DO 4120 I=1,7
      IF(NOALL(I).GT.0) WRITE(OUTP,20120) ATYPE(I),NOALL(I),TYPRMS(I)
      J=I+7
      IF(NOALL(J).GT.0) WRITE(OUTP,20120) ATYPE(J),NOALL(J),TYPRMS(J)
 4120 IF(NOALL(I)+NOALL(J).GT.0) WRITE(OUTP,10102)
      DO 4130 I=15,30
      IF(NOALL(I).LE.0) GO TO 4130
      WRITE(OUTP,20120) ATYPE(I),NOALL(I),TYPRMS(I)
      WRITE(OUTP,10102)
 4130 CONTINUE
      WRITE(OUTP,20130) NMALL,RMSALL
      IF(LOUTER.AND.LITRES) GO TO 10220
      ESTSTA=NMAST
C CALCULATE ADJUSTED COMMON PARAMETER VALUES
      CALL ESTIM(A,B,P,P)
C PRINT ADJUSTED COMMON PARAMETERS
      CALL COMADJ(OUTER)
C UPDATE ARC ADJUSTED PARAMETERS FOR ADJUSTMENTS TO COMMON PARAMETERS
 2620 CALL UPDATE(NARCS,OUTER,NSTART,SUMI,DELTA,
     .   DDELTA,TTL,BSNOS,BBIAS)
      IDIM=NGPCOM+3*NMAST+NTIDEN
      IST=INDXNC(NSTART)+NSTART
C PRINT COMMON ADJUSTED PARAMETER VARIANCE-COVARIANCE MATRIX
      CALL CORREL(SUMI(IST),IDIM,IDIM,OUTER,TTL(NSTART))
      IF(LOUTER) GO TO 10220
      OUTER=OUTER+1
C CHECK FOR OUTER ITERATION CONVERGENCE AND INCREMENT ITERATION
C COUNTER
      IF((RMSPRV-RMSALL)/RMSPRV.LE.OUTCON) LOUTER=.TRUE.
      IF(OUTER.LE.MINOUT) LOUTER=.FALSE.
      LOUTER=LOUTER.OR.OUTER.GE.MAXOUT
      ARCNO=1
C REWIND SCRATCH FILE
      REWIND SCRA
      REWIND SCRC
      REWIND FLIP
      IF(.NOT.STARTW) GO TO 200
C REWIND OUTPUT RESTART TAPE
      END FILE OUTSTR
      REWIND OUTSTR
      GO TO 200
10220 IF(.NOT.GRDSW) STOP 2
C ENDFILE GROUND TRACK TAPE
      ENDFILE GRDTP
```

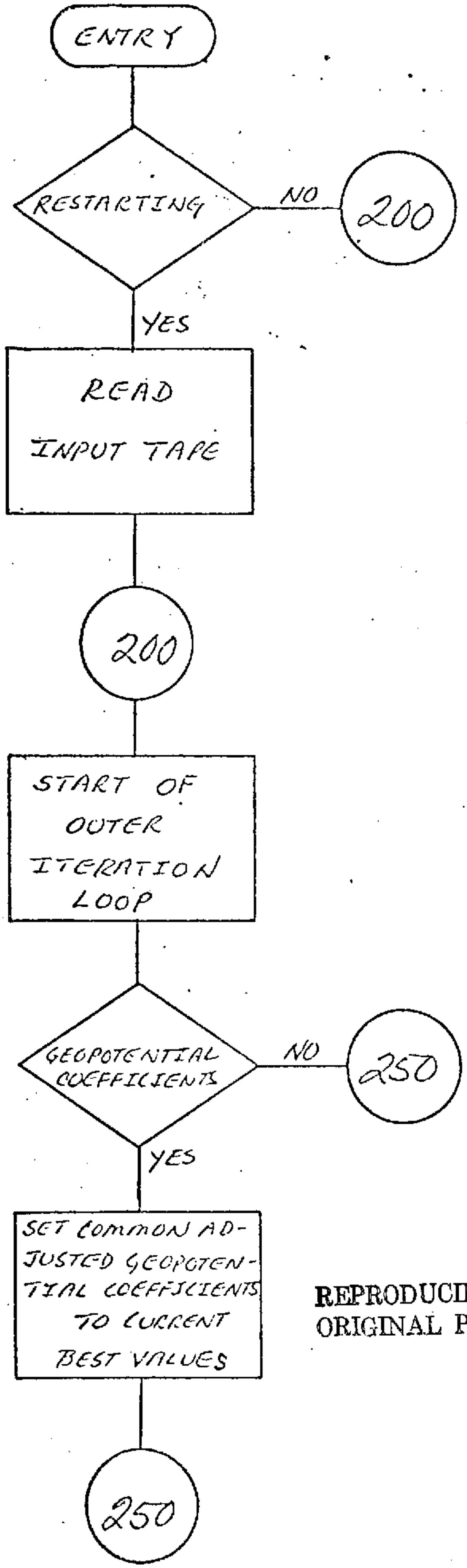
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

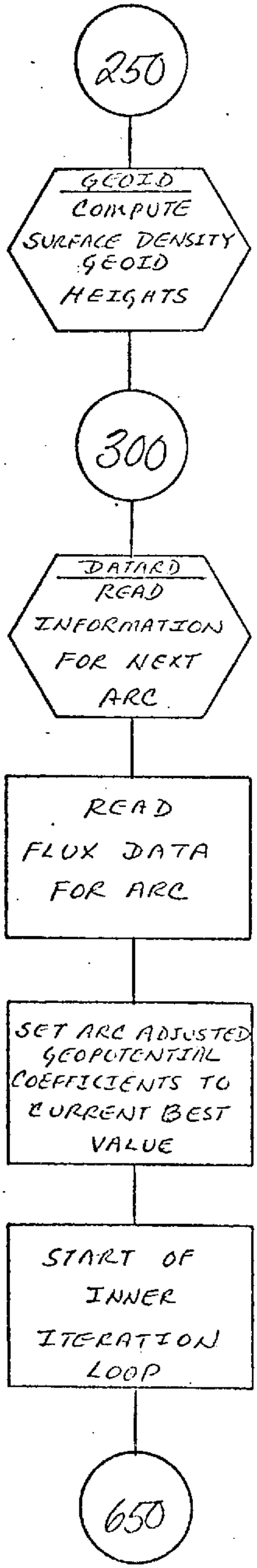
```
      STOP 3                                                            1NON 952
C FORMAT STATEMENTS                                                     1NON 953
10001 FORMAT(I6,I4,F7.4,I2,I1,2PF2.0,I8,2(I6,I4,0PF7.4),I7,I1,A8)       1NON 954
10125 FORMAT(I6,2X,I4,2X,F7.4,2X,A6,2X,F15.8,2X,F15.8,2X,F15.5)         1NON 955
10002 FORMAT(3D24.16)                                                   1NON 956
10440 FORMAT(1H1,19X,3HARC,I3,46H ADJUSTED PARAMETER SUMMARY FOR INNER I1NON 957
     1TERATION,I3,19H OF OUTER ITERATION,I2/ 1H0,10X,7HSTATION,8X,      1NON 958
     2    9HPARAMETER,8X,15HPARAMETER VALUE,13X,                        1NON 959
     2    15HSTANDARD  DEVIATION,4X,24HCOVERAGE --- YYMMDD HHMM/9X,     1NON 960
     3    12HNAME  NUMBER,7X,4HTYPE,9X,8HA PRIORI,3X,8HADJUSTED,11X,    1NON 961
     4    8HA PRIORI,3X,8HADJUSTED,7X,5HBEGIN,9X,3HEND/)                1NON 962
10450 FORMAT(3X,A6,I6,5X,A6,5H BIAS,2X,1P2E11.3,2X,A6,2E11.3,2X,2(I8,I5)1NON 963
     .)                                                                 1NON 964
10100 FORMAT(1H1,28X,3HARC,I3,30H RESIDUALS FOR INNER ITERATION,I3,     1NON 965
     .    19H OF OUTER ITERATION,I2,15X,4HPAGE,I4//5X,12HTIME OF DATA,  1NON 966
     .    7X,3HSAT,2X,                                                  1NON 967
     .    7HSTATION,4X,11HMEASUREMENT,9X,8HRESIDUAL,3X,5HRATIO,6X,      1NON 968
     .    11HMEASUREMENT,8X,8HRESIDUAL,3X,5HRATIO,3X,4HELEV,5X,4HMEAS/  1NON 969
     .    1X,33HYYMMDD HHMM SS.SSSSSS TRKED  NAME,4X,4HTYPE,6X,5HVALUE, 1NON 970
     .    6X,5H(O-C),4X,14HTO SIGMA  TYPE,6X,5HVALUE,7X,5H(O-C),4X,     1NON 971
     .    14HTO SIGMA (DEG),5X,2HNO)                                    1NON 972
10101 FORMAT('0 UNITS FOR INFORMATION ... LINEAR',6X,                   1NON 973
     .    'MEASUREMENTS - KILOMETERS',5X,'RESIDUALS - METERS'/          1NON 974
     .    26X,'ANGULAR',5X,'MEASUREMENTS - DEGREES',8X,                 1NON 975
     .    'RESIDUALS - SECONDS OF ARC'/23X,                             1NON 976
     .    'LINEAR RATE MEASUREMENTS - METERS/SECOND  RESIDUALS - ',     1NON 977
     .    'CENTIMETERS/SECOND'/28X,'TIME',8X,                           1NON 978
     .    'MEASUREMENTS - MICROSECONDS  RESIDUALS - NANOSECONDS'/23X,   1NON 979
     .    'FRINGE RATE MEASUREMENTS - MILLIHERTZ',5X,                   1NON 980
     .    'RESIDUALS - MICROHERTZ'/                                     1NON 981
     .    35X'DIRECTION COSINES ARE SCALED BY A FACTOR OF 1000'/        1NON 982
     .    36X,'ECCENTRICITY IS SCALED BY A FACTOR OF 1000000')          1NON 983
10102 FORMAT(1X)                                                        1NON 984
10103 FORMAT(1X,I5,I5,F10.6,4H SAT,I2,1X,A7,1X,A6,F12.4,F13.3,F8.2,A1,  1NON 985
     .    2X,A6,F11.4,F13.3,F8.2,A1,F7.2,I3)                            1NON 986
10104 FORMAT(1X,I5,I5,F10.6,4H SAT,I2,1X,A7,1X,A6,F12.4,F13.3,F8.2,A1,  1NON 987
     .    41X,F7.2,I5)                                                  1NON 988
10106 FORMAT(1H1,17X,9HSATELLITE,I2,4H ARC,I3,                          1NON 989
     .    33H RECTANGULAR COORDINATE SUMMARY FOR,                       1NON 990
     .    16H INNER ITERATION,I3,19H OF OUTER ITERATION,I2/             1NON 991
     .        1H0,22X,34HEPOCH OF ELEMENTS - YEAR,MONTH,DAY,I7,3X,      1NON 992
     .        16HHOUR,MINUTE,SECOND,I5,F8.4/1H0,33X,1HX,12X,1HY,12X,1HZ,1NON 993
     .        10X,4HXDOT,9X,4HYDOT 9X,4HZDOT/23X,3(10X,3H(M)),1X,3(8X,  1NON 994
     .        5H(M/S))/1X)                                              1NON 995
10109 FORMAT(6X,3A6,1X,3F13.1,3F12.4/1X)                                1NON 996
10110 FORMAT(1H0/35X,12HRMS POSITION,F11.4,4X,12HRMS VELOCITY,F11.6)    1NON 997
10108 FORMAT(1H0/29X,'SATELLITE',I2,4H ARC,I4,'VARIANCE/COVARIANCE ',   1NON 998
     .    'MATRIX OF COORDINATES'/1H0,20X,A6,9X,A6,9X,A6,9X,A6,9X,A6,   1NON 999
     .    9X,A6)                                                        1NON1000
10111 FORMAT(1H0,9X,A6,1X,1P6D15.7)                                     1NON1001
10115 FORMAT(1H0/47X25HKEPLERIAN ELEMENT SUMMARY/1H0,33X,1HA,13X,1HE12X,1NON1002
     .        1HI,7X,33HRA ASC NODE  ARG PERIGEE  MEAN ANOMALY/31X,     1NON1003
     .        8H(METERS),16X,4(4X,9H(DEGREES))/1X)                      1NON1004
10116 FORMAT(5X,3A6,F14.1,F13.9,4F13.5/1X)                              1NON1005
10501 FORMAT(1H1,15X,3HARC,I2,                                          1NON1006
     .    52H ADJUSTED FORCE MODEL PARAMETERS FOR INNER ITERATION,I3,   1NON1007
```

ORIGINAL PAGE IS POOR

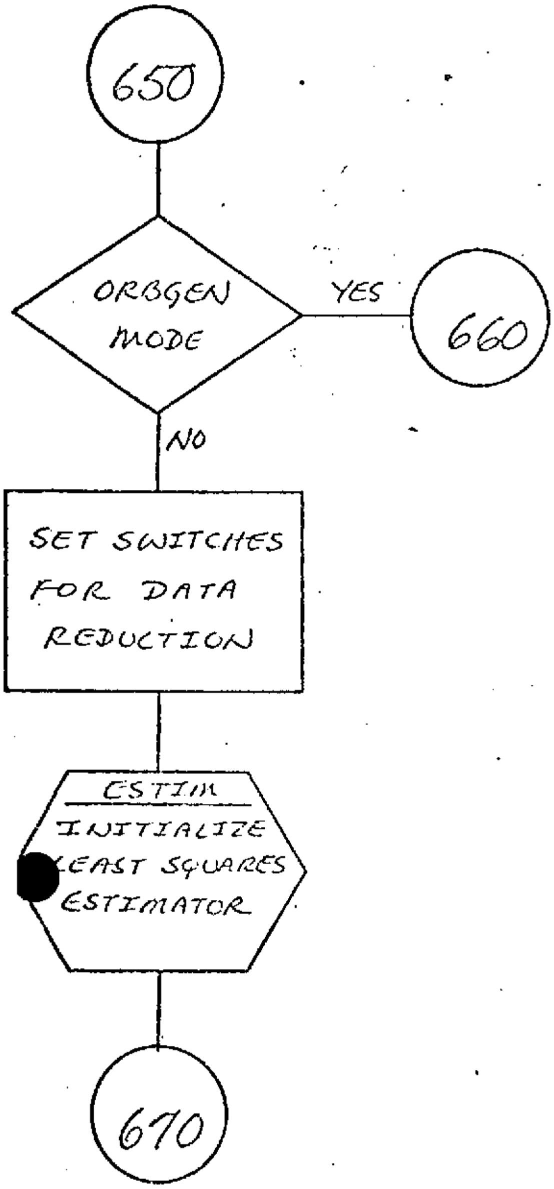
ORIGINAL PAGE IS POOR

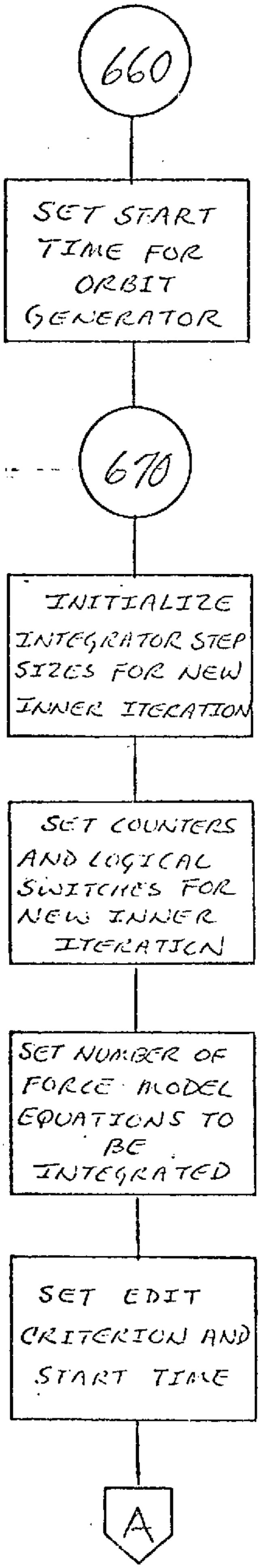
```
     .  15H OF OUTER ITERATION,I3)                                      1NON1008
10520 FORMAT(1H0,40X,33HSATELLITE PHYSICAL PARAMETERS ADJUSTED//20X,    1NON1009
     .  11HCOEFFICIENT,19X,5HVALUE,21X,18HSTANDARD DEVIATION/41X,       1NON1010
     .  8HA PRIORI,7X,8HADJUSTED,10X,8HA PRIORI,8X,8HADJUSTED)          1NON1011
10530 FORMAT(1H0,37X,44HGEOPOTENTIAL RESONANCE COEFFICIENTS ADJUSTED//  1NON1012
     .  20X,11HCOEFFICIENT,19X,5HVALUE,21X,18HSTANDARD DEVIATION/41X,   1NON1013
     .  8HA PRIORI,7X,8HADJUSTED,6X,28HRATIO TO A PRIORI   ADJUSTED)    1NON1014
10505 FORMAT(19X,3HSAT,I2,1X,A8,D19.6,D15.6,2D16.4)                     1NON1015
10510 FORMAT(22X,A1,1H(,I2,1H,,I2,1H),D21.6,D15.6,1X,2D16.4)            1NON1016
10202 FORMAT(8HOSTATION,I5,8H AT TIME,I7,I5,F8.4,18H NOT IN TIME ORDER) 1NON1017
10300 FORMAT(1H1,25X,3HARC,I3,31H SATELLITE EPHEMERIS FOR INNER ,       1NON1018
     .  9HITERATION,I3,19H OF OUTER ITERATION,I2,17X,4HPAGE,            1NON1019
     .  I3/17H0     TIME OF DATA,11X,1HX,12X,1HY,12X,1HZ,9X,4HXDOT,7X,  1NON1020
     .  4HYDOT,7X,4HZDOT,4X,33HGEODETIC LAT  LONG (E)      HEIGHT/     1NON1021
     .  20H YYMMDD HHMM SS.SSSS,6X,4H(KM),2(9X,4H(KM)),1X,              1NON1022
     .  3(6X,5H(M/S)),5X,2(9H(DEGREES),2X),4X,4H(KM))                   1NON1023
10301 FORMAT(1X,I6,I5,F8.4,-3P3F13.4,0P3F11.4,F12.6,F11.5,-3P3F13.4,    1NON1024
     .  0P3F11.4,F12.6,F11.5,-3PF13.4)                                  1NON1025
10304 FORMAT(20X,-3P3F13.4,0P3F11.4,F12.6,F11.5,-3PF13.4)               1NON1026
10302 FORMAT((1H1,12(/1H0),3(/20X,20A4)))                               1NON1027
10303 FORMAT(1H0,44X,9HSATELLITE,I2,30H CURRENT BEST ELEMENTS FOR ARC,  1NON1028
     .  I3//                                                            1NON1029
     .        ,4),1HX,25X,1HY,25X,1HZ//                                 1NON1030
     .        30X,3D24.16//42X,4HXDOT,22X,4HYDOT,22X,4HZDOT//30X,3D24.16)1NON1031
10600 FORMAT(1X,I2,I6,I4,F8.4,D24.16,10X)                               1NON1032
20111 FORMAT(1H1,32X,'TOTAL RESIDUAL SUMMARY BY MEASUREMENT TYPE FOR ', 1NON1033
     1        'OUTER ITERATION',I2/                                     1NON1034
     2        1H0,28X,11HMEASUREMENT,6X,10HNUMBER OF ,                  1NON1035
     3        5HWEIGHTED,8X,8HWEIGHTED/1H ,41X,4HTYPE,16X,9HRESIDUALS,  1NON1036
     4        18X,3HRMS/)                                               1NON1037
20120 FORMAT(1H ,40X,A6,15X,I5,15X,F10.3)                               1NON1038
20130 FORMAT(1H ,43X,11HRMS FOR ALL,I5,22H WEIGHTED MEASUREMENTS,F10.3) 1NON1039
44470 FORMAT(1H0,7X,44HGEOPOTENTIAL RESONANCE COEFFICIENTS ADJUSTED/1H0,1NON1040
     1        11HCOEFFICIENT,10X,8HA PRIORI,16X,8HADJUSTED/19X,5HVALUE, 1NON1041
     2        3X,14HRATIO TO SIGMA,3X,5HVALUE,4X,5HSIGMA/)              1NON1042
44475 FORMAT(2X,A1,1H(,I2,1H,,I2,1H),4X,3(2X,E10.4),1X,F10.4)           1NON1043
44480 FORMAT(1H1,23X,3HARC,I3,44H SATELLITE KEPLERIAN EPHEMERIS FOR ITER1NON1044
     .ATION,I3,25X,4HPAGE,I3//8X,4HTIME,17X,1HA,14X,1HE,15X,1HI,11X,    1NON1045
     .11HRA ASC NODE,5X,11HARG PERIGEE,6X,12HMEAN ANOMALY/1X,           1NON1046
     .19HYYMMDD HHMM SS.SSSS,6X,8H(METERS),17X,4(2X,15HDEG MM SS.SSSSS))1NON1047
44485 FORMAT(1X,I6,I5,F8.4,F16.4,F15.11,4(2X,2I3,F9.5))                 1NON1048
44490 FORMAT(20X,F16.4,F15.11,4(2X2I3,F9.5))                            1NON1049
44500 FORMAT('ARC',I3,' ADJUSTED PARAMETERS FOR INNER',I3,' OF OUTER',I21NON1050
     .   /I6/I6,I4,F7.4/(3D24.16,5HINNER))                              1NON1051
44510 FORMAT(A6,I1,3X,2F15.8,2D15.6)                                    1NON1052
44530 FORMAT(3HSAT,3X,I1,7X,2F15.5)                                     1NON1053
45000 FORMAT('0********** NEGATIVE ARGUMENT TO DSQRT FOR ELEMENT',      1NON1054
     .   I2,' ***********')                                             1NON1055
45010 FORMAT('0**********$ NEGATIVE ARGUMENT TO DSQRT FOR ',A6,'$******')1NON1056
45030 FORMAT('0**********$ NEGATIVE ARGUMENT TO DSQRT FOR BIAS $******')1NON1057
46000 FORMAT(1X,12D11.4)                                                1NON1058
46500 FORMAT(1H1,20X,'THE ARC',I3,' RMS FOR INNER ITERATION',I3,        1NON1059
     .   ' OUTER ITERATION',I3/16X,'CONVERGED WITHIN',2F5.1,            1NON1060
     .   ' PERCENT OF THE RMS FOR THE PREVIOUS ITERATION OF THIS ARC.') 1NON1061
47000 FORMAT(1H+,77X,24HTRANSPONDER RELAY BY SAT,I2)                    1NON1062
47100 FORMAT(1H+,78X,8HBASELINE,2X,A7,1H/,A7,7H(ELEV =,F5.2,1H))        1NON1063
47200 FORMAT(1H+,77X,I1,15H WAY TRANSMITTER,2X,A7,7H(ELEV =,F6.2,1H))   1NON1064
49000 FORMAT(1H1)                                                       1NON1065
      END                                                               1NON1065
```

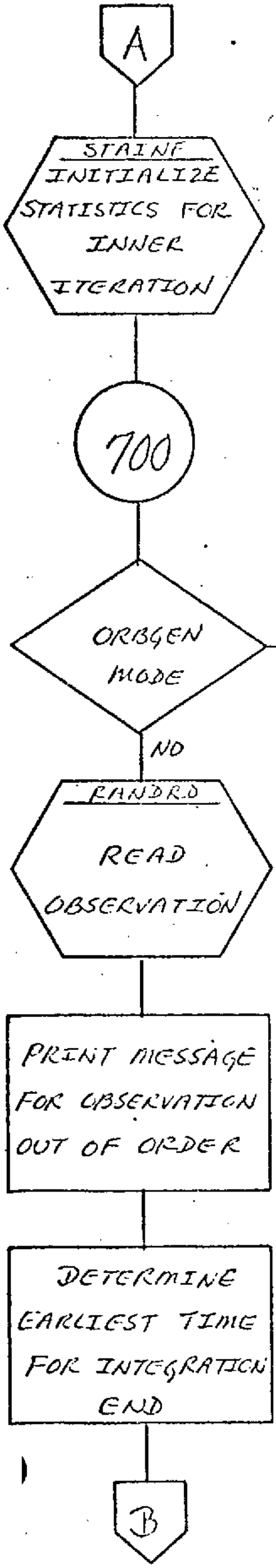


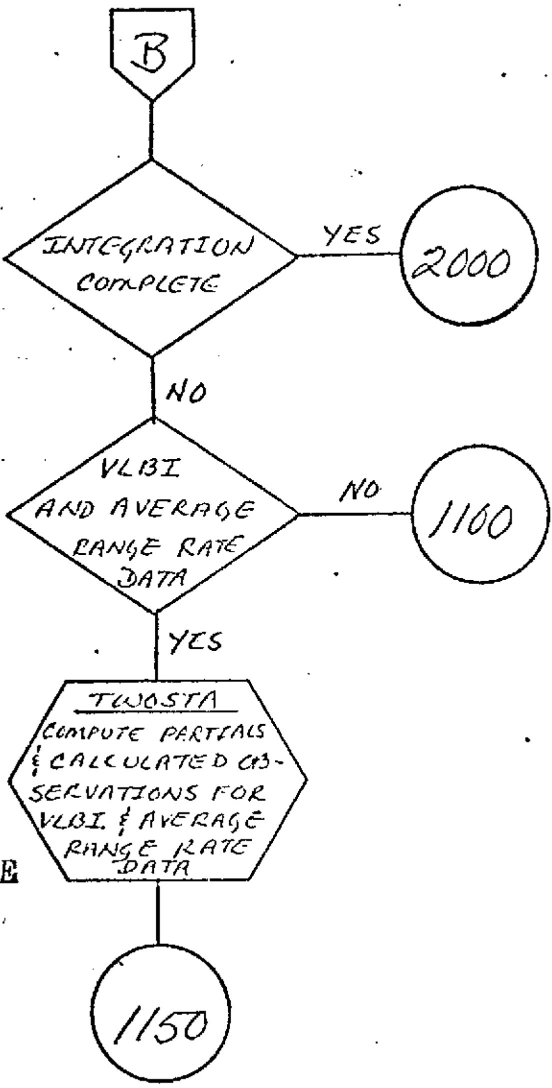


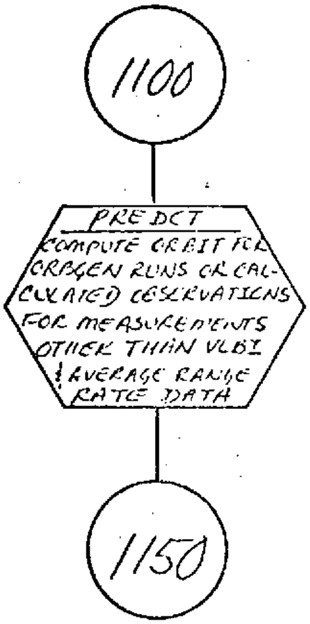
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

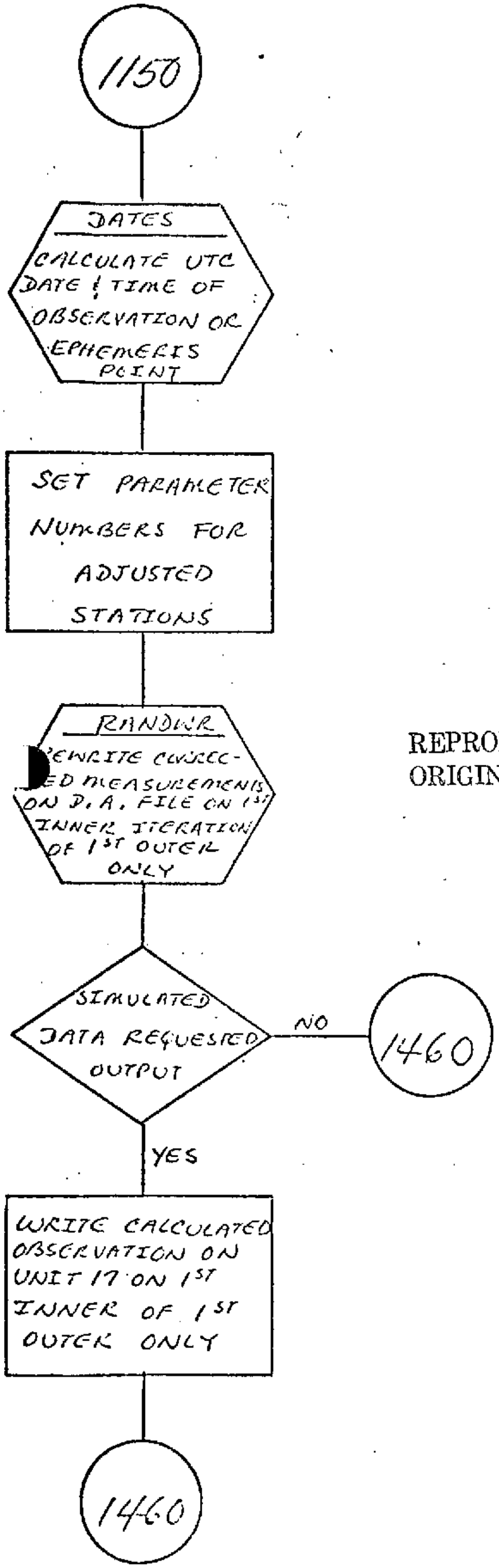

650
ORBGEN MODE
YES
660
NO
SET SWITCHES FOR DATA REDUCTION
ESTIM
INITIALIZE LEAST SQUARES ESTIMATOR
670


660
SET START TIME FOR ORBIT GENERATOR
670
INITIALIZE INTEGRATOR STEP SIZES FOR NEW INNER ITERATION
SET COUNTERS AND LOGICAL SWITCHES FOR NEW INNER ITERATION
SET NUMBER OF FORCE MODEL EQUATIONS TO BE INTEGRATED
SET EDIT CRITERION AND START TIME
A

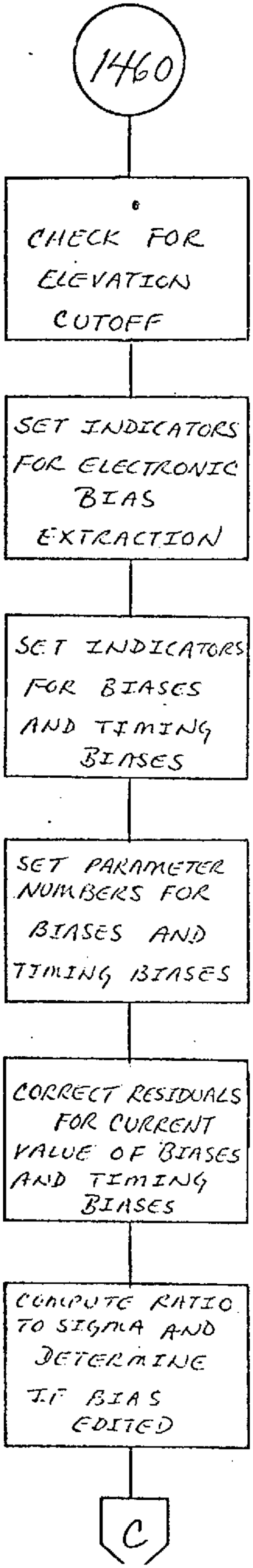

A
STAINF
INITIALIZE STATISTICS FOR INNER ITERATION
700
ORBGEN MODE
YES
930
NO
RANDRD
READ OBSERVATION
PRINT MESSAGE FOR OBSERVATION OUT OF ORDER
DETERMINE EARLIEST TIME FOR INTEGRATION END
B

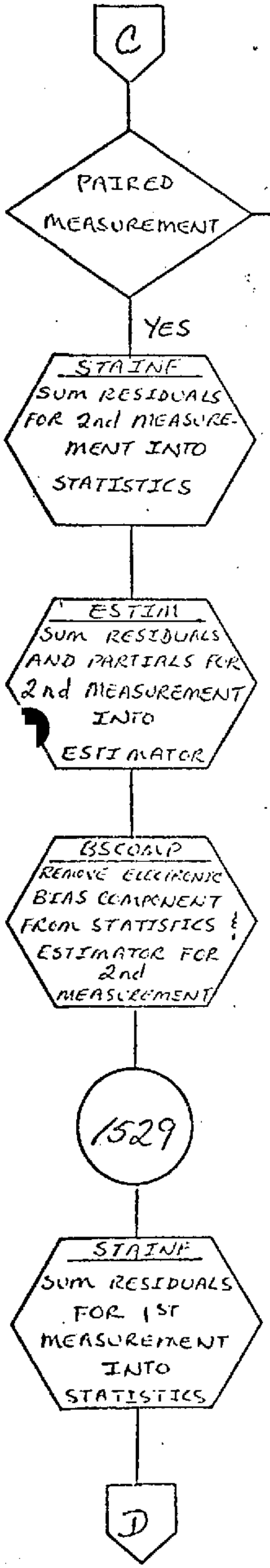

B
INTEGRATION COMPLETE
YES
2000
NO
VLBI AND AVERAGE RANGE RATE DATA
NO
1100
YES
TWOSTA
COMPUTE PARTIALS & CALCULATED OBSERVATIONS FOR VLBI & AVERAGE RANGE RATE DATA
1150

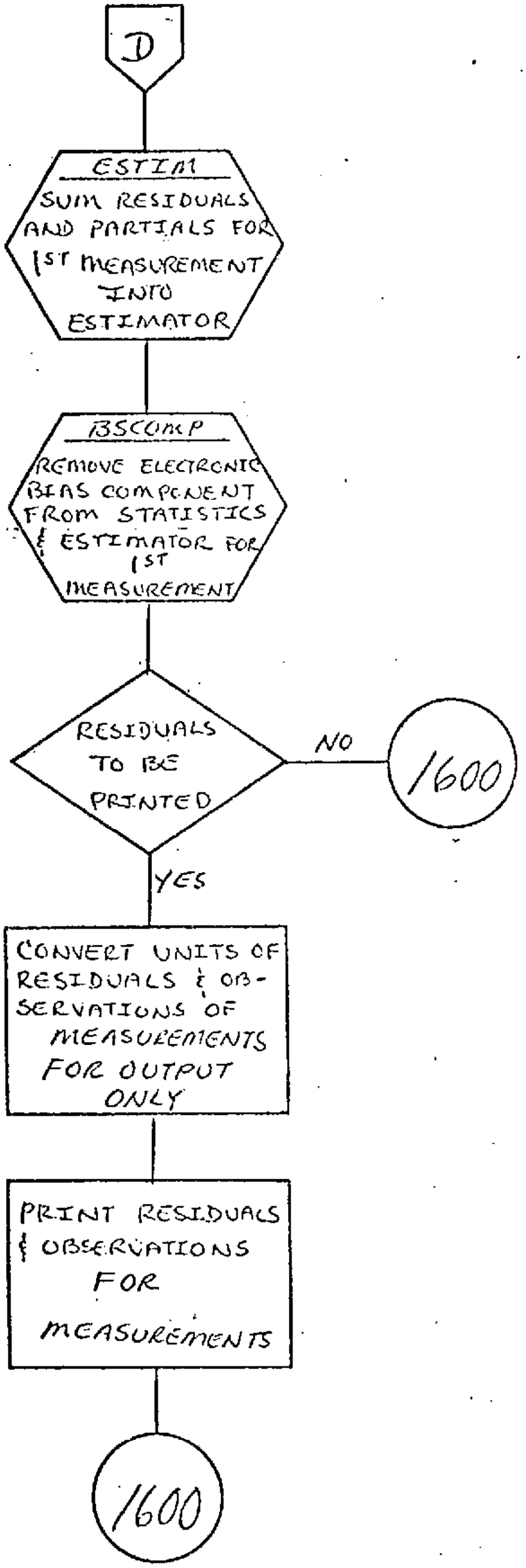

1100
PREDCT
COMPUTE ORBIT FOR ORBGEN RUNS OR CALCULATED OBSERVATIONS FOR MEASUREMENTS OTHER THAN VLBI & AVERAGE RANGE RATE DATA
1150

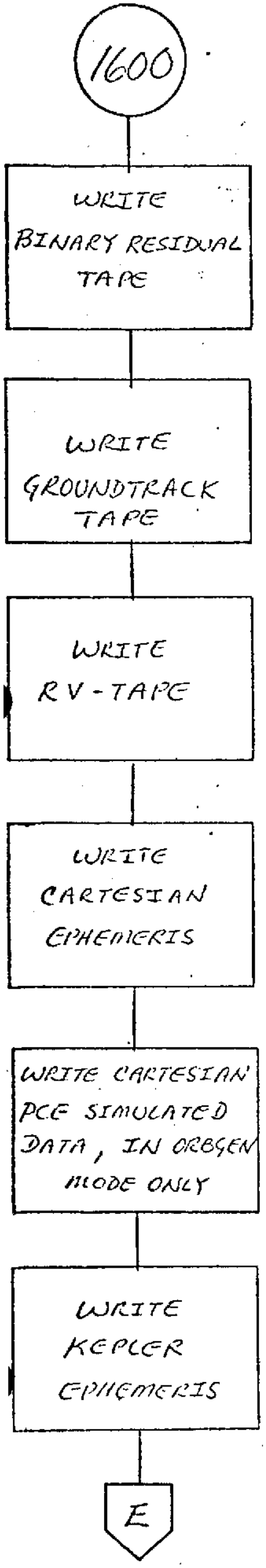


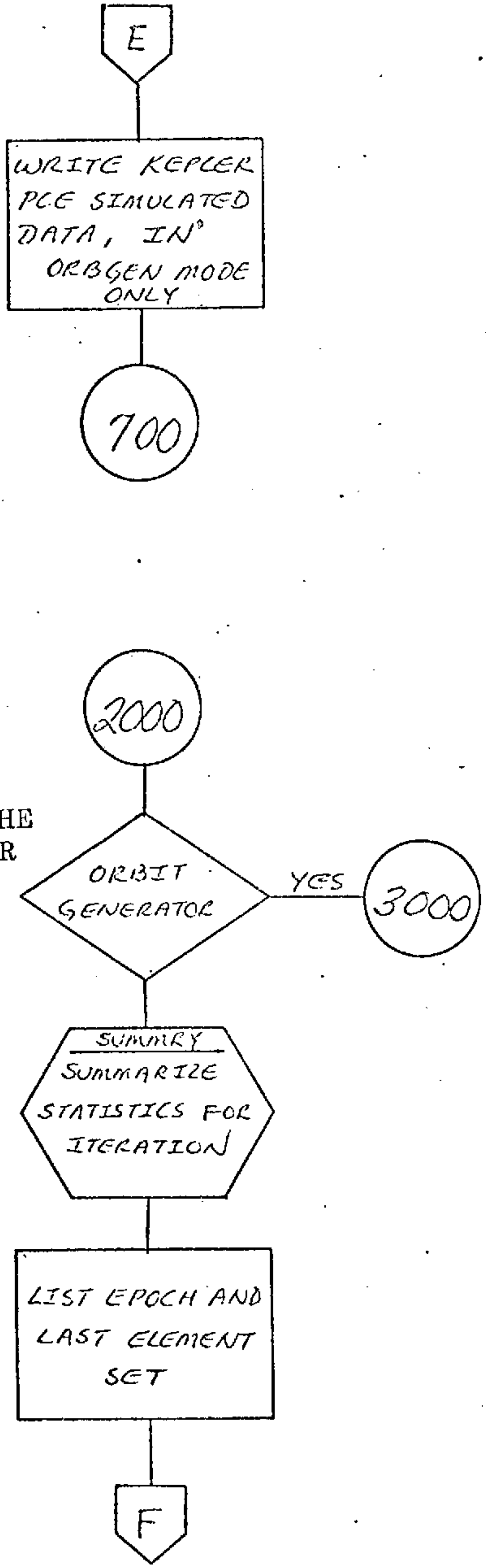
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR



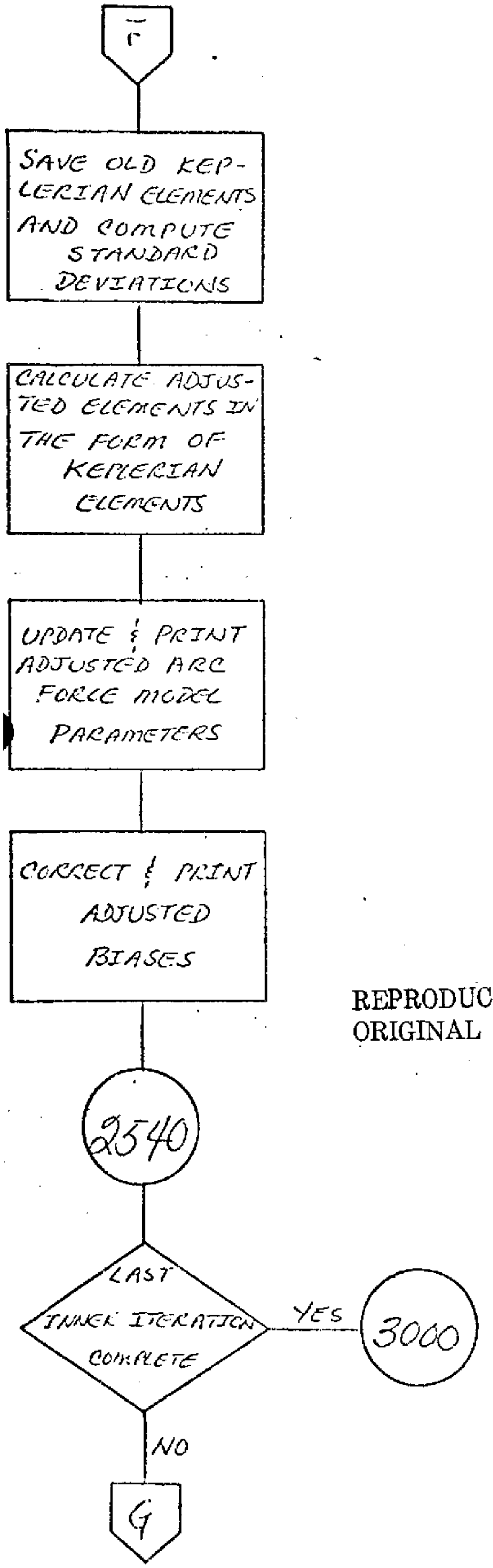

C
PAIRED MEASUREMENT
NO
1529
YES
STAINF
SUM RESIDUALS FOR 2nd MEASUREMENT INTO STATISTICS
ESTIM
SUM RESIDUALS AND PARTIALS FOR 2nd MEASUREMENT INTO ESTIMATOR
BSCOMP
REMOVE ELECTRONIC BIAS COMPONENT FROM STATISTICS & ESTIMATOR FOR 2nd MEASUREMENT
1529
STAINF
SUM RESIDUALS FOR 1ST MEASUREMENT INTO STATISTICS
D


D
ESTIM
SUM RESIDUALS AND PARTIALS FOR 1ST MEASUREMENT INTO ESTIMATOR
BSCOMP
REMOVE ELECTRONIC BIAS COMPONENT FROM STATISTICS & ESTIMATOR FOR 1ST MEASUREMENT
RESIDUALS TO BE PRINTED
NO
1600
YES
CONVERT UNITS OF RESIDUALS & OBSERVATIONS OF MEASUREMENTS FOR OUTPUT ONLY
PRINT RESIDUALS & OBSERVATIONS FOR MEASUREMENTS
1600

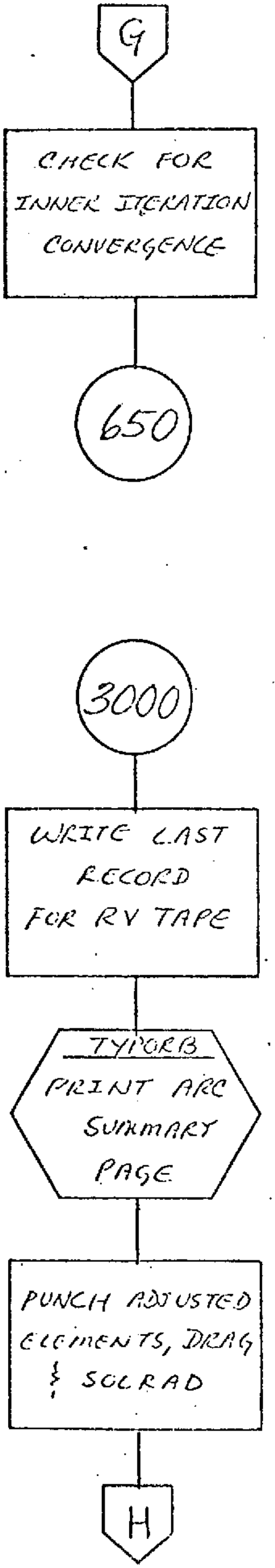


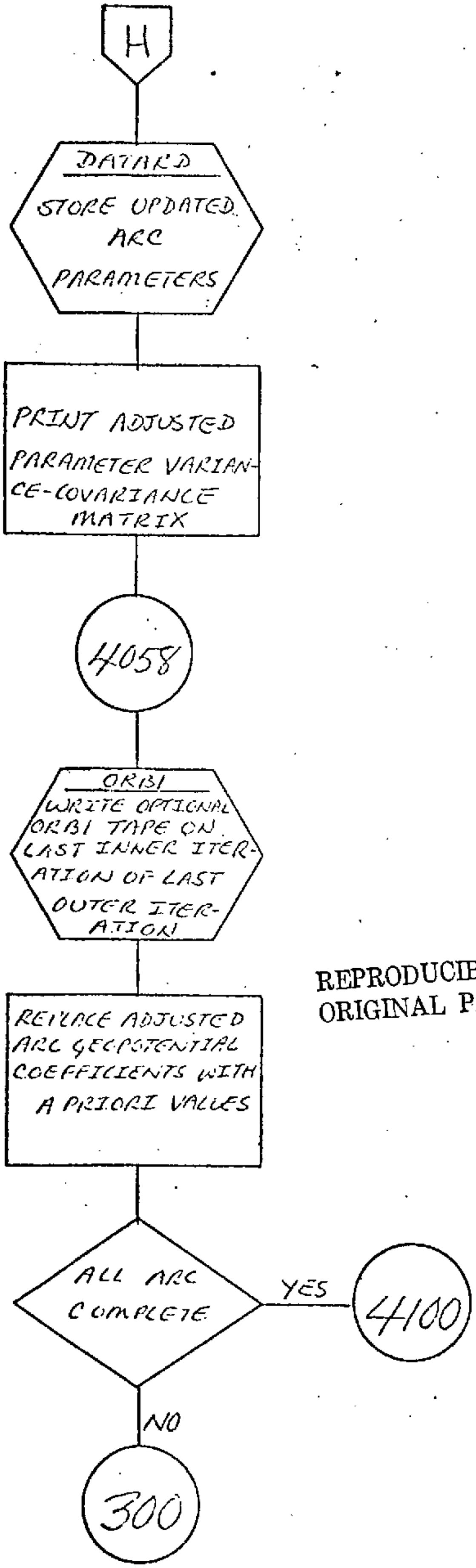


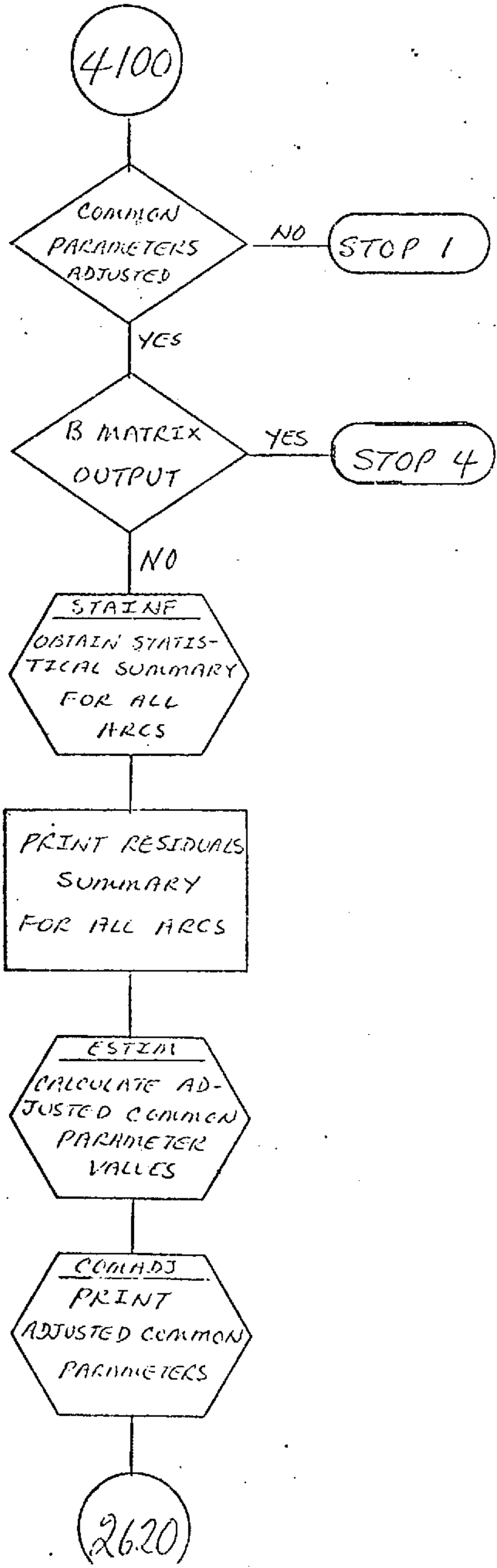
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

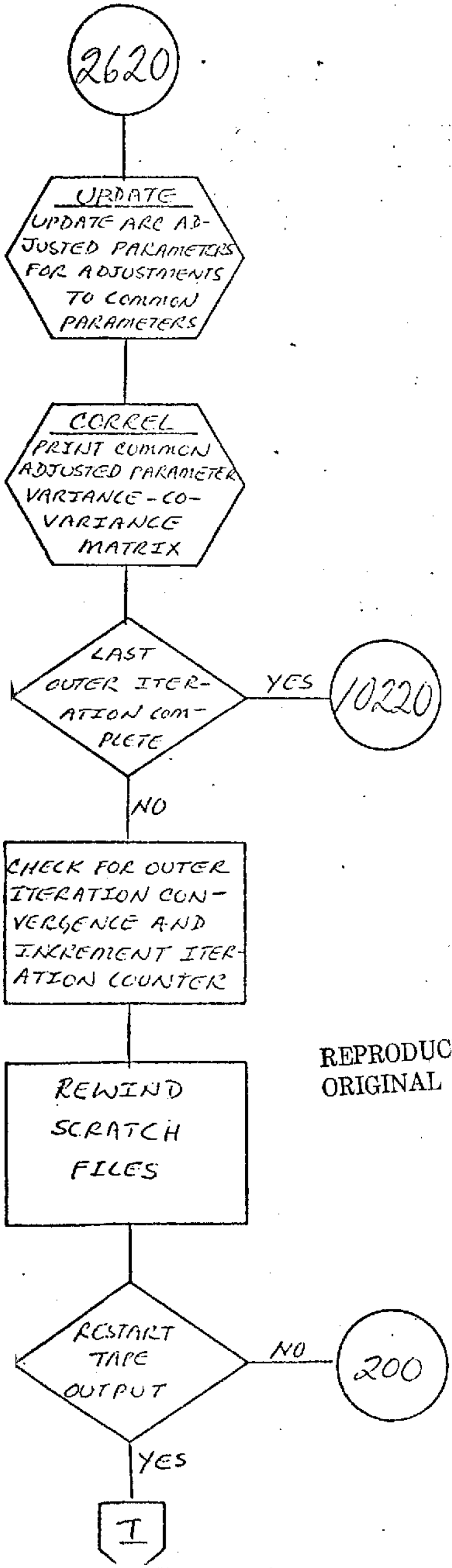


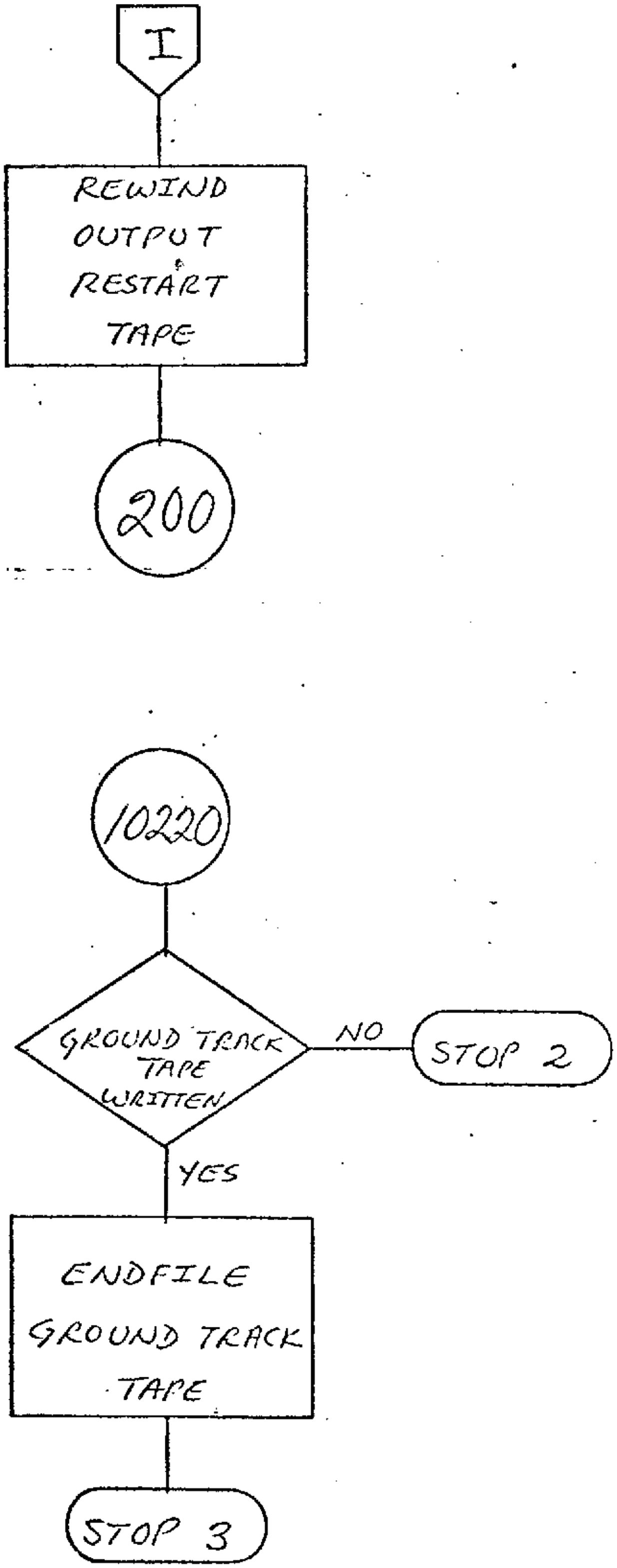
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR




H
DATARD
STORE UPDATED ARC PARAMETERS
PRINT ADJUSTED PARAMETER VARIANCE-COVARIANCE MATRIX
4058
ORBI
WRITE OPTIONAL ORBI TAPE ON LAST INNER ITERATION OF LAST OUTER ITERATION
REPLACE ADJUSTED ARC GEOPOTENTIAL COEFFICIENTS WITH A PRIORI VALUES
ALL ARC COMPLETE
YES
4100
NO
300


4100
COMMON PARAMETERS ADJUSTED
NO
STOP 1
YES
B MATRIX OUTPUT
YES
STOP 4
NO
STAINF
OBTAIN STATISTICAL SUMMARY FOR ALL ARCS
PRINT RESIDUALS SUMMARY FOR ALL ARCS
ESTIM
CALCULATE ADJUSTED COMMON PARAMETER VALUES
COMADJ
PRINT ADJUSTED COMMON PARAMETERS
2620





REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME	ADDYMD
PURPOSE	TO ADD OR SUBTRACT DAYS FROM A DATE IN THE FORM YYMMDD AND TO PROVIDE THE USER WITH THE NEW DATE
CALLING SEQUENCE	CALL ADDYMD(IYMD,IDAY)

SYMBOL	TYPE	DESCRIPTION
IYMD	I	INPUT AND OUTPUT - SIX DIGIT DATE IN THE FORM YYMMDD
IDAY	I	INPUT - NUMBER OF DAYS TO BE ADDED OR SUBTRACTED FROM INPUT TAPE

SUBROUTINES USED	NONE
COMMON BLOCKS	MONTHS
INPUT FILES	NONE
OUTPUT FILES	NONE
RESTRICTIONS	CANNOT PROCESS MULTIPLE CENTURIES

```
      SUBROUTINE ADDYMD(IYMD,IDAY)
      COMMON/MONTHS/MONTH(13,2)
      ISUB(IY)=MIN0(MOD(IY,4),1)+1
C SEPARATE YEAR/MONTH/DAY
      IY = IYMD/10000
      IK = IY*10000
      IM = (IYMD-IK)/100
      ID = IYMD-IK-IM*100
      LY=ISUB(IY)
C COMPUTE ELAPSED DAYS FROM JANUARY 0.0 OF CENTURY
      ID=(IY-1)*36525/100+MONTH(IM,LY)+ID+IDAY
C COMPUTE NEW YEAR/MONTH/DAY
      IY=(ID-1)*100/36525+1
      ID=ID-36525*(IY-1)/100
      LY=ISUB(IY)
      IF(LY.EQ.1.OR.ID.LT.366) GO TO 5
      IY=IY+1
      ID=ID-365
      LY=ISUB(IY)
    5 DO 10 I=1,12
      IF(ID.LE.MONTH(I+1,LY)) GO TO 20
   10 CONTINUE
C PACK NEW YYMMDD
 20   IYMD=IY*10000+I*100+ID-MONTH(I,LY)

      RETURN
      END
```

ADFLUX

Flux values are added to the tables in the location corresponding to the number of days between the date of the flux values and the day prior to the first date for flux data in the table. The stop date for flux data is reset to reflect the presence of the latest available data in the table.

For dates within the range of the table but for which data is not available a linear interpolation between the two non-zero points adjacent to the ends of the missing data span is used to compute values to fill in the table.

The logical function RFTMCD is used to determine when a Reference Time card is encountered signifying the beginning of a new arc.

The function JANTHG is used to calculate the Greenwich mean sideral time for Jan 0.0 of the reference year and to load flux data into COMMON FLXBLK.

A flux file is written containing the information obtained from JANTHG for each arc.

NAME ADFLUX

PURPOSE
1) TO ADD SOLAR AND MAGNETIC FLUX TO STORED TABLES
2) TO COUNT THE NUMBER OF ARCS IN RUN
3) TO WRITE ON SCRATCH FILE THE GREENWICH HOUR ANGLE OF JAN 0.0 OF THE REFERENCE YEAR AND THE FLUX DATA NEEDED FOR EACH ARC

CALLING SEQUENCE CALL ADFLUX(NARCS,MGFLUX,NFLUX,SFLUX,MYMD,LYMD)

SYMBOL	TYPE	DESCRIPTION
NARCS	I	OUTPUT - NUMBER OF ARCS IN RUN
MGFLUX	I	INPUT & OUTPUT - MAGNETIC FLUX DATA
SFLUX	R	INPUT & OUTPUT - SOLAR FLUX DATA
MYMD	I	INPUT - START DATE FOR FLUX DATA
LYMD	I	INPUT & OUTPUT - STOP DATE FOR FLUX DATA

SUBROUTINE USED ALIST AND2 DIFF ERROR JANTHG RETMOD

COMMON BLOCKS FLXBLK INTBLK TPEBLK

CONSTANTS USED MASK

SYMBOL	TYPE	DESCRIPTION
MASK	I*2	HEX NUMBER USED TO EXTRACT DECIMAL NUMBERS FROM THEIR EBCDIC CODE

INPUT FILES INPT - INPUT CARDS

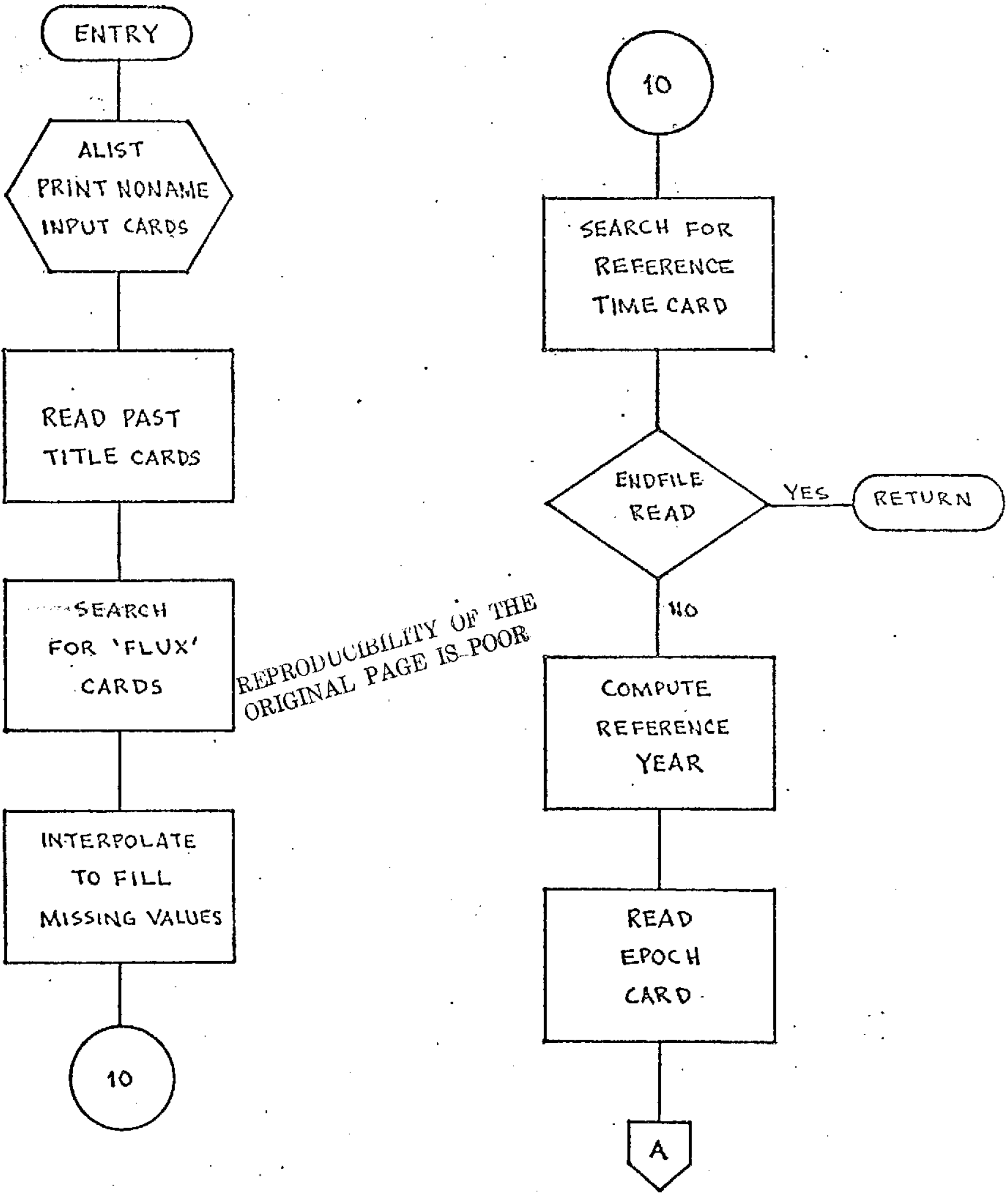
OUTPUT FILES FLTP - FLUX DATA FILE

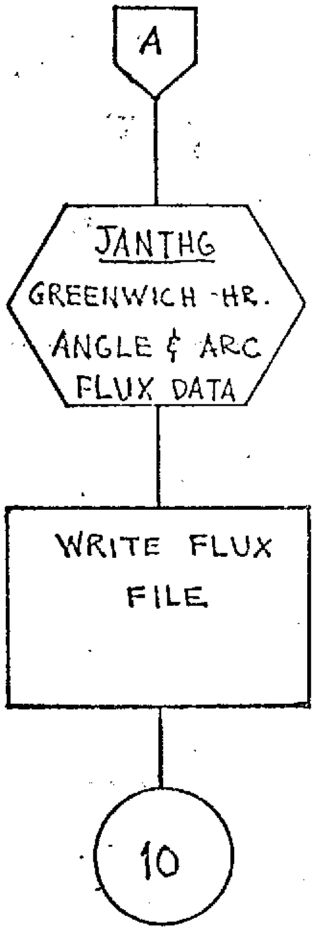
RESTRICTIONS MUST BE CALLED BEFORE FIRST READ

```
      SUBROUTINE ADFLUX(NARCS,MGFLUX,NFLUX,SFLUX,MYMD,LYMD)             ADFL  45
      DOUBLE PRECISION DT1,DATE,DATAN,FLUXS,VALUE1,VALUE2,JANTHG,THETGO ADFL  46
      DOUBLE PRECISION FLUX1,FLUX2,FLUX3,FLUX4,FLUX5                    ADFL  47
      LOGICAL LAST,RETMOD                                               ADFL  48
      INTEGER FLTP,AND2                                                 ADFL  49
      INTEGER*2 A(80),SFLUX(NFLUX),MGFLUX(NFLUX)                        ADFL  50
      COMMON/FLXBLK/FLUX1(405),FLUX2(405),FLUX3(405),FLUX4(405),        ADFL  51
     *   FLUX5(405)                                                     ADFL  52
      COMMON/INTBLK/THDCT1(54),THETGO,MBODY(76)                         ADFL  53
      COMMON/TPEBLK/INTP,IOBS(9),FLTP,GEOTP                             ADFL  54
      INTEGER*2 MASK/Z00F/                                              ADFL  55
```

```
      DATA FLUXS,DATAN/6HFLUX  ,6HDATA  /
C PRINT INPUT CARDS
      CALL ALIST
C READ PAST TITLE CARDS
      READ(INTP,10000)
      JDATE=LYMD
C SEARCH FOR 'FLUX' CARDS
    1 READ(INTP,50000)DT1,DATE,VALUE1,VALUE2
      IF(DT1.EQ.DATAN) GO TO 2
      IF(DT1.NE.FLUXS) GO TO 1
      IDATE=DATE+.1D0
      CALL DIFF(MYMD,0,IDATE,1,IDAY,ISEC)
      IF(IDAY.GT.0.AND.IDAY.LT.NFLUX) GO TO 3
      PRINT 40000,DATE
      CALL ERROR(10,DT1)
      GO TO 1
C ADD FLUX TO TABLE
    3 IF(VALUE1.GT.0.0D0) SFLUX(IDAY+1)=VALUE1*1.0D+1+0.1D0
      IF(VALUE2.GT.0.0D0) MGFLUX(IDAY+1)=VALUE2*1.0D+1+0.1D0
      IF(IDATE.GT.LYMD) LYMD=IDATE
      GO TO 1
    2 IF(JDATE.EQ.LYMD) GO TO 10
      CALL DIFF(MYMD,0,LYMD,1,JDAY,JSEC)
C INTERPOLATE TO FILL MISSING FLUX VALUES UP TO THE MOST RECENT TABLE
C VALUE
      DO 9 LL=1,2
      LAST=.FALSE.
      AM=0.
      CALL DIFF(MYMD,0,JDATE,1,IDAY,ISEC)
      IF(LL.EQ.1) BB=SFLUX(IDAY+1)
      IF(LL.EQ.2) BB=MGFLUX(IDAY+1)
      KDAY=IDAY
    4 IDAY=KDAY
    5 KDAY=KDAY+1
      IF(KDAY.GT.JDAY+1) GO TO 6
      IF(LL.EQ.1.AND.SFLUX(KDAY).LE.0) GO TO 5
      IF(LL.EQ.2.AND.MGFLUX(KDAY).LE.0)  GO TO 5
      IF(KDAY.EQ.JDAY+1) LAST=.TRUE.
      X=IDAY
      IF(LL.EQ.1) Y=SFLUX(IDAY)
      IF(LL.EQ.2) Y=MGFLUX(IDAY)
      X1=KDAY
      IF(LL.EQ.1)Y1=SFLUX(KDAY)
      IF(LL.EQ.2)Y1=MGFLUX(KDAY)
      AM=(Y1-Y)/(X1-X)
      BB=Y1-AM*X1
      GO TO 7
    6 KDAY=KDAY-1
      LAST=.TRUE.
    7 ID=IDAY+1
      DO 8 I=ID,KDAY
      X=I
      IF(LL.EQ.2) MGFLUX(I)=AM*X+BB+.5
      IF(LL.EQ.1) SFLUX(I)=AM*X+BB+.5
    8 CONTINUE
      IF(.NOT.LAST) GO TO 4
```

```
    9   CONTINUE
C SEARCH FOR REFERENCE TIME CARD
   10 READ(INTP,20000,END=20) A
      IF(.NOT.REFTME(A)) GO TO 10
C COMPUTE REFERENCE YEAR
      IY=10*(AND2(A(1),MASK)/255)+AND2(A(2),MASK)/256
C READ EPOCH CARD
      READ(INTP,30000) IYMD,IYMDD
      IF (IYMDD.GT.0) IYMD=IYMDD
      NARCS=NARCS+1
C DETERMINE GREENWICH HOUR ANGLE OF REFERENCE YEAR IN RADIANS AND
C SELECT FLUX VALUES FOR ARC
      THETGO=JANTHG(IYMD,IY,SFLUX,MGFLUX,MYMD,LYMD)
C WRITE FLUX FILE
      WRITE(FLTP) THETGO,FLUX1
      WRITE(FLTP) FLUX2
      WRITE(FLTP) FLUX3
      WRITE(FLTP) FLUX4
      WRITE(FLTP) FLUX5
      GO TO 10
   20 ENDFILE FLTP
      REWIND FLTP
      REWIND INTP
      RETURN
10000 FORMAT(1X//)
20000 FORMAT(80A1)
30000 FORMAT(I6,34X,I6)
40000 FORMAT(1H1,20X,'DATE',D14.7,' ON FLUX CARD OUT OF RANGE OF ',
     .     'TABLES.'/1H0,15X,'CARD IGNORED. EXECUTION CONTINUING.'/)
50000 FORMAT(A6,4X,3D15.8)
      END
```


ENTRY
ALIST
PRINT NONAME
INPUT CARDS
READ PAST
TITLE CARDS
SEARCH
FOR 'FLUX'
CARDS
INTERPOLATE
TO FILL
MISSING VALUES
10
10
SEARCH FOR
REFERENCE
TIME CARD
ENDFILE
READ
YES
RETURN
NO
COMPUTE
REFERENCE
YEAR
READ
EPOCH
CARD
A



REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME	ALIST
PURPOSE	TO LIST GEODYN INPUT CARDS
CALLING SEQUENCE	CALL ALIST
SUBROUTINES USED	NONE
COMMON BLOCKS	TPEBLK
INPUT FILES	INTP - INPUT CARDS
OUTPUT FILES	IOUTP - PRINTER

```
      SUBROUTINE ALIST
      LOGICAL*4 LRECRD(20), END
      INTEGER*4 ICASE,SELECT/4HSELE/,DELETE/4HDELE/,ENDALL/4HENDA/
      EQUIVALENCE(ICASE,LRECRD(1))
      COMMON/TPEBLK/INTP,IOUTP,ITAPES(10)
      DATA IPAGE/1/,LINE/1/,END/.FALSE./,IDATA/4HDATA/
C LIST GEODYN INPUT CARDS
      WRITE(IOUTP,1000)
      WRITE(IOUTP,2000)
      WRITE(IOUTP,3000)
    1 WRITE(IOUTP,101) IPAGE
      IPAGE=IPAGE+1
      I=0
      IF(END) GO TO 20
    5 READ(INTP,100,END=99) LRECRD
    7 I=I+1
      WRITE(IOUTP,102) LRECRD,LINE
      LINE=LINE+1
      IF(I.EQ.56) GO TO 1
      IF(ICASE.EQ.IDATA) END=.TRUE.
      IF(.NOT.END) GO TO 5
   20 READ(INTP,100,END=99) LRECRD
      IF(ICASE.EQ.SELECT.OR.ICASE.EQ.DELETE.OR.ICASE.EQ.ENDALL) GO TO 7
      END=.FALSE.
      IF(I.LE.50) GO TO 40
      WRITE(IOUTP,101) IPAGE
      IPAGE=IPAGE+1
      I=0
      GO TO 7
   40 WRITE(IOUTP,103)
      I=I+4
      GO TO 7
   99 REWIND INTP
      RETURN
  100 FORMAT(20A4)
  101 FORMAT('1LISTING OF MULTI-ARC GEODYN SET-UP',33X,4HPAGE,I3///)
  102 FORMAT(1X,20A4,10X,I4)
  103 FORMAT(///)
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
 1000 FORMAT(1H2,20X,5(1H*)/20X,7(1H*)/19X,9(1H*)/13X,4H****,2X,4H****/
     .      15X,3H***,3X,4H****/17X,4H****,3X,4H****/17X,4H****,3X,4H****
     .      /17X,4H****,3X,4H****,6X,3H***/17X,4H****,3X,4H****,4X,
     .      5(1H*),53X,5(1H*)/17X,4H****,2X,4H****,3X,6(1H*),54X,5(1H*)/
     .      13X,4H****,1X,4H****,1X,8(1H*),53X,6(1H*)/19X,12(1H*),1X,
     .     4H****,53X,4H****/20X,9(1H*),3X,4H****,52X,4H****/21X,5(1H*),
     .      6X,4H****,52X,4H****/20X,4H****,8X,4H****,51X,4H****/19X,
     .      5(1H*),8X,4H****,51X,4H****/18X,5(1H*),9X,4H****,50X,4H****/
     .      17X,5(1H*),10X,4H****,50X,4H****/16X,5(1H*),11X,4H****,49X,
     .      4H****/15X,5(1H*),12X,4H****,49X,4H****/14X,5(1H*),13X,
     .      4H****,48X,4H****/13X,5(1H*),14X,4H****,6X,8(1H*),8X,7(1H*),
     .      11X,7(1H*),1X,4H****,4X,7(1H*),7X,5(1H*),4X,6(1H*),2X,
     .      7(1H*))
 2000 FORMAT(12X,5(1H*),15X,4H****,5X,10(1H*),5X,10(1H*),8X,13(1H*),
     .      5X,7(1H*),7X,5(1H*),4X,6(1H*),1X,9(1H*)/11X,5(1H*),16X,
     .      4H****,4X,11(1H*),4X,12(1H*),5X,13(1H*),5X,7(1H*),7X,4H****,
     .      5X,17(1H*)/2X,5(1H*),2X,5(1H*),17X,4H****,3X,4H****,6X,2H**,
     .      4X,4H****,5X,3H***,6X,4H****,5X,4H****,9X,4H****,6X,4H****,
     .      6X,6(1H*),5X,4H****/ 3X,5(1H*),1X,5(1H*),17X,4H****,4X,
     .      4H****,11X,4H****,6X,3H***,5X,4H****,6X,4H****,5X,4H****,
     .      6X,4H****,8X,5(1H*),6X,4H****/4X,9(1H*),17X,4H****,6X,
     .      5(1H*),5X,3H***,7X,3H***,5X,3H*** ,7X,4H****,8X,4H****,5X,
     .      4H****,8X,4H****,7X,4H****/5X,7(1H*),17X,4H****,8X,6(1H*),
     .      6X,4H****,7X,3H***,4X,4H****,7X,4H****,7X,4H****,7X,4H****,
     .      6X,4H****,7X,4H****/6X,5(1H*),17X,4H****,7X,4H****,10X,3H***,
     .      8X,3H***,4X,3H***,7X,
     .     4H****,7X,4H****,7X,4H****,8X,4H****,7X,4H****/6X,6(1H*),
     .      15X,4H****,7X,4H****,10X,4H****,7X,4H****,3X,4H****,6X,
     .      5(1H*),7X,4H****,6X,5(1H*),2X,4H****,7X,4H****/5X,8(1H*),
     .      13X,4H****,8X,4H****,10X,4H****,6X,4H****,4X,4H****,5X,
     .      6(1H*),6X,4H****,6X,5(1H*),3X,4H****,7X,4H****)
 3000 FORMAT(4X,4H****,2X,19(1H*),2X,4H****,4X,4H****,3X,4H****,5X,
     .      4H****,5X,4H****,4X,7(1H*),4X,4H****,5X,6(1H*),5X,4H****,
     .      7X,4H****/3X,5(1H*),4X,15(1H*),5X,11(1H*),4X,12(1H*),6X,
     .      11(1H*),1X,4H****,6X,9(1H*),1X,3H***,8X,4H****,7X,6(1H*)/
     .      2X,6(1H*),5X,12(1H*),12X,9(1H*),6X,10(1H*),3X,9(1H*),2X,
     .      4H****,7X,7(1H*),2X,3H***,7X,5(1H*),5X,7(1H*)/101X,4H****/
     .      101X,4H****/100X,4H****/100X,4H****/99X,4H****/99X,4H****/
     .    98X,4H****/3X,25(1H*),43(1H*),30(1H*)/2X,99(1H*)/1X,99(1H*))
      END
```

```
NAME            ALPMRC

PURPOSE         BLOCK DATA STORAGE OF ALPHA-NUMERIC INFORMATION
                USED IN GEODYN PRINT FORMATS

COMMON BLOCKS   ALPMRC
```

```
      BLOCK DATA                                                         ALPM  11
      IMPLICIT REAL*8 (A-Z)                                              ALPM  12
      LOGICAL HYPER                                                      ALPM  13
      COMMON/ALPMRC/ITNMS(5),TIMING,BLANK,ATYPE(31),UNITS(15),ELCUT,     ALPM  14
     .  HYPER                                                            ALPM  15
      DATA BLANK/6H      /                                               ALPM  16
      DATA TIMING/6HTIMING/                                              ALPM  17
      DATA UNITS/6HSECOND,6HRADIAN,6HMETERS,5HM/SEC,6HMETERS,1H ,        ALPM  18
     .   3*6HRADIAN,2*1H ,5HM/SEC,1H ,2*6HRADIAN/                        ALPM  19
      DATA ATYPE/6HRT ASC,6HRANGE ,6HR RATE,6HHEIGHT,6HALPHA ,6HX ANGL,  ALPM  20
     .   6HAZMUTH,6HDECLIN,2*1H ,6HH RATE,6HBETA  ,6HY ANGL,6HELEV  ,    ALPM  21
     .   3H  X,3H  Y,3H  Z,5H XDOT,5H YDOT,5H ZDOT,                      ALPM  22
     .   3H  A,3H  E,5H INCL,5H NODE,5H PERG,5H MEAN,                    ALPM  23
     .   6HT DLAY,6HPRG RT,6HAV2 RR,6HAV3 RR,4H ALL/                     ALPM  24
      DATA ITNMS/6HFIRST ,6H LAST ,6HITERAT,3HION,4H AND/                ALPM  25
      END                                                                ALPM  26
```

NAME AND2

PURPOSE TWO BYTE INTEGER 'AND' FUNCTION

CALLING SEQUENCE I=AND2(I1,I2)

SYMBOL	TYPE	DESCRIPTION
I1	I*2	INPUT - FIRST INTEGER
I2	I*2	INPUT - SECOND INTEGER
AND2	I	OUTPUT - AND OF I1 & I2

SUBROUTINES USED NONE

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

```
      INTEGER FUNCTION AND2(I1,I2)                                      AND2  26
      EQUIVALENCE (IAND,RAND)                                           AND2  27
      INTEGER*2 I1,I2                                                   AND2  28
C FIND LOGICAL AND OF THE ARGUMENTS                                     AND2  29
      J1=I1                                                             AND2  30
      J2=I2                                                             AND2  31
      RAND=AND(J1,J2)                                                   AND2  32
      AND2=IAND                                                         AND2  33
      RETURN                                                            AND2  34
      END                                                               AND2  35
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME APPER

PURPOSE TO COMPUTE SATELLITE APOGEE HEIGHT AND PERIGEE HEIGHT

CALLING SEQUENCE CALL APPER

SUBROUTINES USED NONE

COMMON BLOCKS APARAM CELEM CORB1 INTBLK CONSTS

INPUT FILES NONE

OUTPUT FILES NONE

REFERENCES 'GEODYN SYSTEMS DESCRIPTION'
VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE APPER
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/APARAM/INPAR(4),NSAT,NGPARC(5)
      COMMON/CELEM/ELEMST(6,2),ORBELA(6,2),XNU,EC,IRMSTO
      COMMON/CORB1/RANDOT(2),PERDOT(2),PEHT(2),APHT(2),PRD(2)
      COMMON/INTBLK/THDOT(4),AE,AESQ,FLAT,FSQ32,FFSQ32,GM3(57)
      COMMON/CONSTS/DPI,DTWOPI,DRAD,DRSEC
      DO 10 I=1,NSAT
      SPSISQ=DSIN(ORBELA(3,I)*DRAD)*DSIN(ORBELA(5,I)*DRAD)
      SPSISQ=SPSISQ**2
C CALCULATE THE EARTH RADIUS TO THE CIRCLE OF INTERSECTION OF THE
C SATELLITE SEMI-MAJOR AXIS WITH THE SPHEROID
      EARTH=AE+FSQ32*SPSISQ**2-FFSQ32*SPSISQ
      AESAT=ORBELA(1,I)*ORBELA(2,I)
C CALCULATE THE APOGEE AND PERIGEE DISTANCES
C SUBTRACT OUT THE EARTH RADIUS TO CALCULATE APOGEE AND PERIGEE.
C HEIGHTS
      PEHT(I)=(ORBELA(1,I)-AESAT-EARTH)*1.0D-3
   10 APHT(I)=(ORBELA(1,I)+AESAT-EARTH)*1.0D-3
      RETURN
      END
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME ARCPAR

PURPOSE TO LOAD INDIVIDUAL ARC PARAMETERS INTO VARIABLE STORAGE ARRAYS

CALLING SEQUENCE

```
CALL ARCPAR(BSTRT,BSEND,BIASO,BBIAS,BIASSG,BTYPE,
            BSTANO,BIASNO,GPVALO,GPSIG,GPNO,
            GPVAL,INDXCS)
```

SYMBOL	TYPE	DESCRIPTION
BSTRT (1)	DP	OUTPUT - BIAS START TIMES IN DAYS FROM REFERENCE JAN 0.0
BSEND (1)	DP	OUTPUT - BIAS STOP TIMES IN DAYS FROM REFERENCE JAN 0.0
BIASO (1)	DP	OUTPUT - A PRIORI BIAS ESTIMATES
BBIAS (1)	DP	OUTPUT - CURRENT BEST BIAS VALUES
BIASSG (1)	DP	OUTPUT - STANDARD DEVIATION OF BIASES AND DRAG AND SOLAR RADIATION COEFFICIENTS
BTYPE (1)	I*2	OUTPUT - BIAS TYPES
BSTANO (1)	I*2	OUTPUT - BIAS STATION NUMBERS
BIASNO (1)	DP	OUTPUT - BIAS INDICES
GPVALO (1)	DP	OUTPUT - A PRIORI VALUES OF ESTIMATED GEOPOTENTIAL COEFFICIENTS
GPSIG (1)	DP	OUTPUT - A PRIORI SIGMAS OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
GPNO (1)	I*2	OUTPUT - LOCATIONS OF GEOPOTENTIAL PARAMETERS IN NORMAL MATRIX
GPVAL (1)	DP	OUTPUT - CURRENT BEST VALUES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
INDXCS (3,1)	I*2	OUTPUT - INDICES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS

SUBROUTINES USED	STORE	CLEAR2	DATARD	NUMBR2
COMMON BLOCKS	APARAM	CONOUT	CONSTS	CPARAM
	EMODEL	FLXBLK	INTBLK	PRIORI
	TREBLK	VRBLK		

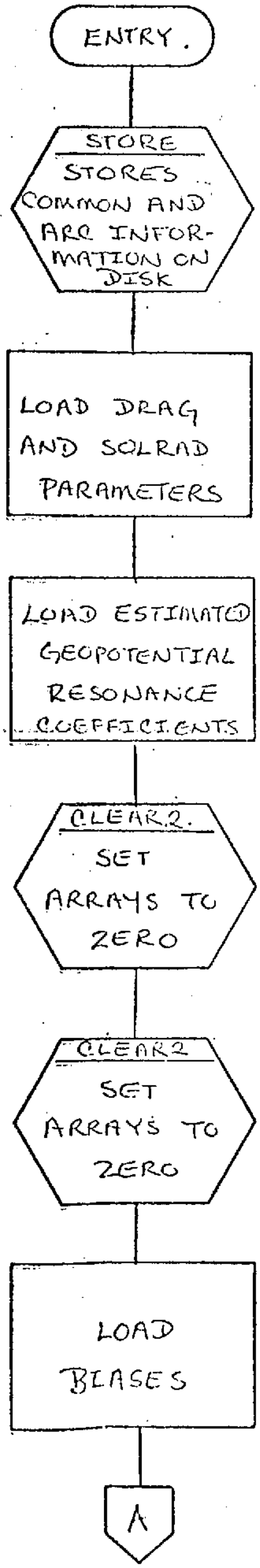
INPUT FILES NONE

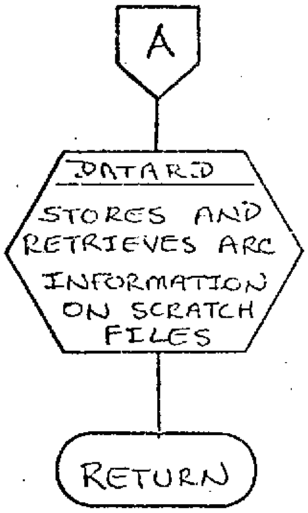
OUTPUT FILES NONE

REFERENCES 'GEODYN SYSTEMS DESCRIPTION'
VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE ARCPAR(BSTRT,BSEND,BIAS0,BBIAS,BIASSG,BTYPE,BSTANO,
     .   BIASNO,GPVAL0,GPSIG,GPNO,GPVAL,INDXCS,BESTNO,BETYPE,NEBIAS,
     .   ISTANO)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL CMPGPR,STARTR,STARTW
      INTEGER*2 BTYPE,BSTANO,BIASNO,BSTYPE,BSTNOS,GPNO,INDXCS
      INTEGER*2 BESTNO,BETYPE,ISTANO
      INTEGER ADDR,SRAD,ASAT,SCRC,FLTP,ARCNO,SCRA,ADDRD,STARTA,STARTO,
     .   OUTSTR
      REAL BIAS,BIASVR
      DIMENSION BSTRT(1),BSEND(1),BIAS0(1),BBIAS(1),BIASSG(1),
     .   BTYPE(1),BSTANO(1),BIASNO(1),GPVAL0(1),GPSIG(1),GPNO(1),
     .   GPVAL(1),INDXCS(3,1),SUM1(1),SUM2(1),BESTNO(NEBIAS),
     .   BETYPE(NEBIAS),ISTANO(1)
      COMMON/APARAM/INPAR,INPAR1,NBIAS,NSTSTA,NSAT,NGPARC,NDREC1,NPARAM,
     .   NBIASE,MAXPAR
      COMMON/CONOUT/IG1(11),NARCS,NSTARD,STARTR,STARTW,STARTA,STARTO,
     .   INSTRT,OUTSTR
      COMMON/CONSTS/PI,TWOPI,DRAD,DRSEC
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,NGPCOM,
     .   NCSEST,CMPGPR,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,INNRSW,
     .   NCONST,NOCONS
      COMMON/FMODEL/INDEXF(4),CS(30,33),MODEL(15)
      COMMON/FLXELK/BTIME1(900),BTIME2(900),BSTYPE(900)
      COMMON/INTBLK/THOOT1(99),ADDR(2,3),LOVE(8)
      COMMON/PRIORI/ELEMIN(78),DRAGSC(2,3),DRAG0(2,3),CD(2,3)
C OBTAIN STORED ARC INFORMATION
      COMMON/TREBLK/INTP(8),SCRA,SCRC,FLTP(2)
      COMMON/VBBLOK/BIAS(900),BIASVR(500),BSTNOS(900)
      EQUIVALENCE (PHASE,LOVE(5))
      DO 300 ARCNO=1,NARCS
      IF(NEBIAS.LE.0) GO TO 10
      READ(SCRC) BESTNO
C STORE DRAG AND SOLAR RADIATION COEFFICIENTS
      READ(SCRC) BETYPE
      GO TO 20
   10 READ(SCRC)
      READ(SCRC)
   20 CALL STORE(.TRUE.,.FALSE.)
      IF(NBIASE.LE.0) GO TO 80
      DO 40 I=1,NBIASE
      J=BSTNO(I)
   40 BSTNO(I)=NUMBR2(J,ISTANO,NSTA)
   80 PHASE=PHASE*DRAD
C STORE ESTIMATED GEOPOTENTIAL RESONANCE COEFFICIENTS
```

```
      DO 100 I=1,MBIAS
  100 BIASNO(I)=0
      I1=0
      L1=6*NSAT
      DO 150 L=1,NSAT
      DO 150 I=1,3
      I1=I1+1
      BIASO(I1)=CD(L,I)
      EBIAS(I1)=CD(L,I)
      BIASSG(I1)=DRAGSG(L,I)
      INDEX=ADDR(L,I)
  150 IF(INDEX.GT.0) BIASNO(I1)=L1+INDEX
      NGPARC=900-NGPARC
      IF(NGPARC.LT.1) GO TO 200
      NGPCLR=NCSEST-NGPCOM
      CALL CLEAR2(GPNO,NGPCLR,1)
      CALL CLEAR2(INDXCS,NGPCLR,3)
      DO 175 I=1,NGPARC
      I1=901-I
      INDXCS(1,I)=BSTYPE(I1)
      NM=BSTNOS(I1)
C STORE BIASES
      N=NM/100
      M=NM-N*100
      INDXCS(2,I)=N
      INDXCS(3,I)=M
      GPVALO(I)=BIAS(I1)
      GPSIG(I)=BIASVR(I1)
      INPARI=INPARI+1
      GPNO(I)=INPARI
      M=M+1
      IF(INDXCS(1,I).EQ.1) GO TO 175
      N=31-N
      M=34-M
C WRITE OUT ARC DATA
  175 GPVAL(I)=CS(N,M)/BIASVR(I1)
  200 IF(NBIAS.LT.1) GO TO 300
      L1=3*NSAT
      DO 250 I=1,NBIAS
      I1=I+L1
      BSTRT(I)=ETIME1(I)
      BSEND(I)=ETIME2(I)
      BIASO(I1)=BIAS(I)
      EBIAS(I1)=BIAS(I)
      BTYPE(I)=BSTYPE(I)
      BIASSG(I1)=BIASVR(I)
      BSTANO(I)=BSTNOS(I)
  250 BIASNO(I1)=INPARI+I
  300 CALL DATARD(ARCNO,.TRUE.,.TRUE.,.FALSE.)
      REWIND SCRC
      END FILE SCRA
      REWIND SCRA
      RETURN
      END
```


ENTRY.
STORE
STORES COMMON AND ARC INFORMATION ON DISK
LOAD DRAG AND SOLRAD PARAMETERS
LOAD ESTIMATED GEOPOTENTIAL RESONANCE COEFFICIENTS
CLEAR2.
SET ARRAYS TO ZERO
CLEAR2
SET ARRAYS TO ZERO
LOAD BIASES
A


A
DATARD
STORES AND RETRIEVES ARC INFORMATION ON SCRATCH FILES
RETURN

NAME AREAS

PURPOSE TO COMPUTE THE ELLIPSOIDAL SURFACE AREA LYING BETWEEN TWO FIXED LATITUDES AND TWO FIXED LONGITUDES

CALLING SEQUENCE X=AREAS(Z1,Z2,TLON)

SYMBOL	TYPE	DESCRIPTION
Z1	DP	INPUT - Z-COORDINATE ASSOCIATED WITH LATITUDE 1
Z2	DP	INPUT - Z-COORDINATE ASSOCIATED WITH LATITUDE 2
TLON	DP	INPUT - THE DIFFERENCE (IN DEGREES) BETWEEN THE TWO LONGITUDES

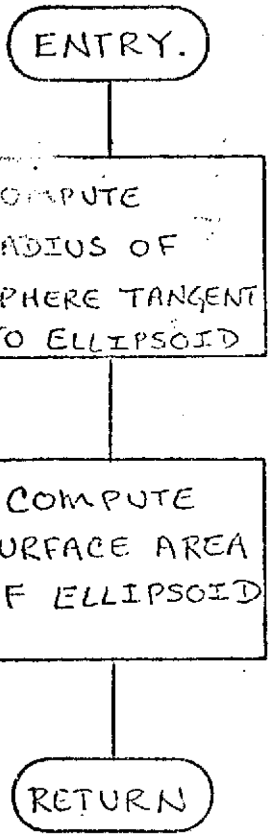
SUBROUTINES USED NONE

COMMON BLOCKS CONSTS INTBLK

INPUT FILES NONE

OUTPUT FILES NONE

```
      DOUBLE PRECISION FUNCTION AREAS(Z1,Z2,TLON)
      IMPLICIT REAL*8(A-H,O-Z)
      LOGICAL NOT1ST
      COMMON/CONSTS/DPI,DTWOPI,DEG2RD,DFSEC
      COMMON/INTBLK/THDOTS(3),GM,AE,AESQ,FLAT,FSQ32(59)
      DATA NOT1ST/.FALSE./
      Z=.5D0*(Z1+Z2)
C COMPUTE SURFACE AREAS OF ELLIPSOIDAL SURFACES
      IF(NOT1ST) GO TO 10
      NOT1ST=.TRUE.
      ESQ =FLAT*(2.D0-FLAT)
      CSQ =AESQ*ESQ
      C   =DSQRT(CSQ)
      BSQ =AESQ-CSQ
      B4  =BSQ**2
   10 ROOT1=DSQRT(CSQ*Z1**2+B4)
      ROOT2=DSQRT(CSQ*Z2**2+B4)
      AREAS=DABS(.5D0*TLON*DEG2RD*AE*((Z2*ROOT2-Z1*ROOT1)/BSQ+BSQ*DLOG((
     *                                C*Z2+ROOT2)/(C*Z1+ROOT1))/C))
      RETURN
      END
```



REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

AVGPOT

DESCRIPTION

Calls EGRAV to compute Legendre polynomials for geopotential expansion for different latitudes. Evaluates potential at these different latitudes using even zonal harmonic coefficients only.

Computes average of these potentials weighting by the cosine of the geocentric latitude.

NAME AVGPOT

PURPOSE TO COMPUTE AVERAGE GRAVITATIONAL POTENTIAL OF EARTH USING EVEN ZONAL HARMONIC COEFFICIENTS ONLY

CALLING SEQUENCE X=AVGPOT(NLAT)

SYMBOL	TYPE	DESCRIPTION
NLAT	I	INPUT - NUMBER OF LATITUDE DIVISIONS USED IN AVERAGE POTENTIAL EVALUATION
X	DP	OUTPUT - AVERAGE GRAVITATIONAL POTENTIAL OF THE EARTH

SUBROUTINES USED EGRAV

COMMON BLOCKS CONSTS FMODEL INTBLK XYZ- VRBLOK

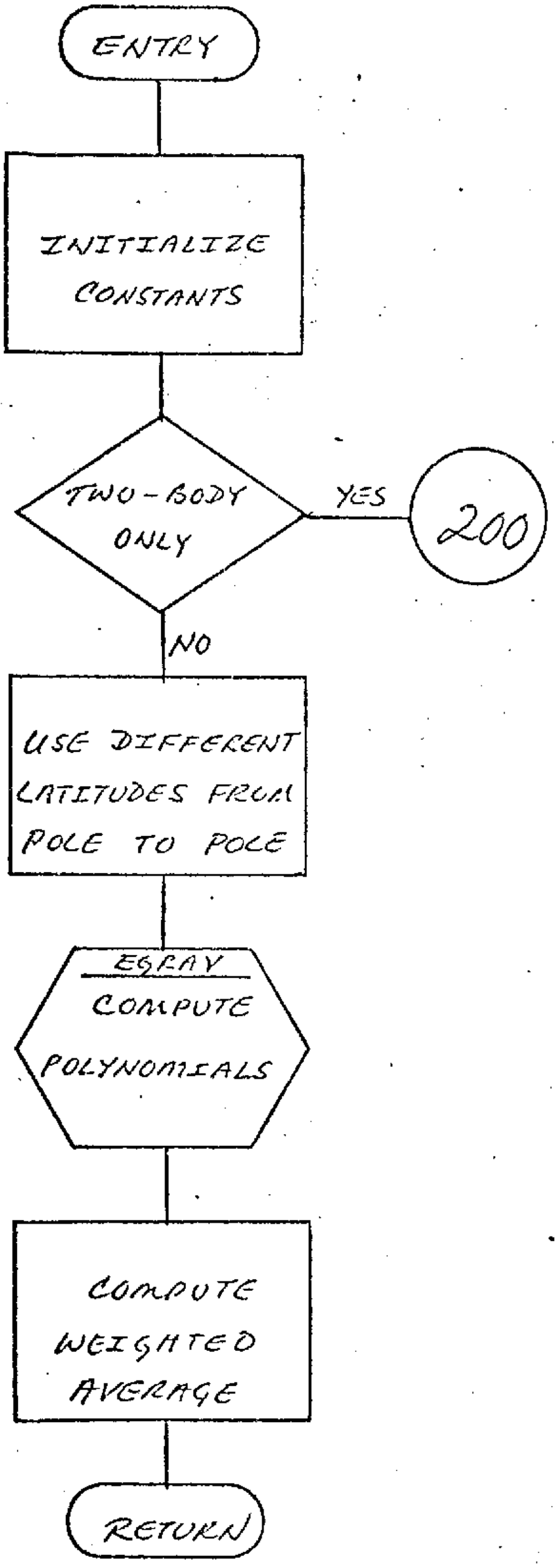
INPUT FILES NONE

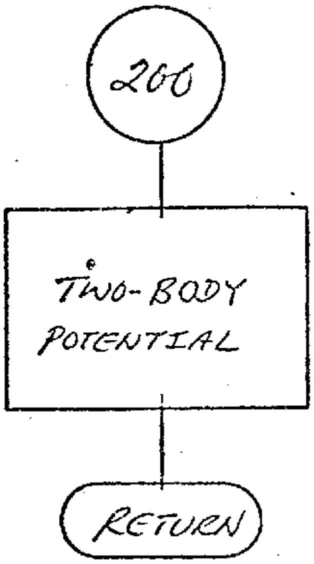
OUTPUT FILES NONE

REFERENCES 'GEODYN SYSTEMS DESCRIPTION'
VOLUME 1 - GEODYN DOCUMENTATION

```
      DOUBLE PRECISION FUNCTION AVGPOT(NLAT)
      IMPLICIT REAL*8 (A-H,O-Z)
      DOUBLE PRECISION MODEL
      DIMENSION FCT(3)
      COMMON/CONSTS/DPI,DTWOPI,DRAD,DRSEC
      COMMON/FMODEL/INDEX1,INDEX2,INDEX3,INDEX4,CS(30,33),MODEL(8)
      COMMON/INTBLK/THDOT1,THDOT2,THDT2S,GM,AE,AESQ,FLAT,FSQ32(59)
      COMMON/XYZ/XYZ(6),R,RSQ,ISAT,IFORCE(2)
      COMMON/VRBLOK/XYSQ,COSLAM(31),SINLAM(31),PR,PPSI,PLAMDA,
     . P(33,30),ACHN(30),TPSIM(39)
      EQUIVALENCE (SINPSI,P(1,1)),(COSPSI,P(2,1))
C INITIALIZE CONSTANTS
      F1=1.0D0-FLAT
      F1SQ=F1*F1
      F2F=FLAT*(2.0D0-FLAT)
      THETG=0.0D0
      XYZ(2)=0.0D0
      DLAT=180.0D0/DFLOAT(NLAT)
      XLAT=-90.0D0+0.25D0*DLAT
      U=0.0D0
      GMR=0.0D0
      SUMCOS=0.0D0
      IF(INDEX2.LT.2) GO TO 200
C USE DIFFERENT LATITUDES FROM POLE TO POLE
      DO 100 LAT=1,NLAT
      PSI=XLAT*DRAD
```

```
      SINPSI=DSIN(PSI)
      SPSISQ=SINPSI**2
      COSPSI=DSQRT(1.0D0-SPSISQ)
       RHO=AE*F1/DSQRT(F1SQ+F2F*SPSISQ)
      XYZ(1)=RHO*COSPSI
      XYZ(3)=RHO*SINPSI
C COMPUTE POLYNOMIALS.
      CALL EGRAV(THETG,RASAT,FCT)
      THDTR2=(R*THDT2S*COSPSI)**2
      SUM=0.0D0
      FN=0.0D0
      DO 50 N=2,INDEX2,2
      FN=FN+2.0D0
   50 SUM=SUM+FN*AORN(N)*P(1,N)*CS(N,1)
      U=U+SUM*COSPSI
      GMR=GMR+(GM/R+0.5D0*THDTR2)*COSPSI
      SUMCOS=SUMCOS+COSPSI
  100 XLAT=XLAT+DLAT
C COMPUTE WEIGHTED AVERAGES
      AVGPOT=(GMR+U)/SUMCOS
      RETURN
C COMPUTE TWO-BODY POTENTIAL
  200 GMR=GM/6367462.55D0+(THDT2S*6367462.55D0*DCOS(DPI*0.25D0))**2*.5D0
      AVGPOT=GMR
      RETURN
      END
```

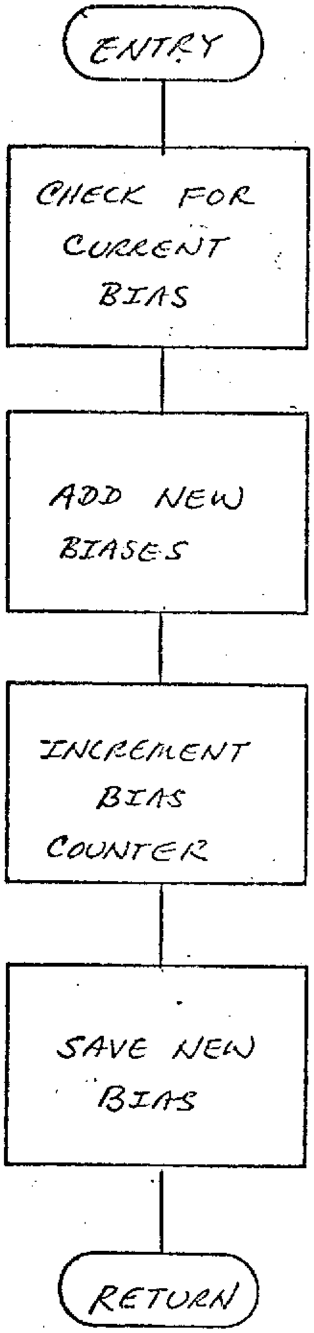

ENTRY
INITIALIZE CONSTANTS
TWO-BODY ONLY
YES
200
NO
USE DIFFERENT LATITUDES FROM POLE TO POLE
EGRAY
COMPUTE POLYNOMIALS
COMPUTE WEIGHTED AVERAGE
RETURN


200
TWO-BODY POTENTIAL
RETURN

NAME	BIAS
PURPOSE	TO EXTRACT BIAS START - STOP TIMES FROM DATA AND COUNT BIASES
CALLING SEQUENCE	CALL BIAS
SUBROUTINES USED	NONE
COMMON BLOCKS -	APARAM CEPHEM PREBLK FLXBLK VRBLOK
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      SUBROUTINE BIAS
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL*1 VHFCHN,PREPRO,TIMING
      LOGICAL TWOSTA
      INTEGER*2 MTYPE,NMEAS,PRETYP,CHANEL,BTYPE,BSTANO,ISTANO
      INTEGER RECNO
      REAL BIASO,BIASSG
      DIMENSION NWBIAS(3)
      COMMON/APARAM/INPAR,INPAR1,NBIAS,NSTSTA(2),NGPARC,NOREC1(4)
      COMMON/CEPHEM/CNAME(300),ISTARG(300),ISTANO(300)
      COMMON/PREBLK/DAY,OBSU(2),SIG(2),SRFNDX,ISTA,MTYPE,NMEAS,
     'PRETYP(2),CHANEL,VHFCHN,PREPRO,RECNO
      COMMON/FLXBLK/BSTRT(900),BSEND(900),BTYPE(900)
      COMMON/VRBLOK/BIASO(900),BIASSG(900),BSTANO(900)
      EQUIVALENCE (NEWB0,NWBIAS(1)),(NEWB1,NWBIAS(2)),(NEWB2,NWBIAS(3))
      DATA TPASS/0.1D-1/
      IF(NBIAS.GE.NGPARC) RETURN
      IF(ISTA.LE.0) RETURN
      TWOSTA=MTYPE.GT.20.AND.CHANEL.NE.ISTA
      TIMING=.FALSE.
      IBSTA=ISTANO(ISTA)
  150 DO 100 I=1,3
C CHECK FOR CURRENT BIAS
  100 NWBIAS(I)=0
      DO 1200 I=1,NBIAS
      IF(IBSTA.NE.BSTANO(I)) GO TO 1200
      I1=BTYPE(I)
      IF(I1.EQ.0) GO TO 175
      IF(TIMING) GO TO 1200
      IF(MTYPE.GT.20.AND.MTYPE.EQ.11) GO TO 175
      IF(MTYPE.NE.(I1-(I1/8)*7)) GO TO 1200
  175 IF(DAY.LT.BSTRT(I)) GO TO 1200
      IF(DAY.LT.BSEND(I)) GO TO 300
      IF(BSEND(I)-BSTRT(I)) 1200,200,700
  200 BSTRT(I)=DAY
      BSEND(I)=DAY+TPASS
  300 IF(BTYPE(I)-MTYPE) 400,500,600
```

```
  400 NEWTB=0
      GO TO 1200
  500 NEWB1=0
      GO TO 1200
  600 NEWB2=0
      GO TO 1200
  700 IF(DAY.GT.BSEND(I)+TPASS) GO TO 800
      BSEND(I)=BSEND(I)+TPASS
      GO TO 300
  800 IF(BTYPE(I)-MTYPE) 900,1000,1100
  900 NEWTB=1
      GO TO 1200
 1000 NEWB1=1
      GO TO 1200
C ADD NEW BIASES
 1100 NEWB2=1
 1200 CONTINUE
      DO 1300 I=1,3
      I1=NWBIAS(I)
C INCREMENT BIAS COUNTER
      IF(I1.EQ.0) GO TO 1300
C SAVE NEW BIASES
      IF(NBIAS.GE.NGFARC) RETURN
      NBIAS=NBIAS+1
      BIASSG(NBIAS)=BIASSG(I1)
      BIASO(NBIAS)=BIASO(I1)
      BTYPE(NBIAS)=BTYPE(I1)
      BSTRT(NBIAS)=DAY
      BSEND(NBIAS)=DAY+TPASS
      BSTAND(NBIAS)=IBSTA
 1300 CONTINUE
      IF(.NOT.TWOSTA) RETURN
      TIMING=.TRUE.
      TWOSTA=.FALSE.
      IBSTA=ISTAND(CHANEL)
      GO TO 150
      END
```


ENTRY
CHECK FOR CURRENT BIAS
ADD NEW BIASES
INCREMENT BIAS COUNTER
SAVE NEW BIAS
RETURN

NAME	BMTWRT
PURPOSE	TO WRITE OUT B-MATRIX
CALLING SEQUENCE	CALL BMTWRT(SUM1,SUM2,INDXCS,GPNO,GPVALO,GPSIG, ESTANO,ISTANO,STAXYZ,FILENO)

SYMBOL	TYPE	DESCRIPTION
SUM1 (1)	DP	INPUT - NORMAL MATRIX IN VECTOR FORM
SUM2 (1)	DP	INPUT - RIGHT HAND SIDE
INDXCS (3,1)	I*2	INPUT - INDICES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
GPNO (1)	I*2	INPUT - LOCATIONS OF GEOPOTENTIAL PARAMETERS IN NORMAL MATRIX
GPVALO (1)	DP	INPUT - A PRIORI ESTIMATES OF GEOPOTENTIAL COEFFICIENTS
GPSIG (1)	DP	INPUT - SIGMAS OF ADJUSTED GEOPOTENTIAL PARAMETERS
ESTANO (1)	I*2	INPUT - MASTER STATION NUMBERS FOR ESTIMATED STATIONS
ISTANO (1)	I*2	INPUT - STATION NUMBERS
STAXYZ (3,1)	DP	INPUT - STATION COORDINATES
FILENO	I	OUTPUT - FILE NUMBER FOR THE B-MATRIX TAPE

SUBROUTINES USED	NUMBR2				
COMMON BLOCKS	APARAM	CPARAM	CSTAT	CSTINF	FLXBLK
	INITBK	INTBLK	PRIORI	TPEBLK	VRBLOK
INPUT FILES	NONE				
OUTPUT FILES	BMATTP - OUTP				
REFERENCES	'GEODYN SYSTEMS DESCRIPTION' VOLUME 1 - GEODYN DOCUMENTATION				
	'GEODYN PROGRAM OPERATIONS DESCRIPTION' - APPENDIX C VOLUME 3 - GEODYN DOCUMENTATION				

```
      SUBROUTINE BMTWRT(SUM1,SUM2,INDXCS,GPNO,GPVALO,GPSIG,ESTANO,
     *   ISTANO,STAXYZ,FILENO)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL CMPGPC,PRNMAT
      INTEGER*2 GPNO,ESTANO,ISTANO,INDXCS
      INTEGER RECTYP,ZERO,CODE,ESTSTA,ONE,BMATNO,BMATTP,ADDR,ADDRD,SRAD,
     *   OUTP,FILENO
      REAL MATNAM,RESID,SIG,WTSUMT
      DIMENSION SUM1(1),SUM2(1),INDXCS(3,1),GPNO(1),GPVALO(1),GPSIG(1),
     *   ESTANO(1),ISTANO(1),STAXYZ(3,1)
      DIMENSION RECTYP(7),MATNAM(3)
      COMMON/APARAM/INPAR,INPAR1,NOBIAS,ESTSTA,NSAT,NGPARC,NOREC1,NPARAM,
     *   NEBIAS,MAXPAR
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,NGPCOM,
     *   NCSEST,CMPGPC,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,INNRSW,
     *   NCONST,NDCONS
      COMMON/CSTAT/RESID,SIG,NMTCT,WTSUMT
      COMMON/CSTINF/MEASNO(262),NOBS,LBASE
      COMMON/FLXBLK/BMATRX(1001),VALUE(1000),V(3),FLXB1(21)
      COMMON/INITBK/INITB1(44),PRNMAT,BMATNO,INITB2(11)
      COMMON/INTBLK/INTB1(118),ADDR(2),ADDRD(2),SRAD(2),INTB2(3)
      COMMON/PRIOR1/ELEMO(6,2),PRIOB1(78),CD(2),CDD(2),EMISS(2)
      COMMON/TPEBLK/INTP,OUTP,ITAPES(10)
      COMMON/VRBLOK/VRB1(56),LABEL(1000),IROW(1000),IVRB2(118)
      DATA RECTYP/10001,10002,10003,10011,10012,10013,-19991/
      DATA MATTYP/1/,MATNAM/3*6H      /,ONE/1/,ZERO/0/,CODE/3/
      DATA BMATTP/71/
      INDXNO(I)=NDIM*(I-1)-(I*(I-1))/2
      NCON=MAX0(ADDR(1),ADDR(2),ADDRD(1),ADDRD(2),SRAD(1),SRAD(2))
      NPARM=NGPCOM+2*NMAST+6*NSAT+NCON
C WRITE HEADER RECORD
      NELM=NPARM+1
      V(1)=WTSUMT
      V(2)=WTSUMT
      V(3)=0.0D0
      WRITE (BMATTP) RECTYP(1),BMATNO,NPARM,NELM,V,NOBS,MATTYP,MATNAM
C FROM THIS POINT TO LABEL 30...
C REORDER SUM1 INTO PROPER ORDER FOR B-MATRIX
      IF (PRNMAT) WRITE (OUTP,10001) BMATNO,NPARM,NOBS
C DENORMALIZE THE PARTIALS FOR GEOPOTENTIAL COEFFICIENTS
      IF (NGPCOM.LE.0) GO TO 35
      K=NCSEST-NGPCOM
      DO 10 I=1,NGPCOM
      K=K+1
      LABEL(I)=(INDXCS(1,K)*100+INDXCS(3,K))*100+INDXCS(2,K)
      IROW(I)=GPNO(I)
      VALUE(I)=GPVALO(I)*GPSIG(I)
      J1=IROW(I)
      I1=INDXNO(J1)
      DO 5 J=J1,NDIM
      IJ=I1+J
      SUM1(IJ)=SUM1(IJ)/GPSIG(I)
    5 CONTINUE
      IJ=J1
      DO 7 I1=1,J1
```

```
      SUM1(IJ)=SUM1(IJ)/GPSIG(I)
      IJ=IJ+NDIM-I1
    7 CONTINUE
      SUM2(J1)=SUM2(J1)/GPSIG(I)
   10 CONTINUE
      K=NGPCOM-1
      DO 30 I=1,K
      IPLUS1=I+1
      DO 20 J=IPLUS1,NGPCOM
      IF (LABEL(I).LE.LABEL(J)) GO TO 20
      NTEMP=LABEL(I)
      LABEL(I)=LABEL(J)
      LABEL(J)=NTEMP
      NTEMP=IROW(I)
      IROW(I)=IROW(J)
      IROW(J)=NTEMP
      TEMP=VALUE(I)
      VALUE(I)=VALUE(J)
      VALUE(J)=TEMP
   20 CONTINUE
   30 CONTINUE
   35 IF (NMAST.LE.0) GO TO 65
C ADD STATION ADJUSTMENT TERMS TO B-MATRIX
      I1=NDIM-3*NMAST-NGPCOM
      L=0
      DO 60 I=1,NMAST
   40 L=L+1
      IF (NUMBR2(L,ESTAND,NSTEST).EQ.0) GO TO 40
      K=NGPCOM+(I-1)*3
      DO 50 J=1,3
      LABEL(J+K)=(J+4)*10000+ISTAND(L)
      IROW(J+K)=I1+J+K
      VALUE(J+K)=STAXYZ(J,L)
   50 CONTINUE
   60 CONTINUE
      IF (NMAST.LT.2) GO TO 65
      K=NGPCOM+3
      L=3*(NMAST-1)+NGPCOM
      DO 57 I=K,L,3
      DO 56 J=I,L,3
      IF (LABEL(I-2).LE.LABEL(J+1)) GO TO 56
      DO 55 IJ=1,3
      I1=I+IJ-3
      J1=J+IJ
      NTEMP=LABEL(I1)
      LABEL(I1)=LABEL(J1)
      LABEL(J1)=NTEMP
      NTEMP=IROW(I1)
      IROW(I1)=IROW(J1)
      IROW(J1)=NTEMP
      TEMP=VALUE(I1)
      VALUE(I1)=VALUE(J1)
      VALUE(J1)=TEMP
   55 CONTINUE
   56 CONTINUE
   57 CONTINUE
```

```
C ADD ELEMENTS, DRAG, & SOLAR RADIATION
   65 L=NPARM-6*NSAT-NCON
      DO 80 I=1,NSAT
      IF (SRAD(I).LE.0) GO TO 61
      L=L+1
      LABEL(L)=201+100*I
      IROW(L)=NSAT*6+SRAD(I)
      VALUE(L)=EMISS(I)
   61 IF (ADDR(I).LE.0) GO TO 62
      L=L+1
      LABEL(L)=111+100*I
      IROW(L)=NSAT*6+ADDR(I)
      VALUE(L)=CD(I)
   62 IF (ADDRD(I).LE.0) GO TO 63
      L=L+1
      LABEL(L)=112+100*I
      IROW(L)=NSAT*6+ADDRD(I)
      VALUE(L)=CDD(I)
   63 DO 70 J=1,6
      LABEL(L+J)=100*I+J
      IROW(L+J)=J+(I-1)*6
      VALUE(L+J)=ELEMO(J,I)
   70 CONTINUE
      L=L+6
   80 CONTINUE
C PRINT B-MATRIX LABELS
      IF (PRNMAT) WRITE (OUTP,20001)
      WRITE (BMATTP) RECTYP(2),ZERO,(LABEL(I),I=1,NPARM)
      IF (PRNMAT) WRITE (OUTP,20002)
      IF (PRNMAT) WRITE (OUTP,10002) (LABEL(I),I=1,NPARM)
C PRINT B-MATRIX BY ROWS
      IF (PRNMAT) WRITE (OUTP,20004)
      DO 130 J=1,NPARM
      J1=IROW(J)
      BMATRX(1)=SUM2(J1)
      DO 110 I=1,NPARM
      I1=IROW(I)
      IJ=INDXNC(MINO(I1,J1))+MAXO(I1,J1)
      BMATRX(I+1)=SUM1(IJ)
  110 CONTINUE
      WRITE (BMATTP) RECTYP(3),(BMATRX(I),I=1,NELM)
      IF (PRNMAT) WRITE (OUTP,10003) LABEL(J),(BMATRX(I),I=1,NELM)
C SET PARAMETER IDENTITIES
  130 CONTINUE
      WRITE (BMATTP) RECTYP(4),BMATNO,ONE,NPARM,(ZERO,I=1,7),CODE,MATNAM
      IF (PRNMAT) WRITE (OUTP,10011) BMATNO,NPARM
      WRITE (BMATTP) RECTYP(5),(LABEL(I),I=1,NPARM)
      IF (PRNMAT) WRITE (OUTP,20003)
      IF (PRNMAT) WRITE (OUTP,10012) (LABEL(I),I=1,NPARM)
      WRITE (BMATTP) RECTYP(6),(VALUE(I),I=1,NPARM)
C PRINT 2 TRAILER RECORDS
      IF (PRNMAT) WRITE (OUTP,10013) (VALUE(I),I=1,NPARM)
      WRITE (BMATTP) RECTYP(7),(ZERO,I=1,14)
      WRITE (BMATTP) RECTYP(7),(ZERO,I=1,14)
      END FILE BMATTP
      FILENO=FILENO+1
```

```
      WRITE (OUTP,20005) BMATNO,FILENO,BMATTP
      RETURN
10001 FORMAT ('1B-MATRIX NO.',I6,' WITH',I4,' ROWS AND REPRESENTING',
     .   I6,' WEIGHTED OBSERVATIONS.'/)
10002 FORMAT ((19X,7(10X,I5)/))
10003 FORMAT (1X,I5,1PD15.6,1X,'*',7D15.6/(22X,'*',7D15.6))
10011 FORMAT ('1B-MATRIX NO.',I6,'.',I5,' A-PRIORI PARAMETER VALUES.'/)
10012 FORMAT ((4X,8(10X,I5)/))
10013 FORMAT (/(2X,1P8D15.6))
20001 FORMAT ('0EXPLANATION OF LABELS'//1X,'GEOPOTENTIAL',8X,'C(N,M)',
     .   4X,'1MMNN'/1X,'COEFFICIENTS',8X,'S(N,M)',4X,'2MMNN'//1X,
     .   'STATION JJJJ',8X,'X(JJJJ)',3X,'5JJJJ'/1X,'COORDINATES',9X,
     .   'Y(JJJJ)',3X,'6JJJJ'/21X,'Z(JJJJ)',3X,'7JJJJ'//1X,
     .   'SOLAR RADIATION',5X,'CR',10X,'301'//1X,'DRAG CONSTANT',7X,
     .   'CD',10X,'211'//1X,'DRAG DOT CONSTANT',3X,
     .   'CDD',9X,'212'//1X,'POSITION',12X,'X',11X,'101'/1X,'AND',17X,
     .   'Y',11X,'102'/1X,'VELOCITY',12X,'Z',11X,'103'/1X,'VECTORS',
     .   13X,'XDOT',8X,'104'/21X,'YDOT',8X,'105'/21X,'ZDOT',8X,'106'/)
20002 FORMAT (74X,'COLUMN LABELS'/)
20003 FORMAT (58X,'PARAMETER LABELS'/)
20004 FORMAT ('0 ROW',5X,'RIGHT-HAND',2X,44('*'),' NORMAL B-MATRIX ',
     .   45('*')/' LABEL',7X,'SIDE',5X,'*')
20005 FORMAT ('1B-MATRIX NO.',I6,' HAS BEEN WRITTEN ON FILE NO.',I4,
     .   ' OF OUTPUT UNIT NO.',I3,'.')
      END
```

NAME BSCOMP

ENTRY POINT	PURPOSE
BSCMP1	INITIALIZATION
BSCOMP	TO COMPUTE ELECTRONIC BIASES AND CORRECT NORMAL EQUATIONS FOR EXTRACTION OF THE ELECTRONIC BIASES

CALLING SEQUENCE CALL BSCMP1(BTWA,BTWR,BTWB,BTIME,SUM1,SUM2,NEBIAS, BESTNO,NUMPER,BETYPE,PARNOS)

SYMBOL	TYPE	DESCRIPTION
BTWA (NEBIAS,1)	DP	SCRATCH - VECTOR OF SUMS OF PRODUCT OF BIAS PARTIAL TIMES WEIGHT TIMES PARTIALS OF ADJUSTED PARAMETERS
BTWR (1)	DP	SCRATCH - VECTOR OF SUMS OF PRODUCT OF BIAS PARTIAL TIMES WEIGHT TIMES RESIDUAL
BTWB (1)	DP	SCRATCH - SUMS OF WEIGHTS
BTIME (1)	DP	SCRATCH - BIAS TIME ARRAY USED TO DETERMINE START OF NEW PASSES .
SUM1 (1)	DP	INPUT/OUTPUT - NORMAL MATRIX
SUM2 (1)	DP	INPUT/OUTPUT - RIGHT-HAND SIDE OF NORMAL EQUATIONS
NEBIAS	I*4	INPUT - MAXIMUM NUMBER OF ELECTRONIC BIASES IN ANY ONE PASS
BESTNO (1)	I*2	INPUT - ELECTRONIC BIAS STATION NUMBER
NUMPER (1)	I*2	SCRATCH - ARRAY FOR COUNTING THE NUMBER OF MEASUREMENTS IN THE PASS
BETYPE (1)	I*2	INPUT - ELECTRONIC BIAS TYPE NUMBERS
PARNOS (1)	I*2	INPUT - MEASUREMENT PARAMETER NUMBERS

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

CALLING SEQUENCE CALL BSCOMP(IBIAS,RESID,SIG,TIME,PMPXC,LINNER)

SYMBOL	TYPE	DESCRIPTION
IBIAS	I*4	INPUT - ELECTRONIC BIAS INDEX NUMBER
RESID	DP	INPUT - MEASUREMENT RESIDUAL

```
SIG      D8       INPUT - MEASUREMENT SIGMA

TIME     R*4      INPUT - TIME OF MEASUREMENT

PMPXC    D8       INPUT - MEASUREMENT PARTIALS
(1)

LINNER   L*4      INPUT - LOGICAL SWITCH -- .TRUE. = LAST INNER
                                                    ITERATION

SUBROUTINES USED   STAIF2

COMMON BLOCKS      APARAM     CPARAM

INPUT FILES        NONE

OUTPUT FILES       NONE
```

```
      SUBROUTINE BSCMP1(BTWA,BTWD,BTWB,BTIME,SUM1,SUM2,NEBIAS,BESTNO,
     . NUMBER,BETYPE,PARNOS)
     .    SUM2,NEBIAS,BESTNO,NUMBER,BETYPE,PARNOS)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL CMPGPR,LINNER,ARCPAR
      INTEGER*2 PARNOS,BESTNO,NUMBER,BETYPE
      INTEGER RECNO1,PARMAX
      REAL BTIME,TIME
      DIMENSION BTWA(NEBIAS,1),BTWD(1),BTWB(1),
     .    BTIME(1),SUM1(1),SUM2(1),PMPXC(1),
     .   PARNOS(1),BESTNO(1),NUMBER(1),BETYPE(1)
      COMMON/APARAM/INPAR,INPAR1,NSTSTA,NBIAS,NSAT,NGPARC,
     .    RECNO1,NPARAM,NBIASE,PARMAX
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,
     .    NGPC1,NGPC2,NGPCOM,NCSEST,CMPGPR,LIM1,
     .    LIM2,NDEN,NDENST,NTIDST,NTIDEN,INNRSW,
     .    NCONST,NCCONS
      INDXNC(I)=NDIM*(I-1)-(I*(I-1))/2
C INITIALIZE
      NSTART = NDIM - 3*NMAST-NGPCOM-NTIDEN+1
      RETURN
      ENTRY BSCOMP(IBIAS,RESID,SIG,TIME,PMPXC,LINNER)
      IB1=1
      IB2=NBIASE
      IF(IBIAS.EQ.0) GO TO 500
      IB1=IBIAS
      IB2=IB1
C IF NEW PASS HAS STARTED, SOLVE FOR BIAS ON PREVIOUS PASS.
      IF(TIME.GT.(BTIME(IBIAS)+0.1E-1)) GO TO 500
C SUM INTO NORMAL EQUATIONS FOR ELECTRONIC BIASES
  100 WT= 1.0D0/(SIG*SIG)
      NUMBER(IBIAS)=NUMBER(IBIAS)+1
      DO 200 J=1,PARMAX
      I1=PARNOS(J)
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      IF(I1.EQ.0) GO TO 200
      BTWA(IBIAS,J)=BTWA(IBIAS,J)+WT*DMPXO(J)
  200 CONTINUE
      BTWD(IBIAS)=BTWD(IBIAS)+WT*RESID
      BTWB(IBIAS)=BTWB(IBIAS)+WT
      BTIME(IBIAS)=TIME
      RETURN
  500 ARCPAR=NSTART.GT.NDIM.OR..NOT.LINNER
  600 J=0
      N=NUMBER(IB1)
      IF(N.LE.0) GO TO 1200
C COMPUTE BIAS
      BTWB1=1.D00/BTWB(IB1)
      BIAS=BTWB1*BTWD(IB1)
      ISTA=BFSTNO(IB1)
      SIG1=DSQRT(DFLOAT(N)*BTWB1)
      MTYPE=BSTYPE(IB1)
      PRINT 2000,ISTA,MTYPE,BIAS
 2000 FORMAT('0***** STATION',I3,' TYPE',I2,' BIAS =',G25.16)
C UPDATE STATISTICAL INFORMATION
      CALL STAIF2(ISTA,MTYPE,SIG1,BIAS,N,NSAT)
C CORRECT NORMAL MATRIX
      DO 900 I=1,NPARAM
      SUM=BTWA(IB1,I)*BTWB1
      DO 700 K=I,NPARAM
      L=J+K
  700 SUM1(L)=SUM1(L)-SUM*BTWA(IB1,K)
      SUM2(I)=SUM2(I)-SUM*BTWD(IB1)
      BTWA(IB1,I)=0.0D0
      IF(ARCPAR) GO TO 900
      DO 800 K=NSTART,NDIM
      L=J+K
  800 SUM1(L)=SUM1(L)-SUM*BTWA(IB1,K)
  900 J=J+NDIM -I
      IF(ARCPAR) GO TO 1200
      J=INDXND(NSTART)
      DO 1100 I=NSTART,NDIM
      SUM=BTWA(IB1,I)*BTWB1
      DO 1000 K=I,NDIM
      L=J+K
 1000 SUM1(L)=SUM1(L)-SUM*BTWA(IB1,K)
      SUM2(I)=SUM2(I)-SUM*BTWD(IB1)
      BTWA(IB1,I)=0.0D0
 1100 J=J+NDIM-I
C ZERO SUMMING ARRAYS
 1200 BTWB(IB1)=0.0D0
      BTWD(IB1)=0.0D0
      BTIME(IB1)=0.0
      NUMBER(IB1)=0
      IB1=IB1 + 1
C IF LAST CALL FOR ARC, PROCESS LAST PASS FOR EACH ITERATION (GO TO 600)
      IF(IB1.LE.IB2) GO TO 600
C AFTER ALL PASSES COMPLETE, RETURN IF LAST CALL FOR ITERATION
      IF(IBIAS.EQ.0) RETURN
      IB1=IB2
C SAVE TIME OF LAST MEASUREMENT FOR CHECK TO DETERMINE NEW PASS
      BTIME(IB1)=TIME
      GO TO 1[illegible]
      END
```

NAME CBROWN

PURPOSE TO SET UP VARIABLE STORAGE ARRAYS AND TO INITIALIZE SUBROUTINES USING THESE ARRAYS

CALLING SEQUENCE

```
CALL CBROWN(IADDR,IBIAS,BBIAS,KSUM,NSUM,PLHN,
            ISTANO,ENV,SUM1,GRPAR,INDXCS,SUM2,
            XYZSIG,XFIT,XYZ,PMPXO,XI,PARNOS,DENSE,
            DENCON,CSUM,EBIAS)
```

SYMBOL	TYPE	DESCRIPTION
IADDR (13)	I	INPUT - ARRAY CONTAINING ADDRESSES OF THE VARIABLE STORAGE ARRAYS
IBIAS (1)	I*2	INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR BIASES
BBIAS (1)	DP	INPUT - VARIABLE STORAGE ARRAY NO. 2 FOR BIASES
KSUM (1)	I	INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR STATISTICS
NSUM (1)	I	INPUT - VARIABLE STORAGE ARRAY NO. 2 FOR STATISTICS
PLHN (1)	DP	INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR STATIONS
ISTANO (1)	I*2	INPUT - VARIABLE STORAGE ARRAY NO. 2 FOR STATIONS
ENV (1)	DP	INPUT - VARIABLE STORAGE ARRAY NO. 3 FOR STATIONS
SUM1 (1)	DP	INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR NORMAL EQUATIONS
GRPAR (1)	DP	INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR PARTIALS
INDXCS (1)	I*2	INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR GEOPOTENTIAL
SUM2 (1)	DP	INPUT - VARIABLE STORAGE ARRAY NO. 2 FOR NORMAL EQUATIONS
XYZSIG (1)	R	INPUT - VARIABLE STORAGE ARRAY NO. 4 FOR STATIONS
XFIT (1)	DP	INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR INTEGRATOR
XYZ	DP	INPUT - VARIABLE STORAGE ARRAY NO. 5 FOR STATIONS

```
(1)

PMPXO   DP       INPUT - VARIABLE STORAGE ARRAY NO. 2 FOR PARTIALS
(1)

XI      DP       INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR
(1)                      INTERPOLATOR

PARNOS  I*2      INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR PARAMETER
(1)                      NUMBERS

DENSE   DP       INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR DENSITIES
(1)

DENCON  DP       INPUT - VARIABLE STORAGE ARRAY NO. 2 FOR DENSITIES
(1)

CSUM    I        INPUT - VARIABLE STORAGE ARRAY NO. 3 FOR STATISTICS
(1)

EBIAS   I*2      INPUT - VARIABLE STORAGE ARRAY NO. 1 FOR ELECTRONIC
(1)                      BIASES

SUBROUTINES USED  CLEAR     BSCMP1    COMAD1    ORBIT1    PROCS1
                  STATF1    VEVAL1    F1        PDEN2     COMPAR
                  ESTIM1    RESPR1    SURON1    DRAG1     CLEAR2
                  TRUEP1    ARCPAR    DATRD1    GRHRA1    NONAME
                  CBSDT1    PREDC1    SQANT1    TWOST1

COMMON BLOCKS     APARAM    CPARAM    CONOUT    CSTINF    VRBLOK

INPUT FILES       NONE

OUTPUT FILES      NONE
```

```
      SUBROUTINE CBROWN(IADDR,IBIAS,EBIAS,KSUM,NSUM,PLHN,ISTANO,ENV,      CBRO  94
     .   SUM1,GRPAR,INDXCS,SUM2,XYZSIG,XFIT,XYZ,PMPXO,XI,PARNOS,          CBRO  95
     .   DENSE,DENCON,CSUM,EBIAS)                                         CBRO  96
      IMPLICIT REAL*8 (A-H,O-Z)                                           CBRO  97
      DIMENSION IADDR(16),IBIAS(1),EBIAS(1),KSUM(1),NSUM(1),PLHN(1),      CBRO  98
     .   ISTANO(1),ENV(1),SUM1(1),GRPAR(1),INDXCS(1),SUM2(1),             CBRO  99
     .   XYZSIG(1),XFIT(1),XYZ(1),PMPXO(1),XI(1),PARNOS(1),DENSE(1)       CBRO 100
      DIMENSION DENCON(1),CSUM(1),EBIAS(1)                                CBRO 101
      INTEGER*2 IBIAS,ISTANO,INDXCS,PARNOS,EBIAS                          CBRO 102
      INTEGER ESTSTA,CSUM                                                 CBRO 103
      LOGICAL CMPGPR                                                      CBRO 104
      REAL XYZSIG,RMSALL,OUTCON                                           CBRO 105
      COMMON/APARAM/INPAR,INPAR1,NBIAS,ESTSTA,NSAT,NGPARC,NOREC1,NPARAM,CBRO 106
     .   NEBIAS,MAXPAR                                                    CBRO 107
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,             CBRO 108
     .   NGPCOM,NCSEST,CMPGRP,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,        CBRO 109
     .   INNRSW,NCONST,NOCONS                                             CBRO 110
      COMMON/CONOUT/RMSALL,OUTCON,MINOUT,MAXOUT,LITRES,MAXSAT,MAX2IN,     CBRO 111
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

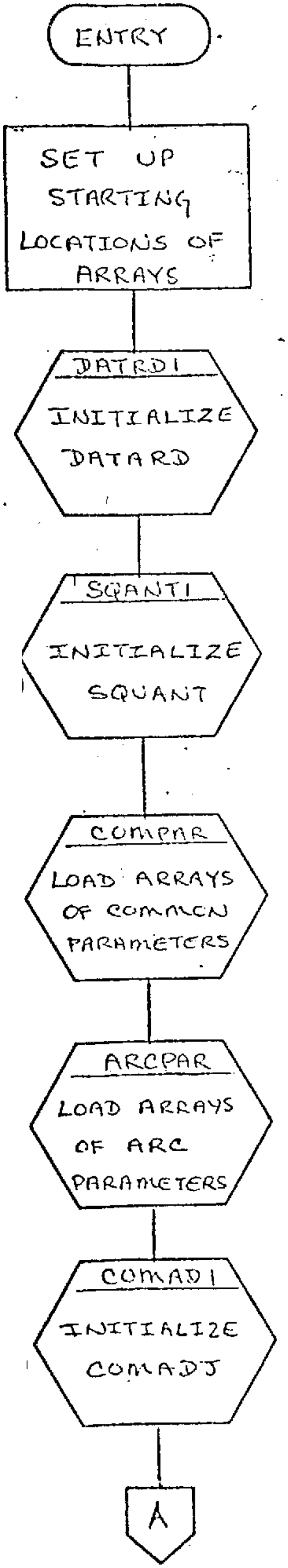
```
     .    NSTART,NEQNMX,IVAR,IORDER,NARCS,NSTARD,LSTART(6)
      COMMON/CSTINF/MEASNO(283),LBASE
      COMMON/VRBLCK/COSLAM(66),PR(1059)
C COMPUTE INDICES OF STARTING LOCATIONS OF DIFFERENT ARRAYS TO BE USED
      NBIASE=NCBIAS
      IB1=1
      IB2=IB1+4*MBIAS
      IB3=IB2+4*MBIAS
      MSAT3=MAXSAT*3
      IB4=IB3+4*(MBIAS+MSAT3+NCSEST)
      IB5=IB4+4*(MBIAS+MSAT3+NCSEST)
      IB6=IB5+MBIAS
      IB7=IB6+MBIAS
      IC1=IB3+4*(MBIAS+MSAT3)
      IC2=IB4+4*(MBIAS+MSAT3)
      IC3=IB7+MBIAS+MSAT3
      IBB1=1
      IBC1=IBB1+MBIAS+MSAT3
      IBD1=IBC1+NCSEST
      IBD2=IBD1+NTICST
      MBIAS=MBIAS+MSAT3
      NSTAT=NSTA+MAXSAT
      IP=1
      IL=IP+NSTA+1
      IH=IL+NSTA+1
      IN=IH+NSTA+1
      IP0=IN+NSTA+1
      IL0=IP0+NSTEST
      IH0=IL0+NSTEST
      ITHERM=IH0+NSTEST
      ISTA=1
      IEST=ISTA+NSTA+1
      ILOC=IEST+NSTEST
      IENV1=1
      IENV2=IENV1+3*(NSTA+1)
      IENV3=IENV2+3*(NSTA+1)
      ISUM2=1
      ITTL=ISUM2+NDIM
      IDELTA=ITTL+NDIM
      ISIG1=1
      ISIG2=9*NSTEST+ISIG1
      ISIG3=9*NSTEST+ISIG2
      ISIG4=9*NSTEST+ISIG3
      ISIG5=6*NSTEST+ISIG4
      IXFIT=1
      IFCT=IXFIT+6*NEQNMX*MAXSAT
      ISTNCS=3*NCSEST+1
      ISTXYZ=1
      ISTXY0=ISTXYZ+3*(NSTA+1)
      IXI=1
      IPXPX0=IXI+6
      IAREA=1
      ICENTR=IAREA+4*NDEN
      IBCENT=ICENTR+12*NDEN
      IDENS0=IBCENT+2*NDENST
      IDENS2=IDENS0+NTICST
```

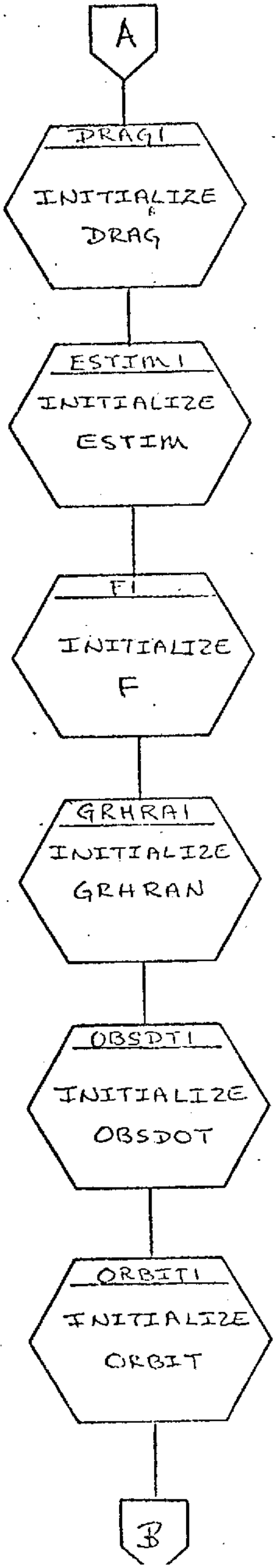
```
      IDSIG=IDENS0+NTIDEN+NCONST
      IDSIG2=IDSIG+NTIDST
      ICSUM=1
      IMSUM=ICSUM+NSTAT*MAXSAT*LBASE*16
      ILSUM=IMSUM+NSTAT*MAXSAT*LBASE*6
      IBTWA=1
      IBTWD=IBTWA+4*NDIM*NEBIAS
      IBTWB=IBTWD+4*NEBIAS
      IBTIME=IBTWB+4*NEBIAS
      IBESTA=IBTIME+2*NEBIAS
      IBETYP=IBESTA+NEBIAS
      INUMBR=IBETYP+NEBIAS
      NCLEAR=INUMBR+NEBIAS-1
      IF(NEBIAS.GT.0) CALL CLEAR2(EBIAS,NCLEAR,1)
      NPARM=NEQNMX+6
      NSTA1=NSTA+1
      NSTART=NDIM-NMAST*3-NGPCOM-NTIDEN
C INITIALIZE SUBROUTINES USING VARIABLE STORAGE ARRAYS
      CALL EATRD1(LBIAS,IBIAS,SUM1,INDXCS,SUM2,EBIAS(IBESTA))
      CALL SOANT1(PLHN(IH),XYZSIG(ISIG3),XYZSIG(ISIG2),XYZ(ISTXYZ),
     .   ENV(IENV1),ENV(IENV2),ENV(IENV3),PLHN(ITHPRM),PLHN(IP),
     .   PLHN(IL),ISTANO)
C LOAD COMMON PARAMETER INFORMATION INTO VARIABLE STORAGE ARRAYS
      CALL COMPAR(IBIAS(IC1),IBIAS(IC2),IBIAS(IC3),BBIAS(IBC1),
     .   PLHN(IP),PLHN(IL),PLHN(IH),PLHN(IN),PLHN(IP0),PLHN(IL0),
     .   PLHN(IH0),ISTANO(ISTA),ISTANO(IEST),INDXCS,SUM2(ITTL),
     .   XYZSIG(ISIG3),XYZSIG(ISIG2),XYZSIG(ISIG1),XYZSIG(ISIG4),
     .   XYZSIG(ISIG5),XYZ(ISTXYZ),XYZ(ISTXY0),INDXCS(ISTNDS),
     .   ISTANO(ILOC),DENSE(IDSIG2),DENSE(IDENS2),BBIAS(IBD2),
     .   DENSE(IAREA),DENSE(ICENTR),DENSE(IBCENT))
C LOAD ARC PARAMETER INFORMATION INTO VARIABLE STORAGE ARRAYS
      CALL ARCPAR(IBIAS(IB1),IBIAS(IB2),IBIAS(IB3),BBIAS(IBB1),
     .   IBIAS(IB4),IBIAS(IB6),IBIAS(IB5),IBIAS(IB7),IBIAS(IC1),
     .   IBIAS(IC2),IBIAS(IC3),BBIAS(IBC1),INDXCS,EBIAS(IBESTA),
     .   EBIAS(IBETYP),NEIASE,ISTANO)
C INITIALIZE REMAINING SUBROUTINES USING VARIABLE STORAGE ARRAYS
      CALL COMAD1(ISTANO(ISTA),ISTANO(IEST),ISTANO(ILOC),XYZ(ISTXYZ),
     .   XYZ(ISTXY0),PLHN(IP),PLHN(IL),PLHN(IH),PLHN(IP0),PLHN(IL0),
     .   PLHN(IH0),PLHN(IN),XYZSIG(ISIG3),XYZSIG(ISIG2),XYZSIG(ISIG5),
     .   XYZSIG(ISIG4),SUM1,INDXCS,BBIAS(IBC1),IBIAS(IC1),IBIAS(IC2))
      CALL DRAG1(GRPAR)
      CALL ESTIM1(SUM1,SUM2(ISUM2),SUM2(IDELTA),XYZ(ISTXYZ),
     .   XYZ(ISTXY0),XYZSIG(ISIG1),BBIAS,IBIAS(IB3),IBIAS(IB4),
     .   ISTANO(ISTA),ISTANO(IEST),ISTANO(ILOC),IBIAS(IB7),PARNOS,
     .   INDXCS,DENSE(IDENS0),DENSE(IDSIG))
      CALL F1(GRPAR)
      A=GRHRA1(ENV,NSTA1)
      A=OBSDT1(ENV(IENV1),ENV(IENV2),ENV(IENV3))
      CALL ORBIT1(XFIT(IFCT),XFIT(IXFIT),XI(IXI))
      CALL PREDC1(ENV(IENV1),ENV(IENV2),ENV(IENV3),PMPX0,XI(7),NPARM,
     .   NEQNMX)
      CALL PROCS1(ISTANO(ISTA),XYZ(ISTXYZ),PLHN(IP),PLHN(IL))
      CALL RESPR1(INDXCS,IBIAS(IC2),GRPAR)
      CALL STAIF1(KSUM,NSUM,NSTAT,MAXSAT,CSUM(ICSUM),CSUM(IMSUM),
     .   LBASE,CSUM(ILSUM))
      IF(LBASE.GT.0) LBASE=NSTA
```

```
      CALL TWOST1(PMPX0,NPARM)                                          CBRO 224
      CALL SORDN1(BBIAS(IBD2),DENSE(IAREA),DENSE(ICENTR),GRPAR,DENCON)  CBRO 225
      CALL PDEN2(DENSE(IPCENT),DENSE(IAREA),DENSE(IDENS2),              CBRO 226
     .   BBIAS(IBD2),DENSE(IDSIG2),SUM1,DENCON,GRPAR)                   CBRO 227
      CALL TRUEP1(XY2)                                                  CBRO 228
      CALL BSCMP1(EBIAS(IBTWA),EBIAS(IBTWD),EBIAS(IBTWB),EBIAS(IBTIME), CBRO 229
     .   SUM1,SUM2,NBIASE,EBIAS(IBESTA),EBIAS(INUMBR),EBIAS(IBETYP),    CBRO 230
     .   PARNOS)                                                        CBRO 231
      CALL VEVAL1(GRPAR)                                                CBRO 232
      CALL CLEAR(COSLAM,66,2)                                           CBRO 233
      COSLAM(2)=1.0D0                                                   CBRO 234
C CALL NONAME AND PERFORM JOB PRIME FUNCTIONS                           CBRO 235
      CALL NONAME(NPARM,PMPX0,IBIAS(IB7),IBIAS(IB3+12),BBIAS,IBIAS(IB4),CBRO 236
     .   IBIAS(IB1),IBIAS(IB2),IBIAS(IB5),IBIAS(IB6),PARNOS,            CBRO 237
     .   SUM2(ISUM2),SUM2(ITTL),SUM2(IDELTA),INDXCS(ISTNOS),            CBRO 238
     .   ISTANO(ISTA),PLPN(IN),SUM1,INDXCS,BBIAS(IBC1),IBIAS(IC1),      CBRO 239
     .   IBIAS(IC2),IBIAS(IC3),DENSE(IAREA),DENSE(ICENTR),DENCON,       CBRO 240
     .   EBIAS(IBESTA),EBIAS(IBETYP))                                   CBRO 241
      RETURN                                                            CBRO 242
      END                                                               CBRO 243
```

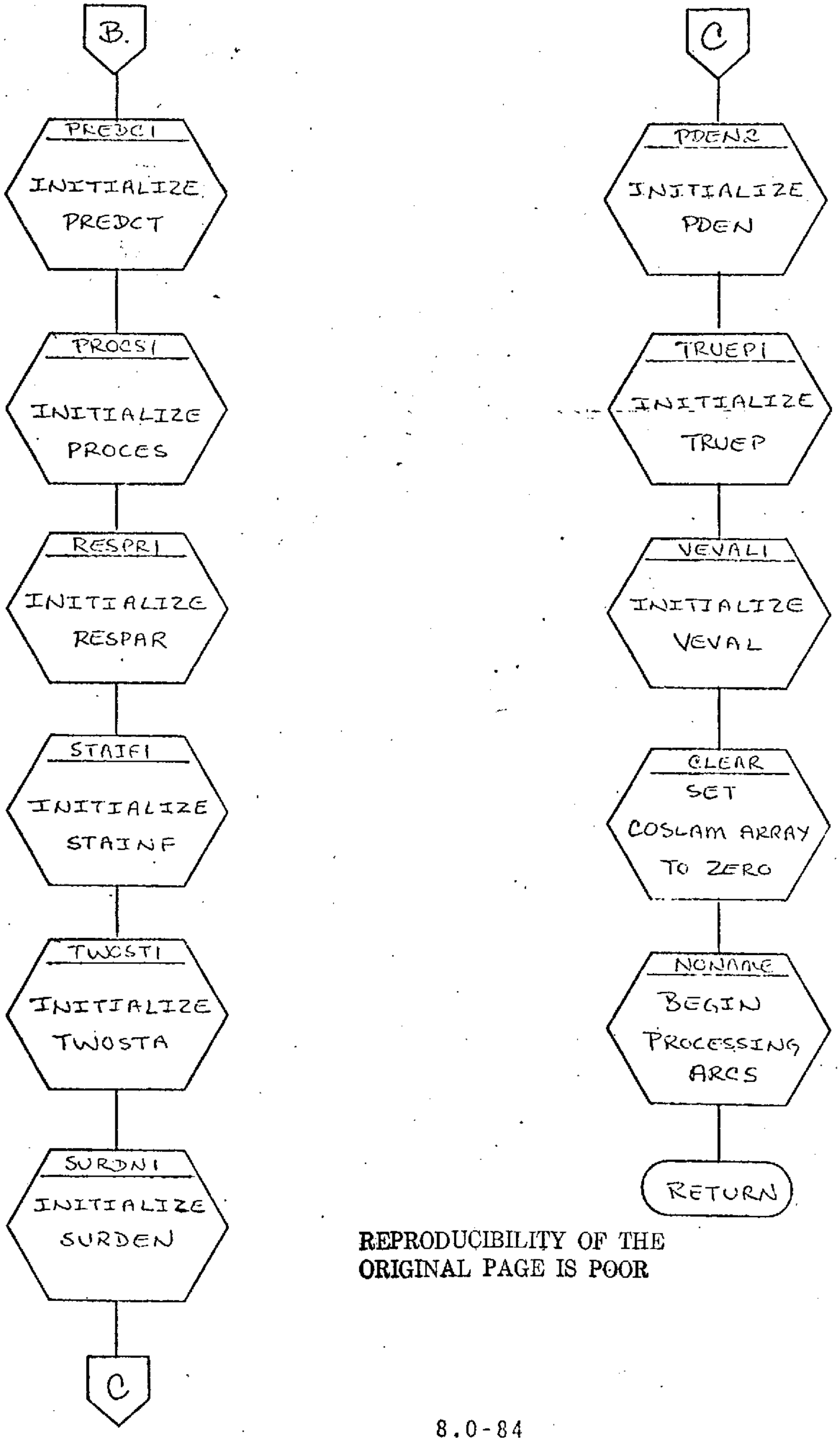
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

CBROWN


ENTRY
SET UP STARTING LOCATIONS OF ARRAYS
DATRDI
INITIALIZE DATARD
SQANTI
INITIALIZE SQUANT
COMPAR
LOAD ARRAYS OF COMMON PARAMETERS
ARCPAR
LOAD ARRAYS OF ARC PARAMETERS
COMADI
INITIALIZE COMADJ
A


A
DRAGI
INITIALIZE DRAG
ESTIMI
INITIALIZE ESTIM
FI
INITIALIZE F
GRHRAI
INITIALIZE GRHRAN
OBSDTI
INITIALIZE OBSDOT
ORBITI
INITIALIZE ORBIT
B

CBROWN



REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

PURPOSE TO ALLOCATE CORE FOR VARIABLE STORAGE ARRAYS

CALLING SEQUENCE CALL CHARLY(NCORE,ICORE,&1050)

SYMBOL	TYPE	DESCRIPTION
NCORE	I*4	NUMBER OF ARRAYS FOR WHICH CORE IS TO BE ALLOCATED
ICORE (NCORE)	I*4	SIZES IN BYTES OF ARRAYS TO BE ALLOCATED CORE
&1050	I*4	RETURN LABEL - RETURN TO THIS NUMBER IF INSUFFICIENT CORE IS AVAILABLE

SUBROUTINES USED NONE

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

```
CHARLY START 0
* STORE OFF REGISTERS
       BC    15,12(15)
       DC    X'7'
       DC    CL7'CHARLY '
       STM   14,12,12(13)
       BALR  12,0
       USING *,12
* SET UP SAVE AREA CHAIN
       LR    10,13
       LA    13,AREA
       ST    13,8(0,10)
       ST    10,4(0,13)
       BC    15,FIRST
AREA   DS    18F
* LOAD REGISTER 2 WITH THE NUMBER OF ARRAYS
FIRST  L     2,0(1)
       L     2,0(2)
* LOAD REGISTER 3 WITH THE ADDRESS OF BYTE COUNT ARRAY
       L     3,4(1)
* CONSTRUCT BYTE COUNT LIST FOR GETMAIN
       SR    6,5
       SR    6,6
LOOP   ST    5,LIST(6)
       L     7,0(3)
       ST    7,LIST(6)
       LA    6,4(6)
       LA    3,4(3)
       BCT   2,LOOP
```

```
         IC    9,=X'80'                                                 CHAR  56
         STC   9,LIST-4(6)                                              CHAR  57
* ISSUE REQUEST FOR STORAGE                                             CHAR  58
         GETMAIN LC,LA=LIST,A=ADDR                                      CHAR  59
         STC   9,ADDR-4(6)                                              CHAR  60
* RETURN IF INSUFFICIENT STORAGE AVAILABLE                              CHAR  61
         NR    15,15                                                    CHAR  62
         BC    1,RET                                                    CHAR  63
* CALL CBROWN                                                           CHAR  64
         LA    1,ADDR                                                   CHAR  65
         ST    1,LADDR                                                  CHAR  66
         LA    1,LADDR                                                  CHAR  67
         L     15,ADCON                                                 CHAR  68
         BALR  14,15                                                    CHAR  69
* RESTORE REGISTER AND RETURN                                           CHAR  70
         SR    15,15                                                    CHAR  71
RET      L     13,AREA+4                                                CHAR  72
         LM    2,12,28(13)                                              CHAR  73
         L     14,12(13)                                                CHAR  74
         MVI   12(13),X'FF'                                             CHAR  75
         BCR   15,14                                                    CHAR  76
ADCON    DC    V(CBROWN)                                                CHAR  77
LIST     DS    30F                                                      CHAR  78
LADDR    DS    F                                                        CHAR  79
ADDR     DS    30F                                                      CHAR  80
         END                                                            CHAR  81
```

NAME	CLEAR
PURPOSE	TO SET AN ARRAY TO ZERO
CALLING SEQUENCE	CALL CLEAR(IA,N,K)

SYMBOL	TYPE	DESCRIPTION
IA (1)	I	OUTPUT - ARRAY TO BE CLEARED
N	I	INPUT - FIRST DIMENSION OF ARRAY
K	I	INPUT - SECOND DIMENSION OF ARRAY

SUBROUTINES USED	NONE
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      SUBROUTINE CLEAR(IA,N,K)
      DIMENSION IA(1)
      J=N*K
C CLEAR THE ARRAY
      DO 10 I=1,J
   10 IA(I)=0
      RETURN
      END
```

NAME	CLEAR2
PURPOSE	TO SET AN ARRAY TO ZERO
CALLING SEQUENCE	CALL CLEAR2(IA,K,L)

SYMBOL	TYPE	DESCRIPTION
IA	I	OUTPUT - ARRAY TO BE CLEARED
K	I	INPUT - FIRST DIMENSION OF ARRAY
L	I	INPUT - SECOND DIMENSION OF ARRAY

SUBROUTINES USED	NONE
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      SUBROUTINE CLEAR2(IA,K,L)
      INTEGER*2 IA(1)
      J=K*L
C CLEAR THE ARRAY
      DO 10 I=1,J
   10 IA(I)=0
      RETURN
      END
```

COEF

DESCRIPTION

COEF computes the set of interpolation coefficients needed based on the order of the interpolation and the distance in units of stepsizes from the most recent time point in the array of back values.

```
NAME                COEF

PURPOSE             TO COMPUTE INTERPOLATOR COEFFICIENTS

CALLING SEQUENCE    CALL COEF(S,IORDER,A,AS)

   SYMBOL   TYPE    DESCRIPTION

   S        DP      INPUT - INTERVAL DISTANCE

   IORDER   I       INPUT - ORDER

   A        DP      OUTPUT - POSITION COEFFICIENT VECTOR
   (1)

   AS       DP      OUTPUT - VELOCITY COEFFICIENT VECTOR
   (1)

SUBROUTINES USED    COM

COMMON BLOCKS       INTERP

INPUT FILES         NONE

OUTPUT FILES        NONE

REFERENCES          GSFC X-553-70-372
                    "GEOSTAR-II  A GEOPOTENTIAL AND STATION POSITION
                     RECOVERY SYSTEM", C. E. VELEZ, G. P. BRODSKY
                                                 OCTOBER 1970
```

```
      SUBROUTINE COEF(S, IORDER,A,AS)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/INTERP/COMB(21,21),M12(4)
      DATA MAXO/15/
      DATA IOL2/0/
      DIMENSION BPZ(21),BPPZ(21),BP(22),BPP(23),B(23),A(1),AS(1)
      IF(IOL2.NE.0)GO TO 900
      CALL COM(20)
C INITIALIZE
      B(1)=1.D0
      BPZ(1)=1.D0
      BPPZ(1)=1.D0
      BP(1)=1.D0
      BPP(1)=1.D0
      DO 10 I=2,MAXO
      IL1=I-1
      SUM=0.D0
      DO 9 L=1,IL1
    9 SUM=SUM-BPZ(L)/DFLOAT(I-L+1)
   10 BPZ(I)=SUM
      DO 20 I=2,MAXO
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      SUM=0.D0
      DO 19 L=1,I
   19 SUM=SUM+BPZ(L)*BPZ(I-L+1)
   20 BPPZ(I)=SUM
C END INITIALIZATION
  900 IOL1=IORDER-1
      IOL2=IORDER-2
      DO 30 I=2,IORDER
   30 B(I)=B(I-1)*(DFLOAT(2-I)-S)/DFLOAT(1-I)
      DO 40 I=2,IORDER
      SUM=0.D0
      DO 39 L=1,I
   39 SUM=SUM+BPZ(L)*B(I-L+1)
   40 BP(I)=SUM
      DO 50 I=2,IORDER
      SUM=0.D0
      DO 49 L=1,I
   49 SUM=SUM+BPPZ(L)*B(I-L+1)
   50 BPP(I)=SUM
      FACT=1.D0
      DO 100 I=1,IOL2
      SUM=0.D0
      DO 80 L=1,IOL2
   80 SUM=SUM+FACT*COMB(L,I)*BPP(L+2)
      A(I)=SUM
  100 FACT=-FACT
      FACT=1.D0
      DO 110 I=1,IOL1
      SUM=0.D0
      DO 105 L=1,IOL1
  105 SUM=SUM+FACT*COMB(L,I)*BP(L+1)
      AS(I)=SUM
  110 FACT=-FACT
      RETURN
      END
```

NAME	COEFL
PURPOSE	TO LIST NON-ZERO GRAVITY MODEL COEFFICIENTS OF THE SPHERICAL HARMONIC EXPANSION OF THE GEOPOTENTIAL
CALLING SEQUENCE	CALL COEFL(IUNIT,NORMAL)

SYMBOL	TYPE	DESCRIPTION
IUNIT	I	INPUT - PRINTER FILE NUMBER
NORMAL	L	INPUT - SWITCH INDICATING NORMALIZED COEFFICIENTS ARE DESIRED

SUBROUTINES USED	DENORM
COMMON BLOCKS	FMODEL
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      SUBROUTINE COEFL(IUNIT,NORMAL)
      IMPLICIT REAL*8(A-H,O-Z)
      LOGICAL NORMAL
      REAL*8 CS
      COMMON/FMODEL/INDEX1, INDEX2,INDEX3,INDEX4,CS(30,33),MODEL(8)
      DIMENSION OUT1(5),OUT2(3),IND1(5),IND2(5)
      REAL*8 MODEL
      DATA CC0,CC1/1HC,1HS/
      WRITE(IUNIT,100) MODEL
C WRITE OUT ZONALS
      I=0
      DO 10 N=3,INDEX1
      NC=N-1
      IF(CS(NC,1).EQ.0.0D0) GO TO 10
      I=I+1
      OUT1(I)=CS(NC,1)
      IF(NORMAL) OUT1(I)=OUT1(I)/DENORM(NC,0)
      IND1(I)=NC
      IND2(I)=0
      IF(I.LT.5) GOTO 10
      I=0
      WRITE(IUNIT,102) (IND1(J),IND2(J),OUT1(J),J=1,5)
 10   CONTINUE
      IF(I.GT.0)WRITE(IUNIT,102) (IND1(J),IND2(J),OUT1(J),J=1,I)
C WRITE SECTORIALS & TESSERALS
      LINES=0
      I=0
      CC=CC0
      DO 20 N=3,INDEX1
      NC=N-1
```

```
      NS=32-N
      MAXC=MINO(INDEX3,N)
      DO 20 M=2,MAXC
      MS=34-M
      MT=M-1
      IF(CS(NC,M).EQ.0.D0.AND.CS(NS,MS).EQ.0.D0) GO TO 20
      I=I+1
      ANORM=1.D0
      IF(NORMAL) ANORM=1.D0/DENORM(NC,MT)
      OUT1(I)=CS(NC,M)*ANORM
      OUT2(I)=CS(NS,MS)*ANORM
      IND1(I)=NC
      IND2(I)=MT
      IF(I.LT.3) GOTO 20
      IF(MOD(LINES,30).EQ.0) WRITE(IUNIT,103) CC
      IF(MOD(LINES,5).EQ.0) WRITE(IUNIT,101)
      I=0
      CC=CC1
      LINES=LINES+1
      WRITE(IUNIT,104) (IND1(J),IND2(J),OUT1(J),OUT2(J),J=1,3)
20    CONTINUE
      IF(I.GT.0)WRITE(IUNIT,104) (IND1(J),IND2(J),OUT1(J),OUT2(J),J=1,I)
      RETURN
 100 FORMAT(1H1,14X,8A8//6X,
    .         6HZONALS/6X,6(1H-)//5(6X,5HINDEX,5X,5HVALUE,1X)/7X,
    .         5(4HN  M,12X)/1X)
101   FORMAT(1X)
 102 FORMAT(1H ,3X,5(2X,I2,I3,2X,1PE12.5,1X))
103   FORMAT(A1,5X,24HSECTORIALS AND TESSERALS/6X,10(1H-),1X,3(1H-),
    .         1X,9(1H-)//6X,3(5HINDEX,13X,
    .         5HVALUE,15X)/7X,3(4HN  M,8X,1HC,13X,1HS,11X))
 104 FORMAT(1H ,5X,3(I2,I3,2X,1PE12.5,3X,1PE12.5,4X))
      END
```

NAME	COM
PURPOSE	TO COMPUTE BINOMIAL COEFFICIENTS
CALLING SEQUENCE	CALL COM(M)

SYMBOL	TYPE	DESCRIPTION
M	I	INPUT - NUMBER OF COEFFICIENTS

SUBROUTINES USED	NONE
COMMON BLOCKS	INTERP
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      SUBROUTINE COM(M)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/INTERP/C(21,21),NDUM(4)
      M1=M+1
      DO 10 I=1,M1
      C(I,1)=1.D0
      C(I,I)=1.D0
      IF(I.LE.2) GO TO 10
      I1=I-1
      DO 5 J=2,I1
    5 C(I,J)=C(I-1,J)+C(I-1,J-1)
   10 CONTINUE
      RETURN
      END
```

NAME COMADJ

ENTRY POINT	PURPOSE
COMADI	INITIALIZATION
COMADJ	TO PRINT ADJUSTMENTS TO COMMON PARAMETERS

CALLING SEQUENCE CALL COMADI(ISTANO,ESTANO,LOC,STAXYZ,STXYZO,RLAT,
RLON,H,RLATO,RLONO,HO,NAME,PSIG,
PLHSIG,XYZNOM,PLHNOM,SUM1,INDXCS,
GPVAL,GPVALO,GPSIG)

SYMBOL	TYPE	DESCRIPTION
ISTANO (1)	I*2	INPUT - STATION NUMBERS
ESTANO (1)	I*2	INPUT - MASTER STATION NUMBERS FOR ESTIMATED STATIONS
LOC	I*2	SCRATCH
STAXYZ (3,1)	DP	INPUT & OUTPUT - CURRENT BEST STATION COORDINATES
STXYZO (3,1)	DP	INPUT - A PRIORI STATION POSITIONS
RLAT (1)	DP	INPUT - STATION LATITUDES IN RADIANS
RLON (1)	DP	INPUT - STATION LONGITUDES IN RADIANS
H (1)	DP	INPUT - STATION HEIGHT IN METERS
RLATO (3)	DP	INPUT - A PRIORI STATION LATITUDE IN RADIANS
RLONO (1)	DP	INPUT - A PRIORI STATION LONGITUDE IN RADIANS
HO (1)	DP	INPUT - A PRIORI STATION HEIGHT IN METERS
NAME (1)	DP	INPUT - STATION NAMES
PSIG (3,3,1)	R	OUTPUT - STATION COVARIANCES
PLHSIG (3,3,1)	R	INPUT & OUTPUT - STATION SPHERICAL COORDINATE COVARIANCES

XYZNOM (6,1)	R	INPUT - NOMINAL SIGMAS AND CORRELATIONS ON RECTANGULAR STATION COORDINATES
PLHNOM (6,1)	R	INPUT - NOMINAL SIGMAS AND CORRELATIONS ON SPHERICAL STATION COORDINATES
SUMI (1)	DP	INPUT - INVERTED LEAST SQUARES MATRIX IN VECTOR FORM
INDXCS (3,1)	I*2	INPUT - INDICES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
GPVAL (1)	DP	INPUT - CURRENT VALUES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
GPVALO (1)	DP	INPUT - A PRIORI VALUES FOR ADJUSTED GEOPOTENTIAL COEFFICIENTS
GPSIG (1)	DP	INPUT - SIGMAS FOR ADJUSTED GEOPOTENTIAL PARAMETERS

CALLING SEQUENCE CALL COMADJ(OUTER)

SYMBOL	TYPE	DESCRIPTION
OUTER	I	INPUT - OUTER ITERATION NUMBER

SUBROUTINES USED   PDEN   SQUANT   OUTRAD   NUMLOC

COMMON BLOCKS   CONSTS   CPARAM   FNODEL   TPEBLK

INPUT FILES   NONE

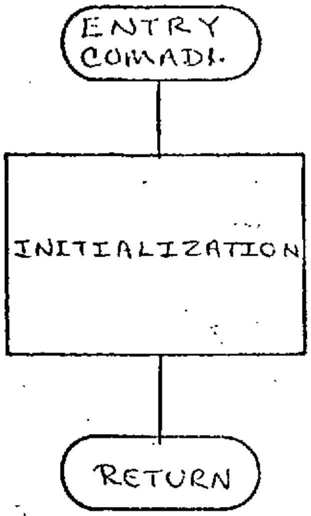
OUTPUT FILES   OUTP - PRINTER

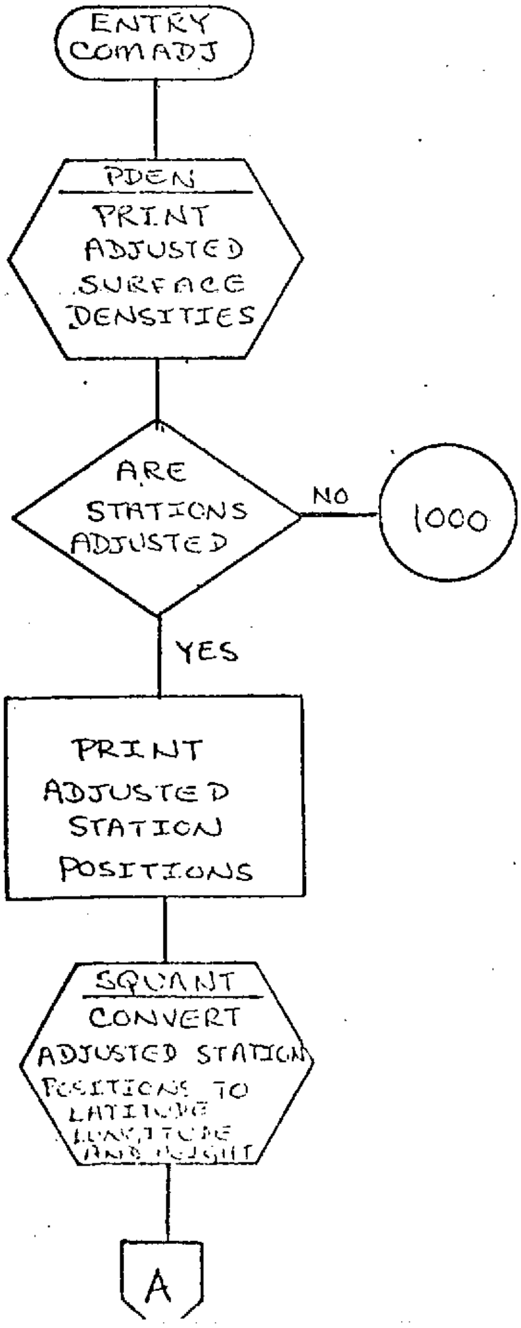
```
      SUBROUTINE COMADJ(ISTANO,ESTANO,LOC,STAXYZ,STXYZO,RLAT,RLON,H,      COMA  95
     *   RLATO,RLONO,HO,NAME,PSIG,PLHSIG,XYZNOM,PLHNOM,SUMI,INDXCS,       COMA  96
     *   GPVAL,GPVALO,GPSIG)                                              COMA  97
      IMPLICIT REAL*8 (A-H,O-Z)                                           COMA  98
      LOGICAL CMPGPF                                                      COMA  99
      INTEGER*2 ISTANO,ESTANO,LOC,INDXCS                                  COMA 100
      INTEGER OUTER,DASH,OUTP,DATP                                        COMA 101
      REAL PLHNOM,XYZNOM,PSIG,SIG,PLHSIG                                  COMA 102
      DOUBLE PRECISION NAME,MODEL                                         COMA 103
      DIMENSION IARRAY(3),SDEV(3),CORR(3),SIG(3,3),SUMI(1),ISTANO(1),     COMA 104
     *   ESTANO(1),LOC(1),STAXYZ(3,1),STXYZO(3,1),RLAT(1),RLON(1),H(1),  COMA 105
     *   RLATO(1),RLONO(1),HO(1),NAME(1),PSIG(3,3,1),PLHSIG(3,3,1),       COMA 106
     *   XYZNOM(6,1),PLHNOM(6,1),INDXCS(3,1),GPVAL(1),GPVALO(1),          COMA 107
     *   GPSIG(1),ICS(3)                                                  COMA 108
      COMMON/CONSTS/DPI,DTWOPI,DRAD,DRSEC                                 COMA 109
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,NGPCOM,      COMA 110
     *   NOBEST,CMPGPF,LIM1,LIM2,NGEN,NOCNST,NTIDST,NTIDEN,INNRSW,        COMA 111
```

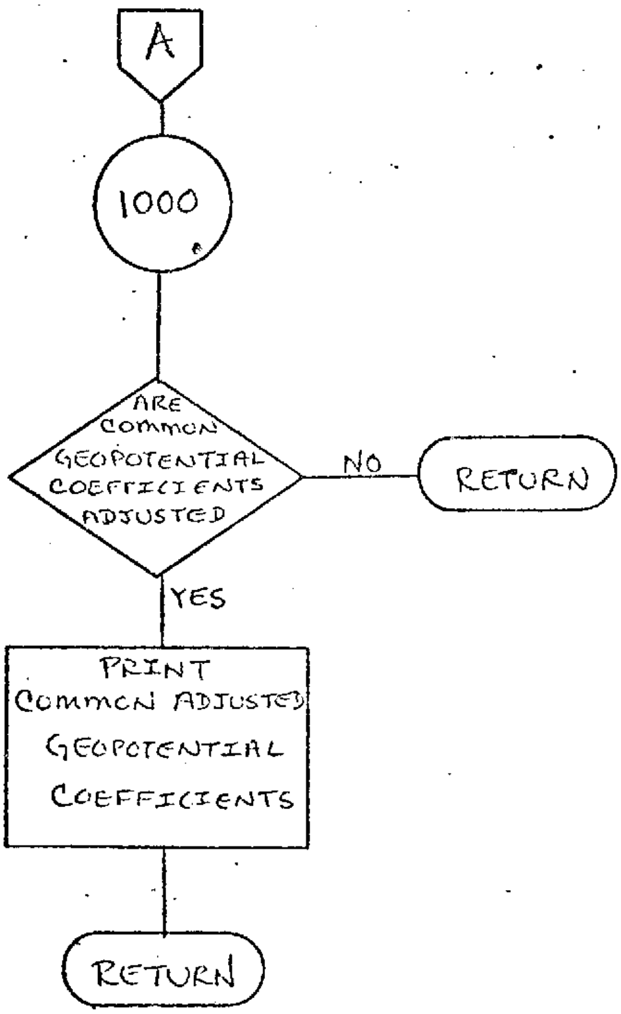
```
     .  NCONST,NDCONS
      COMMON/FMODEL/INDEX(4),CS(30,33),MODEL(8)
      COMMON/TPEBLK/INTP,OUTP,DATP(10)
      DATA IBL/1H /,DASH/1H-/,APRIOR/6HA PRIORI/,ADJUST/8HADJUSTED/
      DATA ICS/1HC,1HS/
      INDXNO(I)=NDIM*(I-1)-(I*(I-1))/2
      RETURN
      ENTRY COMADJ(OUTER)
C PRINT ADJUSTED SURFACE DENSITIES
      IF(NDENST.GT.0) CALL PDEN(OUTER)
      NSTART=NDIM-3*NMAST-NGFCON-NTIDEN+1
      IF(NSTEST.LE.0) GO TO 1000
C PRINT ADJUSTED STATION POSITIONS
      LINES=0
      INSTA=0
      IST=NDIM-3*NMAST+1
      DO 400 L=1,NMAST
      DO 100 J=1,3
      I2=INDXNO(IST)+IST
      SIG(J,J)=SUM1(I2)
      IARRAY(J)=I2
      IF(SUM1(I2).LT.0.0D0) WRITE(OUTP,3020) J,L
      SDEV(J)=DSQRT(DABS(SUM1(I2)))
  100 IST=IST+1
      I2=IARRAY(1)
      I3=IARRAY(2)
      SIG(1,2)=SUM1(I2+1)
      SIG(1,3)=SUM1(I2+2)
      SIG(2,3)=SUM1(I3+1)
      SIG(2,1)=SIG(1,2)
      SIG(3,1)=SIG(1,3)
      SIG(3,2)=SIG(2,3)
      CORI(1)=SUM1(I2+1)/(SDEV(1)*SDEV(2))
      CORI(2)=SUM1(I2+2)/(SDEV(1)*SDEV(3))
      CORI(3)=SUM1(I3+1)/(SDEV(2)*SDEV(3))
  200 INSTA=INSTA+1
      NOCON=NUMLOC(INSTA,ESTAND,NSTEST,LOC)
      IF(NOCON.EQ.0) GO TO 200
      DO 400 K=1,NOCON
      LINES=LINES+1
      IF(MOD(LINES,15).NE.1) GO TO 250
      WRITE(OUTP,3000) OUTER
      WRITE(OUTP,3010)
  250 I=LOC(K)
      DO 300 J=1,3
      DO 300 M=1,3
  300 PSIG(J,M,I)=SIG(J,M)
      WRITE(OUTP,4000)
      WRITE(OUTP,3030) APRIOR,NAME(I),ISTANO(I),(STXYZO(J,I),J=1,3),
     .   (XYZACS(J,I),J=1,6)
  400 WRITE(OUTP,3030) ADJUST,NAME(I),ISTANO(I),(STAXYZ(J,I),J=1,3),
     .   SDEV,CORI
      CALL SQUANT(NSTEST,NSTEST,.FALSE.)
      LINESX=0
      PUNCH 3050,OUTER
      DO 500 I=1,NSTEST
```

```
      LINESX=LINESX+1
      DO 500 J=1,3
  500 PLHSIG(J,J,I)=SQRT(PLHSIG(J,J,I))
      DO 700 J=1,2
      J1=J+1
      DO 600 K=J1,3
  600 PLHSIG(J,K,I)=PLHSIG(J,K,I)/(PLHSIG(J,J,I)*PLHSIG(K,K,I))
  700 PLHSIG(J,J,I)=PLHSIG(J,J,I)/DRSEC
      IF(MOD(LINESX,16).NE.1) GO TO 800
      IF(LINES.GT.7) WRITE(OUTP,3000) OUTER
      WRITE(OUTP,3040)
  800 CALL OUTRAD(RLATO(I),LATD,LATM,SLAT,1)
      CALL OUTRAD(RLONO(I),LOND,LONM,SLON,1)
      WRITE(OUTP,4000)
      WRITE(OUTP,3050) APRIOR,NAME(I),ISTANO(I),LATD,LATM,SLAT,LOND,
     .   LONM,SLON,HO(I),(PLHNOM(J,I),J=1,5)
      CALL OUTRAD(RLAT(I),LATD,LATM,SLAT,1)
      CALL OUTRAD(RLON(I),LOND,LONM,SLON,1)
      J=IBL
      IF(RLAT(I).LT.0.0D0) J=DASH
      LATDP=IABS(LATD)
      LATMP=IABS(LATM)
      SLATP=DABS(SLAT)
      PUNCH 3060,NAME(I),ISTANO(I),J,LATDP,LATMP,SLATP,LOND,LONM,
     .   SLON,H(I)
  900 WRITE(OUTP,3050) ADJUST,NAME(I),ISTANO(I),LATD,LATM,SLAT,LOND,
     .   LONM,SLON,H(I),(PLHSIG(J,J,I),J=1,3),(PLHSIG(1,K,I),K=2,3),
     .   PLHSIG(2,3,I)
 1000 IF(NGPCOM.LE.0) RETURN
C PRINT ADJUSTED GEOPOTENTIAL COEFFICIENTS
      NGP1=NCSEST-NGPCOM+1
      I1=0
      DO 1200 I=NGP1,NCSEST
      I1=I1+1
      IF(MOD(I1,43).EQ.1) WRITE(OUTP,3070) OUTER
      IF(MOD(I1,5).EQ.1) WRITE(OUTP,4000)
      J=INDXCS(1,I)
      N=INDXCS(2,I)
      M=INDXCS(3,I)
      C1=GPVAL(I)*GPSIG(I)
      C2=GPVALO(I)*GPSIG(I)
      IF(J.EQ.1) CS(N,M+1)=C1
      IF(J.EQ.2) CS(31-N,33-M)=C1
      K=INDXNC(NSTART-NGP1+I)+NSTART-NGP1+I
      SIG1=DSQRT(SUM1(K))
      SIG2=GPSIG(I)*SIG1
 1200 WRITE(OUTP,3030) ICS(J),N,M,C2,C1,SIG1,SIG2
      RETURN
 3000 FORMAT(1H1,35X,40HSTATION POSITION ADJUSTMENT SUMMARY FOR ,
     .        16HITERATION NUMBER I3 /)
 3010 FORMAT(1H0/5X,35HEARTH FIXED RECTANGULAR COORDINATES,36X,
     .        18HSTANDARD DEVIATION,11X,11HCORRELATION//21X,7HSTATION,
     .        12X,1HX,12X,1HY,12X,1HZ,11X,1HX,7X,1HY,7X,1HZ,5X,3HX-Y,
     .        5X,3HX-Z,5X,3HY-Z/19X,11HNAME NUMBER,1X,3(8X,3H(M),2X),2X,
     .        3(5X,3H(M)))
 3020 FORMAT('0*********** NEGATIVE ARGUMENT TO DSQRT FOR COMPONENT',
```

```
                                                                      COMA 224
     .   12,' CF MASTER STATION',I3,' $**********'/)                  COMA 225
3030 FORMAT(1H ,5X,A8,4X,A6,I6,3X,3F12.2,1X,3F8.2,3F8.4)             COMA 226
3040 FORMAT(1H0/5X,20HGEODETIC COORDINATES,52X,18HSTANDARD DEVIATION, COMA 227
     .          10X,11HCORRELATION//21X,7HSTATION,3X,17HGEODETIC LATITUDE, COMA 228
     .          2X,14HEAST LONGITUDE,2X,6HHEIGHT,6X,3HLAT,5X,3HLON,6X,2HHT,COMA 229
     .          3X,7HLAT-LON,2X,6HLAT-HT,2X,6HLON-HT/19X,11HNAME NUMBER, COMA 230
     .          2(3X,13HDEG MN SECOND,1X),9H (METERS),2X,2(2X,5H(SEC),1X), COMA 231
     .          3X,3H(M))                                             COMA 232
3050 FORMAT(1H ,5X,A8,4X,A6,I6,2(I6,I3,F7.3,1X),F9.2,2F8.3,F8.2,3F8.4) COMA 233
3060 FORMAT(A6,I4,A1,2I2,F10.7,I3,I2,F10.7,F10.3)                    COMA 234
3070 FORMAT(1H1,31X,38HGEOPOTENTIAL COEFFICIENTS ADJUSTED FOR,       COMA 235
     .     15H OUTER ITERATION,I2//20X,                              COMA 236
     .     11HCOEFFICIENT,19X,5HVALUE,21X,18HSTANDARD DEVIATION/41X, COMA 237
     .     8HA PRIORI,7X,8HADJUSTED,6X,25HRATIO TO A PRIORI   ADJUSTED) COMA 238
3080 FORMAT(22X,A1,1H(,I2,1H,,I2,1H),1PD21.8,D15.5,1X,0P2D16.4)      COMA 239
3090 FORMAT('ADJUSTED STATION COORDINATES FOR OUTER',I3)             COMA 240
4000 FORMAT(1X)                                                      COMA 241
     END
```







REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME COMPAR

PURPOSE TO LOAD COMMON PARAMETERS INTO VARIABLE STORAGE ARRAYS

CALLING SEQUENCE CALL COMPAR(GPVALO,GPSIG,GPNO,GPVAL,RLAT,RLON,H,
NAME,RLATO,RLONO,HO,ISTANO,ESTANO,
INDXCS,TTL,XYZSIG,PLHSIG,STASIG,
PLHNCM,XYZNOV,STAXYZ,STXYZO,STANOS,
LOC,DSIG,DENSO,DENS,AREA,CENTER,
BCENTR)

SYMBOL	TYPE	DESCRIPTION
GPVALO (1)	DP	OUTPUT - A PRIORI VALUES FOR ADJUSTED GEOPOTENTIAL COEFFICIENTS
GPSIG (1)	DP	OUTPUT - SIGMAS FOR ADJUSTED GEOPOTENTIAL PARAMETERS
GPNO (1)	DP	OUTPUT - LOCATIONS OF GEOPOTENTIAL PARAMETERS IN NORMAL MATRIX
GPVAL (1)	DP	INPUT - CURRENT VALUES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
RLAT (1)	DP	INPUT - STATION LATITUDE IN RADIANS
RLON (1)	DP	INPUT - STATION LONGITUDE IN RADIANS
H (1)	DP	INPUT - STATION HEIGHT IN METERS
NAME (1)	DP	INPUT - STATION NAMES
RLATO (1)	DP	OUTPUT - A PRIORI STATION LATITUDE IN RADIANS
RLONO (1)	DP	OUTPUT - A PRIORI STATION LONGITUDE IN RADIANS
HO (1)	DP	OUTPUT - A PRIORI STATION HEIGHT IN METERS
ISTANO (1)	I*2	INPUT - STATION NUMBERS
ESTANO (1)	I*2	INPUT & OUTPUT - MASTER STATION NUMBER TO WHICH ADJUSTED STATIONS ARE CONSTRAINED
INDXCS (3,1)	I*2	OUTPUT - INDICES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS

TTL (1)	DP	OUTPUT - ADJUSTED PARAMETER TITLE ARRAY
XYZSIG (3,3,1)	R	OUTPUT - STATION RECTANGULAR COORDINATE COVARIANCES
PLHSIG (3,3,1)	R	OUTPUT - STATION SPHERICAL COORDINATE COVARIANCES
STASIG (3,3,1)	R	OUTPUT - SIGMAS AND COVARIANCES OF ADJUSTED STATION COORDINATES
PLHNOM (6,1)	R	OUTPUT - NOMINAL SIGMAS AND CORRELATIONS ON ADJUSTED SPHERICAL STATION COORDINATES
XYZNOM (6,1)	R	OUTPUT - NOMINAL SIGMAS AND CORRELATIONS ON ADJUSTED RECTANGULAR STATION COORDINATES
STAXYZ (3,1)	DP	INPUT - TRACKING STATION CARTESIAN COORDINATES
STXYZ0 (3,1)	DP	OUTPUT - A PRIORI STATION POSITIONS
STANOS (3,1)	I*2	OUTPUT - LOCATIONS IN NORMAL MATRIX OF INFORMATION PERTAINING TO ADJUSTED STATION COORDINATES
LOC (1)	I*2	SCRATCH
DSIG (1)	DP	OUTPUT - SIGMAS OF ADJUSTED SURFACE DENSITY
DENS0 (1)	DP	OUTPUT - A PRIORI VALUES OF ADJUSTED SURFACE DENSITIES
DENS (1)	DP	OUTPUT - SURFACE DENSITY VALUES
AREA (1)	DP	OUTPUT - SURFACE DENSITY SUB-BLOCK AREAS
CENTER (1)	DP	OUTPUT - THE GEOCENTRIC COORDINATES OF THE SUB-BLOCK CENTERS
BCENTR (1)	DP	OUTPUT - THE LATITUDE AND LONGITUDE OF THE ADJUSTED SURFACE DENSITY BLOCK

SUBROUTINES USED	STAINP VCONV	INDENT SYMINV	STORE OUTRAD	SQUANT NUMBR2	PLHOUT NUMLOC
COMMON BLOCKS	CONOUT TPDBLK	CONSTS VRBLOK	CPARAM	FLXBLK	FMODEL
INPUT FILES	NONE				

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

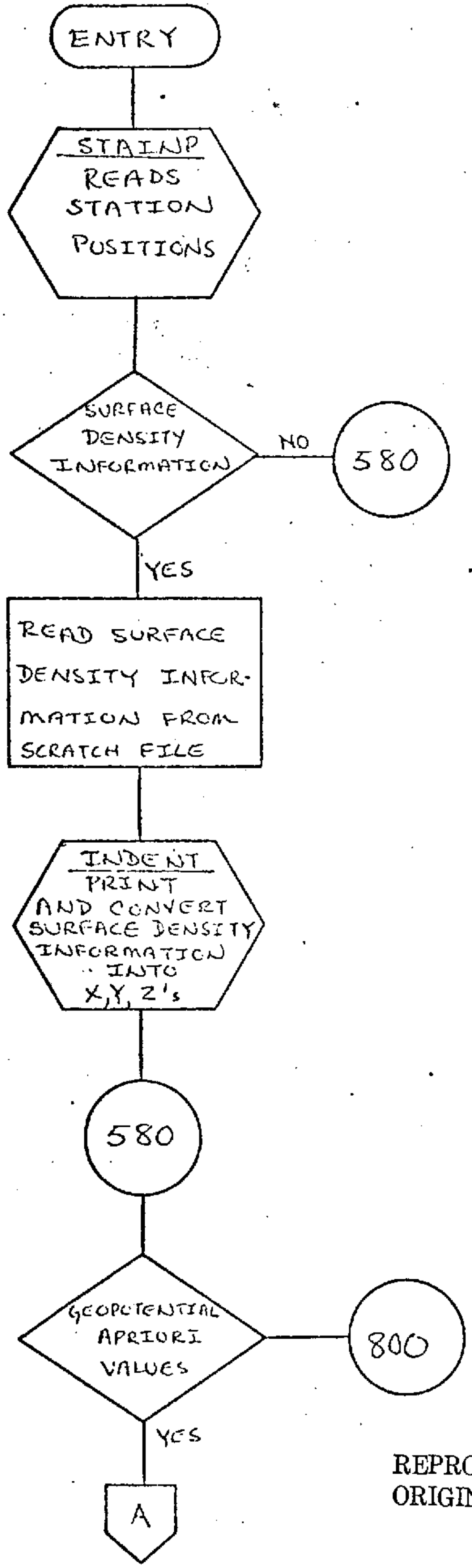
```
OUTPUT FILES      OUTP - PRINTER


      SUBROUTINE COMPAR(GPVALO,GPSIG,GPNO,GPVAL,RLAT,RLON,H,NAME,
     *  RLATO,RLONO,HO,ISTANO,ESTANO,INDXCS,TTL,XYZSIG,PLHSIG,STASIG,
     *  PLHNOM,XYZNOM,STAXYZ,STXYZO,STANOS,LOC,DSIG,DENSO,DENS,AREA,
     *  CENTER,BCENTR)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL*1 PLHSW
      LOGICAL CMPGPR
      INTEGER*2 GPNO,ISTANO,ESTANO,INDXCS,STANOS,NDEXCS,LOC
      INTEGER CLTP,DATP,SCRC,FLTP
      REAL XYZSIG,PLHSIG,STASIG,PLHNOM,XYZNOM,PSIG,GPSIGN,TWO1
      DOUBLE PRECISION NAME,MODEL
      DIMENSION GPVALO(1),GPSIG(1),GPVAL(1),RLAT(1),RLON(1),H(1),
     *  NAME(1),RLATO(1),RLONO(1),HO(1),ISTANO(1),ESTANO(1),LOC(1),
     *  INDXCS(3,1),TTL(1),XYZSIG(3,3,1),PLHSIG(3,3,1),STASIG(3,3,1),
     *  PLHNOM(6,1),STAXYZ(3,1),STXYZO(3,1),STANOS(3,1),SUM1(6),
     *  SUM2(3),TWO1(2),XYZNOM(6,1),GPNO(1),AXYZ(6),DSIG(1),DENSO(1),
     *  DENS(1),AREA(1),CENTER(1),BCENTR(1),ISURF(1),JSURF(1)
      COMMON/CONOUT/IG1(7),NSTART,NEONMX(4),NSTARO,LSTART(6)
      COMMON/CONSTS/DPI,DTWOPI,DRAD,RSEC
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,NBIAS,NGPC1,NGPC2,NGPCOM,
     *  NCSEST,CMPGPR,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,INNRSW,
     *  NCONST,NDCONS
      COMMON/FLXBLK/NDEXCS(360,3),PSIG(6,381),PLHSW(1290)
      COMMON/FMODEL/INDEX(4),CS(30,33),MODEL(9)
      COMMON/TPEBLK/INTP,OUTP,DATP(7),SCRC,FLTP(2)
      COMMON/VFBLOK/GPSIGN(2250)
      DATA AXYZ/3H  X,3H  Y,3H  Z,5H XDOT,5H YDOT,5H ZDOT/,BL/1H /
      DATA CWO/5HCCC(0/,SWD/5HSCD00/
      EQUIVALENCE (ISURF(1),NDEXCS(1,1)),(JSURF(1),GPSIGN(1))
      EQUIVALENCE (TWO,TWO1(1))
      NSTART=NDIM-3*NMAST-NGPCOM-NTIDEN+1
C READ STATION POSITIONS
      CALL STAINP(NSTA,ISTANO,NAME,RLAT,RLON,H,NSTARO,NSTEST,ESTANO)
      END FILE SCRC
      REWIND SCRC
      IF(NDEN.LE.0) GO TO 590
C READ SURFACE DENSITY INFORMATION FROM SCRATCH FILE
      DO 700 I1=1,4050,450
      I2=MINO(I1+449,4050)
  700 READ(SCRC) (ISURF(I),I=I1,I2)
      DO 750 I1=1,2026,450
      I2=MINO(I1+449,2026)
  750 READ(SCRC) (JSURF(I),I=I1,I2)
C PRINT AND CONVERT SURFACE DENSITY INFORMATION INTO X, Y, Z'S
      CALL INDENT(DSIG,DENSO,DENS,AREA,CENTER,BCENTR)
  590 IF(NGPCOM.LE.0) GO TO 800
C READ GEOPOTENTIAL A PRIORI VALUES FROM SCRATCH FILE
      NGPA=NCSEST-NGPCOM
      DO 600 I1=1,NGPCOM,225
      I2=MINO(NGPCOM,I1+224)
  600 READ(SCRC) (GPVAL(I),I=I1,I2)
```

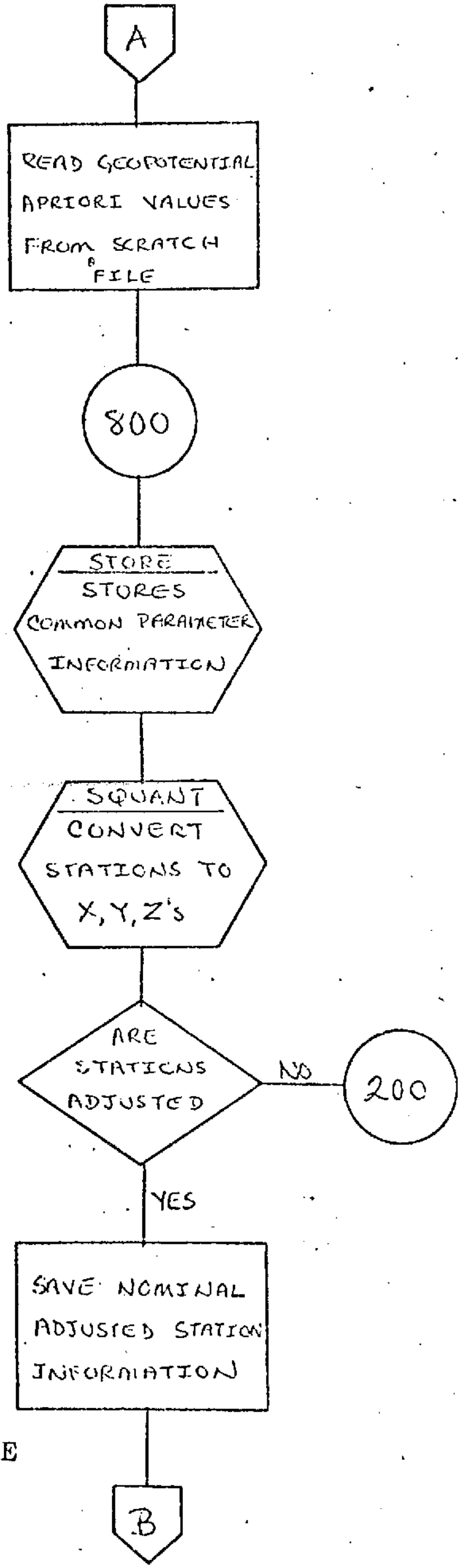
```
      DO 650 I=1,NGPCOM
      I1=I+NGPA
  650 GPVALO(I1)=GPVAL(I)
C OBTAIN STORED COMMON PARAMETER INFORMATION
  800 CALL STORE(.TRUE.,.TRUE.)
C CONVERT STATIONS TO X, Y, Z'S
      IF(NSTA.GT.0) CALL SQUANT(NSTA,NSTEST,.TRUE.)
      IF(NDIM.LE.0) GO TO 5
      DO 1 I=1,NDIM
      TTL(I)=BL
      IF(I.LT.7) TTL(I)=AXYZ(I)
    1 CONTINUE
    5 IF(NSTEST.EQ.0) GO TO 200
C SAVE NOMINAL ADJUSTED STATION INFORMATION
      DO 20 I=1,NSTEST
      STXYZ0(1,I)=STAXYZ(1,I)
      STXYZ0(2,I)=STAXYZ(2,I)
      STXYZ0(3,I)=STAXYZ(3,I)
      RLATO(I)=RLAT(I)
      RLONO(I)=RLON(I)
      HO(I)=H(I)
      J=ESTANO(I)
      ESTANO(I)=NUMER2(J,ISTANO,NSTEST)
      IF(PLHSW(I)) GO TO 10
      STASIG(1,1,I)=PSIG(1,I)
      STASIG(2,2,I)=PSIG(2,I)
      STASIG(3,3,I)=PSIG(3,I)
      STASIG(1,2,I)=PSIG(4,I)
      STASIG(1,3,I)=PSIG(5,I)
      STASIG(2,3,I)=PSIG(6,I)
      GO TO 20
   10 PLHSIG(1,1,I)=PSIG(1,I)
      PLHSIG(2,2,I)=PSIG(2,I)
      PLHSIG(3,3,I)=PSIG(3,I)
      PLHSIG(1,2,I)=PSIG(4,I)
      PLHSIG(1,3,I)=PSIG(5,I)
      PLHSIG(2,3,I)=PSIG(6,I)
   20 CONTINUE
      DO 120 LL=1,2
      DO 120 I=1,NSTEST
      I1=ESTANO(I)
      IF(I1.EQ.I) GO TO 40
      IF(LL.EQ.1) GO TO 120
      DO 30 J=1,3
      DO 30 K=1,3
   30 STASIG(J,K,I)=STASIG(J,K,I1)
      GO TO 90
   40 IF(LL.EQ.2) GO TO 120
      DO 60 J=1,2
      DO 60 K=2,3
      IF(J.EQ.K) GO TO 60
      IF(PLHSW(I)) GO TO 50
      STASIG(J,K,I)=STASIG(J,K,I)*STASIG(J,J,I)*STASIG(K,K,I)
      STASIG(K,J,I)=STASIG(J,K,I)
      GO TO 60
   50 PLHSIG(J,K,I)=PLHSIG(J,K,I)*PLHSIG(J,J,I)*PLHSIG(K,K,I)
```

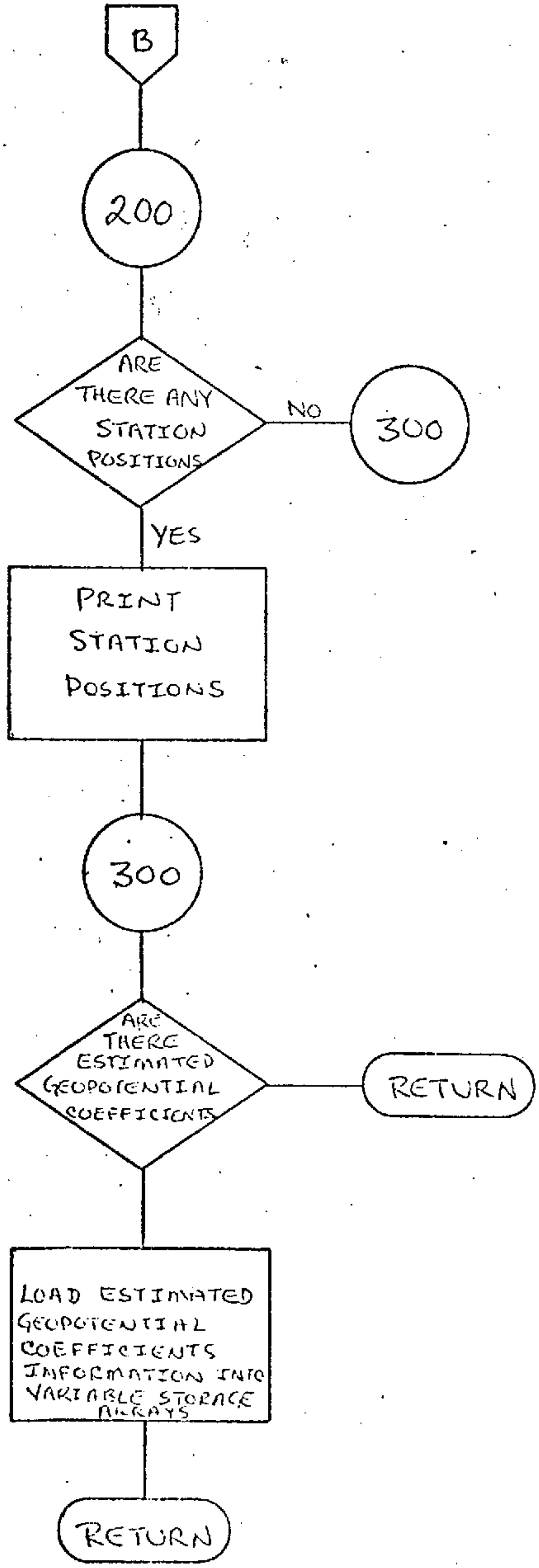
```
      PLHSIG(K,J,I)=PLHSIG(J,K,I)
   60 CONTINUE
      DO 80 J=1,3
      IF(PLHSW(I)) GO TO 70
      STASIG(J,J,I)=STASIG(J,J,I)**2
      GO TO 80
   70 PLHSIG(J,J,I)=PLHSIG(J,J,I)**2
   80 CONTINUE
      IF(PLHSW(I)) GO TO 100
   90 CALL PLHCUT(STAXYZ(1,I),STASIG(1,1,I),PLHSIG(1,1,I),P1,P2,P3)
      GO TO 120
  100 CALL VCONV(PLHSIG(1,1,I),STASIG(1,1,I),XYZSIG(1,1,I))
  120 CONTINUE
      DO 150 I=1,NSTEST
      DO 130 J=1,3
      XYZNOM(J,I)=SQRT(STASIG(J,J,I))
  130 PLHNOM(J,I)=SQRT(PLHSIG(J,J,I))
      DO 140 J=1,3
      J1=J/3+1
      J2=(J+4)/2
      XYZNOM(J+3,I)=STASIG(J1,J2,I)/(XYZNOM(J1,I)*XYZNOM(J2,I))
  140 PLHNOM(J+3,I)=PLHSIG(J1,J2,I)/(PLHNOM(J1,I)*PLHNOM(J2,I))
      DO 150 J=1,2
  150 PLHNOM(J,I)=PLHNOM(J,I)/RSEC
      DO 170 I=1,NSTEST
      I1=1
      DO 160 J=1,3
      DO 160 K=J,3
      SUM1(I1)=STASIG(K,J,I)
  160 I1=I1+1
      CALL SYMINV(SUM1,3,2,SUM2)
      I1=1
      DO 170 J=1,3
      DO 170 K=J,3
      STASIG(J,K,I)=SUM1(I1)
      STASIG(K,J,I)=SUM1(I1)
  170 I1=I1+1
      NSTART=NDIM-3*NMAST
      DO 175 I=1,NSTEST
      STANOS(1,I)=0
      STANOS(2,I)=0
  175 STANOS(3,I)=0
      LL=0
      DO 190 I=1,NMAST
  180 LL=LL+1
      NOCON=NUMLOC(LL,ESTANO,NSTEST,LOC)
      IF(NOCON.EQ.0) GO TO 180
      DO 185 K=1,NOCON
      K1=LOC(K)
      DO 185 J=1,3
  185 STANOS(J,K1)=3*(I-1)+J+NSTART
      TTL(NSTART+3*(I-1)+2)=NAME(LL)
  190 CONTINUE
      NSTART=NSTART+1
  200 IF(NSTA.LE.0) GO TO 300
C PRINT STATION POSITIONS
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      LINES=0
      DO 250 I=1,NSTA
      LINES=LINES+1
      IF(MOD(LINES,45).EQ.1) WRITE(OUTP,10000)
      CALL OUTRAD(RLAT(I),LATD,LATM,SLAT,1)
      CALL OUTRAD(RLON(I),LOND,LONM,SLON,1)
      IF(MOD(LINES,5).EQ.1) WRITE(OUTP,10100)
      WRITE(OUTP,10200) NAME(I),ISTANO(I),LATD,LATM,SLAT,LOND,LONM,SLON,
     .   H(I)
      IF(I.GT.NSTEST) GO TO 250
      WRITE(OUTP,10300) (PLHNEM(J,I),J=1,3)
      WRITE(OUTP,10400)
  250 CONTINUE
  300 NGP=NDIM-NGPCOM-3*NMAST-NTIDEN
      NSTART=NGP+1
      IF(NGPCOM.LE.0) RETURN
C LOAD ESTIMATED GEOPOTENTIAL COEFFICIENT INFORMATION INTO VARIABLE
C STORAGE ARRAYS
      NGP1=NCSEST-NGPCOM
      DO 500 I=1,NGPCOM
      ICS=NDEXCS(I,1)
      N=NDEXCS(I,2)
      M=NDEXCS(I,3)
      I1=I+NGP1
      INDXCS(1,I1)=ICS
      INDXCS(2,I1)=N
      INDXCS(3,I1)=M
      M=M+1
      ITTL=NGP+I
      GPNO(I1)=ITTL
      TWD=CWD
      IF(ICS.EQ.2) TWD=SWD
      J=N/10
      I1=J*65536
      J=N-10*J
      I1=I1+J*256
      J=(M-1)/10
      I1=I1+J
      J=16777216*(M-1-10*J)
      TWD1(1)=OR(TWD1(1),I1)
      TWD1(2)=OR(TWD1(2),J)
      SIG=GPSIGN(I)
      GPSIG(I1)=SIG
      TTL(ITTL)=TWD
      IF(ICS.LT.2) GO TO 500
      N=31-N
      M=34-M
  500 GPVAL(I1)=CS(N,M)/SIG
      RETURN
10000 FORMAT(1H1,42X,19HTRACKING COMPLEMENT/,1H0,12X,7HSTATION,6X,
     1   17HGEODETIC LATITUDE,6X,14HEAST LONGITUDE,6X,9HSPHEROID ,
     2   5HHEIGHT/1H0,10X,11HNAME NUMBER,5X,2(19HDEG MN SECONDS,8X),
     3   3X,8H(METERS))
10100 FORMAT(1X)
10200 FORMAT(10X,A7,I5,5X,2(2I3,F7.3,9X),F11.3)
10300 FORMAT(1H+,30X,1H*,F6.1,17X,1H*,F6.1,15X,1H*,F5.1)
10400 FORMAT(1H+,40X,1H_,21X,1H_,19X,1H_)
      END
```


ENTRY
STAINP
READS STATION POSITIONS
SURFACE DENSITY INFORMATION
NO
580
YES
READ SURFACE DENSITY INFORMATION FROM SCRATCH FILE
INDENT
PRINT AND CONVERT SURFACE DENSITY INFORMATION INTO X,Y,Z's
580
GEOPOTENTIAL APRIORI VALUES
800
YES
A


A
READ GEOPOTENTIAL APRIORI VALUES FROM SCRATCH FILE
800
STORE
STORES COMMON PARAMETER INFORMATION
SQUANT
CONVERT STATIONS TO X,Y,Z's
ARE STATIONS ADJUSTED
NO
200
YES
SAVE NOMINAL ADJUSTED STATION INFORMATION
B


B
200
ARE THERE ANY STATION POSITIONS
NO
300
YES
PRINT STATION POSITIONS
300
ARE THERE ESTIMATED GEOPOTENTIAL COEFFICIENTS
RETURN
LOAD ESTIMATED GEOPOTENTIAL COEFFICIENTS INFORMATION INTO VARIABLE STORAGE ARRAYS
RETURN

NAME	CONSTS
PURPOSE	BLOCKDATA STORAGE OF INPUT/OUTPUT FILE NUMBERS, INTEGRATOR STEPSIZE INFORMATION, CONVERSION CONSTANTS, AND EARTH PARAMETERS
COMMON BLOCKS	CONSTS CPARAM CONOUT INTBLK CGEOS MONTHS VRBLOK TPEBLK

```
      BLOCK DATA
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL TCREFT,VARSTP,NORRAT,INITAL,HLVDSW
      INTEGER XYZTP,RVTP,SCRA,SCRC,FLTP,PLOTP,DATP,OUTP,GRDTP,ORDER
      DOUBLE PRECISION LOVE
      COMMON/CONSTS/DPI,DTWOPI,DRAD,DRSEC
      COMMON/CPARAM/NSTA(19)
      COMMON/CONOUT/NARCS(19)
      COMMON/INTBLK/THDOT1,THDOT2(2),GM,AE,AESQ,FINV,FS032(18),RPRESS,
     .   INITAL,NORRAT,THETG,BODIES(6),STEPSZ(4),HLVERB(2),DBLERB(2),
     .   CTOL(2),RTOL(2),STPLOW(2),STEPUP(2),ORDER(4),ASAT(4),VARSTP(2),
     .   HLVDSW(2),NEON(8),LOVE(3),TCREFT,NBODY
      COMMON/CGEOS/ISATID(2),THETGO(18),IG6(4,17)
      COMMON/MONTHS/MONTH(26)
      COMMON/VRBLOK/COSLAM(66),IG(21,18)
      COMMON/TPEBLK/INTP,OUTP,DATP,XYZTP,KEPTAP,RVTP,PLOTP,IOBS,SCRA,
     .   SCRC,FLTP,GRDTP
      DATA STEPSZ/4*1.0D+2/,         HLVERB/2*0.25D-4/,
     .     DBLERB/2*0.25D-10/,       CTOL/2*0.25D-10/,
     .     RTOL/2*0.25D-7/,          STPLOW/2*12.5D0/,
     .     STEPUP/2*4.0D2/,          ORDER/4*11/,
     .     VARSTP/2*.FALSE./,        HLVDSW/2*.TRUE./,
     .     TCREFT/.FALSE./,          NORRAT/.FALSE./,
     .     FS032,ASAT/21*0.0D0/,     NEON/8*0/
      DATA INTP,OUTP,DATP,XYZTP,KEPTAP,RVTP,PLOTP,IOBS,SCRA,SCRC,
     .     FLTP,GRDTP/5,6,12,8,9,0,19,20,14,16,13,18/
      DATA MONTH/0,31,60,91,121,152,182,213,244,274,305,335,366,
     .            0,31,59,90,120,151,181,212,243,273,304,334,365/
      DATA NSTA/19*0/
      DATA NARCS/19*0/
      DATA NBODY/2/
C GRAVITATIONAL CONSTANT TIMES MASS OF EARTH IN METERS**3/SECONDS**2
      DATA GM/3.986013000D+14/,
C SEMI-MAJOR AXIS OF REFERENCE ELLIPSOID IN METERS
     .          AE/6378155.D0/,
C INVERSE OF FLATTENING OF REFERENCE ELLIPSOID
     .          FINV/298.255D0/,
C RATIOS OF LUNAR, SOLAR, PLANETARY MASSES TO MASS OF EARTH
C ...MOON
     .     BODIES/0.1229997171006607D-01,
C ...SUN
     .                0.3329455619284376D+06,
C ...VENUS
```

```
     .          0.8150003229334958D+00,
C ...MARS
     .          0.1074468525270073D+00,
C ...JUPITER
     .          0.3178809303D+03,
C ...SATURN
     .          0.9514905175D+02/
C RIGHT ASCENSION OF GREENWICH AT JAN 0.0 FOR 1958-1975 IN DEGREES
      DATA THETO/99.15144170D0,98.91272590D0,98.67400650D0,
     .            99.42093470D0,99.18221650D0,98.94349860D0,
     .            98.70478250D0,99.45171170D0,99.21299360D0,
     .            98.97427650D0,98.73555950D0,99.48248960D0,
     .            99.24377500D0,99.0050541067D0,98.7663375D0,
     .            99.5132708333D0,99.2745541066D0,99.0358375000D0/,
C MEAN ADVANCE IN RT ASC OF GREENWICH PER MEAN SOLAR DAY IN DEGREES
     .     THDOT1/.9856473354D0/,
C SOLAR RADIATION PRESSURE IN NEWTONS/METER**2
     .     RPRESS/.45D-5/
C PI IN RADIANS (DOUBLE PRECISION)
      DATA DPI    /3.1415926535897932D0/,
C TWO PI IN RADIANS (DOUBLE PRECISION)
     .     DTWOPI /6.2831853071795864D0/,
C CONVERSION FROM DEGREES TO RADIANS (DOUBLE PRECISION)
     .     DRAD   /.017453292519943296D0/,
C CONVERSION FROM SECONDS OF ARC TO RADIANS (DOUBLE PRECISION)
     .    DRSEC/.4848136811095360D-5/
C CONSTANTS FOR SOLID EARTH TIDES
C          K2
      DATA LOVE/0.29D0,
C          K3
     .       0.0D0,
C          LAMBDA (PHASE ANGLE - DEGREES)
     .       0.0D0/
      DATA COSLAM/0.0D0,1.0D0,64*0.0D0/
      END
```

NAME CORREL

PURPOSE TO COMPUTE AND PRINT CORRELATION COEFFICIENTS FROM THE DIAGONAL AND ABOVE THE DIAGONAL OF A NORMAL MATRIX IN VECTOR FORM

CALLING SEQUENCE CALL CORREL(SUM1,NPARAM,NDIM,INNER,TTL)

SYMBOL	TYPE	DESCRIPTION
SUM1 (1)	DP	INPUT - INVERTED LEAST SQUARES MATRIX IN VECTOR FORM
NPARAM	I	INPUT - ORDER OF LEAST SQUARES MATRIX
NDIM	I	INPUT - FIRST DIMENSION OF SQUARE ARRAY EQUIVALENT TO 'SUM1' VECTOR
INNER	I	INPUT - ITERATION NUMBER
TTL (1)	DP	INPUT - CORRELATION PARAMETER TITLE ARRAY

SUBROUTINES USED NONE

COMMON BLOCKS TPEBLK

INPUT FILES NONE

OUTPUT FILES OUTP - PRINTER

```
      SUBROUTINE CORREL(SUM1,NPARAM,NDIM,INNER,TTL)
      INTEGER OUTP,COL1,COL2
      REAL*8 SUM1(1),TTL(1)
      COMMON/TPEBLK/INTP,OUTP,SCRTP(10)
C COMPUTE CORRELATION COEFFICIENTS
      IF(NPARAM-1)20,10,30
10    SUM1(1)=DSQRT(SUM1(1))
20    RETURN
30    DO 2534 I=1,NPARAM
      II=NDIM*(I-1)-((I-1)*I)/2+I
      SUM1(II)=DSQRT(SUM1(II))
      IF(I.EQ.NPARAM) GO TO 2532
      I1=I+1
      DO 2531 J=I1,NPARAM
      IJ=II+J-I
 2531 SUM1(IJ)=SUM1(IJ)/SUM1(II)
C DIVIDE OFF-DIAGONAL TERMS BY SQUARE ROOT OF ROW AND COLUMN DIAGONAL
C TERM
      IF(I.EQ.1) GO TO 2534
 2532 I1=I-1
      JI=I
```

```
      DO 2533 J=1,I1
      SUM1(JI)=SUM1(JI)/SUM1(II)
 2533 JI=JI+NDIM-J
 2534 CONTINUE
C PRINT CORRELATION COEFFICIENTS
      JSTRT=2
      ISTOP=NPARAM-1
      ISTRT=1
      WRITE(OUTF,10112) INNER
 2535 WRITE(OUTF,10214) (TTL(I),I=JSTRT,NPARAM)
      DO 2536 I=1,ISTOP
      I1=NDIM*(I-1)-(I*(I-1))/2+I+ISTRT
      COL1=ISTRT+I
      COL2=MIN0(COL1+19,NPARAM)
      I2=I1+COL2-COL1
 2536 WRITE(OUTF,10215) TTL(I),(SUM1(J),J=I1,I2)
      JSTRT=JSTRT+20
      IF(JSTRT.GT.NPARAM) RETURN
      ISTOP=ISTOP-20
      ISTRT=ISTRT+20
      GO TO 2535
10112 FORMAT(1H0,5X,71HCORRELATION COEFFICIENTS FOR ADJUSTED PARAMETERS
     .AFTER ITERATION NUMBER,I3)
10214 FORMAT(1H0,6X,20A6/(7X,20A6))
10215 FORMAT(1H0,A6,20F6.3)
      END
```

COWCOF

DESCRIPTION

COWCOF assigns predictor or corrector coefficients for the integrator depending on the order requested. Permissible orders are five through fifteen.

NAME	COWCOF
PURPOSE	TO ASSIGN INTEGRATOR COEFFICIENT VALUES - ORDERS 5 - 15
CALLING SEQUENCE	CALL COWCOF(POS,VEL,IORDER,IPC)

SYMBOL	TYPE	DESCRIPTION
POS (1)	DP	OUTPUT - VECTOR OF POSITION COEFFICIENTS
VEL (1)	DP	OUTPUT - VECTOR OF VELOCITY COEFFICIENTS
IORDER	I	INPUT - ORDER
IPC	I	INPUT - PREDICTOR/CORRECTOR SELECTION SWITCH

SUBROUTINES USED	NONE
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	PRINTER - 6

```
      SUBROUTINE COWCOF(POS,VEL,IORDER,IPC)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION POS(1),VEL(1)
      DIMENSION BETA(99),BETAS(99),ALPHA(99),ALPHAS(99)
      DIMENSION B5(4),B6(5),B7(6),B8(7),B9(8),B10(9),B11(10),B12(11),
     .   B13(12),B14(13),B15(14),BS5(4),BS6(5),BS7(6),BS8(7),BS9(8),
     .   BS10(9),BS11(10),BS12(11),BS13(12),BS14(13),BS15(14)
      DIMENSION A5(4),A6(5),A7(6),A8(7),A9(8),A10(9),A11(10),A12(11),
     .   A13(12),A14(13),A15(14),AS5(4),AS6(5),AS7(6),AS8(7),AS9(8),
     .   AS10(9),AS11(10),AS12(11),AS13(12),AS14(13),AS15(14)
      EQUIVALENCE (B5(1),BETA(1)),(B6(1),BETA(5)),(B7(1),BETA(10)),
     .   (B8(1),BETA(16)),(B9(1),BETA(23)),(B10(1),BETA(31)),
     .   (B11(1),BETA(40)),(B12(1),BETA(50)),(B13(1),BETA(61)),
     .   (B14(1),BETA(73)),(B15(1),BETA(85))
      EQUIVALENCE (BS5(1),BETAS(1)),(BS6(1),BETAS(5)),
     .   (BS7(1),BETAS(10)),(BS8(1),BETAS(16)),(BS9(1),BETAS(23)),
     .   (BS10(1),BETAS(31)),(BS11(1),BETAS(40)),(BS12(1),BETAS(50)),
     .   (BS13(1),BETAS(61)),(BS14(1),BETAS(73)),(BS15(1),BETAS(85))
      EQUIVALENCE (A5(1),ALPHA(1)),(A6(1),ALPHA(5)),(A7(1),ALPHA(10)),
     .   (A8(1),ALPHA(16)),(A9(1),ALPHA(23)),(A10(1),ALPHA(31)),
     .   (A11(1),ALPHA(40)),(A12(1),ALPHA(50)),(A13(1),ALPHA(61)),
     .   (A14(1),ALPHA(73)),(A15(1),ALPHA(85))
      EQUIVALENCE (AS5(1),ALPHAS(1)),(AS6(1),ALPHAS(5)),
     .   (AS7(1),ALPHAS(10)),(AS8(1),ALPHAS(16)),(AS9(1),ALPHAS(23)),
     .   (AS10(1),ALPHAS(31)),(AS11(1),ALPHAS(40)),(AS12(1),ALPHAS(50)),
```

```
     .(AS13(1),ALPHAS(61)),(AS14(1),ALPHAS(73)),(AS15(1),ALPHAS(86))
C     K = (ICRDER-5) * (ICRDER+2) / 2
      DATA  B5/+.16402777777777777D+01, -.22124999999999999D+01,
     .         +.14208333333333333D+01,-.34861111111111110D+00/
      DATA  B6/+.19701388888888888D+01, -.35319444444444444D+01,
     .         +.33999999999999999D+01,-.16680555555555555D+01,
     .         +.32986111111111111D+00/
      DATA  B7/+.22257308201058201D+01, -.51099041005291005D+01,
     .         +.65559193121693121D+01,-.46239748677248677D+01,
     .         +.19073207671957671D+01,-.31559193121693121D+00/
      DATA  B8/+.25899553571428571D+01, -.69352513227513227D+01,
     .         +.11119237367724867D+02,-.10903465608465608D+02,
     .         +.647118882275132274D+01,-.214093915343915343D+01,
     .         +.304224537037037035D+00/
      DATA  B9/+.286462335752377424D+01, -.399932732583774250D+01,
     .         +.173115153769641269D+02,-.212238455233977072D+02,
     .         +.167915608361624214D+02,-.833316715259841269D+01,
     .         +.236830054012345678D+01,-.294863000440917107D+00/
      DATA B10/+.317179880401234557D+01, -.112951303972663139D+02,
     .         +.253460278769841269D+02,-.372394706233977071D+02,
     .         +.365798500881234214D+02,-.244037921625984125D+02,
     .         +.104035130401234567D+02,-.259057157186943353D+01,
     .         +.228975446428571428D+00/
      DATA B11/+.345196540045628223D+01, -.130153372652617444D+02,
     .         +.354235533489658469D+02,-.608353967251353918D+02,
     .         +.721637392401194484D+02,-.597073313146343396D+02,
     .         +.339395331414141414D+02,-.126774970438512105D+02,
     .         +.250865181442400192D+01,-.230139596643936721D+00/
      DATA B12/+.372625394048786145D+01, -.155594026555777349D+02,
     .         +.477756026503976065D+02,-.937472315253802789D+02,
     .         +.129779502646755250D+03,-.125822597402597402D+03,
     .         +.315353025480499430D+02,-.455833610475430976D+02,
     .         +.151505311158459595D+02,-.302296499675992731D+01,
     .         +.274265540031540539D+00/
      DATA B13/+.30992227372615023D+01, -.155138039603873715D+02,
     .         +.625721592229384591D+03,-.133137921245632326D+03,
     .         +.219559022082059346D+03,-.253113920461202313D+03,
     .         +.215926523757475677D+03,-.134353581232947193D+03,
     .         +.595403903374900073D+02,-.179194315693106998D+02,
     .         +.323359235456173557D+01,-.257021846773643774D+00/
      DATA B14/+.425963413562513673D+01, -.226910261604871495D+02,
     .         +.800193752151345187D+02,-.176294317937285758D+03,
     .         +.345412939523524560D+03,-.462430192513375452D+03,
     .         +.460057275648220092D+03,-.343735149139299548D+03,
     .         +.190394308274968229D+03,-.759757282093690419D+02,
     .         +.205607716467377651D+02,-.344126502717292538D+01,
     .         +.254351348366606509D+00/
      DATA B15/+.451977052175573777D+01, -.260727733101459631D+02,
     .         +.100310017111057339D+03,-.270693327179779555D+03,
     .         +.335610462754764309D+03,-.797275734334598026D+03,
     .         +.905441221403188471D+03,-.770129920544262927D+03,
     .         +.525159850091190764D+03,-.251974251441159733D+03,
     .         +.550797911352316517D+02,-.237313339251258077D+02,
     .         +.364612443302541955D+01,-.260135398127501936D+00/
      DATA B55/+.34861111111111111D+00, +.245833333333333333D+00,
     .         -.120833333333333333D+00,+.263888888888888888D-01/
     .
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
 DATA BS6/+.3298611111111111111D+00, +.3208333333333333333D+00,     COWC 112
.           -.2233333333333333333D+00,+.1013888888888888888D+00,     COWC 113
.           -.1874999999999999999D-01/                               COWC 114
 DATA BS7/+.3155919312169312169D+00, +.3921792328042328040D+00,     COWC 115
.           -.3760251322751322746D+00,+.2440806878306873300D+00,     COWC 116
.           -.9009589947089947070D-01,+.1426917989417989410D-01/     COWC 117
 DATA BS8/+.3042245370370370360D+00, +.4603835978835978830D+00,     COWC 118
.           -.5465360449735449730D+00,+.4716235716235714290D+00,     COWC 119
.           -.2606068121693121690D+00,+.3247354497354497340D-01,     COWC 120
.           -.1136739417989417990D-01/                               COWC 121
 DATA BS9/+.2948650004409171070D+00, +.5208773540564373390D+00,     COWC 122
.           -.7430233134920634910D+00,+.7349073522927689580D+00,     COWC 123
.           -.5220655330335076990D+00,+.2729603134920531510D+00,     COWC 124
.           -.7666315035273368590D-01,+.9355335536119929440D-02/     COWC 125
 DATA BS10/+.2669754464285714280D+00,+.5690107851552028210D+00,     COWC 126
.           -.9640148253377425040D+00,+.1240370376984126980D+01,     COWC 127
.           -.1140564373897707230D+01,+.7209438238133421510D+00,     COWC 128
.           -.2976546626984126980D+00,+.7249559859438536140D-01,     COWC 129
.           -.7882554012345679000D-02/                               COWC 130
 DATA BS11/+.2601959596443936721D+00,+.6500921350159151820D+00,     COWC 131
.           -.1200305428284591940D+01,+.1210991779603442350D+01,     COWC 132
.           -.1995591471961680290D+01,+.1575950930247394580D+01,     COWC 133
.           -.6678660614077280730D+00,+.3167875681417343080D+00,     COWC 134
.           -.6595520387405504060D-01,+.6788449934634706850D-02/     COWC 135
 DATA BS12/+.2742555400315590590D+00,+.7093430001402913060D+00,     COWC 136
.           -.1374687565393786750D+01,+.2521716545173961830D+01,     COWC 137
.           -.3239633313552589380D+01,+.3068823182156435430D+01,     COWC 138
.           -.2111917907998637160D+01,+.1027674337622234280D+01,     COWC 139
.           -.3255477424232526450D+00,+.6602641410301133010D-01,     COWC 140
.           -.5924056412337662330D-02/                               COWC 141
 DATA BS13/+.2690236467726467740D+00,+.7669356289777449420D+00,     COWC 142
.           -.1742906093027052430D+01,+.3355342732735751870D+01,     COWC 143
.           -.4967742093675153450D+01,+.5433175437329317130D+01,     COWC 144
.           -.4531270193171665560D+01,+.2765703112745343350D+01,     COWC 145
.           -.1159602129991049330D+01,+.3540445432953277008D+00,     COWC 146
.           -.6352768224979079790D-01,+.5278631257930235060D-02/     COWC 147
 DATA BS14/+.2643513433660650990D+00,+.8230564059462252116D+00,     COWC 148
.           -.2071620937691841090D+01,+.4414492582285057060D+01,     COWC 149
.           -.7263103805162104390D+01,+.9192754175706990670D+01,     COWC 150
.           -.8853273721273721270D+01,+.5160361851123321850D+01,     COWC 151
.           -.3514363341476970810D+01,+.1385076192344575200D+01,     COWC 152
.           -.3722425771145802550D+00,+.6136667414245745920D-01,     COWC 153
.           -.4677433407042264510D-02/                               COWC 154
 DATA BS15/+.2601363961276010360D+00,+.2778307835969323263D+00,     COWC 155
.           -.2400397262574259770D+01,+.5520354922640522300D+01,     COWC 156
.           -.1029679465605101745D+02,+.1461739770730703420D+02,     COWC 157
.           -.1509613376341211265D+02,+.1399321499325571320D+02,     COWC 158
.           -.3939607373077014770D+01,+.4396758406373348290D+01,     COWC 159
.           -.1577716917470165490D+01,+.3701325937348343410D+00,     COWC 160
.           -.5947137751411341160D-01,+.4214352235005472850D-02/     COWC 161
 DATA   A5/+.2453333333333333330D+00, -.2416666666666666660D+00,     COWC 162
.           +.7916666666666666660D-01,+0.0D0/                        COWC 163
 DATA   A6/+.3208333333333333330D+00, -.4604166666666666660D+00,     COWC 164
.           +.2021666666666666660D+00,-.7194479999999999990D-01,     COWC 165
.           +0.0D0/                                                  COWC 166
 DATA   A7/+.3921792328042328040D+00, -.7500705026455026459D+00,     COWC 167
```

```
     .          +.7222420634920634921D+00,-.3603335973335978823D+00,      COWC 168
     .          +.713458994708994708D-01,+0.0D0/                          COWC 169
      DATA A8/+.460383597883597883D+00, -.109307208994708994D+01,         COWC 170
     .          +.141428571428571428D+01,-.104242724867724867D+01,        COWC 171
     .          +.412367724867724867D+00,-.632043650793650792D-01,        COWC 172
     .          +0.0D0/                                                   COWC 173
      DATA A9/+.525373354056437309D+00, -.149604662698412698D+01,         COWC 174
     .          +.239672205687830687D+01,-.235234237213403879D+01,        COWC 175
     .          +.139430406746031745D+01,-.461178902116402116D+00,        COWC 176
     .          +.654957561728395060D-01,+0.0D0/                          COWC 177
      DATA A10/+.529013786155202821D+00, -.192802963157548500D+01,        COWC 178
     .          +.372267113095238095D+01,-.456225749559082892D+01,        COWC 179
     .          +.360471913091710758D+01,-.173712797619047619D+01,        COWC 180
     .          +.507478780864197530D+00,-.631404320987654320D-01,        COWC 181
     .          +0.0D0/                                                   COWC 182
      DATA A11/+.650092436016915182D+00, -.241651035055918389D+01,        COWC 183
     .          +.543270532709032707D+01,-.798232588734672117D+01,        COWC 184
     .          +.787980468123697289D+01,-.520719636844636844D+01,        COWC 185
     .          +.221731297699214365D+01,-.551721430992464325D+00,        COWC 186
     .          +.610725458617123615D-01,+0.0D0/                          COWC 187
      DATA A12/+.709333000140291606D+00, -.294977592767957351D+01,        COWC 188
     .          +.756536564355215551D+01,-.125585332742103575D+02,        COWC 189
     .          +.153441157607824274D+02,-.126715076479918229D+02,        COWC 190
     .          +.719372036335578001D+01,-.253433193943402276D+01,        COWC 191
     .          +.594237726972101971D+00,-.592405541233765233D-01,        COWC 192
     .          +0.0D0/                                                   COWC 193
      DATA A13/+.766036625977744942D+00, -.352591219605410496D+01,        COWC 194
     .          +.101573207920727660D+02,-.102702533747047338D+02;        COWC 195
     .          +.276403771366475959D+02,-.271875211590300131D+02,        COWC 196
     .          +.192904617892209355D+02,-.933681703932839904D+01,        COWC 197
     .          +.319540033965743307D+01,-.635276022497907980D+00,        COWC 198
     .          +.576036258374531356D-01,+0.0D0/                          COWC 199
      DATA A14/+.823065606662252116D+00, -.414324197578358378D+01,        COWC 200
     .          +.132446777466551711D+02,-.291324152205434175D+02,        COWC 201
     .          +.459637708785349534D+02,-.531195723275723276D+02,        COWC 202
     .          +.402225329578632530D+02,-.231197107313157665D+02,        COWC 203
     .          +.124478477356011763D+02,-.372242577114580255D+01,        COWC 204
     .          +.675033415567032031D+00,-.551299806945071741D-01,        COWC 205
     .          +0.0D0/                                                   COWC 206
      DATA A15/+.377860285969323263D+00, -.480077452505853755D+01,        COWC 207
     .          +.168611057679218662D+02,-.411671796242040679D+02,        COWC 208
     .          +.730869885369351712D+02,-.9551693205904725762D+02,       COWC 209
     .          +.958525392527969929D+02,-.715169863345161150D+02,        COWC 210
     .          +.395710653936013946D+02,-.157771391747014549D+02,        COWC 211
     .          +.429165243663372775D+01,-.713652530169350039D+00,        COWC 212
     .          +.547913791070711471D-01,+0.0D0/                          COWC 213
      DATA A55/+.791666666666666666D+01, +.833333333333333332D-02,        COWC 214
     .          -.416666666666666666D-02,+0.0D0/                          COWC 215
      DATA A56/+.749999999999999999D-01, +.208333333333333333D-01,        COWC 216
     .          -.166666666666666666D-01,+.416666666666666666D-02,        COWC 217
     .          +0.0D0/                                                   COWC 218
      DATA A57/+.7124553954708994709D-01, +.354437336437354406D-01,       COWC 219
     .          -.385912698412698412D-01,+.147830687830687830D-01,        COWC 220
     .          -.368410052910052909D-02,+0.0D0/                          COWC 221
      DATA A58/+.982073650793650792D-01, +.511574074074074073D-01,        COWC 222
     .          -.700066137566137566D-01,+.501394126984125753D-01,        COWC 223
```

```
     .          -.193617724867724867D-01,+.314153439153439153D-02,
     .          +0.D00/
      DATA AS9/+.54957561726395060D-01, +.671030603465608465D-01,
     .          -.110635747354497354D+00,+.104370590828924162D+00,
     .          -.559909060846560846D-01,+.153931176306878306D-01,
     .          -.270860890652557319D-02,+0.D00/
      DATA AS10/+.621404320987654320D-01,+.838953293650793650D-01,
     .          -.160097552910052909D+00,+.186806733421516754D+00,
     .          -.142427248677248677D+00,+.685549933862433862D-01,
     .          -.191958774250440916D-01,+.235532407407407407D-02,
     .          +0.D00/
      DATA AS11/+.610726498617123616D-01,+.100433587261503928D+00,
     .          -.217995455547533930D+00,+.302602733696480596D+00,
     .          -.287172005270963604D+00,+.184650798661215327D+00,
     .          -.770937800625300624D-01,+.108975819704985371D-01,
     .          -.206778223705307033D-02,+0.D00/
      DATA AS12/+.592405641233766233D-01,+.119927358905525573D+00,
     .          -.283930542127625460D+00,+.456457340715630716D+00,
     .          -.519014808301266634D+00,+.415493601691511357D+00,
     .          -.230988982082722082D+00,+.848526685505852171D-01,
     .          -.165559538820747153D-01,+.163201573833573333D-02,
     .          +0.D00/
      DATA AS13/+.576035253374531356D-01,+.133296741755750449D+00,
     .          -.357612764991824040D+00,+.642930533027509232D+00,
     .          -.861771833451990390D+00,+.829002097442372430D+00,
     .          -.574745022125664457D+00,+.281285252561403731D+00,
     .          -.922187767486316592D-01,+.162014665975706147D-01,
     .          -.163693826592348764D-02,+0.D00/
      DATA AS14/+.561259500345071741D-01,+.147506835248156026D+00,
     .          -.432653237605210259D+00,+.896081962263592823D+00,
     .          -.134607468281736631D+01,+.130832601300527147D+01,
     .          -.125556599035769372D+01,+.767553097333571043D+00,
     .          -.335370193986715313D+00,+.992513410095984996D-01,
     .          -.178470327683290546D-01,+.147364495294595154D-02,
     .          +0.D00/
      DATA AS15/+.547943791070711471D-01,+.165534057577398350D+00,
     .          -.526812953471698074D+00,+.114991434329951493D+01,
     .          -.200919756266519973D+01,+.250662252573450439D+01,
     .          -.245966503273858771D+01,+.172538470506290446D+01,
     .          -.995649307381554370D+00,+.390903433204552445D+00,
     .          -.105993675007910849D+00,+.175003662821732361D-01,
     .          -.133560177743602708D-02,+0.D00/
      IF(IORDER.GT.15.OR.IORDER.LT.5) GO TO 400
      INDEX=((IORDER-5)*(IORDER+2))/2
      L=IORDER-2
      LP1=L+1
      IF(IPC.EQ.1)GO TO 100
      IF(IPC.NE.0)GO TO 200
      DO 10 I=1,L
      POS(I)=ALPHAS(INDEX+I)
      VEL(I)=BETAS(INDEX+I)
   10 CONTINUE
      VEL(LP1)=BETAS(INDEX+LP1)
      RETURN
  100 DO 110 I=1,L
      POS(I)=ALPHA (INDEX+I)
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      VEL(I)=BETA (INDEX+I)
  110 CONTINUE
      VEL(LP1)=BETA (INDEX+LP1)
      RETURN
  200 WRITE(6,300)IPC
  300 FORMAT(1H1,' PERMISSIBLE VALUES OF IPC ARE 0 AND 1. VALUE PASSED W
     .AS',G10.0,' *****RUN TERMINATED*****')
      GO TO 1000
  400 WRITE(6,500)IORDER
  500 FORMAT(1H1,'PERMISSIBLE VALUES OF IORDER ARE 5 THROUGH 15. VALUE P
     .ASSED WAS',G10.0,'    *****RUN TERMINATED*****')
 1000 STOP 66666
      END
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

COWELL

DESCRIPTION

COWELL is the integration subroutine in GEODYN. It integrates the satellite equations of motion and the variational equations. COWELL features a second order predictor-corrector method with variable stepsize. The order of the integrator is also optional.

The integration procedure is not self-starting. The starting procedure consists of initializing logic and data arrays, using START to obtain the requisite back balues. F is also invoked to evaluate the accelerations; VEVAL is invoked to evaluate the variational equations.

The order of computation for normal processing is as follows:

- Predict position. Predict velocity if drag perturbations present.
- Correct position and velocity at least once, twice if necessary.
- Evaluate (integrated) variational partials based on corrected values for position and velocity.
- Check for decreasing or increasing of stepsize and update tables of back values. If necessary call REARG to increase or decrease stepsize.
- If the time of interest has not been reached, start processing cycle over for the next stop. Otherwise, processing has completed and control is returned to the calling routine.

NAME COWELL

PURPOSE TO SUM COWELL INTEGRATOR WHICH INTEGRATES SATELLITE EQUATIONS OF MOTION AND ALL FORCE MODEL PARTIAL DERIVATIVES TO DESIRED TIME (DAY)

CALLING SEQUENCE CALL COWELL(DAY,Y,FCT,IORDER,H,TIM,SUM,VAR,FAC,M,
NN,ISAT,PP,P,CC,C,VCC,VC)

SYMBOL	TYPE	DESCRIPTION
DAY	DP	INPUT - DESIRED OUTPUT TIME
Y (6,1)	DP	INPUT - ARRAY CONTAINING SATELLITE POSITION & VELOCITY AND FORCE MODEL PARTIALS, AN OUTPUT ARRAY FROM INTERPOLATOR
FCT (3,1)	DP	INPUT & OUTPUT - BACK VALUE ARRAY OF ACCELERATIONS
IORDER (2)	I	INPUT - ARRAY OF INTEGRATION ORDER FOR ORBIT AND PARTIALS
H (2)	DP	INPUT - ARRAY OF STEP SIZES FOR ORBIT AND PARTIALS
TIM (2)	DP	INPUT - ARRAY OF INTEGRATION TIMES OF ORBIT AND PARTIALS
SUM (2,3,1)	DP	INPUT - ARRAY CONTAINING SUMS USED BY INTEGRATOR AND INTERPOLATOR
VAR	L	INPUT - VARIABLE STEP SWITCH
FAC	DP	INPUT - = +1 IF FORWARD = -1 IF BACKWARD
M (2)	I	INPUT - DISPLACEMENT ARRAY USED BY INTEGRATOR
NN	I	INPUT - NUMBER OF EQUATIONS
ISAT	I	INPUT - SATELLITE NUMBER
PP (1)	DP	INPUT - POSITION PREDICTOR COEFFICIENTS
P (1)	DP	INPUT - VELOCITY PREDICTOR COEFFICIENTS
CC (1)	DP	INPUT - POSITION CORRECTION COEFFICIENTS
C (1)	DP	INPUT - VELOCITY CORRECTION COEFFICIENTS

```
 VCC        DP      INPUT - VARIATIONAL PARTIAL CORRECTOR COEFFICIENTS
 (1)

 VC         DP      INPUT - VARIATION PARTIAL FIRST TIME DERIVATIVE
 (1)                        CORRECTOR COEFFICIENTS

SUBROUTINES USED    COWCOF      START       F           REARG       CLEAR
                    INTRP       VEVAL       DNVERT

COMMON BLOCKS       INTBLK      VMAT        CELEM       CTIME

INPUT FILES         NONE

OUTPUT FILES        PRINTER

REFERENCES          'GEODYN SYSTEMS DESCRIPTION'
                    VOLUME 1 - GEODYN DOCUMENTATION
```

```
      SUBROUTINE COWELL(DAY,Y,FCT,IORDER,H,TIM,SUM,VAR,FAC,M,NN,ISAT,  COWE  78
     .  PP,P,CC,C,VCC,VC)                                              COWE  79
      IMPLICIT REAL*8(A-H,O-Z)                                         COWE  80
      DIMENSION SUM(2,3,1),AUX(6),ECON(11),FCT(3,1),H(2),TIM(2),       COWE  81
     .  Y(6,1),        P(1),C(1),PP(1),CC(1),VC(1),VCC(1),PC(3)        COWE  82
      INTEGER IORDER(2),M(2),KOUNT(2)/2*0/,KOUNT2(2)/2*0/              COWE  83
      LOGICAL INITAL,FEVAL,NODRAG,VAR,HLVDSW                           COWE  84
      LOGICAL FEVAL/.TRUE./,STRTP(2)                                   COWE  85
      LOGICAL NORRAT                                                   COWE  86
      DATA ECON/.05D0,.052631579D0,                                    COWE  87
     .  .04672134D0,.044022465D0,                                      COWE  88
     .  .039713131D0,.035961476D0,                                     COWE  89
     .  .032746940D0,.029982465D0,                                     COWE  90
     .  .027632051D0,.025552503D0,                                     COWE  91
     .  .023794802D0/                                                  COWE  92
      EQUIVALENCE(PC1,PC(1)),(PC2,PC(2)),(PC3,PC(3))                   COWE  93
      EQUIVALENCE(AUX(1),AUX1),(AUX(2),AUX2),(AUX(3),AUX3)             COWE  94
      COMMON/INTBLK/LG2(30),E(2),LG4(18),INITAL,NORRAT,LG3(14),        COWE  95
     .  STEPSZ(4),HLVERB(2),                                           COWE  96
     .  DBLEME(2),CTOL(2),                                             COWE  97
     .  RTOL(2),STEPLO(2),STPUP(2),MSAT(14),HLVDSW(2),LOVE(16)         COWE  98
      COMMON/VMAT/VMATRX(3,5),S(6,6),LY1(18)                           COWE  99
      COMMON/CELEM/ELEMST(6,2),IORBEL(25)                              COWE 100
      COMMON/CTIME/DATAEP(2),DAY0,ISTP(17)                             COWE 101
C INITIALIZE                                                           COWE 102
      NODRAG=E(ISAT).GT.0.D0                                           COWE 103
      IORDR1=IORDER(1)                                                 COWE 104
      IORDR2=IORDER(2)                                                 COWE 105
      IOIL1=IORDR1-1                                                   COWE 106
      IOIL2=IOIL1-1                                                    COWE 107
      M1=M(1)                                                          COWE 108
      M2=M(2)                                                          COWE 109
      M1P1=M1+1                                                        COWE 110
      M1P2=M1+2                                                        COWE 111
```

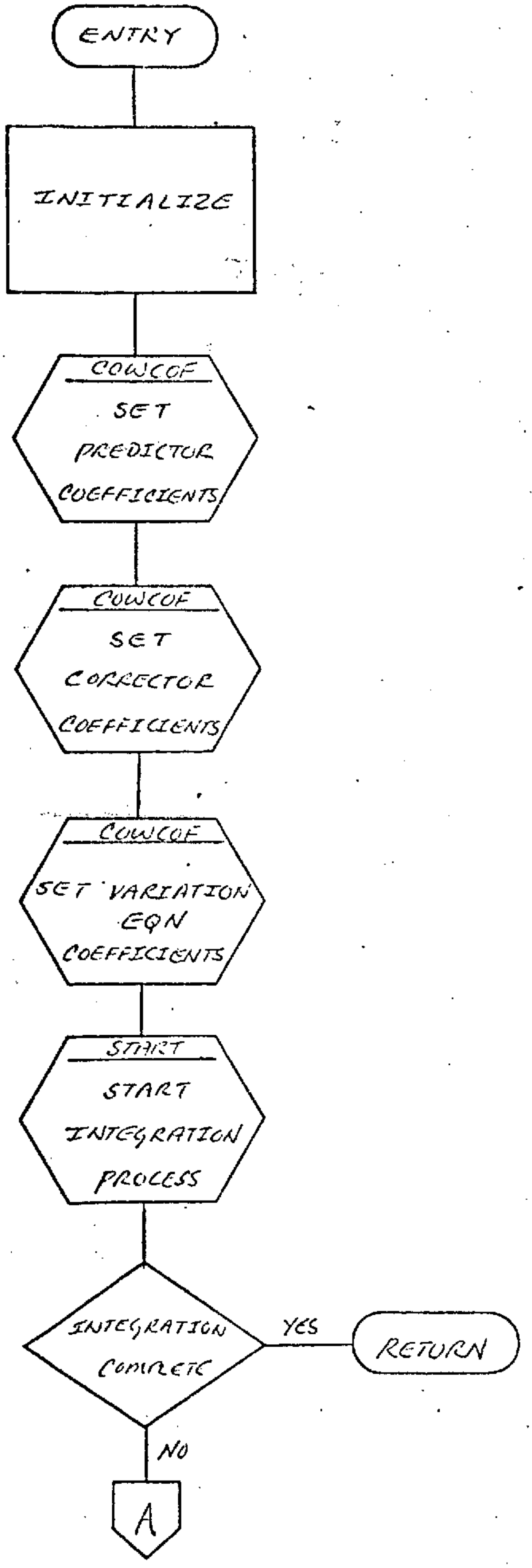
```
      M2P1=M2+1
      IF(NN.EQ.1)GO TO 3
      IO2L1=IORDR2-1
      IO2L2=IORDR2-2
    3 IF(INITAL)GO TO 4
      IF(NN.GT.1)GO TO 260
      GO TO 100
    4 INITAL=.FALSE.
      KOUNT(ISAT)=IO1L1
      M2=0
      M(1)=IO1L1
      IF(NN.EQ.1) GO TO 18
      IF(IORDR2.GT.IORDR1) IORDR2=IORDR1
      IO2L1=IORDR2-1
      IO2L2=IO2L1-1
      KOUNT2(ISAT)=IO2L1
      M(2)=IO2L1
   18 IF(.NOT.VAR) GO TO 20
      M(1)=M(1)+M(1)-1
      IF(NN.GT.1)M(2)=M(2)+M(2)-1
   20 CONTINUE
      M1=M(1)
      M2=M(2)
      M1P1=M1+1
      M1P2=M1P1+1
      M2P1=M2+1
C SET PREDICTOR COEFFICIENTS
   21 CALL COWCCF(PP,P,IORDR1,1)
C SET CORRECTOR COEFFICIENTS
      CALL COWCCF(CC,C,IORDR1,0)
      IF(NN.EQ.1) GO TO 28
      IF(IORDR1.EQ.IORDR2) GO TO 23
C SET VARIATION EQUATION COEFFICIENTS
      CALL COWCCF(VCC,VC,IORDR2,0)
      GO TO 24
   23 DO 22 I=1,IO1L2
      VCC(I)=CC(I)
   22 VC(I)=C(I)
      VC(IO1L1)=C(IO1L1)
   24 CONTINUE
C START INTEGRATION PROCESS
      CALL START(IORDR2,     H(2),FCT,SUM,Y,NN,M1,M2,TIM(2))
      IF(IORDR1.NE.IORDR2.OR.H(1).NE.H(2))GO TO 25
      TIM(1)=TIM(2)
      GO TO 29
   25 CALL START(IORDR1,     H(1),FCT,SUM,Y,1,M1,M1,TIM(1))
   29 STRTR(ISAT)=.FALSE.
C CHECK IF DESIRED TIME HAS BEEN REACHED (DAY)
10000 T2=(TIM(1)-2.D0*H(1))/8.64D4
      IF(DAY*FAC.GE.T2*FAC)GO TO 100
      IF(NN.EQ.1)RETURN
      T2=(TIM(2)-2.D0*H(2))/8.64D4
      IF(DAY*FAC.LT.T2*FAC)RETURN
C PREDICT
  100 DO 150 J=1,3
      PRED = 0.D0
```

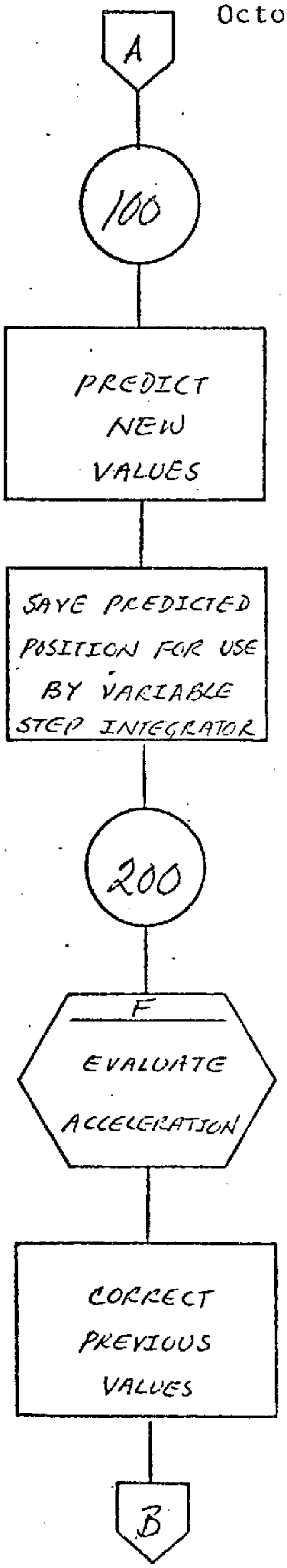
```
      DO 150 I=1,IOIL2
      K=M1P1-I
  150 PRED=PRED+PP(I)*FCT(J,K)
  180 Y(J,1)=      (SUM(2,J,1)+PRED)*H(1)**2
  182 IF(NODRAG) GO TO 195
      DO 190 J=4,6
      J1=J-3
      PRED=0.0D0
      DO 185 I=1,IOIL1
      K=M1P1-I
  185 PRED=PRED+P(I)*FCT(J1,K)
  190 Y(J,1)=   (SUM(1,J1,1)+PRED)*H(1)
C SAVE PREDICTED POSITION FOR USE BY VARIABLE STEP INTEGRATOR
  195 IF(.NOT.(FEVAL.OR.VAR))GO TO 194
      PC1=Y(1,1)
      PC2=Y(2,1)
      PC3=Y(3,1)
  194 NCORR=0
      TYM=TIM(1)+H(1)
  200 NCORR=NCORR+1
C EVALUATE ACCELERATION
      CALL F(TYM,Y,AUX,.FALSE.)
C CORRECT
  220 DO 250 J=1,3
      POS=0.D0
      VEL=0.D0
      DO 240 I=2,IOIL2
      K=M1P2-I
      POS=POS+CC(I)*FCT(J,K)
  240 VEL=VEL+C(I)*FCT(J,K)
      POS=POS+CC(1)*AUX(J)
      VEL=VEL+C(1)*AUX(J)
      VEL=VEL+C(IOIL1)*FCT(J,M1P2-IOIL1)
      Y(J,1)=(POS+SUM(2,J,1))*H(1)**2
      Y(J+3,1)=(VEL+SUM(1,J,1))*H(1)
  250 CONTINUE
  251 IF(NCORR.GT.1)GO TO 256
      IF(.NOT.(FEVAL.OR.VAR))GO TO 255
C COMPUTE DISCREPANCY BETWEEN PREDICTED & CORRECTED VALUES
      YPCR=(PC1-Y(1,1))**2+(PC2-Y(2,1))**2+(PC3-Y(3,1))**2
      YPCR=YPCR*ECCN(IORDR1-4)**2
      IF(.NOT.VAR)GO TO 257
      IF(YPCR.GT.DBLERB(ISAT))GOTO259
C INCREASE STEP
      IF(KOUNT(ISAT).LT.M1)GOTO257
      IF(NN.GT.1.AND.KOUNT2(ISAT).LT.M2)GOTO257
      Q=2.D0
      IF(HLVOSW(ISAT))GOTO252
      IF(YPCR.LT.1.D-28)GO TO 252
      Q=RTOL(ISAT)/YPCR
      Q=Q**(1.D0/(DFLOAT(IOIL1+IOIL1)))
      IF(Q.GT.2.D0)Q=2.D0
      IF(Q.LT..5D0)Q=.5D0
  252 IF(FAC*Q*H(1).GT.STPUP(ISAT))GOTO257
      IF(NN.GT.1.AND.H(2)*Q.GT.STPUP(ISAT))GOTO257
      CALL KSARG(   M1,M1,IORDR1,IORDR1,H(1),H(1),      TIM(1),TIM(1),
```

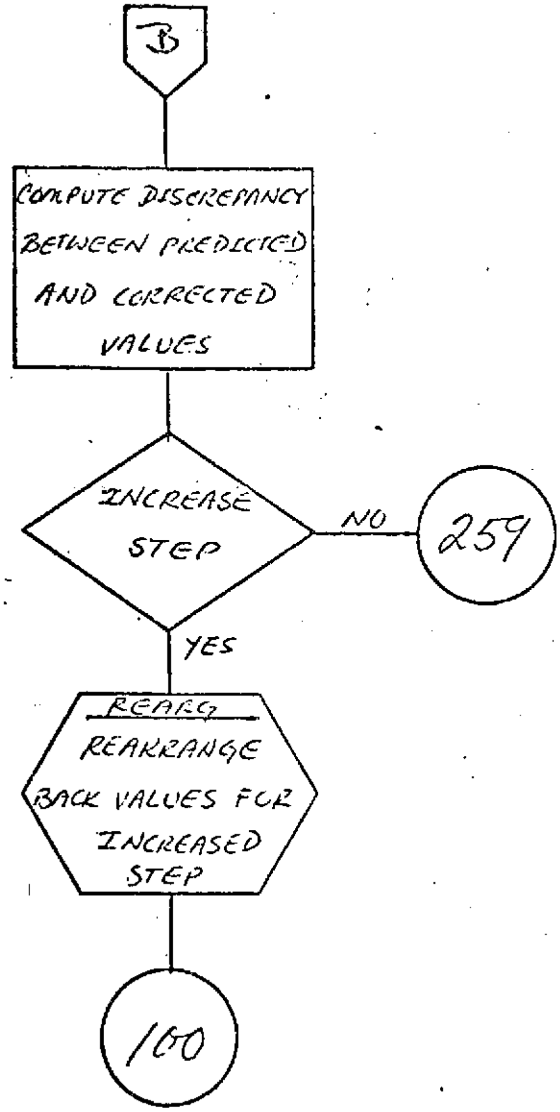
```
     . FCT,1,SUM,SUM,CC,C,Y,Y    ,.FALSE.,FCT,Q)
      IF(NN.GT.1)CALL REARG(M2,M1    ,IORDR1,IORDR2,H(1),H(2),      TIM(1)
     .  ,TIM(2),FCT(1,M1+1),NN,SUM(1,1,2),SUM,VCC,VC,Y(1,2),Y,.TRUE.,
     .  FCT,Q )
      KOUNT(ISAT)=IO1L2
      IF(NN.GT.1)KOUNT2(ISAT)=IO2L2
      GO TO 100
  255 IF(YPCR.LT.HLVERB(ISAT))GOTO257
C REDUCE STEP
      Q=.5D0
      IF(HLVDSW(ISAT))GOTO258
      Q=RTOL(ISAT)/YPCR
      Q=Q**(1.D0/(DFLOAT(IO1L1+IO1L1)))
      IF(Q.GT.2.D0)Q=2.D0
      IF(Q.LT..5D0)Q=.5D0
  258 IF(FAC*H(1)*Q.LT.STEPLC(ISAT))GOTO257
C IF FIRST TIME OUT OF STARTER AND STEP REDUCTION IS NEEDED, RESTART
C WITH SMALLER STEP
      IF(STRTR(ISAT))GOTO253
      H(1)=Q*H(1)
      H(2)=Q*H(2)
      AUX1=DABS(H(1))
      CALL CLEAR(Y,12,NN)
      DO 2520 I=1,6
      Y(I,1)=ELEMST(I,ISAT)
 2520 CONTINUE
      TIM(1)=8.64D4*DAY0
      TIM(2)=TIM(1)
      IF(NN.EQ.1)GO TO 2540
      DO 2525 I=1,6
      Y(I,I+1)=1.D0
 2525 CONTINUE
 2540 WRITE(6,254)ISAT,AUX1
  254 FORMAT(1H1,' SAT',I2 ,/,' INITIAL STEP TOO LARGE',/,
     . ' RESTARTING WITH',G11.4,' SEC STEP')
      IF(NN.EQ.1)GOTO25
      GO TO 24
  253 CALL REARG(    M1,M1,IORDR1,IORDR1,H(1),H(1),      TIM(1),TIM(1),
     .  FCT,1,SUM,SUM,CC,C,Y,Y,      ,FALSE.,FCT,Q )
      IF(NN.GT.1)CALL REARG(M2,M1,    IORDR1,IORDR2,H(1),H(2),      TIM(1)
     .  ,TIM(2),FCT(1,M1+1),NN,SUM(1,1,2),SUM,VCC,VC,Y(1,2),Y,.TRUE.,
     .  FCT,Q)
      KOUNT(ISAT)=IO1L2
      IF(NN.GT.1)KOUNT2(ISAT)=IO2L2
      GO TO 100
  257 IF(FEVAL.AND.YPCR.LE.CTOL(ISAT))GOTO256
  256 IF(NCORR.LT.2)GO TO 200
  259 TIM(1)=TYM
      IF(VAR.AND.KOUNT(ISAT).LT.M1)KOUNT(ISAT)=KOUNT(ISAT)+1
      MM=M1-1
C RESET BACK VALUE ARRAY
      DO 197 II=1,MM
      DO 197 J=1,3
  197 FCT(J,II)=FCT(J,II+1)
      FCT(1,M1)=AUX1
      FCT(2,M1)=AUX2
```

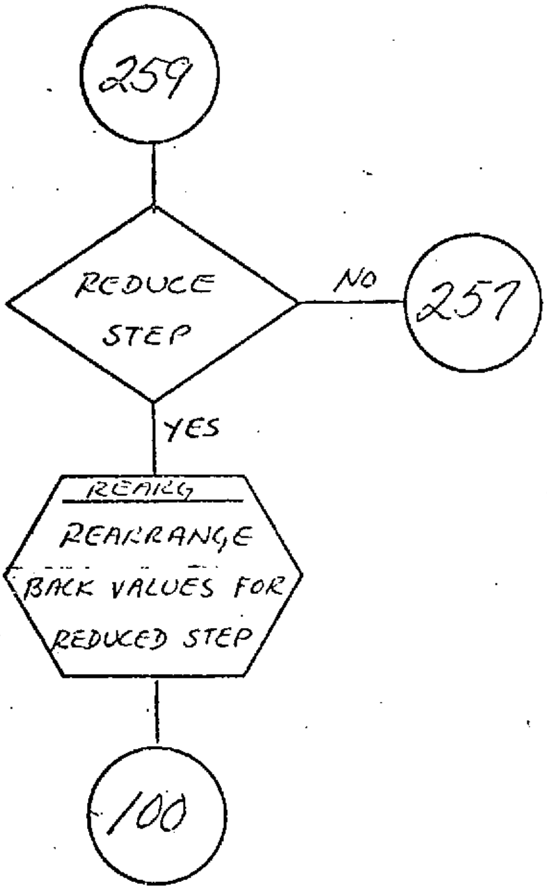
```
      FCT(3,M1)=AUX3
  199 CONTINUE
      STRTR(ISAT)=.TRUE.
C UPDATE SUMS
      DO 998 J=1,3
      SUM(1,J,1)=SUM(1,J,1)+FCT(J,M1)
  998 SUM(2,J,1)=SUM(2,J,1)+SUM(1,J,1)
  260 IF(NN.EQ.1)GOTO950
      PRED=TIM(2)+H(2)
      IF(PRED.GT.TIM(1))GO TO 100
      TIM(2)=PRED
      NNL1=NN-1
      MM=M2-1
      DO 198 JJ=1,NNL1
      KKK=(JJ-1)*M2+M1
      DO 198 II=1,MM
      KK=KKK+II
      DO 198 J=1,3
  198 FCT(J,KK)=FCT(J,KK+1)
      IF(VAR.AND.KOUNT2(ISAT).LT.M2)KOUNT2(ISAT)=KOUNT2(ISAT)+1
      PRED=(TIM(2)-TIM(1))/H(1)
      IF(PRED.EQ.0.D0)GO TO 262
      CALL  INTRP (PRED,H(1),IORDR1,1,Y,FCT,M1,SUM)
  262 CALL  F(TIM(2),Y,AUX,.TRUE.)
C EVALUATE FORCE MODEL PARTIAL ACCELERATION
      CALL  VEVAL(Y,FCT(1,M1+M2),6,.FALSE.,M2)
C INTEGRATE VARIATION EQUATIONS USING CORRECTOR COEFFICIENTS ONLY
      DO 320 L1=2,NN
      KO=M1+(L1-2)*M2
      DO 320 J=1,3
      PC1=0.D0
      DO 315 I=1,IO2L2
      K=M2P1-I
  315 PC1=PC1+VCC(I)*FCT(J,KO+K)
  320 Y(J,L1)=(SUM(2,J,L1)+PC1)*H(2)**2
      H2C=VCC(1)*H(2)**2
      IF(NODRAG) GO TO 331
      NNN=6
      H1C=VC(1)*H(2)
      DO 325 L1=2,NN
      KO=M1+(L1-2)*M2
      DO 325 J=1,3
      J1=J+3
      PC1=0.D0
      DO 322 I=1,IO2L1
      K=M2P1-I
  322 PC1=PC1+VC(I)*FCT(J,KO+K)
  325 Y(J1,L1)=  (SUM(1,J,L1)+PC1)*H(2)
      DO 330 J=1,3
      J1=J+3
      DO 330 I=1,3
      I1=I+3
      S(I,J)=-H2C*VMATRX(I,J)
      S(I,J1)=-H2C*VMATRX(I,J1)
      S(I1,J)=-H1C*VMATRX(I,J)
  330 S(I1,J1)=-H1C*VMATRX(I,J1)
```

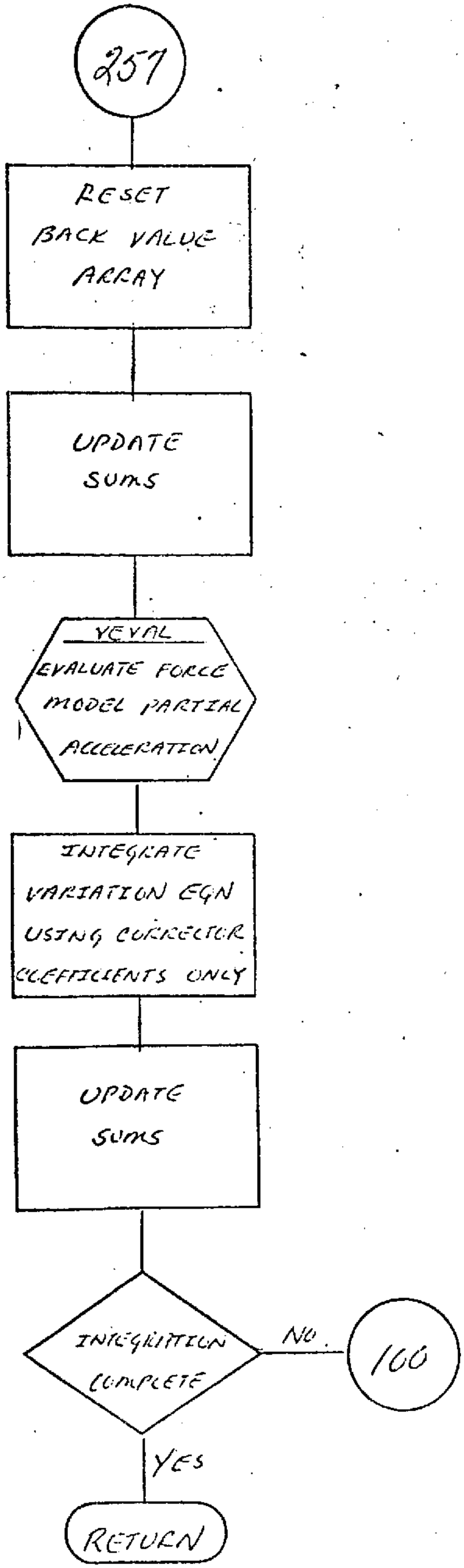
```
      S(J,J)=S(J,J)+1.D0                                                COWE 336
  330 S(J1,J1)=S(J1,J1)+1.D0                                            COWE 337
      GO TO 334                                                         COWE 338
  331 NNN=3                                                             COWE 339
      DO 333 I=1,3                                                      COWE 340
      DO 332 J=1,3                                                      COWE 341
  332 S(I,J)=-H2C*VMATRX(I,J)                                           COWE 342
  333 S(I,I)=S(I,I)+1.D0                                                COWE 343
  334 CALL DNVERT(NNN,S,5,AUX)                                          COWE 344
      DO 336 L1=2,NN                                                    COWE 345
      KK=M1+(L1-1)*M2                                                   COWE 346
      DO 335 J=1,NNN                                                    COWE 347
      PC1=0.D0                                                          COWE 348
      DO 337 K=1,NNN                                                    COWE 349
  337 PC1=PC1+S(J,K)*Y(K,L1)                                            COWE 350
  335 AUX(J)=PC1                                                        COWE 351
      DO 336 I=1,3                                                      COWE 352
      PC1=0.D0                                                          COWE 353
      DO 338 J=1,NNN                                                    COWE 354
  338 PC1=PC1+VMATRX(I,J)*AUX(J)                                        COWE 355
  336 FCT(I,KK)=FCT(I,KK)+PC1                                           COWE 356
      DO 900 L=2,NN                                                     COWE 357
      L1=M1+L*M2-M2                                                     COWE 358
      DO 900 J=1,3                                                      COWE 359
C UPDATE SUMS                                                           COWE 360
      SUM(1,J,L)=SUM(1,J,L)+FCT(J,L1)                                   COWE 361
  900 SUM(2,J,L)=SUM(2,J,L)+SUM(1,J,L)                                  COWE 362
C DETERMINE IF DESIRED TIME HAS BEEN REACHED                            COWE 363
      T2=(TIM(2)-2.D0*H(2))/8.64D4                                      COWE 364
      IF(DAY*FAC.GT.T2*FAC)GO TO 260                                    COWE 365
  950 T2=(TIM(1)-2.D0*H(1))/8.64D4                                      COWE 366
      IF (DAY*FAC.GT.T2*FAC) GO TO 100                                  COWE 367
      RETURN                                                            COWE 368
      END                                                               COWE 369
```


ENTRY
INITIALIZE
COWCOF
SET PREDICTOR COEFFICIENTS
COWCOF
SET CORRECTOR COEFFICIENTS
COWCOF
SET VARIATION EQN COEFFICIENTS
START
START INTEGRATION PROCESS
INTEGRATION COMPLETE
YES
RETURN
NO
A


A
100
PREDICT NEW VALUES
SAVE PREDICTED POSITION FOR USE BY VARIABLE STEP INTEGRATOR
200
F
EVALUATE ACCELERATION
CORRECT PREVIOUS VALUES
B


B
COMPUTE DISCREPANCY BETWEEN PREDICTED AND CORRECTED VALUES
INCREASE STEP
NO
259
YES
REARG
REARRANGE BACK VALUES FOR INCREASED STEP
100


259
REDUCE STEP
NO
257
YES
REARG
REARRANGE BACK VALUES FOR REDUCED STEP
100


257
RESET BACK VALUE ARRAY
UPDATE SUMS
YEVAL
EVALUATE FORCE MODEL PARTIAL ACCELERATION
INTEGRATE VARIATION EQN USING CORRECTOR COEFFICIENTS ONLY
UPDATE SUMS
INTEGRATION COMPLETE
NO
100
YES
RETURN

DATARD

DESCRIPTION

DATARD is a subroutine specifically designed for the MULTI-ARC GEODYN program.

DATARD stores on scratch files, updates and retrieves from scratch files all a priori and parameter estimation in information about each arc.

DATARD uses several switches that operate as follows.

UPDATE - .TRUE. = write information.
.FALSE. = Read information.

APRIOR - .TRUE. = a priori information to be operated on.
.FALSE. = a priori information not to be operated on.

BMAT - .TRUE. = normal equations to be operated on.
.FALSE. = normal equations not to be operated on.

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME	DATARD
ENTRY POINT	PURPOSE
DATRD1	INITIALIZATION
DATARD	TO STORE, UPDATE, AND RETRIEVE INDIVIDUAL ARC PARAMETER INFORMATION

CALLING SEQUENCE CALL DATRD1(CD,BIAS,SUM1,INDXCS,SUM2)

SYMBOL	TYPE	DESCRIPTION
CD (1)	I	INPUT & OUTPUT - ADJUSTED VALUES OF THE INDIVIDUAL ARC FORCE MODEL PARAMETERS
BIAS (1)	I	INPUT - A PRIORI VALUES OF THE INDIVIDUAL ARC PARAMETER
SUM1 (1)	DP	INPUT & OUTPUT - ARC NORMAL MATRIX
INDXCS (1)	I*2	INPUT - INDICES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
SUM2 (1)	DP	INPUT - RIGHT HAND SIDE OF NORMAL EQUATIONS

CALLING SEQUENCE CALL DATARD(ARCNO,APRIOR,UPDATE,BMAT)

SYMBOL	TYPE	DESCRIPTION
ARCNO	I	INPUT - ARC NUMBER
APRIOR	L	INPUT - .TRUE. THEN READ OR WRITE A PRIORI INFORMATION ACCORDING TO 'UPDATE' .FALSE. THEN DON'T READ OR WRITE A PRIORI INFORMATION
UPDATE	L	INPUT - .TRUE. THEN WRITE INFORMATION .FALSE. THEN READ INFORMATION
BMAT	L	INPUT - .TRUE. THEN READ OR WRITE NORMAL MATRIX ACCORDING TO 'UPDATE' .FALSE. THEN DON'T READ OR WRITE NORMAL MATRIX

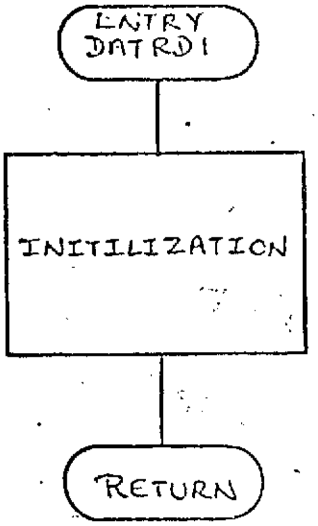
SUBROUTINES USED	CLEAR	DWRITE	DREAD	ERROR	
COMMON BLOCKS	ALPMRC	APARAM	CELEM	CONOUT	CPARAM
	CTIME	FMODEL	INITBK	INTBLK	PREBLK
	PRIORI	TPEBLK	VRBLCK		
INPUT FILES	NONE				

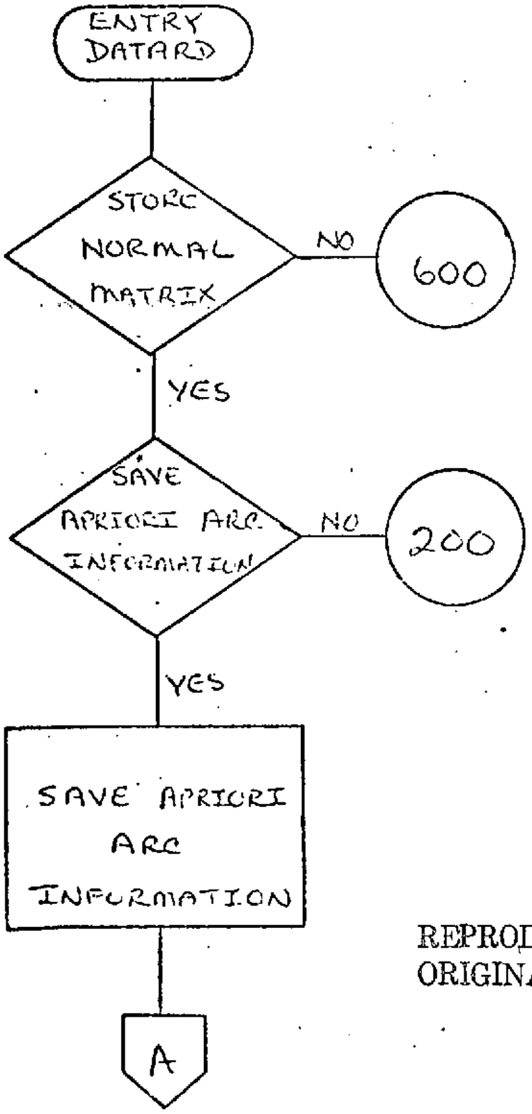
OUTPUT FILES NONE

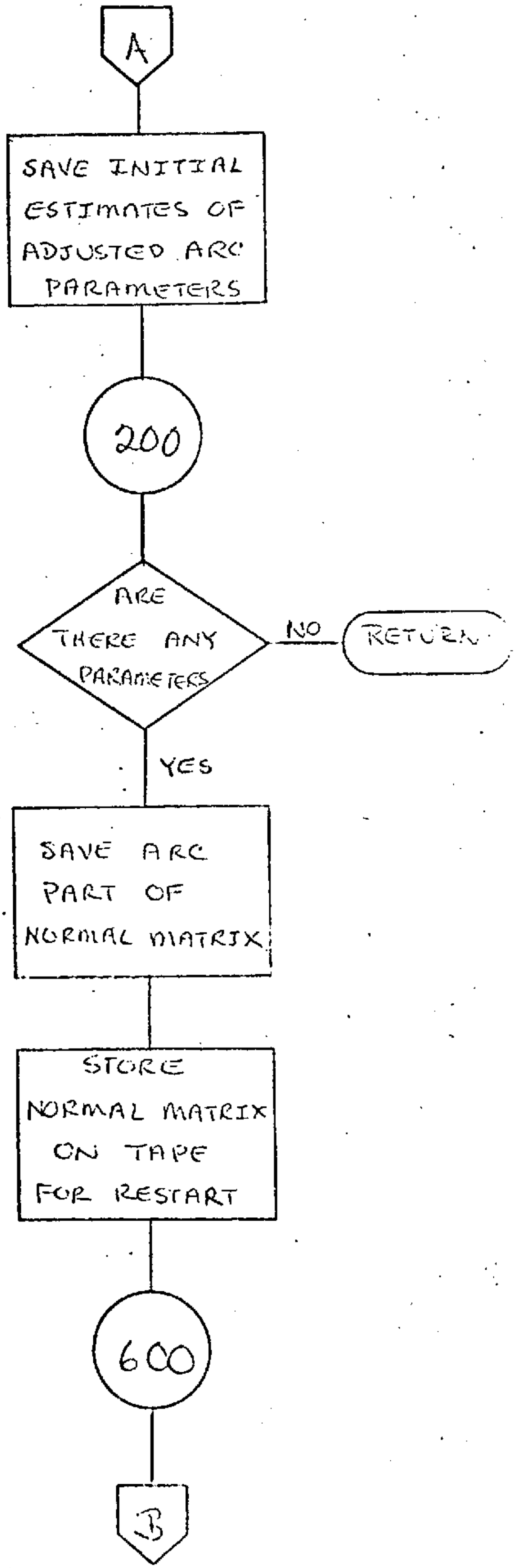
```
      SUBROUTINE DATRD1(CD,BIAS,SUM1,INDXCS,SUM2)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL APRIOR,UPDATE,BMAT,CMPCPR,STARTR,STARTW,HYPER
      INTEGER*2 INDXCS
      INTEGER BIAS,ARCNO,SCRA,SCRB,SCRC,FLTP,OUTP,XYZTP,PLOTP,GRDTP,
     *   THOOT1,ELEMIN,ELEMST,DAYREF,CD,SUM1,DATP,STARTA,STARTO,OUTSTR,
     *   SUM2,EBIAS
      DOUBLE PRECISION MODEL
      DIMENSION CD(1),BIAS(1),SUM1(1),INDXCS(1),NSTORE(6),SUM2(1)
      DIMENSION EBIAS(1)
      COMMON/ALFMPC/ALPHA(53),ELCUT,HYPER
      COMMON/APARAM/INPAR(10)
      COMMON/CELEM/ELEMST(53)
      COMMON/CONOUT/IG1(13),STARTR,STARTW,STARTA,STARTO,INSTRT,OUTSTR
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,NGPCOM,
     *   NCSEST,CMPCPR,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,INNRSW,
     *   NCONST,NOCONS
      COMMON/CTIME/DAYREF(23)
      COMMON/FMODEL/INDEX(4),CS(30,32),MODEL(6)
      COMMON/INITBK/IPPYMD(43),MISLDC(9)
      COMMON/INTBLK/THOOT1(53),NORRAT(79)
      COMMON/PASBLK/DAYSTA,NOBS(15)
      COMMON/PRIORI/ELEMIN(192)
      COMMON/TPEBLK/INTP,OUTP,DATP,XYZTP(3),PLOTP,IOBS,SCRA,SCRC,FLTP,
     *   GRDTP
      COMMON/VRBLCK/A1(66),IBUF(1980),ADRN(69)
      EQUIVALENCE (SCRB,DAFILE),(NGPARC,INPAR(6)),(NPARAM,INPAR(8))
      EQUIVALENCE (MAXSAT,IG1(6)),(NEBIAS,INPAR(9))
      DATA SCRB/15/
      NSTORE(1)=1
      NSTORE(2)=6*MBIAS+2*(NCSEST-NGPCOM)-12*MAXSAT
      NSTORE(3)=NSTORE(2)+2*NGPCOM+1
      NSTORE(4)=NSTORE(3)+MBIAS*3-1+2*(NCSEST-NGPCOM)
      NSTORE(5)=NSTORE(4)+2*NGPCOM+1
      NSTORE(6)=NSTORE(5)+MBIAS-3*MAXSAT+(NGPC1-2)/2
      NREC=(NGPC1-2)/225+2
      NREC1=NREC+1
      NWORD=2*(NGPC1-1)
      RETURN
      ENTRY DATARD(ARCNO,APRIOR,UPDATE,BMAT)
      NSTART=(ARCNO-1)*NREC+1
      IF(.NOT.UPDATE) GO TO 600
C SAVE A PRIORI ARC INFORMATION
      IF(.NOT.APRIOR) GO TO 200
      WRITE(SCRB) INPAR,INDEX,DAYSTA,XYZTP,NORRAT,ELCUT,HYPER
      IF(NEBIAS.LE.0) GO TO 90
      DO 40 I1=1,NEBIAS,350
      I2=MIN0(I1+349,675)
   40 WRITE(SCRB) (EBIAS(I),I=I1,I2)
   90 DO 100 N=1,6
      ISTORT=NSTORE(N)
```

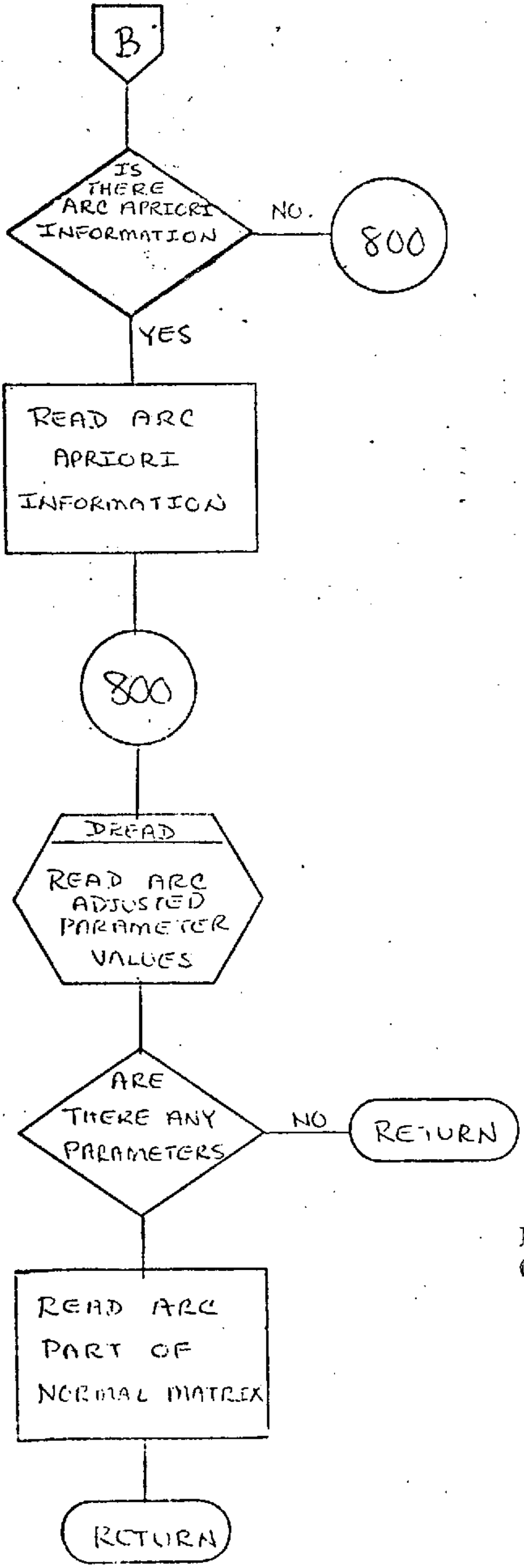
```
      ISTOR2=NSTORE(N+1)
      DO 100 I1=ISTOR1,ISTOR2,450
      I2=MIN0(I1+449,ISTOR2)
  100 WRITE(SCRA) (BIAS(I),I=I1,I2)
C SAVE INITIAL ESTIMATES OF ADJUSTED ARC PARAMETERS
      IF(NGPARC.LE.0) GO TO 200
      NGP=3*NGPARC
      DO 150 I1=1,NGP,900
      I2=MIN0(I1+899,NGP)
  150 WRITE(SCRA) (INDXCS(I),I=I1,I2)
  200 CALL CLEAR(IBUF,990,2)
      DO 300 I=1,53
  300 IBUF(I)=ELEMST(I)
      DO 310 I=1,23
      I1=I+53
  310 IBUF(I1)=DAYREF(I)
      DO 320 I=1,48
      I1=I+76
  320 IBUF(I1)=IERYMD(I)
      DO 330 I=1,192
      I1=I+124
  330 IBUF(I1)=ELEMIN(I)
      IBUF(317)=NPARAM
      CALL DWRITE(SCRB,NSTART,IBUF)
      DO 400 J1=1,NWORD,450
      J2=MIN0(J1+449,NWORD)
      DO 350 J=J1,J2
      I=J-J1+1
C SAVE ARC PART OF NORMAL MATRIX
  350 IBUF(I)=CC(J)
      NSTART=NSTART+1
  400 CALL DWRITE(SCRB,NSTART,IBUF)
      IF(.NOT.EMAT) RETURN
      IF(NPARAM.LE.0) RETURN
C STORE NORMAL MATRIX ON TAPE FOR RESTART
      L=2*NDIM*NPARAM-NPARAM*(NPARAM-1)
      DO 500 I1=1,L,450
      I2=MIN0(I1+449,L)
  500 WRITE(SCRC) (SUM1(I),I=I1,I2)
      IF(.NOT.STARTW) RETURN
      L=L/2
      WRITE(OUTSTR) L
      L=NDIM*(NDIM+1)
      DO 550 I1=1,L,450
      I2=MIN0(I1+449,L)
  550 WRITE(OUTSTR) (SUM1(I),I=I1,I2)
      L1=NDIM*2
      DO 575 I1=1,L1,450
C READ ARC A PRIORI INFORMATION
      I2=MIN0(I1+449,L1)
  575 WRITE(OUTSTR) (SUM2(I),I=I1,I2)
      WRITE(OUTSTR) INDEX
      RETURN
  600 IF(.NOT.APRIOR) GO TO 800
      CALL CLEAR(MISLOG,6,1)
      READ(SCRA) INPAR,INDEX,DAYSTA,XYZTP,NORFAT,ELCUT,HYPER
```

```
      IF(NEBIAS.LE.0) GO TO 680                                         DATA 168
      DO 640 I1=1,NEBIAS,350                                            DATA 169
      I2=MIN0(I1+349,675)                                               DATA 170
  640 READ(SCRA) (EBIAS(I),I=I1,I2)                                     DATA 171
  680 DO 700 N=1,6,2                                                    DATA 172
      ISTOR1=NSTORE(N)                                                  DATA 173
C READ ARC ADJUSTED PARAMETER VALUES                                     DATA 174
      ISTOR2=NSTORE(N+1)                                                DATA 175
C READ ARC PART OF NORMAL MATRIX                                         DATA 176
      DO 700 I1=ISTOR1,ISTOR2,450                                       DATA 177
      I2=MIN0(I1+449,ISTOR2)                                            DATA 178
  700 READ(SCRA) (BIAS(I),I=I1,I2)                                      DATA 179
      IF(NGPARC.LE.0) GO TO 800                                         DATA 180
      NGP=3*NGPARC                                                      DATA 181
      DO 750 I1=1,NGP,900                                               DATA 182
      I2=MIN0(I1+899,NGP)                                               DATA 183
  750 READ(SCRA) (INDXCS(I),I=I1,I2)                                    DATA 184
  800 CALL DREAD(SCRB,NSTART,IBUF,&2000)                                DATA 185
      DO 810 I=1,53                                                     DATA 186
  810 ELEMST(I)=IBUF(I)                                                 DATA 187
      DO 820 I=1,23                                                     DATA 188
      I1=I+53                                                           DATA 189
  820 DAYREF(I)=IBUF(I1)                                                DATA 190
      DO 830 I=1,48                                                     DATA 191
      I1=I+76                                                           DATA 192
  830 IEPYMD(I)=IBUF(I1)                                                DATA 193
      DO 840 I=1,192                                                    DATA 194
      I1=I+124                                                          DATA 195
  840 ELEMIN(I)=IBUF(I1)                                                DATA 196
      NPARAM=IBUF(317)                                                  DATA 197
      DO 900 J1=1,NWORD,450                                             DATA 198
      NSTART=NSTART+1                                                   DATA 199
      CALL DREAD(SCRB,NSTART,IBUF,&2000)                                DATA 200
      J2=MIN0(J1+449,NWORD)                                             DATA 201
      DO 900 J=J1,J2                                                    DATA 202
      I=J-J1+1                                                          DATA 203
  900 CD(J)=IBUF(I)                                                     DATA 204
      IF(.NOT.EMAT) RETURN                                              DATA 205
      IF(NPARAM.LE.0) RETURN                                            DATA 206
      L=2*NDIM*NPARAM-NPARAM*(NPARAM-1)                                 DATA 207
      DO 1000 I1=1,L,450                                                DATA 208
      I2=MIN0(I1+449,L)                                                 DATA 209
 1000 READ(SCRC) (SUM1(I),I=I1,I2)                                      DATA 210
      RETURN                                                            DATA 211
 2000 CALL ERROR(11,SCRB)                                               DATA 212
      RETURN                                                            DATA 213
      END                                                               DATA 214
```


ENTRY DATRD1
INITILIZATION
RETURN


ENTRY DATARD
STORE NORMAL MATRIX
NO
600
YES
SAVE APRIORI ARC INFORMATION
NO
200
YES
SAVE APRIORI ARC INFORMATION
A


A
SAVE INITIAL ESTIMATES OF ADJUSTED ARC PARAMETERS
200
ARE THERE ANY PARAMETERS
NO
RETURN
YES
SAVE ARC PART OF NORMAL MATRIX
STORE NORMAL MATRIX ON TAPE FOR RESTART
600
B



REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

DATBSE

DESCRIPTION

DATBSE is a subroutine specifically designed to read data from the DODS Data Base. It also has the capability to read data tapes in DODS Data Tape Format. DATBSE does not interpret the data it reads. It can read only observation data.

When reading data from the data base, DATBSE first searches the observation file directory record to ascertain if data is present for the satellite requested. If data is present DATBSE determines which portitions contain the desired data and then reads into core one physical record of data containing 35 observations and then returns a header record indicator through COMMON to the calling program. On all subsequent calls DATBSE returns through COMMON one observation per call until all of the requested data has been returned at which time it returns a sentinel record indicator.

DATBSE assumes that the first call for each arc requests data for a different satellite and searches through the observation file director to ascertain if data is available.

When no data is available or when read errors occur DATBSE prints error messages and terminates the run by calling ERROR.

NAME DATBSE

PURPOSE TO READ OBSERVATION DATA FROM DODS DATA BASE

CALLING SEQUENCE CALL DATBSE(IN,DSTART,DSTOP,ISATID,OLDARC)

SYMBOL	TYPE	DESCRIPTION
IN	I	INPUT - DATA FILE NUMBER
DSTART	DP	INPUT - DATA STARTS IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
DSTOP	DP	INPUT - DATA STOP TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
ISATID	I	INPUT - SATELLITE ID
OLDARC	L*1	INPUT & OUTPUT - FALSE ON FIRST CALL FOR EACH ARC CHANGED TO TRUE

SUBROUTINES USED CLEAR DJUL ERROR YMDAY

COMMON BLOCKS DODDAT

INPUT FILES IN - DATA FILE NUMBER

OUTPUT FILES OUTP - PRINTER

RESTRICTIONS DATA FILE NUMBER MUST BE 33 FOR DODS DATA BASE OTHERWISE DATA FILE IS ASSUMED SEQUENSIAL DODS FORMAT DATA TAPE

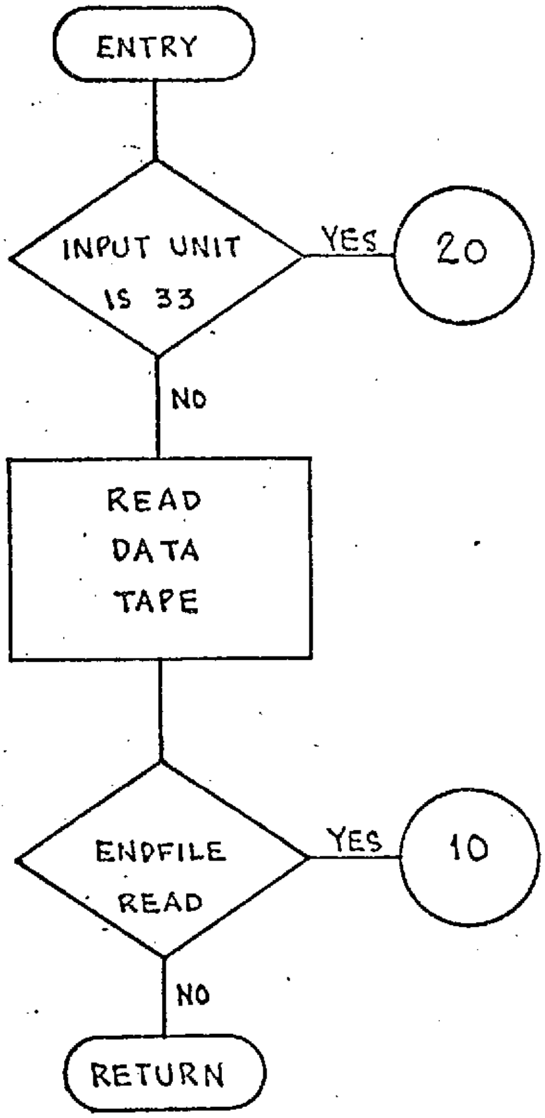
REFERENCES DODS DATA BASE OBSERVATION FILE DESCRIPTION 'GSFC DODS DOCUMENTATION'

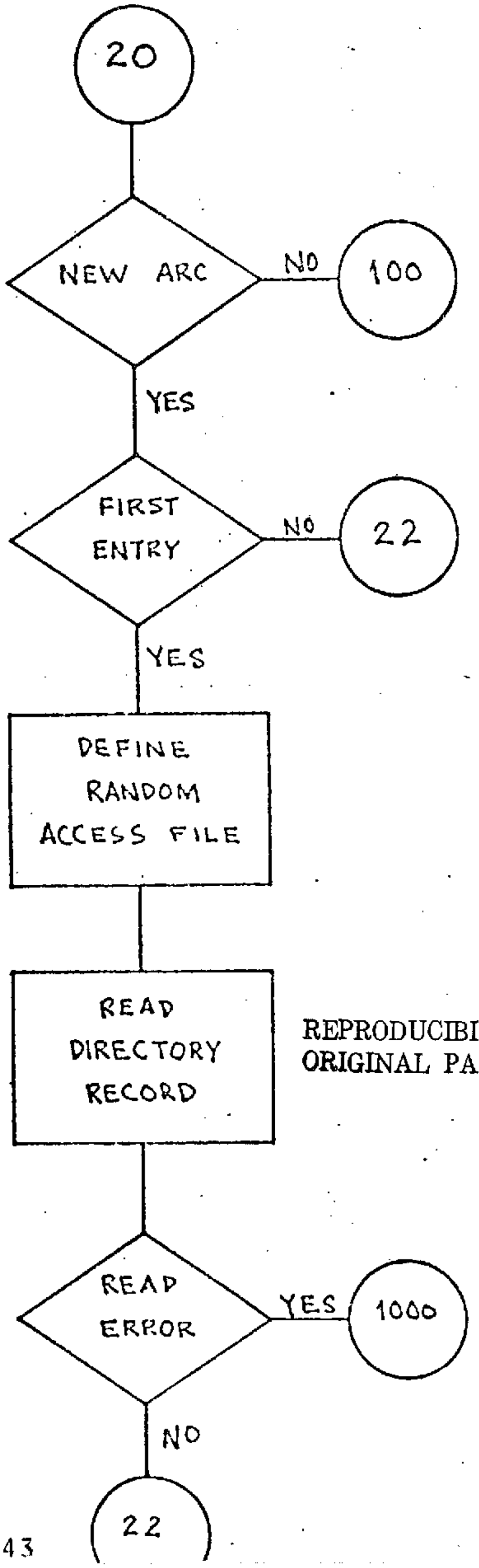
```
      SUBROUTINE DATBSE(IN,DSTART,DSTOP,ISATID,OLDARC)                  DATB  39
      REAL*8 TIME1,STNAM1,OBS1,DG(2),OBSCOR,DSTART,DSTOP,MJSTRT,MJSTOP, DATB  40
     1   MJD,DJUL,DJREF,YMDAY                                           DATB  41
      DIMENSION G(3),IDH(25),MREC(25,35)                                DATB  42
      LOGICAL*1 NOT1ST,OLDARC                                           DATB  43
      INTEGER ISATID,SATNO,RECORD(880),OUTP,ISAT,RECSAT                 DATB  44
      INTEGER*2 LREC,RECPAR,IG,IG1,PRIT1,PRIT2,IG2(2),IT,TTAG,LLREC1(2),DATB  45
     1   IWORD,REC2(1760)                                               DATB  46
      DIMENSION MRREC(25,35)                                            DATB  47
      COMMON/DODDAT/TIME1,STNAM1,OBS1,DG,OBSCOR,SATNO,IOBNO1,WT1,G,TCOR,DATB  48
     1   IG2,IT,IG1,TTAG,PRIT1,PRIT2,IG                                 DATB  49
      EQUIVALENCE (LREC,LLREC1(1),RECORD(1),REC2(1)),(IREC1,ICOUNT)     DATB  50
      EQUIVALENCE (IDH(1),TIME1)                                        DATB  51
      DATA DJREF,NOT1ST,J,OUTP,IERR/2433282.5D0,.FALSE.,-1,6,0/         DATB  52
      IF(IN.EQ.2) GO TO 20                                              DATB  53
C READ DATA TAPE                                                        DATB  54
    5 READ(IN,END=10,ERR=5) TIME1,STNAM1,OBS1,DG,OBSCOR,SATNO,IOBNO1,   DATB  55
```

```
     1     MT1,G,TCOR,IG2,IT,IG1,TTAG,PBIT1,PBIT2,IG
      OLDARC=.TRUE.
      RETURN
C SET SENTINEL RECORD INDICATOR
   10 TIME1=-2.D00
      RETURN
   20 IF(OLDARC) GO TO 100
      IF(NOT1ST) GO TO 22
      NOT1ST=.TRUE.
C DEFINE DATA BASE RANDOM ACCESS FILE
      DEFINE FILE 2(6609,3520,L,ICOUNT)
      ICOUNT=1
C READ DIRECTORY RECORD
      READ(IN'ICOUNT,ERR=1000) RECORD
      CALL CLEAR(M2REC,25,35)
      RECPAR=LLREC1(2)
   22 OLDARC=.TRUE.
C SEARCH DIRECTORY FOR SATELLITE
      DO 25 I=2,LREC
      IF(RECORD(I).GE.0) GO TO 25
      ISAT=-RECORD(I)
      IF(ISAT.EQ.ISATID) GO TO 30
   25 CONTINUE
      GO TO 1010
   30 NOBS=RECORD(I+1)-1
      RECSAT=(NOBS-1)/35+1
      NOPART=(RECSAT-1)/RECPAR+1
      IWORD=I+4
C CONVERT START TIME TO DODS SYSTEM
      MJSTRT=DJUL(DSTART)-DJREF-.00001D0
C CONVERT STOP TIME TO DODS SYSTEM
      MJSTOP=DJUL(DSTOP)-DJREF+.00001D0
      NWORD=IWORD+NOPART-1
C DETERMINE FIRST & LAST PARTITIONS TO BE READ
      DO 35 J=IWORD,NWORD
      KWORD2=2*J-1
      MJD=REC2(KWORD2)
      IF(MJSTRT.LT.(MJD+1.D0)) GO TO 40
   35 CONTINUE
C ONLY LAST PARTITION MIGHT HAVE DATA
      J=NWORD
      INOBS=NOBS-(NOBS-1)/35*35
      NOREC=RECSAT-RECSAT/RECPAR*RECPAR
      IREC1=REC2(KWORD2+1)+NOREC-1
C READ LAST PARTITION AND CHECK TIME
      READ(IN'ICOUNT,ERR=1030) MREC
      IOB(1)=MREC(1,INOBS)
      IOB(2)=MREC(2,INOBS)
      MJD=TIME1+YMDAY(370918,0,0.D0)+0.01D0
      IF(MJD.LT.DSTART) GO TO 1020
C FIRST PARTITION TO BE READ NOW DETERMINED
   40 IPART=MAX0(J-IWORD,1)
      KWORD2=2*(IWORD+IPART-1)
      IREC1=REC2(KWORD2)
      DO 45 K=J,NWORD
      KWORD2=2*K-1
```

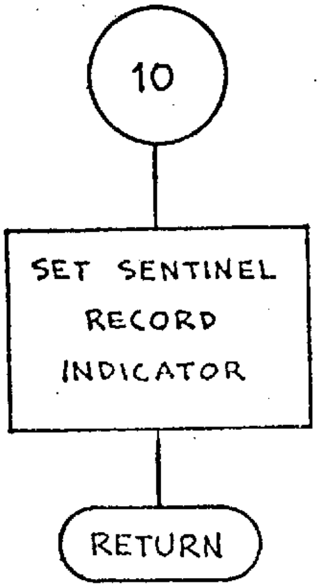
```
      MJD=REC2(LWORD2)
      IF(MJSTOP.LT.(MJD+1.D0)) GO TO 50
   45 CONTINUE
C LAST PARTITION TO BE READ NOW DETERMINED
   50 KPART=MIN0(NOPART,K-INCRD+1)
      NREC=IREC1+RECPAR-1
      NDREC=RECSAT-RECSAT/RECPAR*RECPAR
      IF(NDREC.EQ.0) NDREC=RECPAR
      IF(IPART.EQ.KPART) NREC=IREC1+NDREC-1
      INDOB=35
      NOB=0
      NDOB=NOBS-NOBS/35*35
      IF(NDOB.EQ.0) NDOB=35
      SATNO=ISAT
C SET HEADER RECORD INDICATOR
      TIME1=-1.D0
      LABEL=1
C READ FIRST 35 DATA RECORDS
      READ(IN'ICOUNT,ERR=1030)  MREC
      LABEL=2
      IF(IPART.NE.KPART) RETURN
      NREC=IREC1+NDREC-1
      IF(NDREC.EQ.1) INDOB=NDOB
      RETURN
  100 NOB=NOB+1
      IF(NOB.GT.INDOB) GO TO 150
C CHECK FOR DUPLICATE OBSERVATIONS
      DO 110 I=1,35
      IF(MREC(1,NOB).EQ.M2REC(1,I).AND.MREC(2,NOB).EQ.M2REC(2,I).AND.
     1   MREC(3,NOB).EQ.M2REC(3,I).AND.MREC(4,NOB).EQ.M2REC(4,I).AND.
     2   MREC(23,NOB).EQ.M2REC(23,I)) GO TO 100
  110 CONTINUE
C SELECT ONE OBSERVATION FROM CORE AND LOAD IN COMMON
      DO 120 I=1,23
  120 IOB(I)=MREC(I,NOB)
      RETURN
C SAVE PREVIOUS 35 OBSERVATIONS
  150 DO 160 I=1,23
      DO 160 J=1,35
  160 M2REC(I,J)=MREC(I,J)
      IF(ICOUNT.GT.NREC) GO TO 200
C READ 35 MORE OBSERVATIONS
      READ(IN'ICOUNT,ERR=1030) MREC
      NOB=0
      IF(IPART.EQ.KPART.AND.ICOUNT.EQ.NREC) INDOB=NDOB
      GO TO 100
  200 IF(IPART.GE.KPART) GO TO 10
C DETERMINE FIRST RECORD IN NEW PARTITION
      NOB=0
      IPART=IPART+1
      KWORD2=KWORD2+2
      ICOUNT=REC2(KWORD2)
      NREC=ICOUNT+RECPAR-1
      IF(IPART.LT.KPART) GO TO 100
      NREC=ICOUNT+NDREC-1
      IF(NDREC.EQ.1) INDOB=NDOB
```

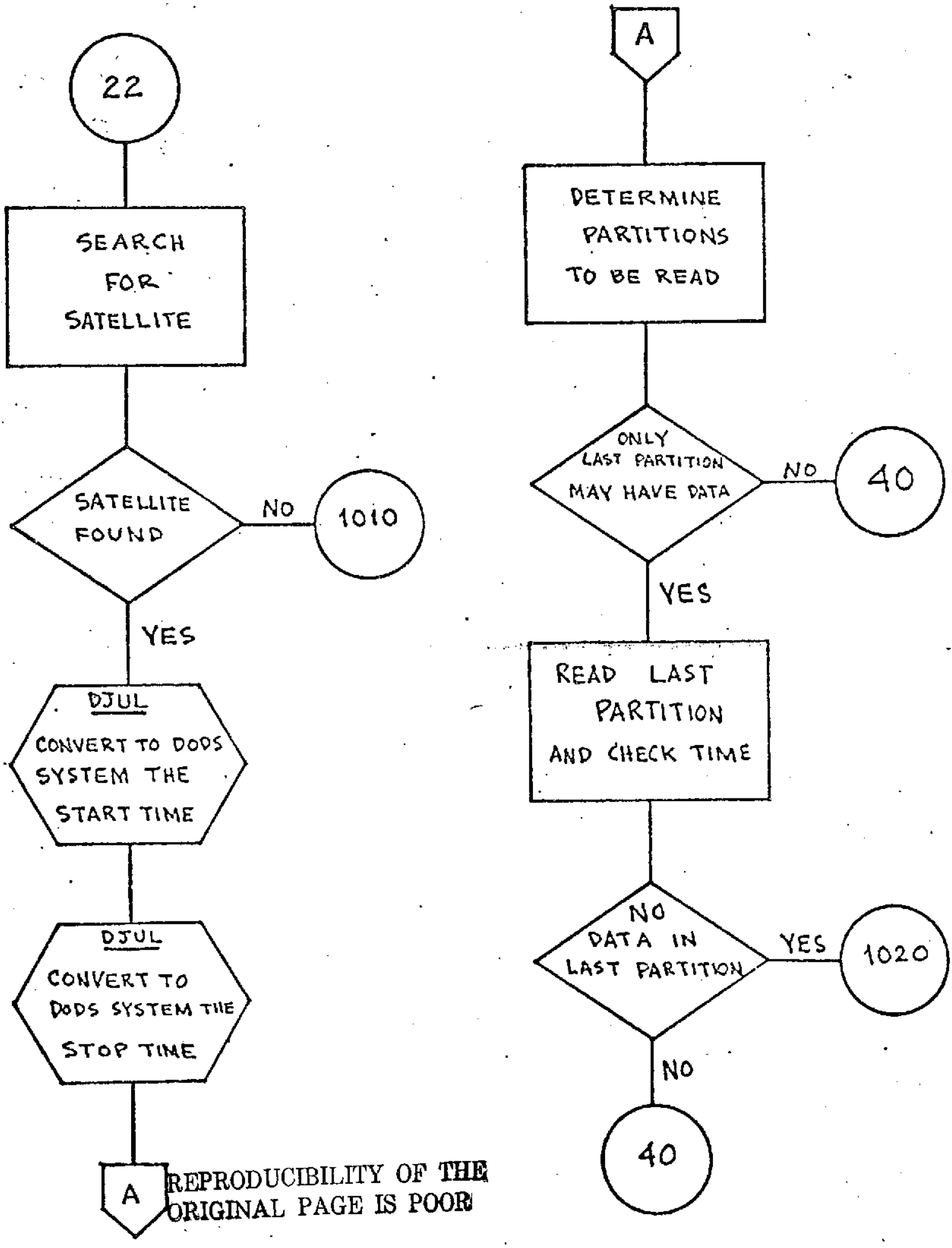
```
      GO TO 100
C PRINT ERROR MESSAGES
 1000 WRITE(OUTF,2000) J
      GO TO 1035
 1010 WRITE(OUTF,2010) ISATID
      GO TO 1035
 1020 WRITE(OUTF,2020) ISATID
      GO TO 1035
 1030 IERR=IERR+1
      WRITE(OUTF,2030) ISATID,ICOUNT
      IF(IERR.LT.10) GO TO (1040,150),LABEL
      WRITE(OUTF,2035)
 1035 WRITE(OUTF,2040) RECORD,MREC
C TAKE ERROR EXIT
      CALL ERROR(7,CSTART)
 1040 IF(IPART.EQ.KPART) NREC=NDREC
      RETURN
 2000 FORMAT(1H1,20X,'UNABLE TO READ DODS DATA BASE DATA DIRECTORY'/
     1    21X,'ERROR CODE IS',I9/21X,'NONAME EXECUTION TERMINATED')
 2010 FORMAT(1H1,20X,'NO DATA AVAILABLE FOR SATELLITE REQUESTED'/21X,
     1    'SATELLITE ID IS',I9/21X,'NONAME EXECUTION TERMINATED')
 2020 FORMAT(1H1,20X,'NO DATA AVAILABLE FOR THE TIME PERIOD REQUESTED ',
     1    'FOR'/21X,'SATELLITE',I9/21X,'NONAME EXECUTION TERMINATED')
 2030 FORMAT(1H0,20X,'DODS DATA BASE READ ERROR OCCURED FOR'/21X,
     1    'SATELLITE',I9/21X,'DATA RECORD NUMBER IS',I9/)
 2035 FORMAT(1H0,20X,'NONAME EXECUTION TERMINATED DUE TO'/21X,
     1    'EXCESSIVE DODS DATA BASE READ ERRORS')
 2040 FORMAT(1H1/(10X,Z8,1X,Z8,2X,Z8,1X,Z8,4X,Z8,1X,Z8,2X,Z8,1X,Z8))
      END
```



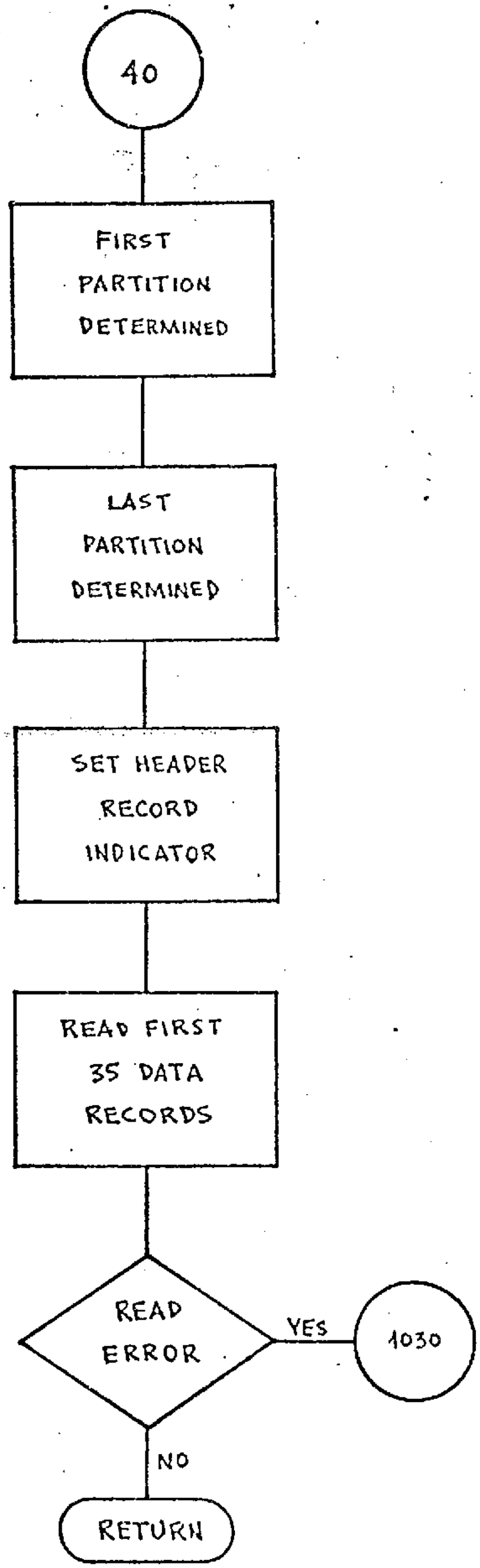


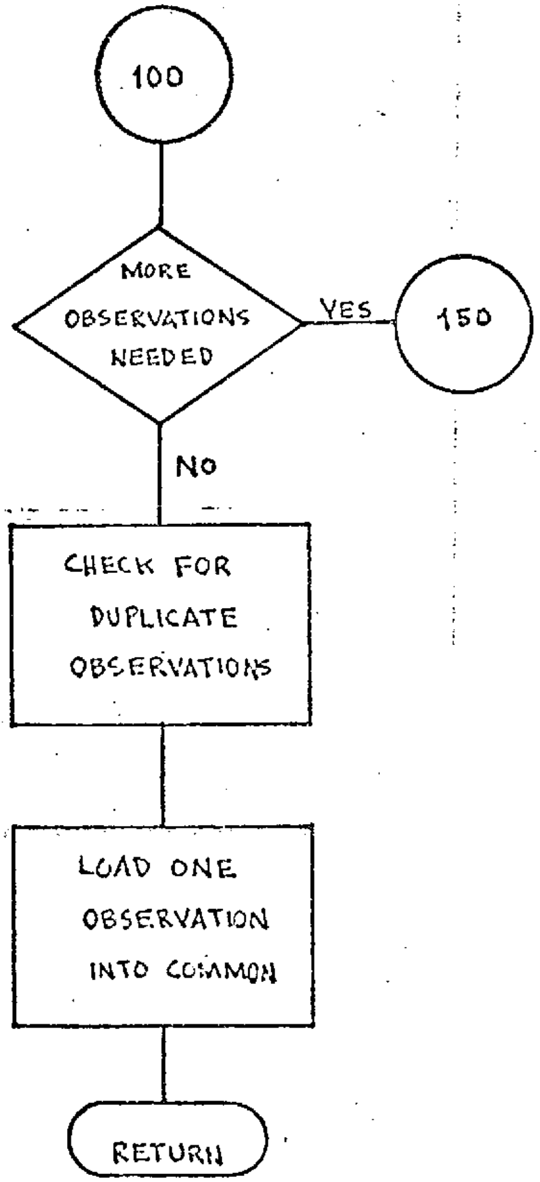
REPRODUCIBILITY OF T
ORIGINAL PAGE IS POC

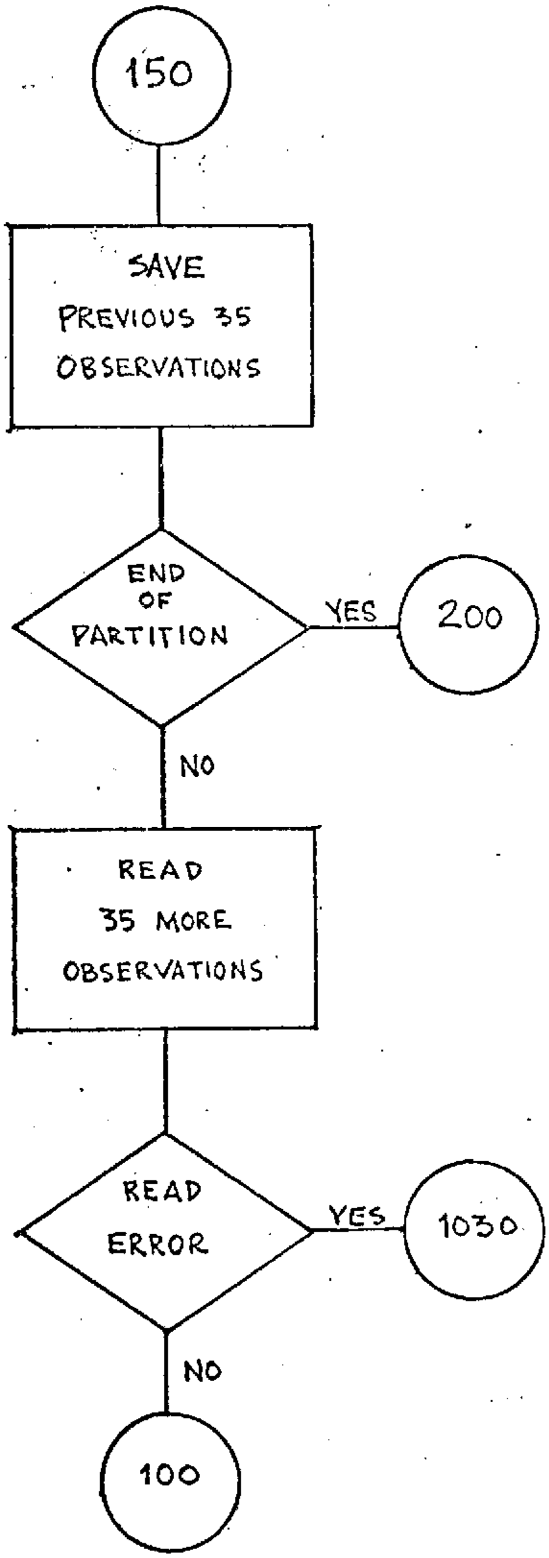


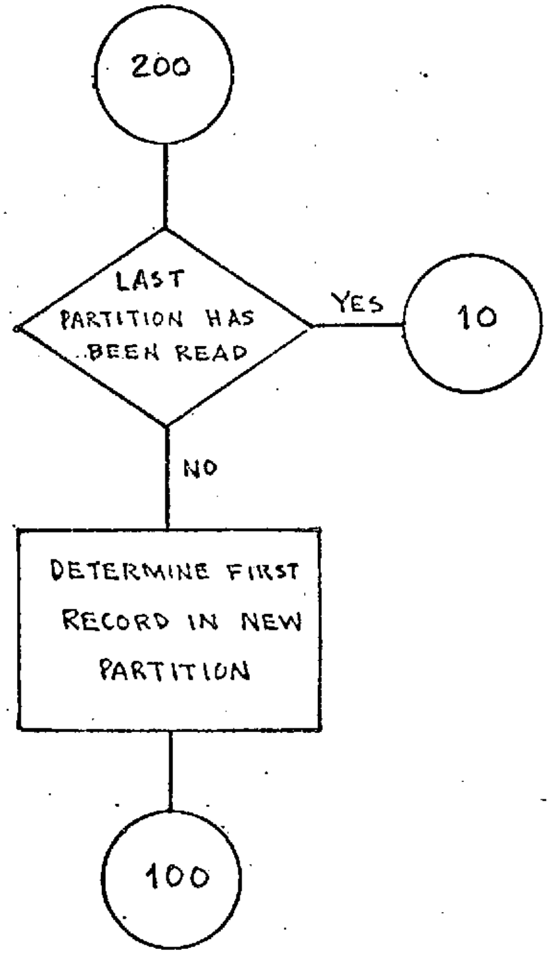


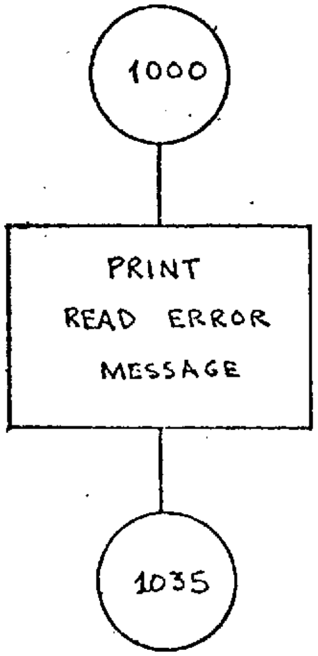
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

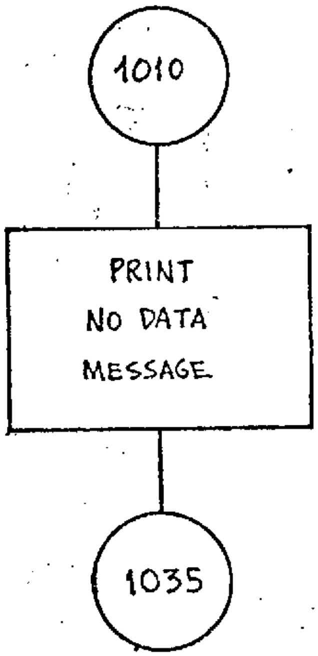

40
FIRST PARTITION DETERMINED
LAST PARTITION DETERMINED
SET HEADER RECORD INDICATOR
READ FIRST 35 DATA RECORDS
READ ERROR
YES
1030
NO
RETURN

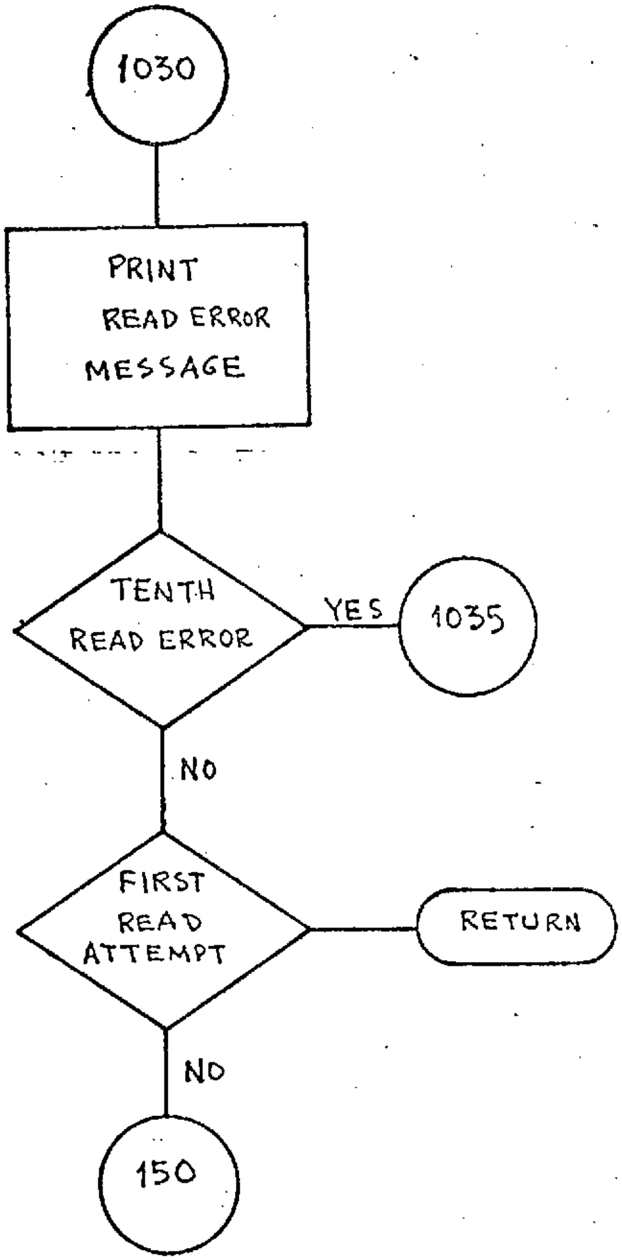

100
MORE OBSERVATIONS NEEDED
YES
150
NO
CHECK FOR DUPLICATE OBSERVATIONS
LOAD ONE OBSERVATION INTO COMMON
RETURN

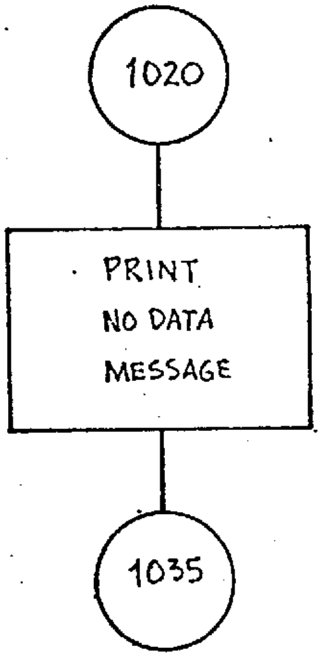

150
SAVE PREVIOUS 35 OBSERVATIONS
END OF PARTITION
YES
200
NO
READ 35 MORE OBSERVATIONS
READ ERROR
YES
1030
NO
100

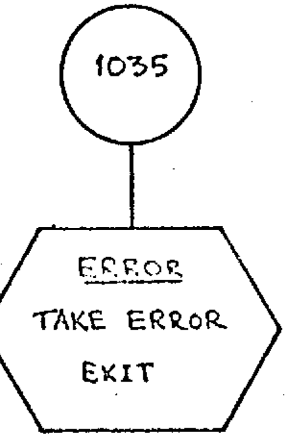

200
LAST PARTITION HAS BEEN READ
YES
10
NO
DETERMINE FIRST RECORD IN NEW PARTITION
100


1000
PRINT READ ERROR MESSAGE
1035


1010
PRINT
NO DATA
MESSAGE
1035


1030
PRINT
READ ERROR
MESSAGE
TENTH
READ ERROR
YES
1035
NO
FIRST
READ
ATTEMPT
RETURN
NO
150


1020
PRINT
NO DATA
MESSAGE
1035


1035
ERROR
TAKE ERROR
EXIT

80-147

NAME DATES

PURPOSE TO CONVERT DAYS ELAPSED FROM JAN 0.0 OF THE ARC REFERENCE YEAR INTO A 3 WORD DATE OF THE FORM :
YYMMDD HHMM SEC

CALLING SEQUENCE CALL DATES(DAYNR,IYMD,IHM,SEC)

SYMBOL	TYPE	DESCRIPTION
DAYNR	DP	INPUT - DAYS ELAPSED FROM JAN 0.0 OF THE REFERENCE YEAR
IYMD	I	OUTPUT - YEAR, MONTH, DAY IN THE FORM OF YYMMDD
IHM	I	OUTPUT - HOUR, MINUTE IN THE FORM OF HHMM
SEC	R	OUTPUT - SECONDS

SUBROUTINES USED ADDYMD TDIF

COMMON BLOCKS CTIME

INPUT FILES NONE

OUTPUT FILES NONE

```
      SUBROUTINE DATES(DAYNR,IYMD,IHM,SEC)
      DOUBLE PRECISION DAYNR,S,DAY,SEC
      COMMON/CTIME/DAYREF(22),IY
C CONVERT TO UTC TIME SYSTEM
      DAY=DAYNR+(DBLE(TDIF(3,4,DAYNR))+1.0D-5)/8.64D4
      IDAY=DAY-1.D0
      IYMD=IY*10000+101
C CALCULATE YEAR/MONTH/DAY OF INTEREST
      CALL ADDYMD(IYMD,IDAY)
C CALCULATE NUMBER OF SECONDS REMAINING
      S=8.64D4*(DAY-DFLOAT(IDAY+1))
      ISEC=S
C CONVERT TO HOUR/MINUTE FORMAT
      IHM=40*(ISEC/3600)+ISEC/60
C REMAINING SECONDS
      SEC=S-DFLOAT(60*(ISEC/60))-1.0D-5
      RETURN
      END
```

NAME DAYEAR

PURPOSE TO COMPUTE A DATE, DAY, NUMBER OF YEARS, AND TIME OF DAY IN SECONDS FROM A GIVEN TIME IN DAYS FROM JAN 0.0 OF REFERENCE YEAR

CALLING SEQUENCE CALL DAYEAR(DAY,IYMD,IDAY,ISEC)

SYMBOL	TYPE	DESCRIPTION
DAY	DP	INPUT - DAYS ELAPSED FROM JAN 0.0 OF THE REFERENCE YEAR
IYMD	I	INPUT - YEAR, MONTH, DAY IN THE FORM OF YYMMDD
IDAY	I	OUTPUT - DAY NUMBER OF YEAR
ISEC	I	OUTPUT - FRACTION OF DAY CONVERTED TO INTEGRAL SECONDS

SUBROUTINES USED DATES DIFF

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

```
      SUBROUTINE DAYEAR(DAY,IYMD,IDAY,ISEC)
      REAL*8 DAY,SEC
      CALL DATES(DAY,IYMD,IHM,SEC)
      JAN0=IYMD/10000*10000+100
      IHMS=IHM*100+IDINT(SEC+0.5D0)
      CALL DIFF(JAN0,0,IYMD,IHMS,IDAY,ISEC)
      RETURN
      END
```

NAME		DELTAZ
PURPOSE		TO COMPUTE THE Z COORDINATE OF A POINT OF GIVEN LATITUDE ON THE ELLIPSOID
CALLING SEQUENCE		X=DELTAZ(SUBLAT)
SYMBOL	TYPE	DESCRIPTION
SUBLAT	DP	INPUT - LATITUDE OF POINT ON THE ELLIPSOID
SUBROUTINES USED		NONE
COMMON BLOCKS		CONSTS INTBLK
INPUT FILES		NONE
OUTPUT FILES		NONE

```
      DOUBLE PRECISION FUNCTION DELTAZ(SUBLAT)
      IMPLICIT REAL*8(A-H,O-Z)
      LOGICAL NOT1ST
      COMMON/CONSTS/DPI,DTWOPI,DDG2RD,DESEC
      COMMON/INTBLK/THDOTS(3),GM,AE,AESQ,FLAT,FSQ32(59)
      DATA NOT1ST/.FALSE./
      IF(NOT1ST) GO TO 10
      NOT1ST=.TRUE.
      F1=1.D0-FLAT
      F1SQ=F1*F1
      F2F=FLAT*(2.D0-FLAT)
   10 SPSI=DSIN(SUBLAT*DDG2RD)
      SPSISQ=SPSI**2
      EARTH=AE*F1/DSQRT(F1SQ+F2F*SPSISQ)
      DELTAZ=SPSI*EARTH
      RETURN
      END
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

DENSTY

DESCRIPTION

DENSTY is a real valued function whose value is the atmospheric density at a given time for a given satellite position. Additionally, DENSTY computes the partial derivative of the atmospheric density with respect to the spheroid height.

The density model used is the Jacchia 1971 Static Density Model. A thorough mathematical description and a list of references are provided in the GEODYN Systems Description, Volume I.

```
NAME                  DENSTY

PURPOSE               THE FUNCTION DENSTY COMPUTES AN ATMOSPHERIC DENSITY
                      WHICH IS DEPENDENT UPON HEIGHT, TEMPERATURE, AND
                      SEASONAL LATITUDINAL VARIATIONS.  TEMPERATURE IS
                      DERIVED FROM THE JACCHIA 1971 MODEL AND IS
                      DEPENDENT UPON THE FOLLOWING:
                           1.  SOLAR ACTIVITY
                           2.  GEOMAGNETIC ACTIVITY AND
                           3.  DIURNAL BULGE
                      SEMIANNUAL VARIATIONS OF DENSITY ARE EXPRESSED AS
                      A PRODUCT FUNCTION OF HEIGHT AND TIME.  THERE ARE
                      SEASONAL LATITUDINAL VARIATIONS OF THE LOWER
                      THERMOSPHERE AND HELIUM

CALLING SEQUENCE      X=DENSITY(RASAT)

   SYMBOL   TYPE      DESCRIPTION

   RASAT    DP        INPUT - RIGHT ASCENSION OF SATELLITE

   X        R         OUTPUT - ATMOSPHERIC DENSITY - KILOGRAMS/METERS**3

SUBROUTINES USED      CPHEM,      YMDAY

COMMON BLOCKS         CEPHEM      CONSTS      FLXBLK      XYZ      CTIME
                      DRGBLK      INTBLK

INPUT FILES           NONE

OUTPUT FILES          NONE

RESTRICTIONS          NONE

REFERENCES            'GEODYN SYSTEMS DESCRIPTION'
                      VOLUME 1 - GEODYN DOCUMENTATION
```

```
      FUNCTION DENSTY(RASAT)                                            DENS  41
C UPDATED AS OF APRIL 1972                                              DENS  42
      IMPLICIT REAL*8 (A-H,O-Z)                                         DENS  43
      DOUBLE PRECISION KP,KP3,LN10                                      DENS  44
      INTEGER ORDER,START                                               DENS  45
      LOGICAL NOT1ST                                                    DENS  46
      COMMON/CEPHEM/G3(4),UVSUN(4),GA(660)                              DENS  47
      COMMON/CONSTS/PI,TWOPI,RAD,RSEC                                   DENS  48
      COMMON/FLXBLK/AVFLX(675),DFLX(675),KP(675)                        DENS  49
      COMMON/XYZ/X,Y,Z,DOTS(3),R,RSQ,ISAT,IFORCE(3)                     DENS  50
      COMMON/CTIME/DATAFR,DSTART(7),DAY1,IDAYYM(13)                     DENS  51
      COMMON/DRGBLK/HT,SPSISQ,C(4),BDD(5),DDRAG                         DENS  52
      COMMON/INTBLK/IS1(40),NOT1ST,IS2(9)                               DENS  53
      DIMENSION DA(3),DC(3),TC(3),EXPKP(3),A(114),B(68),AA(50),BB(20)   DENS  54
      EQUIVALENCE (AA(1),A(65)),(BB(1),B(49))                           DENS  55
```

8.0-152

```
      EQUIVALENCE (PCPHOD,C(1))
      DATA BETA,GAMA,RX,P/-0.6457718232377D0,0.7504915783574D0,
     .    0.3D0,0.1047197551196D0/
      DATA EPSIL,FEAVRG/0.40910516667D0,0.6645999963D-23/
      DATA PI4,SIN3/0.78539816339745D0,0.35355339059327D0/
      DATA DJUL50/2433281.5D0/
      DATA ANGLAT/0.2617993877980D0/
      DATA N3/1/
C TEMPERATURE COEFFICIENTS FOR DENSITY AT DIFFERENT HEIGHT RANGES   DENS
      DATA A/
C 90<HT<200     500<T<1900
     .0.4220893650D+01,0.9439283440D-05,-.69521277D-05,0.1471533540D-08,
     .-.2013423560D+00,-.2341199440D-03,0.1573671120D-06,-.3467492740D-10,
     .0.7359131530D-03,0.1596613990D-05,-.1106041890D-08,0.2503685370D-12,
     .-.1209743620D-05,-.3435599820D-08,0.2245708330D-11,-.5106914110D-15,
C 200<HT<500    500<T<800
     .-.1283875330D+02,0.4070898200D-02,0.9727406610D-05,-.1064251860D-07,
     .0.9228213700D-01,-.3121502920D-03,0.2654264830D-06,-.3519333380D-10,
     .-.6595051460D-03,0.2440152170D-05,-.2705514170D-08,0.9900349670D-12,
     .0.1126273290D-05,-.4180717330D-08,0.5061712120D-11,-.2048373140D-14,
C 200<HT<500    800<T<1900
     .-.8459523220D+01,-.1500046660D-03,-.6264019480D-06,0.2461185090D-09,
     .-.2829431310D-01,0.1776014330D-05,0.6139771390D-08,-.3376154120D-11,
     .0.5599787360D-05,0.7746122330D-07,-.5949204400D-10,0.1492369330D-13,
     .0.1943413460D-08,-.7643521590D-10,0.5833255590D-13,-.1459317730D-16,
C 500<HT<1000   500<T<800
     .-.7753870020D+02,0.1672711980D+00,-.5657018260D-04,-.5042399400D-07,
     .0.3063819930D+00,-.9893629900D-03,0.7493197180D-06,-.1531782724D-10,
     .-.7893435570D-03,0.1297261930D-05,-.1077646500D-08,0.1419093070D-12,
     .0.1596203960D-06,-.5402489521D-09,0.4670859710D-12,-.7188646880D-16/
      DATA AA/
C 500<HT<1000   900<T<1900
     .0.5003811708D+02,-.1260123610D+00,0.8289572940D-04,-.1827624890D-07,
     .-.3057102100D+00,0.6179574550D-03,-.4144259670D-06,0.9109592280D-10,
     .0.4176635600D-03,-.8874291830D-06,0.6103974560D-09,-.1363390960D-12,
     .-.1796402180D-06,0.3938629180D-09,-.2763926920D-12,0.6264914390D-16,
C 1000<HT<2500   500<T<800
     .0.3653278630D+02,-.2619153250D+00,0.4196739640D-03,-.2165121110D-06,
     .-.4635213260D-01,0.2590144140D-03,-.4821295580D-06,0.2709525420D-09,
     .0.1114135130D-04,-.7743977560D-07,0.1604330770D-09,-.9905498570D-13,
     .-.2505590050D-10,0.4472476860D-11,-.1408538370D-13,0.1044294020D-16,
C 1000<HT<2500   900<T<1900
     .0.8241012090D+02,-.2065246860D+00,0.2164156170D-03,-.9062287610D-07,
     .0.1305413230D-10,
     .-.1435455970D+00,0.4311734590D-03,-.4613652400D-06,0.2017886710D-09,
     .-.3098825530D-13,
     .0.8779924150D-04,-.2715985900D-06,0.2974526080D-09,-.1342498850D-12,
     .0.2136953420D-16,
     .-.1571624340D-07,0.4963366830D-10,-.5529740780D-13,0.2543182980D-16,
     .-.4130379740D-20/
C TEMPERATURE COEFFICIENTS FOR HELIUM NUMBER DENSITIES
      DATA D/
C 90<HT<1000     500<T<800        HELIUM
     .0.9371275540D+01,-.5252363370D-02,0.5799790020D-05,-.2047064920D-08,
     .-.1314075050D-01,0.3181432640D-04,-.3259769440D-07,0.1257251200D-10,
     .0.2607133710D-05,-.7572004770D-08,0.9715756230D-11,-.4069390070D-14,
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
     ,-.5215622920D-09,0.1905532270D-11,-.2657923900D-14,0.1253486450D-17,   DENS 1112
C 500<HT<1000    800<T<1900      HELIUM                                      DENS 1113
     ,0.8391376410D+01,-.1642339510D-02,0.7803184720D-06,-.1432273880D-09,   DENS 1114
     ,-.6924941910D-02,0.8413824440D-05,-.4457713850D-08,0.3562711870D-12,   DENS 1115
     ,0.1051034730D-05,-.1266287050D-08,0.7113766110D-12,-.1417052180D-15,   DENS 1116
     ,-.1222207890D-09,0.1474528620D-12,-.9765793350D-16,0.2149831500D-19,   DENS 1117
C 1000<HT<2500    500<T<800       HELIUM                                     DENS 1118
     ,0.9104591260D+01,-.4340957170D-02,0.4029304590D-05,-.1453154360D-08,   DENS 1119
     ,-.1225734690D-01,0.2795196980D-04,-.2797203680D-07,0.1037036640D-10,   DENS 1120
     ,0.1539236780D-05,-.3556255140D-08,0.3947612740D-11,-.1298531940D-14,   DENS 1121
     ,-.1102883080D-09,0.2513780030D-12,-.2522657630D-15,0.8971361000D-19/   DENS 1122
      DATA BD/                                                               DENS 1123
C 1000<HT<2500    800<T<1900      HELIUM                                     DENS 1124
     ,0.8612023040D+01,-.2536333410D-02,0.1997932140D-05,-.7369615190D-09,   DENS 1125
     ,0.1133737680D-12,                                                      DENS 1126
     ,-.8434715420D-02,0.1408766970D-04,-.1138574150D-07,0.4487054680D-11,   DENS 1127
     ,-.6906420770D-15,                                                      DENS 1128
     ,0.1154317610D-05,-.1988309630D-08,0.1663547610D-11,-.6762777690D-15,   DENS 1129
     ,0.1070674500D-18,                                                      DENS 1130
     ,-.9452073010D-10,0.1738700190D-12,-.1536798770D-15,0.6540159030D-19,   DENS 1131
     ,-.1076048010D-22/                                                      DENS 1132
C  INITIALIZE DENSITY CORRECTIONS                                            DENS 1133
      HELIUM=0.0D0                                                           DENS 1134
      DENS=0.0D0                                                             DENS 1135
      DO 500 J=1,9                                                           DENS 1136
  500 C(J)=0.D0                                                              DENS 1137
      IF(NOT1ST) GO TO 1                                                     DENS 1138
      DAY50=YMDAY(500100,0,0.0D0)                                            DENS 1139
      FORPI=4.0D0*PI                                                         DENS 1140
      LN10=DLOG(10.0D0)                                                      DENS 1141
      IND3=-1                                                                DENS 1142
      IX0=-1                                                                 DENS 1143
      IX1=-1                                                                 DENS 1144
      IX2=-1                                                                 DENS 1145
      NOT1ST=.TRUE.                                                          DENS 1146
    1 HT2=HT*1.0D-3                                                          DENS 1147
      HT=HT*1.D-5                                                            DENS 1148
      IF(HT.LT.25.0D0) GO TO 10                                              DENS 1149
      DENSTY=0.0D0                                                           DENS 1150
      RETURN                                                                 DENS 1151
   10 ID=DATA58                                                              DENS 1152
      DAYS=DAY1-ID                                                           DENS 1153
      IX=DAYS+1.5D0                                                          DENS 1154
      IF(IX.LT.1) IX=1                                                       DENS 1155
      IF(IX.GT.673) IX=673                                                   DENS 1156
      I=IX                                                                   DENS 1157
      IF(IX.EQ.IX1) GO TO 20                                                 DENS 1158
      K=1                                                                    DENS 1159
      IF(IX.EQ.IX2) GO TO 12                                                 DENS 1160
      IF(IX.EQ.IX0) GO TO 14                                                 DENS 1161
      J=1                                                                    DENS 1162
      K=0                                                                    DENS 1163
      GO TO 16                                                               DENS 1164
   12 RA(1)=RA(2)                                                            DENS 1165
      DC(1)=DC(2)                                                            DENS 1166
      TC(1)=TC(2)                                                            DENS 1167
```

```
      IX=IX+1                                                         DENS 168
      J=2                                                             DENS 169
      GO TO 16                                                        DENS 170
   14 RA(2)=RA(1)                                                     DENS 171
      DC(2)=DC(1)                                                     DENS 172
      TC(2)=TC(1)                                                     DENS 173
      J=1                                                             DENS 174
   16 IX0=I-1                                                         DENS 175
      IX1=I                                                           DENS 176
      IX2=I+1                                                         DENS 177
   15 CONTINUE                                                        DENS 178
C COMPUTE GLOBAL NIGHTTIME MINIMUM TEMPERATURE AT 12 HOURS GMT          DENS 179
C DAILY SOLAR FLUX VALUE USED IS FOR ONE DAY EARLIER                    DENS 180
      TC(J)=379.D0+3.24D0*AVFLX(IX+1)+1.3D0*(DFLX(IX)-AVFLX(IX+1))   DENS 181
C COMPUTE RIGHT ASCENSION AND DECLINATION OF SUN AT 12 HOURS GMT        DENS 182
      FDAY=ID+IX-1                                                    DENS 183
      CALL EPHEM(FDAY,.FALSE.)                                        DENS 184
      DC(J)=DARSIN(UVSUN(3))                                          DENS 185
      RA(J)=DATAN2(UVSUN(2),UVSUN(1))                                 DENS 186
C  COMPUTE VARIABLES FOR INTERPOLATING VALUES.                          DENS 187
      IF(K.EQ.1) GO TO 19                                             DENS 188
      IX=IX+1                                                         DENS 189
      J=2                                                             DENS 190
      K=1                                                             DENS 191
      GO TO 15                                                        DENS 192
   19 RA(3)=RA(2)-RA(1)                                               DENS 193
      DC(3)=DC(2)-DC(1)                                               DENS 194
      TC(3)=TC(2)-TC(1)                                               DENS 195
   20 FDAY=DAY1-DFLOAT(ID+I-1)                                        DENS 196
      DECS=DC(1)+DC(3)*FDAY                                           DENS 197
      RAS=RA(1)+RA(3)*FDAY                                            DENS 198
      T3=TC(1)+TC(3)*FDAY                                             DENS 199
C  COMPUTE 6.7 HOUR LAG IN GEOMAGNETIC EFFECTS WRT GEOMAGNETIC INDEX    DENS 200
C GEOMAGNETIC INDEX IS REFERENCED TO 12 HOURS GMT                       DENS 201
      DAYLAG=DAYS-0.27916666666D0                                     DENS 202
      IDAY1=DAYLAG+1.0D0                                              DENS 203
      FDAY1=DAYLAG-DFLOAT(IDAY1-1)                                    DENS 204
      S=FDAY1*DFLOAT(N3)                                              DENS 205
      INT1=S                                                          DENS 206
      IND1=IDAY1*N3+INT1+1                                            DENS 207
      IF(IND3.EQ.IND1) GO TO 60                                       DENS 208
      IF(IND3.LT.0) EXPKP(2)=DEXP(KP(IND1))                           DENS 209
      EXPKP(1)=EXPKP(2)                                               DENS 210
      EXPKP(2)=DEXP(KP(IND1+1))                                       DENS 211
      IND3=IND1                                                       DENS 212
C  COMPUTE GEOMAG EFFECTS. IF ALTITUDE IS LOWER THAN 200 KM USE HYBRID  DENS 213
C  FORMULAS.  GEOMAG=TEMP. CORRECTION***DENMAG=LOG DENSITY CORRECTION   DENS 214
   60 S=S-DFLOAT(INT1)                                                DENS 215
      KP3=KP(IND1)+S*(KP(IND1+1)-KP(IND1))                            DENS 216
      EXPKP3=EXPKP(1)+S*(EXPKP(2)-EXPKP(1))                           DENS 217
      IF(HT.LT.2.0D0) GO TO 30                                        DENS 218
      GEOMAG=28.D0*KP3+0.03D0*EXPKP3                                  DENS 219
      GO TO 32                                                        DENS 220
   30 GEOMAG=14.D0*KP3+0.02D0*EXPKP3                                  DENS 221
      DENMAG=0.012D0*KP3+1.2D-5*EXPKP3                                DENS 222
      DENS=DENS+DENMAG                                                DENS 223
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

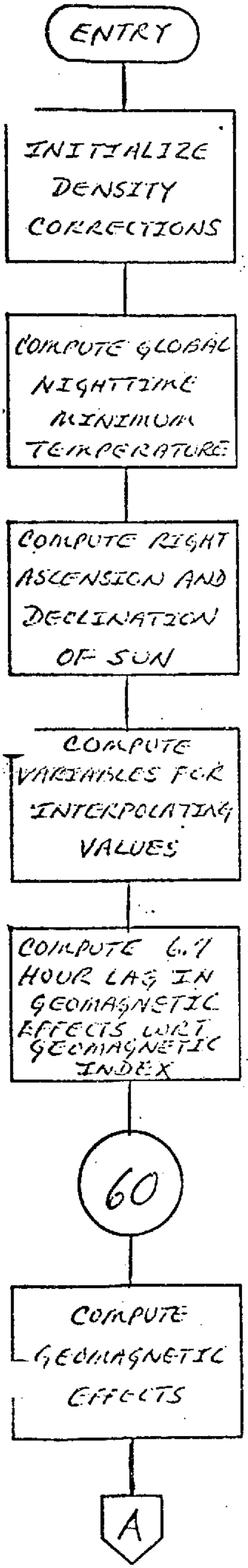
```
   32 CONTINUE                                                             DENS 224
C     COMPUTE DIURNAL VARIATION OF TEMPERATURE                             DENS 225
      HRANG=RASAT-RAS                                                      DENS 226
      IF(HRANG.LT.0.D0)HRANG=HRANG+TWOPI                                   DENS 227
      PHI=Z/R                                                              DENS 228
      SPHI=PHI*PHI                                                         DENS 229
      PHI=DARSIN(PHI)                                                      DENS 230
      THETA=DABS(PHI+DECS)*0.5D0                                           DENS 231
      ETA=DABS(PHI-DECS)*0.5D0                                             DENS 232
      TAU=HRANG+BETA+P*DSIN(HRANG+GAMA)                                    DENS 233
      CE=DCOS(ETA)**2.2D0                                                  DENS 234
      ST=DSIN(THETA)**2.2D0                                                DENS 235
      STT=1.0D0+RX*ST                                                      DENS 236
      TAU=DMOD(TAU+TWOPI,TWOPI)                                            DENS 237
      IF(TAU.GT.PI) TAU=TAU-TWOPI                                          DENS 238
      CT=DCOS(TAU*0.5D0)                                                   DENS 239
      CT=CT**3                                                             DENS 240
C     CORRECTION FOR A GIVEN HOUR AND GEOGRAPHIC LOCATION                  DENS 241
      TL=(T3*STT)*(1.D0+RX*(CE-ST)/STT*CT)                                 DENS 242
      TEM=TL+GEOMAG                                                        DENS 243
      TEM4=TEM**4                                                          DENS 244
C     MODIFIED JULIAN DATE OF 36204 IS JAN. 1, 1958                        DENS 245
      T=DAY1-DAY50+DJUL50-2400000.5D0                                      DENS 246
      PHASE=(T-36204.D0)/365.2422D0                                        DENS 247
C     CORRECTION TO LOG OF DENSITY DUE TO SEMIANNUAL VARIATIONS            DENS 248
      FZA=-2.868D-3*HT2                                                    DENS 249
      IF(FZA.LT.-20.0D0)GO TO 33                                           DENS 250
      EXPFZA=DEXP(FZA)                                                     DENS 251
      HI331=HT2**1.331D0                                                   DENS 252
      FZ=(5.876D-7*(HT2*HI331)+0.06328D0)*EXPFZA                           DENS 253
      DFZDZ=-2.868D-3*FZ+5.876D-7*2.331D0*HI331*EXPFZA                     DENS 254
      PH=TWOPI*PHASE+6.035D0                                               DENS 255
      PH=DMOD(PH,TWOPI)                                                    DENS 256
      SP=(0.5D0+0.5D0*DSIN(PH))**1.65D0-0.5D0                              DENS 257
      TAUT=PHASE+0.09544D0*SP                                              DENS 258
      TT2=TWOPI*TAUT+4.137D0                                               DENS 259
      TT2=DMOD(TT2,TWOPI)                                                  DENS 260
      TT4=FORPI*TAUT+4.259D0                                               DENS 261
      TT4=DMOD(TT4,TWOPI)                                                  DENS 262
      GT=0.02835D0+0.3817D0*(1.D0+0.4671D0*DSIN(TT2))*DSIN(TT4)            DENS 263
C     SEMIANNUAL EFFECTS ON LOG DENSITY                                    DENS 264
      PPO(2)=GT*DFZDZ                                                      DENS 265
      DENS=DENS+FZ*GT                                                      DENS 266
C     SEASONAL LATITUDINAL VARIATIONS OF THE LOWER THERMOSPHERE            DENS 267
   33 IF(HT.GT.3.6D0) GO TO 40                                             DENS 268
      HT90=HT2-90.D0                                                       DENS 269
      HT90SQ=HT90**2                                                       DENS 270
      EXS=-0.0013D0*HT90SQ                                                 DENS 271
      IF(EXS.LT.-20.D0) GO TO 40                                           DENS 272
      SS=0.014D0*DEXP(EXS)                                                 DENS 273
      PH=TWOPI*PHASE+1.72D0                                                DENS 274
      PH=DMOD(PH,TWOPI)                                                    DENS 275
      PP=DSIN(PH)                                                          DENS 276
      PHIABS=DSIGN(1.D0,PHI)                                               DENS 277
      DENLAT=SS*PHIABS*PP*SPHI                                             DENS 278
      PPO(3)=DENLAT*(1.D0-0.0026D0*HT90SQ)                                 DENS 279
```

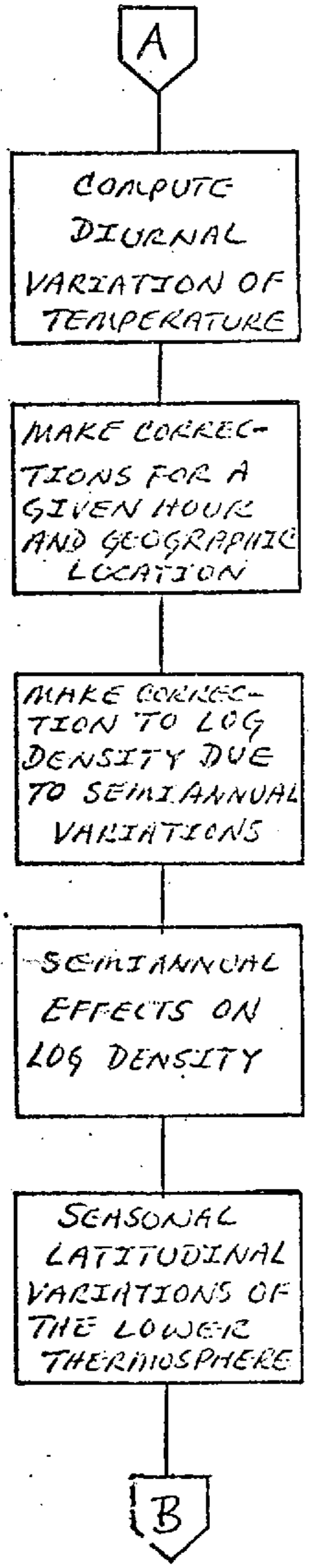
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      DENLAT=DENLAT*HT90
      DENS=DENS+DENLAT
   40 CONTINUE
C  SEASONAL LATITUDINAL VARIATIONS OF HELIUM
C         CORRECTIONS TO LOG HELIUM NUMBER DENSITY
      IF(HT.LE.5.0D0) GO TO 50
      IF(HT.GT.5.0D0.AND.HT.LT.8.0D0.AND.DABS(PHI).LT.ANGLAT) GO TO 50
      ARG=PI4-PHI*0.5D0*DSIGN(1.0D0,DECS)
      SN=DSIN(ARG)
      SN3=SN**3
      DECLP=DECS/EPSIL
      HELOG=0.55D0*DABS(DECLP)*(SN3-SIN3)
C CALCULATION OF LOG OF HELIUM NUMBER DENSITY
      START=0
      ORDER=4
      IF(TEM.GT.8.0D2) GO TO 42
      IF(HT.LE.10.0D0) GO TO 45
      START=32
      GO TO 45
   42 START=16
      IF(HT.LE.10.0D0) GO TO 45
      START=48
      ORDER=5
   45 K=START-ORDER
      DO 55 J=1,4
      K=K+ORDER
   55 C(J)=B(K+1)+TEM*(B(K+2)+TEM*(B(K+3)+TEM*B(K+4)))
      IF(ORDER.LT.5) GO TO 59
      K=START
      DO 56 J=1,4
      K=K+ORDER
   56 C(J)=C(J)+TEM4*B(K)
   59 HEFNCT=C(1)+HT2*(C(2)+HT2*(C(3)+HT2*C(4)))
      PRO(5)=C(2)+(2.0D0*C(3)+3.0D0*C(4)*HT2)*HT2
      HELOG=HELOG+HEFNCT
      HELIUM=(10.0D0**(HELOG)-10.0D0**(HEFNCT))*HEAVCG
   50 CONTINUE
C  USE NEW POLYNOMIAL FIT TO JACCHIA'S DENSITY TABLES 1971
      START=0
      ORDER=4
      IF(HT.LE.2.0D0) GO TO 65
      IF(TEM.GT.9.0D2) GO TO 62
      START=16
      IF(HT.LE.5.0D0) GO TO 65
      START=48
      IF(HT.LE.1.0D1) GO TO 65
      START=80
      GO TO 65
   62 START=32
      IF(HT.LE.5.0D0) GO TO 65
      START=64
      IF(HT.LE.1.0D1) GO TO 65
      START=96
      ORDER=5
   65 K=START-ORDER
      DO 70 J=1,4
```

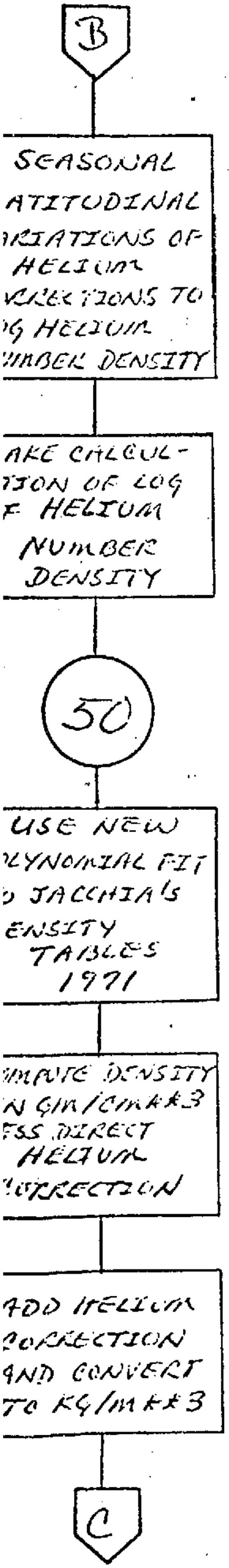
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

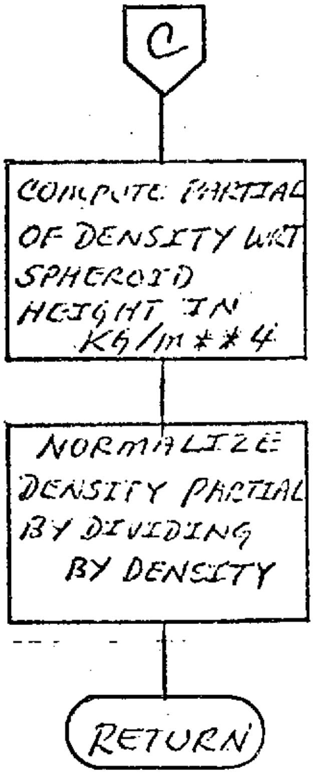
```
      K=K+ORDER
   70 C(J)=A(K+1)+TEM*(A(K+2)+TEM*(A(K+3)+TEM*A(K+4)))
      IF(ORDER.LT.5) GO TO 80
      K=START
      DO 75 J=1,4
      K=K+ORDER
   75 C(J)=C(J)+TEM4*A(K)
   80 EXPT=C(1)+HT2*(C(2)+HT2*(C(3)+HT2*C(4)))
      PPO(1)=C(2)+(2.0D0*C(3)+3.0D0*C(4)*HT2)*HT2
C COMPUTE DENSITY IN GM/CM**3 LESS DIRECT HELIUM CORRECTION
      EXSUMP=10.0D0**(EXPT+DENS)
C ADD HELIUM CORRECTION AND CONVERT TO KG/M**3
      DENSTY=1.0D3*(EXSUMP+HELIUM)
C COMPUTE PARTIAL OF DENSITY WRT SPHEROID HEIGHT IN KG/M**4
      PDPH=LN10*(EXSUMP*(PPO(1)+PPO(2)+PPO(3))+HELIUM*PPO(5))
C NORMALIZE DENSITY PARTIAL BY DIVIDING BY DENSITY
      PDPHDD=PDPH/DENSTY
      RETURN
      END
```





REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR


B
SEASONAL
ATITUDINAL
RIATIONS OF
HELIUM
RRECTIONS TO
G HELIUM
UMBER DENSITY
AKE CALCUL-
TION OF LOG
F HELIUM
NUMBER
DENSITY
50
USE NEW
LYNOMIAL FIT
 JACCHIA's
ENSITY
TABLES
1971
MPUTE DENSITY
N GM/CM**3
ESS DIRECT
HELIUM
ORRECTION
ADD HELIUM
CORRECTION
AND CONVERT
TO KG/M**3
C


C
COMPUTE PARTIAL
OF DENSITY WRT
SPHEROID
HEIGHT IN
KG/M**4
NORMALIZE
DENSITY PARTIAL
BY DIVIDING
BY DENSITY
RETURN

```
NAME                DIFF

PURPOSE             TO CALCULATE THE DIFFERENCE BETWEEN ANY TWO TIME
                    POINTS IN THE 20TH CENTURY

CALLING SEQUENCE    CALL DIFF(IYMD1,IHMS1,IYMD2,IHMS2,IDAY,ISEC)

   SYMBOL  TYPE     DESCRIPTION

   IYMD1    I       INPUT - DATE IN FORM YYMMDD

   IHMS1    I       INPUT - TIME ON IYMD1 IN FORM HHMMSS

   IYMD2    I       INPUT - SECOND DATE IN FORM YYMMDD

   IHMS2    I       INPUT - TIME ON IYMD2 IN FORM HHMMSS

   IDAY2    I       OUTPUT - ELAPSED FULL DAY DIFFERENCE
                             IDAY IS NEGATIVE IF IYMD2,IHMS2 IS THE
                             EARLIER TIME

                             ISEC HAS THE SAME SIGN CONVENTION AS IDAY

SUBROUTINES USED    NONE
   ISEC     I       OUTPUT - REMAINDER OF DIFFERENCE IN SECONDS

COMMON BLOCKS       MONTHS

INPUT FILES         NONE

OUTPUT FILES        NONE
```

```
      SUBROUTINE DIFF(IYMD1,IHMS1,IYMD2,IHMS2,IDAY,ISEC)
      COMMON/MONTHS/MONTH(13,2)
      ISUB(IY)=MIN0(MOD(IY,4),1)+1
C CHECK FOR A DIFFERENCE OF LESS THAN ONE DAY
      ISEC=0
      IF(IYMD1.EQ.IYMD2) GOTO 4000
C SEPARATE IYMD1 AND IYMD2 INTO THREE WORD EACH
      IY1=IYMD1/10000
      ID1=IYMD1-IY1*10000
      IM1=ID1/100
      ID1=ID1-IM1*100
      IY2=IYMD2/10000
      ID2=IYMD2-IY2*10000
      IM2=ID2/100
      ID2=ID2-IM2*100
C COMPUTE THE ELAPSED DAY SINCE JAN 0,1900
      L1=ISUB(IY1)
      IYEAR1=36525*(IY1-1)/100+MONTH(IM1,L1)+ID1
      L2=ISUB(IY2)
      IYEAR2=36525*(IY2-1)/100+MONTH(IM2,L2)+ID2
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
C CONVERT ELAPSED DAYS INTO ELAPSED SECONDS                             DIFF  56
      ISEC=(IYEAR2-IYEAR1)*86400                                        DIFF  57
C CALCULATE ELAPSED SECONDS INTO EACH DAY                               DIFF  58
 4000 ISEC1=IHMS1-40*(IHMS1/100)-2400*(IHMS1/10000)                     DIFF  59
      ISEC2=IHMS2-40*(IHMS2/100)-2400*(IHMS2/10000)                     DIFF  60
C SUBTRACT THE TWO ELAPSED SECONDS VALUES                               DIFF  61
      ISEC=ISEC+ISEC2-ISEC1                                             DIFF  62
C COMPUTE IDAY                                                          DIFF  63
      IDAY=ISEC/86400                                                   DIFF  64
C COMPUTE ISEC                                                          DIFF  65
      ISEC=ISEC-IDAY*86400                                              DIFF  66
      RETURN                                                            DIFF  67
      END                                                               DIFF  68
```

NAME DINRAD

PURPOSE TO CONVERT ANGLES EXPRESSED IN ARC MEASUREMENTS OR TIME MEASUREMENTS TO RADIANS

CALLING SEQUENCE CALL DINRAD(RAD,IH,IM,S,K)

SYMBOL	TYPE	DESCRIPTION
RAD	DP	OUTPUT - CONVERTED ANGLES IN RADIANS
IH	I	INPUT - SIGNED DEGREES OR HOURS
IM	I	INPUT - UNSIGNED MINUTES OF ARC OR TIME
S	DP	INPUT - UNSIGNED SECONDS OF ARC OR TIME
K	I	INPUT - SWITCH FOR TYPE OF INPUT K=1 INPUT IS IN ARC MEASUREMENT K=2 INPUT IS IN TIME MEASUREMENT

SUBROUTINES USED NONE

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

```
      SUBROUTINE DINRAD(RAD,IH,IM,S,K)                                  DINR  33
      DOUBLE PRECISION RAD,S                                            DINR  34
      RAD=(DFLOAT(IH*3600+IM*60)+S)*.4848136811095360D-5                DINR  35
      IF(K.EQ.2)RAD=RAD*15.D0                                           DINR  36
      RETURN                                                            DINR  37
      END                                                               DINR  38
```

DNVERT

DESCRIPTION

DNVERT is a subroutine to perform the double precision inversion of a given input matrix using the Gauss-Jordan Method of Condensation with partial pivoting. The input matrix is destroyed.

It should be noted that this routine was written specifically for GEODYN. Because of the type of matrices it is required to process, pivotal element testing is not incorporated.

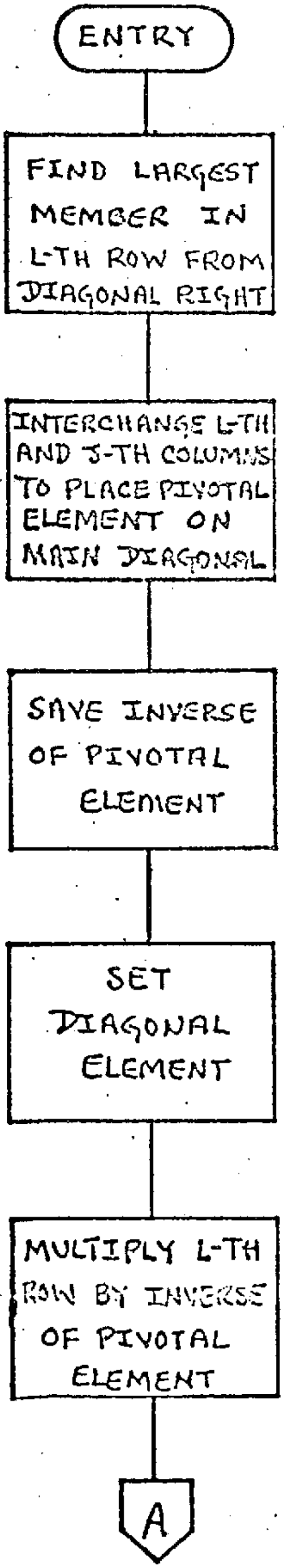
NAME	DNVERT
PURPOSE	TO INVERT A MATRIX USING GAUSS-JORDAN METHOD OF CONDENSATION WITH PARTIAL (COLUMN) PIVOTING
CALLING SEQUENCE	CALL DNVERT(N,AXMX,NT,NR)

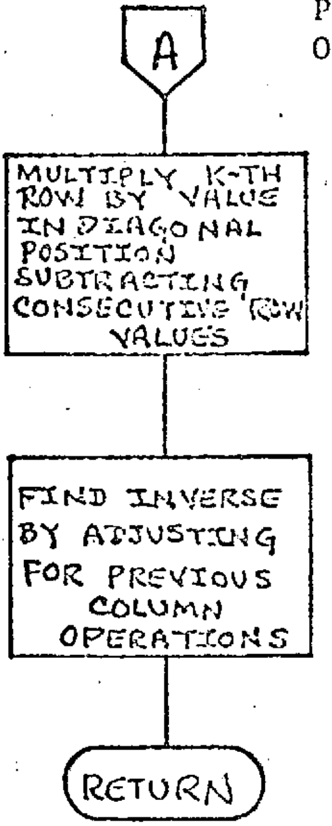
SYMBOL	TYPE	DESCRIPTION
N	I	INPUT - NUMBER OF ROWS OR COLUMNS OF THE MATRIX AXMX WHICH ARE ACTUALLY UTILIZED IN THE CALLING PROGRAM
AXMX (NT,NT)	DP	INPUT - MATRIX TO BE INVERTED OUTPUT - INVERSE OF THE INPUT MATRIX AXMX
NT	I	INPUT - DIMENSION OF THE MATRIX AXMX AS DEFINED BY THE CALLING PROGRAM
NR (N)	I	INPUT - TEMPORARY STORAGE USED BY PROGRAM TO STORE THE INDICES OF THE PIVOTAL COLUMNS THE DIMENSION OF THE ARRAY IN THE CALLING PROGRAM EQUIVALENT TO NR MUST BE AT LEAST EQUAL TO N

SUBROUTINES USED	NONE
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      SUBROUTINE DNVERT(N,AXMX,NT,NR)
      DOUBLE PRECISION AXMX(NT,NT),C,P
      DIMENSION NR(N)
C SEARCH THE LTH ROW STARTING WITH THE DIAGONAL ELEMENT AND WORKING
C RIGHT SAVE THE INDEX NUMBER OF THE JTH COLUMN CONTAINING LARGEST
C NUMBER IN ABSOLUTE MAGNITUDE
      DO 235 L=1,N
      P=0.0D0
      DO 272 J=L,N
      IF(.NOT.P.LT.DABS(AXMX(L,J))) GO TO 272
      P=DABS(AXMX(L,J))
      NR(L)=J
  272 CONTINUE
C INTERCHANGE LTH AND JTH COLUMNS TO PLACE PIVOTAL ELEMENT ON MAIN
C DIAGONAL
      J=NR(L)
      DO 273 K=1,N
      C=AXMX(K,J)
      AXMX(K,J)=AXMX(K,L)
```

```
  273 AXMX(K,L)=C
C SAVE  INVERSE OF PIVOTAL ELEMENT 1./AXMX(L,L)
      C=1.0D0/AXMX(L,L)
C SET DIAGONAL ELEMENT AXMX(L,L) = 1.
      AXMX(L,L)=1.0D0
C MULTIPLY LTH ROW BY INVERSE OF PIVOTAL ELEMENT
      DO 282 J=1,N
  282 AXMX(L,J)=C*AXMX(L,J)
C MULTIPLY KTH ROW BY VALUE IN DIAGONAL POSITION .
C SUBTRACTING CONSECUTIVE ROW VALUES
      DO 285 K=1,N
      IF(L.EQ.K) GO TO 285
      C=AXMX(K,L)
      AXMX(K,L)=0.0D0
      DO 284 J=1,N
  284 AXMX(K,J)=AXMX(K,J)-C*AXMX(L,J)
C INVERSE CAN NOW BE FOUND BY ADJUSTING FOR PREVIOUS COLUMN OPERATIONS
  285 CONTINUE
      DO 289 I=1,N
      L=N+1-I
      K=NR(L)
      DO 289 J=1,N
      C=AXMX(L,J)
      AXMX(L,J)=AXMX(K,J)
  289 AXMX(K,J)=C
      RETURN
      END
```





REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

DODELM

DESCRIPTION

DODELM is a subroutine specifically designed to read orbital element data from the DODS Data Base. DODELM can read both Cartesian elements and Kepler elements.

DODELM may be requested to read an element set number from observation data near the epoch time. When such data is used, it must be read from the DODS Data Base by subroutine DATBSE.

If the element set number of the desired elements is known, this number may be input to DODELM.

DODELM may determine elements either by element set number or by time. If no element set is found using the former procedure, DODELM will switch to the latter procedure attempting to find an element set at or before the input epoch time if possible, and later than the epoch time only when all else fails.

If no element set can be found, or if errors in the data are encountered, DODELM will print an error message and terminate the GEODYN run by calling ERROR.

NAME. DODELM

PURPOSE TO READ ORBITAL ELEMENTS FROM DODS DATA BASE

CALLING SEQUENCE CALL DODELM(IELM,ISATID,DAYEPC)

SYMBOL	TYPE	DESCRIPTION
IELM	I	INPUT & OUTPUT - ELEMENT SET NUMBER
ISATID	I	INPUT - SATELLITE ID
DAYEPC	DP	INPUT & OUTPUT - EPOCH TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR

SUBROUTINES USED CLEAR DATBSE DATES DJUL ERROR
YMDAY TDIF

COMMON BLOCKS CTIME CELEM CONSTS DODDAT INITBK

INPUT FILES DODISK - DODS DATA BASE ELEMENT FILE NUMBER

OUTPUT FILES OUTP - PRINTER

REFERENCES DODS DATA BASE ELEMENT FILE DESCRIPTION
'GSFC DODS DOCUMENTATION'

'GEODYN SYSTEMS DESCRIPTION' - SECTION 2.11.1
VOLUME I - GEODYN DOCUMENTATION

```
      SUBROUTINE DODELM(IELM,ISATID,DAYEPC)
      REAL*8 DAYEPC,DJUL,DUT,DREF,MJEPCH,YMDAY,DRAD,MJD,DJREF,
     1   ELEMST,ORBELA,DUL,DATAEP,XNU,EC,EPSEC,SEC
      LOGICAL*1 NOT1ST,DODELM
      INTEGER ISATID,SATNO,RECORD(880),DODISK,OUTP,RECSAT,ELREC(12),
     1   MREC,JED(2)
      INTEGER*2 LREC,REC2(1760),RECPAR,IG2
      COMMON/CTIME/DATAEP,DAYREF(21)
      COMMON/CELEM/ELEMST(12),ORBELA(12),XNU,EC,RMS
      COMMON/CONSTS/DUM2(4),DRAD,DUM3(2)
      COMMON/DODDAT/DUM4(21),IG2(8)
      DIMENSION MREC(172,5)
      COMMON/INITBK/IEPYMD,IEPHM,EPSEC,DUM1(53)
      EQUIVALENCE (ELREC(1),ORBELA(1)),(JED(1),MJD),(NPART,NDPART),
     *       (LREC,RECORD(1),REC2(1))
      DATA DJREF,DUL,DUT/2430099.5D0,1.0D7,8.64D2/
      DATA NOT1ST/.FALSE./
      DATA OUTP,DODISK,J/6,4,-1/
      IF(IELM.LT.0) GO TO 40
    5 IF(NOT1ST) GO TO 100
      NOT1ST=.TRUE.
C DEFINE RANDOM ACCESS FILE
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

8.0-169

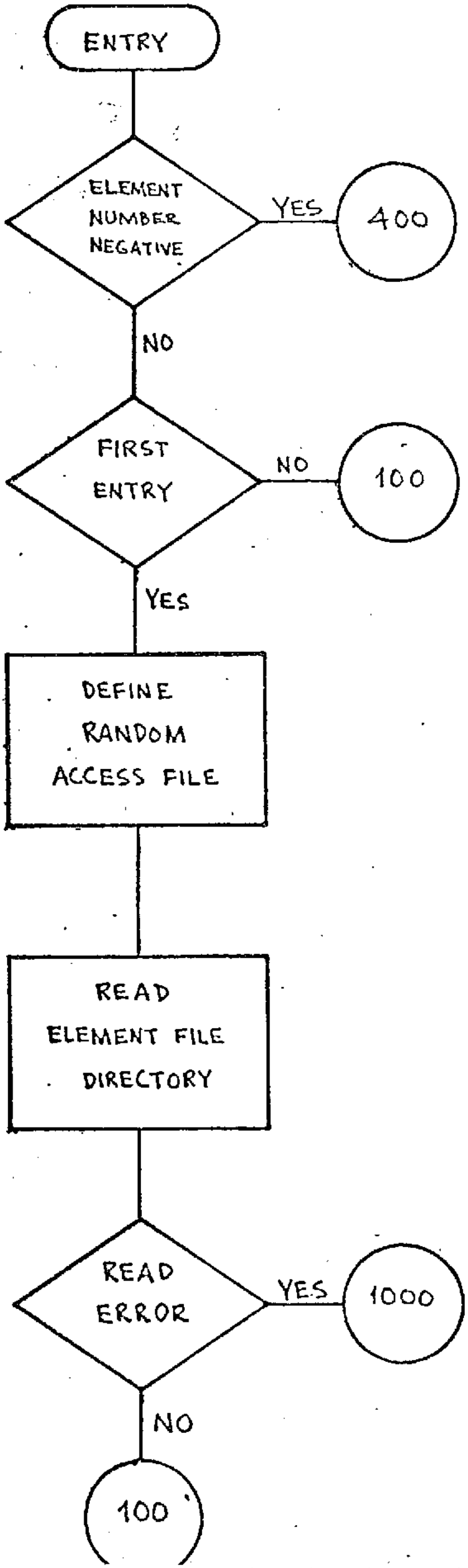
```
      DEFINE FILE 4 (442,3520,L,ICNT)
      ICNT=1
C READ DIRECTORY RECORD
      READ(DDISK'ICNT,ERR=1000) RECORD
      RECPAR=REC2(2)
C SEARCH FOR SATELLITE
  100 DO 10 I=2,LREC
      IF(RECORD(I).GE.0) GO TO 10
      ISAT=-RECORD(I)
      IF(ISAT.EQ.ISATID) GO TO 15
   10 CONTINUE
      GO TO 1010
   15 NOBS=RECORD(I+1)-1
      IF(NOBS.LE.0) GO TO 1010
      RECSAT=(NOBS-1)/5+1
      NDOB=NOBS-NOBS/5*5
      INDOB=5
      NOPART=(RECSAT-1)/RECPAR+1
      IWORD=I+4
C CONVERT EPOCH TO DODS SYSTEM
      MJEPCH=DJUL(DAYEPC)-DJREF
      NWORD=IWORD+NOPART-1
      NOREC=RECSAT-RECSAT/RECPAR*RECPAR
      IF(NOREC.EQ.0) NOREC=RECPAR
      IF(ICLM.GT.0) GO TO 200
C SEARCH FOR PARTITION CONTAINING EPOCH TIME
   18 DO 20 J=IWORD,NWORD
      JWORD2=2*J-1
      MJD=REC2(JWORD2)
      IF(MJEPCH.LT.(MJD+1.D0)) GO TO 25
   20 CONTINUE
      GO TO 200
C EPOCH TIME PARTITION DETERMINED
   25 IPART=MAX0(J-IWORD,1)
      JWORD2=2*(IWORD+IPART-1)
   28 NOB=0
      ICOUNT=REC2(JWORD2)
      NREC=ICOUNT+RECPAR-1
      LABEL=1
      IF(IPART.NE.NPART) GO TO 30
      NREC=ICOUNT+NOREC-1
      IF(NOREC.EQ.1) INDOB=NDOB
C READ PHYSICAL RECORD
   30 ICNT=ICOUNT
      READ(DDISK'ICNT,ERR=1020) MREC
   40 NOB=NOB+1
      IF(NOB.GT.INDOB) GO TO 50
      JED(1)=MREC(2,NOB)
      JED(2)=MREC(3,NOB)
      MJD=MJD+1.0D-2
C CHECK FOR ELEMENT SET AT EPOCH
      IF(MJEPCH.EQ.MJD) GO TO 60
      IF(MJEPCH.GT.MJD) GO TO 40
      NOB=NOB-1
      IF(NOB.EQ.0) GO TO 90
      GO TO 60
```

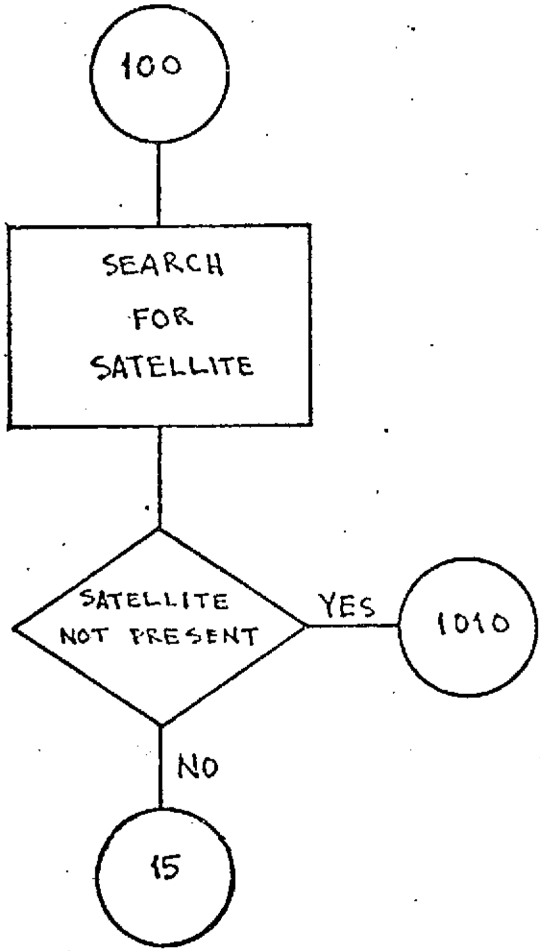
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

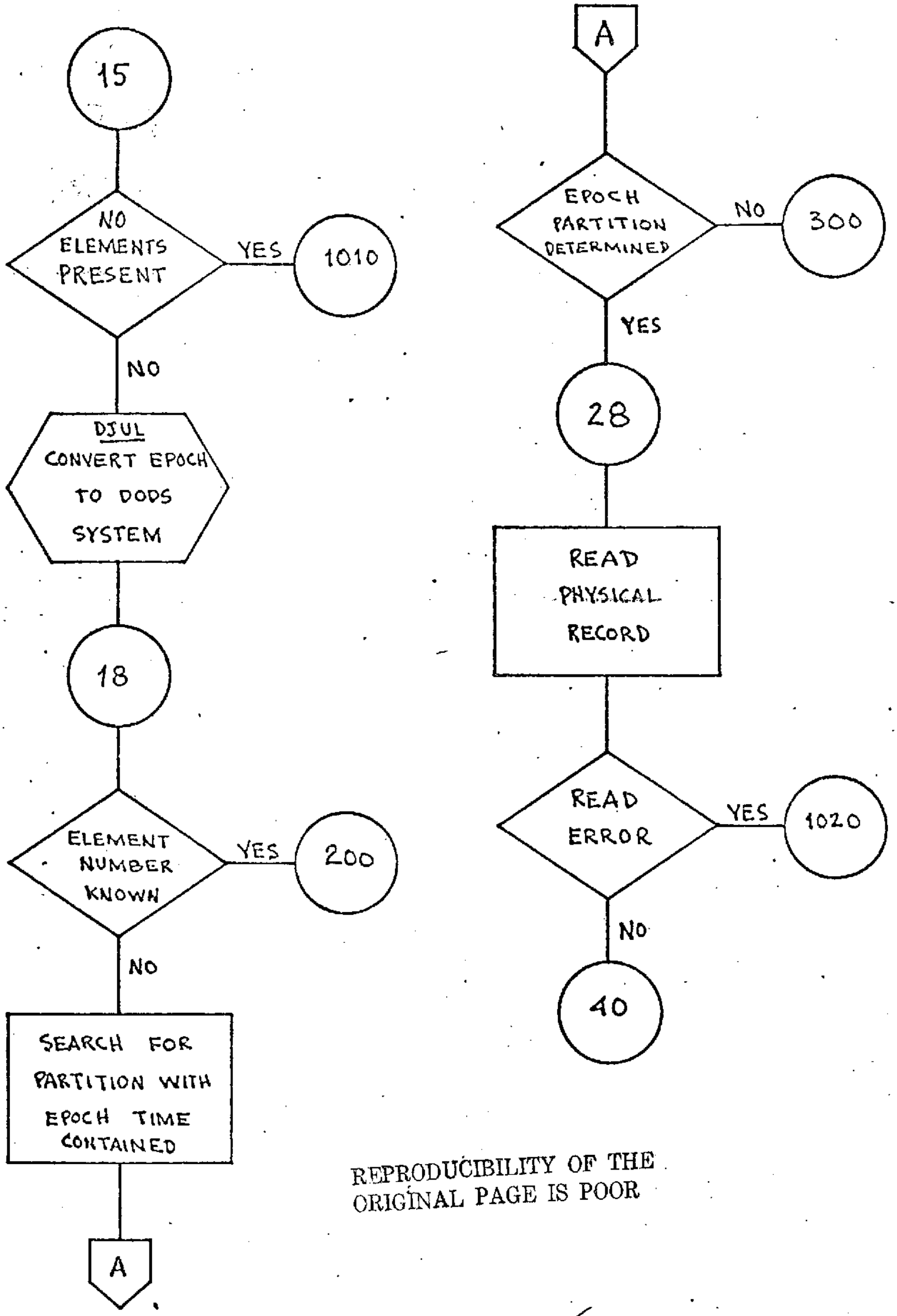
```
   50 ICOUNT=ICOUNT+1                                                   DODE 112
      IF(ICOUNT.GT.NREC) GO TO 500                                      DODE 113
      LABEL=2                                                           DODE 114
      ICNT=ICOUNT                                                       DODE 115
C READ NEW PHYSICAL RECORD                                               DODE 116
      READ(DCDISK'ICNT,ERR=1020) MREC                                   DODE 117
      NOB=0                                                             DODE 118
      IF(IPART.EQ.NPART.AND.ICOUNT.EQ.NREC) INDOB=NDOB                  DODE 119
      GO TO 40                                                          DODE 120
C ELEMENTS HAVE BEEN FOUND                                              DODE 121
   60 DO 61 I=1,12                                                      DODE 122
   61 ELREC(I)=MREC(I+4,NOB)                                            DODE 123
      IELM=MREC(1,NOB)                                                  DODE 124
      JED(1)=MREC(2,NOB)                                                DODE 125
      JED(2)=MREC(3,NOB)                                                DODE 126
      MJD=MJD+1.0D-2                                                    DODE 127
      ITYPE=MREC(4,NOB)                                                 DODE 128
      GO TO (65,75),ITYPE                                               DODE 129
      GO TO 1030                                                        DODE 130
C CARTESIAN ELEMENTS LOADED INTO COMMON                                  DODE 131
   65 DO 70 I=1,3                                                       DODE 132
      ELEMST(I)=ORBELA(I)*DUL                                           DODE 133
   70 ELEMST(I+3)=ORBELA(I+3)*DUL/DUT                                   DODE 134
      GO TO 85                                                          DODE 135
C KEPLERIAN ELEMENTS LOADED INTO COMMON                                  DODE 136
   75 ELEMST(1)=ORBELA(1)*DUL                                           DODE 137
      ELEMST(2)=ORBELA(2)                                               DODE 138
      DO 80 I=3,6                                                       DODE 139
   80 ELEMST(I)=ORBELA(I)/DRAD                                          DODE 140
   85 CALL CLEAR(MREC,480,1)                                            DODE 141
      IF(MJEPCH.EQ.MJD) RETURN                                          DODE 142
C RESET EPOCH                                                           DODE 143
      DAYEPC=MJD+YMDAY(570913,0,0.00)                                   DODE 144
      DAYEPC=DAYEPC+TDIF(4,3,DAYEPC)/8.64E4                             DODE 145
      DATAEP=DAYEPC                                                     DODE 146
      CALL DATES(DAYEPC,IEPYMD,IHM,SEC)                                 DODE 147
      IEPHM=IHM                                                         DODE 148
      EPSEC=SEC                                                         DODE 149
      RETURN                                                            DODE 150
   90 ICOUNT=ICOUNT-1                                                   DODE 151
      NOB=1                                                             DODE 152
      IPART=IPART-1                                                     DODE 153
      IF(IPART.LT.1) GO TO 60                                           DODE 154
C BACK UP ONE PARTITION                                                 DODE 155
      JWORD2=2*(IWORD+IPART-1)                                          DODE 156
      ICOUNT=REC2(JWORD2)+RECPAR-1                                      DODE 157
      GO TO 300                                                         DODE 158
C SEARCH BY ELEMENT SET NUMBER                                          DODE 159
  200 DO 250 J=IWORD,NWORD                                              DODE 160
      JWORD2=2*J                                                        DODE 161
      ICOUNT=REC2(JWORD2)-1                                             DODE 162
      NREC=ICOUNT+RECPAR                                                DODE 163
      IF(J.EQ.NWORD) NREC=ICOUNT+NDREC                                  DODE 164
  210 ICOUNT=ICOUNT+1                                                   DODE 165
      IF(ICOUNT.GT.NREC) GO TO 250                                      DODE 166
      LABEL=3                                                           DODE 167
```

```
      ICNT=ICOUNT
C READ NEW PHYSICAL RECORD
      READ(DD)ISK'ICNT,ERR=1020) MREC
      IF(ICOUNT.EQ.NREC.AND.J.EQ.NWORD) INDOB=NDOB
      DO 240 NOB=1,INDOB
      IF(MREC(1,NOB).EQ.IELM) GO TO 60
  240 CONTINUE
  250 CONTINUE
C NO ELEMENTS AVIALABLE WITH THIS NUMBER
      WRITE(OUTP,2250) IELM
      IELM=0
      GO TO 13
  300 JWORD2=2*NWORD
      ICOUNT=REC2(JWORD2)+NDREC-1
C TAKE LATEST ELEMENT SET
      LABEL=6
      NOB=NDOB
      GO TO 310
  305 NOB=5
  310 ICNT=ICOUNT
C READ LAST PHYSICAL RECORD
C FIND ELEMENT SET NUMBER FROM DATA
      READ(DD)ISK'ICNT,ERR=1020) MREC
      GO TO 60
  400 MJEPCH=DAYEPC+1.D2
      DODELM=.FALSE.
      IN=33
      CALL DATBSE(IN,DAYEPC,MJEPCH,ISATID,DODELM)
      CALL DATBSE(IN,DAYEPC,MJEPCH,ISATID,DODELM)
      IELM=IG2(1)
      GO TO 5
  500 JWORD2=JWORD2+2
      IF(IPART.EQ.NPART) GO TO 1010
C TRY NEXT PARTITION
      IPART=IPART+1
      GO TO 23
C PRINT ERROR MESSAGES
 1000 WRITE(OUTP,2000) J
      GO TO 1035
 1010 WRITE(OUTP,2010) ISATID
      GO TO 1035
 1020 WRITE(OUTP,2020) ISATID,ICOUNT,LABEL
      GO TO 1035
 1030 WRITE(OUTP,2030) ISATID,ICOUNT,ITYPE,LABEL
 1035 WRITE(OUTP,2040) RECORD,MREC
C TAKE ERROR EXIT
      CALL ERROR(7,DAYEPC)
      STOP
 2000 FORMAT(1H1,20X,'UNABLE TO READ DODS DATA BASE ELEMENT DIRECTORY'/
     1    21X,'ERROR CODE IS',I5/21X,'NONAME EXECUTION TERMINATED')
 2010 FORMAT(1H1,30X,'NO ELEMENT SET IN DODS DATA BASE FOR'/21X,
     1    'SATELLITE',I8/21X,'NONAME EXECUTION TERMINATED')
 2020 FORMAT(1H1,20X,'DODS DATA BASE READ ERROR OCCURRED IN ELEMENT ',
     1    'FILE FOR'/21X,'SATELLITE',I9/21X,'DATA RECORD NUMBER IS',I6/
     2    21X,'NONAME EXECUTION TERMINATED'/21X,'LABEL =',I2)
 2030 FORMAT(1H0,20X,'DODS ELEMENT RECORD ERROR ENCOUNTERED FOR ',
```

```
    1     'SATELLITE',I9/21X,'ELEMENT TYPE INCORRECTLY SPECIFIED IN ',    DODE 224
    2     'RECORD',I9/21X,'TYPE SPECIFIED IS',I9/21X,'NONAME EXECUTION',  DODE 225
    3     ' TERMINATED'/21X,'LABEL =',I2)                                 DODE 226
 2240 FORMAT(1H1/(10X,Z8,1X,Z8,2X,Z8,1X,Z8,4X,Z8,1X,Z8,2X,Z8,1X,Z8))      DODE 227
 2250 FORMAT(1H0,20X,'UNABLE TO FIND ELEMENT SET',I5/21X,'IN DODS DATA',DODE 228
    1     ' BASE ELEMENT FILE'/21X,'PROGRAM SEARCHING FOR ELEMENT SET ',  DODE 229
    2     'CORRESPONDING TO EPOCH'//)                                     DODE 230
      END                                                                 DODE 231
```

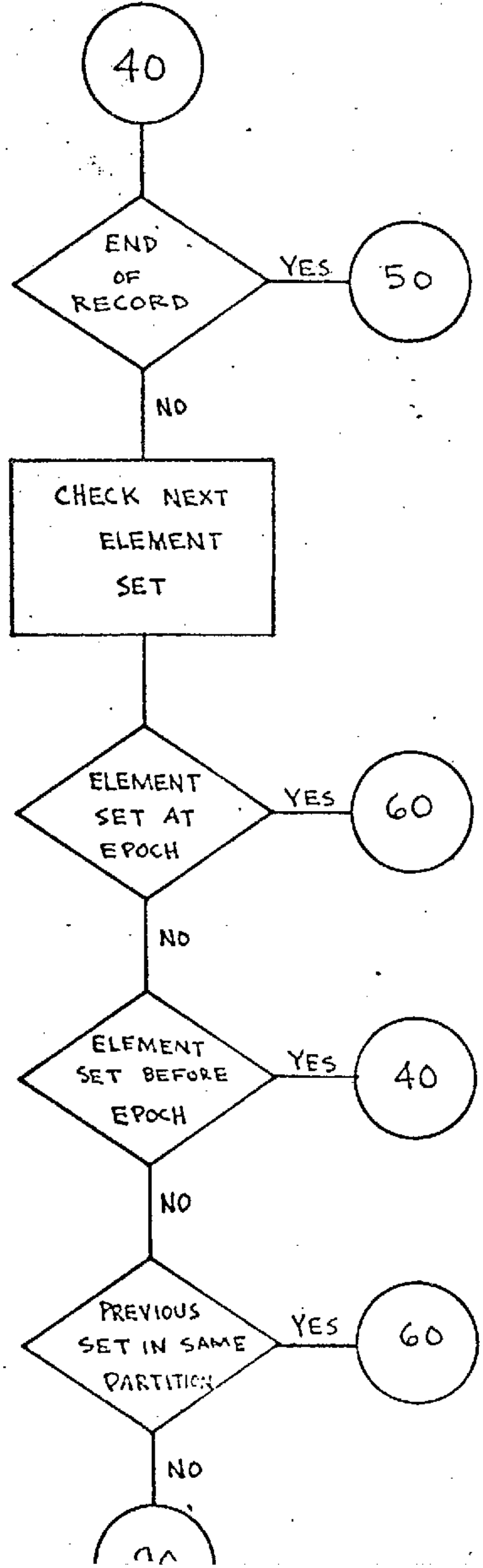

ENTRY
ELEMENT NUMBER NEGATIVE
YES
400
NO
FIRST ENTRY
NO
100
YES
DEFINE RANDOM ACCESS FILE
READ ELEMENT FILE DIRECTORY
READ ERROR
YES
1000
NO
100

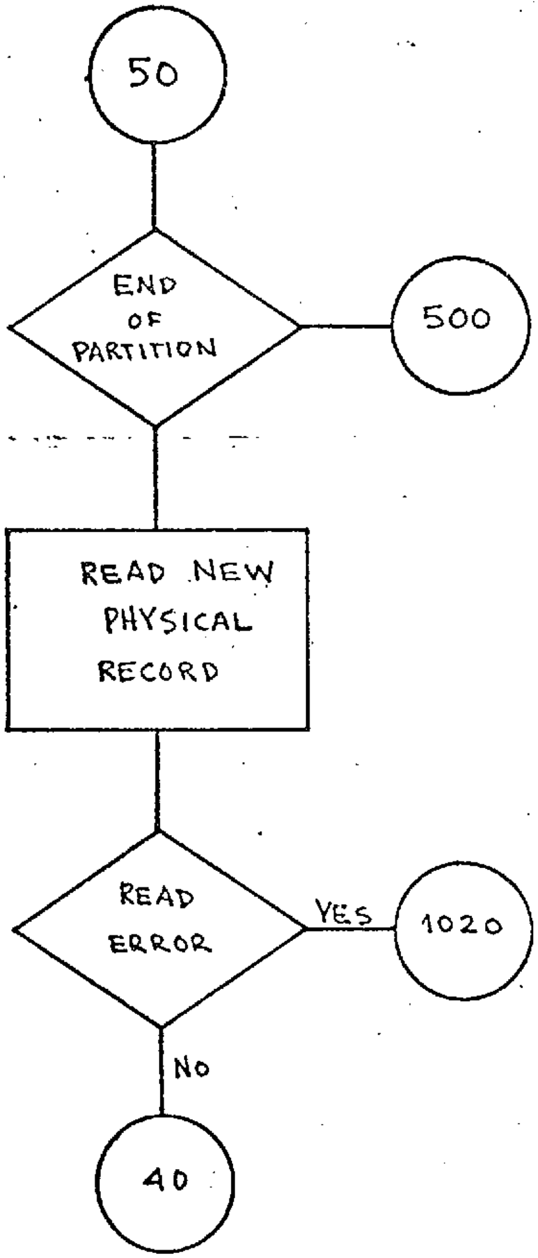

100
SEARCH FOR SATELLITE
SATELLITE NOT PRESENT
YES
1010
NO
15

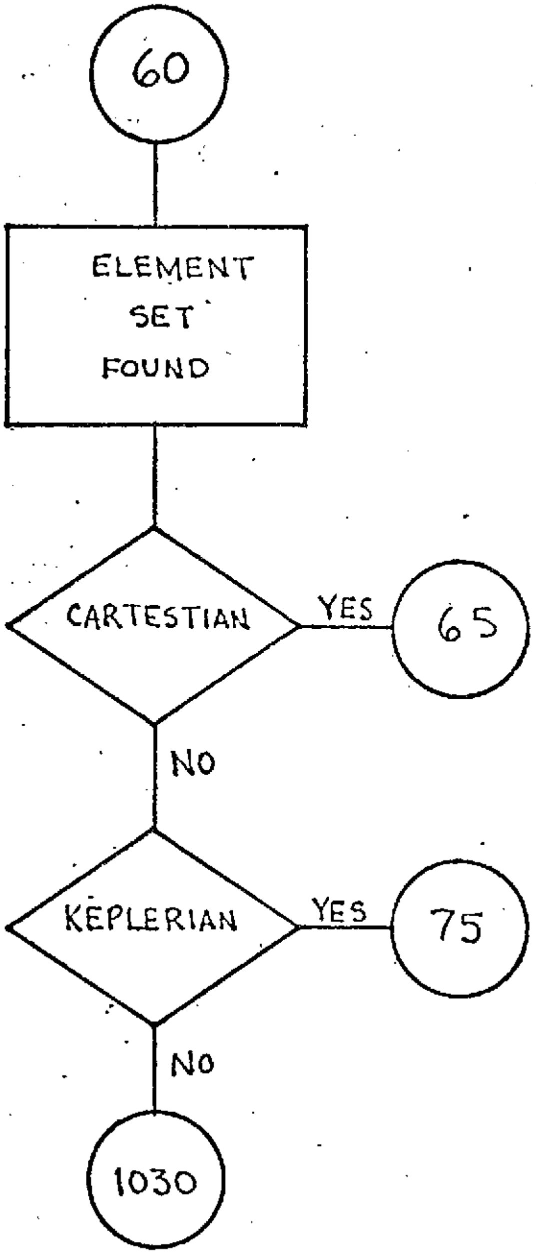


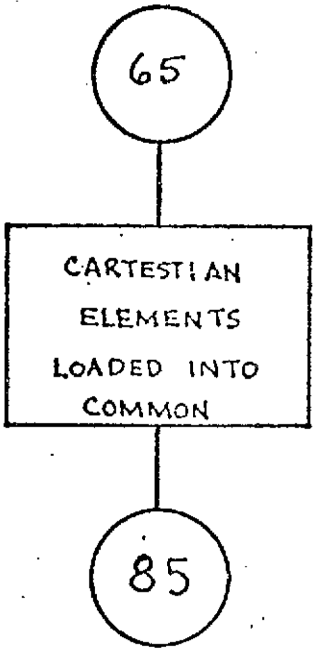
REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

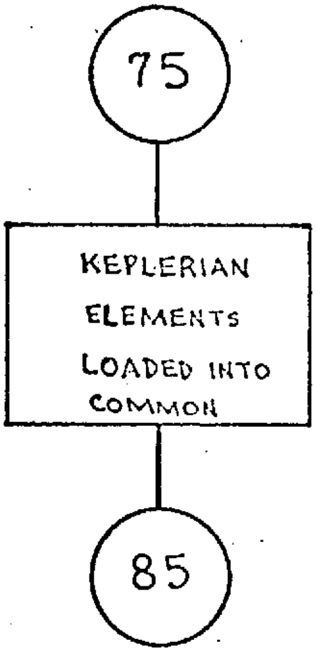
8.0-175


40
END OF RECORD
YES
50
NO
CHECK NEXT ELEMENT SET
ELEMENT SET AT EPOCH
YES
60
NO
ELEMENT SET BEFORE EPOCH
YES
40
NO
PREVIOUS SET IN SAME PARTITION
YES
60
NO

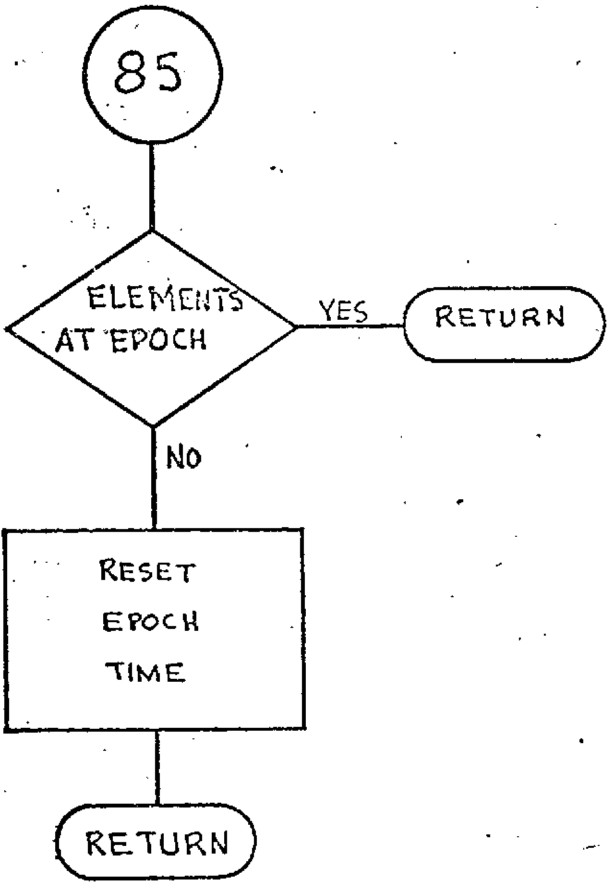

50
END OF PARTITION
500
READ NEW PHYSICAL RECORD
READ ERROR
YES
1020
NO
40

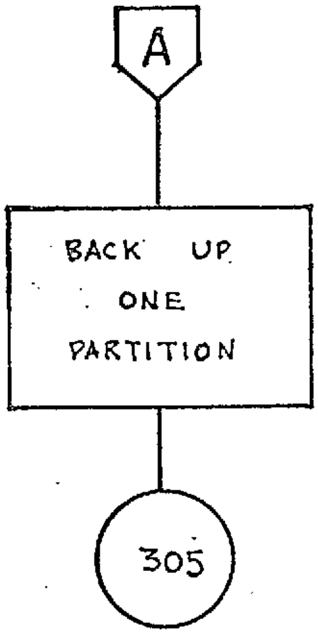


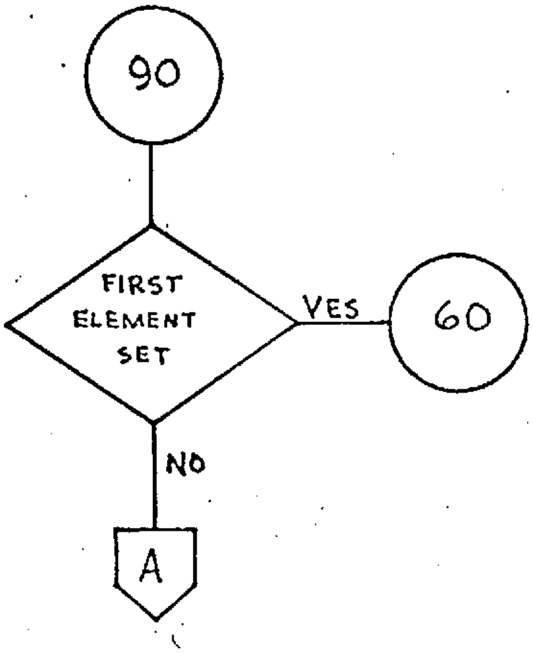


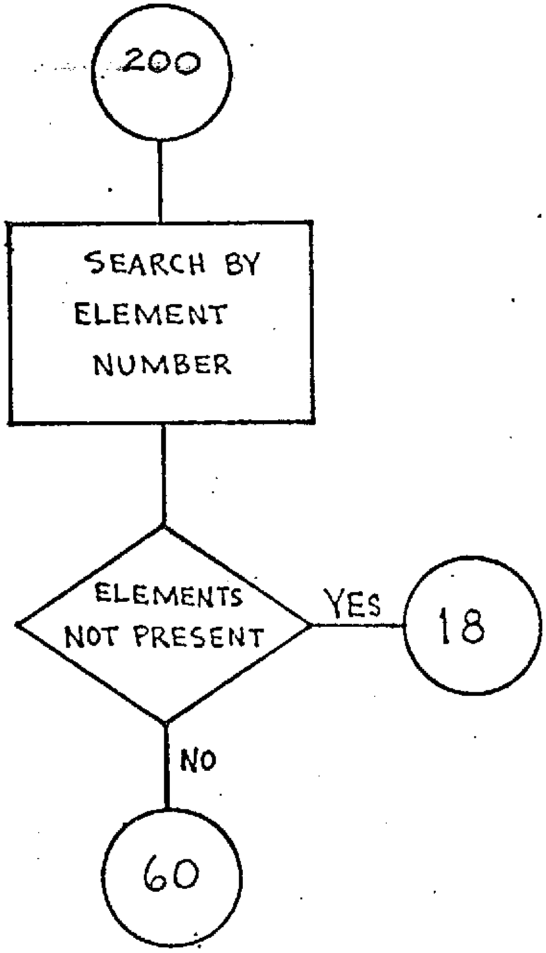


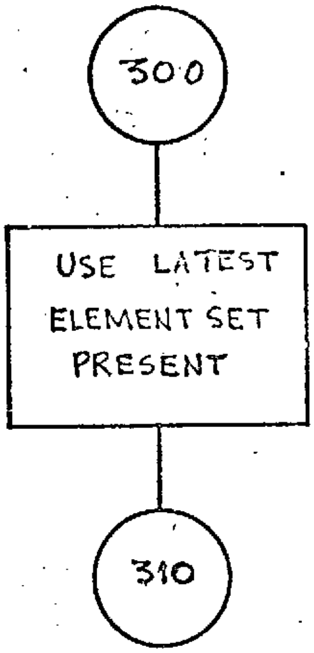
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

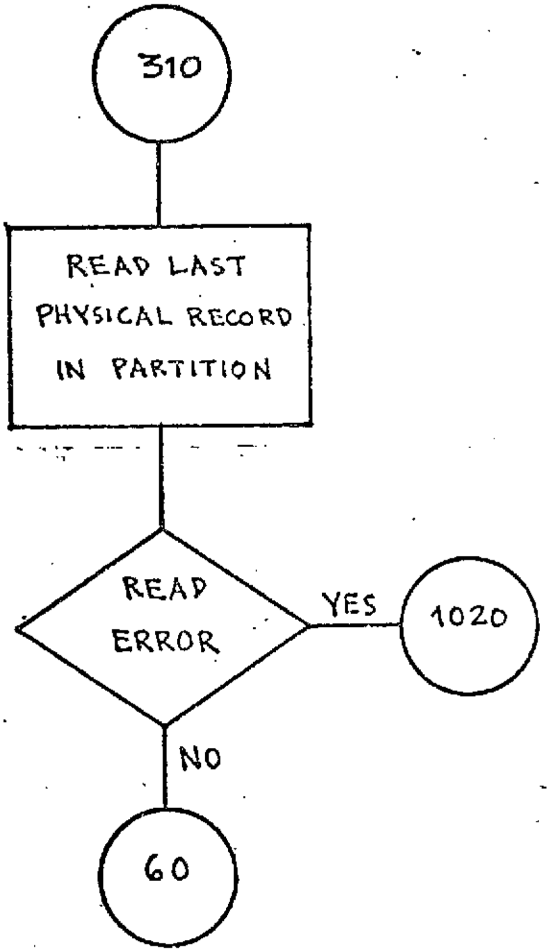

85
ELEMENTS AT EPOCH
YES
RETURN
NO
RESET EPOCH TIME
RETURN

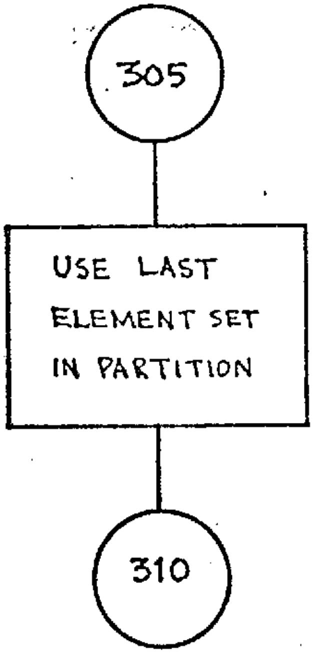

A
BACK UP ONE PARTITION
305

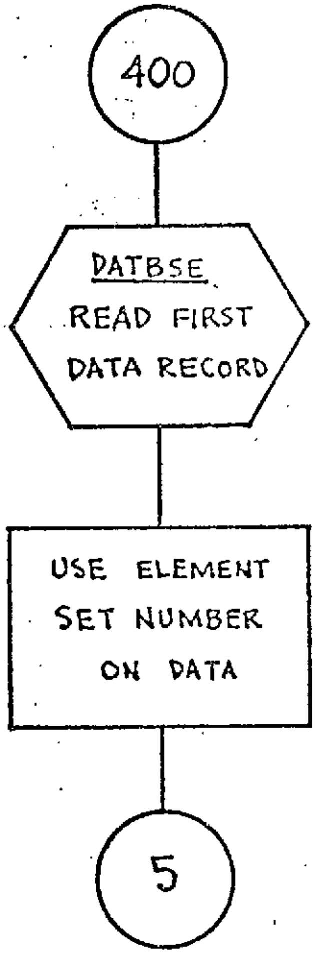

90
FIRST ELEMENT SET
YES
60
NO
A

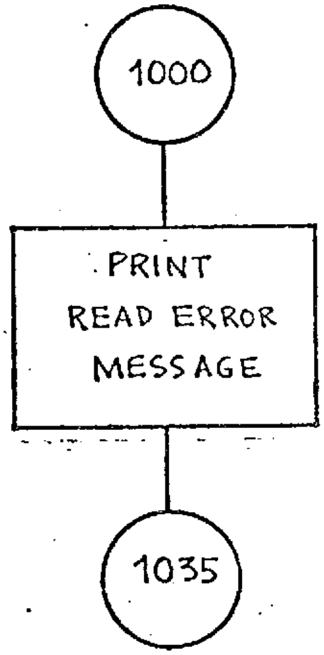

200
SEARCH BY ELEMENT NUMBER
ELEMENTS NOT PRESENT
YES
18
NO
60

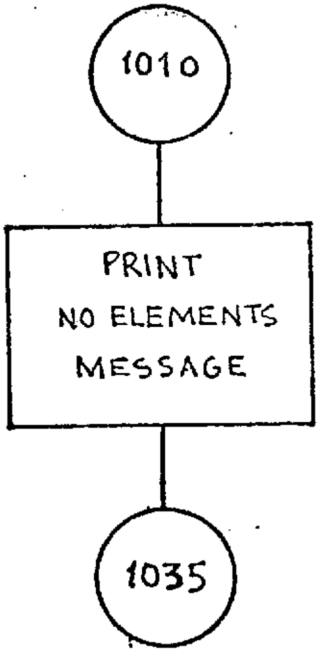

300
USE LATEST ELEMENT SET PRESENT
310

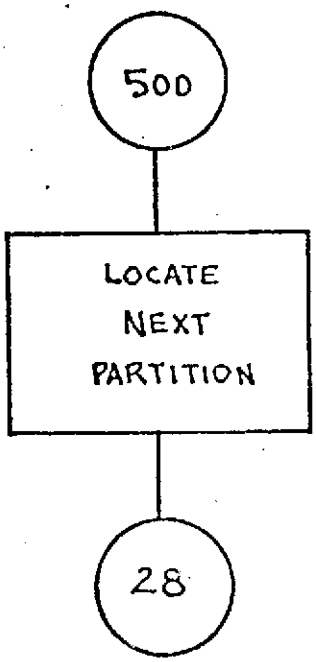

310
READ LAST PHYSICAL RECORD IN PARTITION
READ ERROR
YES
1020
NO
60


305
USE LAST ELEMENT SET IN PARTITION
310

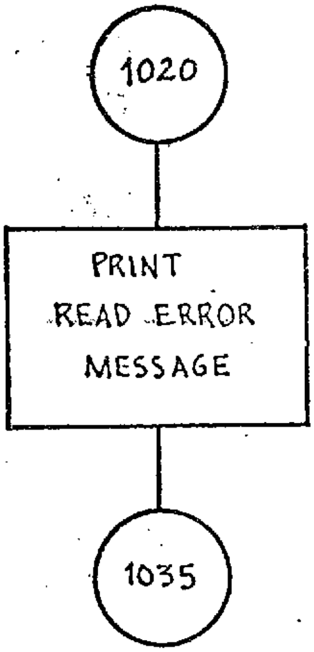

400
DATBSE
READ FIRST
DATA RECORD
USE ELEMENT
SET NUMBER
ON DATA
5

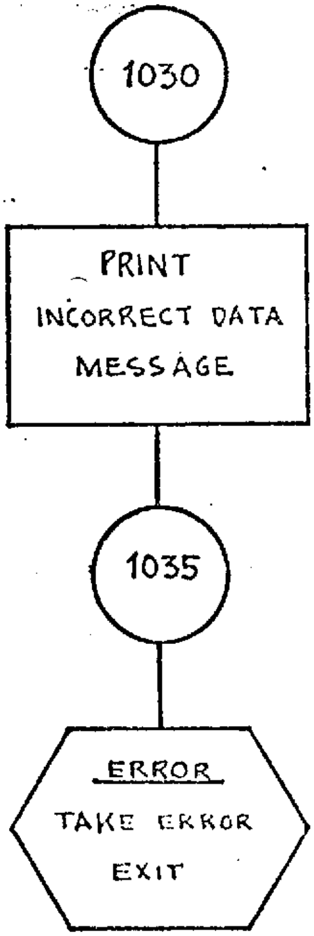

1000
PRINT
READ ERROR
MESSAGE
1035


1010
PRINT
NO ELEMENTS
MESSAGE
1035


500
LOCATE
NEXT
PARTITION
28

8.0 -180


1020
PRINT
READ ERROR
MESSAGE
1035


1030
PRINT
INCORRECT DATA
MESSAGE
1035
ERROR
TAKE ERROR
EXIT

NAME DODSRD

PURPOSE TO READ OBSERVATION DATA IN DODS FORMAT AND PARTIALLY PREPROCESS THE OBSERVATIONS

CALLING SEQUENCE CALL DODSRD(NSTARD)

SYMBOL	TYPE	DESCRIPTION
NSTARD	I	INPUT - NUMBER OF STATIONS THAT WERE READ FROM CARDS

SUBROUTINES USED
AND2 CLEAR CARCTN DATBSE EQUATR
TOIF YMDAY NUMBR4 RANDWR BIAS
NUMBR2

COMMON BLOCKS
APARAM CGEOS CONSTS CPARAM CTIME
DODDAT CEPHEM INTBLK PREBLK SIGBLK
STANUM TPEBLK

INPUT FILES INTP - INPUT CARDS

OUTPUT FILES NEWTAP - OUTPUT DATA TAPE NUMBER
PRINTER

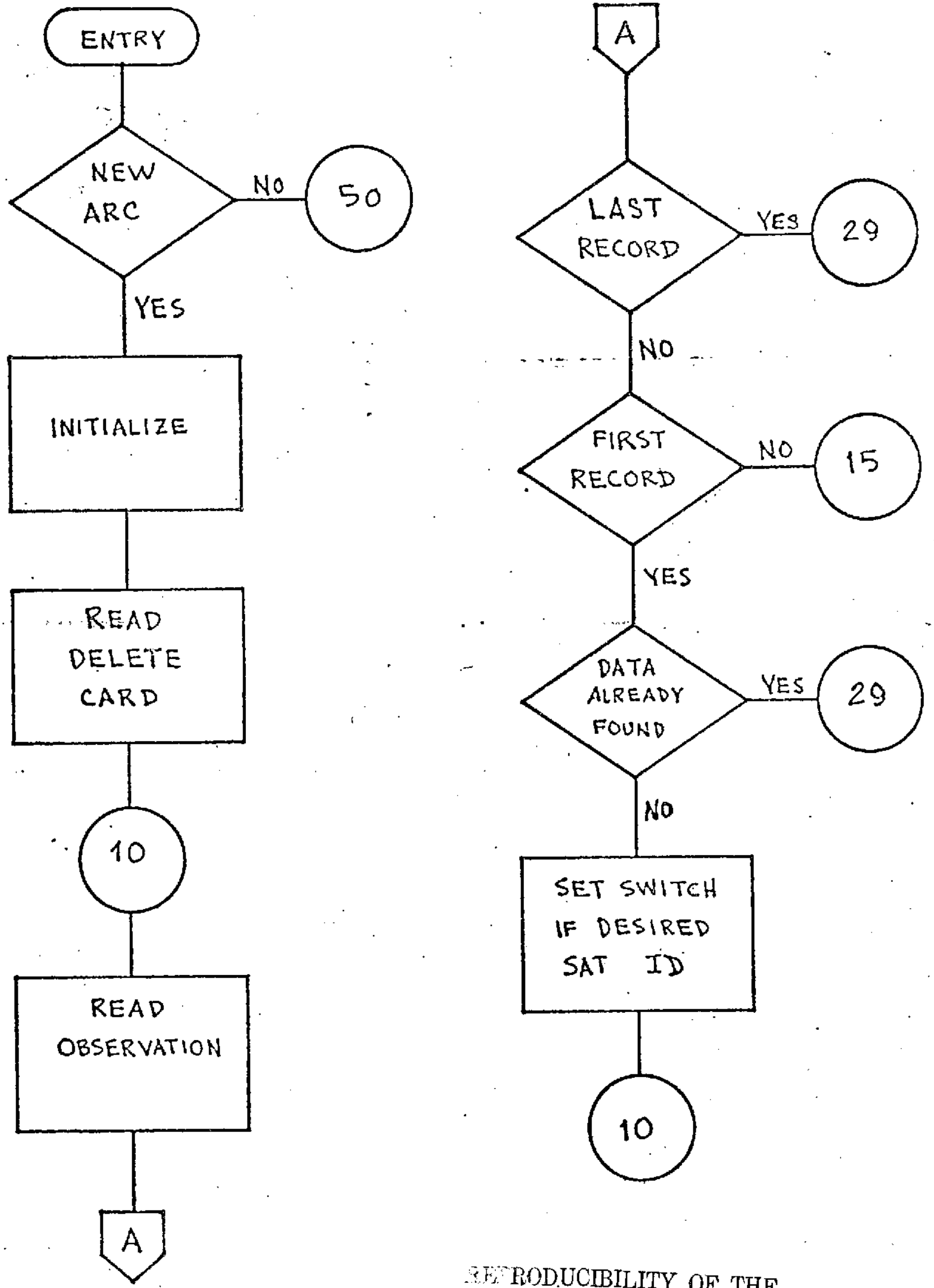
```
      SUBROUTINE DODSRD(NSTARD)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL*1 PREPRO,VHFCHN,NEWTP,LSTANM,BL,NOT1ST,ONEOB,OKSAT,NAME7
      LOGICAL NORATE
      DOUBLE PRECISION NAME,JNAME
       REAL RFINDX,SIGCHG,SIGSTD,S2R,TCOR,SIG1,SIG2,OBREC,TOIF,SQRT
      INTEGER*2 MTYPE,NMEAS,PRETYP,CHANEL,INDPRE,IPREPR,IMTYPE,ISTNO,
     . CHNMSK,OBITS,IG,IT,TTAG,PBIT1,PBIT2,ITYPE,OBIT,IG1,IG2,
     . FEEDBK,CULL,IBIT,ISTARD,ESTANO,ISTANO,STANOS,ISAT
      INTEGER AND2,AMBIG,SATNO,RECNO,DELETE
      DIMENSION TIME(14),STNAM(14),OBS(14),OBREC(25),X(3),OBITS(14),
     . ITYPE(16),FEEDBK(4),DELETE(7),LSTANM(3),ONEOB(7),IBIT(16)
      COMMON/APARAM/INPAR,INPAR1,NBIAS,NSTSTA(7)
      COMMON/CGEOS/ISAT1D,ISAT2,IPREPR(4,50),RFINDX(2,50),INDPRE(2,50),
     . NPRE,NSIG,NCULL,SIGCHG(50),IMTYPE(50),ISTNO(50),CULL(2,100)
      COMMON/CONSTS/DPI,DTWOPI,DRAD,S2R,IS2R
      COMMON/CPARAM/NSTA,NMAST(13)
      COMMON/CTIME/DATAEP,DAYREF,DSTART,DAYSTP,INTDAY(15)
      COMMON/DODDAT/TIME1,STNAM1,OBS1,DG(2),OBSCOR,SATNO,IOBND1,
     . IWT(6),TCOR,IG2(2),IT,IG1,TTAG,PBIT1,PBIT2,IG
      COMMON/CEPHEM/JNAME(381),ISTARD(381),ESTANO(381),ISTANO(386)
      COMMON/INTBLK/INTEG1(53),NORATE,INTEG2(73)
      COMMON/PREBLK/DAY,OBS1,OBS2,SIG(2),RFINDX,ISTA,MTYPE,NMEAS,
     . ISAT,PRETYP,CHANEL,VHFCHN,PREPRO,RECNO
      COMMON/SIGBLK/SIGSTD(60),IN,IDTAPE(3)
      COMMON/STANUM/NAME(280),STANOS(280),NOSTOR
      COMMON/TPEBLK/INTP(7),IND,ISCRA(4)
```

```
      EQUIVALENCE (SIG1,SIG(1)),(SIG2,SIG(2)),(FEEDBK(1),DG(2)),
     .   (LSTANM(1),STNAM1),(OBREC(1),TIME1)
      DATA CHNMSK/Z7/,IBIT/Z1,Z2,Z4,Z8,Z10,Z20,Z40,Z80,Z100,Z200,Z400,
     .   Z800,Z1000,Z2000,Z4000,Z8000/,DUL/1.0D+7/,DUT/8.64D+2/,
     .   MTPLST/0/,STNLST/6H      /,DAYLST/0.0D0/,IM/0/,NEWTAP/3/,
     .   ITYPE/2,5,12,7,14,1,9,0,3,7*0,6,13/,BL/1H /,ONEOB/.FALSE.,
     .   3*.TRUE.,3*.FALSE./,VLIGHT/2.997925D3/
      NUMBER=0
      NOT1ST=.FALSE.
      ISAT=1
C INITIALIZE
      NCRATE=.TRUE.
      IF(IN.LE.0) IN=21
      CALL CLEAR(DELETE,7,1)
C READ DELETE CARD
      IF(MOD(IND,2).EQ.1) READ(5,2000) DELETE
      NEWTP=IND.EQ.2.OR.IND.EQ.3
      IND=0
      CKSAT=.FALSE.
      DO 5 I=1,14
      TIME(I)=0.0D0
    5 STNAM(I)=0.0D0
      REFTIM=YMDAY(570916,0,0.0D0)
      CJEASE=YMDAY(580100,0,0.0D0)
C READ OBSERVATION
   10 CALL DATUSE(IN,DATAEP,DAYSTP,ISATID,NOT1ST)
      IF(TIME1.GT.0.0D0) GO TO 15
      IF(TIME1.LT.-1.5D0) GO TO 200
      IF(CKSAT) GO TO 200
C CHECK SATELLITE ID
      OKSAT=ISATID.EQ.SATNO.OR.ISATID.EQ.0
      IF(OKSAT) ISATID=SATNO
      VHFCHN=ISATID.EQ.67511.OR.ISATID.EQ.6750101
      IF(NEWTP.AND.CKSAT) WRITE(NEWTAP) OBREC
      GO TO 10
   15 IF(.NOT.OKSAT) GO TO 10
      MTYPE=ITYPE(IT)
      IF(MTYPE.EQ.0) GO TO 10
      IF(NEWTP) WRITE(NEWTAP) OBREC
C STORE OBSERVATION UNTIL PAIRS ARE MATCHED
      TIME1=TIME1+TCOR
      OBS1=OBS1+OBSCOR
      TIME(MTYPE)=TIME1
      STNAM(MTYPE)=STNAM1
      OBS(MTYPE)=OBS1
C CHECK PREPROCESSING INDICATORS
      AMBIG=(FEEDBK(4)-(FEEDBK(4)/256)*256)/16
      CHANEL=AND2(CHNMSK,FEEDBK(4))
      I=PBIT1
      J=PBIT2
      OBIDS(MTYPE)=LOR(I,J)
      MTYPE=MTYPE-MTYPE/8*7
      IF(ONEOB(MTYPE)) GO TO 20
      IF(TIME(MTYPE).NE.TIME(MTYPE+7).OR.STNAM(MTYPE).NE.STNAM(MTYPE+7))
     .   GO TO 10
   20 DAY=TIME(MTYPE)/1.0D+2+REFTIM
```

```
C CONVERT TO A.1 TIME
      IF(ITAG/256.NE.2) DAY=DAY+TDIF(4,3,DAY)/86400.
      IF(DAY.LT.DATALP) GO TO 10
      IF(DAY.GT.DAYSTP) GO TO 200
      MT=MTYPE
C CHECK FOR RATE MEASUREMENTS PRESENT
      IF(NUMBR4(MT,DELETE,7).GT.0) GO TO 10
      INAM=0
      LSTANM(6)=6L
  590 DO 600 I=1,NSTARD
      IF(STNAM1.EQ.JNAME(I)) GO TO 630
  600 CONTINUE
      DO 610 I=1,NUSTOR
      IF(STNAM1.EQ.NAME(I)) GO TO 620
  610 CONTINUE
      INAM=INAM+1
      LSTANM(7)=6L
      IF(INAM.EQ.1) GO TO 590
      PRINT 1000,STNAM1
      GO TO 10
  620 ISN=STANOS(I)
      GO TO 640
  630 ISN=ISTANO(I)
  640 ISTA=NUMBR2(ISN,ISTANO,NSTA)
      IF(ISTA.GT.0) GO TO 650
      ISTA=NSTA+1
      NSTA=ISTA
      ISTANO(NSTA)=ISN
  650 IF(NSIG.LE.0) GO TO 24
      IF(MTYPE.NE.3) GO TO 25
      NN1=0
      DO 23 I=1,NSIG
      IF(ISN.NE.ISTNO(I).AND.ISTNO(I).NE.0) GO TO 23
      IF(3.EQ.IMTYPE(I).OR.IMTYPE(I).EQ.0) NN1=I
   23 CONTINUE
      IF(NN1.GT.0.AND.SIGCHG(NN1).NE.0.) NORATE=.FALSE.
      GO TO 25
   24 NORATE=.FALSE.
   25 CONTINUE
      OBIT=OBITS(MTYPE)
      OBS1=OBS(MTYPE)
      OBS2=OBS(MTYPE+7)
C DELETE REDUNDANT OBSERVATIONS
      IF(MTYPE.EQ.MTFLST.AND.STNAM1.EQ.STNLST.AND.DAY.EQ.DAYLST)
     .  GO TO 10
      MTFLST=MTYPE
      STNLST=STNAM1
      DAYLST=DAY
      NMEAS=2
      IPRE=0
      PRETYP=0
      IF(NPRE.EQ.0) GO TO 90
      IND=0
C DETERMINE PREPROCESSING TO BE DONE
      DO 70 I=1,NPRE
 70   IF((INDPRE(1,I).EQ.0.OR.INDPRE(1,I).EQ.ISN).AND.
```

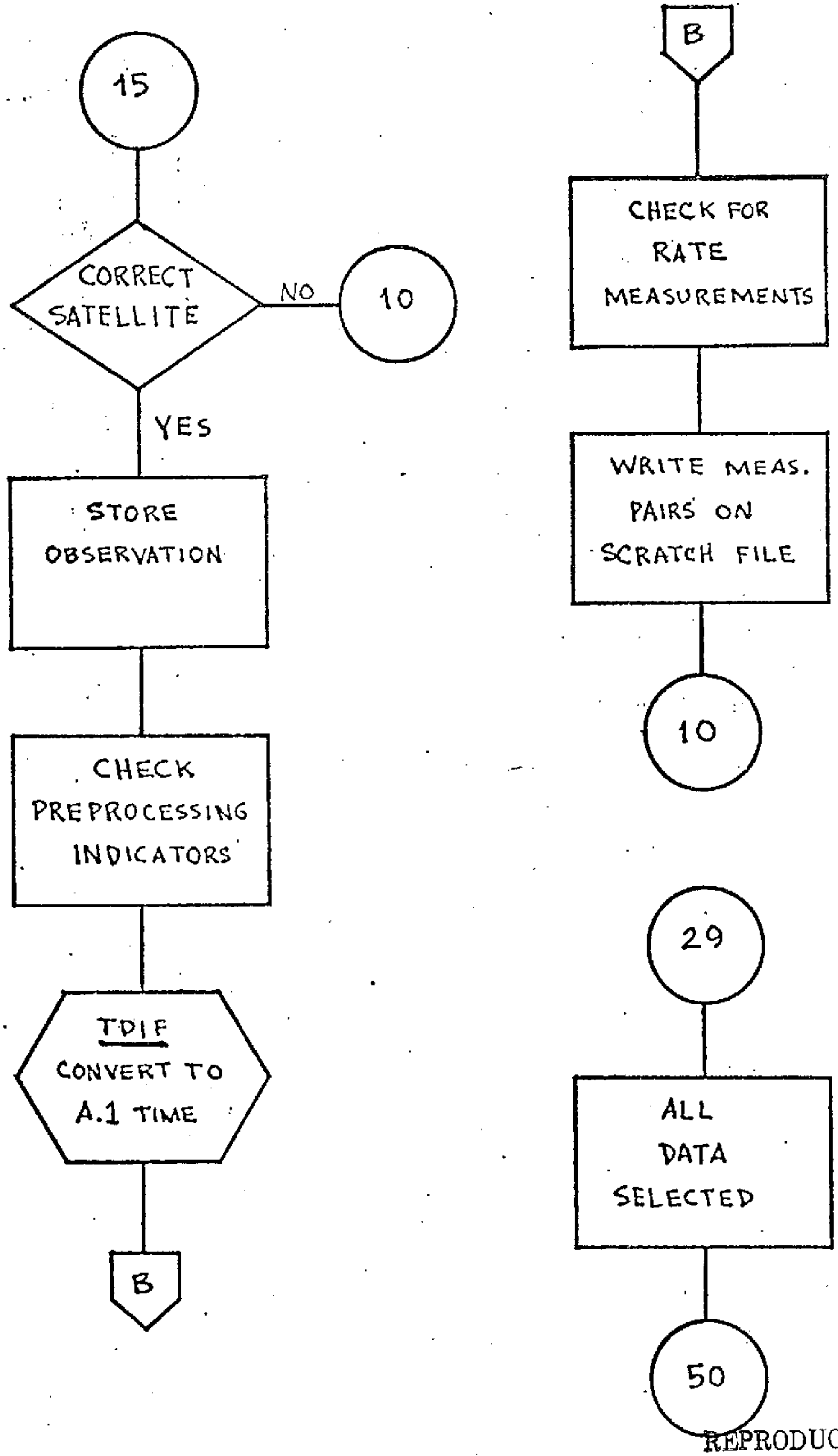
```
     .   (INDPRE(2,I).EQ.0.OR.INDPRE(2,I).EQ.MTYPE))IND=I
      IF(IND.EQ.0) GO TO 90
      SRFNDX=RFINDX(1,IND)
      IF(IPREPR(3,IND).GT.0) DAY=DAY+RFINDX(2,IND)/86400.
      IPRE=IPREPR(1,IND)
      IF(IPRE.EQ.1.AND.AND2(OBIT,IBIT(1)).EQ.1) IPRE=0
      PRETYP=IPREPR(2,IND)
C PROCESS MEASUREMENTS ACCORDING TO TYPE
 90   GO TO (110,120,130,130,150,300,320),MTYPE
C ...OPTICAL
  110 COSD=DCOS(OBS2)
      SIG2=SIGSTD(6)*S2R
      SIG1=SIGSTD(1)*S2R/COSD
      IF(IND.EQ.0) GO TO 400
      IF(IPREPR(4,IND).EQ.0) GO TO 115
      IF(IPREPR(4,IND).EQ.1.AND.AND2(OBIT,IBIT(5)).NE.0) GO TO 115
      IPRE=1
      X(1)=DCOS(OBS1)*COSD
      X(2)=DSIN(OBS1)*COSD
      X(3)=DSIN(OBS2)
      CALL EQUATR(X,DJBASE,.FALSE.,X,DAY,.TRUE.)
      OBS1=DARCTN(X(2),X(1))
      OBS2=DARSIN(X(3))
  115 IF(IPREPR(1,IND).EQ.0) IND=0
      GO TO 400
C ...RANGE RATE
 130  OBS1=OBS1/DLT
      SIG1=0.01*SIGSTD(3)
C ...RANGE & RANGE RATE
 120  OBS1=OBS1+DLL
      NMEAS=1
      IF(MTYPE.EQ.3) GO TO 400
      SIG1=SIGSTD(2)
      IF(IND.EQ.0) GO TO 400
      IF(IPRE.EQ.1.AND.AND2(OBIT,IBIT(6)).NE.0) IPRE=(AMBIG+2)/3
      IF(IPRE.NE.0) DAY=DAY-OBS1/(VLIGHT*8.64D4)
      GO TO 400
C ...MINITRACK
  150 SIG1=OBS1**2
      IF(SIG1.LT.1.) SIG1=SQRT(1.-SIG1)
      SIG1=SIGSTD(5)*1.E-3/SIG1
      SIG2=OBS2**2
      IF(SIG2.LT.1.) SIG2=SQRT(1.-SIG2)
      SIG2=SIGSTD(12)*1.E-3/SIG2
      GO TO 400
C RETURN STATION ZERO
  200 ISTA=0
      MTYPE=0
      IN=0
      IF(NEWTP) END FILE NEWTAP
      RECNO=RECNO+1
      PRINT 3000,NUMBER,IN
      CALL RANDWR
      DAYSTF=DAY
      RETURN
  300 SIG1=SIGSTD(6)*S2R
```

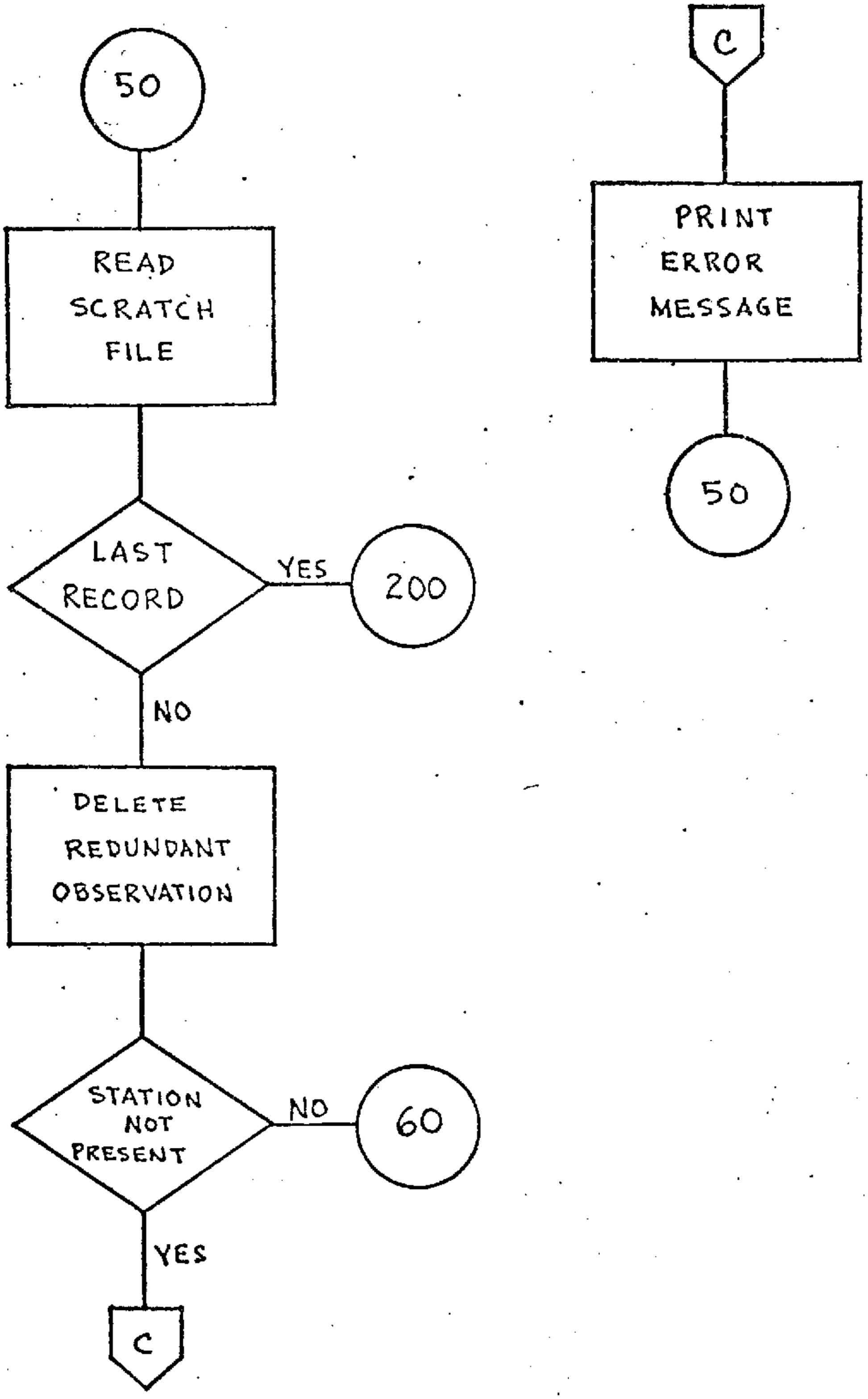
```
                                                                        DODS 224
C ...X-Y ANGLES                                                         DODS 225
      SIG2=SIGSTD(13)*S2R                                               DODS 226
      GO TO 400                                                         DODS 227
  320 SIG1=SIGSTD(7)*S2R                                                DODS 228
C ...AZIMUTH & ELEVATION                                                DODS 229
      SIG2=SIGSTD(14)*S2R                                               DODS 230
  400 IF(NSIG.EQ.0) GO TO 410                                           DODS 231
C MAKE SIGMA CHANGES                                                    DODS 232
      NN1=0                                                             DODS 233
      NN2=0                                                             DODS 234
      DO 405 I=1,NSIG                                                   DODS 235
      IF(ISN.NE.ISTNC(I).AND.ISTNC(I).NE.0) GO TO 405                   DODS 236
      IF(MTYPE.EQ.IMTYPE(I).OR.IMTYPE(I).EQ.0) NN1=I                    DODS 237
      IF(MTYPE+7.EQ.IMTYPE(I).OR.IMTYPE(I).EQ.0) NN2=I                  DODS 238
  405 CONTINUE                                                          DODS 239
      IF(NN1.GT.0) SIG(1)=SIGCHG(NN1)*SIG1/SIGSTD(MTYPE)                DODS 240
      IF(NN2.GT.0) SIG(2)=SIGCHG(NN2)*SIG2/SIGSTD(MTYPE+7)              DODS 241
  410 IF(NCULL.EQ.0) GO TO 420                                          DODS 242
C CULL MEASUREMENTS                                                     DODS 243
      DO 415 I=1,NCULL                                                  DODS 244
      DO 415 J=1,NMEAS                                                  DODS 245
      IF(IM+J-CULL(1,I)) 415,413,411                                    DODS 246
  411 IF(IM+J.GT.CULL(2,I)) GO TO 415                                   DODS 247
  413 SIG(J)=0.0D0                                                      DODS 248
  415 CONTINUE                                                          DODS 249
  420 IM=IM+NMEAS                                                       DODS 250
      PREPRO=NPRE.GT.0.AND.IND.GT.0                                     DODS 251
C SET PREPROCESSING SWITCH                                              DODS 252
      PRETYP=PRETYP+10*IPRE                                             DODS 253
      RECNO=RECNO+1                                                     DODS 254
      IF(NBIAS.GT.0) CALL BIAS                                          DODS 255
      NUMBER=NUMBER+NMEAS                                               DODS 256
      CALL RANDWR                                                       DODS 257
      SIG(2)=0.0D0                                                      DODS 258
      TIME(MTYPE)=0.0D0                                                 DODS 259
      TIME(MTYPE+7)=0.0D0                                               DODS 260
      GO TO 10                                                          DODS 261
 1000 FORMAT (' STATION ',A8,' NOT FOUND')                              DODS 262
 2000 FORMAT(6X,7I2)                                                    DODS 263
 3000 FORMAT(1H0//31X,I6,' OBSERVATIONS SELECTED FROM MASTER DODS ',    DODS 264
     *     'DATA TAPE NUMBER',I3)                                       DODS 265
      END
```



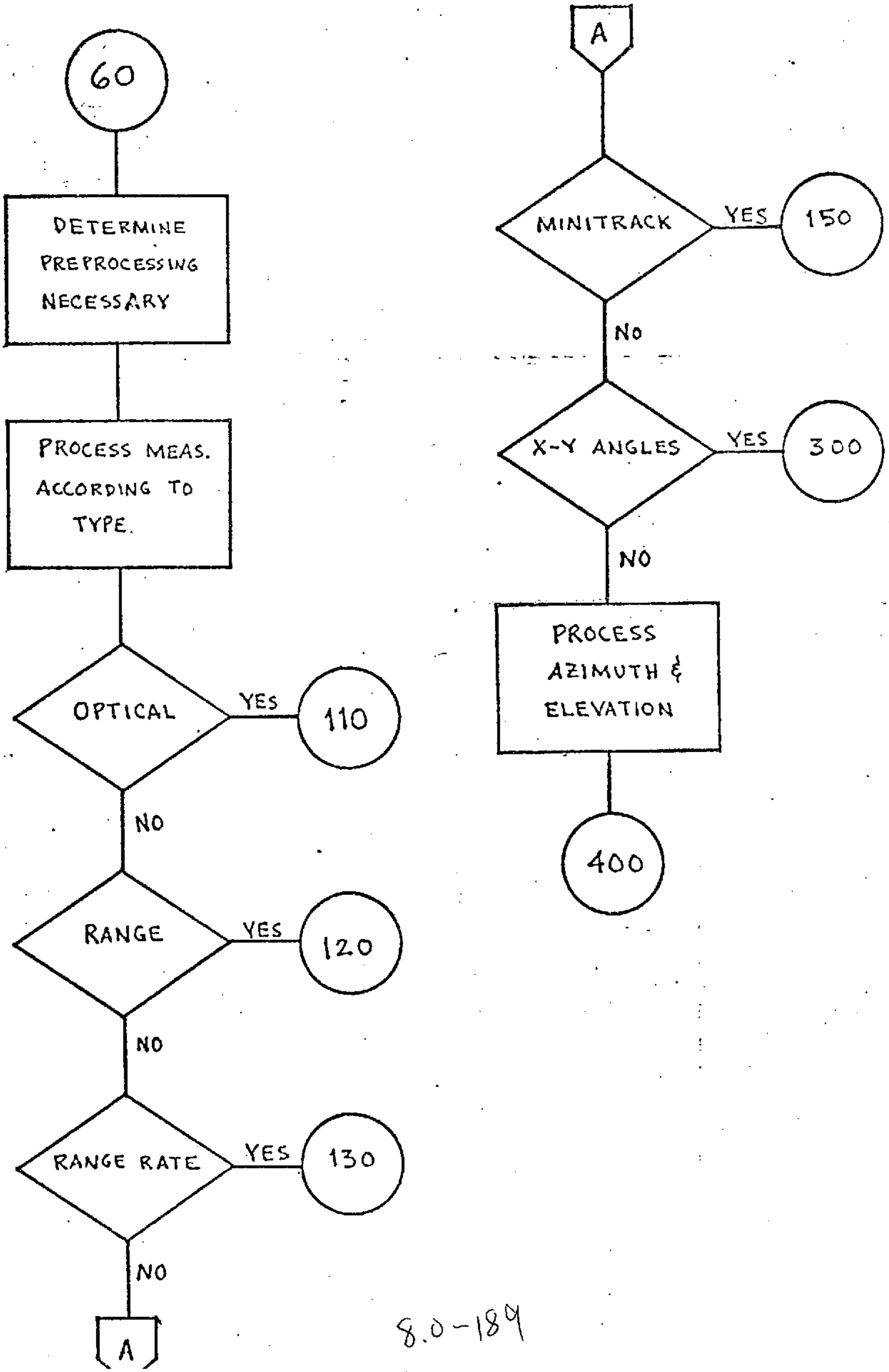
REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

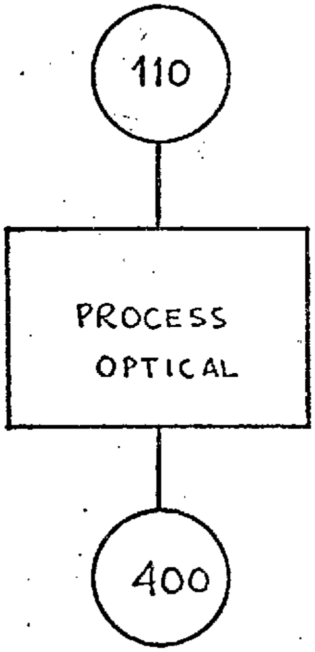
8.0-186

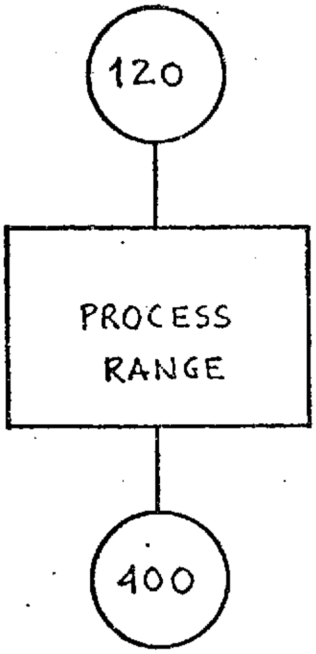

15
CORRECT SATELLITE
NO
10
YES
STORE OBSERVATION
CHECK PREPROCESSING INDICATORS
TDIF
CONVERT TO A.1 TIME
B
B
CHECK FOR RATE MEASUREMENTS
WRITE MEAS. PAIRS ON SCRATCH FILE
10
29
ALL DATA SELECTED
50

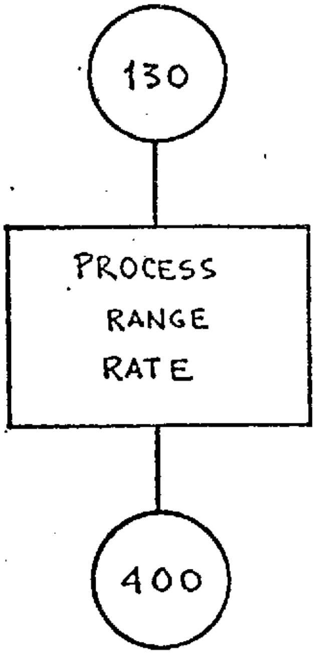


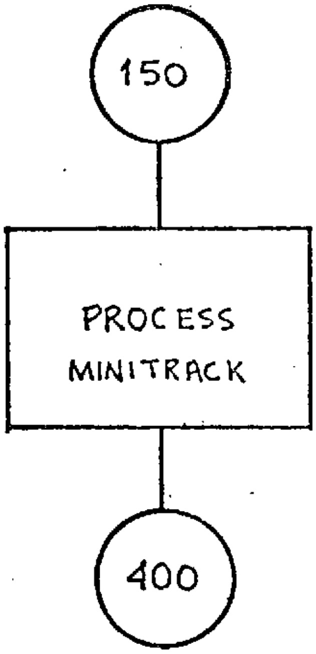
REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

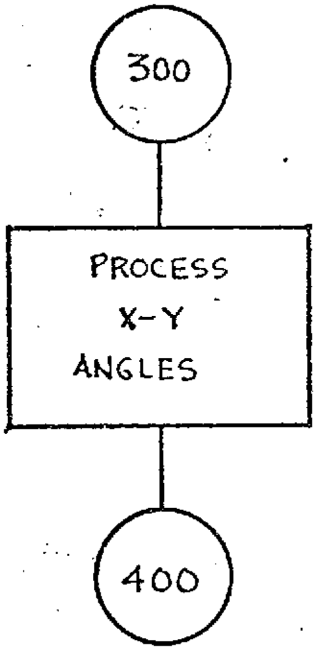

60
DETERMINE PREPROCESSING NECESSARY
PROCESS MEAS. ACCORDING TO TYPE.
OPTICAL
YES
110
NO
RANGE
YES
120
NO
RANGE RATE
YES
130
NO
A
A
MINITRACK
YES
150
NO
X-Y ANGLES
YES
300
NO
PROCESS AZIMUTH & ELEVATION
400

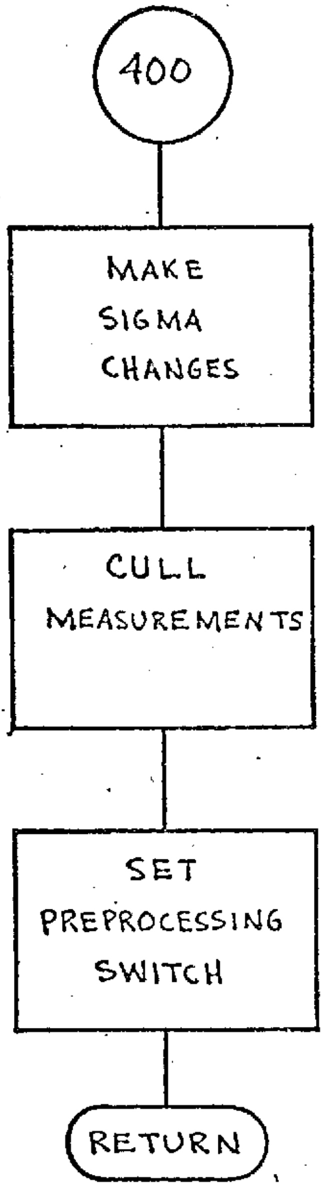

110
PROCESS OPTICAL
400

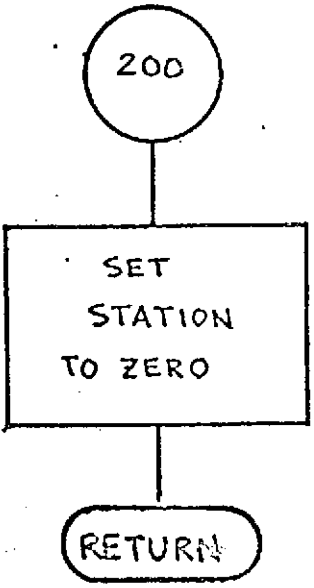

120
PROCESS RANGE
400


130
PROCESS RANGE RATE
400


150
PROCESS MINITRACK
400


300
PROCESS X-Y ANGLES
400


400
MAKE SIGMA CHANGES
CULL MEASUREMENTS
SET PREPROCESSING SWITCH
RETURN


200
SET STATION TO ZERO
RETURN

DPFCT

DESCRIPTION

DPFCT is a multi-purpose multiple entry point function. There exist 8 different points of entry:

DARCTN - is a double argument Arctangent Function providing function values between 0 and 2π raidians

DENORM - computes the denormalization factor for geopotential coefficients

DJUL - computes Julian dates when given dates as computed by subroutine YMDAY

DOTPRD - computes dot products

XEFIX - $X_{ECI}*\cos\ \theta_G + Y_{ECI}*\sin\ \theta_G$

YEFIX - $Y_{ECI}*\cos\ \theta_G - Y_{ECI}*\sin\ \theta_G$

XINERT - $X_{ECF}*\cos\ \theta_G - Y_{ECF}*\sin\ \theta_G$

YINERT - $Y_{ECF}*\cos\ \theta_G + X_{ECF}*\sin\ \theta_G$

NAME	DPFCT
PURPOSE	MULTI-PURPOSE MULTIPLE ENTRY POINT FUNCTION

ENTRY POINT	PURPOSE
DARCTN	TO COMPUTE THE ARCTANGENT BETWEEN 0 & TWO PI
DENORM	TO COMPUTE THE DENORMALIZATION FACTOR FOR GEOPOTENTIAL COEFFICIENTS
DJUL	TO COMPUTE JULIAN DATE FOR AN INPUT TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR FOR THE ARC
DOTPRD	TO COMPUTE THE DOT PRODUCT OF 2 THREE DIMENSIONAL VECTORS
XEFIX	GIVEN INERTIAL X AND Y RETURN EARTH FIXED X
XINERT	GIVEN EARTH FIXED X AND Y RETURN INERTIAL X
YEFIX	GIVEN INERTIAL X AND Y RETURN EARTH FIXED Y
YINERT	GIVEN EARTH FIXED X AND Y RETURN INERTIAL Y

CALLING SEQUENCE DARCTN(Y,X)

SYMBOL	TYPE	DESCRIPTION
Y	DP	INPUT - R*SIN(A)
X	DP	INPUT - R*COS(A)
DARCTN	DP	OUTPUT - ARCTANGENT OF ANGLE A IN RADIANS BETWEEN 0 AND +2PI RADIANS

CALLING SEQUENCE DENORM(N,M)

SYMBOL	TYPE	DESCRIPTION
N	I	INPUT - DEGREE OF LEGENDRE POLYNOMIALS
M	I	INPUT - ORDER OF LEGENDRE POLYNOMIALS
DENORM	DP	OUTPUT - DENORMALIZATION FACTOR

CALLING SEQUENCE DJUL(X)

SYMBOL	TYPE	DESCRIPTION
X	DP	INPUT - TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
DJUL	DP	OUTPUT - JULIAN DATE

CALLING SEQUENCE DOTPRD(A,B)

SYMBOL	TYPE	DESCRIPTION
A (3)	DP	INPUT - COMPONENTS OF VECTOR A
B (3)	DP	INPUT - COMPONENTS OF VECTOR B
DOTPRD	DP	OUTPUT - DOT PRODUCT OF VECTORS A & B

CALLING SEQUENCE XEFIX(X,Y)

SYMBOL	TYPE	DESCRIPTION
X	DP	INPUT - INERTIAL X
Y	DP	INPUT - INERTIAL Y.
XEFIX	DP	OUTPUT - EARTH FIXED X POSITION

CALLING SEQUENCE XINERT(X,Y)

SYMBOL	TYPE	DESCRIPTION
X	DP	INPUT - EARTH FIXED X
Y	DP	INPUT - EARTH FIXED Y
XINERT	DP	OUTPUT - INERTIAL X

CALLING SEQUENCE YEFIX(X,Y)

SYMBOL	TYPE	DESCRIPTION
X	DP	INPUT - INERTIAL X
Y	DP	INPUT - INERTIAL Y
YEFIX	DP	OUTPUT - EARTH FIXED Y POSITION

CALLING SEQUENCE YINERT(X,Y)

SYMBOL	TYPE	DESCRIPTION
X	DP	INPUT - EARTH FIXED X
Y	DP	INPUT - EARTH FIXED Y
YINERT	DP	OUTPUT - INERTIAL Y

SUBROUTINES USED YMDAY

COMMON BLOCKS CSTHET CONSTS INITBK

INPUT FILES NONE

OUTPUT FILES NONE

```
      DOUBLE PRECISION FUNCTION DARCTN(Y,X)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/CSTHET/CTHETG,STHETG
      COMMON/CONSTS/PI,TWOPI,DRAD,DRSEC
      COMMON/INITBK/IG1(49),NDJIST,IG2(7)
      LOGICAL NDJIST
      DARCTN=DATAN2(Y,X)
      IF(DARCTN.LT.0.0D0) DARCTN=TWOPI+DARCTN
      RETURN
      ENTRY DENORM(N,M)
      IF(M.EQ.0)GO TO 120
      XN=4.0D0*DFLOAT(N)+2.0D0
      NMM1=N-M+1
      NPM=N+M
      FACT=1.D0
      DO 110 I=NMM1,NPM
  110 FACT=FACT*DFLOAT(I)
      XN=XN/FACT
      GO TO 130
  120 XN=2.0D0*DFLOAT(N)+1.0D0
  130 DENORM=DSQRT(XN)
      RETURN
      ENTRY DJUL(X)
      IF(NDJIST) GO TO 10
      NDJIST=.TRUE.
      DJ=2433281.5D0-YMDAY(500100,0,0.0D0)
   10 DJUL=DJ+X
      RETURN
      ENTRY DOTPRD(A,B)
      DOUBLE PRECISION A(3),B(3)
      DOTPRD=A(1)*B(1)+A(2)*B(2)+A(3)*B(3)
      RETURN
      ENTRY XEFIX(X,Y)
      XEFIX=X*CTHETG+Y*STHETG
      RETURN
      ENTRY XINERT(X,Y)
      XINERT=X*CTHETG-Y*STHETG
```

```
      RETURN
      ENTRY YEFIX(X,Y)
      YEFIX=Y*CTHETG-X*STHETG
      RETURN
      ENTRY YINCRT(X,Y)
      YINCRT=Y*CTHETG+X*STHETG
      RETURN
      END
```

NAME DRAG

ENTRY POINT	PURPOSE
DRAG1	INITIALIZATION
DRAG	TO COMPUTE ACCELERATION IN RECTANGULAR COORDINATES DUE TO AERO-DYNAMIC DRAG FORCES

CALLING SEQUENCE CALL DRAG1(GRPAR)

SYMBOL	TYPE	DESCRIPTION
GRPAR (1)	DP	OUTPUT - PARTIALS OF FORCE MODEL PARAMETERS

CALLING SEQUENCE DRAG(RASAT,DX,DT)

SYMBOL	TYPE	DESCRIPTION
RASAT	DP	INPUT - RIGHT ASCENSION OF SATELLITE
DX	DP	INPUT & OUTPUT - SATELLITE ACCELERATION VECTOR
DT	DP	INPUT - TIME IN DAYS FROM EPOCH

SUBROUTINES USED DENSTY

COMMON BLOCKS DRGBLK XYZ INTBLK

INPUT FILES NONE

OUTPUT FILES NONE

REFERENCES ' GEODYN SYSTEMS DESCRIPTION '
VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE DRAG1(GRPAR)                                           DRAG  41
      IMPLICIT REAL*8 (A-H,O-Z)                                         DRAG  42
      DOUBLE PRECISION DX(3),X,Y,Z,XDOT,YDOT,ZDOT,R,RSQ,THDT2S,AE,RASAT DRAG  43
      INTEGER ADDR,ADDRD                                                DRAG  44
      DIMENSION GRPAR(1)                                                DRAG  45
      COMMON/DRGBLK/HT,SPSISQ,C(4),C3,C1,VEL,XDOTR,YDOTR,RHO            DRAG  46
      COMMON/XYZ/X,Y,Z,XDOT,YDOT,ZDOT,R,RSQ,ISAT,IFORCE(2)              DRAG  47
      COMMON/INTBLK/THDOT1(2),THDT2S,GM,AE,AESQ,FLAT,FSQ32,FFSQ32,      DRAG  48
     .   GM3(6),B(2),BDOT(2),B0(2),AFGM(38),ADDR(2),ADDRD(2),SRAD(5)    DRAG  49
      RETURN                                                            DRAG  50
      ENTRY DRAG(RASAT,DX,DT)                                           DRAG  51
      SPSISQ=Z**2/RSQ                                                   DRAG  52
      HT=R-AE-(FSQ32*SPSISQ**2-FFSQ32*SPSISQ)                           DRAG  53
C OBTAIN ATMOSPHERIC DENSITY                                            DRAG  54
      RHO=DENSTY(RASAT)                                                 DRAG  55
```

```
      XDOTR=XDOT+THDT2S*Y
      YDOTR=YDOT-THDT2S*X
      VEL=XDOTR**2+YDOTR**2+ZDOT**2
      VEL=DSQRT(VEL)
      C3=VEL*RHO
      C1=C3*(B(ISAT)+BDOT(ISAT)*DT)
C SUM IN ACCELERATION DUE TO DRAG
      DX(1)=DX(1)-C1*XDOTR
      DX(2)=DX(2)-C1*YDOTR
      DX(3)=DX(3)-C1*ZDOT
      IND=IFORCE(ISAT)*3
      IF(ADDR(ISAT).EQ.0) GO TO 100
C COMPUTE PARTIALS FOR DRAG
      IND=IND+3
      C2=-C3*B0(ISAT)
      GRPAR(1)=C2*XDOTR
      GRPAR(2)=C2*YDOTR
      GRPAR(3)=C2*ZDOT
  100 IF(ADDRD(ISAT).EQ.0) RETURN
C COMPUTE PARTIALS FOR DRAG RATE
      C2=-C3*B0(ISAT)*DT
      GRPAR(IND+1)=C2*XDOTR
      GRPAR(IND+2)=C2*YDOTR
      GRPAR(IND+3)=C2*ZDOT
      RETURN
      END
```

EGRAV

DESCRIPTION

EGRAV is the GEODYN System subroutine for determining the acceleration on the satellite due to the Earth's gravitation. Certain intermediate data are saved for use in subroutines VEVAL, RESPAR, AVGPOT and GEOIDH.

The intermediate data mentioned above are computed following the determination of the Earth-fixed spherical coordinates of the satellite. They are:

$$\frac{GM}{r} \qquad \frac{a^{n}}{r}$$

$$P_n^m (\sin \phi)$$

$$\sin m \lambda$$

$$\cos m \lambda$$

$$m \tan \phi$$

for each m and/or n. Multiple angle formulae are used for the sine and cosines and the usual recursive relationships are used for the Legendre functions $P_n^m (\sin \phi)$.

The accelerations are computed in Earth-fixed spherical coordinates and then converted to inertial Cartesian coordinates.

```
NAME                EGRAV

PURPOSE             TO COMPUTE ACCELERATION DUE TO GRAVITY

CALLING SEQUENCE    CALL EGRAV(THETG,RASAT,DX)

   SYMBOL  TYPE     DESCRIPTION

   THETG    DP      INPUT - APPARENT RIGHT ASCENSION OF GREENWICH

   RASAT    DP      INPUT & OUTPUT - RIGHT ASCENSION OF SATELLITE

   DX       DP      OUTPUT - ACCELERATION VECTOR IN METERS/SECONDS**2

SUBROUTINE USED     CLEAR

COMMON BLOCKS       CSLIM     FMODEL     INIT3K     INTBLK     XYZ
                    VRBLOK

INPUT FILES         NONE

OUTPUT FILES        NONE

REFERENCES          'GEODYN SYSTEMS DESCRIPTION'
                    VOLUME 1 - GEODYN DOCUMENTATION
```

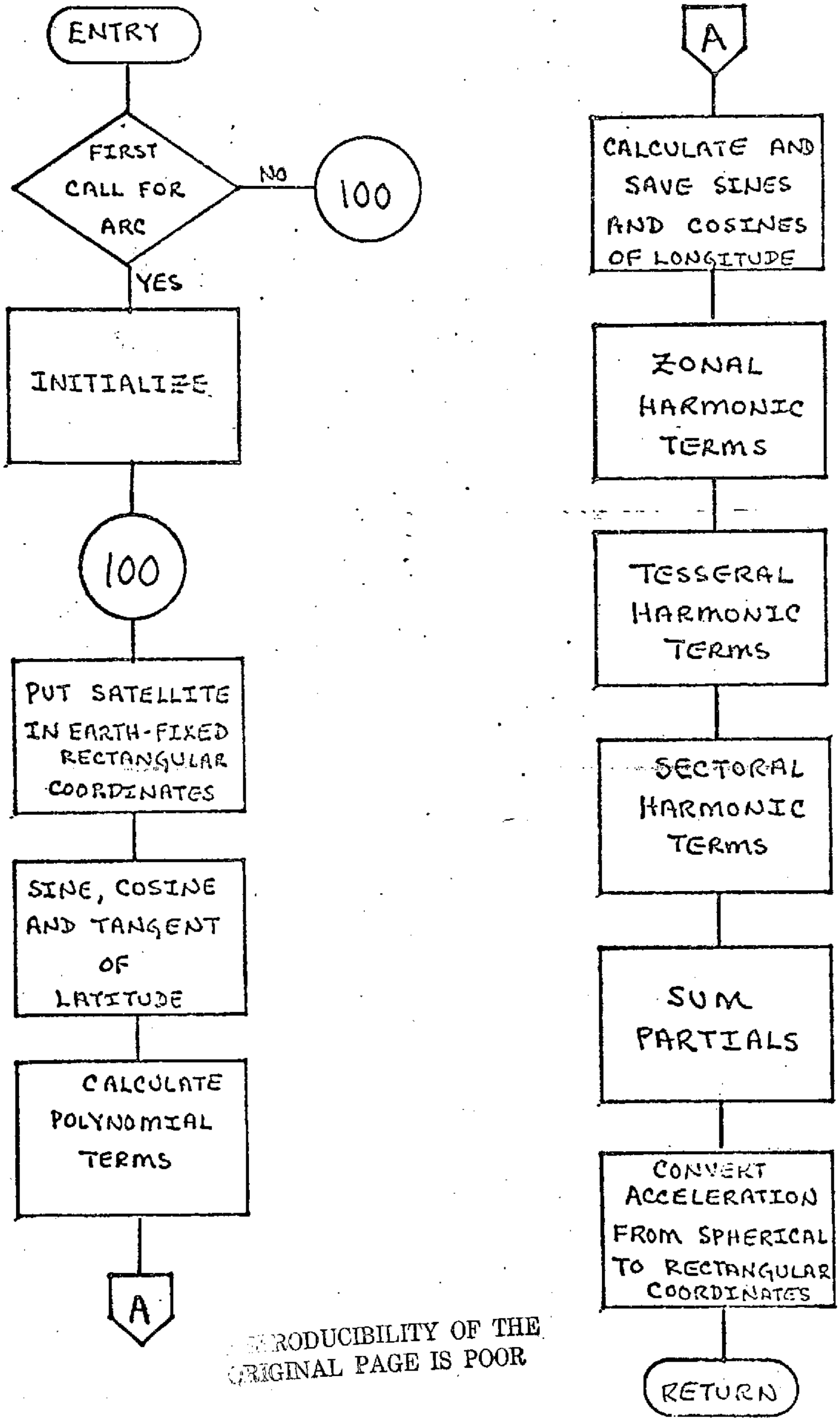
```
****NOTATION****

    PSI-LATITUDE (GEOCENTRIC)
    LAMBDA-LONGITUDE (+EASTWARD)
    R-GEOCENTRIC RADIUS TO SATELLITE IN EARTH RADII
    GM-GRAVITATIONAL CONSTANT TIMES MASS OF EARTH
    P(M,N)-COEFFICIENTS OF LEGENDRE POLYNOMIAL
    C(N,M)-COEFFICIENTS OF COSINE FUNCTION
    S(N,M)-COEFICIENTS OF SINE FUNCTION
    INDEX1-DEGREE OF SUMMATION PLUS 1
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      SUBROUTINE EGRAV(THETG,RASAT,DX)                                   EGRA  42
      IMPLICIT REAL*8(A-H,O-Z)                                           EGRA  43
      LOGICAL NOT1ST                                                     EGRA  44
      INTEGER ULIMIT                                                     EGRA  45
      DOUBLE PRECISION MODEL,LAMBDA                                      EGRA  46
      DIMENSION C(30,33),S(30,33),DX(3)                                  EGRA  47
      COMMON/CSLIM/LLIMIT(31),ULIMIT(31)                                 EGRA  48
      COMMON/FMODEL/INDEX1,INDEX2,INDEX3,INDEX4,CS(30,33),MODEL(3)       EGRA  49
      COMMON/INITBK/IG1(50),NOT1ST,NSW1(6)                               EGRA  50
      COMMON/INTBLK/ITHDTS(6),GM,AE,IDUM(122)                            EGRA  51
      COMMON/XYZ/X,Y,Z,XDOT,YDOT,ZDOT,R,RSQ,ISAT,IFORCE(3)               EGRA  52
      COMMON/VRBLOK/XYSQ,COSLAM(31),SINLAM(31),RR,RPSI,RLAMDA,           EGRA  53
     *   P(33,33),AORN(33),TPSIM(33)                                     EGRA  54
      EQUIVALENCE (TPSIM(2),TANPSI),(C(1,1),S(1,1),CS(1,1)),(P(1,1),     EGRA  55
```

```
     .          SINPSI),(P(2,1),COSPSI),(TPSIM(1),ZERO)
C INITIALIZE SUMMATION INDICES
      IF(NOT1ST) GO TO 100
      CALL CLEAR(P,60,33)
      CALL CLEAR(TPSIM,64,1)
      CALL CLEAR(LLIMIT,31,1)
      CALL CLEAR(ULIMIT,31,1)
      ZERO=0.D0
      DO 50 NC=2,INDEX2
      NS=31-NC
      N1=MIN0(NC+1,INDEX3)
      DO 25 MC=2,N1
      MS=34-MC
      IF(C(NC,MC).EQ.0.D0.AND.S(NS,MS).EQ.0.D0)GO TO 25
      IF(LLIMIT(NC).EQ.0) LLIMIT(NC)=MC
      ULIMIT(NC)=MC
   25 CONTINUE
   50 CONTINUE
      NOT1ST=.TRUE.
C PUT SATELLITE IN EARTH CENTERED - EARTH FIXED RECTANGULAR COORDINATES
  100 RASAT=DATAN2(Y,X)
      XYSQ=X**2+Y**2
      RTXYSQ=DSQRT(XYSQ)
      LAMBDA=RASAT-THETG
      SINLAM(2)=DSIN(LAMBDA)
      COSLAM(2)=DCOS(LAMBDA)
      RSQ=XYSQ+Z**2
      R=DSQRT(RSQ)
C SINE, COSINE, AND TANGENT OF LATITUDE
      SINPSI=Z/R
      COSPSI=RTXYSQ/R
      TANPSI=SINPSI/COSPSI
C CONVERT R TO EARTH RADII
      RINV=AE/R
C CALCULATE POLYNOMIAL TERMS   ...NOTE... P TAKES FORM P(M,N)
      CP3=3.D0*COSPSI
      P(1,2)=1.5D0*SINPSI**2-.5D0
      P(2,2)=CP3*SINPSI
      P(3,2)=CP3*COSPSI
      TPSIM(3)=2.D0*TANPSI
C CALCULATE AND SAVE SINES AND COSINES OF LONGITUDE
      CL2=2.D0*COSLAM(2)
      SINLAM(3)=CL2*SINLAM(2)
      COSLAM(3)=CL2*COSLAM(2)-1.D0
      GMR=GM/R
      ADEN(2)=RINV**2*GMR
      DO 120 N=3,INDEX2
      ADEN(N)=ADEN(N-1)*RINV
      F1=N
      F2=F1-1.D0
      F3=2.D0*F1-1.D0
      F4=F3*COSPSI
      N1=N-1
      N2=N-2
C ZONAL HARMONICS (M=0)
      P(1,N)=(F3*SINPSI*P(1,N1)-F2*P(1,N2))/F1
```

```
      IF(INDEX3.LT.2)GO TO 120                                          EGRA 112
      NX=MIN0(N,INDEX3+2)                                               EGRA 113
      DO 110 M=2,NX                                                     EGRA 114
C TESSERAL HARMONICS (M NON-ZERO, LESS THAN N)                           EGRA 115
 110  P(M,N)=P(M,N2)+F4*P(M-1,N1)                                       EGRA 116
      IF(NX.LT.N)GO TO 120                                              EGRA 117
      NN1=N+1                                                           EGRA 118
C SECTORAL HARMONICS ( M=N, NON-ZERO)                                   EGRA 119
      P(NN1,N)=F4*P(N,N1)                                               EGRA 120
      TPSIM(NN1)=TPSIM(N)+TANPSI                                        EGRA 121
      SINLAM(NN1)=CL2*SINLAM(N)-SINLAM(N1)                              EGRA 122
      COSLAM(NN1)=CL2*COSLAM(N)-COSLAM(N1)                              EGRA 123
 120  CONTINUE                                                          EGRA 124
C INITIALIZATION FOR SUMMATION FOR PARTIALS                             EGRA 125
140   PR=0.D0                                                           EGRA 126
      PLAMDA=0.D0                                                       EGRA 127
      PPSI=0.D0                                                         EGRA 128
      FN1=2.D0                                                          EGRA 129
C SUMMATION FOR PARTIALS                                                EGRA 130
      DO 250 NC=2,INDEX2                                                EGRA 131
      NS=31-NC                                                          EGRA 132
      FN1=FN1+1.D0                                                      EGRA 133
      DLAMDA=0.D0                                                       EGRA 134
      F1=C(NC,1)*COSLAM(1)                                              EGRA 135
      DP=F1*P(1,NC)                                                     EGRA 136
      DPSI=F1*(P(2,NC)-TPSIM(1)*P(1,NC))                                EGRA 137
      NLL=LLIMIT(NC)                                                    EGRA 138
      IF(NLL.EQ.0) GO TO 225                                            EGRA 139
      N1=ULIMIT(NC)                                                     EGRA 140
      FM=NLL-1                                                          EGRA 141
      DO 200 MC=NLL,N1                                                  EGRA 142
      MS=34-MC                                                          EGRA 143
      P1=P(MC,NC)                                                       EGRA 144
C PARTIAL WRT LAMBDA (SUMMATION)                                        EGRA 145
      DLAMDA=DLAMDA+FM*P1*(S(NS,MS)*COSLAM(MC)-C(NC,MC)*SINLAM(MC))     EGRA 146
      F1=C(NC,MC)*COSLAM(MC)+S(NS,MS)*SINLAM(MC)                        EGRA 147
      IF(F1.EQ.0.D0)GO TO 200                                           EGRA 148
C PARTIAL WRT R (SUMMATION)                                             EGRA 149
      DP=DP+F1*P1                                                       EGRA 150
C PARTIAL WRT PSI (SUMMATION)                                           EGRA 151
      DPSI=DPSI+F1*(P(MC+1,NC)-TPSIM(MC)*P1)                            EGRA 152
200   FM=FM+1.D0                                                        EGRA 153
      PLAMDA=PLAMDA+DLAMDA*AORN(NC)                                     EGRA 154
  225 PR=PR+DP*FN1*AORN(NC)                                             EGRA 155
 250  PPSI=PPSI+DPSI*AORN(NC)                                           EGRA 156
C COMPLETE PARTIAL WRT R                                                EGRA 157
      PR=-(GMR+PR)/R                                                    EGRA 158
C CONVERT ACCELERATION IN SPHERICAL COORDINATES TO ACCELERATION IN      EGRA 159
C RECTANGULAR COORDINATES (MULTIPLY BY MATRIX OF PARTIALS OF SPHERICAL  EGRA 160
C WITH RESPECT TO RECTANGULAR)                                          EGRA 161
      PRR=PR/R                                                          EGRA 162
      PLXY=PLAMDA/XYSQ                                                  EGRA 163
      PRTP=PRR-PPSI*Z/(RTXYSQ*RSQ)                                      EGRA 164
      DX(1)=X*PRTP-PLXY*Y                                               EGRA 165
      DX(2)=Y*PRTP+PLXY*X                                               EGRA 166
      DX(3)=PRR*Z+PPSI*RTXYSQ/RSQ                                       EGRA 167
      RETURN                                                            EGRA 168
      END                                                               EGRA 169
```


ENTRY
FIRST CALL FOR ARC
NO
100
YES
INITIALIZE
100
PUT SATELLITE IN EARTH-FIXED RECTANGULAR COORDINATES
SINE, COSINE AND TANGENT OF LATITUDE
CALCULATE POLYNOMIAL TERMS
A
A
CALCULATE AND SAVE SINES AND COSINES OF LONGITUDE
ZONAL HARMONIC TERMS
TESSERAL HARMONIC TERMS
SECTORAL HARMONIC TERMS
SUM PARTIALS
CONVERT ACCELERATION FROM SPHERICAL TO RECTANGULAR COORDINATES
RETURN

ELEM

DESCRIPTION

ELEM converts the inertial rectangular components of the position and velocity vectors of a satellite to the corresponding osculating Kepler elements.

Subroutine ERROR is invoked if the square of the eccentricity is not less than one.

NAME	ELEM
PURPOSE	TO CONVERT INERTIAL POSITION AND VELOCITY VECTOR TO OSCULATING ORBITAL ELEMENTS
CALLING SEQUENCE	CALL ELEM(XYZ,AEI,IDRAD,FRSTME,AEIXYZ)

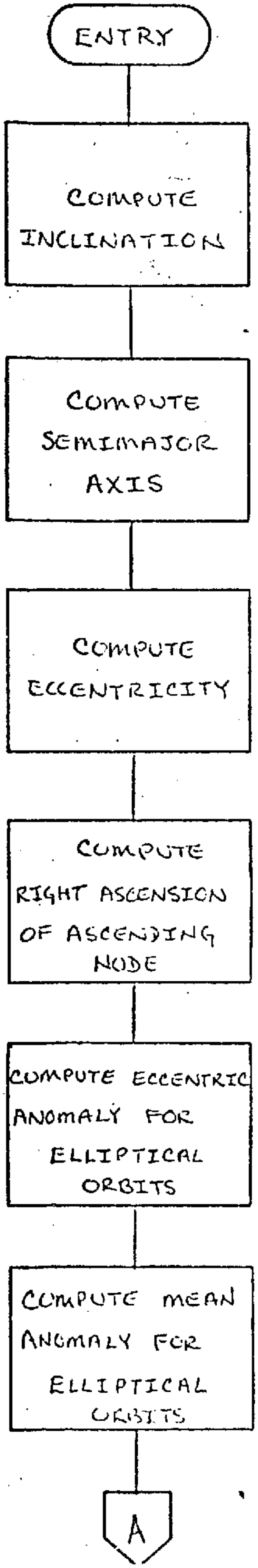
SYMBOL	TYPE	DESCRIPTION
XYZ (6)	DP	INPUT - CARTESIAN ELEMENTS
AEI (6)	DP	OUTPUT - KEPLER ELEMENTS
IDRAD	I	INPUT - =3 - OUTPUT IS IN RADIANS AND PARTIAL MATRIX IS REQUESTED =2 - OUTPUT IS IN RADIANS =1 - OUTPUT IS IN DEGREES
FRSTME	L	INPUT - FIRST TIME SWITCH
AEIXYZ (6,3,2)	DP	OUTPUT - PARTIALS OF KEPLER ELEMENTS WITH RESPECT TO X, Y, Z

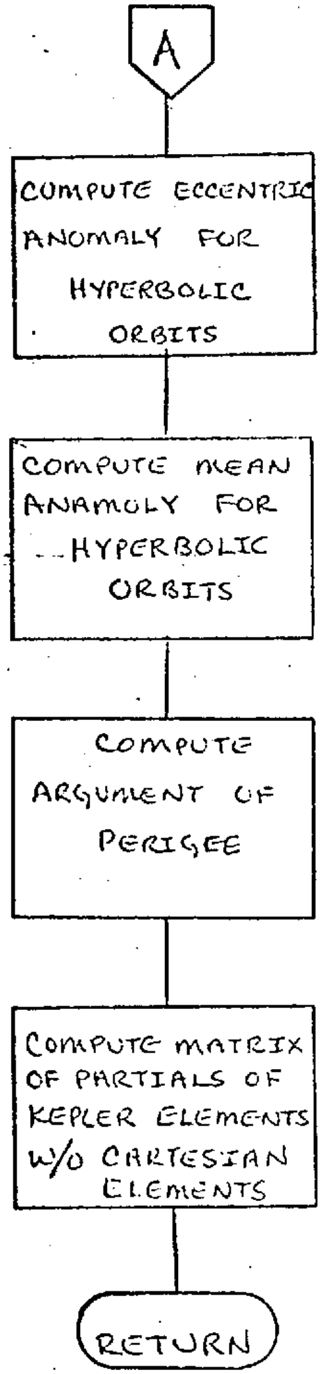
SUBROUTINES USED	ERROR			
COMMON BLOCKS	ALPMRC	CELEM	CONSTS	INTBLK
INPUT FILES	NONE			
OUTPUT FILES	NONE			
REFERENCES	'GEODYN SYSTEMS DESCRIPTION' VOLUME 1 - GEODYN DOCUMENTATION			

```
      SUBROUTINE ELEM(XYZ,AEI,IDRAD,FRSTME,AEIXYZ)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL FRSTME,HYPER,HYPSW
      REAL RMSTUT
      DOUBLE PRECISION INCL,MEAN,NODE
      DIMENSION AEI(6),XYZ(6),AEINPM(6),XYZXYZ(6),AEIXYZ(6,3,2),PHIP(3)
      COMMON/ALPMRC/ALPHA(54),HYPSW
      COMMON/CELEM/ELM(24),XNU,EC,RMSTUT
      COMMON/CONSTS/PI,TWOPI,RAD,RSEC
      COMMON/INTBLK/THDOTS(3),GM,AE(62)
      EQUIVALENCE (A,AEINPM(1)),(E,AEINPM(2)),(INCL,AEINPM(3)),
     .   (NODE,AEINPM(4)),(P,AEINPM(5)),(MEAN,AEINPM(6)),
     .   (X,XYZXYZ(1)),(Y,XYZXYZ(2)),(Z,XYZXYZ(3)),
     .   (XDOT,XYZXYZ(4)),(YDOT,XYZXYZ(5)),(ZDOT,XYZXYZ(6))
      DO 10 I=1,6
   10 XYZXYZ(I)=XYZ(I)
```

```
      R=DSQRT(X**2+Y**2+Z**2)
      VSQ=XDOT**2+YDOT**2+ZDOT**2
      C1=Y*ZDOT-Z*YDOT
      C2=Z*XDOT-X*ZDOT
      C3=X*YDOT-Y*XDOT
      HSQ=C1**2+C2**2+C3**2
      H=DSQRT(HSQ)
      COSI=C3/H
      SINI=DSQRT(1.0D0-COSI**2)
C COMPUTE INCLINATION
      INCL=DATAN2(SINI,COSI)
      IF (INCL.LT.0.0D0) INCL=INCL+TWOPI
      AINV=2.0D0/R-VSQ/GM
C COMPUTE SEMI-MAJOR AXIS
      A=1.0D0/AINV
C COMPUTE ECCENTRICITY
      ESQ=1.0D0-HSQ*AINV/GM
      E=DSQRT(ESQ)
      HYPER=ESQ.GE.1.0D0
      IF(HYPER.AND.FRSTME.AND..NOT.HYPSW) CALL ERROR(1,ESQ)
      IF(HYPER.AND..NOT.(HYPSW.OR.FRSTME)) CALL ERROR(2,ESQ)
      P=A*(1.0D0-ESQ)
      P=A*(1.0D0-ESQ)
      RRDOT=X*XDOT+Y*YDOT+Z*ZDOT
      HSI=H*SINI
      SINN=C1/HSI
      COSN=-C2/HSI
C COMPUTE LONGITUDE OF ASCENDING NODE .
      NODE=DATAN2(SINN,COSN)
      IF(NODE.LT.0.0D0)NODE=NODE+TWOPI
      PER=P/(E*R)
      SNU=RRDOT*PER/H
      CNU=PER-1.0D0/E
      XNU=DATAN2(SNU,CNU)
      IF(XNU.LT.0.0D0)XNU=XNU+TWOPI
      CEC=(CNU+E)/(1.0D0+E*CNU)
      SEC=(DSQRT(DABS(1.0D0-ESQ))*SNU)/(1.0D0+E*CNU)
      IF (HYPER) GO TO 15
      EC = DATAN2(SEC,CEC)
      IF(EC.LT.0.0D0)EC = EC + TWOPI
C COMPUTE MEAN ANOMALY
      MEAN = EC - E*SEC
      GO TO 16
   15 EC=DLOG(CEC+SEC)
      MEAN=E*SEC-EC
   16 CU=( X*COSN+Y*SINN)/R
      SU=((Y*COSN-X*SINN)*COSI+Z*SINI)/R
      U=DATAN2(SU,CU)
      IF(U.LT.0.0D0)U=U+TWOPI
C COMPUTE ARGUMENT OF PERIGEE
      P=U-XNU
      IF(P.LT.0.0D0)P=P+TWOPI
      SCALE=1.0D0
      IF(JDRAD.EQ.1) SCALE=1.0D0/RAD
      DO 20 I=1,6
      AEI(I)=AEINPM(I)
```

```
      IF(I.LE.2) GO TO 20
      AEI(I)=AEINRM(I)*SCALE
   20 CONTINUE
      IF (ILRAD.LT.3) RETURN
C COMPUTE PARTIALS OF KEPLER ELEMENTS W/O CARTESIAN ELEMENTS
      DO 40 I=1,3
      PX=(4-I)/3
      PY=1-MOD(I,2)
      I1=MOD(I,3)+4
      I2=MOD(I1,3)+1
      AEIXYZ(1,I,1)=2.0D0*A**2/R**3*XYZXYZ(I)
      AEIXYZ(1,I,2)=2.0D0*A**2*XYZXYZ(I+3)/GM
      DO 40 J=1,2
      DO 30 K=1,3
      PHIP(K)=0.0D0
      IF (I.EQ.K) GO TO 30
      L=(4-J)*3-I-K
      PHIP(K)=XYZXYZ(L)
      IF (L.NE.I1.AND.L.NE.I2) PHIP(K)=-PHIP(K)
   30 CONTINUE
      PHP=(C1*PHIP(1)+C2*PHIP(2)+C3*PHIP(3))/H
      PR=2-J
      PX=PR*PX
      PY=PR*PY
      PR=PR*XYZXYZ(I)/R
      AEIXYZ(2,I,J)=(H*AINV/2.0D0*AEIXYZ(1,I,J)-PHP)*H*AINV/(E*GM)
      AEIXYZ(3,I,J)=(COSI*PHP-PHIP(3))/HSI
      AEIXYZ(4,I,J)=(SINN*(C3*AEIXYZ(3,I,J)+SINI*PHP)-PHIP(1))/C2
      PUP=((X*SINN-Y*COSN)*AEIXYZ(4,I,J)-COSN*PX-SINN*PY+CU*PR)/(R*SU)
      PNUP=(E*(1.0D0-ESQ)*(A*PR-R*AEIXYZ(1,I,J))-(R+(R-A*(1.0D0+ESQ))*
     *     AEIXYZ(2,I,J))/(ESQ*R**2*SNU)
      AEIXYZ(5,I,J)=PUP-PNUP
      PECP=(AINV*(PR-R*AINV*AEIXYZ(1,I,J))+CEC*AEIXYZ(2,I,J))/(E*SEC)
      AEIXYZ(6,I,J)=R*AINV*PECP-SEC*AEIXYZ(2,I,J)
   40 CONTINUE
      RETURN
      END
```


ENTRY
COMPUTE INCLINATION
COMPUTE SEMIMAJOR AXIS
COMPUTE ECCENTRICITY
COMPUTE RIGHT ASCENSION OF ASCENDING NODE
COMPUTE ECCENTRIC ANOMALY FOR ELLIPTICAL ORBITS
COMPUTE MEAN ANOMALY FOR ELLIPTICAL ORBITS
A


A
COMPUTE ECCENTRIC ANOMALY FOR HYPERBOLIC ORBITS
COMPUTE MEAN ANAMOLY FOR HYPERBOLIC ORBITS
COMPUTE ARGUMENT OF PERIGEE
COMPUTE MATRIX OF PARTIALS OF KEPLER ELEMENTS w/o CARTESIAN ELEMENTS
RETURN

EPHEM

DESCRIPTION

EPHEM is a subroutine specifically designed to read an ephemeris tape prepared from a JPL planetary ephemeris tape but containing only those ephemerides used by GEODYN.

The ephemerides read by EPHEM are those of the Sun, the Moon, Venus, Mars, Jupiter, Saturn, and nutation in right ascension. Lunar and nutation data are provided on the tape at half day intervals. All other data are provided at 4 day intervals. Double buffers used by EPHEM provide GEODYN with 16 days of resident ephemeris data.

EPHEM interpolates the data to any desired time within the data span present on the data tape using a fifth order Everett scheme.

The data read and output by EPHEM is in true coordinates of date.

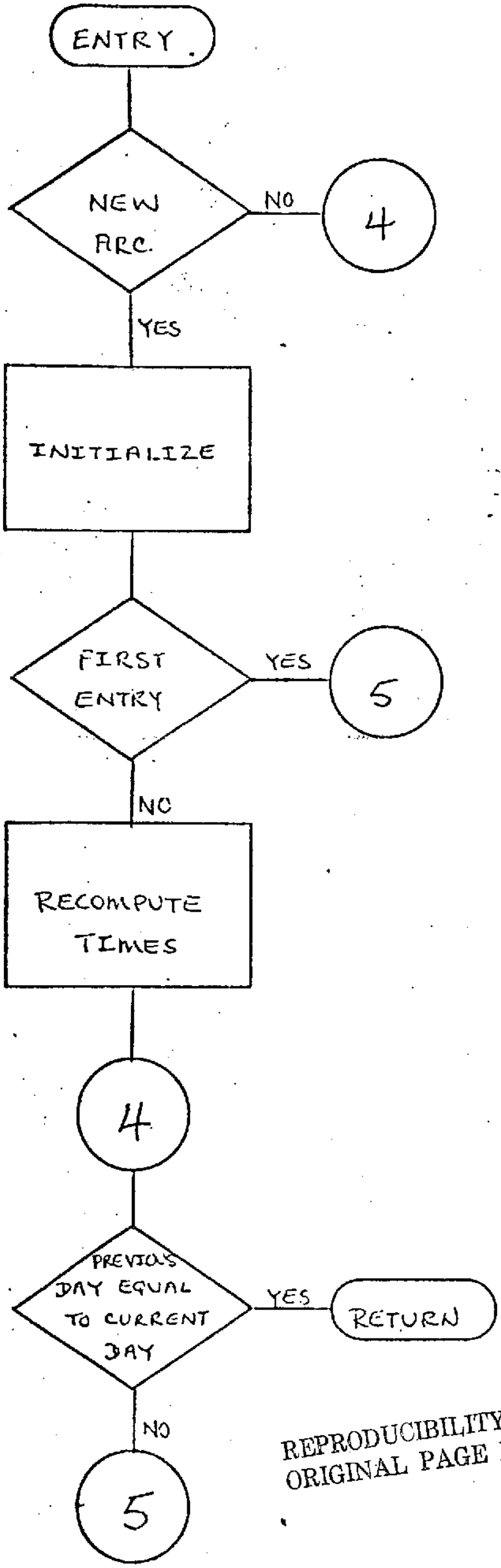
```
NAME                 EPHEM

PURPOSE              1) READ LUNAR , SOLAR & PLANETARY EPHEMERIDES AND
                        NUTATION IN RIGHT ASCENSION
                     2) INTERPOLATE THE DATA USING FIFTH ORDER EVERETT
                        SCHEME

CALLING SEQUENCE     CALL EPHEM(DAY,ONLYEQ)

   SYMBOL  TYPE      DESCRIPTION

   DAY      DP       INPUT - TIME AT WHICH DATA IS DESIRED IN DAYS FROM
                             JAN 0.0 OF THE REFERENCE YEAR FOR THE ARC

   ONLYEQ   L        INPUT - TRUE REQUESTS ONLY NUTATION BE COMPUTED

SUBROUTINES USED     ERROR      TDIF      YMDAY

COMMON BLOCKS        CEPHEM     INITBK    INTBLK

INPUT FILES          DATP - EPHEMERIS DATA FILE

OUTPUT FILES         PRINTER

REFERENCES           'GEODYN OPERATIONS DESCRIPTION' - APPENDIX C
                     VOLUME 3 - GEODYN DOCUMENTATION

                     'GEODYN SUPPORT PROGRAMS' - EPHEMERIS TAPE GENERATOR
                     VOLUME 4 - GEODYN DOCUMENTATION
```

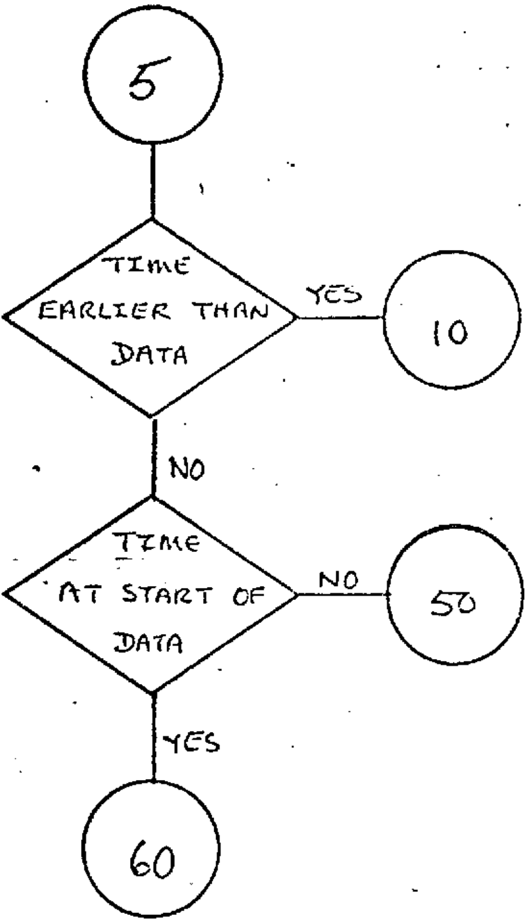
```
      SUBROUTINE EPHEM(DAY,ONLYEQ)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL NOT1ST,ONLYEQ
      INTEGER DATP
      REAL ANUT,BUF1N,BUF2N,SEC
      DIMENSION BUF1M1(51),BUF1M2(51),BUF1M3(51),BUF2M1(51),
     .   BUF2M2(51),BUF2M3(51),BUF1S(27),BUF2S(27),BUF1VM(54),
     .   BUF2VM(54),BUF1JS(54),BUF2JS(54),BUF1N(51),BUF2N(51),IYMD(2),
     .   IHM(2),SEC(2)
      COMMON/CEPHEM/A0(25),PMOON(306),SUN(270),ANUT(102),DUMMY(16)
      COMMON/INITBK/IG1(51),NOT1ST,IG2(5)
      COMMON/INTBLK/THDOT1(2),THDT2S,IG3(125),KBODY
      EQUIVALENCE (BUF1M1(1),PMOON(1)),(BUF1M2(1),PMOON(52)),
     .   (BUF1M3(1),PMOON(103)),(BUF2M1(1),PMOON(154)),
     .   (BUF2M2(1),PMOON(205)),(BUF2M3(1),PMOON(256)),
     .   (BUF1S(1),SUN(1)),(BUF1VM(1),SUN(28)),
     .   (BUF1JS(1),SUN(82)),(BUF2S(1),SUN(136)),
     .   (BUF2VM(1),SUN(163)),(BUF2JS(1),SUN(217)),
     .   (BUF1N(1),ANUT(1)),(BUF2N(1),ANUT(52)),
     .   (S,FS0)
      DATA DATP/1/
      DATA DAYR/9999.0D0/
```

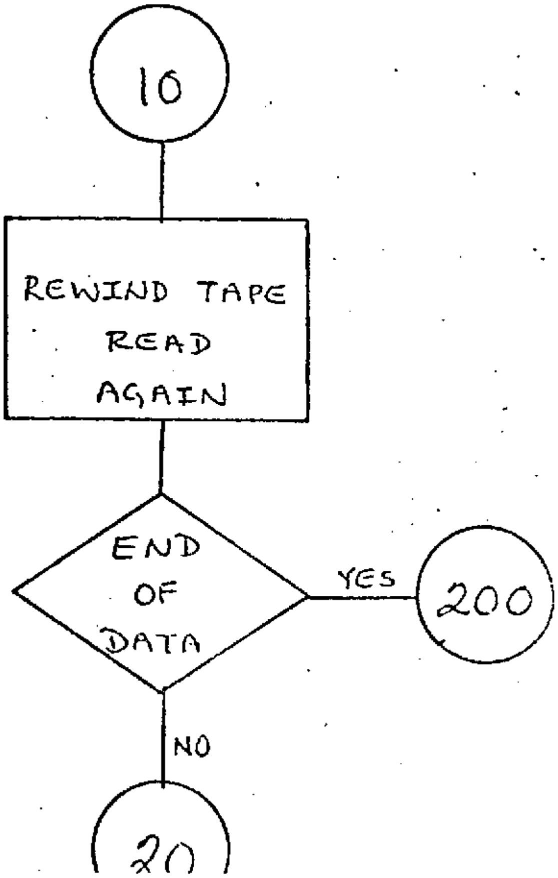
```
      F2(S)=S*(S**2-1.0D0)/6.0D0
      F4(S,FS2)=FS2*(S*S-4.0D0)*0.5D-01
      IF(NOT1ST) GO TO 4
C INITIALIZE
      NOT1ST=.TRUE.
      NBODY=KBODY-1
      IBODY=KBODY*4
      FACTOR=1.0D0/32.302D0
      IF(DAYR.GT.9998.D0) GO TO 5
      SEC1=SEC(1)
C IF NOT FIRST ENTRY THEN RECOMPUTE TIMES
      DAYR1=YMDAY(IYMD(1),IHM(1),SEC1)
      SEC1=SEC(2)
      DAYR2=YMDAY(IYMD(2),IHM(2),SEC1)
      IF (N.EQ.1) GO TO 2
      DAYO=DAYR1
      DAYR=DAYR2
      GO TO 3
    2 DAYO=DAYR2
      DAYR=DAYR1
    4 IF(DAYPRV.EQ.DAY) RETURN
    5 IF(DAY-DAYR) 10,60,50
C IF DATA IN CORE LATER THAN REQUESTED TIME THEN REWIND TAPE AND READ
C AGAIN
   10 REWIND DATP
      READ(DATP,END=200) IYMD(1),IHM(1),SEC(1),BUF1N,BUF1S
      READ(DATP,END=200) BUF1M1
      READ(DATP,END=200) BUF1M2
      READ(DATP,END=200) BUF1M3
      READ(DATP,END=200) BUF1VM
      READ(DATP,END=200) BUF1JS
      IF(IYMD(1).EQ.0) GO TO 200
      SEC1=SEC(1)
      DAYR1=YMDAY(IYMD(1),IHM(1),SEC1)
      READ(DATP,END=200) IYMD(2),IHM(2),SEC(2),BUF2N,BUF2S
      READ(DATP,END=200) BUF2M1
      READ(DATP,END=200) BUF2M2
      READ(DATP,END=200) BUF2M3
      READ(DATP,END=200) BUF2VM
      READ(DATP,END=200) BUF2JS
      IF(IYMD(2).EQ.0) GO TO 200
      SEC1=SEC(2)
      DAYR2=YMDAY(IYMD(2),IHM(2),SEC1)
      DAYR=DAYR1
      DAYO=DAYR2
      N=1
   20 IF(DAY-DAYR) 300,60,50
C READ NEW RECORD
   30 READ(DATP,END=200) IYMD(1),IHM(1),SEC(1),BUF1N,BUF1S
      READ(DATP,END=200) BUF1M1
      READ(DATP,END=200) BUF1M2
      READ(DATP,END=200) BUF1M3
      READ(DATP,END=200) BUF1VM
      READ(DATP,END=200) BUF1JS
      IF(IYMD(1).EQ.0) GO TO 200
      N=2
```

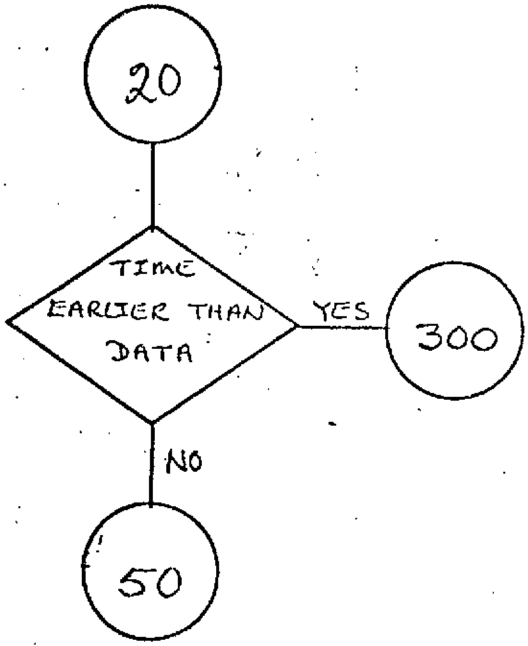
```
      SEC1=SEC(1)
      DAYR1=YMDAY(IYMD(1),IHM(1),SEC1)
      DAYR=DAYO
      DAYO=DAYR1
      GO TO 50
C READ NEW RECORD
   40 READ(DATP,END=200) IYMD(2),IHM(2),SEC(2),BUF2N,BUF2S
      READ(DATP,END=200) BUF2M1
      READ(DATP,END=200) BUF2M2
      READ(DATP,END=200) BUF2M3
      READ(DATP,END=200) BUF2VM
      READ(DATP,END=200) BUF2JS
      N=1
      SEC1=SEC(2)
      DAYR2=YMDAY(IYMD(2),IHM(2),SEC1)
      DAYR=DAYO
      DAYO=DAYR2
   50 IF(DAY.GT.DAYO+3.0D0) GO TO (30,40),N
C DATA FOUND AT CORRECT TIME - INTERPOLATE
   60 DAYI=DAYO
      K=MOD(N,2)+1
      IF(DAY.GE.DAYO) GO TO 70
      DAYI=DAYR
      K=N
   70 INDEX=IDINT((DAY-DAYI)*2.0D0)
      DAYI=DAYI+0.5D0*DFLOAT(INDEX)
      S=(DAY-DAYI)*2.0D0
      FOS=1.0D0-S
      FS2=F2(S)
      F2S=F2(FOS)
      F4S=F4(FOS,F2S)
      FS4=F4(S,FS2)
      INDEX0=INDEX*3+(K-1)*51
C ...NUTATION
      A0(25)=ANUT(INDEX0+3)*F4S + ANUT(INDEX0+6)*FS4 +
     1        ANUT(INDEX0+2)*F2S + ANUT(INDEX0+5)*FS2 +
     2        ANUT(INDEX0+1)*FOS + ANUT(INDEX0+4)*FS0
      A0(25)=A0(25)+TDIF(1,4,DAY)*THDT2S
      DAYPRV=1.0D+50
C IF ONLY NUTATION REQUESTED, RETURN
      IF(ONLYEQ) RETURN
      INDEX5=(K-1)*153+INDEX*9
      INDEX6=INDEX5+9
C ...LUNAR EPHEMERIS
      DO 90 I=1,3
      I3=I*3-3
      INDEX1=INDEX5+I3
      INDEX2=INDEX6+I3
   90 A0(I)=PMOON(INDEX1+3)*F4S + PMOON(INDEX2+3)*FS4 +
     1       PMOON(INDEX1+2)*F2S + PMOON(INDEX2+2)*FS2 +
     2       PMOON(INDEX1+1)*FOS + PMOON(INDEX2+1)*FS0
      INDEX=IDINT((DAY-DAYI)*0.25D0)
      DAYI=DAYI+4.0D0*DFLOAT(INDEX)
      S=(DAY-DAYI)*0.25D0
      FOS=1.0D0-S
      FS2=F2(S)
```

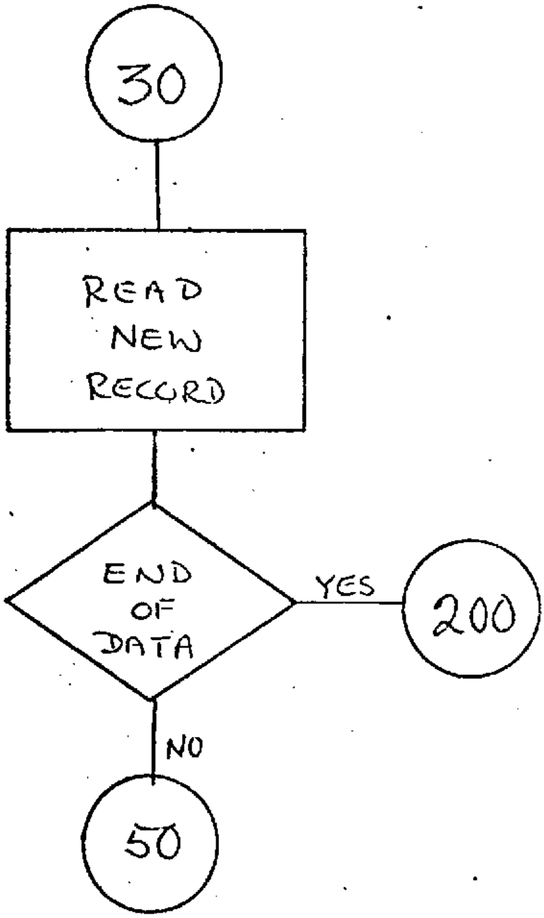
```
      F2S=F2(F0S)
      F4S=F4(F0S,F2S)
      FS4=F4(S,FS2)
      INDEX5=(K-1)*135+INDEX*9
      INDEX6=INDEX5+9
C ...SUN & PLANETS (VENUS,MARS,JUPITER,SATURN)
      DO 100 J=1,NBODY
      INDEX0=4*J
      J27=J*27-27
      INDEX3=INDEX5+J27
      INDEX4=INDEX6+J27
      DO 100 I=1,3
      I3=I*3-3
      INDEX1=INDEX3+I3
      INDEX2=INDEX4+I3
  100 A0(INDEX0+I)=SUN(INDEX1+3)*F4S + SUN(INDEX2+3)*FS4 +
     .   SUN(INDEX1+2)*F2S + SUN(INDEX2+2)*FS2 +
     .   SUN(INDEX1+1)*F0S + SUN(INDEX2+1)*FS0
C CONVERT SUN TO GEOCENTRIC
      A0(5)=A0(5)+FACTOR*A0(1)
      A0(6)=A0(6)+FACTOR*A0(2)
      A0(7)=A0(7)+FACTOR*A0(3)
      IF(NBODY.LT.2) GO TO 110
C CONVERT PLANET TO GEOCENTRIC
      DO 105 I=2,NBODY
      I1=I*4
      A0(I1+1)=A0(I1+1)+A0(5)
      A0(I1+2)=A0(I1+2)+A0(6)
  105 A0(I1+3)=A0(I1+3)+A0(7)
  110 DO 120 I=1,IBODY,4
      L=I+3
      A0(L)=A0(I)*A0(I)+A0(I+1)*A0(I+1)+A0(I+2)*A0(I+2)
      A0(L)=DSQRT(A0(L))
      L1=L-1
      DO 120 J=I,L1
  120 A0(J)=A0(J)/A0(L)
      DAYPRV=DAY
      RETURN
C NO DATA AVAILABLE THIS LATE
  200 WRITE(6,1000) IYMD(K),IHM(K),SEC(K)
      CALL ERROR(7,DAY)
C NO DATA AVAILABLE THIS EARLY
  300 WRITE(6,2000) IYMD(N),IHM(N),SEC(N)
      CALL ERROR(7,DAY)
 1000 FORMAT('1PROGRAM TERMINATED'/'0INSUFFICIENT EPHEMERIS DATA'/
     1         '0LAST DATA POINT',I8,I6,F8.5/)
 2000 FORMAT('1PROGRAM TERMINATED'/'0INSUFFICIENT EPHEMERIS DATA'/
     1         '0FIRST DATA POINT',I8,I6,F8.5/)
      STOP
      END
```

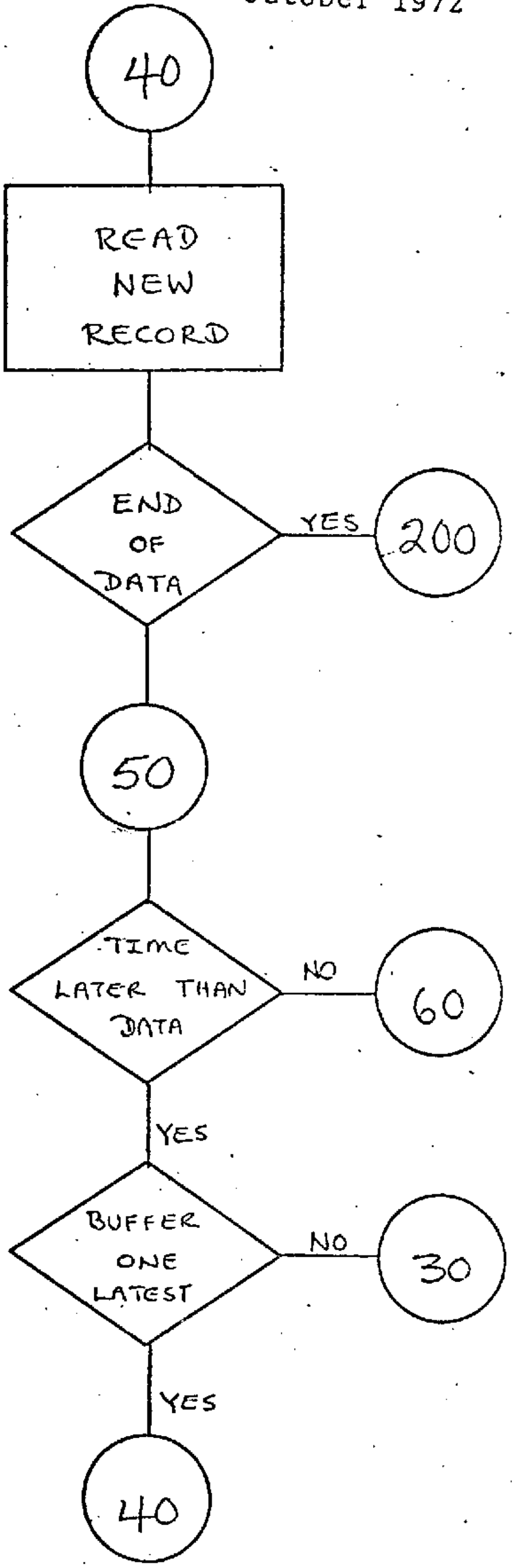

ENTRY
NEW ARC.
NO
4
YES
INITIALIZE
FIRST ENTRY
YES
5
NO
RECOMPUTE TIMES
4
PREVIOUS DAY EQUAL TO CURRENT DAY
YES
RETURN
NO
5

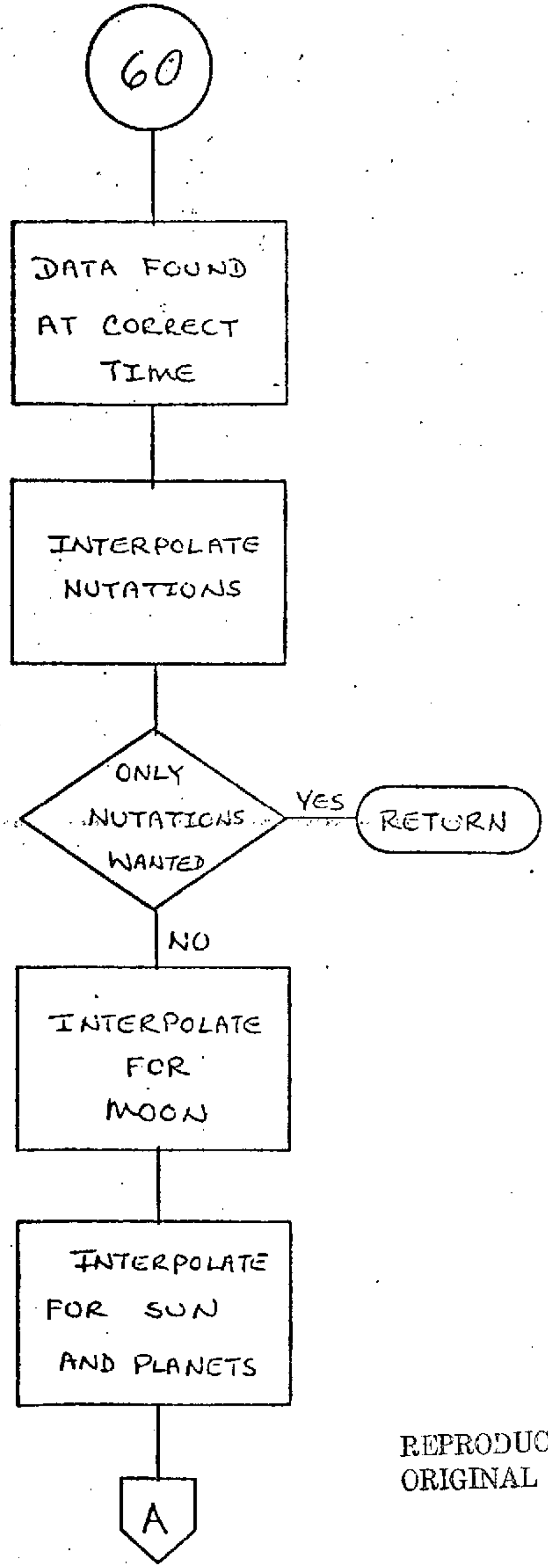

5
TIME EARLIER THAN DATA
YES
10
NO
TIME AT START OF DATA
NO
50
YES
60

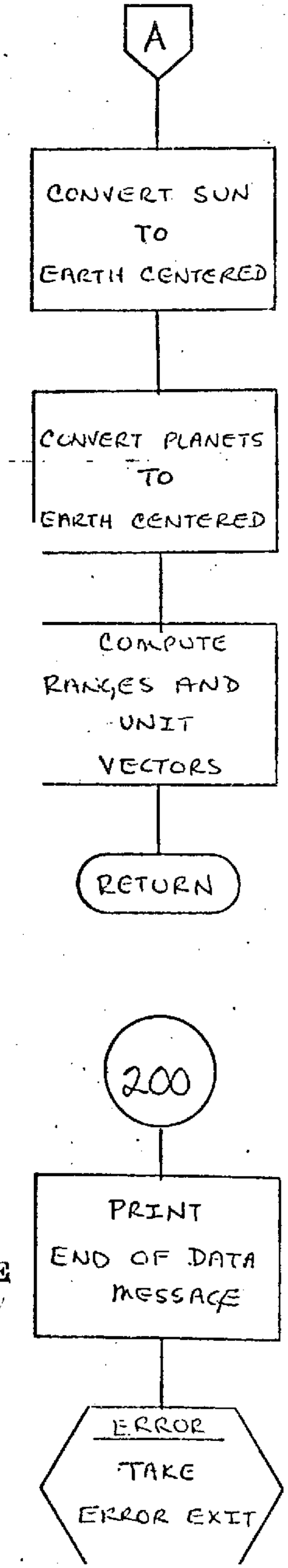

10
REWIND TAPE READ AGAIN
END OF DATA
YES
200
NO
20


20
TIME EARLIER THAN DATA
YES
300
NO
50

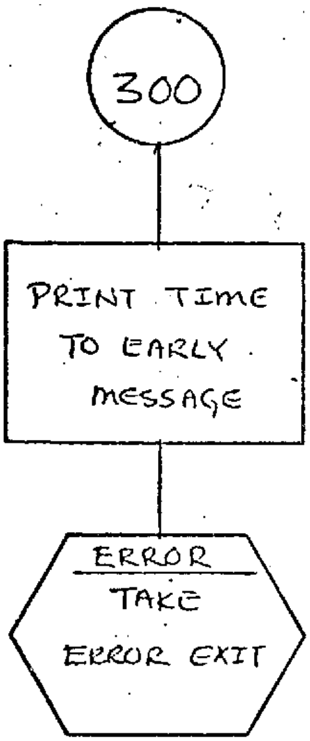

30
READ NEW RECORD
END OF DATA
YES
200
NO
50


40
READ NEW RECORD
END OF DATA
YES
200
50
TIME LATER THAN DATA
NO
60
YES
BUFFER ONE LATEST
NO
30
YES
40





REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR


300
PRINT TIME TO EARLY MESSAGE
ERROR
TAKE ERROR EXIT

EQN

DESCRIPTION

EQN is a real function whose value is the equation of the equinoxes (nutation in right ascension). EQN also returns the nutation in longitude, the nutation in obliquity, and the true obliquity of date.

Multiple angle formulae have been used to reduce the massive number of trigonometric evaluations in Woolard's solution.

NAME EQN

PURPOSE COMPUTES NUTATION IN LONGITUDE, NUTATION IN OBLIQUITY, TRUE OBLIQUITY OF DATE, AND NUTATION IN RIGHT ASCENSION (EQUATION OF THE EQUINOX)

CALLING SEQUENCE EQN(DJ,DPSI,DE,E)

SYMBOL	TYPE	DESCRIPTION
DJ	DP	INPUT - JULIAN DATE
DPSI	DP	OUTPUT - NUTATION IN LONGITUDE - RADIANS
DE	DP	OUTPUT - NUTATION IN OBLIQUITY - RADIANS
E	DP	OUTPUT - TRUE OBLIQUITY - RADIANS
EQN	DP	OUTPUT - NUTATION IN RIGHT ASCENSION - RADIANS (EQUATION OF THE EQUINOX)

SUBROUTINES USED NONE

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

RESTRICTIONS ALL PERIODIC TERMS IN WOOLARD'S THEORY WITH COEFFICIENTS LESS THAN .001 SECONDS OF ARC HAVE BEEN NEGLECTED. ALL SECULAR PORTIONS OF THE COEFFICIENTS HAVE BEEN NEGLECTED WHENEVER THE SECULAR COEFFICIENTS ARE LESS THAN .001 SECONDS OF ARC.

REFERENCES THE FORMULATION BY EDGAR W. WOOLARD MAY BE FOUND IN 4 PUBLICATIONS -

1. ASTRONOMICAL PAPERS PREPARED FOR THE USE OF AMERICAN EPHEMERIS AND NAUTICAL ALMANAC - VOLUME 15, PART 1, PAGE 153 (THEORY OF THE ROTATION OF THE EARTH AROUND ITS CENTER OF MASS - BY EDGAR W. WOOLARD)

2. EXPLANATORY SUPPLEMENT TO THE ASTRONOMICAL EPHEMERIS AND THE AMERICAN EPHEMERIS AND NAUTICAL ALMANAC - PAGES 44 AND 45

3. IMPROVED LUNAR EPHEMERIS 1952-1959 - A JOINT SUPPLEMENT TO THE AMERICAN EPHEMERIS AND THE (BRITISH) NAUTICAL ALMANAC - PAGES IX AND X

4. ASTRONOMICAL JOURNAL, 1953 FEBRUARY, VOL. 58, NO. 1, PAGES 1-3 (A REDEVELOPMENT OF THE

THEORY OF NUTATION - BY EDGAR W. WOOLARD)

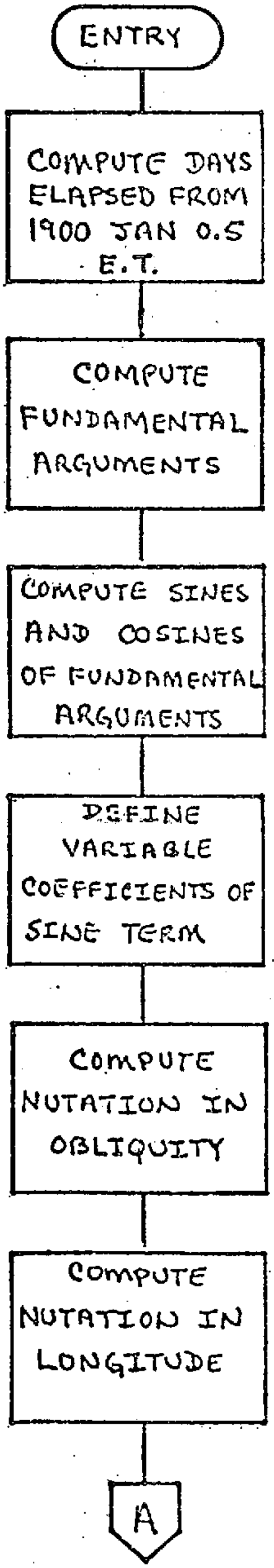
'GEODYN SYSTEMS DESCRIPTION' - SEC. 2.3.6 & 2.3.6.2
VOLUME 1 - GEODYN DOCUMENTATION

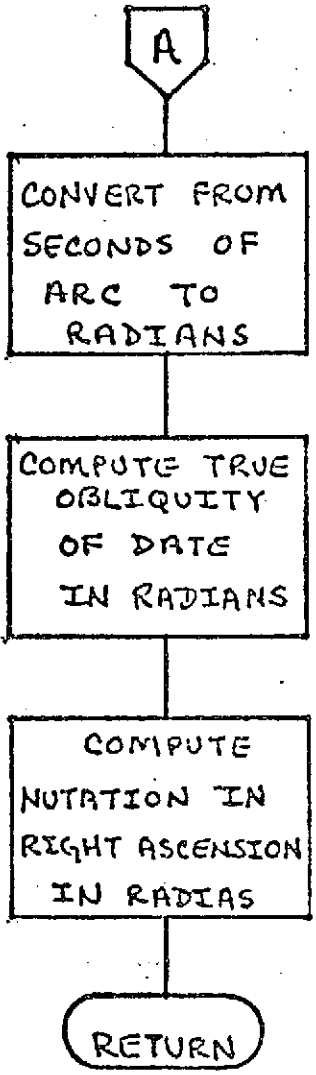
"EXPLANATORY SUPPLEMENT TO THE AMERICAN EPHEMERIS &
NAUTICAL ALMANAC", PP. 44-45,98

```
      FUNCTION EQN(DJ,DPSI,DE,E)
      DOUBLE PRECISION DJ,D,C0,C1,DPSI,DE,E
      DIMENSION SCF(34),S(34),CCF(19),C(19),FARG(5),C0(5),C1(5),C2(5),
     1          C3(5)
      DATA C0/+5.168000345745D0,    +6.256583530497D0,
     1                   +0.196365054867D0,    +6.121523942807D0,
     2                   +4.523601514852D0/
      DATA C1/+0.228027134939576D0,+0.017201969766646D0,
     1                   +0.230895723235372D0,+0.212768711675140D0,
     2                   -0.000924220294225D0/
      DATA C2/+0.12025169E-12,-0.00196240E-12,-0.04200996E-12
     1                  ,-0.01878819E-12,+0.02718291E-12/
      DATA C3/+5.153876E-21,-1.193948E-21,-0.119395E-21,
     1                   +0.676571E-21,+0.795765E-21/
C DEFINE CONSTANTS COEFFICIENTS OF SINE AND COSINE TERMS
      DATA SCF/0.,-1.2729,+0.2088,-0.2037,+0.1261,+0.0675,
     1                     -0.0497,-0.0342,-0.0261,+0.0214,-0.0149,
     2                     +0.0124,+0.0114,+0.0060,+0.0058,-0.0057,
     3                     -0.0052,+0.0045,+0.0045,-0.0044,-0.0032,
     4                     +0.0028,+0.0026,-0.0026,+0.0025,-0.0021,
     5                     +0.0019,+0.0016,-0.0015,-0.0015,+0.0014,
     6                     -0.0013,-0.0010,+0.0010/
      DATA CCF/+9.2100,+0.5522,-0.0904,+0.0884,+0.0216,
     1                     +0.0183,+0.0113,-0.0093,-0.0066,-0.0050,
     2                     -0.0031,+0.0030,-0.0024,+0.0023,+0.0022,
     3                     +0.0014,-0.0011,+0.0011,-0.0010/
      REAL*8 DMOD,ARGMOD,DTWOPI
      DATA DTWOPI/6.283185307179586D0/
C COMPUTE D = NUMBER OF DAYS ELAPSED FROM 1900 JANUARY 0.5 DAYS
C EPHEMERIS TIME
      D=DJ-2415020.D0
      D2=D*D
      D3=D*D2
C COMPUTE FUNDAMENTAL ARGUMENTS AND REDUCE BY MODULUS 2 PI
C FARG(1) = MEAN ANOMALY - MOON
C FARG(2) = MEAN ANOMALY - SUN
C FARG(3) = MEAN ARGUMENT OF LATITUDE - MOON
C FARG(4) = MEAN ELONGATION OF MOON FROM SUN
C FARG(5) = LONGITUDE OF MEAN ASCENDING NODE - MOON
      DO 1 N=1,5
      ARGMOD=C0(N)+C1(N)*D+DBLE(C2(N)*D2)+DBLE(C3(N)*D3)
    1 FARG(N)=DMOD(ARGMOD,DTWOPI)
C COMPUTE SINES AND COSINES OF FUNDAMENTAL ARGUMENTS AND THE
C COMBINATION OF THE FUNDAMENTAL ARGUMENTS
      S(1) = SIN(FARG(5))
```

```
      C(1) = COS(FARG(5))                                               EQN  112
      S(3) = 2.0*S(1)*C(1)                                              EQN  113
      C(3) = C(1)**2-S(1)**2                                            EQN  114
      SF   = SIN(FARG(3))                                               EQN  115
      CF   = COS(FARG(3))                                               EQN  116
      S(25)= 2.0*SF*CF                                                  EQN  117
      C2F  = CF**2-SF**2                                                EQN  118
      SD   = SIN(FARG(4))                                               EQN  119
      CD   = COS(FARG(4))                                               EQN  120
      S(14)= 2.0*SD*CD                                                  EQN  121
      C2D  = CD**2-SD**2                                                EQN  122
      S(4) = S(25)*C(3)+C2F  *S(3)                                      EQN  123
      C(4) = C2F  *C(3)-S(25)*S(3)                                      EQN  124
      AL   = S(4)*C2D                                                   EQN  125
      AL1  = C(4)*S(14)                                                 EQN  126
      AL2  = S(4)*S(14)                                                 EQN  127
      AL3  = C(4)*C2D                                                   EQN  128
      S(2) = AL -AL1                                                    EQN  129
      C(2) = AL3+AL2                                                    EQN  130
      S(21)= AL +AL1                                                    EQN  131
      C(16)= AL3-AL2                                                    EQN  132
      S(5) = SIN(FARG(2))                                               EQN  133
      CL1  = COS(FARG(2))                                               EQN  134
      S(26)= 2.0*S(5)*CL1                                               EQN  135
      C2L1 = CL1**2-S(5)**2                                             EQN  136
      S(6) = SIN(FARG(1))                                               EQN  137
      CL   = COS(FARG(1))                                               EQN  138
      S(22)= 2.0*S(6)*CL                                                EQN  139
      C2L  = CL**2-S(6)**2                                              EQN  140
      S(8) = S(25)*C(1)+C2F  *S(1)                                      EQN  141
      C(6) = C2F  *C(1)-S(25)*S(1)                                      EQN  142
      BE   = S(2)*CL1                                                   EQN  143
      BE1  = C(2)*S(5)                                                  EQN  144
      BE2  = S(2)*S(5)                                                  EQN  145
      BE3  = C(2)*CL1                                                   EQN  146
      S(7) = BE +BE1                                                    EQN  147
      C(5) = BE3-BE2                                                    EQN  148
      S(10)= BE -BE1                                                    EQN  149
      C(8) = BE3+BE2                                                    EQN  150
      GA   = S(4)*CL                                                    EQN  151
      GA1  = C(4)*S(6)                                                  EQN  152
      GA2  = S(4)*S(6)                                                  EQN  153
      GA3  = C(4)*CL                                                    EQN  154
      S(9) = GA +GA1                                                    EQN  155
      C(7) = GA3-GA2                                                    EQN  156
      S(13)= GA -GA1                                                    EQN  157
      C(10)= GA3+GA2                                                    EQN  158
      S(11)= S(6)*C2D-CL  *S(14)                                        EQN  159
      CT   = CL  *C2D+S(6)*S(14)                                        EQN  160
      S(12)= S(8)*C2D-C(6)*S(14)                                        EQN  161
      C(9) = C(6)*C2D+S(8)*S(14)                                        EQN  162
      DE   = S(1)*CL                                                    EQN  163
      DE1  = C(1)*S(6)                                                  EQN  164
      DE2  = S(1)*S(6)                                                  EQN  165
      DE3  = C(1)*CL                                                    EQN  166
      S(15)= DE +DE1                                                    EQN  167
```

```
      C(11)= DE3-DE2
      S(16)= DE -DE1
      C(12)= DE3+DE2
      S(17)= S(21)*CL -C(16)*S(6)
      C(15)= C(16)*CL +S(21)*S(6)
      S(18)= S(8) *C2L-C(6) *S(22)
      C(13)= C(6) *C2L+S(8) *S(22)
      S(19)= S(22)*C2D-C2L  *S(14)
      EP   = S(8)*CL
      EP1  = C(6)*S(6)
      EP2  = S(8)*S(6)
      EP3  = C(6)*CL
      S(20)= EP +EP1
      C(14)= EP3-EP2
      S(27)= EP -EP1
      C(19)=EP3+EP2
      S(23)= S(2)*CL  +C(2)*S(6)
      C(17)= C(2)*CL  -S(2)*S(6)
      S(24)= S(4)*C2L +C(4)*S(22)
      C(16)= C(4)*C2L -S(4)*S(22)
      S(26)= S(25)*C2D -S(14)*C2F
      S(29)= S(2) *C2L1+C(2) *S(28)
      ZE   = S(1)*CL1
      ZE1  = C(1)*S(5)
      S(33)= ZE-ZE1
      S(30)= ZE+ZE1
      AMU  = S(1)*CT
      AMU1 = C(1)*S(11)
      S(31)= AMU-AMU1
      S(32)= AMU+AMU1
      S(34)= S(22)*C2F-C2L*S(25)
C DEFINE VARIABLE COEFFICIENT OF SINE TERM
      SCF(1)= -17.2327 - .47556468E-6*D
C COMPUTE NUTATION IN LONGITUDE BY SUMMING PRODUCTS OF SINE
C COEFFICIENTS WITH THEIR RESPECTIVE SINE TERMS
C COMPUTE NUTATION IN OBLIQUITY BY SUMMING PRODUCTS OF COSINE
C COEFFICIENTS WITH THEIR RESPECTIVE COSINE TERMS
      DPSI=0
      DE  =0
      DO 3 N=1,19
      DPSI=DPSI+SCF(N)*S(N)
    3 DE  =DE  +CCF(N)*C(N)
      DO 4 N=20,34
    4 DPSI=DPSI+SCF(N)*S(N)
C CONVERT FROM SECONDS OF ARC TO RADIANS
      DPSI=DPSI*.48481368D-05
      DE=DE*.48481368D-05
C COMPUTE TRUE OBLIQUITY OF DATE IN RADIANS
      E=0.40931976D0-6.217959D-9*D-.021441D-15*D2+.100037D-21*D3+DE
C COMPUTE NUTATION IN RIGHT ASCENSION (EQUATION OF THE EQUINOXES)
C IN RADIANS
      EQN=DPSI*DCOS(E)
      RETURN
      END
```


ENTRY
COMPUTE DAYS ELAPSED FROM 1900 JAN 0.5 E.T.
COMPUTE FUNDAMENTAL ARGUMENTS
COMPUTE SINES AND COSINES OF FUNDAMENTAL ARGUMENTS
DEFINE VARIABLE COEFFICIENTS OF SINE TERM
COMPUTE NUTATION IN OBLIQUITY
COMPUTE NUTATION IN LONGITUDE
A


A
CONVERT FROM SECONDS OF ARC TO RADIANS
COMPUTE TRUE OBLIQUITY OF DATE IN RADIANS
COMPUTE NUTATION IN RIGHT ASCENSION IN RADIAS
RETURN

EQUATR

DESCRIPTION

Subroutine EQUATR rotates a vector from mean or true equator and equinox of one epoch to mean or true equator and equinox of another epoch.

This routine invokes subroutine PRECES and NUTATE to generate the requisite rotation matrices. Note that the precession matrix is to precess to the given epoch from the mean equator and equinox of 1950.0; the nutation matrix rotates the vector from mean to true equator and equinox of date. Hence, the last two of the possible four rotation matrices generated must be inverted.

NAME EQUATR

PURPOSE TO ROTATE VECTOR FROM MEAN OR TRUE EQUATOR AND EQUINOX OF ONE EPOCH TO MEAN OR TRUE EQUATOR AND EQUINOX OF ANOTHER EPOCH

CALLING SEQUENCE CALL EQUATR(X,DIN,TIN,Y,DOUT,TOUT)

SYMBOL	TYPE	DESCRIPTION
X (3)	DP	INPUT - INPUT VECTOR
DIN	DP	INPUT - DAYS FROM JAN 0.0 OF REFERENCE YEAR FOR INPUT VECTOR
TIN	L	INPUT - TYPES OF INPUT : .TRUE. = TRUE COORDINATE SYSTEM .FALSE. = MEAN COORDINATE SYSTEM
Y (3)	DP	OUTPUT - OUTPUT VECTOR
DOUT	DP	OUTPUT - DAYS FROM JAN 0.0 OF REFERENCE YEAR FOR OUTPUT VECTOR
TOUT	L	OUTPUT - TYPES OF OUTPUT : .TRUE. = TRUE COORDINATE SYSTEM .FALSE. = MEAN COORDINATE SYSTEM

SUBROUTINES USED NUTATE PRECES

COMMON BLOCKS NONE

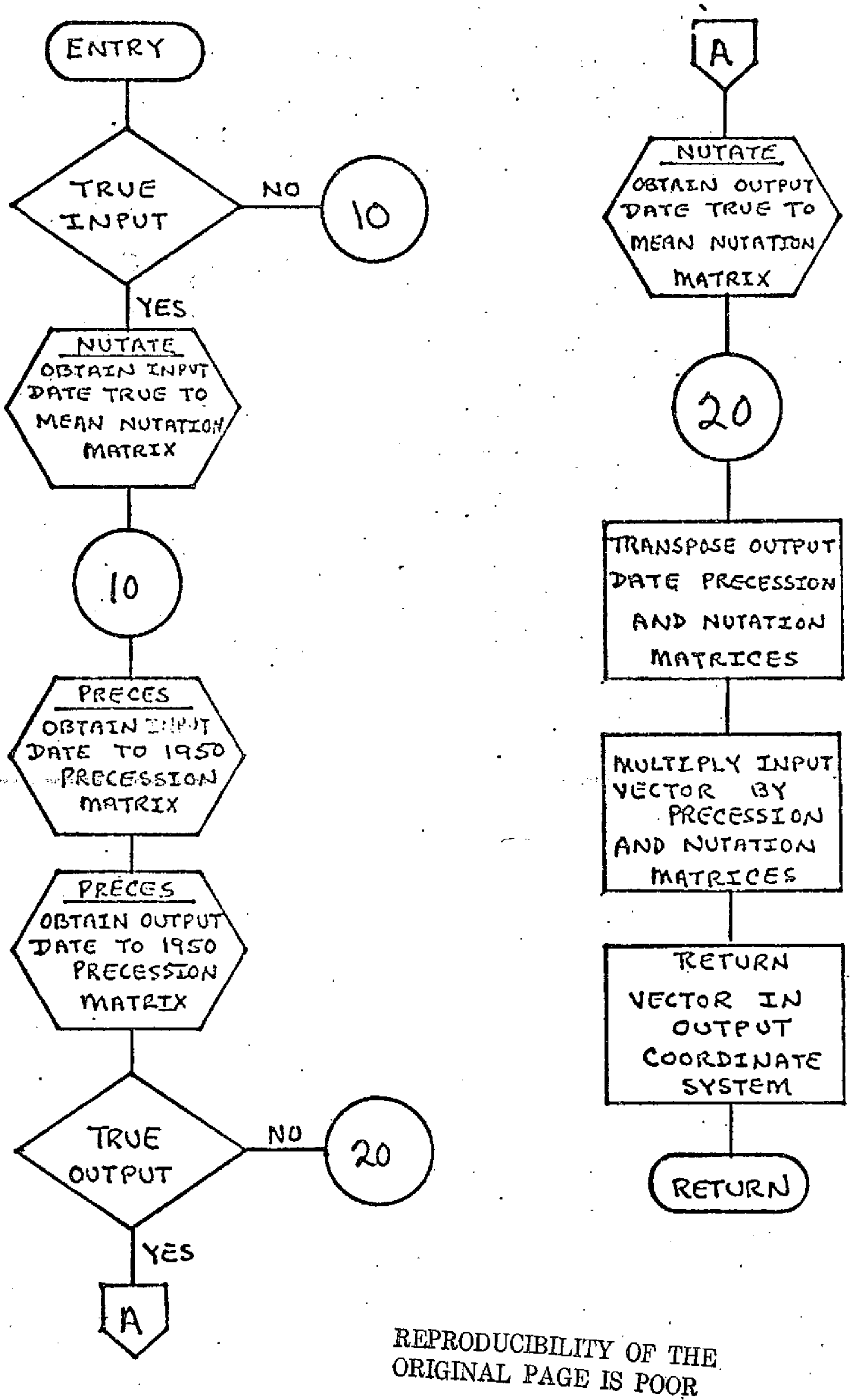
INPUT FILES NONE

OUTPUT FILES NONE

REFERENCES 'GEODYN SYSTEMS DESCRIPTION'
VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE EQUATR(X,DIN,TIN,Y,DOUT,TOUT)
      REAL*8 X(3),Y(3),NP(3,3,4),T(3),DIN,DOUT,TEMP
      LOGICAL TIN,TOUT
      K=2
      IF(.NOT.TIN) GO TO 10
      K=1
C OBTAIN MATRIX TO NUTATE FROM TRUE TO MEAN OF INPUT EPOCH
      CALL NUTATE(DIN,NP(1,1,1))
C OBTAIN MATRIX TO PRECESS FROM INPUT EPOCH TO 1950
 10   CALL PRECES(DIN,NP(1,1,2))
C OBTAIN MATRIX TO PRECESS FROM OUTPUT EPOCH TO 1950
```

```
      CALL PRECES(DULT,NP(1,1,3))
      N=3
      IF(.NOT.TOUT) GO TO 20
      N=4
C OBTAIN MATRIX TO NUTATE FROM TRUE TO MEAN OF OUTPUT EPOCH
      CALL NUTATE(DULT,NP(1,1,4))
C TRANSPOSE OUTPUT EPOCH PRECESSION AND NUTATION MATRICES
 20   DO 30 I=1,3
      Y(I)=X(I)
      DO 30 J=1,3
      DO 30 K=3,N
      TEMP=NP(I,J,K)
      NP(I,J,K)=NP(J,I,K)
 30   NP(J,I,K)=TEMP
C ROTATE INPUT VECTOR TO OBTAIN OUTPUT VECTOR
      DO 60 K=M,N
      DO 50 I=1,3
      T(I)=Y(I)
 50   Y(I)=0.D0
      DO 60 I=1,3
      DO 60 J=1,3
 60   Y(I)=Y(I)+NP(I,J,K)*T(J)
      RETURN
      END
```



REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME ERROR

PURPOSE TO PRINT GEODYN ERROR MESSAGES AND PROVIDE ERROR TERMINATION

CALLING SEQUENCE CALL ERROR(NUMBER,A)

SYMBOL	TYPE	DESCRIPTION
NUMBER	I	ERROR TYPE CODE NUMBER AS FOLLOWS:

NUMBER	ERROR TYPE
1	INVALID INPUT ORBITAL ELEMENTS
2	INVALID ADJUSTED ORBITAL ELEMENTS
3	INVALID OPTION CARD IN INPUT
4	NO TRACKING DATA FOUND IN DATA PERIOD
5	FEWER THAN 6 WEIGHTED OBSERVATIONS LEFT
6	NO DATA CARD TERMINATING INPUT CARDS
7	OTHER ERRORS - NO EXPLANATION PRINTED
8	DUPLICATE STATION IN TRACKING COMPLEMENT
9	REQUEST FOR ADJUSTMENT OF A STATION WHICH IS NOT IN THE TRACKING COMPLEMENT
10	PRINT GEODYN SIGNATURE
11	I/O ERROR IN D.A. READ
12	GEOPOTENTIAL COEFFICIENTS OUT OF RANGE
13 & 14	ILLEGAL SURF CARD
15	INSUFFICIENT DENSITIES ADJUSTED RELATIVE TO THE NUMBER OF CONSTRAINTS

SYMBOL	TYPE	DESCRIPTION
A	DP	NAME OF INVALID OPTION CARD IN A6 FORMAT OR THE STATION NUMBER IN ERROR. NOTE THAT WHEN THIS VARIABLE IS USED AS A STATION NUMBER IT MUST BE AN INTEGER IN THE CALLING PROGRAM AND MUST START ON AN EVEN CORE LOCATION

SUBROUTINES USED NONE

COMMON BLOCKS FERMSG INTBLK

INPUT FILES NONE

OUTPUT FILES PRINTER

```
      SUBROUTINE ERROR(NUMBER,A)
      REAL*8 DEGREE(2)/6HDEGREE,5HORDER/,A
      EQUIVALENCE (ISTA,E)
      COMMON/FERMSG/IMES(26)
      COMMON/INTBLK/DUM(112),STEPSW,DUM2(19)
      LOGICAL STEPSW
      E=A
C SELECT AND PRINT ERROR MESSAGE
      GO TO (10,20,30,40,50,60,100,80,90,110,120,130,140,140,150,160),
     .   NUMBER
C FATAL ERRORS - TERMINATE PROGRAM
   10 WRITE(6,500)
      GO TO 100
   20 WRITE(6,510)
      IF(.NOT.STEPSW) WRITE(6,511)
      IF(STEPSW) WRITE(6,512)
      WRITE(6,513)
      GO TO 100
   30 WRITE(6,560) A
      GO TO 100
   40 WRITE(6,570)
      GO TO 100
   50 WRITE(6,580)
      GO TO 100
   60 WRITE(6,590)
      GO TO 100
   90 WRITE(6,610) ISTA
  100 WRITE(6,99999)
      STOP 9999
C NON-FATAL ERRORS - RETURN
   80 WRITE(6,600) ISTA
  110 WRITE(6,99999)
      RETURN
C FATAL ERRORS - TERMINATE PROGRAM
  120 WRITE(6,620) ISTA,IMES
      GO TO 100
  130 J=NUMBER-11
      WRITE(6,630) DEGREE(J)
      GO TO 100
  140 WRITE (6,640) A
      GO TO 100
  150 WRITE (6,650)
      GO TO 100
  160 WRITE(6,660)
      GO TO 100
  500 FORMAT(1H1,20X,'PROGRAM TERMINATION DUE TO ILLEGAL INPUT'/
     2        1H0,15X,'EXPLANATION: INPUT CARTESIAN EPOCH ELEMENTS ',
     3        'EQUIVALENT TO KEPLERIAN ELEMENTS WITH AN ECCENTRICITY'/
     4        1H ,20X,'GREATER THAN 1. PLEASE CHECK INPUT ELEMENTS AND ',
     5        'EXAMINE INPUT DECKS FOR MISSING OR OUT OF ORDER CARDS.'/)
  510 FORMAT(1H1,20X,'PROGRAM TERMINATION DUE TO INAPPROPRIATE INPUT.'/
     1        1H0,15X,'EXPLANATION: ADJUSTED CARTESIAN EPOCH ELEMENTS ',
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
         'EQUIVALENT TO KEPLERIAN ELEMENTS WITH ECCENTRICITY
     2   1H ,20X,'GREATER THAN 1.'/1H0,15X,'PROBABLE CAUSES:'/)
     3   1H ,20X,'1) POOR FIRST ITERATION ORBIT DUE TO POOR',
 511 FORMAT(1H ,20X,'1) POOR FIRST ITERATION ORBIT DUE TO POOR',
     1   ' STARTING ELEMENTS OR A STEP SIZE TOO LARGE FOR THE'/
     2   1H ,22X,'NUMERICAL ORBIT INTEGRATION IN THE FIXED STEP ',
     3   'MODE.')
 512 FORMAT(1H ,20X,'1) POOR FIRST ITERATION ORBIT DUE TO POOR ',
     1   'STARTING ELEMENTS OR INAPPROPRIATE ERROR BOUNDS.')
 513 FORMAT(1H ,20X,'2) THE EXISTENCE OF WILD DATA POINTS WHICH WERE',
     1   ' NOT EDITED FROM THE SOLUTION.'/)
 560 FORMAT(1H1,20X,'PROGRAM TERMINATION DUE TO ILLEGAL INPUT'/
     1   1H0,15X,'EXPLANATION: THE CARD ',A6,' IN THE OPTION CARD ',
     2   'GROUP IS ILLEGAL. PLEASE CHECK INPUT FOR KEYPUNCH '/
     3   1H ,20X,'ERRORS OR MISSING OR MISPLACED CARDS.'/)
 570 FORMAT(1H1,20X,'PROGRAM TERMINATION DUE TO INSUFFICIENT ',
     1   'OBSERVATIONS'/1H0,15X,'EXPLANATION: THE SETUP AND ',
     2   'OPERATION OF THIS RUN WERE SUCH THAT NO OBSERVATIONS ',
     3   'WERE '/1H ,20X,'AVAILABLE IN THE TIME PERIOD SPECIFIED ',
     4   'FOR THE SATELLITE REQUESTED. PLEASE CHECK YOUR SETUP '/
     5   1H ,20X,'DECK OR TAPE ASSIGNMENTS.'/)
 580 FORMAT(1H1,20X,'PROGRAM TERMINATION DUE TO INSUFFICIENT WEIGHTED',
     1   ' OBSERVATIONS.'/
     2   1H0,15X,'EXPLANATION: INPUT ELEMENTS AND EDITING ',
     3   'CRITERIA WERE SUCH THAT FEWER THAN 8 OBSERVATIONS WERE '/
     4   1H ,20X,'LEFT IN THE SOLUTION.'/)
 590 FORMAT(1H1,20X,'PROGRAM TERMINATION DUE TO MISSING DATA CARD'/
     1  1H0,15X,'EXPLANATION: PROGRAM ENCOUNTERED END OF FILE ON ',
     2  'DATA5 BEFORE READING DATA CARD TERMINATING LAST ARC.'/
     3  1H0,15X,'PLEASE CHECK INPUT DECK FOR MISSING OR MISPLACED ',
     4  'DATA CARD OR MISPLACED END OF FILE'/)
 600 FORMAT(1H1,20X,'ILLEGAL STATION POSITION INPUT'/1H0,15X,
     1  'EXPLANATION: MORE THAN ONE STATION POSITION CARD WAS INPUT ',
     2  'FOR STATION',I5,1H./1H0,15X,'PROGRAM ACTION: THE DUPLICATE ',
     3  'OF THIS STATION HAS BEEN REMOVED FROM THE TRACKING COMPLEMENT'
     4  ,'.'/21X,'THE FIRST STATION ENCOUNTERED WITH THIS NUMBER WAS ',
     5  'USED.'/1H0,15X,'PROGRAM EXECUTION WILL BE CONTINUED.'/)
 610 FORMAT(1H1,20X,'PROGRAM TERMINATION DUE TO INSUFFICIENT ',
     1  'A PRIORI INFORMATION.'/1H0,15X,'EXPLANATION: THE ADJUSTMENT',
     2  ' OF STATION',I5,' HAS BEEN REQUESTED,'/16X,'BUT NO A PRIORI',
     3  ' POSITION IS AVAILABLE FOR THIS STATION.'/)
 620 FORMAT(1H1,20X,'EXECUTION TERMINATING DUE READ ERROR ON RANDOM',
     1  ' ACCESS FILE',I3/1H0,15X,'ERROR BUFFER IS'//15X,Z8,I6,20A4/
     2  16X,4(1X,Z8)/)
 630 FORMAT(1H1,20X,'EXECUTION TERMINATING DUE TO INPUT ERROR'/
     .  1H0,15X,'EXPLANATION: THE SETUP OF THIS RUN IS SUCH THAT THE ',
     .  'ESTIMATION OF'/20X,'GEOPOTENTIAL COEFFICIENTS OF ',A7,
     .  'GREATER THAN USED IN THE'/20X,'GEOPOTENTIAL EXPANSION HAS ',
     .  'BEEN REQUESTED.')
 640 FORMAT (1H1,20X,'PROGRAM TERMINATION DUE TO ILLEGAL INPUT'/
     .  1H0,15X,'EXPLANATION: THE CARD ',A6,' IN THE OPTION CARD ',
     .  'GROUP IS MISSING A CONTINUATION.'/1H ,20X,'PLEASE CHECK ',
     .  'INPUT FOR MISSING OR MISPLACED CARDS.')
 650 FORMAT (1H1,20X,'PROGRAM TERMINATION DUE TO ILLEGAL INPUT'/
     .  1H0,15X,'EXPLANATION: AN ILLEGAL SURFACE DENSITY INCREMENT ',
     .  'SIZE WAS SPECIFIED.'/1H ,20X,'PLEASE CHECK INPUT FOR ',
     .  'KEYPUNCH ERRORS.')
```

```
  660 FORMAT(1H1,20X,'EXECUTION TERMINATING DUE TO INAPPROPRIATE INPUT
       1H0,15X,'EXPLANATION: THE SETUP OF THIS DECK IS SUCH THAT ',
     . 'THE NUMBER OF SURFACE'/21X,'DENSITY CONSTRAINT EQUATIONS ',
     . 'IS GREATER THAN OR EQUAL TO'/21X,
     . 'THE NUMBER OF ADJUSTED SURFACE DENSITIES.'/)
99999 FORMAT(52X,2H**/51X,5(1H*)/50X,2H**,2X,2H**/50X,2H**,2X,2H**,
     .   3X,    /50X,2H**,2X,2H**,2X,2H** /50X,2H**,1X,2H**,2X,3H***,
     . 31X,3H***/50X,4H****,1X,5(1H*),31X,3H***/51X,6(1H*),1X,2H**,30X,
     .   2H**/51X,3H***,4X,2H**,30X,2H**/50X,2H**,5X,2H**,29X,2H**/
     .   49X,2H**,7X,2H**,29X,2H**/48X,2H**,3X,2H**,28X,2H**/47X,2H**,
     .  9X,2H**,2X,4H****,5X,4H****,7X,3H***,1X,3H***,3X,4H****,4X,
     .  3H***,2X,3H***,2X,4H****/46X,2H**,10X,1H*,2X,6(1H*),4X,2H**,
     .   1X,3H***,5X,2H**,1X,4H****,4X,4H****,4X,2H**,3X,3H***,1X,
     .  6(1H*)/41X,3H***,1X,2H**,10X,2H**,1X,3H***,3X,1H*,3X,2H**,3X,
     .   2H**,4X,2H**,3X,3H***,5X,2H**,4X,2H** ,5X,4H****,3X,2H**/
     . 42X,4(1H*),10X,2H**,3X,4H****,4X,2H**,4X,2H**,3X,2H**,4X,2H**,
     .   6X,2H**,3X,2H**,6X,2H**,5X,2H**/43X,4H****,8X,2H**,3X,2H**,6X
     .  2H**,5X,2H**,2X,2H**,5X,2H**,5X,2H**,4X,2H**,5X,2H**,5X,2H**/
     .   43X,5H*****,5X,2H**,3X,3H***,6X,2H**,4X,2H**,3X,2H**,4X,
     .   3H***,4X,2H**,4X,3H*** ,5X,2H**,5X,2H**/42X,2H**,2X,9(1H*),5X
     .  6(1H*),2X,2H**,3X,2H**,4X,2H**,3X,5(1H*),3X,2H**,3X,3H*** ,5X,
     .  2H**,5X,2H**/41X,3H***,4X,6(1H*),7X,4H****,4X,5(1H*),6X,6(1H*),
     .   1X,2H**,4X,4H****,1X,2H**,4X,3H***,4X,4(1H*)/97X,2H**/97X,
     .  2H**/41X,57(1H*)/40X,57(1H*))
      END
```

ESTIM

DESCRIPTION

ESTIM is the Bayesian least squares numerical parameter estimation subroutine designed specifically for the GEODYN system.

ESTIM has four types of calls, each performing a specific task. The tasks performed on each of the ESTIM calls are the following:

1) Initialization.
2) Formation of the estimation matrices.
3) Estimation of single arc parameters and, on the last inner iteration, computation of effects of single arc parameters on adjusted geodetic parameters.
4) Estimation of geopotential parameters.

NAME	ESTIM
ENTRY POINT	PURPOSE
ESTIM1	INITIALIZATION
ESTIM	TO ESTIMATE CORRECTION VECTOR TO STATE VECTOR USING METHOD OF BAYESIAN LEAST SQUARES

CALLING SEQUENCE

```
CALL ESTIM1(SUM1,SUM2,DELTA,STPOS,STPOSO,SIGSTA,
            BBIAS,BIASO,BIASSG,ISTANO,ESTANO,LOC,
            BIASNO,PARNOS,INDXCS,TDENO,TDSIG)
```

SYMBOL	TYPE	DESCRIPTION
SUM1 (1)	DP	INPUT & OUTPUT - NORMAL MATRIX
SUM2 (1)	DP	INPUT & OUTPUT - RIGHT HAND SIDE OF NORMAL EQUATIONS
DELTA (1)	DP	OUTPUT - CORRECTION VECTOR
STPOS (3,1)	DP	INPUT & OUTPUT - CURRENT STATION COORDINATES
STPOSO (3,1)	DP	INPUT - A PRIORI STATION COORDINATES
SIGSTA (3,3,1)	R	INPUT - STATION COORDINATE VARIANCE/COVARIANCE MATRICES
BBIAS (1)	DP	INPUT & OUTPUT - CURRENT DRAG, SOLAR RADIATION, BIAS, GEOPOTENTIAL & SURFACE DENSITY VALUES
BIASO (1)	DP	INPUT - A PRIORI DRAG, SOLAR RADIATION, BIAS, AND GEOPOTENTIAL VALUES
BIASSG (1)	DP	INPUT - DRAG, SOLAR RADIATION, BIAS, GEOPOTENTIAL AND SURFACE DENSITY VALUES
ISTANO (1)	I*2	INPUT - STATION NUMBERS
ESTANO (1)	I*2	INPUT - MASTER STATION LOCATION INDICES
LOC (1)	I*2	SCRATCH
BIASNO (1)	I*2	INPUT - DRAG, SOLAR RADIATION, BIAS, GEOPOTENTIAL AND SURFACE DENSITY LOCATION INDICES

PARNOS (1)	I*2	INPUT - PARAMETER NUMBERS
INDXCS (3,1)	I*2	INPUT - INDICES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
TDENO (1)	DP	INPUT - APRIORI VALUES OF SURFACE DENSITIES
TDSIG (1)	DP	INPUT - APRIORI SIGMAS OF SURFACE DENSITIES

CALLING SEQUENCE CALL ESTIM(ITYPE,SIGMA,RESID,MPARTL)

SYMBOL	TYPE	DESCRIPTION
ITYPE	I	INPUT - INDICATES TYPE OF OPERATION TO BE PERFORMED
SIGMA	DP	INPUT - MEASUREMENT STANDARD DEVIATION
RESID	DP	INPUT - MEASUREMENT RESIDUAL
MPARTL (1)	DP	INPUT - PARTIALS OF MEASUREMENTS WITH RESPECT TO UNKNOWNS

SUBROUTINES USED CLEAR DMTWRT SYMINV NUMBR2 NUMLOC

COMMON BLOCKS APARAM CELEM CONOUT CPARAM INITBK PRIORI

INPUT FILES NONE

OUTPUT FILES NONE

REFERENCES 'GEODYN SYSTEMS DESCRIPTION'
VOLUME 1 - GEODYN DOCUMENTATION

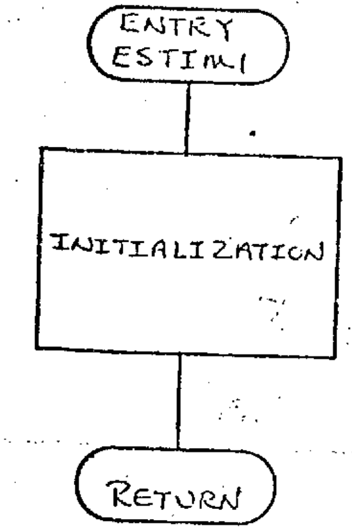
```
      SUBROUTINE ESTIM1(SUM1,SUM2,DELTA,STPOS,STPOSD,SIGSTA,BBIAS,BIASE,ESTI  98
     .   BIASSG,ISTANO,ESTANO,LOC,BIASNO,PARNOS,INDXCS,TDENO,TDSIG)       ESTI  99
      IMPLICIT REAL*8 (A-H,O-Z)                                          ESTI 100
      LOGICAL LINNER,CMPGPR,BMAT                                         ESTI 101
      INTEGER PARMAX,ESTSTA,BMATNO,SIMDAT                                ESTI 102
      INTEGER*2 ISTANO,ESTANO,LOC,BIASNO,PARNOS,INDXCS                   ESTI 103
      REAL SIGSTA,RMSTOT,VARCOV                                          ESTI 104
      DOUBLE PRECISION MPARTL                                            ESTI 105
      DIMENSION SUM1(1),SUM2(1),STPOS(3,1),STPOSD(3,1),SIGSTA(3,3,1),    ESTI 106
     .   BBIAS(1),BIASE(1),BIASSG(1),ISTANO(1),ESTANO(1),LOC(1),         ESTI 107
     .   BIASNO(1),PARNOS(1),DELTA(1),MPARTL(1),INDXCS(3,1),TDENO(1),    ESTI 108
     .   TDSIG(1)                                                        ESTI 109
      COMMON/APARAM/INPAR,INPARI,NBIAS,ESTSTA,NSAT,NGPARC,NDPECI,NPARAM,ESTI 110
     .   NEBIAS,PARMAX                                                   ESTI 111
```

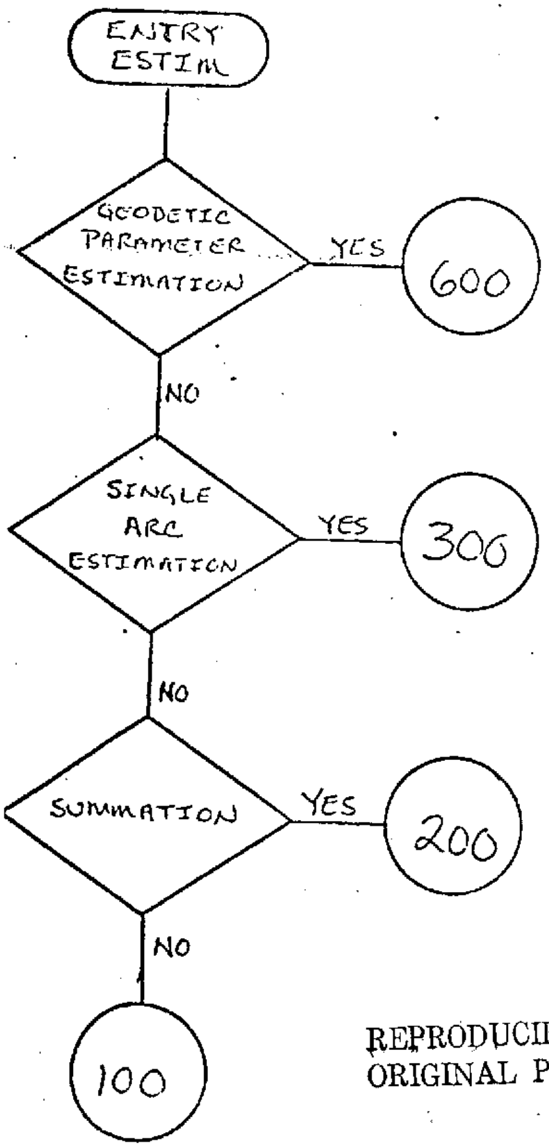
```
      COMMON/CELEM/ELEMST(6,2),ORBELA(14),RMSTOT
      COMMON/CONOUT/IG1(8),NEONMX,IG2(10)
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,NBIAS,NGPC1,NGPC2,
     .   NGPCOM,NCSEST,CMPGPR,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,
     .   INNRSW,NCONST,NOCONS
      COMMON/INITBK/IEPYMD(45),BMATNO,SIMDAT(11)
      COMMON/PRIORI/ELEM0(6,2),VARCOV(6,6,2),TITLE(30),DRAG(18)
      DATA NOFILE/0/
      INDXNO(I)=NDIM*(I-1)-(I*(I-1))/2
      RETURN
      ENTRY ESTIM(ITYPE,SIGMA,RESID,MPARTL)
C INITIALIZE
      GO TO (100,200,300,600),ITYPE
  100 PARMAX=INPARI
      IF(NBIAS.GT.0) PARMAX=NEONMX+2
      ISTART=INPARI+1
      I1=LIM1
      CALL CLEAR(SUM2,2,I1)
      I1=INDXNO(I1)+NDIM
      CALL CLEAR(SUM1,2,I1)
      LINNER=LIM2.GT.0.AND.(NMAST.GT.0.OR.NGPCOM.GT.0.OR.NTIDEN.GT.0)
      BMAT=.FALSE.
      IF(.NOT.LINNER) RETURN
      BMAT=BMATNO.GT.0
      IF(NMAST.GT.0) PARMAX=NEONMX+8
      IF(PARMAX.EQ.INPARI) PARMAX=NEONMX
      NSTART=NDIM-3*NMAST+1-NGPCOM-NTIDEN
C FORM NORMAL MATRIX
      RETURN
  200 WT=1.D0/(SIGMA*SIGMA)
C SINGLE ARC FORCE MODEL PARAMETERS
      IST=0
      DO 240 I=1,INPARI
      T=MPARTL(I)*WT
      SUM2(I)=SUM2(I)+T*RESID
      DO 235 J=I,PARMAX
      I1=PARNOS(J)
      IF(I1.EQ.0) GO TO 235
      I2=IST+I1
      SUM1(I2)=SUM1(I2)+T*MPARTL(J)
  235 CONTINUE
  240 IST=IST+NDIM-I
C BIASES AND GEODETIC PARAMETERS
      IF(PARMAX.EQ.INPARI) RETURN
      DO 250 I=ISTART,PARMAX
      I1=PARNOS(I)
      IF(I1.EQ.0) GO TO 250
      T=MPARTL(I)*WT
      SUM2(I1)=SUM2(I1)+T*RESID
      DO 245 J=I,PARMAX
      I2=PARNOS(J)
      IF(I2.EQ.0) GO TO 245
      IF(I2.LT.I1) GO TO 242
      NCOLM1=I1-1
      IST=NCOLM1*NDIM-(I1*NCOLM1)/2+I2
      GO TO 244
```

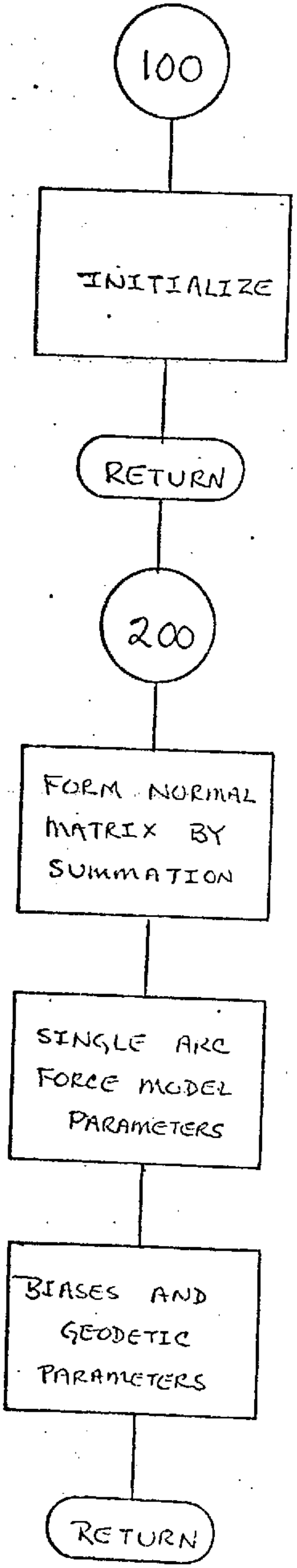
```
  242 NCOLM1=I2-1                                                        ESTI 168
      IST=NCOLM1*NDIM-(I2*NCOLM1)/2 + I1                                 ESTI 169
  244 SUM1(IST)=SUM1(IST)+T*MPARTL(J)                                    ESTI 170
  245 CONTINUE                                                           ESTI 171
  250 CONTINUE                                                           ESTI 172
C ADD SINGLE ARC PARAMETER A PRIORI                                      ESTI 173
      RETURN                                                             ESTI 174
C ELEMENTS                                                               ESTI 175
  300 IST=0                                                              ESTI 176
      DO 310 L=1,NSAT                                                    ESTI 177
      L1=6*(L-1)                                                         ESTI 178
      DO 310 I=1,6                                                       ESTI 179
      I1=L1+I                                                            ESTI 180
      DO 305 J=1,6                                                       ESTI 181
  305 SUM2(I1)=SUM2(I1)+VARCOV(I,J,L)*(ELEMO(J,L)-ELEMST(J,L))           ESTI 182
      DO 308 J=I,6                                                       ESTI 183
      I1=IST+J+L1                                                        ESTI 184
  308 SUM1(I1)=SUM1(I1)+VARCOV(J,I,L)                                    ESTI 185
  310 IST=IST+NDIM-I-L1                                                  ESTI 186
C DRAG, SOLAR RADIATION, AND BIASES                                      ESTI 187
      NBSGP=MBIAS+NGPARC                                                 ESTI 188
      DO 315 I=1,NBSGP                                                   ESTI 189
      I1=BIASNO(I)                                                       ESTI 190
      IF(I1.EQ.0) GO TO 315                                              ESTI 191
      T=1.0D0                                                            ESTI 192
      IF(I.LE.MBIAS) T=1.0D0/BIASSG(I)**2                                ESTI 193
      SUM2(I1)=SUM2(I1)+T*(BIASO(I)-BBIAS(I))                            ESTI 194
      I1=INDXNO(I1)+I1                                                   ESTI 195
      SUM1(I1)=SUM1(I1)+T                                                ESTI 196
  315 CONTINUE                                                           ESTI 197
C ON OPTION, WRITE NORMAL MATRIX                                         ESTI 198
      IF(BMAT) CALL BMTWRT(SUM1,SUM2,INDXCS,BIASNO(NGPC1),BIASO(NGPC1),  ESTI 199
     .    BIASSG(NGPC1),ESTAND,ISTAND,STPOS,NOFILE)                      ESTI 200
C INVERT NORMAL MATRIX OF SINGLE ARC PARAMETERS                          ESTI 201
      CALL SYMINV(SUM1,NDIM,NPARAM,DELTA)                                ESTI 202
C ON LAST INNER ITERATION COMPUTE EFFECTS OF ARC PARAMETERS ON           ESTI 203
C GEODETIC PARAMETERS                                                    ESTI 204
      IF(.NOT.LINNER) GO TO 525                                          ESTI 205
      I1=INDXNO(NSTART)                                                  ESTI 206
      DO 500 I=NSTART,NDIM                                               ESTI 207
      L1=0                                                               ESTI 208
      DO 450 L=1,NPARAM                                                  ESTI 209
      DELTA(L)=0.0D0                                                     ESTI 210
      IST=0                                                              ESTI 211
      DO 400 J=1,L                                                       ESTI 212
      J1=IST+L                                                           ESTI 213
      J2=IST+I                                                           ESTI 214
      DELTA(L)=DELTA(L)+SUM1(J1)*SUM1(J2)                                ESTI 215
  400 IST=IST+NDIM-J                                                     ESTI 216
      IF(L.EQ.NPARAM) GO TO 425                                          ESTI 217
      LP1=L+1                                                            ESTI 218
      DO 410 J=LP1,NPARAM                                                ESTI 219
      J1=L1+J                                                            ESTI 220
      J2=IST+I                                                           ESTI 221
C SUBTRACT OUT SINGLE ARC EFFECTS ON GEODETIC PARAMETERS                 ESTI 222
      DELTA(L)=DELTA(L)+SUM1(J1)*SUM1(J2)                                ESTI 223
```

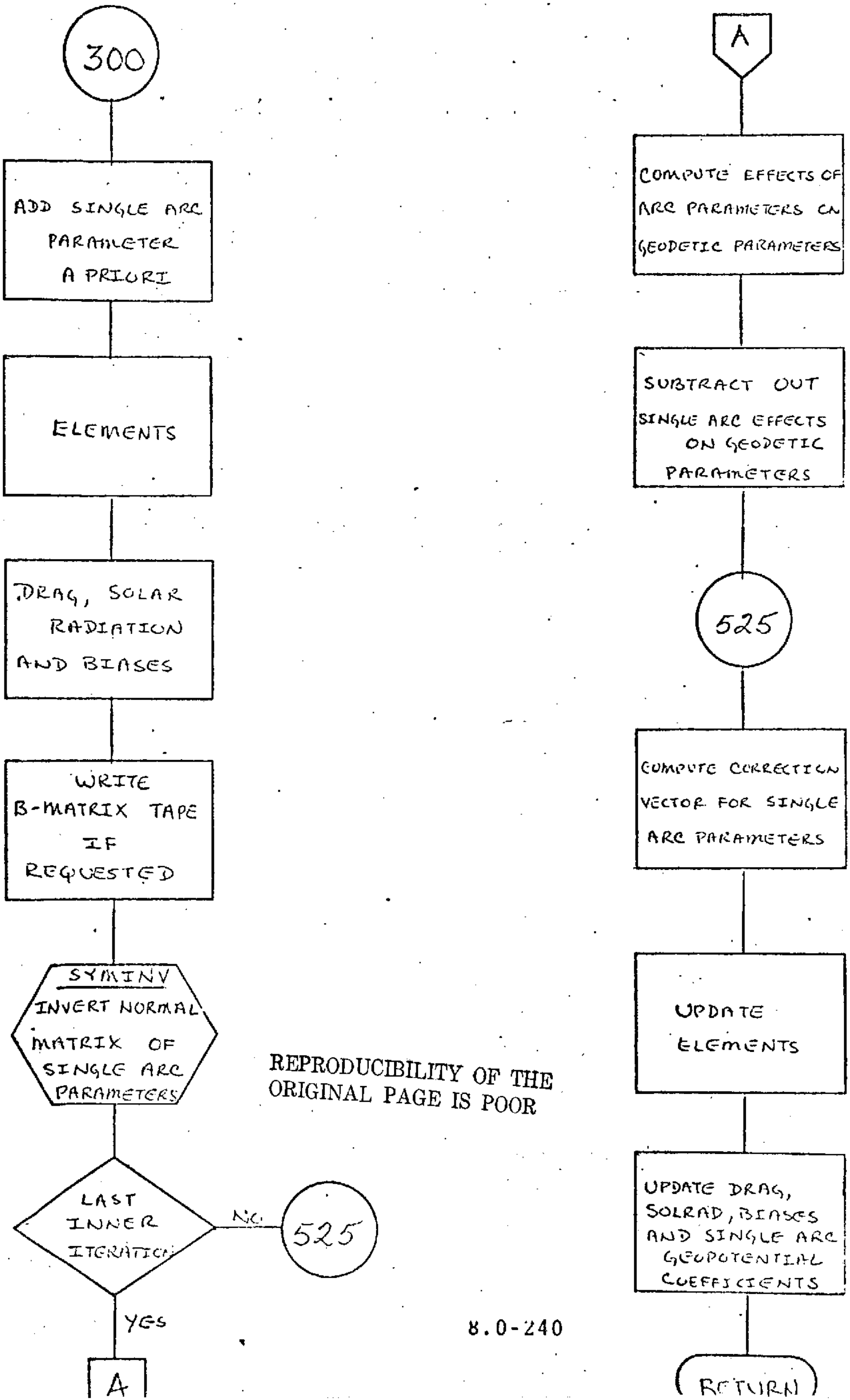
```
  410 IST=IST+NDIM-J                                                    ESTI 224
  425 SUM2(I)=SUM2(I)-SUM2(L)*DELTA(L)                                  ESTI 225
      DO 440 K=I,NDIM                                                   ESTI 226
      K1=I1+K                                                           ESTI 227
      K2=L1+K                                                           ESTI 228
  440 SUM1(K1)=SUM1(K1)-DELTA(L)*SUM1(K2)                               ESTI 229
  450 L1=L1+NDIM-L                                                      ESTI 230
      J1=I                                                              ESTI 231
      DO 495 J=1,NPARAM                                                 ESTI 232
      SUM1(J1)=DELTA(J)                                                 ESTI 233
  495 J1=J1+NDIM-J                                                      ESTI 234
  500 I1=I1+NDIM-I                                                      ESTI 235
C COMPUTE CORRECTION VECTOR FOR SINGLE ARC PARAMETERS                   ESTI 236
  525 IST=0                                                             ESTI 237
      L=6*NSAT                                                          ESTI 238
      DO 550 I=1,NPARAM                                                 ESTI 239
      DELTA(I)=0.0D0                                                    ESTI 240
      I1=I                                                              ESTI 241
      DO 540 J=1,I                                                      ESTI 242
      DELTA(I)=DELTA(I)+SUM1(I1)*SUM2(J)                                ESTI 243
  540 I1=I1+NDIM-J                                                      ESTI 244
      IF(I.EQ.NPARAM) GO TO 548                                         ESTI 245
      IP1=I+1                                                           ESTI 246
      DO 545 J=IP1,NPARAM                                               ESTI 247
      I1=IST+J                                                          ESTI 248
C UPDATE ELEMETS                                                        ESTI 249
  545 DELTA(I)=DELTA(I)+SUM1(I1)*SUM2(J)                                ESTI 250
  548 IF(I.LF.L) ELEMST(I,1)=ELEMST(I,1)+DELTA(I)                       ESTI 251
  550 IST=IST+NDIM-I                                                    ESTI 252
C UPDATE DRAG, SOLAR RADIATION, AND BIASES                              ESTI 253
      IF(NPARAM.EQ.L) RETURN                                            ESTI 254
      DO 560 I=1,NBSGP                                                  ESTI 255
      I1=BIASNO(I)                                                      ESTI 256
  560 IF(I1.NE.0) BBIAS(I)=BBIAS(I)+DELTA(I1)                           ESTI 257
C ADD GEOPOTENTIAL COEFFICICENT A PRIORI                                ESTI 258
      RETURN                                                            ESTI 259
  600 IF(NGPCOM.LE.0) GO TO 606                                         ESTI 260
      T=1.0D0                                                           ESTI 261
      DO 605 I=NGPC1,NGPC2                                              ESTI 262
      I1=BIASNO(I)                                                      ESTI 263
      IF(I1.EQ.0) GO TO 605                                             ESTI 264
      SUM2(I1)=SUM2(I1)+ BIASO(I)-BBIAS(I)                              ESTI 265
      I1=INDXNO(I1)+I1                                                  ESTI 266
      SUM1(I1)=SUM1(I1)+T                                               ESTI 267
  605 CONTINUE                                                          ESTI 268
C ADD STATION POSITION A PRIORI                                         ESTI 269
  606 IF(NTIDEN.LE.0) GO TO 750                                         ESTI 270
      L1=NSTART+NGPCOM-1                                                ESTI 271
      DO 725 I=1,NTIDEN                                                 ESTI 272
      T=1.0D0/TDSIG(I)**2                                               ESTI 273
      I1=L1+I                                                           ESTI 274
      SUM2(I1)=SUM2(I1)+T*(TDENO(I)-BBIAS(I+NGPC2))                     ESTI 275
      I1=INDXNO(I1)+I1                                                  ESTI 276
  725 SUM1(I1)=SUM1(I1)+T                                               ESTI 277
  750 IF(NMAST.EQ.0) GO TO 630                                          ESTI 278
      I1=NSTART+NGPCOM+NTIDEN                                           ESTI 279
```

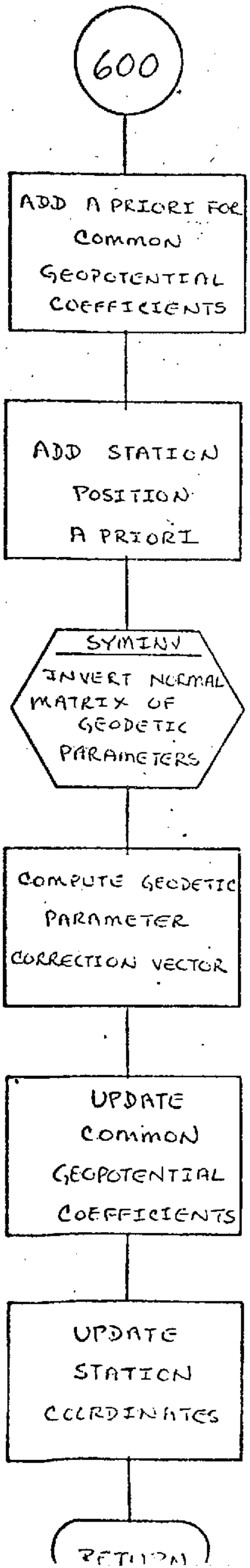
```
      IST=INDXNO(I1)
      L=0
      DO 620 I=1,NMAST
  608 L=L+1
      IF(NUMBR2(L,ESTANO,NSTEST).EQ.0) GO TO 608
      DO 620 J=1,3
      DO 610 K=1,3
  610 SUM2(I1)=SUM2(I1)+SIGSTA(J,K,L)*(STPOS0(K,L)-STPOS(K,L))
      I2=IST+I1
      DO 615 K=J,3
      SUM1(I2)=SUM1(I2)+SIGSTA(K,J,L)
  615 I2=I2+1
      IST=IST+NDIM-I1
  620 I1=I1+1
  630 IST=INDXNO(NSTART)
      I1=IST+NSTART
C INVERT NORMAL MATRIX OF GEODETIC PARAMETERS
      I2=3*NMAST+NGPCOM+NTIDEN
      CALL SYMINV(SUM1(I1),I2,I2,DELTA(NSTART))
      I2=IST
      ISTA=0
      DO 650 I=NSTART,NDIM
      DELTA(I)=0.0D0
C COMPUTE GEODETIC PARAMETER CORRECTION VECTOR
      I1=I2+I
      DO 640 J=NSTART,I
      DELTA(I)=DELTA(I)+SUM1(I1)*SUM2(J)
  640 I1=I1+NDIM-J
      IF(I.EQ.NDIM) GO TO 648
      IP1=I+1
      DO 645 J=IP1,NDIM
      I1=IST+J
  645 DELTA(I)=DELTA(I)+SUM1(I1)*SUM2(J)
C UPDATE GEOPOTENTIAL COEFFICIENTS
  648 I1=I-NSTART+1-NGPCOM-NTIDEN
      IF(I1.GT.0) GO TO 649
      I1=NGPC1+I-NSTART
      BBIAS(I1)=BBIAS(I1)+DELTA(I)
C UPDATE STATION COORDINATES
      GO TO 650
  649 IF(MOD(I1,3).NE.1) GO TO 40
      J1=0
   30 ISTA=ISTA+1
      NOCON=NUMLOC(ISTA,ESTANO,NSTEST,LOC)
      IF(NOCON.EQ.0) GO TO 30
   40 J1=J1+1
      DO 50 J=1,NOCON
      J2=3*(LOC(J)-1)+J1
   50 STPOS(J2,1)=STPOS(J2,1)+DELTA(I)
  650 IST=IST+NDIM-I
      RETURN
      END
```


ENTRY ESTIM1
INITIALIZATION
RETURN


ENTRY ESTIM
GEODETIC PARAMETER ESTIMATION
YES
600
NO
SINGLE ARC ESTIMATION
YES
306
NO
SUMMATION
YES
200
NO
100


100
INITIALIZE
RETURN
200
FORM NORMAL MATRIX BY SUMMATION
SINGLE ARC FORCE MODEL PARAMETERS
BIASES AND GEODETIC PARAMETERS
RETURN


300
ADD SINGLE ARC PARAMETER A PRIORI
ELEMENTS
DRAG, SOLAR RADIATION AND BIASES
WRITE B-MATRIX TAPE IF REQUESTED
SYMINV
INVERT NORMAL MATRIX OF SINGLE ARC PARAMETERS
LAST INNER ITERATION
NO
525
YES
A
A
COMPUTE EFFECTS OF ARC PARAMETERS ON GEODETIC PARAMETERS
SUBTRACT OUT SINGLE ARC EFFECTS ON GEODETIC PARAMETERS
525
COMPUTE CORRECTION VECTOR FOR SINGLE ARC PARAMETERS
UPDATE ELEMENTS
UPDATE DRAG, SOLRAD, BIASES AND SINGLE ARC GEOPOTENTIAL COEFFICIENTS
RETURN


600
ADD A PRIORI FOR COMMON GEOPOTENTIAL COEFFICIENTS
ADD STATION POSITION A PRIORI
SYMINV
INVERT NORMAL MATRIX OF GEODETIC PARAMETERS
COMPUTE GEODETIC PARAMETER CORRECTION VECTOR
UPDATE COMMON GEOPOTENTIAL COEFFICIENTS
UPDATE STATION COORDINATES
RETURN

DESCRIPTION

F evaluates the total acceleration of the satellite due to all of the forces present in the model. F is primarily executive in nature, calling subroutines for all effects except solar radiation pressure. F also computes the partial derivatives of the acceleration due to radiation pressure with respect to the satellite reflectivity coefficient.

The satellite acceleration vector is computed in an instantaneous coordinate system, referred to the true equator and equinox at the integration time. REFCOR is called to transform the position, and velocity if required, from the reference to the instantaneous coordinate system. After the evaluation of the accelerations, REFCOR is again called to transform the acceleration vector to the inertial reference system for integration.

NAME F

ENTRY POINT	PURPOSE
F1	INITIALIZATION
F	TO EVALUATE SATELLITE ACCELERATION VECTOR AND FORCE MODEL PARTIAL DERIVATIVES

CALLING SEQUENCE CALL F1(GRPAR)

SYMBOL	TYPE	DESCRIPTION
GRPAR (3,1)	DP	OUTPUT - FORCE MODEL PARTIAL VECTOR

CALLING SEQUENCE CALL F(T,S,FCT,FCALL)

SYMBOL	TYPE	DESCRIPTION
T	DP	INPUT - TIME IN SECONDS
S (6)	DP	INPUT - SATELLITE STATE VECTOR
FCT (3)	DP	OUTPUT - SATELLITE ACCELERATION VECTOR
FCALL	L	INPUT - =.TRUE. NO ACCELERATIONS NECESSARY =.FALSE. BOTH ACCELERATIONS & PARTIALS NECESSARY

SUBROUTINES USED REFCOR EPHEM EGRAV SURDEN DRAG
SUNGRV TICAL DOTPRD

COMMON BLOCKS CEPHEM CTIME FMODEL INTBLK MOONGR
XYZ

INPUT FILES NONE

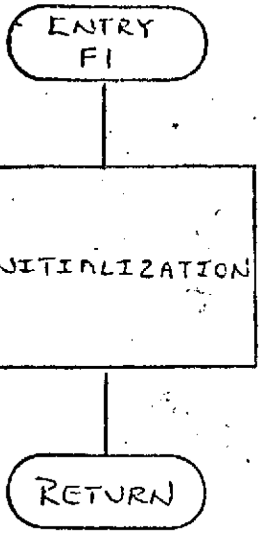
OUTPUT FILES NONE

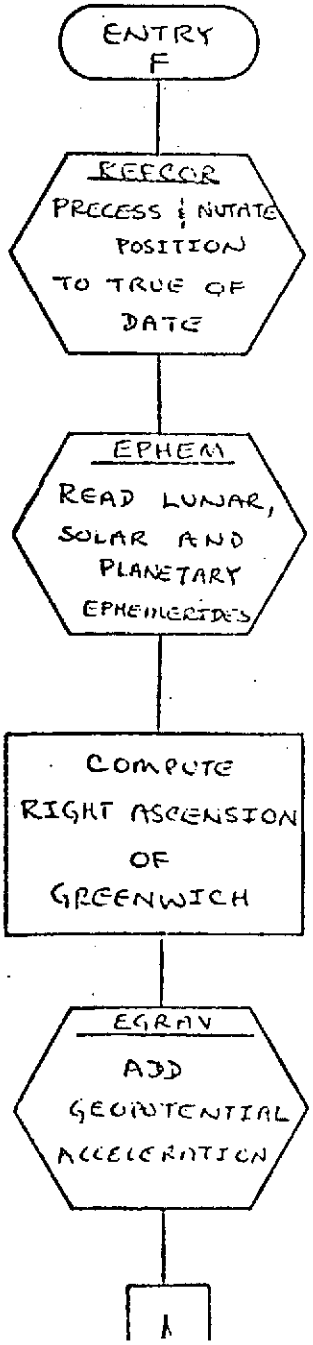
REFERENCES ' GEODYN SYSTEMS DESCRIPTION '
VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE F1(GRPAR)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL FCALL,IOREFT
      INTEGER ADDR,SRAD
      DOUBLE PRECISION MOODY,LOVE,MODEL
      DIMENSION GRPAR(3,1),S(6),P(3),FCT(3)
```

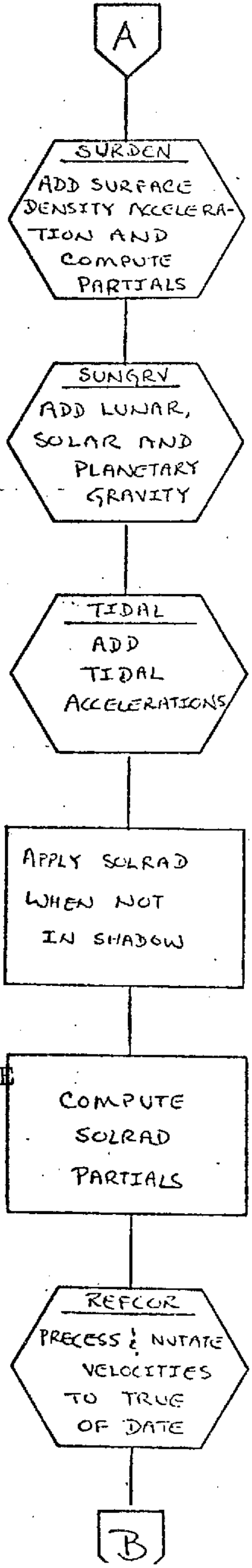
```
      COMMON/CEPHEM/VARD(24),EQ,GEPHEM(643)
      COMMON/CTIME/DATALP(2),DAY0,DSTART,DAY2,DORB1(6),IY
      COMMON/FMODEL/INDEX1,INDEX2,INDEX3,INDEX4,CS(30,33),MODEL(8)
      COMMON/INTBLK/THDOT1,THDOT2,THDT2S,GM(2),AESQ,FLAT(3),GM3(6),B(6),
     .   APGM(2),APLM(4),THETG0,MBODY(31),ADDR(4),SRAD(2),LOVE(3),
     .   TOREFT,NBODY
      COMMON/MOONGR/DPXUV(6),RHOM(3,6),RHCSQ(12)
      COMMON/XYZ/ELEM(6),R,RSQ,ISAT,IFORCE(2)
      RETURN
      ENTRY F(T,S,FCT,FCALL)
      IF(FCALL.AND.T.EQ.TPREV) RETURN
      TPREV=T
      DO 100 I=1,6
  100 ELEM(I)=S(I)
      DAY2=T/8.64D4
C PRECESS AND NUTATE POSITION TO TRUE OF DATE
      CALL REFCOR(DAY2,.FALSE.,ELEM)
C READ LUNAR, SOLAR, PLANETARY EPHEMERIDES
      CALL EPHEM(DAY2,.FALSE.)
      DAY=DINT(DAY2)
      DDAY=DAY2-DAY
C COMPUTE RIGHT ASCENSION OF GREENWICH
      THETG=THETG0+THDOT1*DAY+THDOT2*DDAY+EQ
      INDEX2=INDEX1-1
      IF(FCALL) INDEX2=INDEX4
C ADD GEOPOTENTIAL ACCELERATION
      CALL EGRAV(THETG,EASAT,FCT)
C ADD SURFACE DENSITY ACCELERATION AND COMPUTE PARTIALS
      CALL SURDEN(FCT,THETG)
C ADD LUNAR, SOLAR, PLANETARY GRAVITY
      CALL SUNGRV(FCT)
C ADD TIDAL ACCELERATIONS
      IF(.NOT.FCALL) CALL TIDAL(FCT)
      IF(APGM(ISAT).EQ.0.0D0) GO TO 700
      INDX=SRAD(ISAT)-IFORCE(ISAT)
      IF(APGM(ISAT).LT.0.0D0) GO TO 650
      IF(DPXUV(2).GT.0.0D0) GO TO 500
      DO 450 I=1,3
  450 P(I)=ELEM(I)-DPXUV(2)*VARD(I+4)
      PMAG=DOTPRD(P,P)
      IF(PMAG.LT.AESQ) GO TO 650
  500 IF(FCALL) GO TO 610
C APPLY SOLAR RADIATION WHEN NOT IN SHADOW
      DO 600 I=1,3
  600 FCT(I)=FCT(I)-APGM(ISAT)*VARD(I+4)
  610 IF (INDX.LE.0) GO TO 700
C COMPUTE SOLAR RADIATION PARTIALS
      DO 625 I=1,3
  625 GRPAR(I,INDX)=-APLM(ISAT)*VARD(I+4)
      GO TO 700
  650 IF (INDX.LE.0) GO TO 700
      DO 660 I=1,3
  660 GRPAR(I,INDX)=0.0D0
  700 IF(B(ISAT).EQ.0.0D0) GO TO 800
C PRECESS AND NUTATE VELOCITIES TO TRUE OF DATE
      CALL REFCOR(DAY2,.FALSE.,ELEM(4))
```

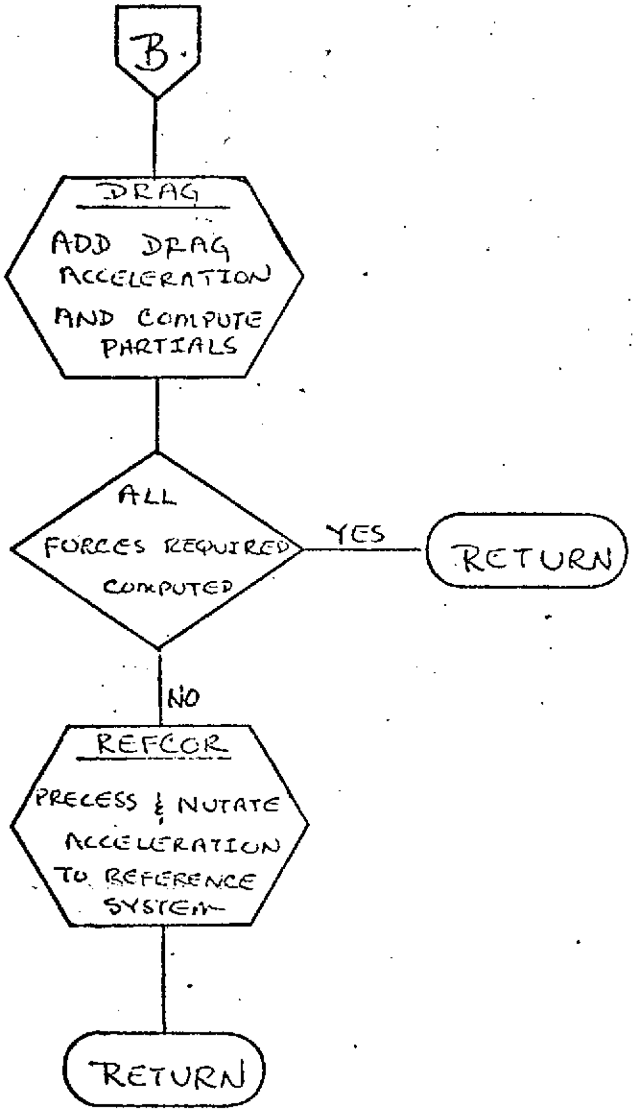
```
      DT=DAY2-DAY0
C ADD DRAG ACCELERATION AND COMPUTE PARTIALS
      CALL DRAG(RASAT,FCT,DT)
  800 IF(FCALL) RETURN
C PRECESS AND NUTATE ACCELERATION TO REFERENCE SYSTEM
      CALL REFCOR(DAY2,.TRUE.,FCT)
      RETURN
      END
```





REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR





REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

FLUXM

DESCRIPTION

FLUXM stores in half-word integers ten times the daily mean of the eight three-hourly geomagnetic indices, K_p.

FLUX tables must be updated using the values obtained from bulletins printed by ESSA., Boulder, Colorado. An example of the bulletin is given on the following page.

GEOMAGNETIC ACTIVITY INDICES

MAY 1972

DAY		K_p THREE-HOUR RANGE INDICES								SUM	C_i	C_p	A_p
		1	2	3	4	5	6	7	8				
1	D	4-	4-	4	3-	3-	3	3+	3-	26-	1.1	1.0	18
2	D	3-	3+	4+	4	3	2+	3-	3+	26-	1.1	1.0	18
3		2	2-	2-	2	3	2	1+	1	15-	0.3	0.4	7
4	Q	1+	2	2-	1	2	2-	1+	0+	11+	0.1	0.2	5
5	Q	2-	2-	1-	1+	1-	1+	1+	3	12-	0.2	0.3	6
6		2	1+	3-	2	3	2+	3-	2-	18-	0.6	0.5	9
7	QQ	2	1-	0	0+	0+	1-	1-	1+	6	0.0	0.1	3
8	QQ	1-	1-	0+	0+	1	1	2+	1-	7	0.1	0.1	4
9		1+	2	4-	3	3-	2+	3+	4-	22	0.9	0.8	14
10		2	2	3-	2+	2	2	3	2	18	0.5	0.5	9
11		1-	2+	1+	2	2	2+	2	2+	15	0.3	0.4	7
12		2	3	3	3-	2+	1+	2-	1+	17+	0.5	0.5	9
13		3	1+	2-	0+	1-	2+	2+	3	15-	0.3	0.4	8
14		2+	2-	2	2-	2	2-	2-	2-	15-	0.3	0.3	7
15	D	1+	0+	1	2-	1	2	8-	6+	21+	1.4	1.4	33
16	D	6	4-	2+	3-	2-	1+	1+	1+	20+	1.1	1.0	18
17		3	2+	3-	1+	2+	3-	1+	2	18-	0.6	0.5	9
18		3+	3+	3-	3-	2-	1	1	1	17-	0.6	0.5	10
19	QQ	1-	1	1-	1-	0+	0+	1	2	7-	0.2	0.1	4
20	QQ	2	1	1-	2-	1-	0+	0+	2-	8+	0.2	0.1	4
21	QQ	1+	1	1-	1-	0+	1	2-	1	8-	0.0	0.1	4
22	Q	2	2+	1+	1-	1-	1	2-	1+	11	0.2	0.2	5
23		2-	1-	0+	0+	1+	3+	2+	3-	13-	0.4	0.4	7
24	Q	2+	2+	1+	0+	1-	1	1-	0+	9	0.2	0.2	5
25	Q	1	1+	2-	1	1	1	2-	2+	11	0.2	0.2	5
26		3-	2	1	1+	1	2+	3-	3	16	0.3	0.5	8
27		2	1+	0+	1-	1+	3-	3	3	14+	0.4	0.4	8
28	D	3	2	3-	3+	4	5+	4+	2-	26+	1.2	1.1	22
29		2+	3-	2	3	2+	2+	2-	2+	19-	0.6	0.5	10
30		3	3	1+	2-	3+	4-	4-	3-	22+	0.8	0.8	14
31		3-	4-	3	3-	2+	2+	1+	3	21	0.8	0.7	12
										MEAN	0.50	0.49	10

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME	FLUXM
PURPOSE	STORES MAGNETIC FLUX INDICES, VALUES ARE DAILY MEANS OF KP TIMES 10
CALLING SEQUENCE	CALL FLUXM(NARCS)
SYMBOL TYPE	DESCRIPTION
NARCS I	OUTPUT - NUMBER OF ARCS
SUBROUTINES USED	FLUXS
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE
REFERENCES	'SOLAR GEOPHYSICAL DATA, PART 1' E.S.S.A., BOULDER, COLO. 'GEODYN SYSTEMS DESCRIPTION'

```
      SUBROUTINE FLUXM(NARCS)
      INTEGER NFLUXS/5785/
      INTEGER*2 MFLUX(5785)
      INTEGER*2
     .   MGF01(122),MGF02(184),MGF03(181),MGF04(184),MGF05(182),
     .   MGF06(184),MGF07(181),MGF08(184),MGF09(181),MGF10(184),
     .   MGF11(181),MGF12(184),MGF13(182),MGF14(184),
     .   MGFLU1(181),MGFLU2(184),MGFLU3(181),MGFLU4(184),MGFLU5(181),
     .   MGFLU6(184),MGFLU7(182),MGFLU8(184),MGFLU9(181),MGFL10(184),
     .   MGFL11(181),MGFL12(184),MGFL13(181),MGFL14(184),MGFL15(182),
     .   MGFL16(184),MGFL17(181),MGFL18(184)
      EQUIVALENCE
     .(MFLUX(1   ), MGF01(1)),(MFLUX(123 ), MGF02(1)),
     .(MFLUX(307 ), MGF03(1)),(MFLUX(488 ), MGF04(1)),
     .(MFLUX(672 ), MGF05(1)),(MFLUX(854 ), MGF06(1)),
     .(MFLUX(1038), MGF07(1)),(MFLUX(1219), MGF08(1)),
     .(MFLUX(1403), MGF09(1)),(MFLUX(1584), MGF10(1)),
     .(MFLUX(1768), MGF11(1)),(MFLUX(1949), MGF12(1)),
     .(MFLUX(2133), MGF13(1)),(MFLUX(2315), MGF14(1)),
     .(MFLUX(2499),MGFLU1(1)),(MFLUX(2680),MGFLU2(1)),
     .(MFLUX(2864),MGFLU3(1)),(MFLUX(3045),MGFLU4(1)),
     .(MFLUX(3229),MGFLU5(1)),(MFLUX(3410),MGFLU6(1)),
     .(MFLUX(3594),MGFLU7(1)),(MFLUX(3776),MGFLU8(1)),
     .(MFLUX(3960),MGFLU9(1)),(MFLUX(4141),MGFL10(1)),
     .(MFLUX(4325),MGFL11(1)),(MFLUX(4506),MGFL12(1)),
     .(MFLUX(4690),MGFL13(1)),(MFLUX(4871),MGFL14(1)),
     .(MFLUX(5055),MGFL15(1)),(MFLUX(5237),MGFL16(1)),
     .(MFLUX(5421),MGFL17(1)),(MFLUX(5602),MGFL18(1))
C------------MAGNETIC FLUX FOR 580301 TO 580630
```

```
      DATA MGF01/                                                      FLUX  56
     .  17, 20, 36, 41, 47, 45, 39, 37, 35, 33, 37, 56, 50, 35, 37, 31.FLUX  57
     .  40, 44, 49, 46, 44, 38, 36, 40, 44, 40, 32, 30, 24, 43, 33,    FLUX  58
     .  34, 44, 33, 45, 39, 37, 32, 23, 23, 13, 20, 17, 17, 35, 39, 43.FLUX  59
     .  52, 49, 38, 30, 29, 17, 20, 30, 25, 28, 24, 38, 43, 41,        FLUX  60
     .  36, 26, 20, 22, 27, 22, 17, 26, 24, 35, 24, 29, 41, 46, 40, 31.FLUX  61
     .  32, 33, 24, 13, 17, 13, 10,  3, 24, 47, 38, 33, 52, 27, 58,    FLUX  62
     .  55, 39, 17, 13, 22, 30, 59, 28, 43, 43, 33, 30, 23, 30, 36, 24.FLUX  63
     .  10, 20, 23, 22, 56, 46, 28, 33, 31, 22, 22, 53, 65, 23/        FLUX  64
C----------- MAGNETIC FLUX FOR 580701 TO 581231                        FLUX  65
      DATA MGF02/                                                      FLUX  66
     .  32, 17, 30, 36, 26, 17, 32, 79, 59, 32, 30, 36, 30, 29, 17, 20.FLUX  67
     .  31, 44, 39, 37, 52, 35, 17, 33, 37, 26, 41, 24, 23, 27, 30,    FLUX  68
     .  30, 27, 24, 10, 13, 13, 27, 10, 20, 31, 30, 24, 27, 20, 23, 27.FLUX  69
     .  60, 35, 26, 17, 17, 44, 28, 61, 33, 29, 56, 28, 26, 23, 20,    FLUX  70
     .  13, 13, 56, 70, 58, 17, 31, 35, 39, 27, 20, 13,  7, 10, 17, 47.FLUX  71
     .  28, 10, 10, 10, 10, 10, 13, 17, 60, 39, 24, 22, 13, 35,        FLUX  72
     .  30, 26, 33, 13, 28, 27, 29, 24, 13, 10, 17,  7, 20, 20, 24, 23.FLUX  73
     .  23, 22, 22, 17, 17, 50, 49, 62, 24, 26, 42, 46, 35, 34, 27,    FLUX  74
     .  20, 31, 27, 24,  7,  7, 17,  7, 10, 27, 32, 23, 17,  7, 17, 23.FLUX  75
     .  23, 22, 13, 10, 10,  7, 20, 22, 24, 22, 23, 33, 27,  3,        FLUX  76
     .   3, 37, 20, 52, 41, 27, 17, 24, 27, 10, 22, 17, 51, 34, 24, 31.FLUX  77
     .  42, 44, 36, 30, 22, 23, 27, 20, 10, 28, 27, 29, 24, 27, 20/    FLUX  78
C----------- MAGNETIC FLUX FOR 590101 TO 590630                        FLUX  79
      DATA MGF03/                                                      FLUX  80
     .   2,  7, 17, 20, 39, 38, 34, 33, 46, 49, 29, 28, 23, 22, 22, 32.FLUX  81
     .  33, 34, 24, 10,  7, 20, 22, 10, 33, 37, 23, 23, 32, 26, 29,    FLUX  82
     .  26, 37, 37, 45, 38, 33, 26, 27, 37, 17, 45, 38, 36, 42, 46, 55.FLUX  83
     .  38, 10, 27, 10, 17, 32, 30, 10, 57, 50, 42, 49,                FLUX  84
     .  48, 43, 37, 29, 27, 17, 17, 26, 10,  3, 10, 33, 27, 22, 17, 10.FLUX  85
     .  17, 20, 17, 13, 17, 13, 26, 22, 43, 60, 77, 62, 58, 35, 34,    FLUX  86
     .  26, 17, 27, 20, 17, 17, 22, 37, 49, 64, 37, 24, 23, 26, 24, 17.FLUX  87
     .  20, 13, 10, 13, 22,  7, 47, 40, 37, 33, 29, 27, 38, 36,        FLUX  88
     .  23, 20, 20, 31, 41,  7, 17, 44, 31, 29, 39, 66, 29, 13, 44, 45.FLUX  89
     .  28, 38, 29, 24, 28, 30, 27, 52, 39, 20, 13,  7,  7, 17, 37,    FLUX  90
     .  20, 34, 31, 37, 30, 27, 22, 26, 34, 27, 38, 17, 10, 23, 22, 17.FLUX  91
     .  17, 22, 20, 22, 22, 26, 30, 39, 17, 29, 43, 44, 51, 46/        FLUX  92
C----------- MAGNETIC FLUX FOR 590701 TO 591231                        FLUX  93
      DATA MGF04/                                                      FLUX  94
     .  13, 29, 13, 30, 31, 26, 29, 29, 33, 28, 49, 38, 26, 31, 83, 50.FLUX  95
     .  66, 68, 43, 33, 31, 28, 28, 41, 46, 44, 39, 28, 24, 10, 30,    FLUX  96
     .  35, 31, 35, 34, 23, 37, 29, 29, 37, 29, 22, 13, 17, 17, 40, 70.FLUX  97
     .  67, 41, 36, 41, 44, 40, 41, 35, 30, 22, 13,  7, 28, 23, 23,    FLUX  98
     .  35, 44, 44, 65, 44, 33, 20, 24, 17, 22, 30, 29, 29, 33, 33, 31.FLUX  99
     .  37, 40, 45, 55, 61, 58, 38, 38, 46, 37, 36, 33, 23, 27,        FLUX 100
     .  49, 37, 49, 51, 46, 52, 30, 22, 17, 10,  7, 20, 10, 28, 30, 10.FLUX 101
     .  32, 45, 30, 22, 17, 34, 24, 17, 37, 39, 26, 10, 17, 41, 46,    FLUX 102
     .  48, 57, 50, 45, 41, 37, 26, 28, 23, 24, 13, 20, 27, 41, 13, 22.FLUX 103
     .  27, 29, 26, 10, 31, 32, 47, 22, 24, 28, 30, 60, 30, 48,        FLUX 104
     .  41, 42, 51, 27, 57, 31, 17, 20, 22, 17, 17, 31, 33, 47, 39, 32.FLUX 105
     .  22, 24, 29, 17, 10, 17, 41, 30, 22, 39, 46, 47, 33, 29, 20/    FLUX 106
C----------- MAGNETIC FLUX FOR 600101 TO 500630                        FLUX 107
      DATA MGF05/                                                      FLUX 108
     .  10, 13, 17, 24, 38, 23, 17, 13, 10, 48, 40, 30, 24, 48, 42, 17.FLUX 109
     .  29, 37, 22, 32, 51, 35, 33, 33, 24, 20, 23, 17, 27,  3,  7,    FLUX 110
     .  22, 30, 34, 33, 34, 34, 13, 24, 17, 17, 22, 17, 26, 41, 23, 40.FLUX 111
```

```
     .  37, 41, 36, 34, 37, 24, 24, 10, 10, 17, 37, 17, 31,            FLUX 112
     .  34, 27, 36, 31, 29, 28, 10, 28, 26, 37, 44, 22, 13, 22, 36, 52,FLUX 113
     .  36, 27, 24, 10, 20, 13, 13, 31, 17, 20, 17, 32, 33, 35, 70,    FLUX 114
     .  84, 55, 57, 39, 44, 32, 37, 31, 29, 44, 39, 45, 41, 29, 38, 41,FLUX 115
     .  42, 26, 10,  3,  7, 13, 30, 56, 54, 33, 43, 61, 53, 76,        FLUX 116
     .  50, 30, 22, 13, 24, 55, 53, 69, 31, 27, 48, 35, 26, 28, 24, 48,FLUX 117
     .  29, 17, 13, 10, 17, 13, 39, 43, 34, 34, 31, 33, 52, 33, 28,    FLUX 118
     .  41, 13, 22, 52, 37, 39, 30, 38, 34, 17, 13, 13, 17, 29, 26, 17,FLUX 119
     .  20, 29, 31, 22, 31, 28, 26, 28, 39, 37, 56, 45, 45, 53/        FLUX 120
C---------- MAGNETIC FLUX FOR 600701 TO 501231                        FLUX 121
      DATA MGF06/                                                     FLUX 122
     .  39, 31, 28, 35, 31, 27, 13,  7, 10, 22, 24, 24, 27, 47, 63, 59,FLUX 123
     .  38, 29, 45, 39, 24, 24, 23, 24, 10, 22, 17, 17, 41, 41, 46,    FLUX 124
     .  29, 33, 20, 20,  7, 20, 20, 31, 37, 29, 37, 38, 22, 29, 22, 52,FLUX 125
     .  66, 31, 36, 39, 41, 29, 20, 17,  7, 10, 31, 29, 49, 54, 31,    FLUX 126
     .   7, 35, 45, 64, 68, 41, 40, 34, 31, 28, 30, 26, 36, 28, 10, 10,FLUX 127
     .  22, 31, 10, 13, 22, 24, 29, 40, 13, 24, 32, 20, 29, 41,        FLUX 128
     .  50, 50, 30, 45, 44, 80, 78, 44, 46, 24, 32, 17, 10,  7, 32, 17,FLUX 129
     .  20, 40, 24, 24, 22,  3,  7, 36, 59, 55, 46, 49, 47, 44, 41,    FLUX 130
     .  27, 31, 33, 52, 26, 17, 17,  7, 13, 17, 33, 57, 86, 50, 57, 63,FLUX 131
     .  33, 13, 17, 23, 49, 42, 24, 31, 47, 32, 36, 36, 23, 32,        FLUX 132
     .  63, 39, 23, 17, 24, 36, 39, 37, 35, 24, 23, 33, 29, 17, 48, 44,FLUX 133
     .  20, 39, 32, 38, 39, 35, 29, 30, 23, 31, 51, 37, 34, 32, 30/    FLUX 134
C---------- MAGNETIC FLUX FOR 610101 TO 510630                        FLUX 135
      DATA MGF07/                                                     FLUX 136
     .  20, 10, 13,  7, 10, 17, 22, 37, 42, 13,  2, 13, 23, 13, 34, 27,FLUX 137
     .  22, 32, 39, 47, 33, 36, 17, 33, 33, 27, 22, 27, 23, 10, 10,    FLUX 138
     .   7,  3, 28, 48, 41, 37, 26, 23, 20, 10, 23,  3, 37, 22, 20, 40,FLUX 139
     .  41, 51, 33, 42, 37, 33, 26, 24,  7, 10, 23, 27,                FLUX 140
     .  22, 22, 13,  7, 29, 46,  7, 13, 28, 49, 17, 20, 24, 39, 38, 35,FLUX 141
     .  29, 27, 46, 32, 26, 28, 24, 22, 13, 23, 37, 32, 31, 24, 17,    FLUX 142
     .  32, 29, 40, 17, 13, 23, 22, 17, 38, 32, 39, 24, 30, 52, 55, 28,FLUX 143
     .  10, 13, 20, 20,  3, 22, 22, 26, 22, 29, 27, 20, 13, 23,        FLUX 144
     .  30, 34, 10, 22, 35, 42, 35, 26, 30, 22, 34, 30, 37, 24, 10, 41,FLUX 145
     .  22,  7, 22, 28, 13, 26, 29, 22, 44, 20, 17, 24, 13, 23, 37,    FLUX 146
     .  42, 41, 24, 22, 23, 32, 38, 31, 17, 10,  3, 22,  7, 10, 26, 24,FLUX 147
     .  17, 29, 20, 24, 54, 54, 23, 17, 23, 17, 20, 10, 39, 10/        FLUX 148
C---------- MAGNETIC FLUX FOR 610701 TO 511231                        FLUX 149
      DATA MGF08/                                                     FLUX 150
     .  24, 27, 31, 31, 49, 31, 29, 24, 23, 29, 23, 13, 65, 64, 39, 37,FLUX 151
     .  45, 63, 33, 34, 45, 27, 32, 28, 29, 37, 67, 33, 22, 22, 17,    FLUX 152
     .  22, 48, 33, 33, 23, 20, 10, 32, 10, 26, 38, 20,  3, 22, 22, 13,FLUX 153
     .  13, 13, 23, 17, 13,  3,  7, 10, 23, 26, 22, 13, 31, 46, 42,    FLUX 154
     .  41, 28, 27, 13, 27, 13, 10, 10, 22, 20, 27, 32, 22, 41, 17, 20,FLUX 155
     .  26, 20,  7, 23,  3, 17,  3, 48, 45, 33, 39, 13, 17, 45,        FLUX 156
     .  67, 17, 13, 20, 10, 13, 17, 17,  7,  3, 27, 35, 29, 17,  7,  2,FLUX 157
     .   2,  7, 17, 33, 22, 13, 17, 13, 22, 43, 42, 69, 43, 17, 13,    FLUX 158
     .  13, 10, 10, 10, 34, 31, 48, 37, 27, 13,  7, 30, 10, 28,  2,  7,FLUX 159
     .  31, 50, 26, 28, 20,  3,  3,  7, 10, 13, 10,  7,  7,  3,        FLUX 160
     .  52, 56, 53, 24, 26, 30, 17,  3,  7, 17, 29, 17, 10, 10, 20, 10,FLUX 161
     .  10,  3,  2,  7,  7, 17, 27, 24,  3, 10, 23, 33, 28, 31, 20/    FLUX 162
C---------- MAGNETIC FLUX FOR 620101 TO 520630                        FLUX 163
      DATA MGF09/                                                     FLUX 164
     .  17, 23,  3,  2,  2,  7,  7, 10, 17, 52, 27, 10, 13, 26, 24, 27,FLUX 165
     .  10,  3, 30, 10, 20,  2,  2,  2, 10, 22, 24,  7, 20, 20,  2,    FLUX 166
     .   2, 10, 10, 35, 20, 17, 35,  7, 17,  7, 28, 35, 28, 30, 27, 41,FLUX 167
```

```
     *  29, 17,  7,  7, 20, 28, 24, 23, 22, 32, 32,  3,                 FLUX 168
     *  20, 20, 20, 17, 32, 43, 22,  3,  3, 27, 27, 31, 22, 10, 24, 10,FLUX 169
     *  10, 22, 28, 28, 30, 10, 10, 20, 23, 10, 10, 10, 22,  7, 10,    FLUX 170
     *  26, 23, 28, 27, 20, 40, 54, 43, 27, 42, 37, 23, 13,  7, 23, 20,FLUX 171
     *  22, 27, 20, 24, 35, 42, 27, 10, 28, 26, 20, 22, 17, 13,         FLUX 172
     *  23, 23, 20,  3, 10, 37, 20, 17,  3, 17, 23,  7, 28, 30, 30, 22,FLUX 173
     *   7,  3, 24, 13,  7,  7,  3,  3,  3, 10, 27, 17, 17,  7, 37,    FLUX 174
     *  23, 13, 13, 28, 24, 24, 23,  7, 37, 32, 17, 22, 13, 22, 26, 17,FLUX 175
     *   3,  3, 10, 10, 27, 24, 30, 17, 17, 20, 34, 32, 29, 28/         FLUX 176
C----------- MAGNETIC FLUX FOR 620701 TC 621231                         FLUX 177
      DATA MGF10/                                                       FLUX 178
     *  23, 22, 22, 34, 31, 27, 22, 24, 17, 22, 23, 20, 26, 23, 17,  7,FLUX 179
     *   3, 10, 24, 31, 27, 20, 22, 31, 30, 49, 41, 33, 24, 13, 26,    FLUX 180
     *  46, 24, 26, 17, 22, 33, 36, 43, 34, 24,  3, 10, 13, 23, 37, 36,FLUX 181
     *  38, 36, 32, 13, 20, 43, 32, 35, 28, 22, 13, 10, 30, 28, 39,    FLUX 182
     *  41, 44, 50, 45, 31, 37, 29, 29, 25, 26, 23, 54, 37, 26, 30, 24,FLUX 183
     *  20, 13, 47, 28, 27, 36, 28, 17, 20, 43, 23, 23, 34, 29,         FLUX 184
     *  48, 37, 26, 24, 27, 32, 27, 45, 43, 37, 39, 23, 26, 42, 24, 35,FLUX 185
     *  20, 32, 41, 26, 29, 37, 37, 40, 44, 44, 41, 33, 27, 29, 23,    FLUX 186
     *  22, 30, 29, 34, 17, 38, 30, 22, 13, 10, 26, 10,  7, 20, 39, 40,FLUX 187
     *  22,  2, 10, 10, 34, 45, 35, 31, 34, 17, 23, 20, 22, 43,         FLUX 188
     *  20,  7, 10, 32, 20,  7, 10, 20, 20, 20, 39, 26, 31, 28, 26, 13,FLUX 189
     *  45, 49, 48, 45, 37, 26, 10, 10, 10, 32, 17, 10, 13, 10, 27/    FLUX 190
C----------- MAGNETIC FLUX FOR 630101 TC 630630                         FLUX 191
      DATA MGF11/                                                       FLUX 192
     *  13,  2,  3, 22,  7,  0, 20,  3,  0,  7, 17, 13, 41, 43, 39, 38,FLUX 193
     *  34, 30, 34, 17,  7, 13, 23, 24, 20,  3,  2,  3, 13, 41, 48,    FLUX 194
     *  28,  3,  7,  7,  7, 10,  7,  2, 22, 49, 36, 36, 40, 34, 22, 10,FLUX 195
     *  10,  7,  3, 23, 17, 17, 17,  3, 10, 13,  3, 20,                 FLUX 196
     *  30, 17, 22, 10,  7, 13, 23, 43, 33, 50, 35, 27, 22,  3,  3,  3,FLUX 197
     *  10, 17, 20, 10,  7,  3, 26, 10,  7,  3,  2, 10, 13,  7, 10,    FLUX 198
     *  27, 10, 10, 34, 43, 34, 30, 20, 20,  3,  7, 23, 24, 32, 30, 17,FLUX 199
     *  20, 30, 24, 20, 10, 26, 23,  3, 10, 13, 24,  7, 10, 37,         FLUX 200
     *  45, 38, 29, 34, 24, 23, 17, 24, 29, 29, 33, 23, 37, 27, 17, 22,FLUX 201
     *  17,  7, 13, 22, 13,  7,  7,  3, 22, 17, 20, 31, 36, 27, 23,    FLUX 202
     *  26, 24, 17, 13,  3, 31, 48, 27, 24, 13, 17, 13, 13, 13, 23,  3,FLUX 203
     *  23, 36, 24, 28, 22, 10, 10, 17, 37, 39, 31, 28, 23, 26/         FLUX 204
C----------- MAGNETIC FLUX FCR 630701 TC 631231                         FLUX 205
      DATA MGF12/                                                       FLUX 206
     *  13,  7,  7, 29, 33, 35, 26, 28, 30, 24, 17, 10, 13, 10, 10, 20,FLUX 207
     *  31, 23, 13, 20, 39, 28, 37, 42, 28, 29, 31, 17, 10, 41, 33,    FLUX 208
     *  32, 30, 23, 28, 27, 26, 24, 17, 23, 20, 10, 10,  7,  7, 20, 17,FLUX 209
     *  23, 39, 38, 54, 40, 22, 33, 23, 22, 24, 31, 39, 30, 24, 32,    FLUX 210
     *  30, 22, 23, 13, 23, 23, 17, 30, 29, 23, 35, 28, 22, 60, 46, 44,FLUX 211
     *  48, 30, 39, 28, 49, 69, 59, 33, 55, 40, 44, 50, 34, 27,         FLUX 212
     *  10,  7, 10, 17, 22, 13, 26, 31, 17, 23, 37, 47, 38, 43, 31, 31,FLUX 213
     *  13, 13, 13, 28, 23,  7,  7, 55, 31, 22, 10, 20, 58, 45, 17,    FLUX 214
     *  22, 31, 31, 17,  3, 27, 47, 50, 44, 41, 31, 29, 13, 13, 10,  7,FLUX 215
     *  31,  7,  3, 10,  3, 17, 22, 35, 31,  3, 10, 13, 22, 33,         FLUX 216
     *  13, 27, 45, 39, 41, 37, 30, 29, 20,  0,  3, 10, 22, 26, 20, 20,FLUX 217
     *  10,  3, 10, 35, 26, 28, 28, 24,  7, 10, 10, 26, 31, 13,  7/    FLUX 218
C----------- MAGNETIC FLUX FCR 640101 TC 640630                         FLUX 219
      DATA MGF13/                                                       FLUX 220
     *  17, 52, 36, 34, 22, 13, 20, 17, 35, 36, 22, 17, 17,  2,  3, 40,FLUX 221
     *  32, 20, 20, 17,  3, 10, 10, 26, 31, 20, 10, 23, 36, 24, 39,    FLUX 222
     *  24, 22,  3, 27, 26, 42, 30, 35, 30, 20,  7, 29, 43, 24, 20, 13,FLUX 223
```

```
     . 17. 13.  3. 33. 30. 20. 22. 20. 34. 33. 30. 32. 22.            FLUX 224
     . 10.  2. 20. 52. 44. 32. 27. 33. 23. 10. 22. 31. 22. 24. 26. 22.FLUX 225
     . 24.  7.  7. 22. 27. 41. 42. 34. 31. 26.  7.  0.  7. 39.  7.    FLUX 226
     . 49. 39. 33. 26. 26. 17. 24. 29. 20. 13. 29. 10. 13. 10. 23. 23.FLUX 227
     . 31. 36. 37. 30. 23.  3.  7. 10. 26. 24. 39. 44. 31. 26.        FLUX 228
     . 37. 26. 10. 10. 23. 10.  7.  2.  3. 35. 34.  3. 33. 40. 41. 36.FLUX 229
     . 23. 13. 10.  7. 13. 10. 17. 36. 43. 20. 26. 22. 13. 17. 10.    FLUX 230
     . 13. 10. 10. 10.  7. 10. 13. 23. 22. 50. 39. 27. 23. 20. 17. 10.FLUX 231
     . 10. 20. 13. 32. 26. 13. 17. 13. 24. 13. 13. 26. 17.  7/        FLUX 232
C---------- MAGNETIC FLUX FOR 640701 TC 541231                        FLUX 233
      DATA MGF14/                                                     FLUX 234
     . 10.  7. 33. 22. 13. 13. 33. 36. 31. 28. 17. 17. 17.  7.  3. 20.FLUX 235
     . 37. 38. 29. 24. 17. 23. 13. 10. 13. 10.  3.  3. 31. 29. 20.    FLUX 236
     . 17. 10. 13. 44. 31. 20. 28. 10. 22.  7. 40. 29. 20. 10.  7. 17.FLUX 237
     . 13. 17. 13. 10. 10. 20. 10.  7. 22. 24. 22.  7. 13.  7. 28.    FLUX 238
     . 35. 24. 23. 23. 17. 27. 41. 37. 31. 24. 10.  3.  3.  3.  7. 28.FLUX 239
     . 23. 13.  7.  3. 13. 49. 22. 23. 13. 10. 17. 45. 20. 37.        FLUX 240
     . 23. 10. 24. 42. 35. 28. 28. 30. 29. 10.  7. 30. 20. 20. 17. 10.FLUX 241
     . 17. 27. 39. 27. 30.  7.  2. 13. 10. 33. 17. 13. 20.  7.  3.    FLUX 242
     . 29. 28. 10. 20. 23. 10.  2. 22. 34. 26. 13. 22. 10.  2. 27. 24.FLUX 243
     . 13. 13.  3.  7.  7. 17. 35.  3.  3. 26. 10. 24. 10. 26.        FLUX 244
     . 22.  7. 13. 10.  3.  7. 24. 17. 13.  7.  7.  3. 23. 20. 22. 30.FLUX 245
     . 30. 17. 23. 10. 10.  7. 10.  7. 10. 13.  3.  7. 10.  3.  7/    FLUX 246
C---------- MAGNETIC FLUX FOR 650101 TC 550630                        FLUX 247
      DATA  MGFLU1  /                                                 FLUX 248
     .  7. 26. 20. 10.  3.  3. 10. 28. 20. 13.  3. 27. 28. 17. 17.  7.FLUX 249
     . 24. 13. 10. 20. 24. 35. 20.  3.  3.  7. 17. 13. 13. 10.  7.    FLUX 250
     . 10.  3. 13. 24. 13. 29. 43. 32. 24. 27. 24. 10. 10. 26. 27. 17.FLUX 251
     .  3. 10. 10. 13. 32. 13. 33. 26. 29. 20. 23. 17.                FLUX 252
     . 20. 22. 39. 36. 20. 10. 20.  3. 10.  3. 10. 10. 24. 17. 27. 10.FLUX 253
     . 13.  3. 10. 13. 22. 22. 39. 27. 35. 28. 23. 13. 17.  3. 10.    FLUX 254
     . 10.  3.  7. 17. 10. 20. 22. 10. 26. 17. 22. 20. 13. 13. 10. 10.FLUX 255
     . 26. 57. 29. 24.  7. 13. 13. 13. 10. 17. 13.  7. 13. 13.        FLUX 256
     . 10.  3.  7. 10. 39. 17. 13. 26. 27. 20.  3. 10.  7.  7. 10. 33.FLUX 257
     . 13. 10.  3. 10. 13. 17. 13. 17.  7. 10. 17. 13.  7.  7. 10.    FLUX 258
     . 13. 17. 23. 26. 17. 13.  7. 23. 27.  3. 13.  7.  3. 17. 34. 58.FLUX 259
     . 44. 26.  3.  3.  3. 10.  7.  7. 26. 24. 17.  7. 26. 29/        FLUX 260
C---------- MAGNETIC FLUX FOR 650701 TC 551231                        FLUX 261
      DATA  MGFLU2  /                                                 FLUX 262
     . 28. 13. 13.  7.  7. 34. 22. 36. 27. 29.  3. 13. 13. 10. 24. 10.FLUX 263
     .  3. 17. 28. 13.  7. 10. 28. 20. 17. 10. 22. 30. 27. 10.  7.    FLUX 264
     . 13. 26. 17. 20. 10.  7. 17. 17. 20. 10. 17. 17.  7. 22. 17. 20.FLUX 265
     . 26. 36. 40. 32. 27. 13. 23. 29. 28. 20. 17.  7. 17. 22. 26.    FLUX 266
     . 17. 13. 10. 31. 23. 23. 20. 10.  7.  7. 10. 26. 17.  7. 30. 45.FLUX 267
     . 33. 31. 32. 13. 13. 13. 24. 24. 27. 23. 35. 40. 20.  7.        FLUX 268
     .  3. 31.  7.  2. 22.  3. 17. 30. 10.  7.  7. 17. 22. 17.  3.  3.FLUX 269
     .  3. 17. 10.  3.  2. 29. 34. 29. 26. 20. 17. 30. 13. 22. 17.    FLUX 270
     . 10. 13.  2. 22. 28. 32. 23. 10. 13.  0.  7.  7. 24.  7.  3.  2.FLUX 271
     . 10. 17. 24. 32. 24. 10.  3. 10. 13. 10. 10.  3.  7. 27.        FLUX 272
     . 34. 23.  3. 26.  7.  7.  7. 10. 17. 24. 24. 24. 17.  3.  2.  2.FLUX 273
     .  3. 27. 20. 10.  3. 17.  7. 23. 27. 34. 24. 31. 22. 17.  7/    FLUX 274
C---------- MAGNETIC FLUX FOR 660101 TC 560630                        FLUX 275
      DATA  MGFLU3  /                                                 FLUX 276
     .  3. 22. 17. 26. 10.  3. 20. 22. 22. 20.  3.  3.  3.  7. 13.  0.FLUX 277
     .  3. 13.  7. 30. 37. 40. 29. 29. 26. 29.  7. 20. 17.  3.  3.    FLUX 278
     .  7.  7. 26. 23. 33. 22. 10. 10.  7. 20. 27. 13. 17.  3. 10. 13.FLUX 279
```

```
     . 10,  7, 29, 32, 10, 29, 41, 34, 24,  7, 10,  3,                FLUX 280
     .  7,  7, 24, 23, 13, 10,  3,  7, 10, 24, 17, 17, 29, 56, 20, 20,FLUX 281
     . 17, 10, 35, 24, 22, 20, 57,  3, 29, 35, 28, 48, 27, 17,  7,    FLUX 282
     . 33, 28, 20, 22, 17, 20, 24, 24, 13, 13,  3,  7, 30, 22, 10,  7,FLUX 283
     . 10,  7,  3, 13, 13, 28, 24, 17,  7,  7,  2, 10, 17, 24,        FLUX 284
     . 22, 27, 13, 27, 17, 20, 10, 13, 13,  3, 24, 17, 20,  3,  3, 13,FLUX 285
     . 20, 13,  7, 22, 10, 10,  3,  3, 13, 59, 13, 13, 10, 17, 50,    FLUX 286
     . 27, 24, 20, 10, 13, 10, 23, 10,  7,  3,  7, 20, 13, 10, 17, 17,FLUX 287
     . 10,  7, 17, 20, 10,  7, 32, 31, 31, 17, 10, 13, 17, 17/        FLUX 288
C---------- MAGNETIC FLUX FOR 660701 TO 661231                        FLUX 289
      DATA  MGFLU4  /                                                 FLUX 290
     . 13, 10, 10, 29, 13, 13, 10, 37, 45, 39, 22, 30,  7, 10, 22, 17,FLUX 291
     . 26, 10, 13, 17, 29, 22, 17, 17, 10, 17, 26, 24, 13, 17, 13,    FLUX 292
     . 17,  2, 20, 22, 24, 17, 13, 13, 23, 27, 29, 29, 17, 23, 13, 10,FLUX 293
     .  3, 24, 35, 20, 13, 10, 37, 31, 22, 17, 13, 10, 28, 60, 37,    FLUX 294
     . 37, 30, 63, 67, 28, 38, 29, 48, 34, 34, 20, 13, 10, 24, 35, 24,FLUX 295
     . 23,  7, 32, 36, 24, 17, 32, 27, 28, 37, 33, 37, 32, 31,        FLUX 296
     . 17,  7, 10, 39, 45, 37, 22, 13, 23,  7,  3, 24, 22, 10, 29, 35,FLUX 297
     . 17, 10, 10, 10,  3,  3,  3, 26, 37, 30, 17, 13, 10, 28, 44,    FLUX 298
     . 43, 29, 32, 23, 23, 20, 17, 20,  7, 23, 17, 22, 20,  3, 10, 10,FLUX 299
     . 26, 27, 27, 22, 17,  7,  3, 17, 10, 24, 17, 34, 30, 41,        FLUX 300
     . 32, 17, 10, 34, 35, 20, 10, 10,  7, 10,  7,  3, 35, 50, 33, 22,FLUX 301
     . 23, 13,  7, 20, 27, 29, 22, 27, 29, 38, 44, 29, 20, 17,  7/    FLUX 302
C---------- MAGNETIC FLUX FOR 670101 TO 570630                        FLUX 303
      DATA  MGFLU5  /                                                 FLUX 304
     . 33, 20, 24,  3,  7, 17, 41, 55, 27, 10, 31,  3, 39, 55, 23, 23,FLUX 305
     . 10, 13, 13, 28, 23, 10, 13,  3, 13, 10, 10, 27, 10,  7,  3,    FLUX 306
     . 10,  3,  3, 26, 30, 20, 42, 49, 22,  3, 26,  3, 10, 10, 20, 56,FLUX 307
     . 30, 13, 13, 13, 17, 20, 29, 10, 30, 26, 13, 10,                FLUX 308
     . 13, 10, 22, 17, 26, 22, 13,  3, 27, 22,  3,  3, 13, 13,  3,  7,FLUX 309
     . 10, 37, 39, 28, 24, 10, 10,  7, 10, 10, 33, 22, 17, 26,  7,    FLUX 310
     . 33, 27, 13, 30, 26, 27, 22, 13, 13, 13, 10, 10,  3,  7, 10, 24,FLUX 311
     . 24, 20, 35, 20, 20, 33, 36, 41, 22,  7,  7,  7, 13,  7,        FLUX 312
     . 29, 39, 62, 27, 24, 17, 31, 13, 13, 20, 24, 28, 29, 24, 17, 17,FLUX 313
     . 26, 20, 22, 17, 17,  7, 23, 26, 70, 72, 35, 53, 49, 48, 48,    FLUX 314
     . 10, 23, 20, 28, 45, 49, 37, 30, 29, 20, 17, 13, 17, 29, 23, 17,FLUX 315
     . 23, 10, 13, 10, 10, 13, 10, 10, 38, 38, 34, 24, 23, 29/        FLUX 316
C---------- MAGNETIC FLUX FOR 670701 TO 571231                        FLUX 317
      DATA  MGFLU6  /                                                 FLUX 318
     . 32, 20, 10, 17, 31, 22, 27, 10,  7,  7, 36, 22, 20, 20, 22, 10,FLUX 319
     . 10, 20, 10, 13, 13,  7, 29, 22, 26, 17, 10, 27, 27, 32,  3,    FLUX 320
     . 10,  3,  7, 22, 22, 17, 23, 26, 20, 28, 39, 22, 17, 24, 17, 22,FLUX 321
     . 34, 29, 23, 26, 17, 13, 13, 20, 29, 24, 20, 17, 22, 24, 26,    FLUX 322
     . 39, 37, 17, 17, 10, 10, 23, 23, 26,  7,  7, 10, 41, 29, 29, 26,FLUX 323
     . 13, 27, 33, 49, 61, 24, 10, 17, 17, 10, 13, 45, 49, 45,        FLUX 324
     . 24, 13, 20, 10, 20, 17, 20, 27, 33, 44, 28, 34, 23, 28, 22, 13,FLUX 325
     . 24, 20, 10,  7,  2, 17, 22, 10,  7,  7, 24, 39, 39, 30, 20,    FLUX 326
     .  7, 24, 37, 23, 26, 10,  7, 33, 24, 10, 32, 37, 32, 26, 24, 23,FLUX 327
     .  3,  7,  7,  3, 13, 28, 22, 35, 26, 23, 22, 32, 30, 28,        FLUX 328
     . 44, 33, 22, 20, 28, 36, 35, 39, 24, 17,  7, 17, 13, 10, 24, 22,FLUX 329
     . 26, 34, 42, 43, 31, 30, 31,  3,  7, 20, 23, 13, 17, 23, 54/    FLUX 330
C---------- MAGNETIC FLUX FOR 680101 TO 580630                        FLUX 331
      DATA  MGFLU7  /                                                 FLUX 332
     . 39, 45, 23, 20, 24, 31, 22, 17,  7, 13, 24, 28, 23, 27, 24, 28,FLUX 333
     . 28, 24, 28, 30, 22, 23, 24, 24, 10, 28, 23, 29, 31, 26, 22,    FLUX 334
     . 24, 34, 31, 32, 22,  3, 10, 29, 34, 45, 51, 29, 33, 13, 39, 29,FLUX 335
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
     * 33, 38, 27, 45, 38, 24, 13, 20, 10, 10, 22, 42, 33,                FLUX 336
     * 24, 23, 29, 31, 32, 20, 13, 10, 23, 30, 22, 22, 13, 37, 41,  39,FLUX 337
     * 27, 27, 26, 35, 20, 13, 23, 37, 34, 29, 31, 27, 29, 40, 31,      FLUX 338
     * 39, 27, 28, 23, 44, 45, 26,  7,  3, 17, 24, 26, 37, 39, 29,  29,FLUX 339
     * 24, 22,  7,  3, 10, 22, 27, 22, 13, 39, 35, 31, 29, 20,          FLUX 340
     * 28, 29, 26, 17, 10,  7, 52, 20, 36, 24, 32, 40, 29, 26, 23,  26,FLUX 341
     * 30, 32, 26, 34, 38, 31, 28, 33, 17, 10, 10, 17, 23, 24, 24,      FLUX 342
     * 32, 32, 29, 23, 13, 13, 26, 28, 23, 45, 65, 46, 50, 39, 22,  28,FLUX 343
     * 33, 27, 29, 20,  7, 24, 17,  3, 10, 29, 26, 17, 28, 26/          FLUX 344
C-----------MAGNETIC FLUX FOR 680701 TO 631231                          FLUX 345
      DATA MGFL08 /                                                     FLUX 346
     * 24, 23, 31, 24, 22, 17, 20, 17, 10, 45, 24, 20, 37, 37, 17,  23,FLUX 347
     * 17, 24, 22, 13, 23, 34, 26,  7, 20, 30, 26, 22, 13, 20, 13,      FLUX 348
     * 10, 10, 27, 17, 27, 31, 28, 29, 28, 20, 17, 17, 26, 40, 37,  47,FLUX 349
     * 47, 34, 24, 17, 10, 13, 26, 34, 10, 10, 13, 10,  3,  7, 29,      FLUX 350
     * 23, 24, 33, 29, 27, 34, 33, 50, 27, 17, 17, 34, 48, 41, 42,  24,FLUX 351
     * 17,  7, 27, 20, 28, 26, 37, 10,  3, 10,  7, 22, 23, 20,          FLUX 352
     * 31, 50, 41,  7,  3, 17, 30, 20, 23, 17,  3, 51, 42, 29, 13,  13,FLUX 353
     * 23, 22, 26, 20,  3,  2,  7, 20, 17, 13, 17, 13, 46, 30, 67,      FLUX 354
     * 68, 60, 45, 40, 17, 20, 32, 24, 37, 27, 27,  7, 17, 10,  7,  32,FLUX 355
     * 34, 37, 17, 32, 20, 13, 17, 13, 22, 20, 24, 20, 10,  3,          FLUX 356
     * 23, 17, 33, 36, 39, 23, 13, 26, 17, 29, 17, 22, 17,  7, 17,  22,FLUX 357
     * 10, 20, 27, 38, 28, 28, 29, 23, 35, 10, 28, 13, 23, 22, 26/      FLUX 358
C----------- MAGNETIC FLUX FOR 690101 TO 590630                         FLUX 359
      DATA MGFL09/                                                      FLUX 360
     * 22, 10,  2,  7,  7,  2, 23, 22, 10,  7, 10, 17,  2, 23, 28,  24,FLUX 361
     * 36, 35, 26, 26, 17, 10, 13, 27, 41, 37, 27, 13,  3, 17, 17,      FLUX 362
     *  7, 50, 52, 28, 24, 31, 23, 23, 10, 27, 55, 24, 24, 24, 33,  26,FLUX 363
     * 10,  2, 22, 22, 22, 10, 23, 17, 20, 27, 43, 30,                  FLUX 364
     * 24, 23,  7, 17, 24, 29, 32, 26, 23, 13, 33, 47, 22, 22, 33,  30,FLUX 365
     * 46, 26, 32, 37, 28, 26, 50, 60, 37, 24, 17, 17, 28, 31, 30,      FLUX 366
     * 38, 31, 32, 26, 24, 26, 35, 22, 27, 17, 20, 26, 35, 31, 29,  32,FLUX 367
     * 36, 30, 13, 22, 17, 24, 13, 23, 23, 17, 26, 55, 27, 40,          FLUX 368
     * 22, 41, 30, 22, 24, 22, 13, 13, 26, 24, 13, 22, 47, 51, 70,  48,FLUX 369
     * 29, 35, 24, 22, 28, 26, 24, 24, 20, 10,  7, 23, 13, 26, 27,      FLUX 370
     * 13, 17, 13, 13, 17, 10, 22, 29, 30, 26, 22, 33, 30, 43, 22,  33,FLUX 371
     * 33, 10, 17, 24, 13,  3, 13, 27, 24, 17, 13, 10, 10, 13/          FLUX 372
C----------- MAGNETIC FLUX FOR 690701 TO 591231                         FLUX 373
      DATA MGFL10/                                                      FLUX 374
     * 32, 17,  7,  7,  3, 10, 17, 13, 20, 20, 17, 27, 29, 28, 13,  24,FLUX 375
     * 10,  7,  3,  7, 10, 20, 13, 10, 10, 35, 49, 10,  3, 28, 17,      FLUX 376
     *  7, 13, 29, 28, 20, 13, 22, 23, 26, 17, 10, 36, 20, 17, 10,  10,FLUX 377
     * 17, 20, 28, 20, 17, 13, 26, 17, 10, 30, 36, 17, 10, 10, 13,      FLUX 378
     *  7,  3, 10, 10, 33, 35, 26, 34, 24, 22, 22, 10,  3, 26, 34,  20,FLUX 379
     * 27, 34, 23, 22, 13,  7, 20, 17, 24, 10, 17, 50, 58, 62,          FLUX 380
     * 35, 43, 33, 24, 17, 31, 17,  7, 23, 36, 24, 23, 13,  7,  7,  13,FLUX 381
     * 17, 20, 24, 13, 22, 20, 10, 29, 20, 10, 22, 17, 10,  2, 13,      FLUX 382
     *  7, 26, 31, 17, 17,  7, 27, 29, 47, 46, 26, 20, 13,  3,  3,   3,FLUX 383
     *  7, 10, 20, 10,  3, 24, 13, 17, 17, 26, 37, 23, 20, 33,          FLUX 384
     * 13, 10, 10, 20, 33, 34, 13, 13, 27, 22, 26, 13,  7, 10, 13,  26,FLUX 385
     * 10,  7, 10,  7,  7, 20, 26, 23, 23, 24, 23, 10, 10,  3,  7/      FLUX 386
C----------- MAGNETIC FLUX FOR 700101 TO 700630                         FLUX 387
      DATA MGFL11/                                                      FLUX 388
     * 22, 42, 23,  7, 17, 13, 13, 17, 24, 17, 10, 22, 10, 17, 20,  32,FLUX 389
     * 26, 17, 13, 20, 17, 13, 13, 20,  3,  3, 17, 13, 17, 27, 22,      FLUX 390
     * 23, 35, 20, 27, 24, 10,  3,  3,  7, 13,  7,  7, 13, 26, 22,  13,FLUX 391
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
     . 24, 23, 10,  7,  3,  3, 10, 26, 10, 24, 22, 29,                FLUX 392
     . 37, 31, 30, 31, 27, 39, 48, 73, 50, 20,  7, 17, 22,  7, 20,  7,FLUX 393
     . 13, 17, 13, 10,  7,  3, 13,  3,  7, 13, 31, 36, 32, 31, 51,    FLUX 394
     . 17, 17, 26, 24, 23, 37, 24, 27, 33,  7, 22, 17, 10,  7, 13, 29,FLUX 395
     . 41, 33, 36, 36, 62, 47, 29, 31, 32, 28, 24, 13, 20, 33,        FLUX 396
     . 26, 23, 24, 17, 27, 17, 17, 10, 10,  3,  7, 30, 17, 28, 24, 17,FLUX 397
     . 24, 13, 17, 28, 22, 17, 20, 17, 20,  3, 30, 49, 28, 26, 17,    FLUX 398
     . 41, 23, 26, 22, 17,  7, 23, 28, 13, 17, 10, 13, 24, 22, 28, 23,FLUX 399
     . 28, 40, 22, 33, 32, 10, 10, 17, 17, 28, 45, 20, 17, 13/        FLUX 400
C---------- MAGNETIC FLUX FOR 700701 TC 701231                        FLUX 401
      DATA MGFL12/                                                    FLUX 402
     . 24, 26, 38, 39, 31, 30, 13, 24, 62, 44, 24, 29, 24, 23, 13, 17,FLUX 403
     . 22, 13, 13, 17, 42, 27, 23, 39, 63, 36, 29, 10, 49, 20, 29,    FLUX 404
     . 10, 17, 10, 10,  7, 22, 27, 39, 34, 20, 23, 23, 17, 10, 20, 36,FLUX 405
     . 67, 45, 29, 10, 13, 17, 20, 10, 22, 32, 27, 27, 24, 13, 27,    FLUX 406
     . 37, 33, 30, 26, 22, 13, 17, 22, 13, 17,  2, 13, 38, 34, 26, 22,FLUX 407
     . 22, 24, 33, 30, 37, 27, 17, 17, 22, 17, 34, 20, 13, 23,        FLUX 408
     . 24, 23, 29, 39, 23, 17, 10,  3,  3, 20, 30, 31, 23, 22, 10, 46,FLUX 409
     . 44, 47, 26, 20,  2, 33, 41, 29, 17, 10, 13, 30, 28, 23, 10,    FLUX 410
     .  3, 10, 22, 20, 23, 27, 54, 22, 22, 32, 35, 24, 23, 20, 10, 13,FLUX 411
     . 17, 33, 33,  7, 42, 33, 36, 27, 28, 22, 24, 22,  3,  2,        FLUX 412
     .  2, 10, 10, 13, 22, 22, 22, 34, 17,  7,  3, 10, 10, 56, 29, 10,FLUX 413
     .  7,  7, 24, 22, 10, 17, 20, 29, 10, 10, 22, 33, 27, 28,  3/    FLUX 414
C---------- MAGNETIC FLUX FOR 710101 TC 710630                        FLUX 415
      DATA MGFL13/                                                    FLUX 416
     . 17, 34, 44, 31, 23, 10,  2,  0,  2, 17, 22,  7, 17, 23, 22, 20,FLUX 417
     . 17, 30, 28, 41, 27, 24, 17, 24, 23,  3, 46, 47, 31, 39, 32,    FLUX 418
     . 31, 23, 10, 10, 10, 17, 17, 24, 24, 24, 13, 17, 10, 31, 41, 41,FLUX 419
     . 28, 27, 23, 22, 20,  7, 28, 28, 48, 43, 24, 17,                FLUX 420
     . 10, 10, 23, 29, 13, 10, 10, 30, 17, 30, 20, 35, 47, 41, 37, 29,FLUX 421
     . 23, 17, 31, 28,  7,  7,  7, 29, 27, 30, 23,  3,  3, 23, 39,    FLUX 422
     . 27, 20, 31, 40, 26, 29, 17, 17, 52, 42, 40, 24, 22, 47, 45, 30,FLUX 423
     . 22, 26, 22, 10, 37, 29, 24,  7,  2, 13, 22, 36, 23, 22,        FLUX 424
     . 20, 32, 22, 22, 22, 51, 45, 28, 27, 26, 10, 10, 10, 29, 27, 13,FLUX 425
     . 58, 47, 24, 13, 13, 13, 26, 20, 17, 23,  3, 10, 17, 34, 10,    FLUX 426
     . 37, 43, 39, 23, 17, 13,  7, 20,  7, 10, 17,  7, 22, 13, 13, 17,FLUX 427
     . 24, 17,  3, 10, 10, 13, 20, 13, 43, 27, 10, 23, 37, 24/        FLUX 428
C---------- MAGNETIC FLUX FOR 710701 TC 711231                        FLUX 429
      DATA MGFL14/                                                    FLUX 430
     . 24, 29, 20, 20, 20, 23,  7, 22, 17,  3, 17, 20, 23, 22, 22, 17,FLUX 431
     . 10, 20, 20, 10, 36, 22, 22, 13,  7, 32, 23, 17, 17, 22, 23,    FLUX 432
     . 17, 32, 10, 20, 29,  3, 13, 26, 23, 26, 30, 24, 17, 10, 13, 22,FLUX 433
     . 26, 29, 10, 10, 22, 23, 27, 20, 20, 24,  7, 17, 17, 17, 43,    FLUX 434
     . 24,  3, 13, 23, 29, 27, 35, 27, 20, 17, 17, 22, 32, 23, 23, 26,FLUX 435
     . 28, 50, 23, 31, 13, 10,  3, 17, 40, 36, 46, 24, 17, 37,        FLUX 436
     . 36, 36, 35, 29, 31, 35, 31, 37, 46, 22, 24, 23, 29, 23, 22, 10,FLUX 437
     .  7,  3,  3, 10, 13, 23, 17, 28, 17,  3, 10, 32, 40, 29, 13,    FLUX 438
     . 17,  7,  7, 17, 13,  7, 20, 22, 10, 13, 31, 23,  7,  2,  3,  3,FLUX 439
     .  2, 20, 20, 17, 26, 43, 49, 45, 48, 34, 22, 24, 13, 13,        FLUX 440
     . 17, 13, 27, 22, 13,  7,  2,  7, 20,  7, 17, 23, 28,  7,  7, 22,FLUX 441
     . 57, 37, 26,  7, 24, 34, 27, 20, 17, 27, 10, 10, 30, 31, 20/    FLUX 442
C---------- MAGNETIC FLUX FOR 720101 TC 720630                        FLUX 443
      DATA MGFL15/                                                    FLUX 444
     . 20, 22, 17, 24, 17,  3, 10, 10, 13, 20, 30, 17, 10,  7, 34, 41,FLUX 445
     . 35, 34, 26, 22, 37, 39, 43, 24, 32, 37, 30, 37, 29, 23, 20,    FLUX 446
     . 17, 28, 20, 21, 15, 14, 18, 15,  9, 23, 17,  9, 32, 28, 26, 20, FLUX 447
```

```
     . 33, 20, 24, 24, 22, 11, 16, 41, 30, 14, 11, 14,  4,
     . 19, 23, 30, 19, 13, 24, 46, 23, 23,  7, 14,  8, 17, 10, 13, 31,
     . 30, 22, 12, 13, 13, 21, 19, 33, 20, 23, 32, 19, 34, 38, 28,
     . 29, 18,  8, 30, 28, 17, 20, 14,  5, 14, 14, 23, 25, 18, 17, 14,
     . 15, 37, 20, 18, 30, 17, 20, 12,  7,  5, 14, 33, 47, 29,
     . 32, 32, 18, 14, 15, 22,  8,  9, 28, 23, 19, 22, 18, 18, 27, 25,
     . 22, 21,  8, 10, 10, 14, 16, 11, 14, 20, 18, 33, 23, 28, 26,
     . 14, 17, 25, 27, 25, 15, 19, 15, 10, 10,  8,  7, 11, 21, 23, 20,
     . 48, 66, 29, 18, 13, 25, 25, 26, 15, 21, 29, 27, 25,  7/
C----------- MAGNETIC FLUX FOR 720701 TC 721231
      DATA MGFL16/
     .  31*14,153*0/
C----------- MAGNETIC FLUX FOR 730101 TC 730630
      DATA MGFL17/
     .  181*0/
C----------- MAGNETIC FLUX FOR 730701 TC 731231
      DATA MGFL18/
     .  184*0/
      CALL FLUXS(NARCS,NFLUX,NFLUXS)
C-----DATE OF LAST MAGNETIC FLUX VALUE      720630
      RETURN
      END
```

FLUXS

DESCRIPTION

FLUXS stores in half-word integers ten times the daily 10.7 cm. flux line observed at Ottawa and adjusted to one A.U.

FLUXS tables must be updated using the values obtained from bulletins printed by E.S.S.A., Boulder, Colorado. An example of the bulletin is given on the following page.

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

DAILY SOLAR INDICES

JUNE 1972

JUN. 1972	YEAR DAY	BARTELS 27-DAY CYCLE NUMBER	SUNSPOT NUMBERS		OBSERVED FLUX OTTAWA 2800	SOLAR FLUX ADJUSTED TO 1 A.U.								
			R_Z	$R_{A'}$		AFCRL 15400	AFCRL 8800	AFCRL 4995	OTTAWA 2800	AFCRL 2695	AFCRL 1415	AFCRL 606	AFCRL 410	AFCRL 245
1	153	3	78	75	124.6	534	281	162	128.2	117.9	76.3	53.1	24.7	10.5
2	154	4	96	86	126.3	54	287	163	130.0	126.0	78.7	54.4	25.2	10.9
3	155	5	102	100	133.6*	54	293	169	137.5*	135.3	86.7	59.6	28.1	11.6
4	156	6	116	121	145.3*	54	300	184	149.5*	146.4	90.3	60.9	29.8	14.6
5	157	7	132	130	160.0	54	303	203	164.8	158.5	98.3	65.3	29.3	10.9
6	158	8	103	115	154.5	54	301	194	159.1	150.9	96.1	61.9	28.8	10.9
7	159	9	95	99	143.3	54	300	194	152.7	145.4	91.0	63.2	28.8	10.7
8	160	10	87	90	145.8*	54	310	199	150.2*	143.2	94.0	62.2	30.6	11.5
9	161	11	76	79	144.6*	55	311	197	149.1*	143.6	95.1	62.0	28.6	11.6
10	162	12	68	65	135.3	55	302	182	139.5	133.5	88.9	59.6	28.4	11.7
11	163	13	63	41	139.6*	54	301	186	143.2*	143.1	94.5	64.4	29.0	12.2
12	164	14	48	38	145.2	55	300	183	149.7	143.0	93.3	64.2	29.3	11.5
13	165	15	43	45	134.6*	54	280	174	138.9*	133.7	90.9	63.7	28.9	11.8
14	166	16	60	56	132.3	53	273	167	136.5	132.0	89.9	63.5	30.1	11.8
15	167	17	88	96	135.2*	55	293	195	139.5*	154.4	101.0	66.3	32.9	21.3
16	168	18	98	95	138.3*	54	281	176	142.7*	134.5	89.9	61.9	28.6	14.0
17	169	19	101	96	145.0	54	280	177	149.6	139.5	93.6	62.1	30.3	11.8
18	170	20	86	84	144.2*	54	281	178	149.0*	145.2	95.8	61.9	31.0	15.9
19	171	21	83	80	138.6*	53	277	173	143.2*	140.0	93.6	59.8	28.2	11.9
20	172	22	92	88	137.9	53	272	168	142.5	133.5	86.5	82.6	58.2	51.4
21	173	23	96	98	134.6*	53	279	171	139.0*	137.0	86.7	70.0	43.1	26.8
22	174	24	87	97	130.8	54	279	169	135.1	131.1	86.6	58.5	31.3	14.6
23	175	25	84	84	126.5	54	271	166	130.7	127.1	83.6	56.5	29.4	12.2
24	176	26	79	80	124.4	54	271	162	128.5	121.8	82.9	56.4	28.4	13.3
25	177	27	77	74	121.5	53	270	159	125.5	118.9	80.6	55.1	28.9	11.1
26	178	1	73	71	116.8	54	265	155	120.8	114.1	83.1	54.1	27.4	10.5
27	179	2	66	63	117.9*	53	265	156	121.9*	119.0	82.3	54.2	27.6	10.9
28	180	3	78	69	123.1	54	269	160	127.3	124.5	81.3	55.2	27.8	10.5
29	181	4	73	70	128.0	53	269	163	132.4	127.2	82.3	55.9	27.4	10.9
30	182	5	73	66	130.1	53	269	164	134.5	127.2	80.9	58.6	28.2	10.8
MEAN			83.4	81.7	135.4	54	284	175	139.7	134.9	88.8	60.9	30.3	14.0

NAME	FLUXS
PURPOSE	STORES SOLAR FLUX DATA THROUTH FEB., 1972
CALLING SEQUENCE	CALL FLUXS(NARCS,MFLUX,NFLUXS)

SYMBOL	TYPE	DESCRIPTION
NARCS	I	INPUT - NUMBER OF ARCS
MFLUX (1)	I	INPUT - MAGNETIC FLUX ARRAY
NFLUXS	I	INPUT - NUMBER OF SOLAR FLUX VALUES IN ARRAY

SUBROUTINES USED	ADFLUX
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE
REFERENCES	'SOLAR GEOPHYSICAL DATA, E.S.S.A., BOULDER, COLO. 'GEODYN SYSTEMS DESCRIPTION'

```
      SUBROUTINE FLUXS(NARCS,MFLUX,NFLUXS)
      INTEGER BEGYMD,ENDYMD
      INTEGER*2 SFLUX,MFLUX
      DIMENSION SFLUX(5785),MFLUX(1)
      INTEGER*2                   SFL01(61),SFL02(123),SFL03(122),
     .   SFL04(120),SFL05(123),SFL06(122),SFL07(121),SFL08(123),
     .   SFL09(122),SFL10(120),SFL11(123),SFL12(122),SFL13(120),
     .   SFL14(123),SFL15(122),SFL16(120),SFL17(123),SFL18(122),
     .   SFL19(121),SFL20(123),SFL21(122),
     .                            SFLUX1(120),SFLUX2(123),SFLUX3(122),
     .   SFLUX4(120),SFLUX5(123),SFLUX6(122),SFLUX7(120),SFLUX8(123),
     .   SFLUX9(122),SFLU10(121),SFLU11(123),SFLU12(122),SFLU13(120),
     .   SFLU14(123),SFLU15(122),SFLU16(120),SFLU17(123),SFLU18(122),
     .   SFLU19(120),SFLU20(123),SFLU21(122),SFLU22(121),SFLU23(123),
     .   SFLU24(122),SFLU25(120),SFLU26(123),SFLU27(122)
      EQUIVALENCE
     .(SFLUX(1    ), SFL01(1)),(SFLUX(62  ), SFL02(1)),
     .(SFLUX(185 ), SFL03(1)),(SFLUX(307 ), SFL04(1)),
     .(SFLUX(427 ), SFL05(1)),(SFLUX(550 ), SFL06(1)),
     .(SFLUX(672 ), SFL07(1)),(SFLUX(793 ), SFL08(1)),
     .(SFLUX(916 ), SFL09(1)),(SFLUX(1038), SFL10(1)),
     .(SFLUX(1158), SFL11(1)),(SFLUX(1281), SFL12(1)),
     .(SFLUX(1403), SFL13(1)),(SFLUX(1523), SFL14(1)),
     .(SFLUX(1646), SFL15(1)),(SFLUX(1768), SFL16(1)),
     .(SFLUX(1888), SFL17(1)),(SFLUX(2011), SFL18(1)),
```

```
     .(SFLUX(2133), SFL19(1)),(SFLUX(2254), SFL20(1)),
     .(SFLUX(2377), SFL21(1)),(SFLUX(2499),SFLUX1(1)),
     .(SFLUX(2619),SFLUX2(1)),(SFLUX(2742),SFLUX3(1))
      EQUIVALENCE
     .(SFLUX(2864),SFLUX4(1)),(SFLUX(2984),SFLUX5(1)),
     .(SFLUX(3107),SFLUX6(1)),(SFLUX(3229),SFLUX7(1)),
     .(SFLUX(3349),SFLUX8(1)),(SFLUX(3472),SFLUX9(1)),
     .(SFLUX(3594),SFLU10(1)),(SFLUX(3715),SFLU11(1)),
     .(SFLUX(3838),SFLU12(1)),(SFLUX(3960),SFLU13(1)),
     .(SFLUX(4080),SFLU14(1)),(SFLUX(4203),SFLU15(1)),
     .(SFLUX(4325),SFLU16(1)),(SFLUX(4445),SFLU17(1)),
     .(SFLUX(4568),SFLU18(1)),(SFLUX(4690),SFLU19(1)),
     .(SFLUX(4810),SFLU20(1)),(SFLUX(4933),SFLU21(1)),
     .(SFLUX(5055),SFLU22(1)),(SFLUX(5176),SFLU23(1)),
     .(SFLUX(5299),SFLU24(1)),(SFLUX(5421),SFLU25(1)),
     .(SFLUX(5541),SFLU26(1)),(SFLUX(5664),SFLU27(1))
      DATA BEGYMD,ENDYMD/580301,720731/
C------------SOLAR FLUX FOR 580301 TO 580430
      DATA SFL01/
     . 1950, 2090, 2230, 2320, 2330, 2510, 2560, 2510, 2550, 2420,
     . 2350, 2320, 2380, 2270, 2170, 2140, 2080, 2100, 2200, 2320,
     . 2240, 2560, 2680, 2740, 2580, 2840, 3020, 2950, 3320, 3440,
     . 3380,
     . 3310, 3260, 3020, 2950, 2900, 2890, 2930, 2720, 2500, 2440,
     . 2160, 1960, 1790, 1770, 1880, 1970, 2070, 2130, 2210, 2260,
     . 2290, 2370, 2350, 2440, 2480, 2450, 2470, 2580, 2550, 2650/
C----------- SOLAR FLUX FOR 580501 TO 580831
      DATA SFL02/
     . 2660, 2760, 2780, 2800, 2690, 2630, 2490, 2360, 2390, 2090,
     . 2110, 2090, 2030, 1940, 1960, 1940, 1940, 1970, 1970, 1970,
     . 1590, 1990, 2060, 2110, 2070, 2100, 2010, 2000, 2190, 2130,
     . 2090,
     . 2190, 2200, 2270, 2460, 2560, 2600, 2380, 2330, 2520, 2340,
     . 2350, 2270, 2200, 2080, 1970, 1910, 1820, 1770, 1890, 1930,
     . 1940, 2130, 2170, 2210, 2260, 2330, 2370, 2320, 2200, 2170,
     . 2150, 2150, 2240, 2320, 2380, 2320, 2370, 2320, 2180, 2090,
     . 2030, 1980, 1910, 1820, 1810, 1920, 1980, 1900, 2050, 2000,
     . 2090, 2140, 2130, 2280, 2400, 2600, 2610, 2900, 2850, 2870,
     . 2880,
     . 2740, 2540, 2370, 2160, 2210, 2390, 2350, 2360, 2250, 2220,
     . 2250, 2220, 2360, 2350, 2310, 2150, 2110, 2180, 2190, 2200,
     . 2310, 2390, 2430, 2530, 2630, 2640, 2520, 2440, 2520, 2490,
     . 2590/
C----------- SOLAR FLUX FOR 580901 TO 581231
      DATA SFL03/
     . 2610, 2610, 2700, 2560, 2330, 2160, 2100, 2110, 2350, 2450,
     . 2500, 2700, 2850, 2900, 2710, 2630, 2590, 2460, 2430, 2310,
     . 2210, 2490, 2260, 2250, 2220, 2180, 2190, 2250, 2270, 2290,
     . 2310, 2210, 2190, 2150, 1990, 1890, 1990, 1970, 1920, 1980,
     . 2100, 2190, 2250, 2280, 2300, 2530, 2860, 2860, 2960, 2790,
     . 2770, 2700, 2400, 2270, 1910, 1940, 1910, 2090, 2200, 2290,
     . 2220,
     . 2340, 2410, 2240, 2200, 2220, 2040, 2060, 1910, 1900, 1690,
     . 1660, 1560, 1630, 1660, 1660, 1690, 1730, 1740, 1830, 1870,
     . 1940, 2030, 2130, 2290, 2430, 2640, 2600, 2590, 2630, 2640,
     . 2690, 2590, 2410, 2410, 2530, 2490, 2530, 2620, 2540, 2690,
```

```
     . 2560, 2570, 2600, 2580, 2350, 2170, 2040, 2020, 1870, 1990,          FLUX 112
     . 1980, 2110, 2150, 2240, 2310, 2380, 2270, 2230, 2190, 2240,          FLUX 113
     . 2260/                                                                FLUX 114
C---------- SOLAR FLUX FOR 590101 TO 590430                                 FLUX 115
      DATA SFL04/                                                           FLUX 116
     . 2350, 2500, 2660, 2700, 2820, 2940, 3060, 2780, 2680, 2820,          FLUX 117
     . 2600, 2510, 2350, 2240, 2070, 2130, 2200, 2370, 2650, 2940,          FLUX 118
     . 3150, 3370, 3280, 3340, 3210, 3140, 3220, 3040, 2620, 2240,          FLUX 119
     . 2140,                                                                FLUX 120
     . 2050, 2050, 2040, 2010, 1910, 1950, 1920, 1820, 1740, 1900,          FLUX 121
     . 1990, 1990, 2010, 1990, 2130, 2140, 2200, 2120, 2180, 2280,          FLUX 122
     . 2240, 2190, 2120, 2200, 2270, 2190, 2110, 2010,                      FLUX 123
     . 1870, 1810, 1810, 1780, 1790, 1900, 1880, 1910, 1980, 2040,          FLUX 124
     . 2010, 1940, 2070, 2150, 2350, 2460, 2590, 2740, 2810, 2850,          FLUX 125
     . 2870, 2620, 2580, 2470, 2480, 2470, 2460, 2480, 2450, 2580,          FLUX 126
     . 2540,                                                                FLUX 127
     . 2560, 2360, 2160, 2150, 1960, 1960, 2170, 2150, 2150, 2200,          FLUX 128
     . 2320, 2240, 2090, 1980, 1890, 1880, 1900, 1810, 1960, 2030,          FLUX 129
     . 2030, 1990, 2130, 2130, 2110, 2180, 2120, 2160, 2110, 2200/          FLUX 130
C---------- SOLAR FLUX FOR 590501 TO 590831                                 FLUX 131
      DATA SFL05/                                                           FLUX 132
     . 2040, 1940, 1950, 1840, 1860, 2020, 2090, 2440, 2490, 2590,          FLUX 133
     . 2640, 2660, 2640, 2440, 2280, 2220, 2240, 2130, 1990, 2010,          FLUX 134
     . 2010, 2010, 1980, 1990, 2030, 2050, 2080, 1950, 1760, 1770,          FLUX 135
     . 1790,                                                                FLUX 136
     . 1930, 1980, 1980, 1900, 1970, 2100, 1980, 2130, 2230, 2290,          FLUX 137
     . 2260, 2200, 2120, 2080, 2250, 2200, 2250, 2280, 2370, 2260,          FLUX 138
     . 2280, 2190, 2200, 2320, 2330, 2380, 2400, 2240, 2190, 1950,          FLUX 139
C                                                                           FLUX 140
     . 1880, 1740, 1710, 1770, 1760, 1880, 1920, 1850, 1950, 2010,          FLUX 141
     . 1940, 2340, 2430, 2640, 2450, 2610, 2400, 2310, 2220, 2080,          FLUX 142
     . 1890, 1780, 1780, 1810, 1820, 1820, 2000, 2050, 2070, 2040,          FLUX 143
     . 2080,                                                                FLUX 144
     . 2140, 2290, 2400, 2360, 2290, 2270, 2120, 2040, 2040, 2020,          FLUX 145
     . 2000, 1960, 1940, 1890, 1900, 2010, 2240, 2290, 2150, 2240,          FLUX 146
     . 2300, 2430, 2620, 2450, 2530, 2570, 2790, 3020, 3080, 3120,          FLUX 147
     . 3050/                                                                FLUX 148
C---------- SOLAR FLUX FOR 590901 TO 591231                                 FLUX 149
      DATA SFL06/                                                           FLUX 150
     . 2820, 2590, 2570, 2390, 2200, 2000, 1920, 1990, 2090, 2010,          FLUX 151
     . 2030, 1950, 1890, 1960, 1840, 1680, 1700, 1670, 1750, 1850,          FLUX 152
     . 1820, 1980, 1820, 1830, 1750, 1640, 1630, 1620, 1590, 1560,          FLUX 153
     . 1500, 1450, 1440, 1480, 1550, 1740, 1590, 1550, 1530, 1490,          FLUX 154
     . 1470, 1540, 1550, 1530, 1600, 1670, 1690, 1710, 1730, 1750,          FLUX 155
     . 1750, 1870, 1830, 1810, 1860, 1900, 1830, 1770, 1720, 1670,          FLUX 156
     . 1610,                                                                FLUX 157
     . 1590, 1580, 1650, 1540, 1510, 1570, 1610, 1730, 1830, 1940,          FLUX 158
     . 1930, 1920, 1910, 1870, 1820, 1750, 1610, 1550, 1570, 1540,          FLUX 159
     . 1540, 1730, 1870, 2050, 2260, 2210, 2150, 2270, 2250, 2300,          FLUX 160
     . 2220, 2170, 2020, 1990, 2040, 2020, 1910, 1930, 1730, 1740,          FLUX 161
     . 1710, 1570, 1620, 1650, 1710, 1640, 1670, 1600, 1800, 1790,          FLUX 162
     . 1650, 1710, 1650, 1630, 1620, 1610, 1670, 1700, 1710, 1790,          FLUX 163
     . 1670/                                                                FLUX 164
C---------- SOLAR FLUX FOR 600101 TO 600430                                 FLUX 165
      DATA SFL07/                                                           FLUX 166
     . 1710, 1750, 1820, 1930, 2130, 2150, 2240, 2190, 2010, 1940,          FLUX 167
```

```
     . 2000, 1840, 1780, 1760, 1830, 1930, 1790, 1760, 1640, 1570,         FLUX 168
     . 1620, 1720, 1880, 2100, 2300, 2420, 2480, 2520, 2370, 2300,         FLUX 169
     . 2240,                                                               FLUX 170
     . 2250, 2130, 2150, 2090, 2090, 1920, 1870, 1830, 1830, 1780,         FLUX 171
     . 1750, 1560, 1670, 1670, 1600, 1580, 1530, 1510, 1470, 1420,         FLUX 172
     . 1560, 1490, 1430, 1400, 1470, 1470, 1470, 1400, 1400,               FLUX 173
     . 1370, 1370, 1380, 1390, 1400, 1350, 1390, 1410, 1430, 1320,         FLUX 174
     . 1320, 1290, 1350, 1340, 1370, 1420, 1400, 1330, 1370, 1430,         FLUX 175
     . 1450, 1500, 1540, 1560, 1570, 1660, 1650, 1750, 1810, 1930,         FLUX 176
     . 1820,                                                               FLUX 177
     . 2010, 1840, 1790, 1880, 1820, 1690, 1650, 1470, 1480, 1560,         FLUX 178
     . 1590, 1580, 1790, 1830, 1900, 1830, 1780, 1760, 1700, 1750,         FLUX 179
     . 1630, 1500, 1660, 1650, 1470, 1430, 1400, 1420, 1530, 1610/         FLUX 180
C---------- SOLAR FLUX FOR 600501 TO 600831                                FLUX 181
      DATA SFL08/                                                          FLUX 182
     . 1520, 1500, 1580, 1560, 1520, 1560, 1620, 1680, 1700, 1700,         FLUX 183
     . 1800, 1790, 1700, 1620, 1620, 1550, 1510, 1530, 1530, 1600,         FLUX 184
     . 1640, 1540, 1630, 1640, 1630, 1580, 1660, 1710, 1700, 1700,         FLUX 185
     . 1590,                                                               FLUX 186
     . 1660, 1570, 1670, 1720, 1700, 1750, 1850, 1850, 1810, 1780,         FLUX 187
     . 1710, 1570, 1620, 1660, 1660, 1570, 1530, 1390, 1400, 1330,         FLUX 188
     . 1310, 1300, 1360, 1320, 1400, 1550, 1640, 1840, 1900, 1940,         FLUX 189
     . 2080, 2070, 2100, 2120, 2090, 2000, 1970, 1760, 1760, 1650,         FLUX 190
     . 1530, 1420, 1350, 1350, 1460, 1440, 1530, 1590, 1560, 1520,         FLUX 191
     . 1530, 1480, 1510, 1590, 1480, 1490, 1500, 1490, 1540, 1460,         FLUX 192
     . 1450,                                                               FLUX 193
     . 1400, 1340, 1250, 1220, 1260, 1270, 1340, 1450, 1520, 1700,         FLUX 194
     . 1870, 2140, 2340, 2380, 2400, 2410, 2470, 2500, 2340, 2190,         FLUX 195
     . 2010, 1890, 1710, 1620, 1580, 1620, 1500, 1400, 1290, 1290,         FLUX 196
     . 1320/                                                               FLUX 197
C---------- SOLAR FLUX FOR 600901 TO 601231                                FLUX 198
      DATA SFL09/                                                          FLUX 199
     . 1370, 1520, 1490, 1420, 1420, 1490, 1620, 1700, 1730, 1750,         FLUX 200
     . 1750, 1770, 1810, 1810, 1780, 1770, 1850, 1900, 1990, 1950,         FLUX 201
     . 1890, 1840, 1750, 1620, 1550, 1480, 1420, 1320, 1240, 1210,         FLUX 202
     . 1150, 1120, 1200, 1320, 1320, 1320, 1440, 1430, 1510, 1590,         FLUX 203
     . 1520, 1590, 1620, 1660, 1650, 1650, 1670, 1540, 1530, 1490,         FLUX 204
     . 1440, 1410, 1340, 1290, 1300, 1320, 1320, 1220, 1310, 1280,         FLUX 205
     . 1270,                                                               FLUX 206
     . 1240, 1290, 1300, 1310, 1440, 1480, 1570, 1680, 1750, 2000,         FLUX 207
     . 1880, 1690, 1800, 1920, 1330, 1740, 1640, 1530, 1500, 1470,         FLUX 208
     . 1390, 1270, 1160, 1130, 1110, 1170, 1190, 1170, 1190, 1310,         FLUX 209
     . 1360, 1450, 1520, 1630, 1590, 1610, 1520, 1540, 1500, 1510,         FLUX 210
     . 1440, 1400, 1360, 1320, 1380, 1340, 1250, 1180, 1150, 1190,         FLUX 211
     . 1160, 1360, 1030, 1060, 1110, 1160, 1250, 1360, 1450, 1590,         FLUX 212
     . 1630/                                                               FLUX 213
C---------- SOLAR FLUX FOR 610101 TO 610430                                FLUX 214
      DATA SFL10/                                                          FLUX 215
     . 1640, 1760, 1750, 1650, 1600, 1430, 1320, 1250, 1220, 1150,         FLUX 216
     . 1100, 1030,  960,  960,  970, 1000, 1020, 1030, 1020, 1020,         FLUX 217
     . 1040, 1020, 1000, 1030, 1030, 1080, 1090, 1250, 1320, 1290,         FLUX 218
     . 1230,                                                               FLUX 219
     . 1230, 1220, 1180, 1180, 1190, 1210, 1140, 1110, 1080, 1040,         FLUX 220
     . 1010,  980,  970,  970,  980,  960,  960,  960,  960,  990,         FLUX 221
     . 1000, 1020, 1030, 1040, 1060, 1010, 1030, 1030,                     FLUX 222
     . 1030, 1030, 1040,  960,  940,  930,  950,  940,  900,  910,         FLUX 223
```

```
     .  980,  920,  930,  910,  980,  990,  980, 1010, 1020, 1050,       FLUX 224
     . 1050, 1060, 1100, 1160, 1180, 1210, 1250, 1260, 1260, 1250,       FLUX 225
     . 1170,                                                             FLUX 226
     . 1130, 1050, 1010, 1030, 1070, 1060,  980, 1040,  960,  930,       FLUX 227
     .  920,  890,  880,  930,  990, 1030, 1050, 1070, 1050, 1030,       FLUX 228
     . 1040, 1030, 1050, 1110, 1110, 1260, 1200, 1140, 1210, 1220/       FLUX 229
C---------- SOLAR FLUX FOR 610501 TO 610831                              FLUX 230
      DATA SFL11/                                                        FLUX 231
     . 1250, 1190, 1110, 1040, 1030,  970,  970,  940,  960,  920,       FLUX 232
     .  980, 1010,  970,  930,  910,  880,  880,  950, 1000, 1050,       FLUX 233
     . 1100, 1090, 1100, 1080, 1060,  880,  950,  910,  910,  880,       FLUX 234
     .  880,                                                             FLUX 235
     .  860,  880,  920,  890,  860,  880,  890,  910, 1000, 1020,       FLUX 236
     . 1100, 1080, 1140, 1230, 1290, 1320, 1370, 1360, 1310, 1310,       FLUX 237
     . 1320, 1340, 1350, 1170, 1110, 1080,  990,  950, 1020, 1030,       FLUX 238
     . 1040,  990, 1040, 1030, 1060, 1020, 1050, 1070, 1120, 1240,       FLUX 239
     . 1380, 1370, 1410, 1360, 1360, 1320, 1370, 1310, 1260, 1230,       FLUX 240
     . 1180, 1190, 1180, 1180, 1170, 1150, 1110, 1050, 1030,  920,       FLUX 241
     .  910,                                                             FLUX 242
     .  900,  870,  910,  880,  900,  920,  990, 1050, 1130, 1220,       FLUX 243
     . 1300, 1280, 1280, 1270, 1230, 1190, 1190, 1160, 1130, 1090,       FLUX 244
     . 1040, 1030,  980,  970,  930,  950,  950, 1000, 1030, 1060,       FLUX 245
     . 1080/                                                             FLUX 246
C---------- SOLAR FLUX FOR 610901 TO 611231                              FLUX 247
      DATA SFL12/                                                        FLUX 248
     . 1100, 1100, 1170, 1180, 1140, 1120, 1150, 1170, 1260, 1300,       FLUX 249
     . 1270, 1300, 1300, 1370, 1350, 1330, 1240, 1150, 1080, 1010,       FLUX 250
     .  960,  920,  900,  970,  970,  980,  960,  960, 1020, 1000,       FLUX 251
     .  980,  970,  970, 1020, 1080, 1010,  990,  980, 1070, 1060,       FLUX 252
     . 1070, 1110, 1110, 1050, 1060, 1000,  970,  950,  950,  930,       FLUX 253
     .  920,  890,  850,  850,  830,  830,  840,  860,  850,  870,       FLUX 254
     .  860,                                                             FLUX 255
     .  860,  830,  810,  820,  870,  870,  930,  990,  980, 1010,       FLUX 256
     .  990,  940,  910,  860,  800,  860,  830,  790,  770,  800,       FLUX 257
     .  830,  840,  870,  870,  920,  930,  950,  980,  980, 1040,       FLUX 258
     . 1050, 1080, 1110, 1050, 1010, 1010,  940,  960,  920,  870,       FLUX 259
     .  820,  730,  820,  810,  810,  810,  790,  810,  820,  880,       FLUX 260
     .  900,  990, 1010, 1040, 1030, 1020, 1030,  980,  980,  940,       FLUX 261
     .  930/                                                             FLUX 262
C---------- SOLAR FLUX FOR 620101 TO 620430                              FLUX 263
      DATA SFL13/                                                        FLUX 264
     .  890,  840,  790,  810,  780,  770,  770,  740,  740,  750,       FLUX 265
     .  760,  770,  740,  820,  860,  840,  870,  940,  990, 1070,       FLUX 266
     . 1120, 1110, 1160, 1140, 1150, 1150, 1150, 1150, 1090, 1010,       FLUX 267
     . 1020,                                                             FLUX 268
     . 1100, 1030, 1010, 1040,  920,  860,  920,  820,  830,  810,       FLUX 269
     .  820,  810,  840,  830,  830,  860,  870,  910, 1080, 1070,       FLUX 270
     . 1140, 1210, 1360, 1340, 1290, 1290, 1360, 1220,                   FLUX 271
     . 1210, 1120, 1000,  950,  860,  810,  800,  770,  790,  750,       FLUX 272
     .  780,  920,  810,  820,  940,  860,  940,  980, 1160, 1180,       FLUX 273
     . 1270, 1280, 1300, 1260, 1280, 1180, 1170, 1090, 1030,  990,       FLUX 274
     .  920,                                                             FLUX 275
     .  880,  830,  800,  780,  760,  780,  770,  770,  780,  810,       FLUX 276
     .  890,  930, 1020, 1100, 1110, 1190, 1140, 1100, 1090, 1090,       FLUX 277
     . 1120, 1120, 1000, 1050, 1010, 1000, 1000,  960,  930,  910/       FLUX 278
C---------- SOLAR FLUX FOR 620501 TO 620831                              FLUX 279
```

```
      DATA SFL14/                                                       FLUX 280
     .  940.  950.  940.  910.  870.  870.  830.  840.  870.  910.      FLUX 281
     .  980.  980.  960.  940.  910.  890.  930.  950.  970. 1030.      FLUX 282
     . 1060. 1100. 1110. 1110. 1120. 1100. 1090. 1030. 1040. 1050.      FLUX 283
     . 1040.                                                            FLUX 284
     .  980.  920.  870.  850.  850.  870.  920.  900.  910.  900.      FLUX 285
     .  890.  880.  890.  890.  930.  950.  980.  970.  980.  950.      FLUX 286
     .  900.  900.  860.  870.  900.  920.  930.  910.  910.  910.      FLUX 287
     .  900.  900.  900.  900.  880.  860.  880.  830.  800.  810.      FLUX 288
     .  830.  820.  860.  860.  850.  840.  840.  820.  800.  800.      FLUX 289
     .  790.  800.  780.  780.  740.  760.  740.  740.  730.  720.      FLUX 290
     .  730.                                                            FLUX 291
     .  710.  730.  720.  720.  700.  720.  710.  720.  730.  750.      FLUX 292
     .  740.  760.  790.  830.  920.  900.  890.  850.  830.  840.      FLUX 293
     .  820.  800.  790.  790.  770.  750.  730.  720.  720.  720.      FLUX 294
     .  750/                                                            FLUX 295
C---------- SOLAR FLUX FOR 620901 TO 621231                             FLUX 296
      DATA SFL15/                                                       FLUX 297
     .  840.  930.  980.  990.  980. 1000. 1000.  970.  940.  910.      FLUX 298
     .  900.  930.  920.  950.  930.  910.  890.  860.  840.  840.      FLUX 299
     .  830.  810.  820.  820.  840.  840.  840.  830.  860.  900.      FLUX 300
     .  860.  860.  830.  820.  860.  840.  850.  870.  860.  930.      FLUX 301
     .  930.  930.  950.  950.  940.  910.  910.  890.  870.  880.      FLUX 302
     .  870.  850.  840.  890.  870.  870.  860.  820.  800.  820.      FLUX 303
     .  810.                                                            FLUX 304
     .  800.  800.  800.  820.  820.  830.  840.  850.  860.  870.      FLUX 305
     .  870.  880.  930.  990.  950.  990.  940.  880.  890.  860.      FLUX 306
     .  810.  790.  770.  800.  770.  770.  750.  740.  750.  770.      FLUX 307
     .  770.  810.  830.  820.  820.  830.  860.  840.  830.  840.      FLUX 308
     .  780.  760.  770.  760.  760.  760.  780.  830.  840.  860.      FLUX 309
     .  850.  820.  790.  790.  780.  760.  750.  740.  740.  740.      FLUX 310
     .  750/                                                            FLUX 311
C---------- SOLAR FLUX FOR 630101 TO 630430                             FLUX 312
      DATA SFL16/                                                       FLUX 313
     .  750.  760.  770.  790.  770.  770.  770.  760.  780.  800.      FLUX 314
     .  810.  780.  790.  860.  850.  820.  820.  800.  780.  780.      FLUX 315
     .  760.  750.  740.  730.  740.  730.  810.  800.  790.  780.      FLUX 316
     .  820.                                                            FLUX 317
     .  870.  860.  850.  880.  870.  850.  830.  820.  790.  790.      FLUX 318
     .  760.  740.  740.  750.  760.  770.  790.  810.  790.  770.      FLUX 319
     .  740.  760.  750.  760.  780.  770.  750.  740.                  FLUX 320
     .  740.  750.  780.  800.  820.  850.  840.  830.  820.  800.      FLUX 321
     .  780.  770.  740.  800.  800.  790.  790.  800.  770.  770.      FLUX 322
     .  760.  760.  750.  750.  730.  730.  740.  730.  750.  740.      FLUX 323
     .  730.                                                            FLUX 324
     .  730.  740.  740.  700.  720.  780.  800.  810.  820.  820.      FLUX 325
     .  880.  930.  890.  870.  880.  880.  870.  880.  840.  790.      FLUX 326
     .  740.  720.  710.  730.  720.  720.  750.  780.  780.  800/      FLUX 327
C---------- SOLAR FLUX FOR 630501 TO 630831                             FLUX 328
      DATA SFL17/                                                       FLUX 329
     .  820.  820.  810.  820.  840.  870.  880.  860.  880.  870.      FLUX 330
     .  840.  870.  890.  950.  980. 1000. 1000.  980.  990.  910.      FLUX 331
     .  880.  890.  930.  890.  830.  760.  900.  790.  800.  830.      FLUX 332
     .  890.                                                            FLUX 333
     .  840.  810.  810.  790.  780.  770.  840.  900.  930.  990.      FLUX 334
     . 1030. 1090. 1070. 1000.  960.  890.  860.  820.  790.  750.      FLUX 335
```

```
     .  730.  720.  720.  720.  740.  740.  720.  740.  730.  760.        FLUX 336
     .  760.  770.  780.  780.  780.  770.  770.  770.  770.  750.        FLUX 337
     .  750.  740.  760.  770.  760.  760.  740.  740.  740.  770.        FLUX 338
     .  750.  730.  720.  720.  740.  730.  740.  730.  770.  840.        FLUX 339
     .  850.                                                              FLUX 340
     .  870.  870.  870.  880.  860.  880.  850.  810.  800.  770.        FLUX 341
     .  720.  730.  740.  710.  720.  760.  820.  800.  790.  810.        FLUX 342
     .  840.  860.  900.  870.  850.  820.  800.  770.  770.  770.        FLUX 343
     .  770/                                                              FLUX 344
C---------- SOLAR FLUX FOR 630901 TO 631231                               FLUX 345
      DATA SFL18/                                                         FLUX 346
     .  730.  730.  740.  750.  740.  740.  780.  750.  770.  760.        FLUX 347
     .  720.  770.  890.  980.  990. 1050.  990.  970. 1020. 1090.        FLUX 348
     .  900. 1050.  990.  950.  860.  840.  780.  740.  710.  690.        FLUX 349
     .  680.  690.  700.  710.  730.  770.  790.  850.  860.  870.        FLUX 350
     .  870.  840.  840.  860.  880.  870.  840.  830.  880.  890.        FLUX 351
     .  940.  960.  940.  940.  960.  960.  930.  840.  850.  850.        FLUX 352
     .  870.  850.  830.  830.  800.  780.  760.  750.  760.  750.        FLUX 353
     .  760.  770.  770.  780.  810.  810.  800.  820.  860.  840.        FLUX 354
     .  860.  860.  840.  830.  820.  820.  810.  790.  790.  790.        FLUX 355
     .  790.  800.  790.  770.  760.  760.  770.  780.  780.  790.        FLUX 356
     .  800.  820.  810.  790.  810.  780.  780.  790.  780.  790.        FLUX 357
     .  790.  770.  760.  760.  750.  740.  740.  730.  720.  710.        FLUX 358
     .  710/                                                              FLUX 359
C---------- SOLAR FLUX FOR 640101 TO 640430                               FLUX 360
      DATA SFL19/                                                         FLUX 361
     .  700.  683.  707.  704.  713.  725.  728.  708.  710.  709.        FLUX 362
     .  721.  737.  736.  731.  721.  720.  692.  715.  724.  735.        FLUX 363
     .  725.  723.  724.  720.  709.  715.  711.  749.  752.  727.        FLUX 364
     .  721.                                                              FLUX 365
     .  708.  695.  688.  692.  704.  707.  701.  712.  699.  708.        FLUX 366
     .  698.  709.  715.  708.  709.  713.  721.  743.  739.  745.        FLUX 367
     .  763.  780.  826.  834.  827.  848.  833.  828.  793.              FLUX 368
     .  761.  739.  725.  733.  709.  724.  718.  727.  706.  722.        FLUX 369
     .  741.  763.  773.  780.  778.  762.  767.  746.  736.  737.        FLUX 370
     .  739.  779.  769.  765.  737.  740.  749.  755.  748.  780.        FLUX 371
     .  767.                                                              FLUX 372
     .  773.  753.  768.  769.  761.  758.  757.  737.  752.  731.        FLUX 373
     .  743.  731.  734.  720.  719.  712.  724.  724.  715.  721.        FLUX 374
     .  723.  715.  711.  725.  714.  705.  705.  709.  698.  700/        FLUX 375
C---------- SOLAR FLUX FOR 640501 TO 650831                               FLUX 376
      DATA SFL20/                                                         FLUX 377
     .  700.  695.  710.  715.  732.  722.  722.  729.  722.  715.        FLUX 378
     .  715.  709.  699.  698.  695.  716.  712.  721.  703.  693.        FLUX 379
     .  697.  688.  690.  698.  693.  702.  695.  715.  710.  701.        FLUX 380
     .  696.                                                              FLUX 381
     .  697.  700.  702.  702.  698.  705.  717.  719.  711.  725.        FLUX 382
     .  725.  710.  724.  728.  739.  729.  738.  741.  724.  727.        FLUX 383
     .  720.  718.  696.  702.  693.  699.  697.  696.  694.  695.        FLUX 384
     .  697.  693.  696.  704.  699.  701.  693.  702.  694.  699.        FLUX 385
     .  699.  691.  687.  715.  718.  716.  709.  702.  696.  689.        FLUX 386
     .  685.  682.  685.  682.  679.  669.  673.  674.  679.  683.        FLUX 387
     .  682.                                                              FLUX 388
     .  682.  697.  701.  690.  710.  702.  696.  694.  698.  707.        FLUX 389
     .  699.  706.  759.  780.  777.  748.  733.  713.  722.  714.        FLUX 390
     .  709.  706.  703.  699.  691.  680.  696.  686.  685.  699.        FLUX 391
```

```
     .  702/                                                                          FLUX 392
C---------- SOLAR FLUX FOR 640901 TO 641231                                           FLUX 393
      DATA SFL21/                                                                     FLUX 394
     .  713,  704,  708,  710,  708,  715,  718,  719,  724,  728,                    FLUX 395
     .  730,  732,  729,  725,  719,  702,  690,  690,  694,  695,                    FLUX 396
     .  691,  591,  690,  684,  681,  697,  700,  705,  709,  716,                    FLUX 397
     .  720,  716,  718,  708,  716,  726,  740,  768,  729,  727,                    FLUX 398
     .  715,  697,  719,  703,  702,  705,  704,  718,  722,  719,                    FLUX 399
     .  703,  718,  723,  730,  754,  754,  749,  735,  732,  732,                    FLUX 400
     .  740,                                                                          FLUX 401
     .  736,  733,  727,  721,  716,  726,  715,  711,  704,  701,                    FLUX 402
     .  702,  707,  707,  711,  706,  703,  737,  733,  731,  740,                    FLUX 403
     .  720,  715,  699,  693,  695,  681,  696,  688,  710,  715,                    FLUX 404
     .  739,  738,  744,  757,  753,  746,  738,  750,  750,  760,                    FLUX 405
     .  770,  743,  755,  761,  765,  782,  778,  781,  791,  776,                    FLUX 406
     .  762,  749,  735,  725,  724,  715,  738,  747,  751,  741,                    FLUX 407
     .  759/                                                                          FLUX 408
C---------- SOLAR FLUX FOR 650101 TO 650430                                           FLUX 409
      DATA SFLUX1/                                                                    FLUX 410
     .  827,  815,  794,  780,  774,  777,  757,  749,  741,  735,                    FLUX 411
     .  727,  726,  725,  720,  724,  714,  712,  721,  715,  737,                    FLUX 412
     .  741,  744,  734,  741,  731,  735,  768,  751,  766,  786,                    FLUX 413
     .  777,                                                                          FLUX 414
     .  762,  768,  764,  752,  741,  741,  752,  734,  734,  739,                    FLUX 415
     .  718,  712,  706,  701,  708,  714,  716,  705,  706,  698,                    FLUX 416
     .  700,  703,  715,  729,  727,  723,  746,  749,                                FLUX 417
     .  751,  740,  737,  738,  748,  755,  758,  734,  719,  721,                    FLUX 418
     .  709,  726,  746,  737,  719,  705,  708,  743,  763,  738,                    FLUX 419
     .  734,  721,  725,  720,  734,  730,  718,  714,  717,  712,                    FLUX 420
     .  714,                                                                          FLUX 421
     .  711,  715,  713,  711,  708,  710,  708,  707,  719,  734,                    FLUX 422
     .  741,  737,  740,  759,  755,  751,  739,  737,  748,  731,                    FLUX 423
     .  737,  735,  740,  716,  704,  700,  702,  705,  712,  709/                    FLUX 424
C---------- SOLAR FLUX FOR 650501 TO 650831                                           FLUX 425
      DATA SFLUX2/                                                                    FLUX 426
     .  722,  719,  719,  709,  702,  717,  727,  734,  738,  735,                    FLUX 427
     .  725,  733,  760,  770,  823,  883,  932,  926,  946,  970,                    FLUX 428
     .  951,  945,  884,  873,  831,  800,  781,  765,  758,  761,                    FLUX 429
     .  736,                                                                          FLUX 430
     .  728,  758,  774,  805,  806,  807,  795,  807,  814,  807,                    FLUX 431
     .  789,  784,  792,  785,  791,  784,  787,  808,  774,  785,                    FLUX 432
     .  784,  807,  828,  807,  819,  814,  803,  793,  790,  796,                    FLUX 433
     .  786,  782,  788,  785,  780,  812,  843,  838,  841,  830,                    FLUX 434
     .  826,  808,  788,  772,  770,  745,  743,  743,  749,  754,                    FLUX 435
     .  750,  739,  732,  724,  723,  717,  723,  729,  734,  734,                    FLUX 436
     .  730,                                                                          FLUX 437
     .  742,  752,  796,  807,  785,  811,  818,  795,  798,  782,                    FLUX 438
     .  788,  779,  767,  756,  744,  756,  742,  750,  761,  755,                    FLUX 439
     .  753,  746,  746,  743,  738,  735,  760,  747,  757,  764,                    FLUX 440
     .  763/                                                                          FLUX 441
C---------- SOLAR FLUX FOR 650901 TO 651231                                           FLUX 442
      DATA SFLUX3/                                                                    FLUX 443
     .  769,  773,  779,  780,  800,  783,  789,  798,  772,  767,                    FLUX 444
     .  767,  763,  759,  761,  757,  745,  745,  777,  734,  734,                    FLUX 445
     .  731,  717,  723,  765,  762,  774,  787,  808,  876,  893,                    FLUX 446
     .  922,  933,  961,  975,  916,  851,  835,  826,  830,  801,                    FLUX 447
```

```
     .  757,  745,  754,  743,  733,  718,  720,  716,  712,  720,          FLUX 448
     .  726,  754,  779,  755,  770,  773,  770,  762,  756,  751,          FLUX 449
     .  769,                                                                FLUX 450
     .  776,  782,  798,  784,  767,  792,  837,  789,  804,  824,          FLUX 451
     .  825,  791,  757,  744,  748,  724,  726,  733,  716,  709,          FLUX 452
     .  705,  700,  695,  693,  688,  699,  721,  749,  719,  730,          FLUX 453
     .  733,  729,  727,  723,  732,  740,  730,  744,  727,  730,          FLUX 454
     .  732,  735,  717,  724,  744,  751,  759,  759,  743,  721,          FLUX 455
     .  717,  700,  703,  688,  697,  744,  809,  810,  819,  792,          FLUX 456
     .  781/                                                                FLUX 457
C---------- SOLAR FLUX FOR 660101 TO 660430                                 FLUX 458
      DATA SFLUX4/                                                          FLUX 459
     .  793,  763,  759,  778,  774,  771,  782,  779,  774,  772,          FLUX 460
     .  782,  812,  843,  901,  985, 1026,  984, 1014, 1051,  990,          FLUX 461
     .  957,  918,  906,  889,  854,  827,  799,  781,  783,  763,          FLUX 462
     .  754,                                                                FLUX 463
     .  776,  769,  775,  790,  806,  821,  828,  823,  829,  838,          FLUX 464
     .  836,  832,  839,  839,  833,  827,  821,  822,  811,  828,          FLUX 465
     .  857,  860,  827,  819,  793,  831,  832,  841,                      FLUX 466
     .  797,  767,  758,  755,  748,  755,  762,  764,  785,  786,          FLUX 467
     .  780,  783,  800,  814,  871,  929, 1051, 1095, 1146, 1110,          FLUX 468
     . 1203, 1051,  962,  929,  911,  847,  831,  876,  961,  990,          FLUX 469
     . 1104,                                                                FLUX 470
     . 1069, 1063, 1021, 1026, 1020, 1042, 1028, 1073, 1003,  948,          FLUX 471
     .  939,  948,  931,  910,  963,  932,  952,  929,  890,  935,          FLUX 472
     .  917,  934,  938, 1037, 1039, 1013,  960,  949,  945,  933/          FLUX 473
C---------- SOLAR FLUX FOR 660501 TO 660831                                 FLUX 474
      DATA SFLUX5/                                                          FLUX 475
     .  917,  940,  940,  925,  886,  875,  899,  878,  875,  866,          FLUX 476
     .  883,  926,  929,  972,  992, 1001,  989,  987, 1071, 1155,          FLUX 477
     . 1236, 1210, 1139, 1177, 1151, 1123, 1085, 1076, 1068, 1016,          FLUX 478
     . 1056,                                                                FLUX 479
     . 1048, 1039, 1026, 1020, 1017, 1019,  959,  995,  989,  958,          FLUX 480
     .  961,  959,  961,  969,  947,  979,  995,  982,  969,  943,          FLUX 481
     .  935,  961,  992, 1035, 1048, 1056, 1008, 1014,  998, 1007,          FLUX 482
     . 1001,  982,  993, 1045, 1050, 1097, 1126, 1144, 1078, 1081,          FLUX 483
     . 1090, 1027, 1004,  998, 1011, 1028, 1012, 1013, 1015, 1018,          FLUX 484
     . 1037, 1065, 1149, 1206, 1260, 1276, 1238, 1242, 1329, 1290,          FLUX 485
     . 1246,                                                                FLUX 486
     . 1259, 1196, 1182, 1160, 1105, 1060, 1015,  977,  964,  943,          FLUX 487
     .  925,  928,  932,  928,  937,  951,  968,  975, 1000, 1016,          FLUX 488
     . 1027, 1055, 1147, 1222, 1263, 1302, 1334, 1326, 1298, 1261,          FLUX 489
     . 1209/                                                                FLUX 490
C---------- SOLAR FLUX FOR 660901 TO 661231                                 FLUX 491
      DATA SFLUX6/                                                          FLUX 492
     . 1166, 1063, 1032, 1019, 1005,  979,  959,  962,  953,  939,          FLUX 493
     .  966, 1008, 1024, 1074, 1120, 1246, 1291, 1426, 1466, 1460,          FLUX 494
     . 1372, 1315, 1275, 1260, 1188, 1094, 1029,  979,  986,  957,          FLUX 495
     . 1014, 1020, 1032, 1006, 1000, 1019, 1031,  974, 1035, 1065,          FLUX 496
     . 1098, 1149, 1228, 1203, 1206, 1203, 1205, 1135, 1156, 1241,          FLUX 497
     . 1209, 1198, 1111, 1061, 1008,  977,  920,  941,  997,  957,          FLUX 498
     .  971,                                                                FLUX 499
     .  946,  967,  931,  917,  979, 1047, 1134, 1169, 1172, 1219,          FLUX 500
     . 1261, 1262, 1264, 1240, 1226, 1213, 1132, 1134, 1119, 1109,          FLUX 501
     . 1107, 1165, 1147, 1133, 1107, 1073, 1111, 1041,  990,  946,          FLUX 502
     .  922,  951,  999, 1048, 1109, 1156, 1177, 1237, 1462, 1573,          FLUX 503
```

```
     . 1628, 1576, 1555, 1495, 1449, 1351, 1249, 1112, 1123, 1076,      FLUX 504
     . 1065, 1055, 1106, 1105, 1116, 1109, 1096, 1075, 1093, 1151,      FLUX 505
     . 1205/                                                            FLUX 506
C---------- SOLAR FLUX FOR 670101 TO 670430                             FLUX 507
      DATA SFLUX7/                                                      FLUX 508
     . 1244, 1430, 1540, 1607, 1682, 1605, 1536, 1429, 1447, 1456,      FLUX 509
     . 1398, 1391, 1381, 1352, 1266, 1202, 1169, 1174, 1164, 1270,      FLUX 510
     . 1382, 1399, 1488, 1468, 1427, 1543, 1583, 1562, 1582, 1590,      FLUX 511
     . 1564,                                                            FLUX 512
     . 1516, 1435, 1387, 1373, 1468, 1488, 1625, 1483, 1459, 1405,      FLUX 513
     . 1337, 1329, 1300, 1292, 1264, 1249, 1222, 1242, 1210, 1285,      FLUX 514
     . 1318, 1460, 1493, 1622, 1595, 1733, 1767, 1802,                  FLUX 515
     . 1942, 1978, 1964, 2059, 1792, 1774, 1638, 1561, 1579, 1484,      FLUX 516
     . 1416, 1343, 1291, 1272, 1324, 1321, 1326, 1322, 1360, 1404,      FLUX 517
     . 1472, 1495, 1557, 1619, 1692, 1639, 1628, 1807, 1784, 1758,      FLUX 518
     . 1676,                                                            FLUX 519
     . 1586, 1410, 1333, 1250, 1221, 1190, 1262, 1357, 1333, 1303,      FLUX 520
     . 1311, 1298, 1268, 1335, 1241, 1261, 1259, 1287, 1267, 1271,      FLUX 521
     . 1340, 1315, 1280, 1305, 1329, 1254, 1292, 1351, 1375, 1353/      FLUX 522
C---------- SOLAR FLUX FOR 670501 TO 670831                             FLUX 523
      DATA SFLUX8/                                                      FLUX 524
     . 1366, 1318, 1283, 1258, 1281, 1228, 1197, 1169, 1136, 1094,      FLUX 525
     . 1061, 1097, 1073, 1086, 1110, 1131, 1154, 1248, 1356, 1463,      FLUX 526
     . 1603, 1827, 1940, 2009, 2107, 2189, 2138, 2026, 1883, 1774,      FLUX 527
     . 1754,                                                            FLUX 528
     . 1746, 1580, 1504, 1484, 1387, 1305, 1191, 1101, 1037,  971,      FLUX 529
     .  969,  965,  998, 1038, 1072, 1119, 1120, 1154, 1190, 1213,      FLUX 530
     . 1229, 1206, 1331, 1324, 1331, 1279, 1315, 1367, 1334, 1261,      FLUX 531
     . 1250, 1257, 1296, 1272, 1237, 1166, 1160, 1118, 1097, 1058,      FLUX 532
     . 1071, 1061, 1138, 1224, 1276, 1301, 1296, 1213, 1286, 1355,      FLUX 533
     . 1402, 1524, 1618, 1764, 1955, 1999, 2132, 2131, 2157, 2116,      FLUX 534
     . 1889,                                                            FLUX 535
     . 1766, 1667, 1635, 1493, 1554, 1554, 1502, 1434, 1412, 1367,      FLUX 536
     . 1362, 1339, 1375, 1300, 1274, 1329, 1483, 1571, 1723, 1729,      FLUX 537
     . 1788, 1785, 1783, 1659, 1708, 1720, 1712, 1699, 1727, 1640,      FLUX 538
     . 1657/                                                            FLUX 539
C---------- SOLAR FLUX FOR 670901 TO 671231                             FLUX 540
      DATA SFLUX9/                                                      FLUX 541
     . 1605, 1566, 1470, 1409, 1387, 1338, 1284, 1739, 1315, 1336,      FLUX 542
     . 1318, 1317, 1276, 1280, 1304, 1264, 1267, 1287, 1277, 1306,      FLUX 543
     . 1266, 1246, 1238, 1318, 1341, 1344, 1373, 1346, 1317, 1317,      FLUX 544
     . 1397, 1350, 1315, 1259, 1257, 1299, 1258, 1215, 1221, 1273,      FLUX 545
     . 1365, 1354, 1302, 1273, 1226, 1180, 1173, 1133, 1193, 1312,      FLUX 546
     . 1301, 1373, 1394, 1413, 1421, 1482, 1611, 1770, 1634, 1595,      FLUX 547
     . 1486,                                                            FLUX 548
     . 1403, 1409, 1375, 1305, 1218, 1196, 1131, 1128, 1111, 1173,      FLUX 549
     . 1241, 1327, 1347, 1369, 1429, 1518, 1549, 1569, 1667, 1660,      FLUX 550
     . 1614, 1594, 1566, 1525, 1547, 1579, 1550, 1572, 1503, 1444,      FLUX 551
     . 1400, 1115, 1205, 1203, 1216, 1303, 1276, 1296, 1309, 1277,      FLUX 552
     . 1365, 1391, 1430, 1534, 1643, 1813, 2020, 2124, 2054, 2015,      FLUX 553
     . 1980, 1823, 1640, 1529, 1594, 1653, 1770, 1841, 1765, 1589,      FLUX 554
     . 1526/                                                            FLUX 555
C---------- SOLAR FLUX FOR 680101 TO 680430                             FLUX 556
      DATA SFLU10/                                                      FLUX 557
     . 1714, 1718, 1793, 1833, 1912, 2097, 2239, 2302, 2220, 2083,      FLUX 558
     . 2025, 1999, 1972, 1870, 1765, 1608, 1574, 1485, 1416, 1373,      FLUX 559
```

```
     . 1325, 1357, 1379, 1486, 1601, 1740, 1850, 2130, 2207, 2291,     FLUX 560
     . 2391,                                                           FLUX 561
     . 2541, 2530, 2443, 2203, 1914, 1722, 1642, 1515, 1517, 1557,     FLUX 562
     . 1557, 1591, 1491, 1457, 1418, 1425, 1396, 1352, 1375, 1386,     FLUX 563
     . 1485, 1518, 1565, 1640, 1774, 1839, 1757, 1707, 1674,           FLUX 564
     . 1678, 1574, 1508, 1385, 1378, 1760, 1314, 1261, 1223, 1241,     FLUX 565
     . 1283, 1304, 1278, 1279, 1259, 1282, 1352, 1322, 1301, 1296,     FLUX 566
     . 1400, 1422, 1455, 1544, 1560, 1568, 1592, 1571, 1562, 1603,     FLUX 567
     . 1558,                                                           FLUX 568
     . 1438, 1441, 1371, 1317, 1291, 1251, 1247, 1307, 1411, 1428,     FLUX 569
     . 1384, 1427, 1419, 1383, 1400, 1410, 1322, 1277, 1253, 1224,     FLUX 570
     . 1234, 1156, 1117, 1175, 1199, 1143, 1191, 1284, 1280, 1328/     FLUX 571
C---------- SOLAR FLUX FOR 680501 TO 680831                            FLUX 572
      DATA SFLU11/                                                     FLUX 573
     . 1448, 1482, 1563, 1545, 1560, 1485, 1454, 1416, 1399, 1397,     FLUX 574
     . 1335, 1273, 1283, 1382, 1418, 1507, 1665, 1784, 1853, 1936,     FLUX 575
     . 1946, 1903, 1857, 1826, 1810, 1771, 1709, 1581, 1509, 1513,     FLUX 576
     . 1489,                                                           FLUX 577
     . 1520, 1454, 1438, 1440, 1383, 1455, 1522, 1546, 1490, 1479,     FLUX 578
     . 1465, 1434, 1434, 1395, 1397, 1382, 1396, 1432, 1475, 1515,     FLUX 579
     . 1572, 1546, 1636, 1575, 1540, 1462, 1476, 1428, 1391, 1359,     FLUX 580
     . 1283, 1203, 1172, 1160, 1194, 1201, 1350, 1471, 1536, 1570,     FLUX 581
     . 1628, 1561, 1556, 1559, 1479, 1495, 1436, 1352, 1353, 1339,     FLUX 582
     . 1330, 1390, 1461, 1527, 1582, 1544, 1468, 1436, 1443, 1385,     FLUX 583
     . 1347,                                                           FLUX 584
     . 1342, 1342, 1409, 1355, 1359, 1460, 1398, 1418, 1467, 1466,     FLUX 585
     . 1507, 1616, 1708, 1851, 1857, 1790, 1692, 1648, 1598, 1599,     FLUX 586
     . 1586, 1538, 1377, 1308, 1242, 1180, 1145, 1194, 1217, 1255,     FLUX 587
     . 1235/                                                           FLUX 588
C---------- SOLAR FLUX FOR 680901 TO 681231                            FLUX 589
      DATA SFLU12/                                                     FLUX 590
     . 1295, 1342, 1428, 1432, 1396, 1360, 1411, 1494, 1494, 1562,     FLUX 591
     . 1525, 1560, 1506, 1467, 1357, 1320, 1315, 1297, 1273, 1274,     FLUX 592
     . 1336, 1275, 1268, 1430, 1581, 1600, 1578, 1598, 1540, 1443,     FLUX 593
     . 1365, 1363, 1452, 1461, 1479, 1453, 1469, 1416, 1373, 1346,     FLUX 594
     . 1339, 1368, 1272, 1321, 1370, 1372, 1425, 1496, 1537, 1555,     FLUX 595
     . 1633, 1648, 1670, 1733, 1717, 1725, 1728, 1710, 1911, 1639,     FLUX 596
     . 1595,                                                           FLUX 597
     . 1534, 1515, 1461, 1403, 1329, 1314, 1291, 1331, 1354, 1352,     FLUX 598
     . 1326, 1293, 1315, 1313, 1335, 1276, 1405, 1480, 1413, 1391,     FLUX 599
     . 1338, 1310, 1323, 1342, 1353, 1352, 1341, 1296, 1282, 1286,     FLUX 600
     . 1402, 1499, 1496, 1479, 1466, 1429, 1402, 1418, 1456, 1480,     FLUX 601
     . 1436, 1390, 1315, 1343, 1341, 1270, 1302, 1339, 1419, 1447,     FLUX 602
     . 1471, 1473, 1463, 1540, 1553, 1540, 1535, 1507, 1472, 1442,     FLUX 603
     . 1396/                                                           FLUX 604
C---------- SOLAR FLUX FOR 690101 TO 690430                            FLUX 605
      DATA SFLU13/                                                     FLUX 606
     . 1399, 1421, 1441, 1545, 1625, 1789, 1830, 1831, 1833, 1693,     FLUX 607
     . 1685, 1631, 1572, 1533, 1541, 1528, 1491, 1443, 1316, 1280,     FLUX 608
     . 1319, 1339, 1248, 1308, 1341, 1422, 1315, 1296, 1292, 1261,     FLUX 609
     . 1264,                                                           FLUX 610
     . 1298, 1390, 1391, 1378, 1389, 1385, 1403, 1347, 1333, 1296,     FLUX 611
     . 1299, 1246, 1263, 1250, 1248, 1290, 1343, 1304, 1500, 1591,     FLUX 612
     . 1692, 1847, 2014, 2058, 2071, 1947, 1971, 1305,                 FLUX 613
     . 1656, 1520, 1478, 1439, 1359, 1394, 1381, 1373, 1418, 1378,     FLUX 614
     . 1360, 1385, 1340, 1390, 1619, 1749, 2001, 2109, 2085, 2137,     FLUX 615
```

```
     . 2308, 2225, 2058, 1954, 1811, 1912, 1777, 1775, 1819, 1829,        FLUX 616
     . 1855,                                                              FLUX 617
     . 1887, 1914, 1898, 1773, 1757, 1626, 1542, 1476, 1438, 1489,        FLUX 618
     . 1508, 1555, 1725, 1794, 1812, 1671, 1547, 1468, 1464, 1493,        FLUX 619
     . 1572, 1478, 1441, 1447, 1477, 1437, 1350, 1349, 1282, 1291/        FLUX 620
C---------- SOLAR FLUX FOR 690501 TO 690831                               FLUX 621
      DATA SFLU14/                                                        FLUX 622
     . 1250, 1278, 1319, 1355, 1541, 1382, 1298, 1385, 1354, 1377,        FLUX 623
     . 1501, 1603, 1541, 1577, 1588, 1594, 1641, 1624, 1576, 1590,        FLUX 624
     . 1707, 1774, 1728, 1714, 1685, 1637, 1535, 1432, 1226, 1167,        FLUX 625
     . 1119,                                                              FLUX 626
     . 1133, 1212, 1334, 1590, 1778, 1964, 2222, 2384, 2358, 2435,        FLUX 627
     . 2466, 2367, 2287, 2235, 2005, 1747, 1572, 1539, 1470, 1449,        FLUX 628
     . 1397, 1331, 1316, 1289, 1202, 1172, 1142, 1183, 1238, 1368,        FLUX 629
     . 1492, 1621, 1675, 1692, 1666, 1652, 1667, 1652, 1658, 1644,        FLUX 630
     . 1581, 1514, 1451, 1343, 1265, 1261, 1242, 1199, 1181, 1154,        FLUX 631
     . 1157, 1151, 1165, 1157, 1178, 1192, 1263, 1351, 1371, 1470,        FLUX 632
     . 1670,                                                              FLUX 633
     . 1757, 1925, 1888, 1929, 1877, 1719, 1631, 1503, 1455, 1393,        FLUX 634
     . 1359, 1286, 1231, 1175, 1155, 1123, 1079, 1052, 1039, 1095,        FLUX 635
     . 1167, 1210, 1332, 1460, 1458, 1564, 1678, 1715, 1776, 1696,        FLUX 636
     . 1638/                                                              FLUX 637
C---------- SOLAR FLUX FOR 690901 TO 691231                               FLUX 638
      DATA SFLU15/                                                        FLUX 639
     . 1563, 1552, 1517, 1505, 1382, 1357, 1283, 1241, 1173, 1166,        FLUX 640
     . 1190, 1207, 1312, 1299, 1353, 1342, 1333, 1344, 1315, 1365,        FLUX 641
     . 1378, 1405, 1455, 1557, 1693, 1589, 1513, 1427, 1399, 1370,        FLUX 642
     . 1339, 1432, 1325, 1322, 1409, 1432, 1450, 1501, 1417, 1357,        FLUX 643
     . 1301, 1280, 1243, 1191, 1177, 1154, 1223, 1288, 1458, 1619,        FLUX 644
     . 1770, 1862, 1931, 2045, 2040, 2046, 2008, 1896, 1725, 1617,        FLUX 645
     . 1539,                                                              FLUX 646
     . 1394, 1389, 1267, 1256, 1300, 1319, 1358, 1304, 1277, 1228,        FLUX 647
     . 1220, 1223, 1266, 1341, 1379, 1455, 1601, 1772, 1852, 1989,        FLUX 648
     . 1982, 2053, 2095, 2007, 1857, 1753, 1728, 1568, 1415, 1365,        FLUX 649
     . 1302, 1315, 1317, 1389, 1333, 1234, 1207, 1121, 1161, 1148,        FLUX 650
     . 1179, 1223, 1296, 1327, 1359, 1369, 1441, 1441, 1472, 1542,        FLUX 651
     . 1603, 1630, 1563, 1529, 1536, 1510, 1469, 1490, 1534, 1579,        FLUX 652
     . 1518/                                                              FLUX 653
C---------- SOLAR FLUX FOR 700101 TO 700430                               FLUX 654
      DATA SFLU16/                                                        FLUX 655
     . 1509, 1445, 1371, 1315, 1265, 1176, 1130, 1133, 1196, 1345,        FLUX 656
     . 1601, 1716, 1729, 1718, 1800, 1743, 1806, 1794, 1753, 1684,        FLUX 657
     . 1770, 1605, 1534, 1436, 1495, 1564, 1630, 1620, 1637, 1546,        FLUX 658
     . 1434,                                                              FLUX 659
     . 1346, 1331, 1237, 1194, 1210, 1319, 1394, 1526, 1541, 1703,        FLUX 660
     . 1843, 2010, 1976, 1962, 2016, 1933, 1939, 1977, 1985, 1965,        FLUX 661
     . 1911, 1830, 1839, 1859, 1796, 1783, 1769, 1741,                    FLUX 662
     . 1757, 1730, 1673, 1653, 1698, 1678, 1684, 1728, 1644, 1593,        FLUX 663
     . 1563, 1544, 1524, 1473, 1449, 1390, 1337, 1335, 1302, 1411,        FLUX 664
     . 1497, 1509, 1649, 1699, 1672, 1664, 1592, 1538, 1531, 1494,        FLUX 665
     . 1493,                                                              FLUX 666
     . 1608, 1571, 1635, 1704, 1794, 1929, 1934, 2031, 2185, 2247,        FLUX 667
     . 2064, 1985, 1853, 1733, 1636, 1522, 1480, 1419, 1357, 1312,        FLUX 668
     . 1263, 1286, 1301, 1301, 1348, 1353, 1390, 1486, 1561, 1533/        FLUX 669
C---------- SOLAR FLUX FOR 700501 TO 700831                               FLUX 670
      DATA SFLU17/                                                        FLUX 671
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
     . 1595, 1607, 1635, 1623, 1679, 1633, 1624, 1567, 1549, 1628,      FLUX 672
     . 1762, 1803, 1931, 1969, 2063, 2060, 1970, 1946, 1974, 1839,      FLUX 673
     . 1749, 1703, 1637, 1626, 1606, 1628, 1552, 1538, 1585, 1637,      FLUX 674
     . 1652,                                                            FLUX 675
     . 1456, 1418, 1334, 1317, 1329, 1300, 1345, 1342, 1382, 1459,      FLUX 676
     . 1695, 1717, 2003, 2074, 2092, 1986, 1906, 1818, 1751, 1633,      FLUX 677
     . 1606, 1501, 1437, 1423, 1493, 1514, 1600, 1605, 1680, 1748,      FLUX 678
     . 1799, 1890, 1946, 1950, 1910, 1864, 1663, 1604, 1484, 1438,      FLUX 679
     . 1377, 1354, 1307, 1263, 1219, 1214, 1259, 1310, 1358, 1364,      FLUX 680
     . 1506, 1538, 1587, 1640, 1690, 1763, 1359, 1821, 1610, 1561,      FLUX 681
     . 1415,                                                            FLUX 682
     . 1340, 1285, 1279, 1274, 1284, 1280, 1282, 1318, 1383, 1373,      FLUX 683
     . 1272, 1369, 1497, 1670, 1691, 1625, 1516, 1490, 1454, 1486,      FLUX 684
     . 1428, 1417, 1462, 1402, 1365, 1357, 1372, 1416, 1487, 1490,      FLUX 685
     . 1545/                                                            FLUX 686
C---------- SOLAR FLUX FOR 700901 TO 701231                             FLUX 687
      DATA SFLU18/                                                      FLUX 688
     . 1541, 1567, 1632, 1634, 1623, 1615, 1644, 1569, 1572, 1439,      FLUX 689
     . 1344, 1261, 1201, 1219, 1180, 1184, 1169, 1207, 1347, 1469,      FLUX 690
     . 1494, 1552, 1616, 1560, 1558, 1559, 1450, 1445, 1387, 1347,      FLUX 691
     . 1287, 1254, 1260, 1373, 1345, 1280, 1333, 1420, 1449, 1476,      FLUX 692
     . 1428, 1354, 1318, 1349, 1402, 1340, 1315, 1385, 1390, 1372,      FLUX 693
     . 1365, 1350, 1490, 1612, 1686, 1871, 1918, 1903, 1890, 1752,      FLUX 694
     . 1686,                                                            FLUX 695
     . 1633, 1591, 1532, 1483, 1475, 1498, 1506, 1411, 1419, 1478,      FLUX 696
     . 1556, 1551, 1791, 1864, 2000, 1935, 1841, 1931, 1855, 1737,      FLUX 697
     . 1633, 1542, 1424, 1339, 1279, 1293, 1351, 1489, 1484, 1525,      FLUX 698
     . 1464, 1484, 1481, 1540, 1619, 1675, 1665, 1673, 1715, 1709,      FLUX 699
     . 1726, 1784, 1643, 1546, 1541, 1467, 1468, 1516, 1524, 1556,      FLUX 700
     . 1463, 1368, 1321, 1244, 1240, 1223, 1198, 1173, 1234, 1290,      FLUX 701
     . 1333/                                                            FLUX 702
C---------- SOLAR FLUX FOR 710101 TO 710430                             FLUX 703
      DATA SFLU19/                                                      FLUX 704
     . 1305, 1348, 1347, 1402, 1461, 1469, 1502, 1491, 1528, 1521,      FLUX 705
     . 1490, 1505, 1481, 1539, 1537, 1564, 1553, 1693, 1658, 1692,      FLUX 706
     . 1787, 1806, 1829, 1770, 1686, 1649, 1667, 1662, 1607, 1630,      FLUX 707
     . 1701,                                                            FLUX 708
     . 1656, 1605, 1547, 1486, 1447, 1437, 1334, 1232, 1104, 1095,      FLUX 709
     . 1097, 1105, 1132, 1143, 1171, 1296, 1326, 1304, 1385, 1362,      FLUX 710
     . 1361, 1435, 1432, 1475, 1464, 1432, 1400, 1366,                  FLUX 711
     . 1292, 1258, 1217, 1160, 1137, 1091, 1052, 1040, 1045, 1070,      FLUX 712
     . 1095, 1153, 1125, 1117, 1103, 1145, 1151, 1142, 1091, 1063,      FLUX 713
     . 1126, 1124, 1082, 1078, 1083, 1078, 1035, 1062, 1034, 1009,      FLUX 714
     . 1029,                                                            FLUX 715
     . 1005, 1076, 1090, 1077, 1083, 1073, 1132, 1101, 1089, 1098,      FLUX 716
     . 1241, 1297, 1396, 1410, 1400, 1396, 1399, 1372, 1384, 1368,      FLUX 717
     . 1289, 1205, 1175, 1091, 1060, 1048, 1017,  975,  958,  948/      FLUX 718
C---------- SOLAR FLUX FOR 710501 TO 710831                             FLUX 719
      DATA SFLU20/                                                      FLUX 720
     .  953,  996, 1062, 1126, 1179, 1256, 1311, 1327, 1383, 1358,      FLUX 721
     . 1353, 1302, 1250, 1223, 1180, 1161, 1128, 1112, 1098, 1060,      FLUX 722
     . 1020,  934,  897,  882,  923,  937, 1039, 1070, 1085, 1103,      FLUX 723
     . 1116,                                                            FLUX 724
     . 1101, 1083, 1078, 1073, 1062, 1050, 1015, 1018, 1009,  975,      FLUX 725
     .  960,  933,  937,  942,  913,  908,  931,  954,  946,  942,      FLUX 726
     .  952,  984,  949, 1053, 1101, 1159, 1225, 1311, 1411, 1488,      FLUX 727
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
     . 1451, 1394, 1305, 1347, 1346, 1270, 1225, 1167, 1077, 1085,        FLUX 728
     . 1087, 1096, 1112, 1139, 1256, 1253, 1215, 1210, 1269, 1188,        FLUX 729
     . 1136, 1157, 1213, 1345, 1214, 1223, 1190, 1167, 1143, 1163,        FLUX 730
     . 1132,                                                              FLUX 731
     . 1097, 1092, 1064, 1064, 1069, 1050, 1026, 1086, 1121, 1097,        FLUX 732
     . 1047, 1021, 1019, 1007,  973,  988, 1036, 1105, 1262, 1411,        FLUX 733
     . 1503, 1512, 1486, 1576, 1435, 1388, 1353, 1236, 1120, 1033,        FLUX 734
     .  953/                                                              FLUX 735
C---------- SOLAR FLUX FOR 710901 TO 711231                               FLUX 736
      DATA SFLU21/                                                        FLUX 737
     .  903,  901,  925,  926,  952, 1018, 1073, 1035,  983,  918,        FLUX 738
     .  893,  916,  989, 1081, 1135, 1125, 1165, 1166, 1131, 1103,        FLUX 739
     . 1092, 1103, 1081, 1063, 1092, 1119, 1166, 1156, 1148, 1169,        FLUX 740
     . 1146, 1142, 1134, 1091, 1061, 1024,  983,  981,  995,  954,        FLUX 741
     .  945,  949,  910,  859,  898,  893,  947,  991, 1098, 1156,        FLUX 742
     . 1193, 1222, 1239, 1257, 1243, 1193, 1161, 1105, 1051, 1075,        FLUX 743
     . 1062,                                                              FLUX 744
     . 1095, 1117, 1152, 1128, 1160, 1095, 1046, 1007, 1016, 1027,        FLUX 745
     . 1045, 1026, 1033, 1031, 1011, 1047, 1054, 1041, 1076, 1130,        FLUX 746
     . 1149, 1188, 1204, 1216, 1217, 1283, 1281, 1256, 1172, 1141,        FLUX 747
     . 1169, 1094, 1097, 1062, 1117, 1118, 1130, 1147, 1212, 1224,        FLUX 748
     . 1222, 1240, 1187, 1185, 1173, 1176, 1312, 1346, 1344, 1352,        FLUX 749
     . 1357, 1392, 1329, 1294, 1268, 1202, 1202, 1132, 1127, 1124,        FLUX 750
     . 1068/                                                              FLUX 751
C---------- SOLAR FLUX FOR 720101 TO 720430                               FLUX 752
      DATA SFLU22/                                                        FLUX 753
     . 1047, 1031,  996,  979,  991, 1031,  995,  998,  943,  923,        FLUX 754
     .  958,  996,  990, 1059, 1154, 1135, 1179, 1109, 1159, 1230,        FLUX 755
     . 1279, 1232, 1363, 1372, 1318, 1238, 1189, 1195, 1161, 1129,        FLUX 756
     . 1119,                                                              FLUX 757
     . 1067, 1065, 1048, 1035, 1026,  986, 1022, 1043, 1063, 1136,        FLUX 758
     . 1184, 1239, 1263, 1367, 1458, 1532, 1682, 1844, 1906, 2025,        FLUX 759
     . 1896, 1804, 1756, 1642, 1519, 1493, 1403, 1332, 1305,              FLUX 760
     . 1255, 1297, 1311, 1371, 1402, 1437, 1433, 1416, 1374, 1325,        FLUX 761
     . 1349, 1285, 1293, 1297, 1353, 1330, 1339, 1319, 1310, 1352,        FLUX 762
     . 1358, 1312, 1400, 1265, 1164, 1105, 1041, 1011,  965,  983,        FLUX 763
     .  957,                                                              FLUX 764
     .  960,  973,  981, 1010, 1053, 1181, 1213, 1238, 1305, 1303,        FLUX 765
     . 1298, 1283, 1284, 1280, 1260, 1243, 1203, 1157, 1154, 1113,        FLUX 766
     . 1069, 1094, 1084, 1102, 1116, 1099, 1090, 1032,  986,  970/        FLUX 767
C---------- SOLAR FLUX FOR 720501 TO 720831                               FLUX 768
      DATA SFLU23/                                                        FLUX 769
     .  950,  949,  977, 1027, 1087, 1176, 1209, 1288, 1330, 1393,        FLUX 770
     . 1408, 1436, 1616, 1628, 1626, 1618, 1593, 1655, 1567, 1523,        FLUX 771
     . 1482, 1433, 1352, 1355, 1273, 1163, 1143, 1108, 1162, 1168,        FLUX 772
     . 1225,                                                              FLUX 773
     . 1282, 1300, 1375, 1495, 1648, 1591, 1527, 1502, 1491, 1395,        FLUX 774
     . 1432, 1497, 1389, 1365, 1395, 1427, 1495, 1490, 1432, 1425,        FLUX 775
     . 1390, 1351, 1307, 1285, 1255, 1208, 1219, 1273, 1324, 1345,        FLUX 776
     . 1330, 1370, 1399, 1467, 1502, 1461, 1432, 1367, 1271, 1217,        FLUX 777
     . 1236, 1193, 1199, 1195, 1176, 1151, 1146, 1089, 1099, 1109,        FLUX 778
     . 1117, 1116, 1096, 1095, 1124, 1204, 1272, 1333, 1391, 1412,        FLUX 779
     . 1502,                                                              FLUX 780
     . 31*0/                                                              FLUX 781
C---------- SOLAR FLUX FOR 720901 TO 721231                               FLUX 782
      DATA SFLU24/                                                        FLUX 783
```

```
     .    122*0/
C----------- SOLAR FLUX FOR 730101 TO 730430
      DATA SFLU25/
     .    120*0/
C----------- SOLAR FLUX FOR 730501 TO 730831
      DATA SFLU26/
     .    123*0/
C----------- SOLAR FLUX FOR 730901 TO 731231
      DATA SFLU27/
     .    122*0/
      CALL ADFLUX(NARCS,MFLUX,NFLUXS,SFLUX,BEGYMD,ENDYMD)
C     DATE OF LAST SOLAR FLUX VALUE      720731
      RETURN
      END
```

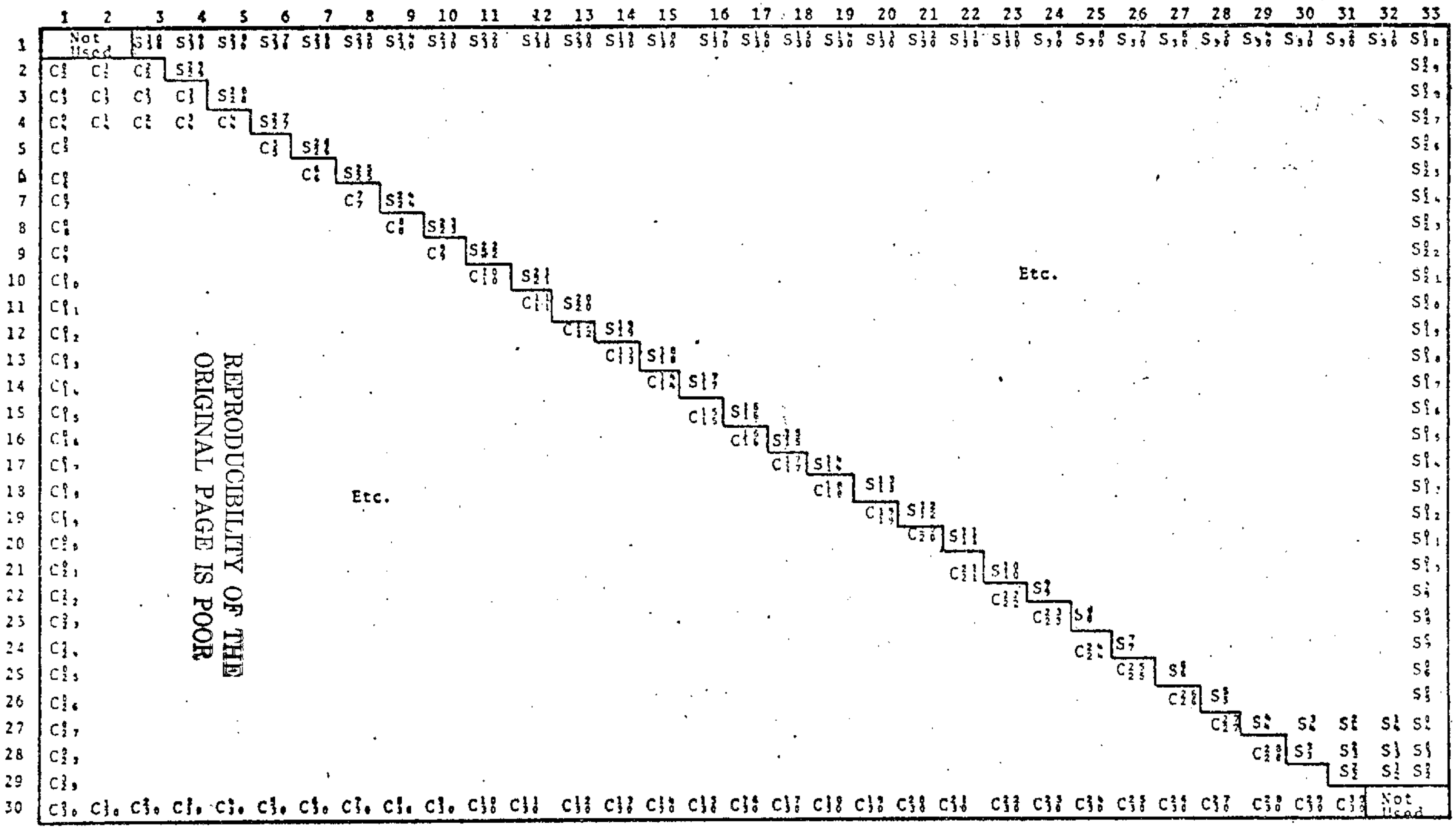
FMODEL

DESCRIPTION

FMODEL is a BLOCK DATA routines which contains the geopotential model information.

The storage of the spherical harmonic coefficients, C_{nm} and S_{nm}, is of particular note. Because for physical reasons $m \leq n$, the set of C_{nm} and the set of S_{nm} each require only half a column more than half the matrix. To conserve storage, both sets of coefficients are stored in the same matrix. This storage algorithm is illustrated in Figure 1 for the case of 30 x 30 model used in GEODYN.

The matrix CS (30,33) containing the coefficients C_{nm} and S_{nm}



	Index	Matrix Subscript Computation
C coefficients	n	N
	m	M+1
S coefficients	n	31-N
	m	33-M

NAME FMODEL

PURPOSE BLOCK DATA STORAGE OF THE COEFFICIENTS OF THE SPHERICAL HARMONIC EXPANSION OF THE GEOPOTENTIAL

COMMON BLOCKS FMODEL

REFERENCES 'GEODYN SYSTEMS DESCRIPTION'
VOLUME 1 - GEODYN DOCUMENTATION

```
      BLOCK DATA
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/FMODEL/INDEX1,INDEX2,INDEX3,INDEX4,CS01(30),CS02(30),CS03(3
     .0),CS04(30),
     .CS05(30),CS06(30),CS07(30),CS08(30),CS09(30),CS10(30),CS11(30),
     .CS12(30),CS13(30),CS14(30),CS15(30),CS16(30),CS17(30),CS18(30),
     .CS19(30),CS20(30),CS21(30),CS22(30),CS23(30),CS24(30),CS25(30),
     .CS26(30),CS27(30),CS28(30),CS29(30),CS30(30),CS31(30),CS32(30),
     .CS33(30),MODEL(8)
      REAL*8 MODEL
      DATA MODEL/8HSAO 1969,8H STANDAR,8HD EARTH ,8H
     .      8H      ,8H      ,8H      ,8H      /
      DATA INDEX1,INDEX2,INDEX3,INDEX4/23,4,23,4/
      DATA CS01/
     .0.0         ,-.1082630D-02,0.253200D-05,0.159300D-05,0.230000D-06,
     .-.502000D-06,0.3620000D-06,0.118000D-06,0.100000D-05,0.354000D-06,
     .-.202000D-06,0.4200000D-07,0.123000D-06,0.730000D-07,0.174000D-06,
     .-.187000D-06,-.8499995D-07,0.231000D-06,0.215000D-06,0.500000D-08,
     .-.144000D-06,0.0         ,0.0         ,0.0         ,0.0         ,
     .0.0         ,0.0         ,0.0         ,0.0         ,0.0         /
      DATA CS02/
     .0.0         ,0.0         ,0.212763D-05,-.502693D-06,-.460853D-07,
     .-.778801D-07,0.1767010D-05,0.214773D-07,0.898199D-07,0.695215D-07,
     .0.259153D-08,-.2601690D-07,-.305263D-07,-.121347D-07,-.182829D-08,
     .-.117163D-07,0.0         ,0.0         ,0.0         ,0.0         ,
     .0.0         ,0.0         ,0.0         ,0.0         ,0.0         ,
     .0.0         ,0.0         ,0.0         ,0.0         ,0.0         /
      DATA CS03/
     .0.0         ,0.1557520D-05,0.304690D-06,0.730439D-07,0.991820D-07,
     .0.662041D-08,0.2819340D-07,0.395567D-09,0.462298D-09,-.135660D-08,
     .0.253180D-08,0.1253700D-08,-.192671D-08,0.117044D-08,-.147706D-08,
     .0.635345D-09,0.0         ,0.0         ,0.0         ,0.0         ,
     .0.0         ,0.0         ,0.0         ,0.0         ,0.0         ,
     .0.0         ,0.0         ,0.0         ,0.0         ,0.0         /
      DATA CS04/
     .0.0         ,0.0         ,0.956999D-07,0.591297D-07,-.142322D-07,
     .0.577915D-09,0.2857320D-08,-.560760D-09,-.729038D-09,-.135117D-09,
     .-.201726D-09,0.2174830D-09,0.729369D-10,0.495814D-10,0.178795D-10,
     .-.870479D-10,0.0         ,0.0         ,0.0         ,0.0         ,
     .0.0         ,0.0         ,0.0         ,0.0         ,0.0         ,
     .0.0         ,0.0         ,0.0         ,0.0         ,0.0         /
      DATA CS05/
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
.0.0         ,0.0          ,0.0          ,-.168330D-08,-.207839D-08,FMOD  56
.-.152714D-11,-.4185089D-09,-.200713D-09,0.428842D-10,-.283791D-10,FMOD  57
.-.127131D-10,-.1354910D-10,-.468964D-11,0.141454D-11,0.199491D-11,FMOD  58
.-.132129D-11,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  59
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  60
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          /FMOD  61
 DATA CS06/                                                           FMOD  62
.0.0         ,0.0          ,0.0          ,0.0          ,0.3100690D-09,FMOD  63
.-.170635D-09,-.3079970D-12,-.102686D-10,-.628361D-12,-.4956600D-11,FMOD  64
.0.129397D-11,0.6221820D-12,0.178209D-11,-.341444D-12,0.3046880D-12,FMOD  65
.-.317402D-12,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  66
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  67
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          /FMOD  68
 DATA CS07/                                                           FMOD  69
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  70
.0.205637D-10,-.1794029D-10,-.150564D-11,0.319361D-12,-.225484D-12,FMOD  71
.0.147796D-12,-.5659740D-13,-.124773D-12,0.187652D-13,0.464651D-13,FMOD  72
.-.265757D-13,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  73
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  74
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          /FMOD  75
 DATA CS08/                                                           FMOD  76
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  77
.0.0         ,0.2952499D-11,0.175356D-12,-.961519D-13,0.462616D-13,FMOD  78
.0.189873D-13,0.3221650D-14,-.660104D-14,0.836665D-15,0.559127D-14,FMOD  79
.0.322361D-14,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  80
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  81
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          /FMOD  82
 DATA CS09/                                                           FMOD  83
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  84
.0.0         ,0.0          ,-.936207D-13,0.763504D-13,0.547126D-14,FMOD  85
.0.307451D-14,-.1284880D-15,-.470290D-15,-.184524D-15,-.339521D-15,FMOD  86
.-.175492D-15,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  87
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  88
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          /FMOD  89
 DATA CS10/                                                           FMOD  90
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  91
.0.0         ,0.0          ,0.0          ,-.635510D-14,-.127399D-15,FMOD  92
.0.722552D-15,-.7810450D-16,-.275137D-16,0.257042D-16,0.591825D-17,FMOD  93
.0.131799D-17,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  94
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  95
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          /FMOD  96
 DATA CS11/                                                           FMOD  97
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD  98
.0.0         ,0.0          ,0.0          ,0.0          ,0.514254D-15,FMOD  99
.-.111359D-15,-.5544930D-17,0.969366D-17,0.254507D-17,-.452211D-18,FMOD 100
.-.324010D-18,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 101
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 102
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 103
 DATA CS12/                                                           FMOD 104
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 105
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 106
.0.235410D-16,-.1975550D-17,-.445266D-18,0.131834D-18,-.625906D-19,FMOD 107
.0.564217D-20,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 108
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 109
.0.0         ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 110
 DATA CS13/                                                           FMOD 111
```

```
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 112
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 113
,0.0          ,-.1742170D-18,-.246832D-20,0.669374D-20,0.194518D-20,FMOD 114
,0.164555D-20,0.2561270D-20,0.165292D-21,0.454057D-21,0.0          ,FMOD 115
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 116
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 117
 DATA CS14/                                                          FMOD 118
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 119
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 120
,0.0          ,0.0          ,-.257198D-19,0.376257D-20,-.753195D-21,FMOD 121
,0.237411D-21,0.8241809D-22,0.487412D-23,0.153383D-22,0.127350D-22,FMOD 122
,0.400205D-24,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 123
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 124
 DATA CS15/                                                          FMOD 125
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 126
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 127
,0.0          ,0.0          ,0.0          ,-.713342D-21,0.322877D-22,FMOD 128
,-.519129D-23,-.3629150D-23,-.225483D-23,-.130602D-24,0.157393D-24,FMOD 129
,0.337218D-24,-.2515790D-25,0.0          ,0.0          ,0.0          ,FMOD 130
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 131
 DATA CS16/                                                          FMOD 132
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 133
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 134
,0.0          ,0.0          ,0.0          ,0.0          ,0.701751D-24,FMOD 135
,-.317911D-23,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 136
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 137
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 138
 DATA CS17/                                                          FMOD 139
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 140
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 141
,0.0          ,0.0          ,0.0          ,0.0          ,0.135546D-24,FMOD 142
,-.467554D-24,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 143
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 144
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 145
 DATA CS18/                                                          FMOD 146
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 147
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 148
,0.0          ,0.0          ,0.0          ,0.0          ,0.753667D-25,FMOD 149
,-.715626D-23,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 150
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 151
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 152
 DATA CS19/                                                          FMOD 153
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 154
,0.0          ,0.0          ,0.0          ,0.825801D-25,0.199486D-26,FMOD 155
,-.226649D-24,-.7919560D-24,-.397434D-23,0.616671D-23,-.159612D-22,FMOD 156
,-.709307D-22,-.1770960D-21,0.0          ,0.0          ,0.0          ,FMOD 157
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 158
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 159
 DATA CS20/                                                          FMOD 160
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 161
,0.0          ,0.0          ,0.0          ,0.0          ,-.176520D-23,FMOD 162
,0.726916D-23,-.2916410D-22,-.369401D-22,0.100015D-23,0.139418D-21,FMOD 163
,0.811700D-22,0.3317300D-20,0.273134D-19,0.0          ,0.0          ,FMOD 164
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 165
,0.0          ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 166
 DATA CS21/                                                          FMOD 167
```

```
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 168
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 169
,0.0           ,-.5712840D-22,0.120077D-21,0.109186D-21,-.245695D-20,FMOD 170
,0.127028D-20,-.3073530D-19,0.153357D-18,-.516611D-18,0.0             ,FMOD 171
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 172
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           /FMOD 173
 DATA CS22/                                                               FMOD 174
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 175
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 176
,0.0           ,0.0           ,0.0           ,0.0           ,-.503517D-19,FMOD 177
,0.180659D-18,-.3866820D-18,0.772359D-18,-.212001D-17,-.823118D-17,FMOD 178
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 179
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           /FMOD 180
 DATA CS23/                                                               FMOD 181
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 182
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 183
,0.0           ,0.0           ,0.0           ,0.0           ,-.569527D-20,FMOD 184
,0.117661D-19,-.2044699D-17,-.117861D-17,0.144311D-17,-.179335D-16,FMOD 185
,-.107550D-15,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 186
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           /FMOD 187
 DATA CS24/                                                               FMOD 188
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 189
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 190
,0.0           ,0.0           ,0.0           ,0.0           ,-.155638D-16,FMOD 191
,0.714333D-17,0.4791489D-16,0.115615D-15,0.103853D-15,-.511562D-16,FMOD 192
,0.116130D-15,0.7112868D-14,0.0           ,0.0           ,0.0           ,FMOD 193
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           /FMOD 194
 DATA CS25/                                                               FMOD 195
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 196
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 197
,0.0           ,0.0           ,0.0           ,0.0           ,-.511060D-17,FMOD 198
,-.124152D-15,-.3708268D-15,-.227841D-15,0.749597D-15,-.809426D-15,FMOD 199
,-.912577D-14,0.1370900D-13,0.861829D-13,0.0           ,0.0           ,FMOD 200
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           /FMOD 201
 DATA CS26/                                                               FMOD 202
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 203
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 204
,0.0           ,0.0           ,0.0           ,0.0           ,0.246566D-15,FMOD 205
,0.201326D-14,0.6363828D-15,-.411635D-15,0.221610D-13,-.575625D-13,FMOD 206
,0.269375D-13,-.2420470D-12,0.454672D-12,-.125606D-11,0.0           ,FMOD 207
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           /FMOD 208
 DATA CS27/                                                               FMOD 209
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 210
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 211
,0.0           ,0.0           ,0.0           ,0.0           ,-.197608D-13,FMOD 212
,-.767624D-13,-.5757119D-13,0.894201D-13,-.158330D-14,0.271842D-13,FMOD 213
,-.139681D-11,0.2908659D-11,0.724190D-11,0.708604D-11,-.174166D-10,FMOD 214
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           /FMOD 215
 DATA CS28/                                                               FMOD 216
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 217
,0.0           ,0.0           ,0.0           ,0.0           ,0.0           ,FMOD 218
,0.0           ,0.0           ,0.0           ,0.0           ,0.231440D-12,FMOD 219
,0.175500D-13,-.3070600D-12,-.607460D-12,0.113499D-11,0.525778D-11,FMOD 220
,-.886408D-11,0.3431620D-12,0.111560D-10,0.348475D-10,-.421805D-09,FMOD 221
,-.147513D-08,0.0           ,0.0           ,0.0           ,0.0           /FMOD 222
 DATA CS29/                                                               FMOD 223
```

```
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 224
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 225
.0.0          ,0.0          ,0.0          ,0.0          ,0.490181D-11,FMOD 226
.0.550577D-11,-.6804069D-11,0.133859D-10,-.691034D-11,0.223707D-10,FMOD 227
.-.254714D-10,0.5569749D-10,0.982234D-10,-.241870D-09,-.152698D-08,FMOD 228
.0.546339D-09,0.7168588D-08,0.0          ,0.0          ,0.0          /FMOD 229
 DATA CS30/                                                             FMOD 230
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 231
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 232
.0.0          ,0.0          ,0.0          ,0.0          ,0.599398D-10,FMOD 233
.-.123233D-10,0.3040570D-11,-.885354D-10,0.204056D-09,-.504650D-09,FMOD 234
.-.324246D-09,-.8930923D-09,0.172850D-09,-.330235D-08,0.325271D-09,FMOD 235
.-.266235D-08,-.5243293D-08,0.199460D-06,0.0          ,0.0          /FMOD 236
 DATA CS31/                                                             FMOD 237
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 238
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 239
.0.0          ,0.0          ,0.0          ,0.0          ,0.919401D-09,FMOD 240
.-.523980D-09,-.1650310D-08,0.705093D-09,0.346654D-08,-.476299D-08,FMOD 241
.-.621344D-08,-.5551427D-08,0.691077D-08,0.155828D-07,-.437569D-07,FMOD 242
.-.567625D-07,0.1579400D-06,-.216784D-06,-.830522D-06,0.0          /FMOD 243
 DATA CS32/                                                             FMOD 244
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 245
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 246
.0.0          ,0.0          ,0.0          ,0.0          ,0.375404D-07,FMOD 247
.0.204023D-07,0.2510030D-07,0.143192D-07,-.175503D-07,0.175628D-07,FMOD 248
.-.628233D-07,-.1046160D-07,0.176579D-07,0.846617D-07,0.296244D-07,FMOD 249
.-.836407D-07,-.4526250D-06,0.280954D-06,0.0          ,0.0          /FMOD 250
 DATA CS33/                                                             FMOD 251
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 252
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 253
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 254
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 255
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          ,FMOD 256
.0.0          ,0.0          ,0.0          ,0.0          ,0.0          /FMOD 257
 END                                                                    FMOD 258
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME	GEODYN
PURPOSE	BLOCK DATA TO STORE DATE AND SOURCE TAPE NUMBER OF THIS VERSION OF GEODYN
COMMON BLOCKS	GEODYN

```
BLOCK DATA                                                                 GEOD  12
IMPLICIT REAL*8 (A-H,O-Z)                                                  GEOD  13
COMMON/GEODYN/DATE(5)                                                      GEOD  14
DATA DATE/'  GEODYN','(720628)',' SOURCE ','TAPE    ','32607D  '/ GEOD  15
END                                                                        GEOD  16
```

GEOIDH

DESCRIPTION

This subroutine performs two functions critical to the proper evaluation and estimation of surface density potentials.

1. GEOIDH alters the nominal Cartesian coordinates of surface density sub-blocks in such a manner so as to place these sub-blocks on the equipotential surface defined by the Earth's average global potential as computed by the function AVGPOT.

2. Also computed are the coefficients interrelating all surface density values such that adjusted surface density values are constrained not to conflict with the potential defined by evaluation of the spherical harmonic expansion below some specified maximum degree and order.

The GEODYN Systems Description, Volume I provides a thorough description of the physical and mathematic function performed by this subroutine.

NAME GEOIDH

PURPOSE TO POSITION SURFACE DENSITY LOCATIONS ON GEOID SURFACE AND COMPUTE MATRIX OF CONSTRAINT FOR DENSITY ADJUSTMENT

CALLING SEQUENCE CALL GEOIDH(AREA,CENTER,DENCON)

SYMBOL	TYPE	DESCRIPTION
AREA (1)	DP	INPUT & OUTPUT - SURFACE DENSITY SUB-BLOCK AREAS
CENTER (3,1)	DP	INPUT & OUTPUT - THE GEOCENTRIC COORDINATES OF THE SUB-BLOCK CENTERS
DENCON (NCONST,1)	DP	OUTPUT - COEFFICIENTS RELATING CONSTRAINED AND UNCONSTRAINED ADJUSTED SURFACE DENSITIES

SUBROUTINES USED CLEAR EGRAV DNVERT AVGPOT

COMMON BLOCKS CSLIM CPARAM FMODEL INITBK INTBLK XYZ VRBLOK

INPUT FILES NONE

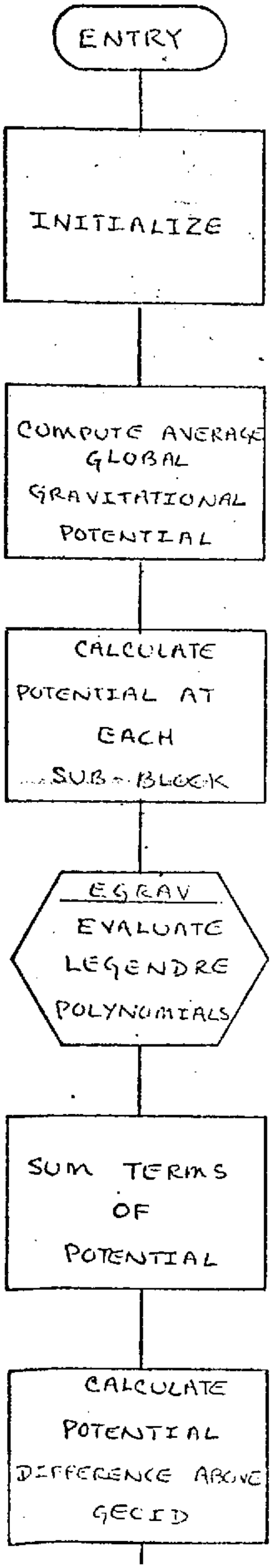
OUTPUT FILES NONE

REFERENCES 'GEODYN SYSTEMS DESCRIPTION'
VOLUME 1 - GEODYN DOCUMENTATION

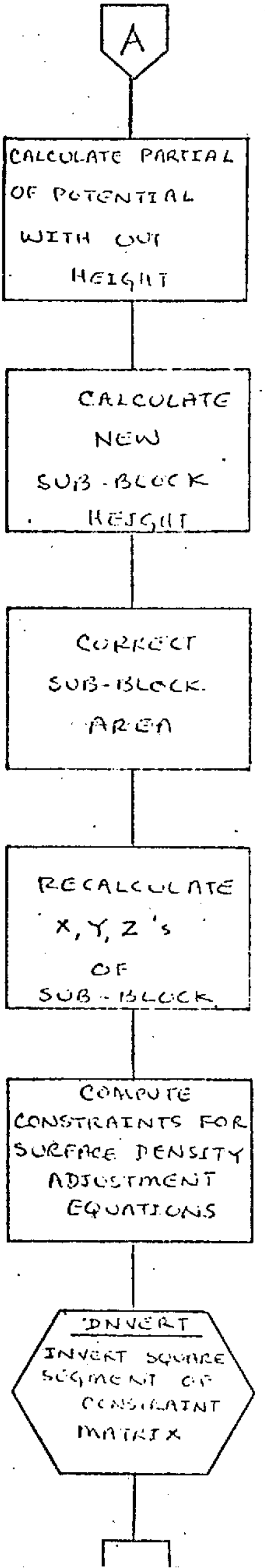
```
      SUBROUTINE GEOIDH(AREA,CENTER,DENCON)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL CMPGPR
      INTEGER ULIMIT
      DOUBLE PRECISION MODEL
      DIMENSION D(1),AREA(1),CENTER(3,1),FCT(3),DENCON(NCONST,1),INDS(4)
      COMMON/CSLIM/LLIMIT(31),ULIMIT(31)
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,NGPCOM,
     *   NCSEST,CMPGPR,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,INNRSW,
     *   NCONST,NOCCNS
      COMMON/FMODEL/INDEX1,INDEX2,INDEX3,INDEX4,CS(30,33),MODEL(8)
      COMMON/INITBK/IGI(50),NOTIST,NSWI(6)
      COMMON/INTBLK/THDOT1,THDOT2,THDT2S,GM,AE,AESQ,DUM(60)
      COMMON/XYZ/XYZ(6),R,RSQ,ISAT,IFORCE(2)
      COMMON/VRBLOK/XYSQ,COSLAM(31),SINLAM(31),PR,PPSI,PLAMDA,
     *   P(33,30),ACRN(30),TPSIM(39)
      EQUIVALENCE (D(1),SINPSI,P(1,1)),(COSPSI,P(2,1))
C INITIALIZE
      NOTIST=0
      THETG=0.0D0
```

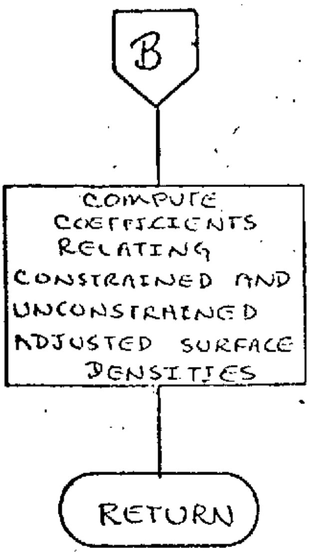
```
      INDEX2=INDEX1-1                                                   GEOI  56
      NCENTR=4*NDEN                                                     GEOI  57
C COMPUTE AVERAGE GLOBAL GRAVITATIONAL POTENTIAL                         GEOI  58
      REFPOT=AVGPOT(45)                                                 GEOI  59
      CALL CLEAR(DENCON,2*NCONST,NDEN)                                  GEOI  60
C CALCULATE POTENTIAL  AT EACH SUB-BLOCK                                GEOI  61
      DO 1000 L=1,NCENTR                                                GEOI  62
      XYZ(1)=CENTER(1,L)                                                GEOI  63
      XYZ(2)=CENTER(2,L)                                                GEOI  64
      XYZ(3)=CENTER(3,L)                                                GEOI  65
C EVALUATE LEGENDRE POLYNOMIALS                                         GEOI  66
      CALL EGRAV(THETG,RASAT,FCT)                                       GEOI  67
      THDT2R=R*(THDT2S*COSPSI)**2                                       GEOI  68
      SUM0=0.0D0                                                        GEOI  69
      SUM=0.0D0                                                         GEOI  70
      FN=1.0D0                                                          GEOI  71
C SUM TERMS OF POTENTIAL                                                GEOI  72
      DO 100 NC=2,INDEX2                                                GEOI  73
      NS=31-NC                                                          GEOI  74
      FN=FN+1.0D0                                                       GEOI  75
      FR=ACRN(NC)                                                       GEOI  76
      NUL=MAX0(ULIMIT(NC),1)                                            GEOI  77
      DO 50 MC=1,NUL                                                    GEOI  78
      MS=34-MC                                                          GEOI  79
      F1=FR*P(MC,NC)                                                    GEOI  80
      F=FN*F1                                                           GEOI  81
      CCSS=CS(NC,MC)*COSLAM(MC)+CS(NS,MS)*SINLAM(MC)                    GEOI  82
      SUM0=SUM0+F1*CCSS                                                 GEOI  83
   50 SUM=SUM+F*CCSS                                                    GEOI  84
  100 CONTINUE                                                          GEOI  85
C CALCULATE POTENTIAL DIFFERENCE ABOVE GEOID                            GEOI  86
      U0=GM/R+SUM0                                                      GEOI  87
      POTDIF=0.5D0*R*THDT2R+U0-REFPOT                                   GEOI  88
C CALCULATE PARTIAL OF POTENTIAL WITH RESPECT TO HEIGHT                 GEOI  89
      PARPOT=-THDT2R+(U0+SUM)/R                                         GEOI  90
C CALCULATE NEW SUB-BLOCK HEIGHT                                        GEOI  91
      DR=POTDIF/PARPOT                                                  GEOI  92
      RNEW=R+DR                                                         GEOI  93
C CORRECT SUB-BLOCK AREA                                                GEOI  94
      AREA(L)=AREA(L)*(RNEW/R)**2                                       GEOI  95
      RCL=RNEW*COSPSI                                                   GEOI  96
C RECALCULATE X,Y,Z'S OF SUB-BLOCK                                      GEOI  97
      CENTER(1,L)=RCL*COSLAM(2)                                         GEOI  98
      CENTER(2,L)=RCL*SINLAM(2)                                         GEOI  99
      CENTER(3,L)=RNEW*SINPSI                                           GEOI 100
      IND=(L+3)/4                                                       GEOI 101
      NCON=0                                                            GEOI 102
      A=AREA(L)                                                         GEOI 103
      RA=RNEW/AE                                                        GEOI 104
      RN=1.0D0                                                          GEOI 105
C COMPUTE CONSTRAINT EQUATIONS FOR SURFACE DENSITY ADJUSTMENT           GEOI 106
      DO 500 N=1,NDCONS                                                 GEOI 107
      RN=RN*RA                                                          GEOI 108
      RNA=RN*A                                                          GEOI 109
      N1=N+1                                                            GEOI 110
      DO 500 M1=1,N1                                                    GEOI 111
```

```
      NCON=NCON+1
      RNPA=RNA*P(M1,N)
      DENCON(NCON,IND)=DENCON(NCON,IND)+RNPA*COSLAM(M1)
      IF(M1.EQ.1) GO TO 500
      NCON=NCON+1
      DENCON(NCON,IND)=DENCON(NCON,IND)+RNPA*SINLAM(M1)
  500 CONTINUE
 1000 CONTINUE
      IF(NDENST.LE.0) RETURN
      NADJ=NDENST-NCONST
      NADJ1=NADJ+1
C INVERT SQUARE SEGMENT OF CONSTRAINT MATRIX
      CALL DNVERT(NCONST,DENCON(1,NADJ1),NCONST,P)
      LM=2
      IF(NDEN.GT.NDENST) LM=4
      INDS(1)=1
      INDS(2)=NADJ
      INDS(3)=NDENST+1
      INDS(4)=NDEN
C COMPUTE COEFFICIENTS RELATING CONSTRAINED AND UNCONSTRAINED ADJUSTED
C SURFACE DENSITIES
      DO 2500 L=1,LM,2
      K1=INDS(L)
      K2=INDS(L+1)
      DO 2500 K=K1,K2
      DO 2000 I=1,NCONST
      A=0.0D0
      DO 1500 J=1,NCONST
      J1=J+NADJ
 1500 A=A+DENCON(I,J1)*DENCON(J,K)
 2000 D(I)=-A
      DO 2500 I=1,NCONST
 2500 DENCON(I,K)=D(I)
      RETURN
      END
```



REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR




B
COMPUTE COEFFICIENTS RELATING CONSTRAINED AND UNCONSTRAINED ADJUSTED SURFACE DENSITIES
RETURN

NAME GEOSRD

PURPOSE TO READ OBSERVATION DATA IN GEOS FORMAT AND PARTIALLY PREPROCESS THE OBSERVATIONS

CALLING SEQUENCE CALL GEOSRD(NSTARD)

SYMBOL	TYPE	DESCRIPTION
NSTARD	I	INPUT - NUMBER OF STATIONS THAT WERE READ FROM CARDS

SUBROUTINES USED
CLEAR2 RANDWR DINRAD EQUATR SATCLC
NUMBR2 NUMBR4 BIAS SATCL2 YMDAY
TDIF

COMMON BLOCKS
APARAM CGEOS PREBLK CTIME CEPHEM
CPARAM CSTINF STANUM CONSTS INTBLK
SIGBLK TPEBLK SRFBLK

INPUT FILES IDTAPE - INPUT FILE NUMBERS

OUTPUT FILES PRINTER

SCRATCH FILES
DATP - 10
ISCR - 11

REFERENCES 'GEODYN OPERATIONS DESCRIPTION' - APPENDIX C
VOLUME 3 - GEODYN DOCUMENTATION

```
      SUBROUTINE GEOSRD(NSTARD)                                         GEOS  32
      IMPLICIT REAL*8 (A-H,O-Z)                                         GEOS  33
      LOGICAL*1 GEOS2F,OKSAT,VHFCHN,PREPRO,LAST,SELSW                   GEOS  34
      LOGICAL TIN,IY,NORATE,NEWTAP                                      GEOS  35
      LOGICAL SATSAT                                                    GEOS  36
      INTEGER*2 IPREPR,INDPRE,IMTYPE,ISTNO,CULL,MINUS,DSTA,IS,CHANEL,   GEOS  37
     .  NMEAS,MTYPE,PRETYP,ESTANO,ISTANO,ISTARD,STANOS,ITMOD,ITSEL,     GEOS  38
     .  IMOD,SELSTA,MTP,ISTA4,IT4,ID4,ISATNO,JBASE,KBASE                GEOS  39
      INTEGER A,DATE,RECNO,OUTP                                         GEOS  40
      REAL TDIF,SATCLC,SATCL2,GS2R,GINTBK,SIGSTD,SIG1,SIG2,DAYINT,      GEOS  41
     .  SIGCHG,RFINDX,SIGMA1,SIGMA2,S4,SCPRNT,RFNDX                     GEOS  42
      DOUBLE PRECISION NAME,JNAME,ICARD                                 GEOS  43
      DIMENSION A(12,4),X(3),LAST(4),ISAT(4),MTP(4),IST4(4),IT4(4),     GEOS  44
     .  ISTA4(4),IY4(4),IH4(4),S4(4),ID4(4),RFNDX(2)                    GEOS  45
      COMMON/APARAM/INPAR,INPARI,NBIAS,NSTSTA,NSAT,NGPARC(5)            GEOS  46
      COMMON/CGEOS/ISAT2(2),IPREPR(4,50),RFINDX(2,50),INDPRE(2,50),     GEOS  47
     .  NPRE,NSIG,NCULL,SIGCHG(50),IMTYPE(50),ISTNO(50),CULL(2,100)     GEOS  48
      COMMON/PREBLK/DAY,OBS1,OBS2,SIG(2),SRFNDX,ISTA,MTYPE,NMEAS,       GEOS  49
     .  ISATNO,PRETYP,CHANEL,VHFCHN,PREPRO,RECNO                        GEOS  50
      COMMON/CTIME/DATAEP,DAYREF,DAYE,DAYSTP,DAYINT(15)                 GEOS  51
      COMMON/CEPHEM/JNAME(381),ISTARD(381),ESTANO(381),ISTANO(386)      GEOS  52
      COMMON/CPARAM/NSTA,NMAST(18)                                      GEOS  53
      COMMON/CSTINF/JBASE(283),KBASE(283),LBASE                         GEOS  54
      COMMON/STANUM/NAME(280),STANOS(290),NOSTDR                        GEOS  55
```

```
      COMMON/CONSTS/DPI,DTWOPI,D2R,S2R
      COMMON/SRFBLK/LSTA(600),DTIME(600),BFREQ(600),IYS1(100),IHS1(100),
     .   IYS2(100),IHS2(100),ITMOD(366,11),ITSEL(100),IMOD(100),
     .   SELSTA(100),SELSW(100),IEBIAS(476)
      COMMON/INTBLK/CINTEK(53),NORATE,INTCPK(78)
      COMMON/SIGBLK/SIGSTO(30),SGPRNT(30),IDTAPE(4)
      COMMON/TPEBLK/INTP,OUTP(5),IOBS,ISCRA(4)
      EQUIVALENCE (SIG1,SIG(1)),(KKSAT,SIG2,SIG(2)),(NLAST,LAST(1))
      EQUIVALENCE (RFNDX(1),SRFNDX)
      DATA DELETE/6HDELETE/,ENDALL/6HENDALL/,NEWTP/3/
      DATA IGEOSA,IGEOS1/65891,650891/
      DATA IGEOSB,IGEOS2/68021,680021/
      DATA C/2.997925D0/
      DATA MINUS/1H-/,ISCR,DATP/11,10/
C INITIALIZE
      NUMBER=0
      GEOS2F=MTYPE.NE.1
      NORATE=.TRUE.
      NEWTAP=IOBS.EQ.2.OR.IOBS.EQ.3
      IOBS=MOD(IOBS,2)
      SIG(1)=0.0D0
      SIG(2)=0.0D0
C DETERMINE NUMBER OF INPUT UNITS
      DO 610 I=1,3
      IF(IDTAPE(I).GT.0) GO TO 610
      I1=I+1
      DO 600 J=I1,4
  600 IDTAPE(J-1)=IDTAPE(J)
      IDTAPE(4)=0
  610 CONTINUE
      NTAPE=NUMOR4(0,IDTAPE,4)-1
      IF(NTAPE.LT.0) NTAPE=4
      IF(NTAPE.GT.0) GO TO 612
      NTAPE=1
      IDTAPE(1)=20
  612 IYMIN=0
      IHMIN=250000
      IYMAX=999999
      IHMAX=0
      CALL CLEAR2(ITMOD,366,11)
      MLAST=0
      DAYPRV=DATAEP
      NORATE=.TRUE.
      ISEL=0
      DJBASE=YMDAY(500100,0,0.0D0)
      IF(IOBS.LE.0) GO TO 620
C READ SELECT AND DELETE CARDS IF NECESSARY
      IYMAX=0
      IYMIN=999999
  613 READ(INTP,4000) ICARD,IYMD,IHM,IT,IST,ISTA,MTYPE,LMOD
      IF(ICARD.EQ.ENDALL) GO TO 620
      ISEL=ISEL+1
      SELSW(ISEL)=ICARD.NE.DELETE
      IF(IT.LE.0) IT=999999
      IYS1(ISEL)=IYMD
      IHS1(ISEL)=IHM
```

```
      IYS2(ISEL)=IT
      IHS2(ISEL)=IST
      ITSEL(ISEL)=MTYPE
      IMOD(ISEL)=MAXO(LMCD,1)
      SELSTA(ISEL)=ISTA
      IF(.NOT.SELSW(ISEL)) GO TO 618
      IYMIN=MINO(IYMIN,IYMD)
      IHMIN=MINO(IHMIN,IHM)
      IYMAX=MAXO(IYMAX,IT)
      IHMAX=MAXO(IHMAX,IST)
      GO TO 618
C READ FIRST RECORD FROM EACH UNIT
  620 DO 625 I=1,NTAPE
      IOBS=IDTAPE(I)
  622 READ(IOBS,2000,ERR=622,END=624) ISAT(I),MTP(I),ID4(I),IST4(I),
     .    IT4(I),ISTA4(I),IY4(I),IH4(I),S4(I),(A(J,I),J=1,12)
      IF(ISAT2(I).EQ.0) ISAT2(I)=ISAT(I)
      DO 670 J=1,NSAT
      IF(ISAT(I).NE.ISAT2(J)) GO TO 670
      GO TO 625
  670 CONTINUE
      GO TO 622
  624 LAST(I)=.TRUE.
  625 CONTINUE
  630 IHM=IH4(1)
      IYMD=IY4(1)
      IF(LAST(1)) IYMD=9999999
      ITAPE=1
      IOBS=IDTAPE(1)
      IF(NTAPE.EQ.1) GO TO 640
C TEST FOR EARLIEST TIME
      DO 635 I=2,NTAPE
      IF(LAST(I).OR.IYMD.LT.IY4(I)) GO TO 635
      IHMS=IH4(I)
      IF(IHM.LT.IHMS.AND.IYMD.EQ.IY4(I)) GO TO 635
      ITAPE=I
      IOBS=IDTAPE(I)
      IHM=IHMS
      IYMD=IY4(I)
  635 CONTINUE
      IF(LAST(ITAPE)) GO TO 700
      GO TO 640
  637 IF(LAST(ITAPE)) GO TO 630
C READ NEXT RECORD FROM UNITS LAST USED
  639 READ(IOBS,2000,ERR=639,END=629) ISAT(ITAPE),MTP(ITAPE),ID4(ITAPE),
     .    IST4(ITAPE),IT4(ITAPE),ISTA4(ITAPE),IY4(ITAPE),IH4(ITAPE),
     .    S4(ITAPE),(A(J,ITAPE),J=1,12)
      IF(ISTA4(ITAPE).EQ.0) LAST(ITAPE)=.TRUE.
      CKSAT=.FALSE.
C CHECK SATELLITE ID
      DO 665 I=1,NSAT
      IF(CKSAT) GO TO 665
      ISATNC=I
      OKSAT=ISAT(ITAPE).EQ.ISAT2(I)
  665 CONTINUE
      IF(.NOT.OKSAT) GO TO 637
```

```
      GO TO 630
  639 LAST(ITAPE)=.TRUE.
      GO TO 630
C REWIND INPUT TAPES WHEN SELECTION COMPLETE
  700 ISTA=0
      IF(.NOT.NEWTAP) GO TO 20
      WRITE(NEWTP,2000) ISAT(ITAPE),MTYPE,ID,IST,IT,ISTA,IYMD,IHM,
     .   S4(ITAPE),(A(J,ITAPE),J=1,12)
      END FILE NEWTP
      GO TO 20
C CHECK FOR ACCEPTABLE TIME
  640 IF(IYMD.LT.IYMIN.OR.(IYMD.EQ.IYMIN.AND.IHM.LT.IHMIN)) GO TO 537
      IF(IYMD.GT.IYMAX.OR.(IYMD.EQ.IYMAX.AND.IHM.GE.IHMAX)) GO TO 700
      MTYPE=MTP(ITAPE)
      ISTA=ISTA4(ITAPE)
      IF(GEOS2F)ISTA=ISTA/10
      ISTA=MOD(ISTA,10000)
      IF(ISTA.EQ.0) GO TO 700
      IF(ISEL.LE.0) GO TO 660
C CHECK IF MEASUREMENT HAS BEEN SELECTED
      DO 645 I=1,ISEL
      IF((.NOT.SELSW(I)).OR.(MTYPE.NE.ITSEL(I).AND.ITSEL(I).GT.0).OR.
     .   (ISTA.NE.SELSTA(I).AND.SELSTA(I).NE.0).OR.IYMD.LT.IYS1(I).OR.
     .   IYMD.GT.IYS2(I)) GO TO 645
      IF(IYMD.EQ.IYS1(I).AND.IHM.LT.IHS1(I)) GO TO 645
      IF(IYMD.EQ.IYS2(I).AND.IHM.GT.IHS2(I)) GO TO 645
      ISMOD=IMOD(I)
      GO TO 650
  645 CONTINUE
      GO TO 637
C CHECK IF MEASUREMENT HAS BEEN DELETED
  650 DO 655 J=1,ISEL
      IF(SELSW(J).OR.(MTYPE.NE.ITSEL(J).AND.ITSEL(J).GT.0).OR.(ISTA
     .   .NE.SELSTA(J).AND.SELSTA(J).NE.0).OR.IYMD.LT.IYS1(J).OR.
     .   IYMD.GT.IYS2(J)) GO TO 655
      IF(IYMD.EQ.IYS1(J).AND.IHM.LT.IHS1(J)) GO TO 655
      IF(IYMD.EQ.IYS2(J).AND.IHM.GT.IHS2(J)) GO TO 655
      GO TO 637
  655 CONTINUE
  660 ISN=ISTA
      ID=ID4(ITAPE)
      ITEMP=ISCR
      IF(MTYPE.EQ.4.AND.ID.EQ.5)ITEMP=DATP
      IF(.NOT.NORATE) GO TO 16
      IF(MTYPE.NE.3.AND.MTYPE.NE.4) GO TO 16
      IF(NSIG.LE.0) GO TO 17
      NN1=0
C DETERMINE IF SIGMA IS CHANGED
      DO 15 I=1,NSIG
      IF(ISN.NE.ISTNO(I).AND.ISTNO(I).NE.0) GO TO 15
      IF(3.EQ.IMTYPE(I).OR.IMTYPE(I).EQ.0) NN1=I
   15 CONTINUE
      IF(NN1.GT.0.AND.SIGCHG(NN1).NE.0.) NORATE=.FALSE.
      GO TO 16
   17 NORATE=.FALSE.
   16 ISTA=ISN
```

```
C CHECK FOR STATION PRESENT
      ISN=NUMBR2(ISTA,ISTANO,NSTA)
      IF(ISN.GT.0) GO TO 73
      ISN=NUMBR2(ISTA,ISTARD,NSTARD)
      IF(ISN.GT.0) GO TO 71
      ISN=NUMBR2(ISTA,STANOS,NOSTOR)
      IF(ISN.GT.0) GO TO 71
      IF(ISTA.EQ.-4.AND.MTYPE.EQ.9.AND.ID.EQ.4) GO TO 71
      PRINT 3000,ISTA
      GO TO 637
   71 NSTA=NSTA+1
      ISN=NSTA
      ISTANO(ISN)=ISTA
   73 IF(ISEL.LE.0) GO TO 72
      K=MTYPE
      KTYPE=MOD(K,19)
      K=ITMOD(ISN,KTYPE)+1
      K=MOD(K,ISMOD)
C WRITE SELECTED MEASUREMENTS ON SCRATCH FILE
      ITMOD(ISN,KTYPE)=K
      IF(K.NE.0) GO TO 637
72    WRITE(ITEMP,2010) ISAT(ITAPE),MTYPE,ID,IST4(ITAPE),IT4(ITAPE),
     .   ISN,IYMD,IHM,S4(ITAPE),(A(J,ITAPE),J=1,12)
      IF(.NOT.NEWTAP) GO TO 637
      K=ISTANO(ISN)*10
C END FILE AND REWIND SCRATCH FILE
      WRITE(NEWTP,2000) ISAT(ITAPE),MTYPE,ID,IST4(ITAPE),IT4(ITAPE),K,
     .   IYMD,IHM,S4(ITAPE),(A(J,ITAPE),J=1,12)
      GO TO 637
   20 ENDFILE ISCR
      REWIND ISCR
      ENDFILE DATP
      REWIND DATP
      VFFCHN=.FALSE.
      IY=.TRUE.
      IM=0
C READ DOPPLER FILE AND FILL FREQUENCY TABLE
      NDOP=0
      DJBASE=YMDAY(500100,0,0.D0)
   30 READ(DATP,2020,END=40)DSTA(NDOP+1),IYMD,IHM,SEC,BFREQ(NDOP+1)
      NDOP=NDOP+1
      DTIME(NDOP)=YMDAY(IYMD,IHM,SEC)+1.D0/1440.D0
      IF(NDOP.LT.600) GO TO 30
      PRINT 10000
   40 REWIND DATP
   50 READ(ISCR,1000,END=60)ISATID,MTYPE,ID,IST,IT,ISN,IYMD,IHM,SEC
C READ FIRST RECORD FOR EACH OBSERVATION
      SATSAT=.FALSE.
      CHANEL=0
C STATION AND TYPE ZERO FOR LAST OBSERVATION STORED
      IF(ISN.GT.0) GO TO 70
   60 REWIND ISCR
      ISTA=0
      MTYPE=0
      RECNO=RECNO+1
      PRINT 5000,NUMBER,(IDTAPE(I),I=1,NTAPE)
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      CALL RANDWR
      IM=0
C COMPUTE MEASUREMENT TIME IN A.1 DAYS FROM REFERENCE JAN 0.0
      DAYSTP=DAY
      RETURN
   70 DAY=YMDAY(IYMD,IHM,SEC)
      IF(IT.EQ.7) IT=4
      IF(IT.EQ.0.OR.IT.GT.4) IT=3
      DAY=DAY+TDIF(4,IT,DAY)/86400.
      IF(DAY.LT.DATAEP) GO TO 180
      IF(DAY.GT.DAYSTP) GO TO 60
      NMEAS=1
      SIG2=0.
      IND=0
      IPRE=0
      PRETYP=0
      ISATNO=NUMBR4(ISATID,ISAT2,NSAT)
      MT=MTYPE
      IF(MTYPE.EQ.8) MT=3
      IF(MTYPE.EQ.5) MT=26+ID
      IF(MTYPE.EQ.9) MT=ID
      KSTA=ISTANO(ISN)
C DETERMINE PREPROCESSING TO BE DONE
      IF(NPRE.EQ.0) GO TO 100
      DO 75 I=1,NPRE
   75 IF((INDPRE(1,I).EQ.0.OR.INDPRE(1,I).EQ.KSTA).AND.
     .   (INDPRE(2,I).EQ.0.OR.INDPRE(2,I).EQ.MT)) IND=I
      IF(IND.EQ.0) GO TO 100
      SRFNDX=RFINDX(1,IND)
      IF(IPREPR(3,IND).GT.0) DAY=DAY+RFINDX(2,IND)/8.64E4
      IPRE=IPREPR(1,IND)
      PRETYP=IPREPR(2,IND)
      IF(IPRE.EQ.1) IPRE=1-IST/5
C DETERMINE SIGMA CHANGES IF ANY
  100 NN1=0
      NN2=0
      IF(NSIG.EQ.0) GO TO 108
      DO 106 I=1,NSIG
      IF(KSTA.NE.IST NO(I).AND.ISTNO(I).NE.0) GO TO 106
      IF(MT.EQ.IMTYPE(I).OR.IMTYPE(I).EQ.0) NN1=I
      IF(MT.GT.7) GO TO 106
      IF(MT+7.EQ.IMTYPE(I).OR.IMTYPE(I).EQ.0) NN2=I
  106 CONTINUE
C PROCESS AND READ SECOND RECORD ACCORDING TO MEASUREMENT TYPE
C ...OPTICAL, AZIMUTH & ELEVATION
  108 GO TO (110,120,130,130,150,160,110,170,290),MTYPE
110   READ(ISCR,1010) M1,N1,OBSO1,IS,M2,N2,OBSO2,IMEAN,IEQAT,SIG1,SIG2
      NMEAS=2
      CALL DINRAD(OBS1,M1,N1,OBSO1,MTYPE+1)
      CALL DINRAD(OBS2,M2,N2,OBSO2,1)
      IF(IS.EQ.MINUS)OBS2=DSIGN(OBS2,-1.D0)
      IF(NN1.GT.0) SIG1=SIGCHG(NN1)
      IF(NN2.GT.0) SIG2=SIGCHG(NN2)
      COSD=DCOS(OBS2)
      SIG1=SIG1*S2R
      IF(IND.NE.0.OR.NN1.GT.0) SIG1=SIG1/COSD
```

```
      SIG2=SIG2*S2R
      IF(MTYPE.NE.1) GO TO 400
      PRETYP=0
      IF(IND.EQ.0.OR.IPREPR(4,IND).EQ.0) GO TO 400
      PRETYP=1
      IF(ID.EQ.0.AND.(ISATID.EQ.IGEOSA.OR.ISATID.EQ.IGEOS1))
     .   DAY=DAY+SATCLC(IY,DAY+TDIF(3,4,DAY)/8.64E4)/8.64E4
      IF(ID.EQ.0.AND.(ISATID.EQ.IGEOSB.OR.ISATID.EQ.IGEOS2))
     .   DAY=DAY+SATCL2(IY,DAY+TDIF(3,4,DAY)/8.64E4)/8.64E4
      IF(ID.EQ.0.AND.(ISATID.EQ.IGEOS1.OR.ISATID.EQ.IGEOSA.OR.
     .   ISATID.EQ.IGEOS2.OR.ISATID.EQ.IGEOSB)) IPRE=0
      TIN=IMEAN.EQ.1
      IF(TIN.AND.IEQAT.EQ.3) GO TO 400
      DEQATR=DJBASE
      IF(IEQAT.EQ.2) DEQATR=0.D0
      IF(IEQAT.EQ.3) DEQATR=DAY
      X(1)=DCOS(OBS1)*COSD
      X(2)=DSIN(OBS1)*COSD
      X(3)=DSIN(OBS2)
      CALL EQUATR(X,DEQATR,TIN,X,DAY,.TRUE.)
      OBS1=DATAN2(X(2),X(1))
      IF(OBS1.LT.0.D0) OBS1=OBS1+DTWOPI
      OBS2=DARSIN(X(3))
      GO TO 400
C ...RANGE
  120 READ(ISCR,1020) OBS1,ISCOR,ITR,IRC,ITROP,SIG1,REFRAC,
     .   IVALID,CHANEL
      IF(IVALID.LT.6) GO TO 123
      IF(ISCOR.GT.99) ISCOR=ISCOR-100
      DAY=DAY+DFLOAT(ISCOR)/8.64D+10
  123 IF(NN1.GT.0) SIG1=SIGCHG(NN1)
      IF(PRETYP.EQ.1.AND.(ITROP.EQ.1.OR.ITROP.EQ.3)) PRETYP=0
      IF(ITROP.NE.9.OR.PRETYP.LE.0) GO TO 125
      PRETYP=3
      SRFNDX=REFRAC
  125 IF(IPRE.LE.0) GO TO 400
      IF(IND.GT.0) DAY=DAY-OBS1/(C*8.64D4)
      IPRE=CHANEL+1
      GO TO 400
C ...RANGE RATE & DOPPLER
  130 READ(ISCR,1030) OBS1,ITROP,SIG1,CHANEL
      IF(PRETYP.EQ.1.AND.(ITROP.EQ.1.OR.ITROP.EQ.3)) PRETYP=0
      IF(MTYPE.EQ.3) GO TO 135
      MTYPE=3
      IF(NDOP.EQ.0) GO TO 50
      DO 140 I=1,NDOP
         IF(DAY.GT.DTIME(I).OR.ISN.NE.DSTA(I)) GO TO 140
         OBS1=C*(BFREQ(I)-OBS1)/OBS1
         SIG1=C*SIG1/BFREQ(I)
  135 IF(NN1.GT.0) SIG1=0.01*SIGCHG(NN1)
      GO TO 400
140   CONTINUE
      GO TO 50
C ...MINITRACK
150   READ(ISCR,1040) OBS1,OBS2
      SIGMA1=SIGSTD(5)
```

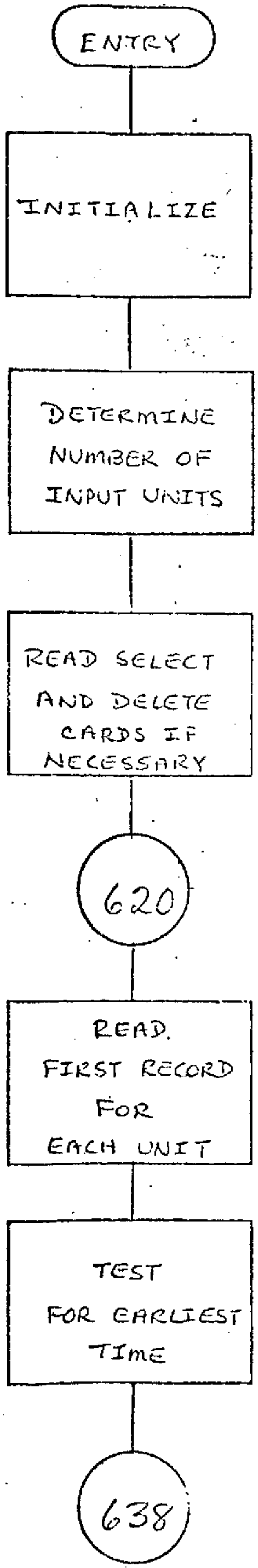
```
      SIGMA2=SIGSTD(12)
      IF(NN1.GT.0) SIGMA1=SIGCHG(NN1)
      IF(NN2.GT.0) SIGMA2=SIGCHG(NN2)
      SIG1=OBS1**2
      SIG2=OBS2**2
      IF(SIG1.LT.1.) SIG1=SQRT(1.-SIG1)
      IF(SIG2.LT.1.) SIG2=SQRT(1.-SIG2)
      SIG1=SIGMA1*1.E-3/SIG1
      SIG2=SIGMA2*1.E-3/SIG2
      NMEAS=2
      GO TO 400
C ...X-Y ANGLES
150   READ(ISCR,1050)OBS1,OBS2,SIG1,SIG2
      NMEAS=2
      OBS1=OBS1*D2R
      OBS2=OBS2*D2R
      IF(NN1.GT.0) SIG1=SIGCHG(NN1)
      IF(NN2.GT.0) SIG2=SIGCHG(NN2)
      SIG1=SIG1*S2R
      SIG2=SIG2*S2R
      GO TO 400
C ...TIME DELAY, FIRING RATE, 2 & 3 WAY AVERAGE RANGE RATE
  170 READ(ISCR,1060) ITCOR,ITRCP,ION,ISTA2,OBS1,OBS2
      DAY=DAY+DFLOAT(ITCOR)/8.64D+10
      IPRE=1
      MTYPE=20+ID
      SIGMA1=SIGSTD(MTYPE)
      IF(NN1.GT.0) SIGMA1=SIGCHG(NN1)
      SIG1=SIGMA1
      IF(PRETYP.GT.0) PRETYP=PRETYP-1+ITRCP
      IF(PRETYP.LE.0) GO TO 172
      IND2=0
      RFNDX(2)=0.
      DO 171 I=1,NPRE
  171 IF((INDPRE(1,I).EQ.0.OR.INDPRE(1,I).EQ.ISTA2).AND.
     .   (INDPRE(2,I).EQ.0.OR.INDPRE(2,I).EQ.MTYPE)) IND2=I
      IF(IND2.GT.0) RFNDX(2)=RFINDX(1,IND2)
  172 ISN2=NUMBR2(ISTA2,ISTANO,NSTA)
      IF(ISN2.GT.0) GO TO 175
      ISN2=NUMBR2(ISTA2,ISTARD,NSTARD)
      IF(ISN2.GT.0) GO TO 173
      ISN2=NUMBR2(ISTA2,STANOS,NOSTOR)
      IF(ISN2.GT.0) GO TO 173
      PRINT 3000,ISTA2
      GO TO 50
  173 NSTA=NSTA+1
      ISN2=NSTA
      ISTANO(ISN2)=ISTA2
  175 CHANEL=ISN2
      SIG(2)=OBS2
      IF(LBASE.EQ.0) GO TO 185
      DO 186 I=1,LBASE
      IF(ISN.NE.JBASE(I)) GO TO 186
      IF(ISN2.NE.KBASE(I)) GO TO 186
      GO TO 190
  186 CONTINUE
```

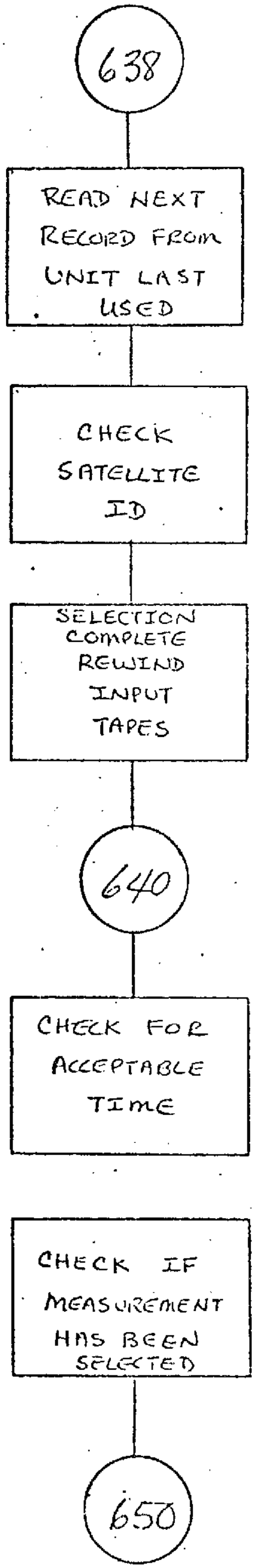
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

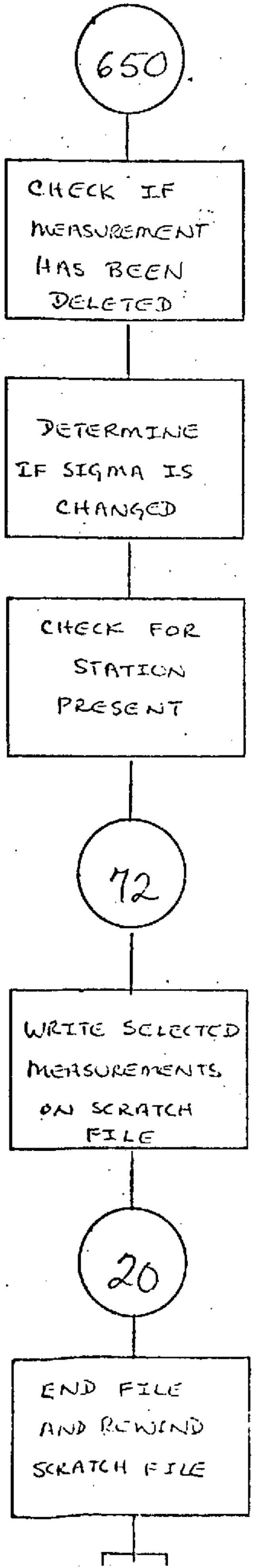
```
  193 LBASE=LBASE+1
      JBASE(LBASE)=ISN
      KBASE(LBASE)=ISN2
  190 GO TO (177,178,179,179),ID
  177 SIG1=SIG1*1.0E-9
      GO TO 400
  178 SIG1=SIG1*1.0E-5
      GO TO 400
  179 OBS2=DAY-OBS2/8.64D+8
      SIG1=SIG1*1.0E-2
      GO TO 400
180   READ(ISCR,1000)
      GO TO 50
  290 J=ID-1
      GO TO (292,393,494),J
  292 READ(ISCR,1092) IREFR,OBS1,SIG1,IRELAY,CHANEL
      CHANEL=CHANEL-1
      IF(NN1.GT.0) SIG1=SIGCHG(NN1)
      GO TO 90
  393 READ(ISCR,1093) IREFR,OBS1,SIG1,IRELAY
      IF(NN1.GT.0) SIG1=SIGCHG(NN1)*0.01
   90 IF(PRETYP.GT.0) PRETYP=PRETYP-MOD(IREFR,2)
      J=ISATNO
      J=MOD(J,2)+1
      IF(IRELAY.EQ.ISAT2(J)) GO TO 92
      PRINT 6000,IRELAY
      GO TO 50
   92 KKSAT=J
      MTYPE=ID
      IPRE=1
      OBS2=DAY
      SATSAT=.TRUE.
      GO TO 400
  494 READ(ISCR,1094) IREFR,OBS1,SIG1,SIG2,IGEO,OBS2
C     IF(IST.EQ.0) CALL SATCL3(DAY)
      IF(NN1.GT.0) SIG1=SIGCHG(NN1)
      IF(NN2.GT.0) SIG2=SIGCHG(NN2)
      SIG2=SIG2*0.01
      IF(PRETYP.GT.0) PRETYP=PRETYP-MOD(IREFR,2)
      NMEAS=2
      MTYPE=ID
  400 IF(NCULL.EQ.0) GO TO 420
C CULL MEASUREMENTS
      DO 415 I=1,NCULL
      DO 415 J=1,NMEAS
      IF(IM+J-CULL(1,I)) 415,413,411
  411 IF(IM+J.GT.CULL(2,I)) GO TO 415
  413 SIG(J)=0.0D0
  415 CONTINUE
  420 IM=IM+NMEAS
C SET PREPROCESSING SWITCH
      PREPRO=IND.NE.0.AND.NPRE.NE.0
      PREPRO=PREPRO.OR.MTYPE.GT.26
      IF(SATSAT) PREPRO=.TRUE.
      PRETYP=PRETYP+10*IPRE
      NUMBER=NUMBER+NMEAS
```

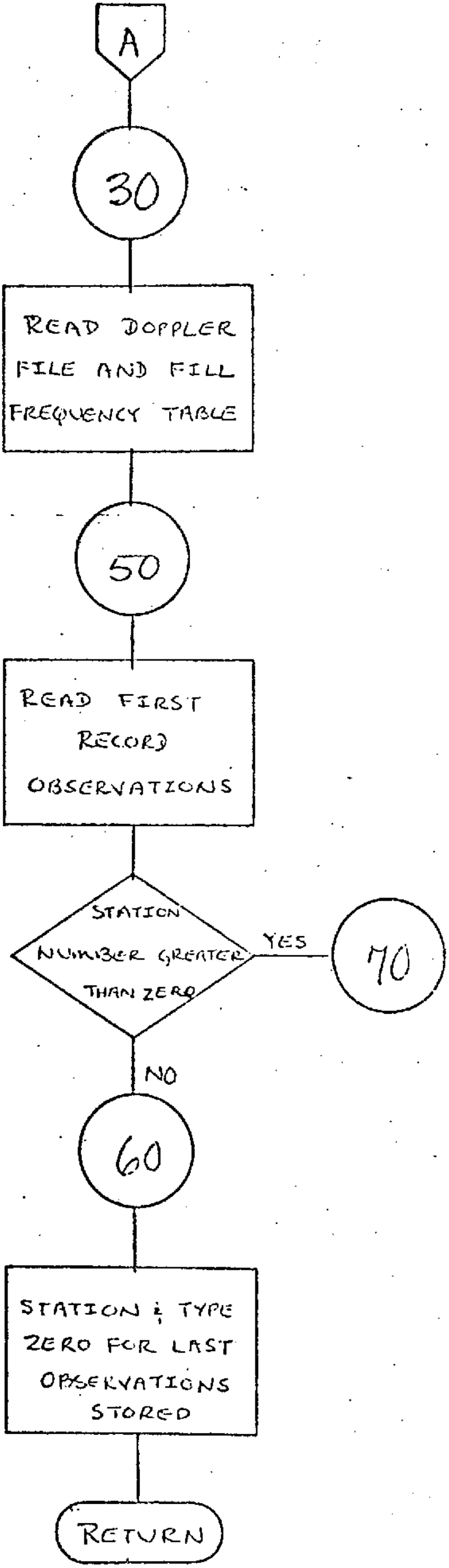
```
      ISTA=ISN
      IF(MTYPE.GT.7) GO TO 425
      CHANEL=CHANEL+1
      IF(IB.GT.0) CHANEL=0
  425 RECNO=RECNO+1
      IF(NBIAS.GT.0) CALL BIAS
      CALL RANDWR
      SIG(2)=0.0D0
      GO TO 50
1000  FORMAT(I6,2I1,3X,2I1,1X,I4,I6,I4,F6.4)
1010  FORMAT(I3,I2,D5.3,A1,2I2,F4.2,8X,2I1,8X,2F3.2)
 1020 FORMAT(D19.3,3I2,4X,I1,F6.3,F6.4,1X,I1,1X,I1)
 1030 FORMAT(D19.3,10X,I1,10X,F3.2,2X,I1)
 1040 FORMAT(3X,D6.7,F5.7)
1050  FORMAT(D5.2,5X,F5.2,22X,2F3.2)
 1080 FORMAT(I2,2I1,I4,D22.16,D16.5)
 1092 FORMAT(I1,D19.3,F6.3,I6,I1)
 1093 FORMAT(I1,D19.6,3PF6.3,I6,I1)
 1094 FORMAT(I1,D19.3,F3.1,F3.2,I1,D19.6)
 2000 FORMAT(I6,2I1,A4,A1,I5,2I6,A4,A2,11A4)
 2010 FORMAT(I6,2I1,A4,A1,I5,2I6,A4/A2,11A4)
2020  FORMAT(14X,I4,I5,I4,F6.4/D19.3)
 3000 FORMAT(1H ,'STATION ',I4,' NOT FOUND IN FILE')
10000 FORMAT(1H0,20X,'***** OVER 200 DOPPLER PASSES-TABLE OVERFLOW ***')
4000  FORMAT(A6,2(I6,1X,I6,5X),1X,I4,1X,I1,I2)
 5000 FORMAT(1H0//31X,I6,' OBSERVATIONS SELECTED FROM MASTER GEOS ',
     .    'DATA TAPE NUMBER(S)',A14)
 6000 FORMAT(' RELAY SATELLITE',I8,' NOT PRESENT')
      END
```

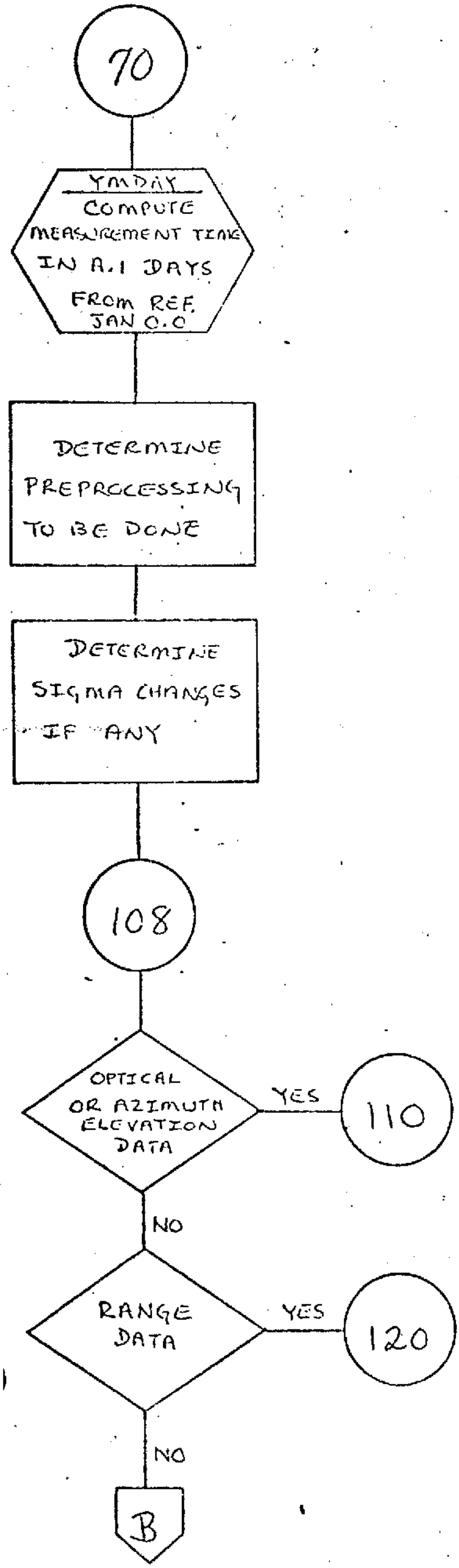
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

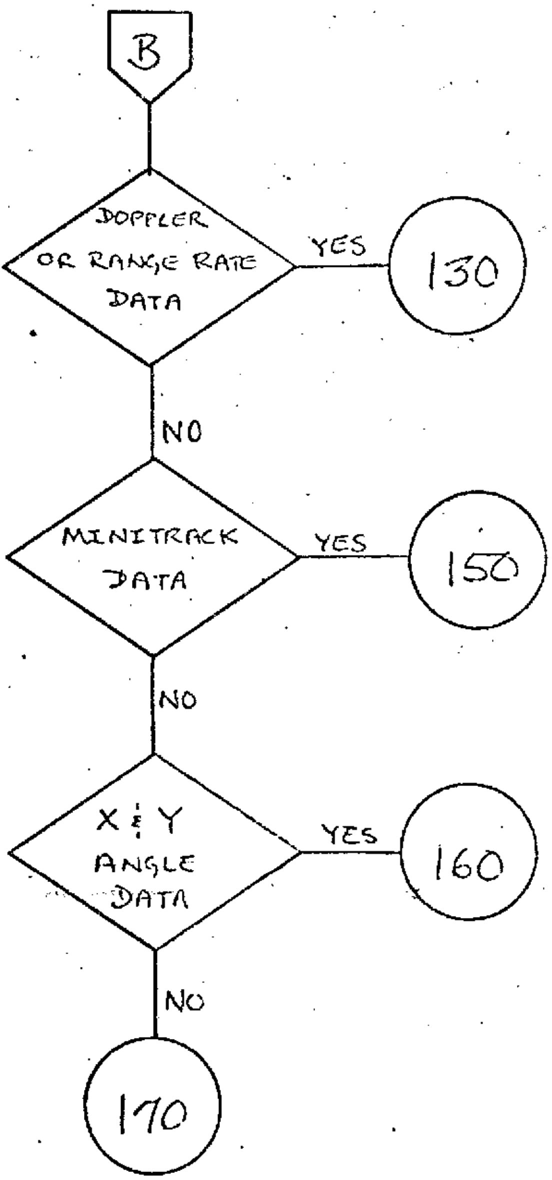

ENTRY
INITIALIZE
DETERMINE NUMBER OF INPUT UNITS
READ SELECT AND DELETE CARDS IF NECESSARY
620
READ FIRST RECORD FOR EACH UNIT
TEST FOR EARLIEST TIME
638

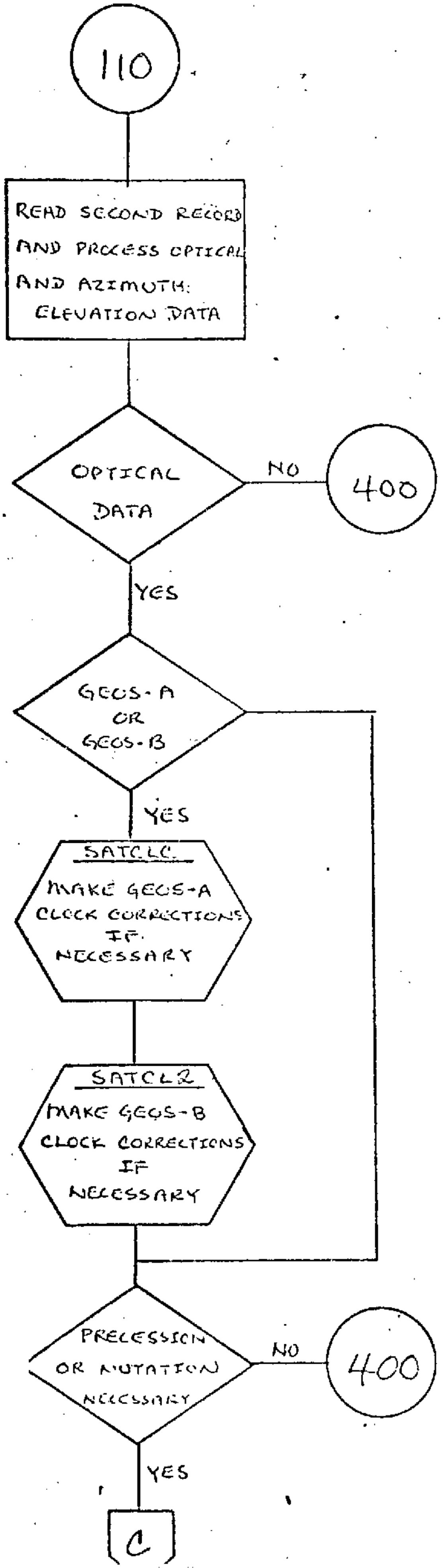

638
READ NEXT RECORD FROM UNIT LAST USED
CHECK SATELLITE ID
SELECTION COMPLETE REWIND INPUT TAPES
640
CHECK FOR ACCEPTABLE TIME
CHECK IF MEASUREMENT HAS BEEN SELECTED
650

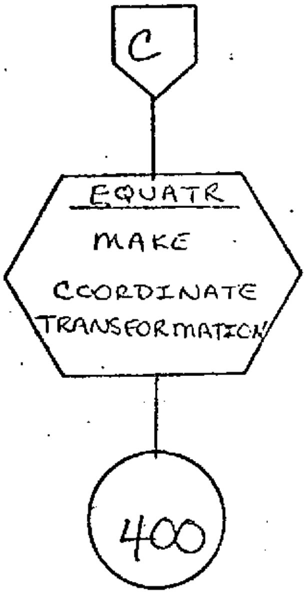

650
CHECK IF MEASUREMENT HAS BEEN DELETED
DETERMINE IF SIGMA IS CHANGED
CHECK FOR STATION PRESENT
72
WRITE SELECTED MEASUREMENTS ON SCRATCH FILE
20
END FILE AND REWIND SCRATCH FILE

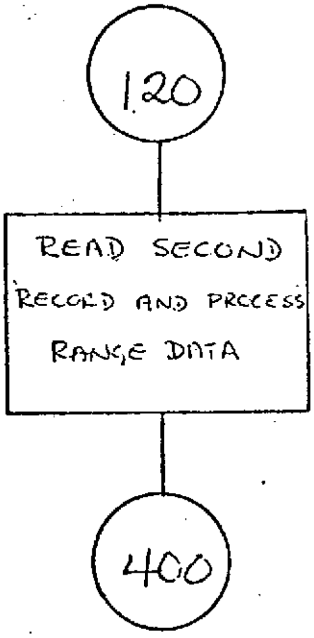

A
30
READ DOPPLER FILE AND FILL FREQUENCY TABLE
50
READ FIRST RECORD OBSERVATIONS
STATION NUMBER GREATER THAN ZERO
YES
70
NO
60
STATION & TYPE ZERO FOR LAST OBSERVATIONS STORED
RETURN

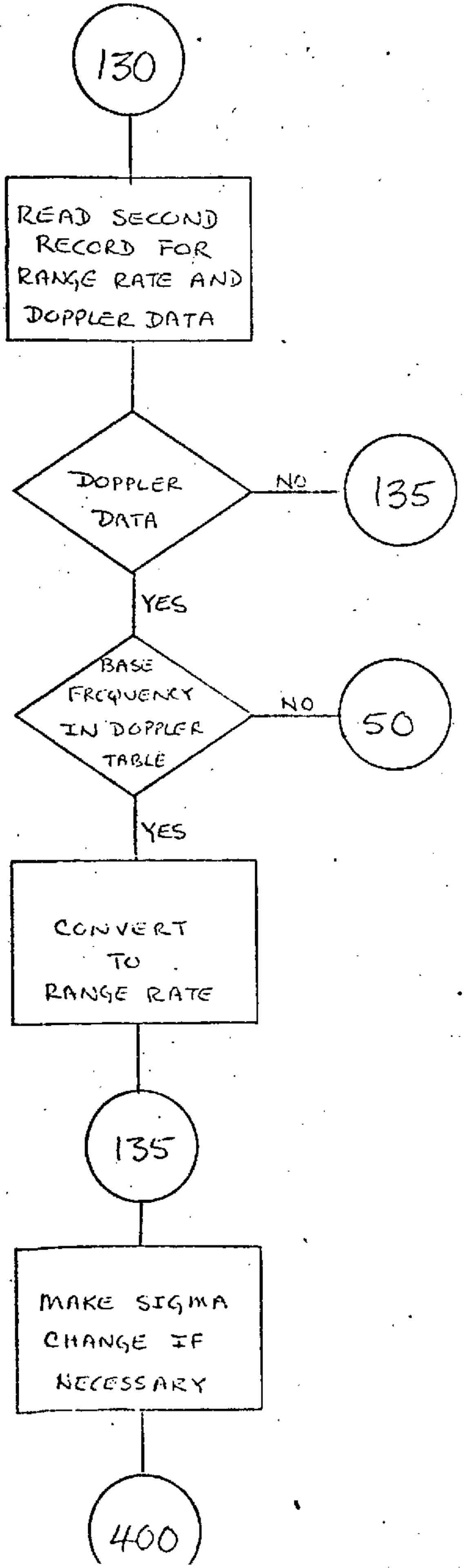

70
YMDAY
COMPUTE
MEASUREMENT TIME
IN A.1 DAYS
FROM REF.
JAN 0.0
DETERMINE
PREPROCESSING
TO BE DONE
DETERMINE
SIGMA CHANGES
IF ANY
108
OPTICAL
OR AZIMUTH
ELEVATION
DATA
YES
110
NO
RANGE
DATA
YES
120
NO
B

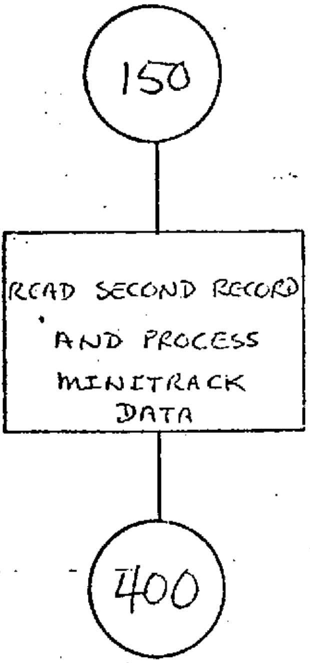

B
DOPPLER
OR RANGE RATE
DATA
YES
130
NO
MINITRACK
DATA
YES
150
NO
X & Y
ANGLE
DATA
YES
160
NO
170

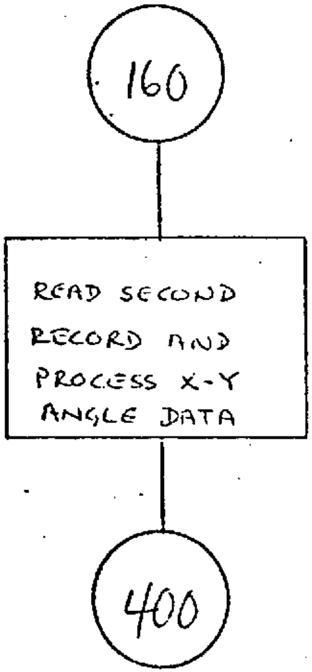

110
READ SECOND RECORD AND PROCESS OPTICAL AND AZIMUTH: ELEVATION DATA
OPTICAL DATA
NO
400
YES
GEOS-A OR GEOS-B
YES
SATCLC
MAKE GEOS-A CLOCK CORRECTIONS IF NECESSARY
SATCL2
MAKE GEOS-B CLOCK CORRECTIONS IF NECESSARY
PRECESSION OR NUTATION NECESSARY
NO
400
YES
C

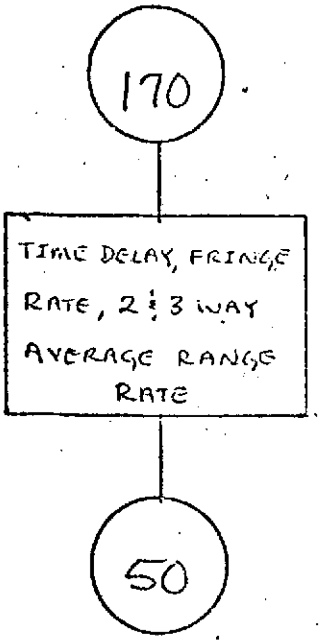

C
EQUATR
MAKE COORDINATE TRANSFORMATION
400

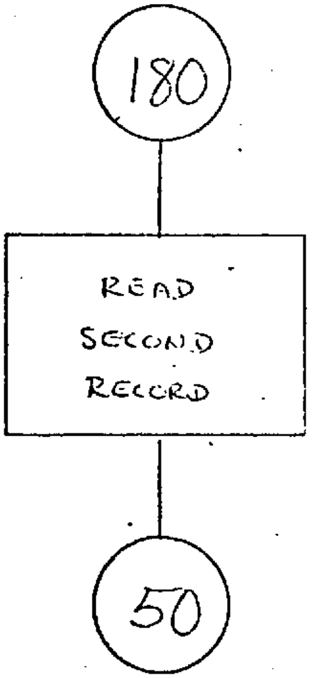

120
READ SECOND RECORD AND PROCESS RANGE DATA
400

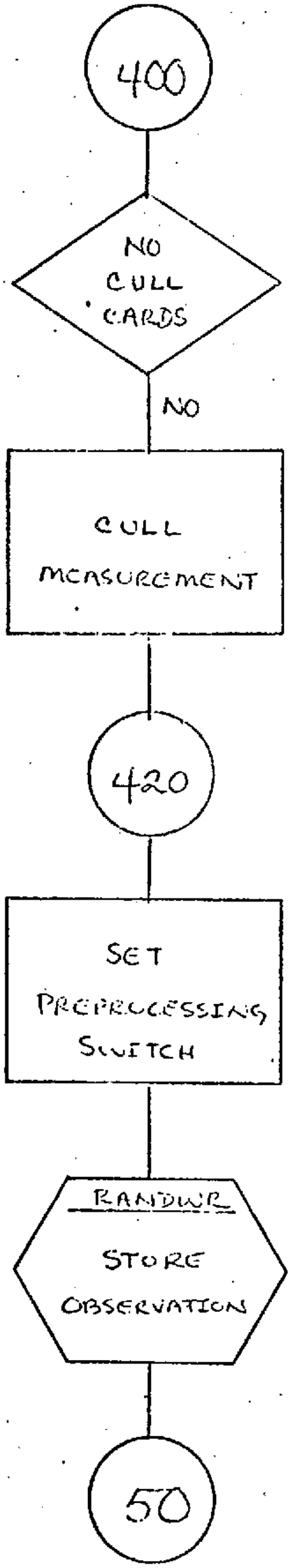

130
READ SECOND RECORD FOR RANGE RATE AND DOPPLER DATA
DOPPLER DATA
NO
135
YES
BASE FREQUENCY IN DOPPLER TABLE
NO
50
YES
CONVERT TO RANGE RATE
135
MAKE SIGMA CHANGE IF NECESSARY
400


150
READ SECOND RECORD AND PROCESS MINITRACK DATA
400


160
READ SECOND RECORD AND PROCESS X-Y ANGLE DATA
400


170
TIME DELAY, FRINGE RATE, 2 & 3 WAY AVERAGE RANGE RATE
50


180
READ SECOND RECORD
50


400
NO CULL CARDS
NO
CULL MEASUREMENT
420
SET PREPROCESSING SWITCH
RANDWR
STORE OBSERVATION
50

GRHRAN

DESCRIPTION

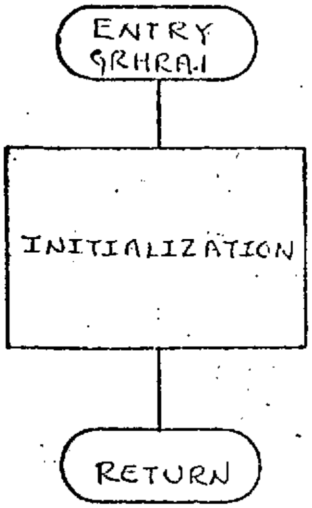
GRHRAN is a real valued DOUBLE PRECISION function which

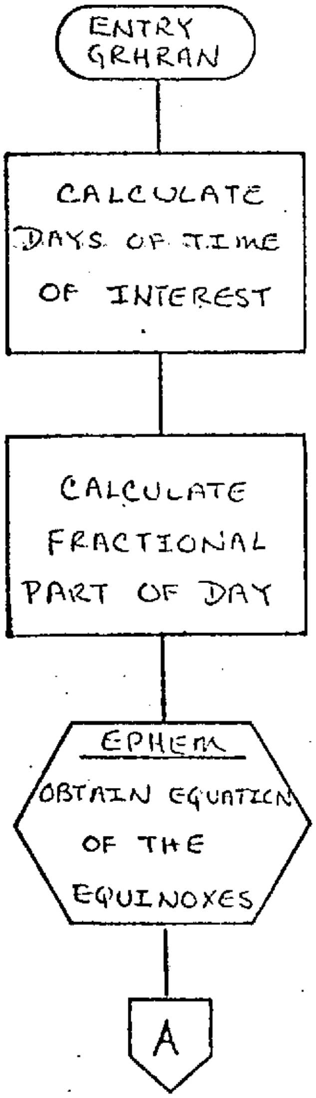
- Computes the right ascension of Greenwich,
- Computes the satellite vectors used in the calculation of the computed measurements and used in the calculation of the measurement partials.

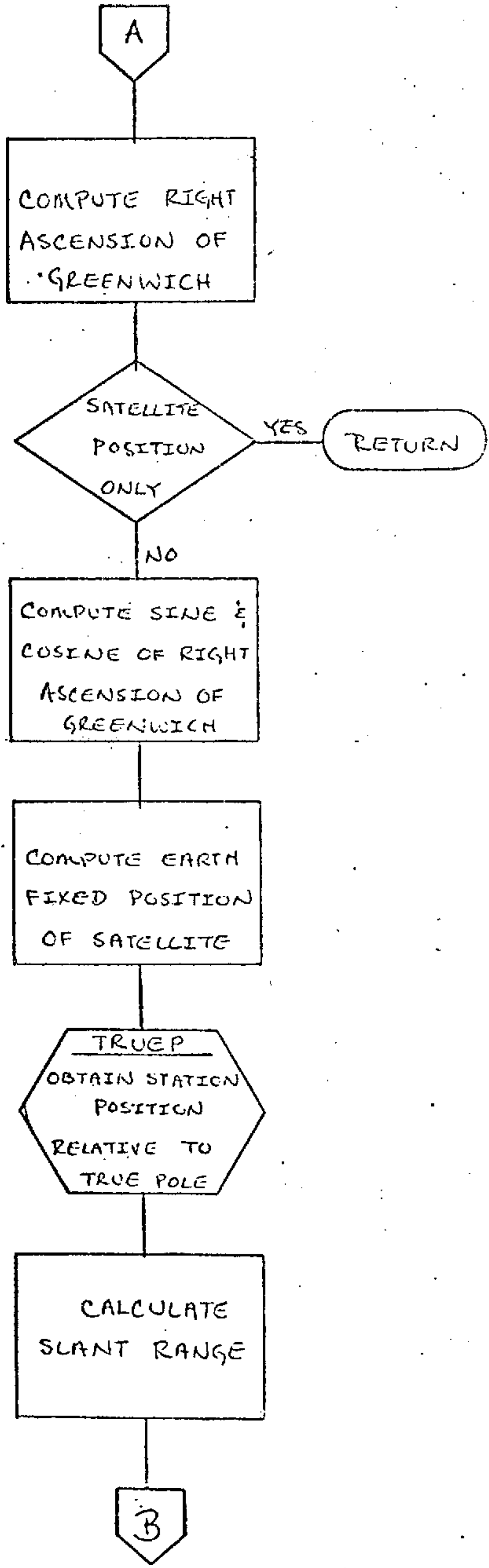
```
NAME                GRHRAN

 ENTRY POINT        PURPOSE

 GRHRA1             INITIALIZATION

 GRHRAN             TO COMPUTE THE RIGHT ASCENSION OF GREENWICH AND
                    THE SATELLITE VECTORS USED IN COMPUTING
                    MEASUREMENT PARTIALS

 CALLING SEQUENCE   X=GRHRA1(ENV,NSTA)

   SYMBOL  TYPE     DESCRIPTION

   ENV      DP      INPUT - STATION UNIT EAST, NORTH, AND VERTICAL
   (3,NSTA,1)               VECTORS

   NSTA     I       INPUT - NUMBER OF TRACKING STATIONS

   GRHRA1   DP      OUTPUT - NOT USED

 CALLING SEQUENCE   X=GRHRAN(DAY,ISTA)

   SYMBOL  TYPE     DESCRIPTION

   DAY      DP      INPUT - OBSERVATION TIME IN DAYS FROM JAN 0.0 OF
                            THE REFERENCE YEAR FOR THE ARC

   ISTA     I       INPUT - STATION INDEX

   GRHRAN   DP      OUTPUT - RIGHT ASCENSION OF GREENWICH

SUBROUTINES USED    EPHEM      TRUEP      XEFIX      YEFIX
                    DOTPRD

COMMON BLOCKS       APARAM     CONSTS     CEPHEM     CSTHET
                    CUVECT     INTBLK     TRUPOL     XYZOUT

INPUT FILES         NONE

OUTPUT FILES        NONE

REFERENCES          'GEODYN SYSTEMS DESCRIPTION'
                    VOLUME 1 - GEODYN DOCUMENTATION
```

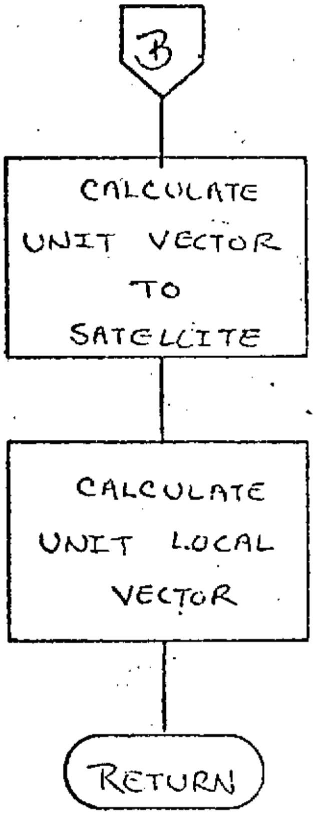
```
      DOUBLE PRECISION FUNCTION GRHRA1(ENV,NSTA)
      IMPLICIT REAL*8 (A-H,O-Z)
      DOUBLE PRECISION MBODY
      DIMENSION ENV(3,NSTA,1)
      COMMON/APARAM/INPAR(4),NSAT,NGPARC(5)
      COMMON/CONSTS/PI,DTWOPI,RAD,RSEC
      COMMON/CEPHEM/A0(24),EQ,GEPHEM(643)
```

```
      COMMON/CSTHET/CTHETG,STHETG
      COMMON/CUVEC1/UHAT(3,2),XYZ(3,2),RXYZ(3,2),RENV(3,2),R(2),RSQ(2),
     *   XYSQ(2)
      COMMON/INTBLK/THDOT1,THDOT2,THDT2S,CM(24),THETG0,MDUDY(33)
      COMMON/TRUPOL/TRUE(3)
      COMMON/XYZOUT/XYZI(6,4)
      RETURN
      ENTRY GRHRAN(DAY,ISTA)
C CALCULATE TIME QUANTITIES NEEDED FOR RIGHT ASCENSION OF GREENWICH
C
C ...DAYS OF TIME OF INTEREST
      T2=IDINT(DAY)
C ...FRACTIONAL PART OF DAY OF TIME OF INTEREST
      T3=DAY-T2
C COMPUTE EQUATIONS OF THE EQUINOXES
      CALL EPHEM(DAY,.TRUE.)
C COMPUTE RIGHT ASCENSION OF GREENWICH
      GRHRAN=THETG0+T2*THDOT1+T3*THDOT2+EC
      GRHRAN=DMOD(GRHRAN,DTWOPI)
C TEST IF ONLY SATELLITE POSITION IS WANTED
      IF(ISTA.EQ.0) RETURN
      STHETG=DSIN(GRHRAN)
      CTHETG=DCOS(GRHRAN)
C OBTAIN STATION POSITION RELATIVE TO TRUE POLE
      CALL TRUEP(DAY,ISTA)
C OBTAIN EARTH FIXED POSITION OF SATELLITE
      DO 350 L=1,NSAT
      XYZ(1,L)=XEFIX(XYZI(1,L),XYZI(2,L))
      XYZ(2,L)=YEFIX(XYZI(1,L),XYZI(2,L))
      XYZ(3,L)=XYZI(3,L)
C CALCULATE SLANT RANGE
      RSQ(L)=0.0D0
      DO 300 I=1,3
      RXYZ(I,L)=XYZ(I,L)-TRUE(I)
      RSQ(L)=RSQ(L)+RXYZ(I,L)**2
  300 IF(I.EQ.2) XYSQ(L)=RSQ(L)
      R(L)=DSQRT(RSQ(L))
C CALCULATE UNIT VECTORS TO SATELLITE
      DO 325 I=1,3
  325 UHAT(I,L)=RXYZ(I,L)/R(L)
C CALCULATE UNIT LOCAL VECTOR
      DO 350 I=1,3
  350 RENV(I,L)=DOTPRD(UHAT(1,L),ENV(1,ISTA,I))
      RETURN
      END
```


ENTRY
GRHRAI
INITIALIZATION
RETURN


ENTRY
GRHRAN
CALCULATE DAYS OF TIME OF INTEREST
CALCULATE FRACTIONAL PART OF DAY
EPHEM
OBTAIN EQUATION OF THE EQUINOXES
A


A
COMPUTE RIGHT ASCENSION OF GREENWICH
SATELLITE POSITION ONLY
YES
RETURN
NO
COMPUTE SINE & COSINE OF RIGHT ASCENSION OF GREENWICH
COMPUTE EARTH FIXED POSITION OF SATELLITE
TRUEP
OBTAIN STATION POSITION RELATIVE TO TRUE POLE
CALCULATE SLANT RANGE
B



REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

NAME	INDENT
PURPOSE	TO COMPUTE X,Y,Z'S AND AREA OF SURFACE DENSITY BLOCKS
CALLING SEQUENCE	CALL INDENT(SIG,ADENSE,DENSE,AREA,CENTER,BLKLOC)

SYMBOL	TYPE	DESCRIPTION
SIG (1)	DP	OUTPUT - SIGMAS OF ADJUSTED SURFACE DENSITY
ADENSE (1)	DP	OUTPUT - A PRIORI VALUES OF ADJUSTED SURFACE DENSITIES
DENSE (1)	DP	OUTPUT - SURFACE DENSITY VALUES
AREA (1)	DP	OUTPUT - SURFACE DENSITY SUB-BLOCK AREAS
CENTER (3,1)	DP	OUTPUT - THE GEOCENTRIC COORDINATES OFTHE SUB-BLOCK CENTERS
BLKLOC (2,1)	DP	OUTPUT - LATITUDE AND LONGITUDE OF BLOCK CENTER

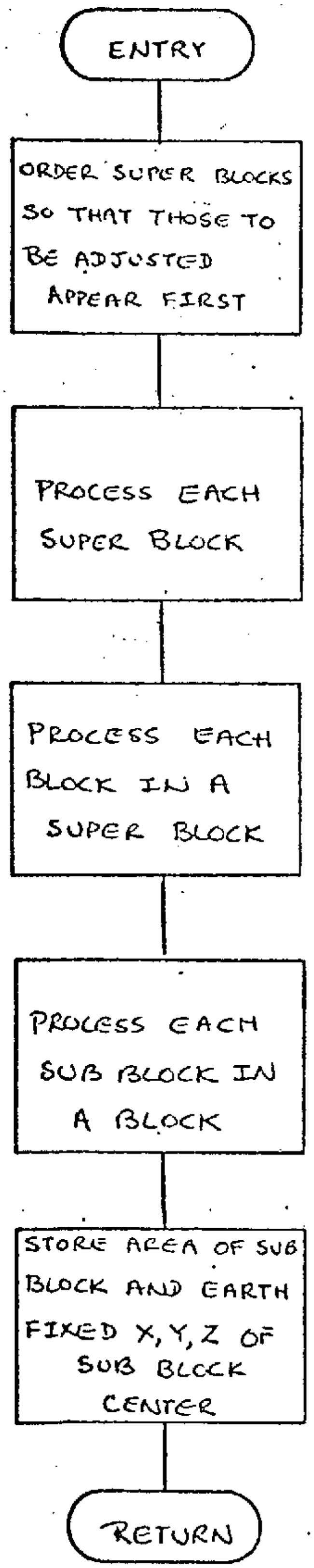
SUBROUTINES USE	DELTAZ	AREAS			
COMMON BLOCKS	CONSTS	CPARAM	FLXBLK	INTBLK	VPBLOK
INPUT FILES	NONE				
OUTPUT FILES	NONE				

```
      SUBROUTINE INDENT(SIG,ADENSE,DENSE,AREA,CENTER,BLKLOC)
      IMPLICIT REAL*8(A-H,O-Z)
      LOGICAL CMPGPS
      INTEGER*2 NLAT,NLON,NEXT
      REAL DLAT,DLON,SIGD,DEN
      DIMENSION SIG(1),ADENSE(1),DENSE(1),AREA(1),CENTER(3,1),
     *   BLKLOC(2,1),NEXT(675)
      COMMON/CONSTS/DPI,DTWOPI,DEG2RD,DRSEC
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,NBIAS,NGPC1,NGPC2,NGPCOM,
     *   NCSEST,CMPGPS,LIM1,LIM2,NBLOCK,NADJ,NTICST,NTICEN,INNFSW,
     *   NCONST,NDCONS
      COMMON/FLXBLK/SLAT(675),SLON(675),DLAT(675),DLON(675)
      COMMON/INTBLK/THDOTS(3),GM,AE,ATSC,FLAT,FS032(59)
      COMMON/VPBLOK/DEN(675),SIGD(675),NLAT(675),NLON(675),NSUPER,
     *   KSURF(224)
      DATA NSBLAT,NSBLON/2,2/
C ORDER THE SUPER BLOCKS SO THAT THOSE TO BE ADJUSTED APPEAR FIRST
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      NSUB=NSBLAT*NSBLON
      F1=1.0D0-FLAT
      F1SQ=F1*F1
      F2F=FLAT*(2.0D0-FLAT)
      NBLOCK=0
      NADJ  =0
      ISUB  =0
      M=0
      DO 10 N=1,NSUPER
      IF(SIGD(N) .EQ. 0.)GO TO 10
      M=M+1
      NEXT(M)=N
   10 CONTINUE
      DO 20 N=1,NSUPER
      IF(SIGD(N) .NE. 0.)GO TO 20
      M=M+1
      NEXT(M)=N
   20 CONTINUE
C THE DO LOOP TO STATEMENT 500 PROCESSES EACH SUPER BLOCK
      DO 500 NS=1,NSUPER
      N=NEXT(NS)
      LAT=NLAT(N)
      LON=NLON(N)
      IF(SIGD(N) .NE. 0.)NADJ=NADJ+LAT*LON
      RLAT  =DLAT(N)
      RLON  =DLON(N)
      RLAT2 =RLAT*.5D0
      RLON2 =RLON*.5D0
      TLAT  =RLAT/DFLOAT(NSBLAT)
      TLAT2=TLAT*0.5D0
      TLON  =RLON/DFLOAT(NSBLON)
      TLON2 =TLON*.5D0
      BEGLAT=SLAT(N)-DFLOAT(LAT)*RLAT2
      BEGLON=SLON(N)-DFLOAT(LON)*RLON2
C THE DO LOOPS TO 400 AND 450 PROCESS EACH BLOCK IN A SUPER BLOCK
      DO 450 LT=1,LAT
      BEGLN =BEGLON
      CENLAT=BEGLAT+RLAT2
      DO 400 LN=1,LON
      NBLOCK=NBLOCK+1
C STORE BLOCK DENSITY AND UNCERTAINTY
      DENSE(NBLOCK) =DEN(N)
      IF(SIGD(N) .EQ. 0.)GO TO 200
      ADENSE(NBLOCK)=DEN(N)
      SIG(   NBLOCK) =SIGD(N)
C STORE LATITUDE AND LONGITUDE OF BLOCK CENTER
      BLKLOC(1,NBLOCK)=CENLAT
      CENLON=BEGLN+RLON2
      IF(CENLON .GE. .36D3)CENLON=CENLON-.36D3
      IF(CENLON .LT. .0D3)CENLON=CENLON+.36D3
      BLKLOC(2,NBLOCK)=CENLON
  200 CONTINUE
      SUBLAT=BEGLAT
      ZORGIN=DELTAZ(SUBLAT)
C THE DO LOOP TO 300 AND 350 PROCESS EACH SUBBLOCK IN A BLOCK
      DO 350 LTSUB=1,NSBLAT
```

```
      ZEND=DELTAZ(SUBLAT+TLAT)                                          INDE 112
      PSI=SUBLAT+TLAT2                                                  INDE 113
      ARIA=AREAS(ZBEGIN,ZEND,TLON)                                      INDE 114
      CLON   =BEGLN+TLON2                                               INDE 115
      SPSI=DSIN(PSI*DEG2RD)                                             INDE 116
      SPSISQ=SPSI**2                                                    INDE 117
      CPSI=DSQRT(1.0D0-SPSISQ)                                          INDE 118
      RHO=AE*F1/DSQRT(F1SQ+F2F*SPSISQ)                                  INDE 119
      RCL=RHO*CPSI                                                      INDE 120
      RSL=RHO*SPSI                                                      INDE 121
      DO 300 LNSUB=1,NSBLON                                             INDE 122
      ISUB=ISUB+1                                                       INDE 123
C STORE AREA OF SUB-BLOCK AND EARTH FIXED X,Y,Z'S OF SUB-BLOCK CENTER   INDE 124
      AREA(ISUB)=ARIA                                                   INDE 125
      CENTER(1,ISUB)=RCL*DCOS(CLON*DEG2RD)                              INDE 126
      CENTER(2,ISUB)=RCL*DSIN(CLON*DEG2RD)                              INDE 127
      CENTER(3,ISUB)=RSL                                                INDE 128
      CLON=CLON+TLON                                                    INDE 129
  300 CONTINUE                                                          INDE 130
      ZBEGIN=ZEND                                                       INDE 131
      SUBLAT=SUBLAT+TLAT                                                INDE 132
  350 CONTINUE                                                          INDE 133
      BEGLN=BEGLN+BLON                                                  INDE 134
  400 CONTINUE                                                          INDE 135
      BEGLAT=BEGLAT+BLAT                                                INDE 136
  450 CONTINUE                                                          INDE 137
  500 CONTINUE                                                          INDE 138
      RETURN                                                            INDE 139
      END                                                               INDE 140
```


ENTRY
ORDER SUPER BLOCKS SO THAT THOSE TO BE ADJUSTED APPEAR FIRST
PROCESS EACH SUPER BLOCK
PROCESS EACH BLOCK IN A SUPER BLOCK
PROCESS EACH SUB BLOCK IN A BLOCK
STORE AREA OF SUB BLOCK AND EARTH FIXED X, Y, Z OF SUB BLOCK CENTER
RETURN

INOUPT

DESCRIPTION

INOUPT is a subroutine specifically designed as a part of the Multi-Arc GEODYN program.

INOUPT performs four basic tasks for GEODYN. The tasks that are performed by INOUPT are the following:

1) INOUPT reads the GEODYN Input Cards which define parameters common to all arcs.

2) INOUPT prints the GEODYN run heading.

3) INOUPT reads the GEODYN INput Cards which define parameters for a single arc.

4) INOUPT prints a heading for each arc.

INOUPT performs tasks 1 & 2 on the first entry and performs tasks 3 & 4 on all subsequent entries.

```
NAME               INOUPT

PURPOSE            1) TO READ OPTIONAL CARDS
                   2) TO PRINT RUN DESCRIPTIONS
                   3) TO CALL SUBROUTINES TO READ DATA TAPES

CALLING SEQUENCE   CALL INOUPT(NARCS,ARCNO)

   SYMBOL  TYPE    DESCRIPTION

   NARCS    I      INPUT - THE NUMBER OF ARCS IN THE RUN

   ARCNO    I      INPUT - THE ARC NUMBER

SUBROUTINES USED   TDIF      YMDAY     DENORM    PRNTPR    CLEAR
                   NUMBR2    COEFL     DCOSRD    DOTPRD    ELEM
                   PDEN1     GEOSRL    DAVERT    DATES     PCERD
                   ERROR     SYMINV    SIMRD

COMMON BLOCKS      ALPHRC    APARAM    CELEM     CEPHEM    CGEOS
                   CONOUT    CONSTS    CEPB1     CPARAM    CTIME
                   FLXBLK    FMODEL    INITBK    INTBLK    PREBLK
                   PRIORI    SIGBLK    SRFBLK    TPEBLK    VMAT
                   VRBLOK

INPUT FILES        INTP - INPUT CARDS

OUTPUT FILES       OUTP - PRINTER

SCRATCH FILES      SCRC - 16
```

```
      SUBROUTINE INOUPT(/NARCS/,/ARCNO/)                                INOU  35
      IMPLICIT REAL*8(A-H,O-Z)                                          INOU  36
      LOGICAL GRDFSW,KEPLER,TOPERT,VARSTP,OBBTSW,XYZFSW,XYZLSW,TDRLFC,  INOU  37
     .   PLTLSW,NOPRNT,INITAL,NEWCEL,SUBSAT,CMPGPR,PARTGP,PBMAT,SIMDAT, INOU  38
     .   PCESIN,HLVLSW,LITRES,STARTR,STARTW,HYPER,SEQUEN                INOU  39
      LOGICAL*1 VHFCHN,PREPRO,PLHSW,NAME8,NAME7                         INOU  40
      INTEGER*2 IPREPR,INOPRT,CULL,ISTNC,IMTYPE,DRAGNO,EMISNO,          INOU  41
     .   BTYPE,DSTANO,ESTANO,ISTANO,ISTARC,MTYPE,NMFAS,PRETYP,CHANEL,   INOU  42
     .   INDXCS,NP,NL,OBSTNC,BETYPE                                     INOU  43
      INTEGER OUTP,RVTP,CUSUPR,XYZTP,ARCNU,ADDR,SSAD,FSTSTA,GROTP,RECNO,INOU  44
     .   DATP,FLTP,SCRA,PLOTP,SCRC,ADDRD,PRCNOI,CFUUR,PMATNO,STARTA,    INOU  45
     .   STARTO,OUTSTR,PARMAX                                           INOU  46
      REAL RAD,RSEC,GPSIG,RMSTOT,SIGCHG,DASH,AITIME,PLHSIG,             INOU  47
     .   SIGSTD,VARCOV,S15,PI,RATE,TDIF,BIASC,EDITN,TWOPI,BIASSG,       INOU  48
     .   CONVRG,REINDX,SGPRNT,CUTCCN,SD,SSD,DP,DL,DSEC                  INOU  49
      DOUBLE PRECISION MODEL,NAME,JNAME,ITNMS,MFOTY,MSAT,LCVS           INOU  50
      DIMENSION CONAML(32),PLHSW(1),PLHSIG(6,1),INDXCS(960,3),          INOU  51
     .   ISURF(4056),GPSIG(960),POOLS(6),CONINT(3,2),TIMES(7),          INOU  52
     .   GPVALU(960),A(6,5),NAME6(6),ESTINO(575),PLTYPE(675),DT15(5),   INOU  53
     .   KTAPES(4)                                                      INOU  54
      COMMON/ALPHRC/ITNMS(5),TIMING,BLANK,ATYPE(31),UNITS(15),PLCUT,    INOU  55
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
     *    HYPER                                                           INOU  56
      COMMON/APARAM/INPAR,INPARI,NBIAS,ESTSTA,NSAT,NGPARC,RECND1,         INOU  57
     *    NPARAM,NBIASE,PARMAX                                            INOU  58
      COMMON/CELEM/ELEMST(6,2),ORBELA(6,2),XNU,IC,EMSTCT                  INOU  59
      COMMON/CEPHEM/JNAME(381),ISTARC(381),ESTANO(381),ISTANO(386)        INOU  60
      COMMON/CGEOS/ISATID(2),IPREPR(4,50),RFINDX(2,50),INDPRE(2,50),      INOU  61
     *    NCPRPR,NSIG,NCULL,SIGCHG(50),IMTYPE(50),ISTNO(50),CULL(2,100)   INOU  62
      COMMON/CONOUT/IG1,OUTCON,MINCUT,MAXOUT,LITRES,MAXSAT(7),NSTARC,     INOU  63
     *    STARTP,STARTW,STARTA,STARTC,INSTRT,OUTSTR                       INOU  64
      COMMON/CONSTS/DPI,DTWOPI,DRAD,DRSEC                                 INOU  65
      COMMON/CORB1/RANDCT(2),PERDCT(2),PERHT(2),APHT(2),PER(2)            INOU  66
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NCIN,NBIAS,NGPC1,NGPC2,             INOU  67
     *    NGPCCM,NCSEST,CMPGPE,LIM1,LIM2,NDEN,NDENST,NTICST,NTIDEN,       INOU  68
     *    INNRSW,NCONST,NOCONS                                            INOU  69
      COMMON/CTIME/DATAEP,DAYREF,DSTART,DAYSTP,DAYINT,DORBIT,DAYEND,      INOU  70
     *    DRATE,DORB1,DORB1E,DRBRT,IYREF                                  INOU  71
      COMMON/FLXBLK/BSTRT(900),BSEND(900),BTYPE(900)                      INOU  72
      COMMON/FMODEL/INDEX1,NDEX2,INDEX3,NDEX4,CS(30,33),MODEL(8)          INOU  73
      COMMON/INITBK/IEPYMD,IEPHM,EPSEC,IYREF,INNMAX,INNMIN,CONVRG,        INOU  74
     *    ORBEL(6,2),EDITN,INSURF,ICSAT(2),ORBTSW,XYZFSW,XYZLSW,PLTLSW,   INOU  75
     *    GRDFSW,KEPLER,SUBSAT,PARTGP,REMAT,EMATNO,SIMDAT,PCESIM,         INOU  76
     *    MISLOG(9)                                                       INOU  77
      COMMON/INTBLK/THDOT1,THDOT2,THDT2S,GM,AE,AESQ,FLAT,                 INOU  78
     *    FSQ32,FFSQ32,GM3(6),B(2),BDOT(2),RC(2),APGM(2),APLM(2),RPRESS,  INOU  79
     *    INITAL,NCFRAT,THETGO,NBODY(6),STEPSZ(2,2),HLVERB(2),RPLERR(2),  INOU  80
     *    CTOL(2),RTOL(2),STPLOW(2),STPUP(2),ORDER(2,2),ASAT(2),          INOU  81
     *    MSAT(2),VARSTP(2),HLVDSW(2),NCON(2),ADDF(2),ADDRD(2),SRAD(2),   INOU  82
     *    LOW(3),TOFFFT,NBODY                                             INOU  83
      COMMON/PREBLK/DAYSTA,OBS1,OBS2,SIG1,SIG2,SPENCX,ISTA,               INOU  84
     *      MTYPE,NMEAS,PRETYP(2),CHANEL,VHFCHN,PREPRO,RECNO              INOU  85
      COMMON/PRIORI/ELEMIN(6,2),VARCOV(6,6,2),TITLE(20),DRAGSG(2),        INOU  86
     *    DRGDSG(2),EMISSG(2),DRAGO(2),DRGDO(2),EMISSO(2),CD(2),          INOU  87
     *    CDD(2),EMISS(2)                                                 INOU  88
      COMMON/SIGBLK/SIGSTD(30),SGPRNT(30),IARRAY(4)                       INOU  89
      COMMON/SRFBLK/PHI(675),XLAM(675),CP(675),CL(675),SD(675),           INOU  90
     *    SSD(675),NP(675),NL(675),NSD                                    INOU  91
      COMMON/TPEBLK/INTP,OUTP,DATP,XYZTP,KEPTAP,RVTP,PLOTP,IOBS,SCRA,     INOU  92
     *    SCRC,FLTP,GRDTP                                                 INOU  93
      COMMON/VMAT/PKEPLR(6,6),SUM1(27)                                    INOU  94
      COMMON/VRBLCK/BIASO(900),BIASSG(900),ESTANO(900)                    INOU  95
      EQUIVALENCE (RSEC,DRSEC),(JSTANC,RANDCT(1)),                        INOU  96
     *    (INDXCS(1,1),BSTRT(1)),(GPSIG(1),BIASO(1)),(PLHSIG(1,1),        INOU  97
     *    BSTRT(721)),(PLHSW(1),BTYPE(253))                               INOU  98
      EQUIVALENCE (ISURF(1),PHI(1)),(CARDID,NAME8(1)),(NAME7,NAME8(7))    INOU  99
      EQUIVALENCE (HRSTNO(1),NP(1)),(RETYPE(1),NL(1)),(DT1,DT15(1)),      INOU 100
     *    (T2,DT15(2)),(T3,DT15(3)),(T4,DT15(4)),(T5,DT15(5))             INOU 101
      DATA NCARDS/32/,DASH/1H-/                                           INOU 102
      DATA CONAME/6HORBIT ,6HEARTH ,6HSAT   ,6HTIDES ,6HBODIES,6HDRAG    ,INOU 103
     *    6HSOLRAD,6HOUTPUT,6HSIGMA ,6HSTEP  ,6HSTAEST,6HCORREL,6HBIAS   ,INOU 104
     *    6HVARCOV,6HEDIT  ,6HRESID ,6HPREPRO,6HCARD1 ,6HCULL  ,6HFLUX   ,INOU 105
     *    6HSTAPOS,6HMODEL ,6HDATA  ,6HEMATRX,6HSIMDAT,6HTOLS  ,6HRSTART,INOU 106
     *    6HSURF  ,6HLLCUT ,6HCOEF  ,6HEBIAS ,6HSTDDAT/                   INOU 107
      DATA CONINT/8HEQUATION,8HS OF MOT,3HION,8HVARIATIO,8HNAL EQUA,      INOU 108
     *    5HTIONS/                                                        INOU 109
      DATA BODIES/6H LUNAR,6H SOLAR,6H VENUS,7HMARTIAN,7HJUPITER,         INOU 110
     *    6HSATURN/                                                       INOU 111
```

```
      DATA TIDES/5HLUNAR,5H AND ,5HSOLAR,5H EFFE,5HCTS I,5HNCLUD,2HED/
      DATA HERTZ/5HHERTZ/,SEQUEN/.FALSE./,KTAPES/4*0/
      A1TIME(ISTART)=TDIF(4,3,DSTART)/8.64D4
      DO 50 I=1,30
   50 SGPRNT(I)=SIGSTD(I)
      IELM=MTYPE
      IF(ARCNO.NE.0) GO TO 100
C INITIALIZE FOR COMMON OPTIONS
      LARC=NARCS
      STARTR=.FALSE.
      STARTW=.FALSE.
      STARTA=1
      STARTO=1
      NDXOBG=0
      NDXOBD=0
      INSUPR=3
      NSTARD=0
      NSD=0
      NDCONS=0
      CALL CLEAR(JNAME,1336,1)
      CALL CLEAR(ESTRT,4050,1)
      CALL CLEAR(BIAS0,2250,1)
      NSTEST=0
      RMSTOT=1000.
      NRMCFL=.FALSE.
      GO TO 200
C INITIALIZE FOR ARC OPTIONS
  100 LOOPS=NARCS
      IF(ARCNO.EQ.LARC) SEQUEN=.FALSE.
      ELCUT=0.0D0
      NDXOBG=NDXOM1
      ISATID(1)=0
      ISATID(2)=0
      NDXOBD=NDXOM2
      NGPARC=900
      RECNO1=RECNO
C SET DRAG & INTEGRATION DEFAULTS
      DO 130 I=1,NSAT
      IF(PERHT(I).GT.8.0D2) GO TO 120
      IF(ORBELA(2,I).GT.1.0D0.AND.DOTPRD(ELEMST(1,I),ELEMST(1,I))
     .    .GT.51.5259D+12) GO TO 120
      CD(I)=2.3D0
      DRAGO(I)=2.3D0
      DRAGSG(I)=0.5D0
  120 IF(ORBELA(2,I).LT.1.0D-2.AND.PERHT(I).GT.1.5D3) STEPSZ(I,1)=
     .    PERHT(I)*3.649557D-3+86.72566D0
      CDD(I)=0.0D0
      DRGDO(I)=0.0D0
      DRGDSG(I)=0.0D0
      VARSTP(I)=ORBELA(2,I).GT.0.2D0
      IF(VARSTP(I)) STEPSZ(I,1)=25.0D0
      IF(DRAGO(I).GT.0.0D0.AND..NOT.VARSTP(I)) STEPSZ(I,1)=75.0D0
      IF(STEPSZ(I,1).GT.4.0D2) STEPSZ(I,1)=4.0D2
  130 STEPSZ(I,2)=STEPSZ(I,1)
C READ OPTION CARD
  200 READ(INTP,10004,END=900) CARDID,IARRAY,DT1,T2,T3,T4,T5
```

```
C DETERMINE TYPE OF CARD READ
      DO 250 I=1,NCARDS
      IF(CARDID.EQ.CONAME(I)) GO TO(400,405,410,415,420,425,430,435,
     .   440,445,450,455,460,470,480,485,490,475,495,200,550,500,525,
     .   300,310,320,340,350,360,500,370,380),I
 250  CONTINUE
C ILLEGAL OPTION CARD - PRINT ERROR MESSAGE AND STOP
      CALL ERROR(3,CARDID)
C BMATRX CARD
  300 BMATNO=DT1+0.5D0
      PBMAT=T2.GT.0.5D0
      GO TO 200
C SIMDAT CARD
  310 SIMDAT=.TRUE.
      PCSSIM=IARRAY(1).EQ.0
      GO TO 200
C TOLS CARD
  320 J=IARRAY(1)
      K=IARRAY(3)*10+IARRAY(4)
      L=T3+0.5D0
      IF(ARCNO.EQ.0) NOCONS=L
      IF(K.GT.0) NDEX4=MIN0(30,MAX0(3,K))
      IF(J.EQ.0) GO TO 325
      IF(J.GT.2) GO TO 325
      IF(DT1.GT.0.0D0) RTOL(J)=DT1
      IF(T2.GT.0.0D0) CTOL(J)=T2
      GO TO 200
  325 DO 326 J=1,2
      IF(DT1.GT.0.0D0) RTOL(J)=DT1
      IF(T2.GT.0.0D0) CTOL(J)=T2
  326 CONTINUE
      GO TO 200
C RSTART CARD
  340 IF(ARCNO.NE.0) GO TO 200
      INSTRT=IARRAY(1)*10+IARRAY(2)
      OUTSTR=IARRAY(3)*10+IARRAY(4)
      STARTA=DT1+0.5D0
      STARTO=T2+0.5D0
      STARTA=MAX0(1,STARTA)
C     STARTO=MAX0(STARTO,1)
      STARTO=1
      STARTP=INSTRT.GT.0
      STARTW=OUTSTR.GT.0
      GO TO 200
C SURFAN CARD
  350 IF(ARCNO.NE.0) GO TO 200
      NSD=NSD+1
      SD(NSD)=DT1
      SSD(NSD)=DABS(T2)
      PHI(NSD)=T3
      XLAM(NSD)=T4
      READ(INTP,1004,END=900) CARDID,IARRAY,DT1,T2,T3,T4,T5
      IF (CARDID.NE.BLANK) CALL ERROR(14,CONAME(24))
      NP(NSD)=MAX0(IARRAY(1)*10+IARRAY(2),1)
      NL(NSD)=MAX0(IARRAY(3)*10+IARRAY(4),1)
      IF (DT1.LE.0.0D0) CALL ERROR(15,CARDID)
```

```
      DP(NSD)=DT1
      IF (T2.LE.0.0D0) CALL ERROR(15,CARDID)
      DL(NSD)=T2
      J=NP(NSD)*NL(NSD)
      NDEN=NDEN+J
      IF(SSD(NSD).GT.0.0) NDENST=NDENST+J
      GO TO 200
C ELCUT CARD
  360 ELCUT=DT1
      GO TO 200
C EBIAS CARD
  370 IF(ARGNO.EQ.0) GO TO 200
      J=IARRAY(1)*1000+IARRAY(2)*100+IARRAY(3)*10+IARRAY(4)
      IF(J.LE.0) GO TO 200
      DO 375 I=1,5
      K=DT15(I)+0.5D0
      IF(K.LE.0) GO TO 375
      IF(K.GT.14) GO TO 375
      IF(NBIASE.GE.675) GO TO 200
      NBIASE=NBIASE+1
      BESTNO(NBIASE)=J
      BETYPE(NBIASE)=K
  375 CONTINUE
      GO TO 200
C SEQDAT CARD
  380 IF(ARGNO.EQ.0) SEQUEN=.TRUE.
      GO TO 200
C ORBIT CARD
 400  ORBTSW=.TRUE.
      DRAGSG(1)=0.0D0
      DRAGSG(2)=0.0D0
      DRGDSG(1)=0.0D0
      DRGDSG(2)=0.0D0
      XYZFSW=.TRUE.
      RVTP=10*IARRAY(1)+IARRAY(2)
      BSEC=EPSEC
      IF(RVTP.GT.0) WRITE(RVTP) DSTART,IEPYMD,IEPHM,BSEC,(ELEMST(I,1),
     .    I=1,6)
      IYSTRT=DT1+.5D0
      IYEND=T3+.5D0
      IHM=T2/100.D0+1.D-4
      SEC=T2-DFLOAT(IHM*100)
      DORBIT=YMDAY(IYSTRT,IHM,SEC)
      DORBIT=DORBIT+A1TIME(DORBIT)
      DATAEP=DMIN1(DATAEP,DORBIT)
      IHM=T4/100.D0+1.D-4
      SEC=T4-DFLOAT(IHM*100)
      DAYEND=YMDAY(IYEND,IHM,SEC)
      DAYEND=DAYEND+A1TIME(DAYEND)
      DAYSTP=DAYEND
      GO TO 200
C EARTH CARD
  405 J=10*IARRAY(1)+IARRAY(2)
      IF(J.GT.0) INDEX1=J
      J=10*IARRAY(3)+IARRAY(4)
      IF(J.GT.0) INDEX3=J
```

```
      INDEX1=MAX0(4,INDEX1)
      INDEX3=MAX0(2,INDEX3)
      INDEX3=MIN0(INDEX3,INDEX1)
      IF(ARONE.NE.0) GO TO 200
      IF(DT1.GT.0.D00) GM=DT1
      IF(T2.GT.0.D0) AE=T2
      IF(T3.GT.0.D0) FLAT=T3
      NRMCFL=T5.GT.0.D0
      IF(T4.LE.0.D0) GO TO 200
      DO 406 J=1,33
      DO 406 K=1,33
  406 CS(J,K)=0.D0
      GO TO 200
C SAT CARD
  410 J=IARRAY(1)
      IF(J.EQ.0) GO TO 412
      IF(J.GT.2) GO TO 412
      ASAT(J)=DT1
      MSAT(J)=T2
      ISATID(J)=T3+0.5D0
      GO TO 200
  412 DO 413 J=1,NSAT
      ASAT(J)=DT1
      MSAT(J)=T2
  413 ISATID(J)=T3+0.5D0
      GO TO 200
C TIDES CARD
  415 IF(ARONE.NE.0) GO TO 200
      LOVE(1)=DT1
      LOVE(2)=T2
      LOVE(3)=T3
      GO TO 200
C BODIES CARD
  420 J=IARRAY(1)
      K=IARRAY(2)
      IF(K.GT.0) NBODY=MIN0(6,MAX0(2,K))
      IF(J.EQ.0.OR.J.GT.6) GO TO 200
      MBODY(J)=DT1
      GO TO 200
C DRAG CARD
  425 J1=1
      J2=NSAT
      J=IARRAY(1)
      IF(J.EQ.0.OR.J.GT.2) GO TO 426
      J1=J
      J2=J
  426 DO 428 J=J1,J2
      CD(J)=DT1
      DRAGC(J)=DT1
      DRAGSS(J)=T2
      IF(DT1.GT.0.D0) GO TO 427
      T3=0.D0
      T4=0.D0
  427 CDD(J)=T2
      DRGCD(J)=T3
  428 DRGCSS(J)=T4
```

```
      GO TO 200
C SOLRAD CARD
  430 J1=1
      J2=NSAT
      J=IARRAY(1)
      IF(J.EQ.0.OR.J.GT.2) GO TO 431
      J1=J
      J2=J
  431 DO 432 J=J1,J2
      EMISS(J)=DT1
      EMISSD(J)=DT1
  432 EMISSG(J)=T2
      GO TO 200
C OUTPUT CARD
  435 RATE=DT1
      IF(DT1.LE.0.D0) GO TO 436
      DRATE=DT1+DT1*(A1TIME(DSTART+7.D0)-A1TIME(DSTART))/7.D0
      DRATE=DRATE/8.64D4
  436 IF(IARRAY(1).EQ.0.AND.IARRAY(2).EQ.0) GO TO 437
      XYZFSW=IARRAY(1).EQ.1.OR.IARRAY(1).EQ.3
      XYZLSW=IARRAY(1).EQ.2.OR.IARRAY(1).EQ.3
      GPDFSW=IARRAY(2).GT.0
 437  PLTLSW=IARRAY(3).GT.0
      TORBFT=T2.GT.0.D0
      KEPLER=IARRAY(1).GT.0.AND.IARRAY(4).GT.0
      J=T3+0.5D0
      IF(J.EQ.6.OR.J.EQ.8) KEPTAP=J
      PARTGP=T4.GT.0.D0
      GO TO 200
C SIGMA CARD
  440 NSIG=NSIG+1
      ISTA=DT1+.5D0
      MTYPE=T2+.5D0
      ISTNO(NSIG)=ISTA
      IMTYPE(NSIG)=MTYPE
      SIGCHG(NSIG)=T3
      IF(ISTA.GT.0) GO TO 200
      IF(MTYPE.LE.0) GO TO 442
      SGPRNT(MTYPE)=T3
      GO TO 200
  442 DO 443 I=1,30
  443 SGPRNT(I)=T3
      GO TO 200
C STEP CARD
  445 J=IARRAY(1)
      IORDER=IARRAY(3)*10+IARRAY(4)
      IF(IORDER.LE.0) GO TO 448
      IORDER=MIN0(15,MAX0(5,IORDER))
  448 J1=1
      J2=4
      IF(J.EQ.0.OR.J.GT.4) GO TO 446
      J1=J
      J2=J
  446 DO 447 J=J1,J2
      IF(DT1.GT.0.D0) STEPSZ(J,1)=DT1
      IF(IORDER.GT.0) ORDER(J,1)=IORDER
```

```
      IF(J.GT.2) GO TO 447                                              INOU 392
      IF(T2.GT.0.0D0) HLVERB(J)=T2                                      INOU 393
      IF(T3.GT.0.0D0) DBLERB(J)=T3                                      INOU 394
      IF(T4.GT.0.0D0) STEPUP(J)=T4                                      INOU 395
      IF(T5.GT.0.0D0) STPLOW(J)=T5                                      INOU 396
      VARSTP(J)=IARRAY(2).GT.0                                          INOU 397
      HLVDSW(J)=IARRAY(2).EQ.2                                          INOU 398
  447 CONTINUE                                                          INOU 399
      GO TO 200                                                         INOU 400
C STAEST CARD                                                           INOU 401
  450 IF(ARCNC.NE.0) GO TO 200                                          INOU 402
      ISN=DT1+0.5D0                                                     INOU 403
      IST=NUMBR2(ISN,ISTANO,NSTEST)                                     INOU 404
      IF(IST.GT.0) GO TO 452                                            INOU 405
      NSTEST=NSTEST+1                                                   INOU 406
      IST=NSTEST                                                        INOU 407
452   ISTANO(IST)=ISN                                                   INOU 408
      ESTANO(IST)=ISN                                                   INOU 409
      JSN=T5+0.5D0                                                      INOU 410
      IF(JSN.LE.0.OR.JSN.EQ.ISN) GO TO 453                              INOU 411
      ESTANO(IST)=JSN                                                   INOU 412
      GO TO 200                                                         INOU 413
  453 PLHSW(IST)=IARRAY(1).NE.1                                         INOU 414
      NMAST=NMAST+1                                                     INOU 415
      PLHSIG(3,IST)=T4                                                  INOU 416
      IF(PLHSW(IST))GO TO 454                                           INOU 417
      PLHSIG(1,IST)=T2                                                  INOU 418
      PLHSIG(2,IST)=T3                                                  INOU 419
      GO TO 200                                                         INOU 420
  454 PLHSIG(1,IST)=T2*RSEC                                             INOU 421
      PLHSIG(2,IST)=T3*RSEC                                             INOU 422
      GO TO 200                                                         INOU 423
C CORREL CARD                                                           INOU 424
  455 ISN=DT1+.5D0                                                      INOU 425
      IST=0                                                             INOU 426
      IF(NSTEST.GT.0)IST=NUMBR2(ISN,ISTANO,NSTEST)                      INOU 427
      IF(IST.GT.0) GO TO 457                                            INOU 428
      NSTEST=NSTEST+1                                                   INOU 429
      IST=NSTEST                                                        INOU 430
457   ISTANO(IST)=ISN                                                   INOU 431
      PLHSIG(4,IST)=T2                                                  INOU 432
      PLHSIG(5,IST)=T3                                                  INOU 433
      PLHSIG(6,IST)=T4                                                  INOU 434
      GO TO 200                                                         INOU 435
C BIAS CARD                                                             INOU 436
  460 IF(ASCNC.LE.0) GO TO 465                                          INOU 437
      NBIAS=NBIAS+1                                                     INOU 438
      BIASC(NBIAS)=DT1                                                  INOU 439
      BIASSG(NBIAS)=T2                                                  INOU 440
      IYMD=T3/1.0D6+1.0D-8                                              INOU 441
      T3=T3-1.0D6*DFLOAT(IYMD)                                          INOU 442
      IHM=T3/1.0D2+1.0D-4                                               INOU 443
      SEC=T3-1.0D2*DFLOAT(IHM)                                          INOU 444
      IF(IYMD.GT.0)BSTRT(NBIAS)=YMDAY(IYMD,IHM,SEC)+A1TIMC(DSTART)      INOU 445
      IF(IYMD.EQ.0) BSTRT(NBIAS)=DATALF                                 INOU 446
      IYMD=T4/1.0D6+1.0D-8                                              INOU 447
```

```
      T4=T4-1.0D6*DFLOAT(IYMD)
      IHM=T4/1.0D2+1.0D-4
      SEC=T4-1.0D2*DFLOAT(IHM)
      IF(IYMD.GT.0)BSEND(NBIAS)=YMDAY(IYMD,IHM,SEC)+AITIME(DSTART)
      IF(IYMD.EQ.0) BSEND(NBIAS)=DATAEP
      BSTAND(NBIAS)=1000*IARRAY(1)+100*IARRAY(2)+10*IARRAY(3)+IARRAY(4)
      BTYPE(NBIAS)=T5+0.1D0
      GO TO 200
  465 J=1
      WRITE(OUTP,44600) CARDID,J
      CALL ERROR(10,CARDID)
      GO TO 200
C VARCOV CARD
  470 J=IARRAY(1)
      K=IARRAY(2)
      L=IARRAY(3)
      J1=1
       J2=2
      IF(J.EQ.0.OR.J.GT.2) GO TO 471
      J1=J
      J2=J
  471 DO 474  J=J1,J2
      IF(L.GT.0) GO TO 472
      READ(INTP,10003) (VARCOV(N,N,J),N=1,5)
      GO TO 473
  472 READ(INTP,10003) ((VARCOV(N,M,J),M=1,6),N=1,6)
  473 IF(K.EQ.0) GO TO 474
      CALL ELEM(ELEMST(1,J),A,3,.TRUE.,PKEPLR)
      CALL DNVERT(6,PKEPLR,6,SUM1)
      DO 467 I=1,6
      DO 467 N=1,6
      A(I,N)=0.0D0
      DO 467 M=1,6
  467 A(I,N)=A(I,N)+PKEPLR(I,M)*VARCOV(M,N,J)
      DO 469 I=1,6
      DO 469 N=1,6
      PSUM=0.0D0
      DO 468 M=1,6
  468 PSUM=PSUM+A(I,M)*PKEPLR(N,M)
  469 VARCOV(I,N,J)=PSUM
  474 CONTINUE
      GO TO 200
C ORB1 CARD
  475 IYSTRT=DT1+.5D0
      IYEND=T3+.5D0
      IHM=T2/100.D0+1.D-4
      SEC=T2-DFLOAT(IHM*100)
      DORB1=YMDAY(IYSTRT,IHM,SEC)
      DORB1=AITIME(DORB1)+DORB1
      TORBFD=IARRAY(1).GT.0
      IHM=T4/100.D0+1.D-4
      SEC=T4-DFLOAT(IHM*100)
      DORB1E=YMDAY(IYEND,IHM,SEC)
      DORB1E=DORB1E+AITIME(DORB1E)
      ORBRT=0.0D0
      DO 476 I=2,4
```

```
      [illegible]
  476 ORBRT=ORBRT*10.D0+DFLOAT(IARRAY(I))
      IF(ORBRT.LE.0.D0) ORBRT=60.D0
      IDSAT(1)=T5+0.5D0
      ORBRT=ORBRT+DRGRT*(AITIME(ICHR1+7.D0)-AITIME(ICORE1))/7.D0
      ORBRT=ORBRT/864.D2
      GO TO 200
C EDIT CARD
  480 IF(DT1.GT.0.D0) EDITN=DT1
      IF(T2.GT.0.D0) RMSTOT=T2
      GO TO 200
C RESID CARD
  485 INSUPP=IARRAY(1)
      IF(INSUPP.GT.4) INSUPP=3
      GO TO 200
C PREPRO CARD
 490  NOPRPR=NOPRPR+1
      DO 491 I=1,4
 491  IPREPR(I,NOPRPR)=IARRAY(I)
      INDPRP(1,NOPRPR)=DT1+.1D0
      INDPRP(2,NOPRPR)=T2+.1D0
      RFINDX(1,NOPRPR)=T3
      RFINDX(2,NOPRPR)=T4
      GO TO 200
C CULL CARD
  495 NCULL=NCULL+1
      CULL(1,NCULL)=DT1+.5D0
      CULL(2,NCULL)=T2+.5D0
      IF(T3.EQ.0.D0) GO TO 200
      NCULL=NCULL+1
      CULL(1,NCULL)=T3+.5D0
      CULL(2,NCULL)=T4+.5D0
      GO TO 200
C RECODE (COEF) CARD
  500 N=IARRAY(1)*10+IARRAY(2)
      M=IARRAY(3)*10+IARRAY(4)
      T3=DABS(T3)
      T4=DABS(T4)
      IF(N.EQ.0) GO TO 502
      K=2
      IF(M.EQ.0.AND.N.LT.3) K=1
      TSIG=3.163D-3**K
      IF(T5.GE.0.D0) GO TO 503
      T5=DENORM(N,M)
      DT1=DT1*T5
      T2=T2*T5
      IF(T3.LT.TSIG) T3=T3*T5
      IF(T4.LT.TSIG) T4=T4*T5
      T5=-1.D0
      GO TO 620
  503 IF(T5.LE.0.D0) GO TO 620
      DT1=CS(N,M+1)
      T2=CS(31-N,33-M)
  620 IF(T3.LE.0.D0.OR.ARCNO.EQ.0) CS(N,M+1)=DT1
      IF(T3.GT.TSIG) T3=T3*DABS(DT1)
      F(T4.GT.TSIG) T4=T4*DABS(T2)
      SL31=31-N
```

8.0-325

```
      ISUB2=33-M
      IF(T4.LE.0.0D0.OR.ARCNO.EQ.0) CS(ISUB1,ISUB2)=T2
      IF(ARCNO.GT.0) GO TO 600
      IF(T3.LE.0.D0) GO TO 501
      NOSEST=NOSEST+1
      GPVALU(NOSEST)=DT1/T3
      GPSIG(NOSEST)=T3
      INDXOS(NOSEST,1)=1
      INDXOS(NOSEST,2)=N
      INDXOS(NOSEST,3)=M
      NDXDEG=MAX0(NDXDEG,N)
      NDXORD=MAX0(NDXORD,M)
  501 IF(T4.LE.0.D0) GO TO 200
      NOSEST=NOSEST+1
      GPVALU(NOSEST)=T2/T4
      GPSIG(NOSEST)=T4
      INDXOS(NOSEST,1)=2
      INDXOS(NOSEST,2)=N
      INDXOS(NOSEST,3)=M
      NDXDEG=MAX0(NDXDEG,N)
      NDXORD=MAX0(NDXORD,M)
      GO TO 200
  502 J=M+1
      K=33-M
      I1=J
      IF(M.GT.0) I1=J-1
      DO 505 I=I1,30
      CS(I,J)=0.0D0
  505 CS(31-I,K)=0.0D0
      GO TO 200
  600 IF(T3.LE.0.0D0) GO TO 601
      IF(NGPARC.LE.NBIAS) GO TO 200
      BIASSG(NGPARC)=T3
      BTYPE(NGPARC)=1
      BSTAND(NGPARC)=N*100+M
      BIASD(NGPARC)=DT1/T3
      NDXDEG=MAX0(NDXDEG,N)
      NDXORD=MAX0(NDXORD,M)
      NGPARC=NGPARC-1
  601 IF(T4.LE.0.0D0) GO TO 200
      IF(NGPARC.LE.NBIAS) GO TO 200
      BIASSG(NGPARC)=T4
      BTYPE(NGPARC)=2
      BSTAND(NGPARC)=N*100+M
      BIASD(NGPARC)=T2/T4
      NDXDEG=MAX0(NDXDEG,N)
      NDXORD=MAX0(NDXORD,M)
      NGPARC=NGPARC-1
      GO TO 200
C DATA CARD
  525 JSTAND=IARRAY(1)*1000+IARRAY(2)*100+IARRAY(3)*10+IARRAY(4)
      IF(NARCNO.EQ.0) GO TO 590
C SET DATA TAPE UNITS
      IOBS=JSTAND/100
      IARRAY(1)=JSTAND-IOBS*100
      IARRAY(2)=DT1+0.500
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
      IARRAY(3)=T2+0.5D0
      IARRAY(4)=T3+0.5D0
      JOBS=20
      IF(MOD(IGIDS,4).EQ.0) JOBS=21
      DO 530 I=1,4
      IF(IARRAY(I)) 530,530,700
  530 IARRAY(I)=0
      IARRAY(1)=JOBS
      GO TO 700
C STAPOS CARD
550   NSTARD=0
      PRD(1)=DT1
      PRD(2)=T2
C READ STATION POSITIONS
  560 READ(INTP,10005,END=900) CARDID,JSTANO,DT1,NAME7
      IF(CARDID.EQ.CDNAME(23)) GO TO 590
      NSTARD=NSTARD+1
      ISTARD(NSTARD)=JSTANO
      JNAME(NSTARD)=CARDID
      GO TO 560
C SET COMMON PARAMETER INDICATORS
  590 MINOUT=JSTANO/100
      OUTCON=FLOAT(MINOUT)*1.0D-2
      NSTA=NSTEST
      IF(OUTCON.LE.0.) OUTCON=0.02
      MAXOUT=JSTANO-100*MINOUT
      MINOUT=MAXOUT/10
      MAXOUT=MAXOUT-MINOUT*10
      MAXOUT=MAX0(MAXOUT,1)
      MINOUT=MAX0(MINOUT,1)
      LITRES=DT1.GT.0.0D0
C COMPUTE FLATTENING
      FLAT=1.0D0/FLAT
      FSQ32=1.5D0*AE*FLAT**2
      FFSQ32=AE*FLAT+FSQ32
      AESQ=AE**2
C WRITE RUN TITLE
  620 WRITE(OUTP,44447) NARCS
      IF(NSTEST.GT.0) WRITE(OUTP,44448) NSTEST
      WRITE(OUTP,44449) (DASH,I=1,15),TITLE,MAXOUT,OUTCON,MINOUT
      NDXCM1=NDXOLG
      NDXCM2=NDXORD
      NDCONS=MAX0(1,NDCONS)
      NDCONS=MIN0(NDCONS,INDEXI-1)
      IF(NDCNST.LE.0) NDCONS=0
      NCONST=NDCONS*(NDCONS+2)
C PRINT GEOPOTENTIAL COEFFICIENTS & EARTH MODEL
      CALL COEFL(OUTP,NRMCFL)
      PPNF=1.D0/FLAT
      WRITE(OUTP,10153) (DASH,I=1,11),AE,PPNF,GM
      IF (NSD.LT.0) GO TO 640
      NADJ=NDCNST-NCONST
      IF(NADJ.LT.0.AND.NCONST.GT.0) CALL ERROR(16,CARDID)
C PRINT SURFACE DENSITIES
      CALL PDEN1
C WRITE SURFACE DENSITY INFORMATION ON SCRATCH FILE
```

```
      DO 630 I1=1,4050,450
      I2=MIN0(I1+449,4050)
  630 WRITE(SCRC) (ISURF(I),I=I1,I2)
      DO 635 I1=1,2026,450
      I2=MIN0(I1+449,2026)
  635 WRITE(SCRC) (SD(I),I=I1,I2)
  640 CONTINUE
      IF(NCSEST.LE.0) RETURN
C WRITE ADJUSTED GEOPOTENTIAL A PRIORI VALUES ON SCRATCH FILE
      DO 650 I1=1,NCSEST,225
      I2=MIN0(NCSEST,I1+224)
  650 WRITE(SCRC) (GPVALO(I),I=I1,I2)
      RETURN
C DETERMINE NUMBER OF ARC FORCE MODEL EQUATIONS EXCEPT ADJUSTED
C GEOPOTENTIAL COEFFICIENTS
  700 NEON(1)=0
      NEON(2)=0
      IF(ORBTSW) GO TO 702
      INPARI=6*NSAT
      J=INPARI
      DO 704 I=1,NSAT
      IF(ASAT(I).LE.0.0D0.OR.MSAT(I).LE.0.0D0) GO TO 704
      IF(DRAGSG(I).LE.0.0D0) GO TO 701
      NEON(I)=NEON(I)+1
      INPARI=INPARI+1
      ADDR(I)=INPARI-J
  701 IF(DRGLSG(I).LE.0.0D0) GO TO 703
      NEON(I)=NEON(I)+1
      INPARI=INPARI+1
      ADDRD(I)=INPARI-J
  703 IF(EMISSG(I).LE.0.0D0) GO TO 704
      NEON(I)=NEON(I)+1
      INPARI=INPARI+1
      SRAD(I)=INPARI-J
  704 CONTINUE
C PRINT ARC DESCRIPTION
  702 WRITE(OUTP,10302) ARCNO,(DASH,I=1,22)
      IF(ISATID(1).GT.0) WRITE(OUTP,20080) (I,ISATID(I),I=1,NSAT)
      WRITE(OUTP,10303) TITLE
      WRITE(OUTP,10150) (DASH,I=1,8)
      IF(ORBTSW) GO TO 705
      WRITE(OUTP,10151) INNMAX,CONVRG,INNMIN
      WRITE(OUTP,10102)
      WRITE(OUTP,10102)
      GO TO 710
  705 WRITE(OUTP,10152)
C PRINT ARC FORCE MODEL DESCRIPTION
  710 J=INDEX1-1
      K=INDEX2-1
      WRITE(OUTP,10149) (DASH,I=1,11),J,K
      IF(NDXDLG.GT.J) CALL ERROR(12,CARDID)
      IF(NDXDRG.GT.K) CALL ERROR(13,CARDID)
      DO 725 I=1,6
      IF(I.GT.NBODY) GO TO 715
      IF(MBODY(I).GT.0.0D0) GO TO 720
  715 WRITE(OUTP,10154) I,BODIES(I)
```

```
      GO TO 725
  720 WRITE(OUTP,10155) I,(BODIES(I),J=1,2),MBODY(I)
  725 CONTINUE
      IF((LOVE(1).EQ.0.0D0.AND.LOVE(2).EQ.0.0D0).OR.(MBODY(1).EQ.0.0D0
     *    .AND.MBODY(2).EQ.0.0D0)) GO TO 850
      IF(MBODY(1).EQ.0.0D0) WRITE(OUTP,11002) (TIDES(I),I=3,7)
      IF(MBODY(2).EQ.0.0D0) WRITE(OUTP,11002) TIDES(1),(TIDES(I),I=4,7)
      IF(MBODY(1).NE.0.0D0.AND.MBODY(2).NE.0.0D0) WRITE(OUTP,11002)TIDES
      WRITE(OUTP,11003) LOVE
      GO TO 860
  850 WRITE(OUTP,11001)
  860 DO 745 I=1,NSAT
      NOPRNT=ASAT(I).GT.0.0D0.AND.MSAT(I).GT.0.0D0
      WRITE(OUTP,10156) I
      IF(CD(I).GT.0.0D0.AND.NOPRNT) GO TO 730
      WRITE(OUTP,10158)
      GO TO 735
  730 IF(ADDR(I).EQ.0) WRITE(OUTP,10159) CD(I)
      IF(ADDR(I).GT.0) WRITE(OUTP,10504) CD(I),DRAGSG(I)
      IF(ADDRD(I).EQ.0.AND.CDD(I).GT.0.0D0) WRITE(OUTP,10157) CDD(I)
      IF(ADDRD(I).GT.0) WRITE (OUTP,10502) CDD(I),DRGDSG(I)
  735 IF(EMISS(I).GT.0.0D0.AND.NOPRNT) GO TO 740
      WRITE(OUTP,10160)
      GO TO 745
  740 IF(SRAD(I).EQ.0) WRITE(OUTP,10161) PRESS,EMISS(I)
      IF(SRAD(I).GT.0) WRITE(OUTP,10503) PRESS,EMISS(I),EMISSG(I)
  745 IF(NOPRNT) WRITE(OUTP,10514) ASAT(I),MSAT(I)
      WRITE(OUTP,10162)
      WRITE(OUTP,10118) (DASH,I=1,21)
      DO 750 I=1,NSAT
      WRITE(OUTP,10117) I,(ORDER(I,J),(EONINT(K,J),K=1,3),J=1,2)
      IF(.NOT.VARSTP(I)) WRITE(OUTP,10119) (STEPSZ(I,J),(EONINT(K,J),
     *    K=1,3),J=1,2)
      IF(VARSTP(I)) WRITE (OUTP,10121)
      IF (VARSTP(I)) WRITE(OUTP,10120) ((EONINT(K,1),K=1,3),STEPSZ(I,1),
     *    STEPUP(I),STPLOW(I),DBLERB(I),HLVERB(I),(EONINT(K,2),K=1,3),
     *    STEPSZ(I,2)
      RTOL(I)=RTOL(I)**2
      CTOL(I)=CTOL(I)**2
      HLVERB(I)=HLVERB(I)**2
  750 DBLERB(I)=DBLERB(I)**2
C PRINT OUTPUT REQUESTS
      JJ=INSUPP
      IF(DORB1.LT.0.0D0.AND.JJ.LE.0.AND..NOT.(PLTLSW.OR.XYZFSW.OR.XYZLSW
     *    .OR.GROFSW)) GO TO 765
      WRITE(OUTP,10162) (DASH,I=1,23)
      IF(PLTLSW) WRITE(OUTP,10164)(ITNMS(I),I=2,4)
  755 IF(DORB1.LT.0.0D0) GO TO 756
      CALL DATES(DORB1,IYMD,IHM,SEC)
      CALL DATES(DORB1E,IYMDE,IHME,SECE)
      WRITE(OUTP,20010) IYMD,IHM,SEC,IYMDE,IHME,SECE
      DT1=ORBRT*8.64D4
      WRITE(OUTP,20050) DT1
      IF(.NOT.TCHECK) GO TO 756
      WRITE(OUTP,20062)
      WRITE(OUTP,20060)
```

```
 756  IF(.NOT.(XYZFSW.OR.XYZLSW)) GO TO 7057
      IF(TORDFT) WRITE(OUTP,20064)
      IF(TORDFT) WRITE(OUTP,20060)
      IF(XYZLSW.AND.XYZFSW) GO TO 757
      IF(XYZFSW) WRITE(OUTP,10165) ITNMS(1),(ITNMS(I),I=3,4)
      IF(XYZLSW) WRITE(OUTP,10165) (ITNMS(I),I=2,4)
      GO TO 7056
  757 WRITE(OUTP,10165) ITNMS(1),(ITNMS(I),I=3,5),(ITNMS(I),I=2,4)
 7056 IF(ORBTSW) GO TO 7057
      IF(RATE.GT.0.) WRITE(OUTP,20025) RATE
      WRITE(OUTP,10167)
 7057 IF(.NOT.ORBTSW) GO TO 758
      CALL DATES(DORBIT,IYMD,IHM,SEC)
      CALL DATES(DAYEND,IYMDE,IHME,SECE)
      WRITE(OUTP,20020) IYMD,IHM,SEC,IYMDE,IHME,SECE
      WRITE(OUTP,10166) RATE
      IF(RVTP.GT.0) WRITE(OUTP,20030)
  758 IF(GPJFSW) WRITE(OUTP,10182) ITNMS(1),(ITNMS(I),I=3,4)
  760 IF(ORBTSW.OR.JJ.LE.0) GO TO 765
      WRITE(OUTP,20040)
      IF(MOD(JJ,2).EQ.1) WRITE(OUTP,20041)
      IF(JJ.EQ.3) WRITE(OUTP,20042)
      IF(MOD(JJ,4).GT.1) WRITE(OUTP,20043)
      IF(JJ.EQ.4) WRITE(OUTP,20044)
C PRINT NOMINAL ORBIT
  765 DO 769 I=1,NSAT
      WRITE(OUTP,10168) I,(DASH,J=1,28)
      IF(IELM.GT.0) WRITE(OUTP,20110) IELM
      WRITE(OUTP,20100) IYREF,IEPYMD,IEPHM,EPSEC,(ELEMIN(J,I),J=1,6),
     . (ORBEL(J,I),J=1,6)
  769 IF(.NOT.ORBTSW) WRITE(OUTP,10173) ((VARCOV(J,K,1),K=1,6),J=1,6)
      IF(TORDFO) ORBRT=ORBRT+1.000
      DO 767 K=1,NSAT
      I1=1
      DO 766 I=1,6
      DO 766 J=I,6
      SUM1(I1)=VARCOV(J,I,K)
  766 I1=I1+1
      CALL SYMINV(SUM1,6,6,VARCOV(1,1,K))
      I1=1
      DO 767 I=1,6
      DO 767 J=I,6
      VARCOV(J,I,K)=SUM1(I1)
      VARCOV(I,J,K)=SUM1(I1)
  767 I1=I1+1
C PRINT SUMMARY OF PREPROCESSING INSTRUCTIONS
      IF(NOPRPR.NE.0) CALL PRNTPR(OUTP,ITYPE)
C PRINT NUMBERS OF MEASUREMENTS WHICH ARE TO BE CULLED
      IF(NCULL.EQ.0) GO TO 770
      DO 768 J=1,NCULL
  768 IF(CULL(2,J).EQ.0) CULL(2,J)=CULL(1,J)
      WRITE(OUTP,20070)((CULL(I,J),I=1,2),J=1,NCULL)
C PRINT SIGMAS TO BE USED FOR MEASUREMENTS
  770 WRITE(OUTP,10600)
      WRITE(OUTP,10610) ATYPE(2),SGPRNT(2)
      WRITE(OUTP,10620) UNITS(3)
```

```
      WRITE(OUTP,10610) ATYPE(3),SGPRNT(3)
      WRITE(OUTP,10630)
      WRITE(OUTP,10670) ATYPE(4),SGPRNT(4),UNITS(3),ATYPE(11),SGPRNT(11)
      WRITE(OUTP,10630)
      DO 780 I=1,7
      IF(I.GT.1.AND.I.LT.5) GO TO 780
      WRITE(OUTP,10610) ATYPE(I),SGPRNT(I)
      IF(I.NE.5) WRITE(OUTP,10620) UNITS(1)
      IF(I.EQ.5) WRITE(OUTP,10625)
      WRITE(OUTP,10610) ATYPE(I+7),SGPRNT(I+7)
      IF(I.EQ.5) WRITE(OUTP,10625)
      IF(I.NE.5) WRITE(OUTP,10620) UNITS(1)
  780 CONTINUE
      IF(IGEOS.GT.2) WRITE(OUTP,10640) (ATYPE(I),SGPRNT(I),I=15,26)
      K=(IGEOS+1)/2-IGEOS/4
      IF(K.EQ.1) WRITE(OUTP,10680) (ATYPE(I),SGPRNT(I),I=27,30)
      IF(NSIG.EQ.0) GO TO 784
      DO 783 I=1,NSIG
      IF(ISTNO(I).EQ.0) GO TO 783
      I1=IMTYPE(I)
      IF(I1.LE.0) I1=31
      MTYPE=IMTYPE(I)-(IMTYPE(I)/8)*7
      WRITE(OUTP,10610) ATYPE(I1),SIGCHG(I)
      WRITE(OUTP,10650) ISTNO(I)
      IF(MTYPE.EQ.5) WRITE(OUTP,10625)
      IF(MTYPE.EQ.2) WRITE(OUTP,10620) UNITS(3)
      IF(MTYPE.EQ.3.OR.MTYPE.GT.28) WRITE(OUTP,10620)
      IF(MTYPE.EQ.27) WRITE(OUTP,10660)
      IF(MTYPE.EQ.28) WRITE(OUTP,10690)
      IF(MTYPE.EQ.1.OR.MTYPE.GT.5) WRITE(OUTP,10620) UNITS(1)
      IF(IMTYPE(I).GT.26) MTYPE=I1
  783 CONTINUE
  784 MTYPE=IGEOS
C WRITE ELECTRONIC BIAS INFORMATION ON SCRATCH FILE
      WRITE(SCRC) BESTNO
      WRITE(SCRC) BETYPE
      IF(.NOT.SLOLEN) GO TO 895
      DO 890 I=1,4
      ITAPE=KTAPES(I)
      IF(ITAPE.EQ.5) GO TO 890
      IF(ITAPE.EQ.IARRAY(I)) GO TO 890
      IF(ITAPE.NE.0) REWIND ITAPE
  890 CONTINUE
  895 IF(ORBTSY) RETURN
      IF(NBIASE.LE.0) GO TO 880
C PRINT ELECTRONIC BIASES TO BE EXTRACTED
      WRITE(OUTP,44700)
      WRITE(OUTP,44710) (BESTNO(I),BETYPE(I),I=1,NBIASE)
  880 I1=IGEOS+1
C CALL SUBROUTINE TO READ DATA
      GO TO (785,786,786,787,788),I1
      WRITE(OUTP,44620) IGEOS
      CALL ERROR(7,CARDID)
      RETURN
  785 CALL DOCSRC(NSTARD)
      GO TO 790
```

```
  786 CALL GEOSRD(NSTARD)
      GO TO 790
  787 CALL PCBRD
      GO TO 790
  788 CALL SIMRD(NSTARD)
  790 IF(SEQUEN) GO TO 794
      DO 792 I=1,4
      ITAPE=IARRAY(I)
      IF(ITAPE.GT.0) REWIND ITAPE
  792 CONTINUE
      GO TO 798
  794 DO 796 I=1,4
  796 KTAPES(I)=IARRAY(I)
  798 IF(NBIAS.EQ.0) RETURN
C PRINT ADJUSTED BIAS INFORMATION
      LINES=0
      WRITE(OUTP,10410)
      DO 801 I=1,NBIAS
      I1=BSTAND(I)
      BSTAND(I)=NUMBR2(I1,ISTAND,NSTA)
      I1=BSTAND(I)
      IF(I1.EQ.0) GO TO 801
      T2=TIMING
      I2=BTYPE(I)
      IF(I2.GT.0) T2=ATYPE(I2)
      I2=I2+1
      IF(I1.LT.15) UNIT=UNITS(I2)
      IF(I1.EQ.27) UNIT=UNITS(1)
      IF(I1.EQ.28) UNIT=HERTZ
      IF(I1.GT.28) UNIT=UNITS(4)
      IF(MOD(LINES,5).EQ.0) WRITE(OUTP,10102)
      LINES=LINES+1
      WRITE(OUTP,10411) ISTAND(I1),T2,BIASO(I),UNIT,BIASSG(I)
      IF(BSEND(I).LE.BSTRT(I)) WRITE(OUTP,10413)
      IF(BSEND(I).LE.BSTRT(I)) GO TO 801
      CALL DATES(BSTRT(I),IYMD1,IHM1,SEC)
      CALL DATES(BSEND(I),IYMD2,IHM2,SEC)
      WRITE(OUTP,10412) IYMD1,IHM1,IYMD2,IHM2
  801 CONTINUE
      RETURN
C IF END OF FILE ENCOUNTERED WITHOUT A PRECEDING DATA CARD, TERMINATE
  900 CALL ERROR(6,CARDID)
      STOP
C FORMAT STATEMENTS
10003 FORMAT(6D12.6)
10182 FORMAT(/1H0,5X52HSATELLITE X,Y,Z AND GROUND-TRACK PLOT REQUESTED
     .FOR,2A6,A3,A4,2A6,A3)
10004 FORMAT(A6,4I1,4D15.3,D10.8)
10410 FORMAT(1H1,39X,45HNOMINAL ESTIMATES OF ADJUSTED BIAS PARAMETERS/
     .   1H0,18X,7HSTATION,5X,9HPARAMETER,12X,15HA PRIORI VALUES,12X,
     .   28HCOVERAGE ------- YYMMDD HHMM/19X,6HNUMBER,7X,4HTYPE,
     3   11X,4HBIAS,14X,5HSIGMA,11X,5HBEGIN,13X,3HEND)
10411 FORMAT(19X,I6,5X,A6,5H BIAS,2X,1PD11.3,2X,A6,D11.3)
10412 FORMAT(1H+,78X,I6,I5,6X,I6,I5)
10413 FORMAT(1H+,78X,24HDIFFERENT BIAS FOR EACH PASS)
10102 FORMAT(1X)
```

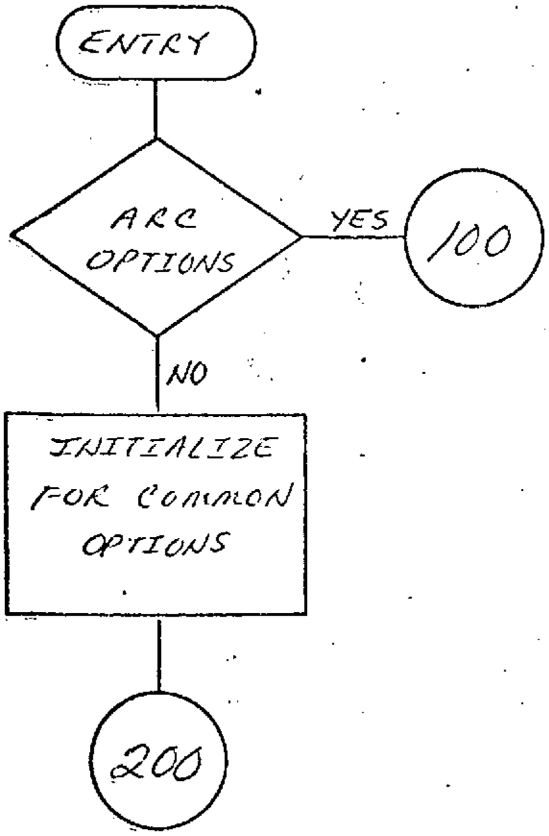
```
10117 FORMAT(1H0,5X,'COWELL PREDICTOR-CORRECTOR FOR SAT. NO.',I2,          INOU 952
     .   2(/10X,I2,' TH ORDER FOR ',2A8,A5))                               INOU 953
10118 FORMAT(1H1,5X,21HNUMERICAL INTEGRATION/6X,21A1)                      INOU 954
10119 FORMAT(9X,F6.1,' SECOND FIXED STEP FOR ',2A8,A5)                     INOU 955
10120 FORMAT(10X,'FOR ',2A8,A5/15X,0PF6.1,' INITIAL STEP SIZE'/            INOU 956
     .    15X,F6.1,' MAXIMUM STEP SIZE'/15X,F6.1,' MINIMUM STEP SIZE'/     INOU 957
     .    15X,'STEP SIZE INCREASED FOR PREDICTOR-CORRECTOR DIFFERENCE ',  INOU 958
     .    'LESS THAN ',1PD7.1,' METERS'/15X,'STEP SIZE DECREASED FOR ',   INOU 959
     .    'PREDICTOR-CORRECTOR DIFFERENCE GREATER THAN ',1PD7.1,'METERS')INOU 960
10121 FORMAT(8X,'VARIABLE STEP INTEGRATION USED')                          INOU 961
10150 FORMAT(1H0,5X,8HRUN TYPE/6X,8A1)                                     INOU 962
10151 FORMAT(1H0,5X,'DATA REDUCTION -',I3,' ITERATIONS OR CONVERGENCE ',INOU 963
     1          'WITHIN',2PF4.1,' PER CENT'/1H ,7X,'WITH A MINIMUM OF',I2, INOU 964
     2          ' ITERATIONS'/)                                            INOU 965
10152 FORMAT(1H0 5X19HORBIT GENERATOR RUN // )                             INOU 966
10149 FORMAT          (1H0,5X11HFORCE MODEL / 6X11A1 // 6X                 INOU 967
     .        28HA. GEOPOTENTIAL COEFFICIENTS /                            INOU 968
     . 10X,'LISTED COEFFICIENTS THROUGH DEGREE',I3,' AND ORDER',I3//6X,   INOU 969
     .          22HB. OTHER PERTURBATIONS )                                INOU 970
10153 FORMAT(1H05X11HEARTH MODEL/ 6X11A1// 6X15HSEMI MAJOR AXIS 4X         INOU 971
     .          10HFLATTENING 4X 22HGRAVITATIONAL CONSTANT / 9X8H(METERS)  INOU 972
     .    22X,22H(METERS**3/SECONDS**2) //8X,F11.2,7X,3H1./,F7.3,7X,      INOU 973
     .          1PD14.8 //)                                                INOU 974
10154 FORMAT(1H ,6X,I1,2H. ,A7,' GRAVITATION NOT APPLIED')                 INOU 975
10155 FORMAT(1H ,6X,I1,2H. ,A7,' GRAVITATION APPLIED - RATIO OF ',A7,      INOU 976
     .    ' MASS TO MASS OF EARTH =',1PD13.6)                              INOU 977
10156 FORMAT(1H0,6X,'SAT. NO.',I2,' PERTURBATIONS')                        INOU 978
10158 FORMAT(10X,'8. DRAG NOT APPLIED')                                    INOU 979
10157 FORMAT(15X,'- DRAG COEFFICIENT RATE =',F7.4)                         INOU 980
10159 FORMAT(10X,'8. DRAG APPLIED'/15X,' DRAG COEFFICIENT =',F7.4)         INOU 981
10160 FORMAT(10X,'9. SOLAR RADIATION PRESSURE NOT APPLIED')                INOU 982
10161 FORMAT(10X,'9. SOLAR RADIATION PRESSURE APPLIED'/15X,'- SOLAR ',     INOU 983
     .    'RADIATION PRESSURE (NEWTONS/METER**2) =',1PD10.3/15X,           INOU 984
     . 16H- REFLECTIVITY =,0PF7.3)                                         INOU 985
10502 FORMAT(15X,'- A PRIORI DRAG COEFFICIENT RATE =',F8.4/15X,            INOU 986
     .    '- STANDARD DEVIATION OF A PRIORI DRAG COEFFICIENT RATE =',      INOU 987
     .    F8.4)                                                            INOU 988
10503 FORMAT(1H ,9X,44H9. ADJUSTED SOLAR RADIATION PRESSURE APPLIED/15X,INOU 989
     .    47H- SOLAR RADIATION PRESSURE (NEWTONS/METER**2) =,1PD10.3/15X, INOU 990
     .    25H- A PRIORI REFLECTIVITY =,0PF7.3/15X,                         INOU 991
     .    44H- A PRIORI REFLECTIVITY STANDARD DEVIATION =,F7.3)            INOU 992
10504 FORMAT(10X,46H8. DRAG APPLIED WITH DRAG COEFFICIENT ADJUSTED/        INOU 993
     .    15X,29H- A PRIORI DRAG COEFFICIENT =,F8.3/                       INOU 994
     .    15X,51H- STANDARD DEVIATION OF A PRIORI DRAG COEFFICIENT =,F8.3)INOU 995
10514 FORMAT(15X,46H- SATELLITE CROSS SECTIONAL AREA (METERS**2) =,        INOU 996
     .     1PD10.3/15X,30H- SATELLITE MASS (KILOGRAMS) =,D10.3)            INOU 997
10162 FORMAT(1H0/,6X,'SPECIAL OUTPUT REQUESTS'/1H ,5X,23A1/)               INOU 998
10164 FORMAT(1H0,5X,35HBINARY RESIDUAL TAPE REQUESTED FOR ,2A6,A3,A4,      INOU 999
     1          2A6,A3)                                                    INOU1000
10165 FORMAT(1H0 5X47HSATELLITE X,Y,Z AND GROUND TRACK REQUESTED FOR       INOU1001
     .          2A6,A3,A4,2A6,A3 )                                         INOU1002
10166 FORMAT(6X,5HEVERY,F9.3,8H SECONDS/1H0)                               INOU1003
10167 FORMAT(1H 7X14HAT DATA POINTS // )                                   INOU1004
10173 FORMAT(1H0,5X,26HVARIANCE/COVARIANCE MATRIX //                       INOU1005
     .          18X1HX 14X1HY 14X1HZ 13X4HXDOT 11X4HYDOT 11X4HZDOT //      INOU1006
     .     7X,1HX,2X,1PD15.7/7X,1HY,2X,2D15.7/7X,1HZ,2X,6E15.7/           INOU1007
```

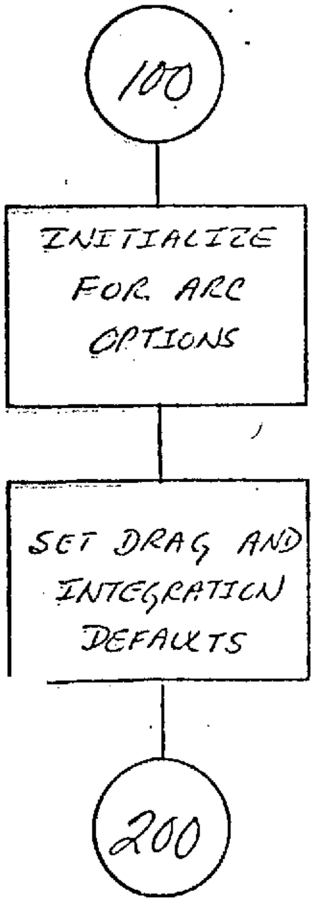
```
     .    6X,4HXDOT,6E15.7/6X,4HYDOT,6E15.7/6X,4HZDOT,6E15.7)            INOU1008
 10168 FORMAT(1H1,5X,'NOMINAL ORBIT FOR SAT. NO.',I2/6X,28A1)            INOU1009
 10005 FORMAT(A6,I4,D15.3,30X,A1)                                        INOU1010
 10302 FORMAT(1H1,10X,3HARC,I3,' RUN DESCRIPTION'/1H ,10X,22A1)          INOU1011
 10303 FORMAT(1X/3(11X,10A8/))                                           INOU1012
 11001 FORMAT(7X,3H7. ,'EARTH TIDES NOT APPLIED')                        INOU1013
 11002 FORMAT(7X,3H7. ,'EARTH TIDES APPLIED - ',7A5)                     INOU1014
 11003 FORMAT(10X,'- K2 =',E9.3/10X,'- K3 =',E9.3/10X,'- PHASE ANGLE =', INOU1015
     .    F7.2,' DEGREES')                                               INOU1016
 10600 FORMAT(1H1,47X,31HMEASUREMENT STANDARD DEVIATIONS/1H0,50X,        INOU1017
     .      7HSTATION,13X,5HSIGMA/1H ,47X,6HNUMBER,3X,4HTYPE,6X,         INOU1018
     .       5HVALUE,2X,5HUNITS/)                                        INOU1019
 10610 FORMAT(1H ,55X,A6,F10.1)                                          INOU1020
 10620 FORMAT(1H+,73X,A6)                                                INOU1021
 10625 FORMAT(1H+,73X,4HMILS)                                            INOU1022
 10630 FORMAT(1H+,73X,6HCM/SEC)                                          INOU1023
 10640 FORMAT(3(49X,4HNONE,3X,A6,F10.1,2X,6HMETERS/),3(49X,4HNONE,3X,    INOU1024
     .    A6,F10.1,2X,6HCM/SEC/),49X,4HNONE,3X,A6,F10.1,2X,6HMETERS/     INOU1025
     .    49X,4HNONE,3X,A6,F10.1,2X,4HE-05/4(49X,4HNONE,3X,A6,F10.1,     INOU1026
     .    2X,7HSECONDS/))                                                INOU1027
 10650 FORMAT(1H+,47X,I5)                                                INOU1028
 10660 FORMAT(1H+,73X,7HNANOSEC)                                         INOU1029
 10670 FORMAT(48X,6HALTYPE,2X,A6,F10.1,2X,A5)                            INOU1030
 10680 FORMAT(49X,3HALL,4X,A6,F10.1,2X,7HNANOSEC/                        INOU1031
     .          49X,3HALL,4X,A6,F10.1,2X,10HMICROHERTZ/                  INOU1032
     .          49X,3HALL,4X,A6,F10.1,2X,6HCM/SEC,2X,7HTWO-WAY/          INOU1033
     .          49X,3HALL,4X,A6,F10.1,2X,6HCM/SEC,2X,9HTHREE-WAY)        INOU1034
 10690 FORMAT(1H+,73X,10HMICROHERTZ)                                     INOU1035
 20010 FORMAT(1H0,5X,25HORB1 TAPE REQUESTED FROM ,2I6,F7.4,4H TO ,2I6,   INOU1036
     1          F7.4)                                                    INOU1037
 20020 FORMAT(1H0,5X,48HSATELLITE X,Y,Z AND GROUND TRACK REQUESTED FROM ,INOU1038
     .          2I6,F7.4,4H TO ,2I6,F7.4)                                INOU1039
 20025 FORMAT(1H ,7X,5HEVERY,F10.2,12H SECONDS AND)                      INOU1040
 20030 FORMAT(1H0,5X,37HRV TAPE REQUESTED FOR FIRST ITERATION)           INOU1041
 20040 FORMAT(1H0,5X,'RESIDUAL PRINTING HAS BEEN REQUESTED FOR')         INOU1042
 20041 FORMAT(1H+,46X,'THE FIRST INNER ITERATION ON THE FIRST OUTER ',   INOU1043
     .    'ITERATION'/)                                                  INOU1044
 20042 FORMAT(1H+,42X,'AND')                                             INOU1045
 20043 FORMAT(1H+,46X,'THE LAST INNER ITERATION ON THE LAST OUTER ',     INOU1046
     .    'ITERATION'/)                                                  INOU1047
 20044 FORMAT(1H+,46X,'ALL ITERATIONS'/)                                 INOU1048
 20050 FORMAT(1H ,8X,22HOUTPUT REQUESTED EVERY,F9.3,8H SECONDS)          INOU1049
 20060 FORMAT(1H+,15X,42HREFERENCED TO TRUE EQUATOR AND EQUINOX OF ,     INOU1050
     1          14HREFERENCE TIME)                                       INOU1051
 20062 FORMAT(1H0,5X,9HORB1 TAPE)                                        INOU1052
 20064 FORMAT(1H0,5X,9HEPHEMERIS)                                        INOU1053
 20070 FORMAT(1H1,37X,39HINDIVIDUAL MEASUREMENTS MANUALLY CULLED,        INOU1054
     1         1X,17HFROM THE SOLUTION/1H0,33X,4(13HMEASUREMENT ,6X)/    INOU1055
     2         1H ,35X,4(7HNUMBERS,11X)/1H /25(1H ,32X,4(I5,3H TO,I5,5X)/))INOU1056
 20080 FORMAT(1X/(11X,'SATELLITE ID FOR SAT. NO.',I2,' IS',I8))          INOU1057
 20100 FORMAT(1H0,5X,                                                    INOU1058
     .          'REFERENCE TIME    -  YEAR,MONTH,DAY ',I6/               INOU1059
     .          1H0,5X,20HEPOCH OF ELEMENTS - ,                          INOU1060
     .          1X15HYEAR,MONTH,DAY I6,AX19HHOUR,MINUTE,SECOND I4,F7.3//6X INOU1061
     .          32HINERTIAL RECTANGULAR COORDINATES/1H0,                 INOU1062
     .          16X1HX26X1HY26X1HZ/16X,3H(M),24X,3H(M),24X,3H(M)//       INOU1063
```

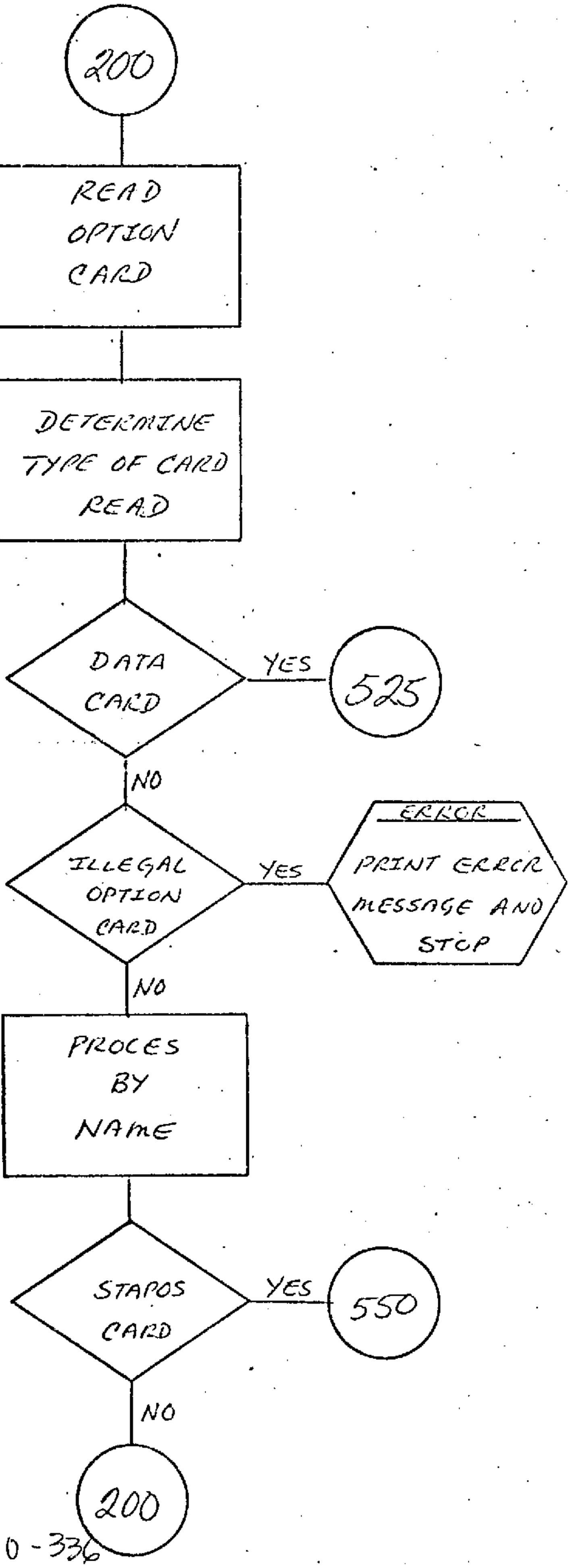
REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
     .        2X,7P3D28.10//15X4HXDOT,23X4HYDOT,23X4HZDOT/                     INOU1064
     .        15X5H(M/S),22X5H(M/S),22X5H(M/S)//2X,4P3D28.13                    INOU1065
     .        /1H05X18HKEPLERIAN ELEMENTS/1H0,11X,1HA,12X,1HE,12X,1HI,7X,INOU1066
     .        36HRA ASC NODE  ARG PERIGEE  MEAN ANOMALY/9X,8H(METERS),          INOU1067
     .        13X,4(4X,9H(DEGREES))/1H0,0PF16.1,F13.9,4F13.6)                   INOU1068
20110 FORMAT(1H0,5X,'CODS DATA BASE ELEMENT SET',I5,' USED')                    INOU1069
44447 FORMAT(1H1,20(/),10X,'MULTI-ARC GEODYN RUN USING DATA FROM',             INOU1070
    1         I3,' SATELLITE ARCS')                                             INOU1071
44448 FORMAT(1H ,12X,'WITH THE ADJUSTMENT OF',I3,' STATION POSITIONS')          INOU1072
44449 FORMAT(1H0/10X,'RUN DESCRIPTION'/10X,15A8//3(10X,10A8/)/10X,I2,           INOU1073
    1         ' OUTER ITERATIONS OR CONVERGENCE WITHIN',2PF4.1,' PER CENT'INOU1074
    2         /12X,'WITH A MINIMUM OF',I2,' OUTER ITERATIONS')                  INOU1075
44600 FORMAT(1H1,20X,'ILLEGAL OPTION CARD INPUT'/1H0,15X,'EXPLANATION : INOU1076
     .THE OPTION CARD ',A6,' IS ILLEGAL IN CONTROL SET',I2,1H./1H0,15X,        INOU1077
     . 'PROGRAM ACTION : CARD IGNORED -- EXECUTION CONTINUING.'/)               INOU1078
44610 FORMAT(1H1,20X,'ILLEGAL OPTION CARD INPUT'/1H0,15X,'EXPLANATION : INOU1079
     .THE OPTION CARD ',A6,' IN CONTROL SET 2 ARC',I3,' REQUESTS THE '/ INOU1080
     .   21X,'ADJUSTMENT OF A COMMON ARC PARAMETER.'/1H0,15X,'PROGRAM ',INOU1081
     .   'ACTION : THE REQUESTED COMMON PARAMETER ADJUSTMENT WILL NOT BEINOU1082
     . PERFORMED --'/21X,'EXECUTION CONTINUING.'/)                             INOU1083
44620 FORMAT(1H1,20X,'EXECUTION TERMINATED DUE TO IMPROPER SETUP.'//           INOU1084
     .   16X,'EXPLANATION: DATA TYPE INDICATOR ON EPOCH CARD =',I2,             INOU1085
     .   ' GREATER THAN 4')                                                     INOU1086
44700 FORMAT(1H1,38X,'MEASUREMENTS FOR WHICH ELECTRONIC BIASES WILL ',          INOU1087
     .   'BE REMOVED'/1H0,7(13HSTATION  MEAS,4X),13HSTATION  MEAS/              INOU1088
     .   7(3X,3HNO.,4X,4HTYPE,3X),3X,3HNO.,4X,4HTYPE/)                          INOU1089
44710 FORMAT(I6,I7,I10,I7,I10,I7,I10,I7,I10,I7,I10,I7,I10,I7,I10,I7)            INOU1090
      END                                                                       INOU1091
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

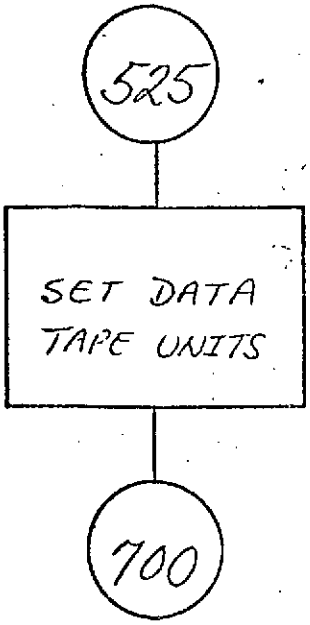


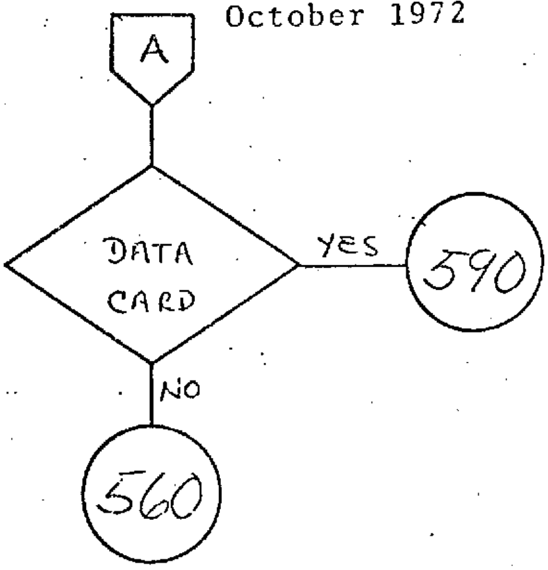


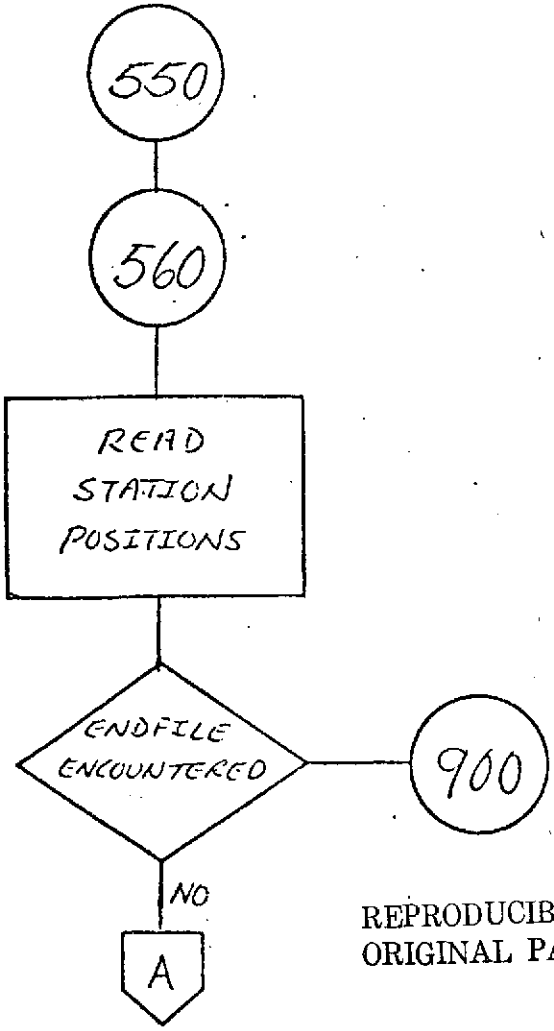


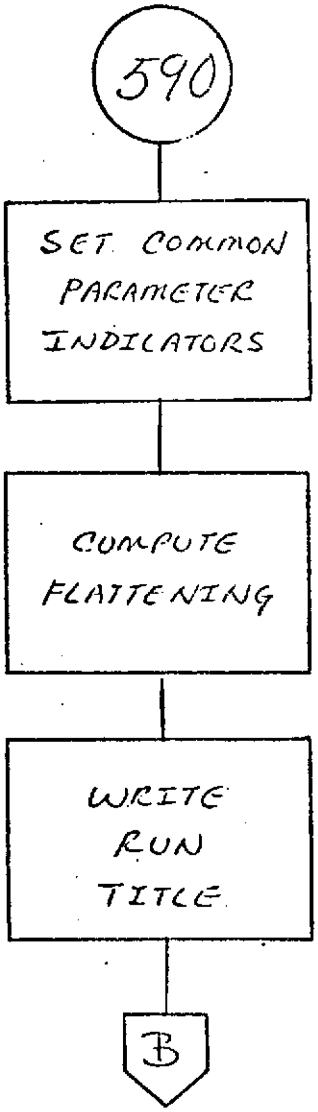
REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

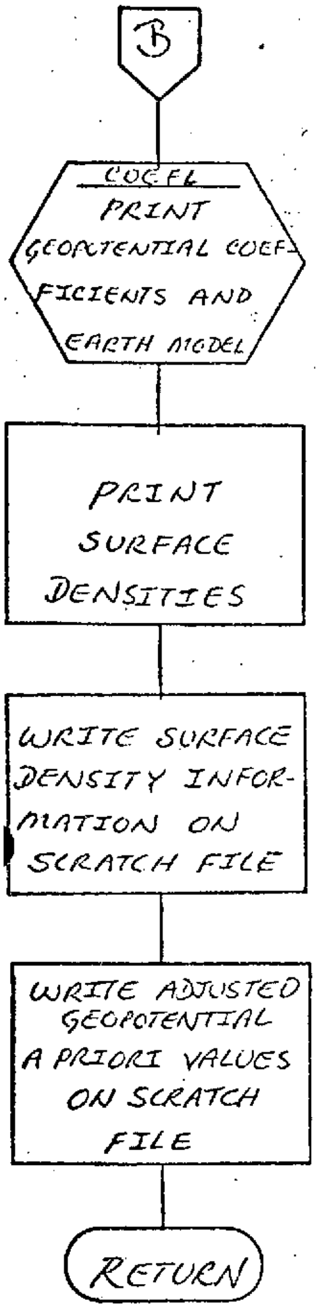
80-336

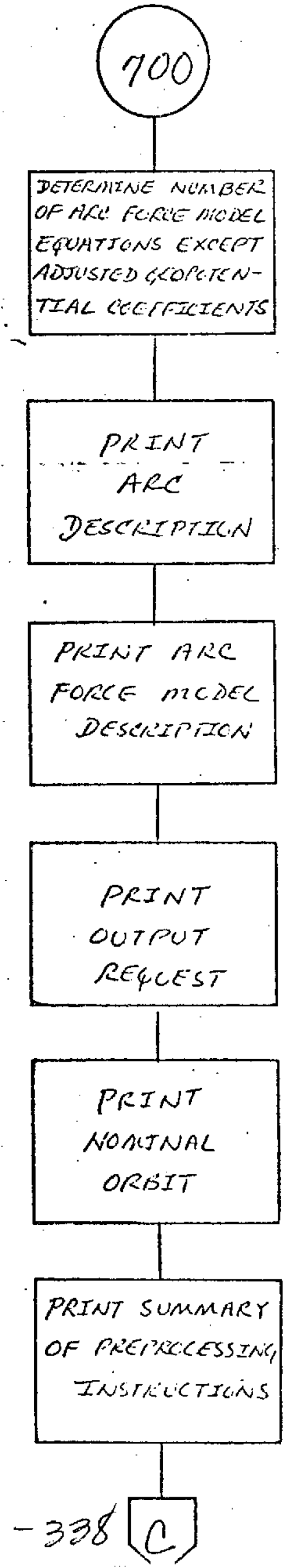

525
SET DATA TAPE UNITS
700

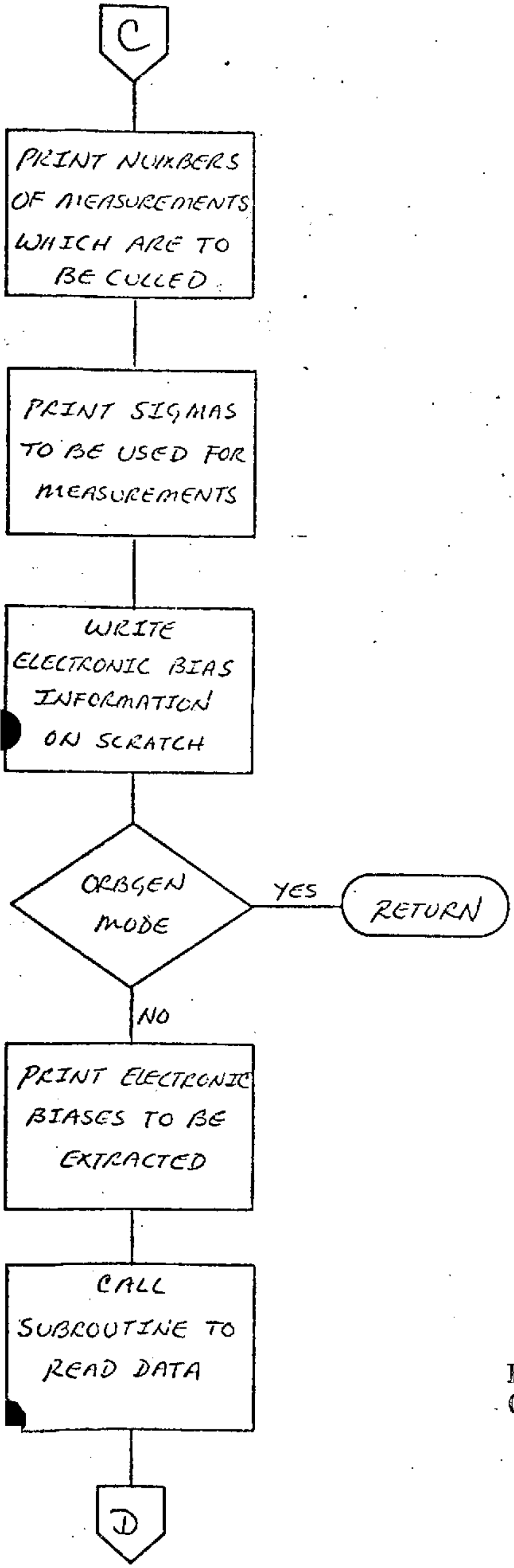

A
DATA CARD
YES
590
NO
560

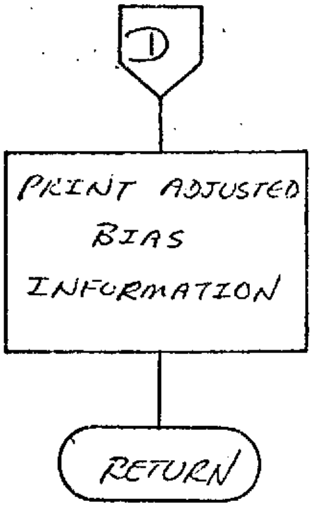

550
560
READ STATION POSITIONS
ENDFILE ENCOUNTERED
900
NO
A

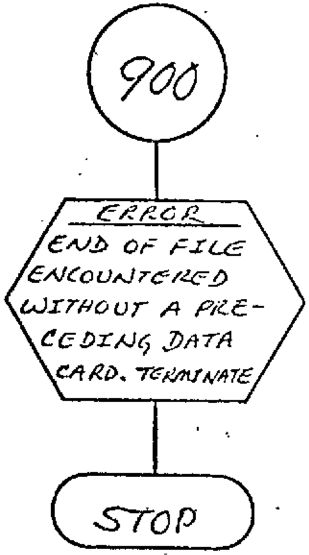

590
SET COMMON PARAMETER INDICATORS
COMPUTE FLATTENING
WRITE RUN TITLE
B


B
COEFL
PRINT GEOPOTENTIAL COEFFICIENTS AND EARTH MODEL
PRINT SURFACE DENSITIES
WRITE SURFACE DENSITY INFORMATION ON SCRATCH FILE
WRITE ADJUSTED GEOPOTENTIAL A PRIORI VALUES ON SCRATCH FILE
RETURN


700
DETERMINE NUMBER OF ARC FORCE MODEL EQUATIONS EXCEPT ADJUSTED GEOPOTENTIAL COEFFICIENTS
PRINT ARC DESCRIPTION
PRINT ARC FORCE MODEL DESCRIPTION
PRINT OUTPUT REQUEST
PRINT NOMINAL ORBIT
PRINT SUMMARY OF PREPROCESSING INSTRUCTIONS
C


C
PRINT NUMBERS OF MEASUREMENTS WHICH ARE TO BE CULLED
PRINT SIGMAS TO BE USED FOR MEASUREMENTS
WRITE ELECTRONIC BIAS INFORMATION ON SCRATCH
ORBGEN MODE
YES
RETURN
NO
PRINT ELECTRONIC BIASES TO BE EXTRACTED
CALL SUBROUTINE TO READ DATA
D


D
PRINT ADJUSTED BIAS INFORMATION
RETURN


900
ERROR
END OF FILE ENCOUNTERED WITHOUT A PRECEDING DATA CARD. TERMINATE
STOP

NAME INTRP

PURPOSE INTERPOLATION SUBROUTINE

CALLING SEQUENCE CALL INTRP(S,H,IORDER,NN,X,FCT,M,SUM)

SYMBOL	TYPE	DESCRIPTION
S	DP	INPUT - DESIRED OUTPUT TIME IN FRACTIONS OF A STEPSIZE FROM TIME POINT OF SUM ARRAY
H	DP	INPUT - STEPSIZE
IORDER	I	INPUT - ORDER
NN	I	INPUT - NUMBER OF EQUATIONS TO BE INTEGRATED
X (6,1)	DP	OUTPUT - OUTPUT ARRAY
FCT (3,1)	DP	INPUT - ARRAY OF ACCELERATION BACK VALUES
M	I	INPUT - DISPLACEMENT INDEX SET BY COWELL
SUM	DP	INPUT - SUM ARRAY UPDATED BY INTEGRATOR

SUBROUTINES USED COEF

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

```
      SUBROUTINE INTRP(S,H,IORDER,NN,X,FCT,M,SUM)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION AP(20),AV(20),X(6,1),FCT(3,1),SUM(2,3,1)
      DATA S1/0.D0/,IOR/0/
      IF(S.EQ.S1.AND.IOR.EQ.IORDER) GO TO 1
      S1=S
      IOR=IORDER
C DETERMINE INTERPOLATION COEFFICIENTS
      CALL COEF(S,IORDER,AP,AV)
      IOL2=IORDER-2
      IOL1=IOL2+1
    1 DO 100 N=1,NN
      K0=N*M+1
      DO 100 J=1,3
      A=0.D0
      B=0.D0
      DO 10 K=1,IOL2
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

80-340

```
      KK=K0-K
      A=A+AV(K)*FCT(J,KK)
   10 B=B+AP(K)*FCT(J,KK)
      A=A+AV(IOL1)*FCT(J,K0-IOL1)
      A=A+SUM(1,J,N)
      B=B+SUM(1,J,N)*(S-1.D0)+SUM(2,J,N)
      X(J,N)=B*H**2
      X(J+3,N)=A*H
  100 CONTINUE
      RETURN
      END
```

JANTHG

DESCRIPTION

JANTHG is a subroutine specifically designed for the GEODYN system. Its functions are to

- Recover the Greenwich mean sidereal time on Jan 0.0 of the reference year (θ_{g0}), and
- Set up the solar flux and geomagnetic activity variation tables,

for each arc.

Note that this routine computes the 81 day (3 solar rotations) midpoint average of solar flux values for each arc.

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME JANTHG

PURPOSE TO SELECT GREENWICH MEAN SIDEREAL TIME ON JAN 0.0 OF THE REFERENCE YEAR FOR EACH ARC AND SELECT FLUX DATA FOR EACH ARC FROM BLOCK DATA STORAGE AND COMPUTE AVERAGE SOLAR FLUX VALUES FOR EACH ARC

CALLING SEQUENCE JANTHG(IEPYMD,IY,SFLUX,MGFLUX,MYMD,LYMD)

SYMBOL	TYPE	DESCRIPTION
IEPYMD	I	INPUT - EPOCH YEAR, MONTH, AND DAY
IY	I	INPUT - LAST 2 DIGITS OF REFERENCE YEAR
SFLUX (1)	I*2	INPUT - SOLAR FLUX DATA
MGFLUX (1)	I*2	INPUT - MAGNETIC FLUX DATA
MYMD	I	INPUT - START DATE FOR FLUX DATA
LYMD	I	INPUT - STOP DATE FOR FLUX DATA

SUBROUTINES USED DIFF

COMMON BLOCKS CGEOS CONSTS FLXBLK

INPUT FILES NONE

OUTPUT FILES NONE

```
      DOUBLE PRECISION FUNCTION JANTHG(IEPYMD,IY,SFLUX,MGFLUX,MYMD,LYMD)
      DOUBLE PRECISION THETG0,CRAD,DTWOPI,AVFLX,DFLX,KP
      INTEGER*2 MGFLUX(1),SFLUX(1)
      COMMON/CGEOS/GS(2),THETG0(15),GC(423)
      COMMON/CONSTS/DPI(2),DTWOPI,CRAD,CRSEC(2)
      COMMON/FLXBLK/AVFLX(675),DFLX(675),KP(675)
C RIGHT ASCENSION OF GREENWICH FOR REFERENCE JAN 0.0
      JANTHG=THETG0(IY-57)*CRAD
      CALL DIFF(MYMD,0,IEPYMD,0,IDAY,ISEC)
      CALL DIFF(MYMD,0,LYMD,0,IDAY1,ISEC)
C FLUX VALUES ARE FOR 12 HRS. GMT OF TABULAR DATE
C NEED FLUX VALUES 1.5 DAYS BEFORE EARLIEST DATE FOR INTEGRATION
      IDAY=IDAY-2
      IDAY2=IDAY
      IF(IDAY2.LT.0) IDAY2=0
      IF(IDAY.LT.40) IDAY=40
      IDAY1=IDAY1+1
      DO 20 I=1,675
      I1=I+IDAY2
```

```
      IF(I1.GT.IDAY1) I1=IDAY1
      KP(I)=MGFLUX(I1)*1.0E-1
      DFLX(I)=SFLUX(I1)*1.0E-1
      I2=I+IDAY+40
      IF(I2.GT.IDAY1) I2=IDAY1
      I1=I2-80
      AVFLX(I)=0.0D0
C COMPUTE FLUX AVERAGE
      DO 15 J=I1,I2
   15 AVFLX(I)=AVFLX(I)+SFLUX(J)
   20 AVFLX(I)=AVFLX(I)/810.0D0
      RETURN
      END
      DOUBLE PRECISION THETGO,DRAD,DTWCPI,AVFLX,DFLX,AP
      INTEGER*2 MGFLUX(1),SFLUX(1)
      COMMON/CGEOS/G5(2),THETGO(15),G6(423)
      COMMON/CONSTS/DPI(2),DTWCPI,DRAD,DESEC(2)
      COMMON/FLXBLK/AVFLX(675),DFLX(675),AP(675)
      JANTHG=THETGO(IY-57)*DRAD
      CALL DIFF(MYMD,0,IBFYMD,0,IDAY,ISEC)
      CALL DIFF(MYMD,0,LYMD,0,IDAY1,ISEC)
      IDAY2=IDAY
      IF(IDAY2.LT.0) IDAY2=0
      IF(IDAY.LT.54) IDAY=54
      IDAY1=IDAY1+1
      DO 20 I=1,675
      I1=I+IDAY2
      IF(I1.GT.IDAY1) I1=IDAY1
      AP(I)=MGFLUX(I1)
      DFLX(I)=SFLUX(I1)*1.0E-1
      I2=I+IDAY
      IF(I2.GT.IDAY1) I2=IDAY1
      I1=I2-54
      AVFLX(I)=0.0D0
      DO 15 J=I1,I2
   15 AVFLX(I)=AVFLX(I)+SFLUX(J)
   20 AVFLX(I)=AVFLX(I)/550.D0
      RETURN
      END
```

NAME	MULMAT
PURPOSE	TO MULTIPLY THREE 3X3 MATRICES
CALLING SEQUENCE	CALL MULMAT(X,X1,X2,X3)

SYMBOL	TYPE	DESCRIPTION
X1 (3,3)	DP	INPUT - MATRIX OF DIMENSION 3X3
X2 (3,3)	DP	INPUT - MATRIX OF DIMENSION 3X3
X3 (3,3)	DP	INPUT - MATRIX OF DIMENSION 3X3
X (3,3)	DP	OUTPUT - PRODUCT OF THE THREE 3X3 MATRICES

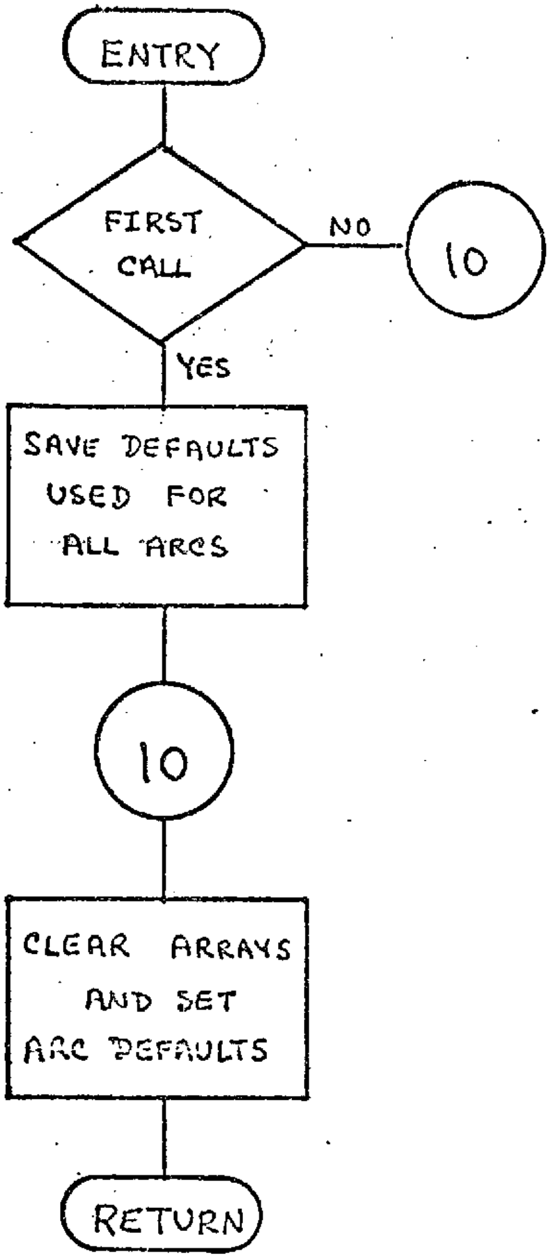
SUBROUTINES USED	NONE
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      SUBROUTINE MULMAT(X,X1,X2,X3)
      REAL*8 X(3,3),X1(3,3),X2(3,3),X3(3,3),SUM
      DO 40 J=1,3
      DO 10 I=1,3
 10   X(I,J)=0.D0
      DO 40 K=1,3
      SUM=0.D0
      DO 20 L=1,3
 20   SUM=SUM+X2(K,L)*X3(L,J)
      DO 40 I=1,3
 40   X(I,J)=X(I,J)+SUM*X1(I,K)
      RETURN
      END
```

NAME	NEWARC
PURPOSE	TO INITIALIZE SWITCHES AND CONSTANTS FOR EACH ARC
CALLING SEQUENCE	CALL NEWARC
SUBROUTINES USED	CLEAR
COMMON BLOCKS	APARAM CELEM CGEOS CONSTS CTIME FLXBLK FMODEL INITBK INTBLK PREBLK PRIORI TPEBLK VRBLCK
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      SUBROUTINE NEWARC
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL NOT1ST,TORSFT
      INTEGER XYZTP,RVTP,SCRA,SCRC,FLTP,PLOTP,DATP,OUTP,ESTSTA,GRDTP
      REAL RMSTOT,VARCOV,EDITN
      DOUBLE PRECISION MODEL,LOVE
      DIMENSION SBODY(6),SSTEP(20)
      COMMON/APARAM/INPAR,INPAR1,NBIAS,ESTSTA,NSAT,NGPARC,NURECI,NPARAM,
     .   NSBIAS,MAXPAR
      COMMON/CELEM/ELEMST(24),XNU,EC,RMSTOT
      COMMON/CGEOS/ISATID(252),NCRERR(203)
      COMMON/CONSTS/DPI,DTWOPI,DRAD,DRSEC
      COMMON/CTIME/DATAEP(3),DAYSTR,DAYINT,DORBIT,DAYEND,DRATE,DORB1,
     .   DORBIE(2),IY
      COMMON/FLXBLK/ISTORE(450,9)
      COMMON/FMODEL/INDEX1,NDEX2,INDEX3,NDEX4,CS(30,33),MODEL(8)
      COMMON/INITBK/IG1(57)
      COMMON/INTBLK/THDOT1,THDOT2,THDT25,IG2(50),RODIFS(6),STEPSZ(24),
     .   NSON(8),LOVE(3),TORSFT,NBODY
      COMMON/PREBLK/DAYSTA,JOBS(15)
      COMMON/PRIORI/ELEMIN(12),VARCOV(6,5,2),TITLE(30),DRAG(18)
      COMMON/TPEBLK/INTP,OUTP,DATP,XYZTP,KEPTAP,RVTP,PLOTP,IOBS,SCRA,
     .   SCRC,FLTP,GRDTP
      COMMON/VRBLCK/JSTORE(450,5)
      EQUIVALENCE (EDITN,IG1(33))
      DATA NOT1ST/.FALSE./
      EDITN=3.5
      CALL CLEAR(IG1(34),24,1)
      IF(NOT1ST) GO TO 20
C SAVE DEFAULT CONSTANTS USED FOR ALL ARCS
      NOT1ST=.TRUE.
      INDS1=INDEX1
      INDS3=INDEX3
      NBODYS=NBODY
      DO 5 I=1,6
    5 SBODY(I)=BODIES(I)
```

```
      THDOT1=THDOT1*DRAD                                               NEWA  56
      THDOT2=THDOT1+DTWOPI                                             NEWA  57
      THDT2S=THDOT2/8.64D4                                             NEWA  58
      DO 10 I=1,20                                                     NEWA  59
C CLEAR ARRAYS AND SET ARC DEFAULTS                                     NEWA  60
   10 SSTEP(I)=STEPSZ(I)                                               NEWA  61
   20 DO 25 I=1,20                                                     NEWA  62
   25 STEPSZ(I)=SSTEP(I)                                               NEWA  63
      NBODY=NBODYS                                                     NEWA  64
      DO 26 I=1,6                                                      NEWA  65
   26 BODIES(I)=SBODY(I)                                               NEWA  66
      CALL CLEAR(NEQN,8,1)                                             NEWA  67
      TOREFT=.FALSE.                                                   NEWA  68
      DRATE=999.0D0                                                    NEWA  69
      DORBIT=0.D0                                                      NEWA  70
      DAYEND=0.D0                                                      NEWA  71
      DAYSTA=999.D0                                                    NEWA  72
      DAYSTP=999.D0                                                    NEWA  73
      DORB1=-1.D0                                                      NEWA  74
      RMSTOT=200.                                                      NEWA  75
      CALL CLEAR(VARCOV,84,2)                                          NEWA  76
      CALL CLEAR(ISTORE,450,9)                                         NEWA  77
      CALL CLEAR(JSTORE,450,5)                                         NEWA  78
      DO 30 I=1,6                                                      NEWA  79
      VARCOV(I,I,1)=1.0D+14                                            NEWA  80
   30 VARCOV(I,I,2)=1.0D+14                                            NEWA  81
      INPAR=0                                                          NEWA  82
      NSTSTA=0                                                         NEWA  83
      NPARAM=0                                                         NEWA  84
      NEBIAS=0                                                         NEWA  85
      MAXPAR=0                                                         NEWA  86
      NBIAS=0                                                          NEWA  87
      INPARI=0                                                         NEWA  88
      DRAG(17)=1.5D0                                                   NEWA  89
      DRAG(18)=1.5D0                                                   NEWA  90
      CALL CLEAR(NOPRER,3,1)                                           NEWA  91
      INDEX1=INDS1                                                     NEWA  92
      INDEX3=INDS3                                                     NEWA  93
      KEPTAP=9                                                         NEWA  94
      NOREC1=0                                                         NEWA  95
      NGPARC=900                                                       NEWA  96
      NSAT=1                                                           NEWA  97
      IOBS=20                                                          NEWA  98
      XYZTP=8                                                          NEWA  99
      RVTP=0                                                           NEWA 100
      RETURN                                                           NEWA 101
      END                                                              NEWA 102
```


ENTRY
FIRST CALL
NO
10
YES
SAVE DEFAULTS USED FOR ALL ARCS
10
CLEAR ARRAYS AND SET ARC DEFAULTS
RETURN

NAME NUMBR2

PURPOSE TO SEARCH AN ARRAY TO DETERMINE IF THE ARRAY CONTAINS AN ENTRY WHICH MATCHES GIVEN OUTPUT NUMBER IF FOUND THE INDEX NUMBER OR LOCATION IN THE ARRAY 'IA' IS RETURNED. 0 IS RETURNED IF NO MATCH IS FOUND

CALLING SEQUENCE X=NUMBR2(K.IA.ID)

SYMBOL	TYPE	DESCRIPTION
K	I	INPUT - NUMBER OR BIT CONFIGURATION TO BE LOCATED IN ARRAY
IA (ID)	I*2	INPUT - ARRAY TO BE SEARCHED
ID	I	INPUT - NUMBER OF ENTRIES IN ARRAY 'IA'
NUMBR2	I	OUTPUT - THE NUMBER OF ENTRY THAT MATCHED THE GIVEN INPUT NUMBER

SUBROUTINES USED NONE

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

```
      INTEGER FUNCTION NUMBR2(K.IA.ID)                                  NUMB  35
      INTEGER*2 IA(ID)                                                  NUMB  36
      NUMBR2=0                                                          NUMB  37
      IF(ID.LT.1) RETURN                                                NUMB  38
      DO 10 I=1,ID                                                      NUMB  39
      IF(K.EQ.IA(I)) GO TO 20                                           NUMB  40
   10 CONTINUE                                                          NUMB  41
      RETURN                                                            NUMB  42
   20 NUMBR2=I                                                          NUMB  43
      RETURN                                                            NUMB  44
      END                                                               NUMB  45
```

NAME	NUMBR4
PURPOSE	TO SEARCH AN ARRAY TO DETERMINE IF THE ARRAY CONTAINS AN ENTRY WHICH MATCHES A GIVEN INPUT NUMBER. IF FOUND THE INDEX NUMBER OR LOCATION IN THE ARRAY 'IA' IS RETURNED. 0 IS RETURNED IF NO MATCH IS FOUND
CALLING SEQUENCE	NUMBR4(K,IA,ID)

SYMBOL	TYPE	DESCRIPTION
K	I	INPUT - NUMBER OR BIT CONFIGURATION TO BE LOCATED IN 'IA' ARRAY
IA (ID)	I	INPUT - ARRAY TO BE SEARCHED
ID	I	INPUT - NUMBER OF ENTRIES IN 'IA' ARRAY
NUMBR4	I	OUTPUT - INDEX NUMBER OF THE MEMBER OF 'IA' ARRAY WHICH CONTAINS THE ENTRY WHICH MATCHES K

SUBROUTINES USED	NONE
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      INTEGER FUNCTION NUMBR4(K,IA,ID)
      INTEGER*4 IA(ID)
      NUMBR4=0
      IF(ID.LT.1) RETURN
      DO 10 I=1,ID
      IF(K.EQ.IA(I)) GO TO 20
   10 CONTINUE
      RETURN
   20 NUMBR4=I
      RETURN
      END
```

NAME	NUMLOC
PURPOSE	TO SEARCH AN ARRAY TO DETERMINE IF THE ARRAY CONTAINS ENTRIES WHICH MATCH GIVEN INPUT NUMBER. IF FOUND THE INDEX NUMBERS OF LOCATIONS IN THE ARRAY 'IA' AND THE NUMBER OF SUCH MATCHING ENTRIES FOUND ARE RETURNED
CALLING SEQUENCE	NUMLOC(NO,IA,NT,LOC)

SYMBOL	TYPE	DESCRIPTION
NO	I	INPUT - NUMBER OR BIT CONFIGURATION TO BE LOCATED IN ARRAY
IA (1)	I*2	INPUT - ARRAY TO BE SEARCHED
NT	I	INPUT - NUMBER OF ENTRIES IN 'IA' ARRAY
LOC (1)	I*2	OUTPUT - THE INDEX NUMBERS OF THE MEMBERS OF 'IA' ARRAY WHICH MATCH 'NO'
NUMLOC	I	OUTPUT - THE NUMBER OF ENTRIES THAT MATCH THE GIVEN INPUT NUMBER

SUBROUTINES USED	NUMBR2
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      INTEGER FUNCTION NUMLOC(NO,IA,NT,LOC)
      INTEGER*2 LOC(1),IA(1)
      NUMLOC=0
      I1=NUMBR2(NO,IA,NT)
      IF(I1.EQ.0) RETURN
      N2=NT-I1
      NUMLOC=1
      LOC(1)=I1
      DO 20 I=1,NT
      J1=LOC(NUMLOC)+1
      I1=NUMBR2(NO,IA(J1),N2)
      IF(I1.EQ.0) RETURN
      NUMLOC=NUMLOC+1
      LOC(NUMLOC)=I1+J1-1
   20 N2=N2-I1
      RETURN
      END
```

NUTATE

DESCRIPTION

Subroutine NUTATE generates the rotation matrix to nutate a vector from true to mean equator and equinox of date.

EQN is invoked to compute the nutation in longitude, the nutation in obliquity; and the true obliquity of the ecliptic. The rotation matrices are constructed by ROTMAT; MULMAT evaluates the output rotation matrix as a product of the three input rotation matrices.

NAME	NUTATE
PURPOSE	TO GENERATE NUTATION ANGLES TO TRANSFORM FROM TRUE EQUATOR AND EQUINOX TO MEAN EQUATOR AND EQUINOX
CALLING SEQUENCE	CALL NUTATE(DAY,X)

SYMBOL	TYPE	DESCRIPTION
DAY	DP	INPUT - TIME FOR ANGLES
X (3,3)	DP	OUTPUT - NUTATION MATRIX

SUBROUTINES USED	ROTMAT	MULMAT	EQN	YMDAY
COMMON BLOCKS	INITBK			
INPUT FILES	NONE			
OUTPUT FILES	NONE			
REFERENCES	'GEODYN SYSTEMS DESCRIPTION' VOLUME 1 - GEODYN DOCUMENTATION			

```
      SUBROUTINE NUTATE(DAY,X)
      DOUBLE PRECISION DAY,X(3,3),DJBASE,DELPSI,DJ,DEPS,EPS,YMDAY
      REAL*8 X1(3,3),X2(3,3),X3(3,3)
      COMMON/INITBK/IC1(52),NOT1ST,IC2(4)
      LOGICAL NOT1ST
      IF(NOT1ST) GO TO 10
      DJBASE=2433281.500-YMDAY(500100,0,0.D0)
      NOT1ST=.TRUE.
   10 DJ=DJBASE+DAY
      G=EQN(DJ,DELPSI,DEPS,EPS)
      CALL ROTMAT(EPS,1,X1)
      CALL ROTMAT(+DELPSI,3,X2)
      CALL ROTMAT(-EPS+DEPS,1,X3)
      CALL MULMAT(X,X3,X2,X1)
      RETURN
      END
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

8.0-353

OBSDOT

DESCRIPTION

Subroutine OBSDOT calculates the time derivatives of the computed observations. The measurement types implemented are:

1. Right ascension and declination
2. Range
3. Range Rate
4. Not used
5. ℓ and m direction cosines
6. X and Y angles
7. Azimuth and elevation

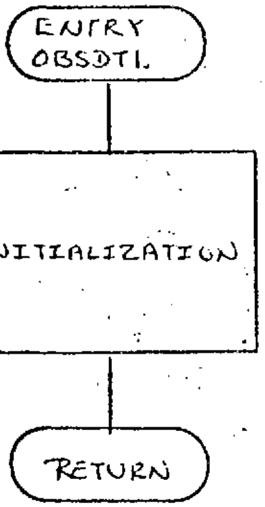
Note that the functions XEFIX and YEFIX are treated as transformations. When applied to the velocity vector, the output of these transformations differs from the Earth fixed velocity by a term involving the rotation rate of the Earth.

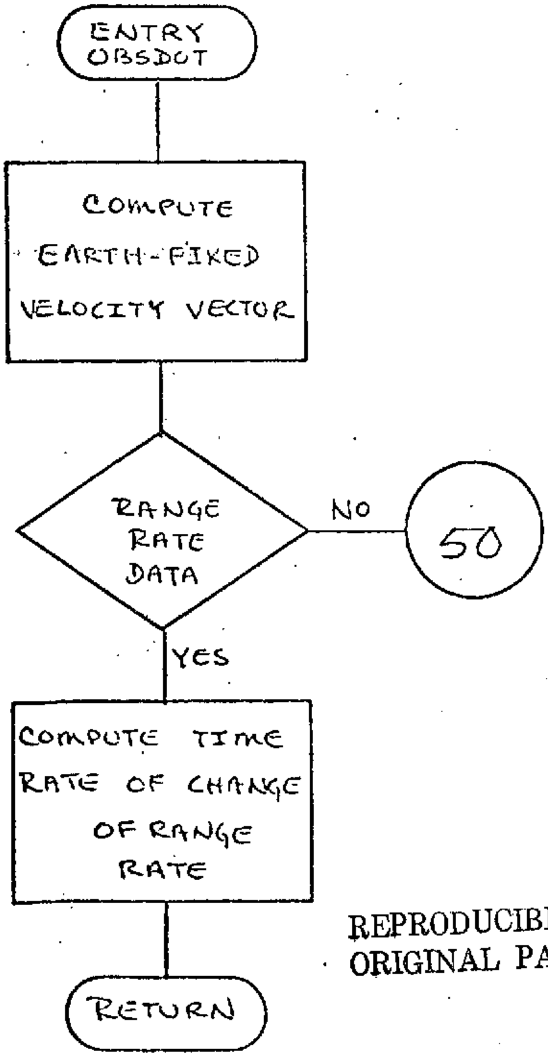
```
NAME                OBSDOT

ENTRY POINT         PURPOSE

 OBSDT1             INITIALIZATION

 OBSDOT             TO CALCULATE THE TIME DERIVATIVES OF COMPUTED
                    OBSERVATIONS

 CALLING SEQUENCE   OBSDT1(EHAT,NHAT,ZHAT)

   SYMBOL   TYPE    DESCRIPTION

   EHAT      DP     INPUT - STATION UNIT EAST VECTOR
   (3,1)

   NHAT      DP     INPUT - STATION UNIT NORTH VECTOR
   (3,1)

   ZHAT      DP     INPUT - STATION UNIT VERTICAL VECTOR
   (3,1)

 CALLING SEQUENCE   OBSDOT(MTYPE,ISTA,OBSDT2)

   SYMBOL   TYPE    DESCRIPTION

   MTYPE     I      INPUT - MEASUREMENT TYPE

   ISTA      I      INPUT - INTERNAL STATION NUMBER

   OBSDT2    DP     OUTPUT - TIME DERIVATIVE OF SECOND MEASUREMENT

   OBSDOT    DP     OUTPUT - TIME DERIVATIVE OF FIRST MEASUREMENT

SUBROUTINES USED    YEFIX       DOTPRD      XEFIX

COMMON BLOCKS       CLVECT      INTBLK      PREBLK      XYZOUT

INPUT FILES         NONE

OUTPUT FILES        NONE

RESTRICTIONS        COMPUTE THE TIME DERIVATIVES OF MEASUREMENT TYPES:
                      RIGHT ASCENSION AND DECLINATION, RANGE, RANGE
                      RATE, L AND M DIRECTION COSINES, X AND Y ANGLES,
                      AZIMUTH AND ELEVATION

REFERENCES          'GEODYN SYSTEMS DESCRIPTION'
                    VOLUME 1 - GEODYN DOCUMENTATION
```

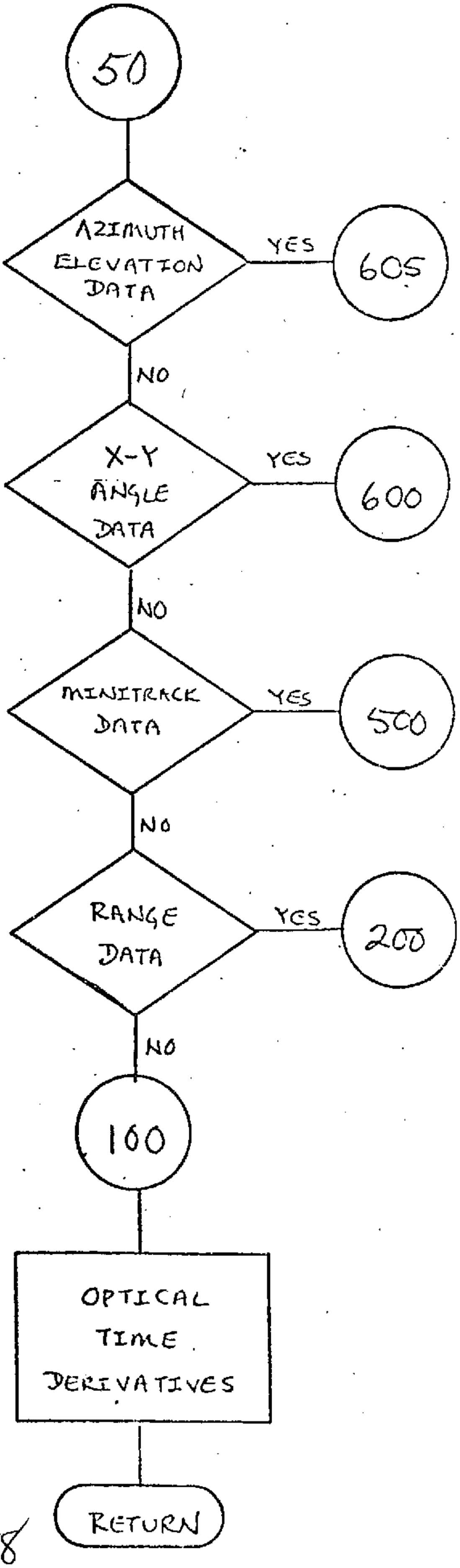
```
      DOUBLE PRECISION FUNCTION OBSDT1(EHAT,NHAT,ZHAT)                 OBSD  54
      IMPLICIT REAL*8 (A-H,O-Z)                                        OBSD  55
```

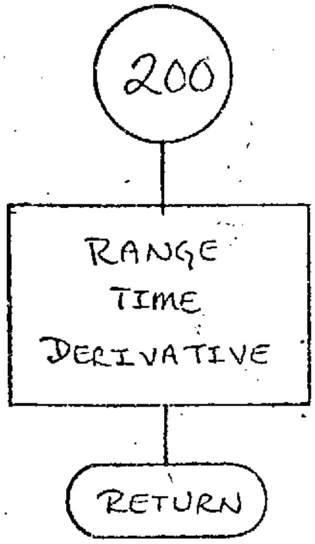
```
      INTEGER*2 ISAT,PRETYP
      DOUBLE PRECISION NHAT
      DIMENSION EHAT(3,1),NHAT(3,1),ZHAT(3,1),DELDOT(3),DELDDT(3)
      COMMON/CUVECT/UHAT(3,2),XYZ(3,2),RXYZ(3,2),RENV(3,2),R(2),RSQ(2),
     *   XYSQ(2)
      COMMON/INTBLK/THETGO(2),THDT2S,GM,AE(62)
      COMMON/PREBLK/DAYSTA(7),ISAT,PRETYP(5)
      COMMON/XYZOUT/XYZI(6,4)
      RETURN
      ENTRY OBSDOT(MTYPE,ISTA,OBSDT2)
C CALCULATE SATELLITE EARTH-FIXED VELOCITY VECTOR
      OBSDT2=0.0D0
      OBSDOT=0.0D0
      IF(MTYPE.GT.7) RETURN
      DELDOT(1)=XEFIX(XYZI(4,ISAT),XYZI(5,ISAT))
      DELDOT(2)=YEFIX(XYZI(4,ISAT),XYZI(5,ISAT))
      DELDOT(3)=XYZI(6,ISAT)
      IF(MTYPE.NE.3) GO TO 50
C COMPUTE TIME RATE OF CHANGE OF...
C
C ...RANGE RATE
      R1SQ=DOTPRD(XYZI(1,ISAT),XYZI(1,ISAT))
      R1=DSQRT(R1SQ)
      C4=GM/(R1SQ*R1)
      C=5.0D0*(XYZI(3,ISAT)/R1)**2-1.0D0
      C2=C-2.0D0
      C3=6.506446D10/R1SQ
      C=(1.0D0-C*C3)*C4
      DELDDT(1)=-C*XYZ(1,ISAT)+THDT2S*DELDOT(2)
      DELDDT(2)=-C*XYZ(2,ISAT)-THDT2S*DELDOT(1)
      DELDDT(3)=-(1.0D0-C2*C3)*C4*XYZI(3,ISAT)
      DELDOT(1)=DELDOT(1)+THDT2S*XYZ(2,ISAT)
      DELDOT(2)=DELDOT(2)-THDT2S*XYZ(1,ISAT)
      DELDDT(1)=DELDDT(1)+THDT2S*DELDOT(2)
      DELDDT(2)=DELDDT(2)-THDT2S*DELDOT(1)
      RDOT=DOTPRD(DELDOT,UHAT(1,ISAT))
      OBSDOT=DOTPRD(DELDOT,DELDOT)/R(ISAT)+DOTPRD(UHAT(1,ISAT),DELDDT)
     *   -RDOT**2/R(ISAT)
      RETURN
   50 DELDOT(1)=DELDOT(1)+THDT2S*XYZ(2,ISAT)
      DELDOT(2)=DELDOT(2)-THDT2S*XYZ(1,ISAT)
      RDOT=DOTPRD(DELDOT,UHAT(1,ISAT))
      GO TO(100,200,700,700,500,600,605),MTYPE
C ...OPTICAL
  100 C=1.0D0-UHAT(3,ISAT)**2
      OBSDT2=(XYZI(6,ISAT)-UHAT(3,ISAT)*RDOT)/(R(ISAT)*DSQRT(C))
      OBSDOT=(UHAT(1,ISAT)*DELDOT(2)-UHAT(2,ISAT)*DELDOT(1))/(R(ISAT)*C)
      RETURN
C ...RANGE
  200 OBSDOT=RDOT
      RETURN
C ...MINITRACK
  500 OBSDOT=(DOTPRD(DELDOT,EHAT(1,ISTA))-RENV(1,ISAT)*RDOT)/R(ISAT)
      OBSDT2=(DOTPRD(DELDOT,NHAT(1,ISTA))-RENV(2,ISAT)*RDOT)/R(ISAT)
      RETURN
C ...X-Y ANGLES
```

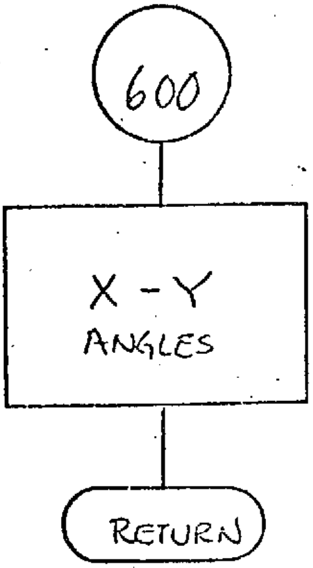
```
 600 C=1.0D0-RENV(2,ISAT)**2
      OBSDOT=(RENV(3,ISAT)*DOTPRD(DELDOT,EHAT(1,ISTA))-RENV(1,ISAT)*
     .   DOTPRD(DELDOT,ZHAT(1,ISTA)))/(C*R(ISAT))
      OBSDT2=(DOTPRD(DELDOT,NHAT(1,ISTA))-RDOT*RENV(2,ISAT))/(R(ISAT)*
     .   DSQRT(C))
      RETURN
C ...AZIMUTH AND ELEVATION
 605 C=1.0D0-RENV(3,ISAT)**2
      OBSDOT=(RENV(2,ISAT)*DOTPRD(DELDOT,EHAT(1,ISTA))-RENV(1,ISAT)*
     .   DOTPRD(DELDOT,NHAT(1,ISTA)))/(C*R(ISAT))
      OBSDT2=(DOTPRD(DELDOT,ZHAT(1,ISTA))-RDOT*RENV(3,ISAT))/(R(ISAT)*
     .   DSQRT(C))
 700 RETURN
      END
```

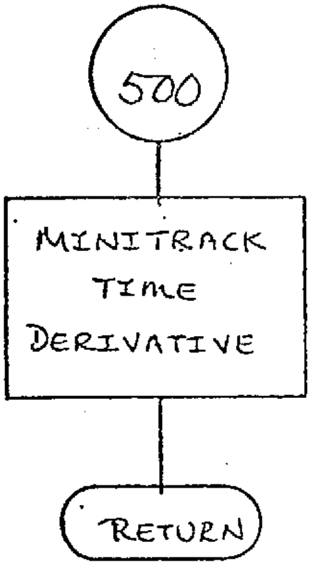

ENTRY OBSDTL
INITIALIZATION
RETURN

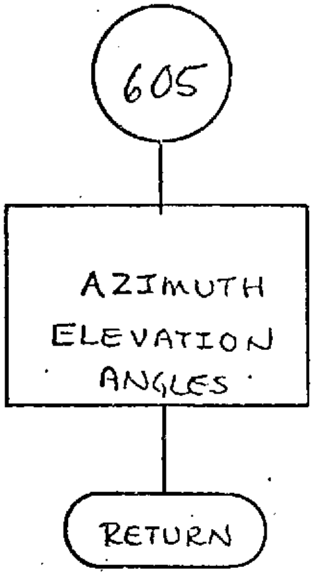

ENTRY OBSDOT
COMPUTE EARTH-FIXED VELOCITY VECTOR
RANGE RATE DATA
NO
50
YES
COMPUTE TIME RATE OF CHANGE OF RANGE RATE
RETURN


50
AZIMUTH ELEVATION DATA
YES
605
NO
X-Y ANGLE DATA
YES
600
NO
MINITRACK DATA
YES
500
NO
RANGE DATA
YES
200
NO
100
OPTICAL TIME DERIVATIVES
RETURN


200
RANGE
TIME
DERIVATIVE
RETURN


600
X - Y
ANGLES
RETURN


500
MINITRACK
TIME
DERIVATIVE
RETURN

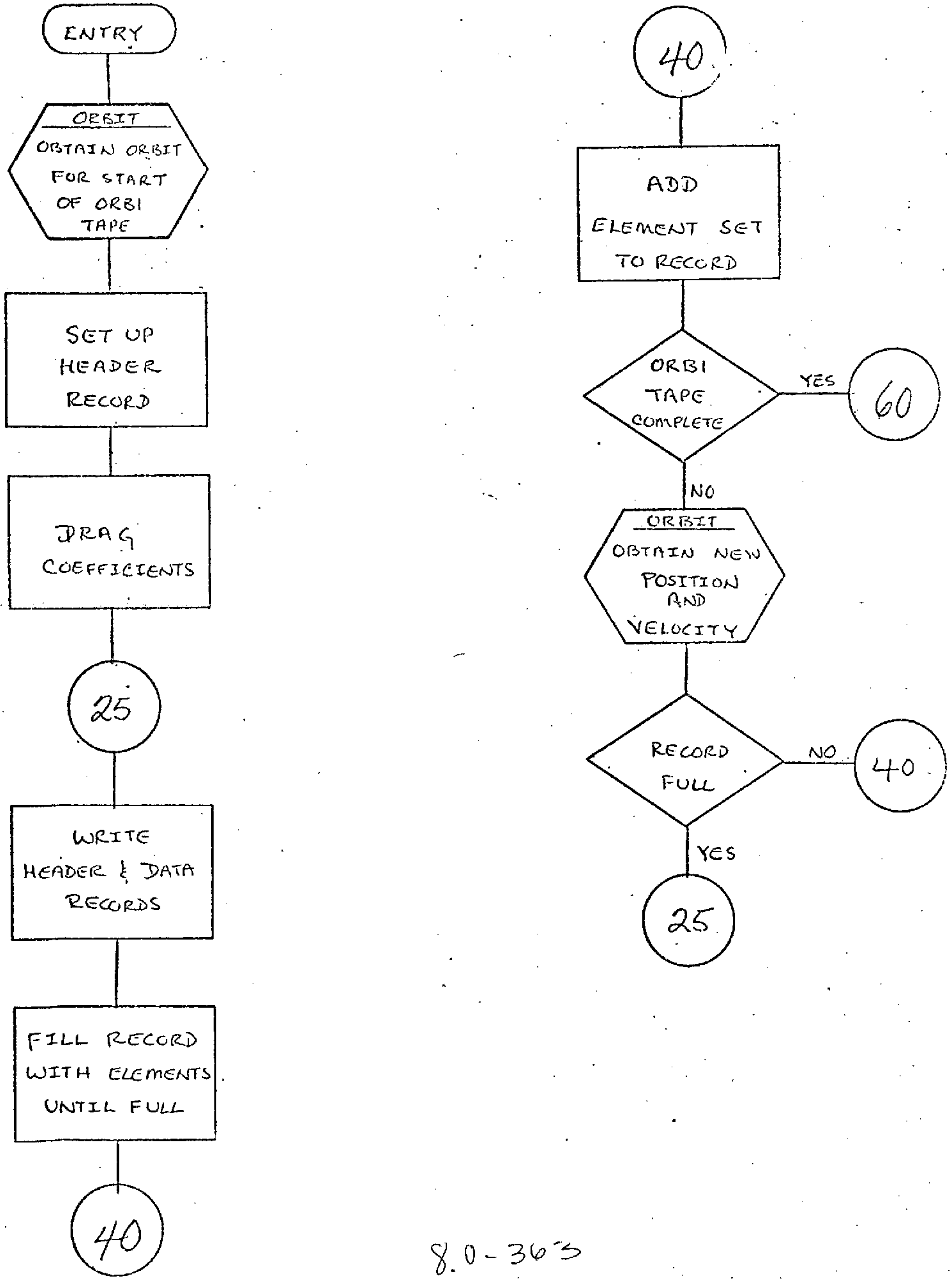

605
AZIMUTH
ELEVATION
ANGLES
RETURN

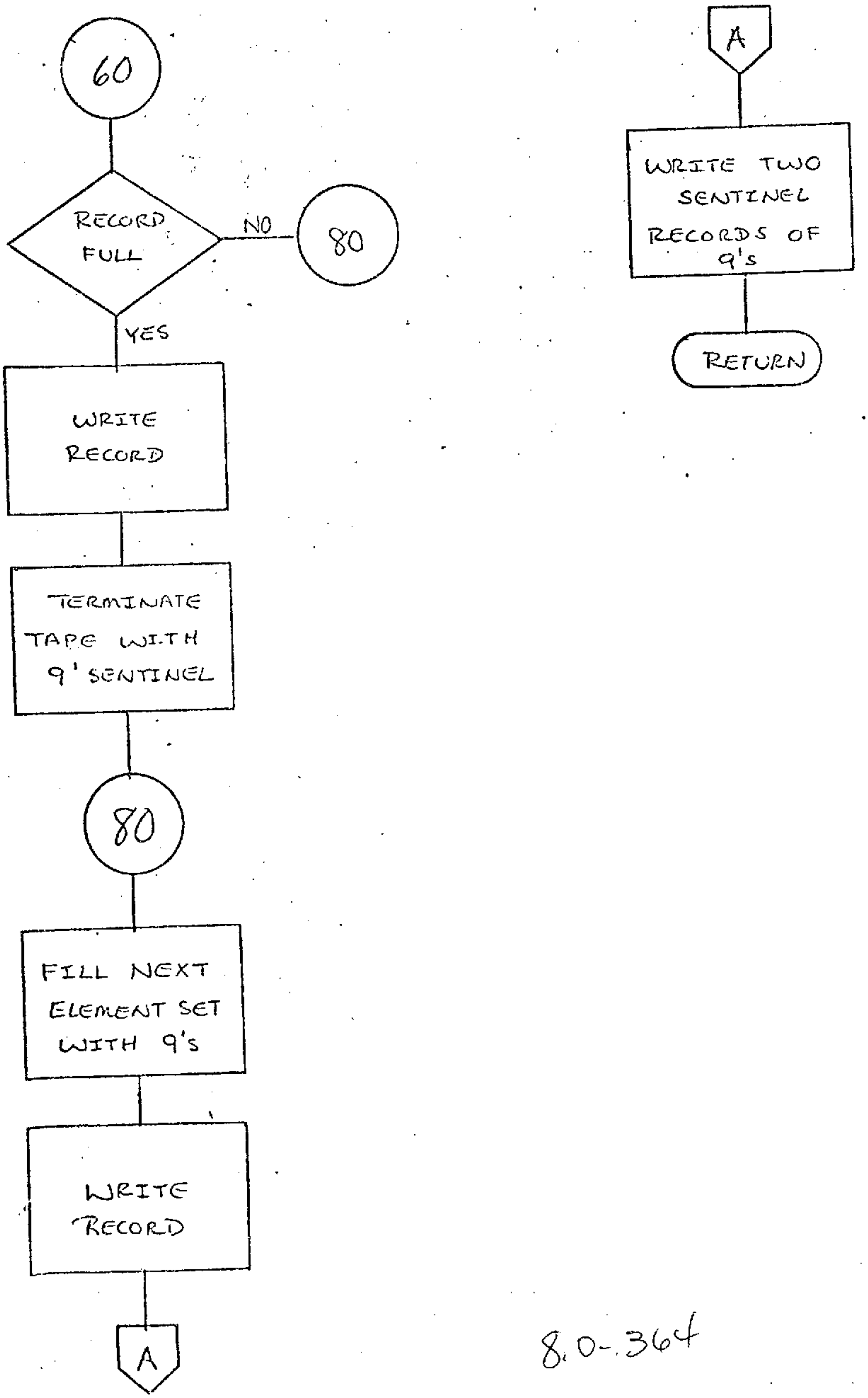
NAME	ORB1				
PURPOSE	TO GENERATE A SATELLITE EPHEMERIS TAPE IN ORB1 TAPE FORMAT				
CALLING SEQUENCE	CALL ORB1(ORB1RT)				
SYMBOL TYPE	DESCRIPTION				
ORB1RT DP	INPUT - ORB1 TAPE OUTPUT RATE IN INTEGRAL SECONDS				
SUBROUTINES USED	ORBIT	DAYEAR	ELEM	DATES	ERROR
	EQN	DJUL	DOTPRD		
COMMON BLOCKS	CONSTS	EMODEL	CELEM	CGEOS	CORB1
	INTBLK	PRIORI	XYZOUT	CTIME	
INPUT FILES	NONE				
OUTPUT FILES	22 - ORB1 FILE				
REFERENCE	'GEODYN PROGRAM OPERATIONS DESCRIPTION' - APPENDIX C VOLUME 3 - GEODYN DOCUMENTATION				

```
      SUBROUTINE ORB1(ORB1RT)
      COMMON/CTIME/DATAEP(2),DAYREF,DSTART,DAYSTP(10),DORB1,DORB1E,
     . ORBR1(2),IYMG
      REAL*8 ELEMIN,CD,CDD,EMISS,SEC
      COMMON/CONSTS/DPI,DTWOPI,DRAD,DRSEC(2)
      REAL*8 MODEL,CS
      COMMON/EMODEL/INDEX1, NDEX2,INDEX3,INDEX4,CS(30,33),MODEL(8)
      REAL*8 ORGPAR,MMOON,MSUN,ASAT,MSAT
      REAL*8 BUF(350),TBUF(6,59),DORB1,LORB1,ORB1RT,XYZEND,DOTPRD,
     .        DAYREF,THETG0,THDOT1,EQN,DJUL,G1,G2,ORBELA,ELEMST,AE,GM,
     .        DPI,DTWOPI,DRAD,AE,DSTART,DAYREF,GM,XNU,EC,PRD,APHT,PLPHT,
     .   RANDOT,PERDOT,THDT2S,G3
      COMMON/CELEM/ELEMST(12),ORBELA(12),XNU,EC,RMSTOT
      COMMON/CGEOS/ISATID,G4(454)
      COMMON/CORB1/RANDOT(2),PERDOT(2),PLPHT(2),APHT(2),PRD(2)
      COMMON/INTBLK/THDOT1,THDOT2(2),THDT2S,GM,AE,AESQ(44),THETG0,
     .   MMOON,MSUN,MBODY(44),ASAT(2),MSAT(2),VARSTR(20)
      COMMON/PRIORI/ELEMIN(92),CD(2),CDD(2),EMISS(2)
      COMMON/XYZOUT/XYZEND(12),ORGPAR(12)
      REAL*8 NINES/99999999.D0/
      INTEGER IORB1/22/,NFILES/0/
      EQUIVALENCE (BUF(6),TBUF(1,1))
C OBTAIN ORBIT FOR BEGINNING OF ORB1 TAPE
      CALL ORBIT(DORB1)
C SET UP HEADER RECORD
      DO 10 I=1,350
   10 BUF(I)=0.D0
      BUF(1)=76796291.D0
```

```
      BUF(2)=ISATID
      CALL DAYEAR(DORBI,IYMD,IDAY,ISEC)
      BUF(4)=IYMD
      BUF(5)=IDAY
      BUF(6)=ISEC
      CALL DAYEAR(DORBIE,IYMD,IDAY,ISEC)
      BUF(7)=IYMD
      BUF(8)=IDAY
      BUF(9)=ISEC
      DT=IDINT(ORB1RT*86400.D0+.5D0)
      BUF(10)=DT
      CALL DAYEAR(DAYREF,IYMD,IDAY,ISEC)
      BUF(27)=IYMD
      BUF(28)=IDAY
      BUF(29)=THETG0+IDINT(DAYREF)*THDOT1+EQN(DJUL(DAYREF),G1,G2,G3)
      CUL=AE
      BUF(37)=-1.5D0*CS(2,1)*CUL**2
      BUF(38)=-2.5D0*CS(3,1)*CUL**3
      BUF(39)=+3.75D0*CS(4,1)*CUL**4
      BUF(40)=+1.D0*CS(5,1)*CUL**5
      CUT=DSQRT(AE**3/GM)
      BUF(80)=CD(1)
      BUF(81)=ASAT(1)*1.004
      BUF(82)=MSAT(1)*1.003
      IF(MMOON.GT.0.0D0) BUF(91)=1.0D0
      IF(MSUN.GT.0.0D0) BUF(92)=1.0D0
      CALL ELEM(ELEMST,ORBELA,1,.TRUE.)
      BUF(101)=(DSTART-DAYREF)*86400.D0/CUT
      BUF(102)=ORBELA(1)/CUL
      BUF(103)=ORBELA(2)
      BUF(104)=XNU
      DO 15 I=1,6
15    BUF(104+I)=ELEMST(I)/CUL
      DO 20 I=4,6
20    BUF(104+I)=BUF(104+I)*CUT
      BUF(111)=DSQRT(DOTPRD(BUF(105),BUF(105)))
      BUF(112)=DSQRT(DOTPRD(BUF(108),BUF(108)))
      BUF(114)=ORBELA(6)
      BUF(116)=ORBELA(5)
      BUF(117)=ORBELA(3)
      BUF(118)=ORBELA(4)
      BUF(120)=DTWOPI/PRD(1)
      BUF(121)=EC
      BUF(122)=PERDOT(1)*DRAD*CUT/8.64D4
      BUF(123)=RANDOT(1)*DRAD*CUT/8.64D4
      BUF(124)=PRD(1)/CUT
      BUF(125)=PERHT(1)*1.D3/CUL
      BUF(126)=APHT(1)*1.D+3/CUL
      CALL DATES(DSTART,IYMD,IHM,SEC)
      IYE=IYMD/10000
      IDE=IYMD-IYE*10000
      IMO=IDE/100
      IDE=IDE-IMO*100
      IH=IHM/100
      IM=IHM-IH*100
      BUF(151)=IYE
```

```
      BUF(192)=IME
      BUF(193)=IDE
      BUF(194)=IH
      BUF(195)=IM
      BUF(196)=IDINT(SEC*1000.D0+.5D0)
      BUF(200)=2.D0
C WRITE HEADER AND DATA RECORDS
 25   WRITE(TORB1)BUF
      DO 28 I=1,350
 28   BUF(I)=0.D0
C FILL RECORDS WITH ELEMENTS UNTIL FULL
   30 CALL DAYEAR(DORB1,IYMD,IDAY,ISEC)
      BUF(1)=IYMD
      BUF(2)=IDAY
      BUF(3)=ISEC
      BUF(4)=DT
      IBUF=1
 40   DO 50 I=1,6
 50   DBUF(I,IBUF)=XYZENC(I)*1.D-3
      IF(DORB1.GE.DORB1E) GO TO 60
      DAYO=DORB1
      DORB1=DORB1+ORB1RT
      CALL ORBIT(DORB1)
      IBUF=IBUF+1
      IF(IBUF.LE.50) GO TO 40
      GO TO 25
C WRITE LAST RECORD
 60   IF(IBUF.EQ.50) GO TO 80
      WRITE(TORB1)BUF
      IBUF=1
      DO 70 I=1,8
 70   BUF(I)=NINES
C TERMINATE TAPE WITH 9'S
 80   DO 90 I=1,6
 90   DBUF(I,IBUF)=NINES
      WRITE(TORB1)BUF
      WRITE(TORB1)(NINES,I=1,350)
      WRITE(TORB1)(NINES,I=1,350)
      ENDFILE TORB1
      NFILES=NFILES+1
      PRINT 100,NFILES,TORB1
      CALL ERROR(10,NINES)
      RETURN
  100 FORMAT(1H1,20X,'FOR THE ARC JUST COMPLETED AN ORB1 TAPE HAS BEEN'/
     .    23X,'SUCCESSFULLY WRITTEN ON FILE',I4,' OF UNIT',I3,1H./)
      END
```


ENTRY
ORBIT
OBTAIN ORBIT FOR START OF ORBI TAPE
SET UP HEADER RECORD
DRAG COEFFICIENTS
25
WRITE HEADER & DATA RECORDS
FILL RECORD WITH ELEMENTS UNTIL FULL
40
40
ADD ELEMENT SET TO RECORD
ORBI TAPE COMPLETE
YES
60
NO
ORBIT
OBTAIN NEW POSITION AND VELOCITY
RECORD FULL
NO
40
YES
25


60
RECORD FULL
NO
80
YES
WRITE RECORD
TERMINATE TAPE WITH 9' SENTINEL
80
FILL NEXT ELEMENT SET WITH 9's
WRITE RECORD
A
A
WRITE TWO SENTINEL RECORDS OF 9's
RETURN