N67-40277
X-554-66-283
NASA TMX 60549

PROJECT
GEMINI

GODDARD REAL TIME PROGRAMMING SYSTEM

VOLUME III
REAL TIME PROCESSORS

MARCH 1965

GODDARD SPACE FLIGHT CENTER
GREENBELT, MD.

FACILITY FORM 602

N67-40277 (ACCESSION NUMBER)	(THRU)
1335 (PAGES)	1 (CODE)
TMX-60549 (NASA CR OR TMX OR AD NUMBER)	08 (CATEGORY)

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
GODDARD SPACE FLIGHT CENTER
GREENBELT, MARYLAND

FOR OFFICIAL USE ONLY

THIS SERIES OF VOLUMES DESCRIBES IN DETAIL THE GODDARD REALTIME PROGRAMMING SYSTEM USED TO SUPPORT THE GEMINI, GT-3 AND GT-4 MANNED MISSIONS.

THE VOLUMES REPRESENT OUR FIRST ATTEMPT AT LARGE SCALE COMPUTERIZED DOCUMENTATION. IT HAS BEEN A LEARNING PROCESS AND WE FEEL THE TECHNIQUE HAS GREAT POSSIBILITIES. ALTHOUGH THE ANTICIPATED ECONOMIES WERE NOT REALIZED ON THIS FIRST ATTEMPT, WE FEEL WITH THE EXPERIENCE GAINED, FUTURE AUTOMATED DOCUMENTATION TASKS CAN BE ACCOMPLISHED MORE ECONOMICALLY THAN BY USING CONVENTIONAL TECHNIQUES.

FURTHER WE FEEL THAT COMPUTERIZED DOCUMENTATION PERMITS REALTIME DOCUMENTATION OF THE COMPUTER PROGRAMMING EFFORT INCLUDING FLOW CHARTING. THIS WAS DEMONSTRATED IN OUR LATEST AIMP EFFORT, AND IS A SOLUTION TO THE PROBLEM OF KEEPING DOCUMENTATION CURRENT WITH COMPUTER PROGRAM DEVELOPMENT.

PREPARED BY IBM FOR

DATA OPERATIONS BRANCH
MANNED FLIGHT OPERATIONS DIVISION
TRACKING AND DATA SYSTEMS DIRECTORATE
NASA-GODDARD

RELEASED BY J. J. Donegan
J. J. DONEGAN, HEAD
DATA OPERATIONS BRANCH

X-544-66-283

Project
GEMINI

GODDARD
REAL TIME PROGRAMMING
SYSTEM

Volume III
Real Time Processors

prepared for
National Aeronautics and Space Administration
Contract No. NAS 5-9762

March 1965

by
Federal Systems Division
International Business Machines Corporation

GODDARD REAL TIME PROGRAMMING SYSTEM

INDEX OF VOLUMES

I INTRODUCTION

II MONITOR AND I/O PROGRAMS

III REAL TIME PROCESSORS

IV NON-REAL TIME PROGRAMS

FOREWORD

PROJECT GEMINI WAS ESTABLISHED BY THE NATIONAL AERONAUTICS AND SPACE ADMINISTRATION (NASA) AS ITS SECOND MAN-IN-SPACE PROGRAM TO CARRY FORWARD THE MANNED SPACE MISSION STARTED BY PROJECT MERCURY AND TO PROVIDE THE KNOW-HOW NEEDED TO ENSURE A SUCCESSFUL MANNED LUNAR LANDING IN PROJECT APOLLO. IN PROJECT GEMINI, NASA WILL CONDUCT A SERIES OF SPACE FLIGHTS THAT WILL MEASURE THE PHYSIOLOGICAL ASPECTS OF EXPOSING MEN TO SPACE ENVIRONMENT FOR PROLONGED PERIODS, PROVIDE EXPERIENCE IN SPACE-CRAFT CONTROL, AND PERFECT RENDEZVOUS AND DOCKING TECHNIQUES WITH OTHER SPACE VEHICLES IN FLIGHT.

THE GODDARD SPACE FLIGHT CENTER (GSFC), LOCATED AT GREENBELT, MARYLAND, IS RESPONSIBLE FOR PROVIDING PRIME (REAL TIME) COMPUTING SUPPORT FOR THE GEMINI SPACE FLIGHTS THROUGH THE GT-3 (GEMINI TITAN-3) MISSION, AND TO PROVIDE BACKUP SUPPORT FOR THE MANNED SPACECRAFT CENTER AT HOUSTON, TEXAS FOR LATER MISSIONS.

THIS VOLUME IS ONE OF A SERIES OF FOUR MANUALS THAT DESCRIBE THE GSFC REAL TIME COMPUTING SYSTEMS AS DEVELOPED FOR MISSION SUPPORT THROUGH THE GT-3 MISSION. THE FOUR VOLUMES IN THIS SERIES ARE -

VOLUME I, INTRODUCTION
VOLUME II, MONITOR AND I/O PROGRAMS
VOLUME III, REAL TIME PROCESSORS
VOLUME IV, NON-REAL TIME PROGRAMS

THE PAGE IMPRESSIONS USED FOR PHOTO-OFFSET PRINTING OF THE TEXT AND FLOWCHARTS OF THIS SERIES OF MANUALS WERE PRODUCED BY AN IBM 1401 COMPUTER, WHICH WAS ALSO USED TO CONTROL THE PAGE AND LINE FORMAT. DUE TO CERTAIN LIMITATIONS IN THIS METHOD OF PRODUCTION, SOME FLOWCHARTS, TABLES AND ILLUSTRATIONS WERE PREPARED IN WHOLE OR IN PART, USING CONVENTIONAL TYPEWRITER/LINE ART METHODS.

CONTENTS

SEC	TITLE	PAGE
PART I	LAUNCH/ABORT	
1.	GENERAL CONTROL DURING LAUNCH/ORBIT	1
2.	PROGRAM SYSTEM OPERATION DURING NORMAL LAUNCH	2
3.	PROGRAM SYSTEM OPERATION DURING HOLD SUBPHASE	6
4.	PROGRAM SYSTEM OPERATION DURING LOW ABORT	8
5.	PROGRAM SYSTEM OPERATION DURING HIGH ABORT	10
6.	NPCCGB - MONITOR PREFIX TO COCOO1	12
7.	COCOO1 - LAUNCH COORDINATE CONVERSION OF B-GE DATA	15
8.	ROTATION OF B-GE COORDINATE SYSTEM TO GEMINI COORDINATE SYSTEM	18
9.	MFCCGB - MONITOR SUFFIX TO COCOO1	21
10.	MPCCIP - MONITOR PREFIX TO CSCOO2	24
11.	CSCOO2 - LAUNCH COORDINATE CONVERSION OF DATA	27
12.	ROTATION OF IP-7094 COORDINATE SYSTEM TO GEMINI COORDINATE SYSTEM	31
13.	MFCCIP - MONITOR SUFFIX TO CSCOO2	34
14.	MPEOBD - MONITOR PREFIX TO EOBDIT	37
15.	EOBDIT - BERMUDA HIGH-SPEED EDIT	48
16.	MFEBAD - MONITOR BAD POINTS SUFFIX TO EOBDIT	65
17.	MFEOBD - MONITOR SUFFIX TO EOBDIT	69
18.	MPBDAS - MONITOR PREFIX TO CSOGLS AND CASARC	73
19.	CASARC - SHORT ARC PROGRAM	83

CONTENTS

SEC	TITLE	PAGE
20.	CAMPSO - APPROXIMATE MIDPOINT SOLUTION SUBROUTINE	111
21.	CALTOI - LOCAL TO INERTIAL COORDINATE SUBROUTINE	116
22.	CANORM - SUBROUTINE TO FORM NORMAL EQUATIONS	121
23.	CASOLN - SUBROUTINE TO SOLVE NORMAL EQUATIONS	126
24.	CLSTOI - LOCAL TO INERTIAL COORDINATE CONVERSION	130
25.	CSOGLS - GENERAL LEAST SQUARES SMOOTHING	142
26.	CS1GLS - INITLIAIZATION SUBROUTINE FOR CSOGLS	165
27.	CS2GLS - RECURSIVE SUBROUTINE FOR CSOGLS	169
28.	CS3GLS - SUBROUTINE TO GENERATE SOLUTIONS FOR CSOGLS	173
29.	MFBDAS - MONITOR SUFFIX TO CASARC AND CSOGLS	177
30.	MPSTRP - MONITOR PREFIX TO CCSTRP	182
31.	CCSTRP - LAUNCH STRIP CHARTS PROCESSOR	184
32.	MFSTRP - MONITOR SUFFIX TO CCSTRP	194
33.	MPCCMN - MONITOR PREFIX TO CCMAIN	198
34.	CCMAIN - LAUNCH MAIN CALCULATIONS PROGRAM	220
35.	CFCFAV - LAUNCH COMPUTATIONS FOR HOLD PHASE	238
36.	C1DVGS - COMPUTE $\pm \Delta V_G$, AND $-\Delta V$	254
37.	CELEVP - ELEVATE PERIGEE	260

VOL III REAL TIME PROCESSORS

CONTENTS

SEC	TITLE	PAGE
38.	CEAPRV - COMPUTE THE VECTORS AT APOGEE OR PERIGEE	272
39.	CPRVCM - LAUNCH PRESENT VECTORS COMPUTATIONS	283
40.	CDELTV - LAUNCH PROGRAM TO INCREMENT THE VELOCITY VECTOR BY FIRING RETROROCKETS AS POSIGRADES	315
41.	CBIPRE - IMPACT PREDICTOR FOR AVERAGE VECTOR COMPUTATION	322
42.	C6RETC - LAUNCH RETROFIRE PROGRAM	329
43.	CTSFT3 - COMPUTATION OF TIME STEP FOR TYPE 3 INTEGRATION	392
44.	C6PTLU - LAUNCH PROGRAM FOR FOUR POINT LAGRANGIAN INTERPOLATION OF TIME	397
45.	C6PTLI - LAUNCH PROGRAM FOR SIX POINT INTERPOLATION OF POSITION AND VELOCITY VECTORS	404
46.	CRADES - LAUNCH PROGRAM TO SELECT RECOVERY AREA	412
47.	C6TDRF - RETROFIRE TIME COORECTION	418
48.	CGFTCI - COMPUTATION OF IMPACT, GIVEN THE TIME TO FIRE RETROROCKETS	423
49.	CVAFPA - COMPUTATION OF RELATIVE VELOCITY AND RELATIVE FLIGHT PATH ANGLE AT A SPECIFIED HEIGHT	427
50.	A3GSCP - COMPUTATION OF LATITUDE, LONGITUDE AND ALTITUDE	433
51.	BOSCER - EARTH RADIUS DETERMINATION	444

CONTENTS

SEC	TITLE	PAGE
52.	MFCCMN - SUFFIX TO CCMAIN	448
53.	CORKNI - RUNGE-KUTTA NUMERICAL INTEGRATION	467
54.	C6HPEP - COMPUTATION OF GEOCENTRIC HEIGHT AND ACCELERATION DUE TO THE POTENTIAL OF THE EARTH	493
55.	C4DRFT - COMPUTATION OF ACCELERATION DUE TO DRAG, LIFT, AND THRUST	498
56.	C4PDEN - COMPUTATION OF ATMOSPHERIC DENSITY	510
57.	C4COEF - COMPUTATION OF COEFFICIENTS OF DRAG AND LIFT	515
58.	C3GSCP - COMPUTATION OF LATITUDE AND LONGITUDE FOR NUMERICAL INTEGRATION	520
59.	MPMPRK - MONITOR PREFIX TO LIMPRK	523
60.	LIMPRK - CONTROL PROGRAM FOR CIMPRK	525
61.	CIMPRK - IMPACT PREDICTOR FOR NORMAL LAUNCH	528
62.	MFMPRK - MONITOR SUFFIX TO LIMPRK	535
63.	CUPDAT - LAUNCH PROGRAM FOR UPDATING A GIVEN	538
64.	MABORT - MONITOR CONTROL FOR ABORT PHASE	542
65.	CLABRT - LOW ABORT PROCESSOR	572
66.	CHABRT - LAUNCH PROCESSOR FOR HIGH ABORT	583
67.	C2TAFH - COMPUTATION OF TIME AS A FUNCTION OF HEIGHT	607
68.	CABDAT - LAUNCH ABORT PHASE PROGRAM FOR UPDATING A GIVEN	612

CONTENTS

SEC	TITLE	PAGE
PART II	ORBIT/REENTRY	
1.	BASIC COMPUTING CYCLE DURING ORBIT/REENTRY	621
2.	MSVTCA - GENERATION OF TMVTCA TABLE	631
3.	MSLCUC - MONITOR SUBROUTINE TO MOVE A RADAR MESSAGE BLOCK TO THE 32K CORE BUFFER MODULE	635
4.	MSWTMS - MONITOR SUBROUTINE TO COMPUTE A TMDATA BLOCK ADDRESS	641
5.	MSDELT - MONITOR SUBROUTINE TO DELETE AN ENTRY IN THE TMSTMS TABLE	644
6.	MSTINT - SUBROUTINE TO REMOVE RADAR OBSERVATIONS FROM DIFFERENTIAL CORRECTION	648
7.	MSINSR - SUBROUTINE TO INSERT RADAR OBSERVATIONS INTO DIFFERENTIAL CORRECTION	654
8.	MYOUR1 - ORBIT BASIC CYCLE CONTROL PROCESSOR	657
9.	GVMMRF - MANEUVER REFLECTING VECTORS	670
10.	MYREN2 - REENTRY CONTROL PROGRAM	676
11.	RUPDAT - FOR UPDATING THE POST RETROFIRE TRAJECTORY	712
12.	RUCIVI - PLATFORM TO INERTIAL COORDINATES CONVERSION	719
13.	MYAPE3 - APOGEE-PERIGEE CONTROL PROCESSOR	725
14.	MYTTF4 - TIME-TO-FIRE CONTROL PROCESSOR	744
15.	MYNAV5 - ORBIT NAVIGATION VECTOR FOR DCS	762
16.	MYGMT6 - CALCULATE GMT	767

CONTENTS

SEC	TITLE	PAGE
17.	MYMUV7 - MANEUVER PARAMETER COMPUTATION CONTROL PROCESSOR	722
18.	MYLIF8 - REVOLUTION CAPABILITY MONITOR CONTROL PROCESSOR	777
19.	MYDIC9 - DIFFERENTIAL CORRECTION MONITOR	781
20.	MYQPRA - MONITOR QUEUE NEXT PRIMARY AREA PROCESSOR	784
21.	MYQNEA - MONITOR QUEUE NEXT EMERGENCY AREA PROCESSOR	789
22.	MPLED1 - MONITOR PREFIX TO EOLED1	793
23.	MPLED2 - MONITOR PREFIX TO EOLED1	797
24.	EOLED1 - EDIT	800
25.	MFLED1 - MONITOR SUFFIX TO EOLED1	806
26.	MPCPNI - MONITOR PREFIX TO CONTROL PROGRAM NUMERICAL INTEGRATION	817
28.	ROCKNR - THRUST PROGRAM	860
29.	MFCPNI - MONITOR SUFFIX TO NUMERICAL	875
30.	N1SUFT - SET UP FUNCTION TABLE	879
31.	N2EXCR - EXTRAPOLATION AND CORRECTION EQUATION DERIVATION	884
32.	N3SDCL - SECOND DERIVATIVE CALCULATION CONTROL	891
33.	N3TRB - COMPUTATION OF INSTANTANEOUS BANK ANGLE FOR TIME TO REVERSE BANK	895
34.	N4DRAG - CAPSULE DRAG FORMULATION	898
35.	N4PDEN - COMPUTATION OF ATMOSPHERIC DENSITY	912
36.	N4COEF - COEFFICIENT OF DRAG CALCULATION	919

CONTENTS

SEC	TITLE	PAGE
37.	N5OTBI - NUMERICAL INTEGRATION OUTPUT TABLE INITIALIZATION	925
38.	N5OTBL - NUMERICAL INTEGRATION OUTPUT TABLE STORING	929
39.	N5CAPI - NUMERICAL INTEGRATION CAPE VECTOR INITIALIZATION	935
40.	N5CAPS - NUMERICAL INTEGRATION CAPE VECTOR STORING	939
41.	N6OPEP - EARTH'S EFFECT ON SPACECRAFT CALCULATION	946
42.	N7VARS - VARIABLE STEP INTEGRATION EQUATIONS	950
43.	N7PCAL - ON-BOARD COMPUTER EQUATIONS	957
44.	MPDIFC - MONITOR PREFIX TO DODIFC	965
45.	MPDIFK - MONITOR PREFIX TO DODIFC	973
46.	MPDIFM - MONITOR PREFIX TO DODIFC	976
47.	DIFFERENTIAL CORRECTION METHOD	983
48.	DODIFC - DIFFERENTIAL CORRECTION CONTROL PROGRAM	986
49.	MFDIFC - MONITOR SUFFIX TO DODIFC	1011
51.	D2CSFE - CALCULATE STANDARD ERROR FIT	1026
52.	D2SLNE - SOLVE NORMAL EQUATIONS	1029
53.	D2SSMT - STATION SUM OF SQUARES OVER M	1034
54.	D2SUNE - SET UP NORMAL EQUATIONS	1037
55.	D2WEQT - WEIGHTING THE EQUATIONS OF CONDITION	1045
56.	D3CDRM - CHOOSE DESIRED RADAR MESSAGE	1052
57.	D3PTWT - WEIGHT TABLE LOOKUP	1059

CONTENTS

SEC	TITLE	PAGE
58.	D4CFRV - CALCULATE FUNCTIONS OF $\bar{r}$ AND $\bar{v}$	1062
59.	R1NDCP - NUMERICAL DIFFERENTIAL CORRECTION	1066
60.	R2AINT - NUMERICAL INTEGRATION INPUT TABLES TO BE USED FOR GENERATION OF PERTURBED TABLES FOR NUMERICAL PARTIAL DERIVATIVES	1073
61.	R5RARF - TIME TO FIRE PROGRAM	1081
62.	AQDATA - SUBROUTINE TO GENERATE ACQUISITION DATA	1125
63.	MPORMC - MONITOR PREFIX TO O5ORMC	1143
64.	O5ORMC - ORBIT, REENTRY HIGH-SPEED OUTPUT COMPUTATION PROCESSOR	1146
65.	MFORMC - MONITOR SUFFIX TO O5ORMC	1158
66.	RVECHT - COMPUTE HEIGHT VECTOR	1160
67.	R6ESTF - ESTIMATED TIME-TO-FIRE PROCESSOR	1164
68.	R7EOAM - FIRST GUESS FOR TIME TO FIRE RETROS	1169
69.	MPBUDC - MONITOR PREFIX TO BUDCCP AND BUOBCP	1179
70.	BUDCCP - REENTRY PHASE BACK UP TO DIFFERENTIAL CORRECTICN PROGRAM	1182
71.	BUOBCP - ORBIT PHASE BACK UP TO DIFFEREN-TIAL CORRECTION PROGRAM	1189
72.	BULTOI - LOCAL TO INERTIAL CONVERSION PROGRAM	1196
73.	BUMAIN - COMPUTATION OF POSITION AND VELOCITY VECTORS AND ASSOCIATED INFORMATION	1202

CONTENTS

SEC	TITLE	PAGE
74.	BUCFIT - SECOND DEGREE CURVE FITTING PROGRAM FOR BUMAIN	1213
75.	BUEDIT - EDITING PROGRAM FOR BUMAIN	1217
76.	BUGSCP - SPACECRAFT POSITION AND HEIGHT DETERMINATION PROGRAM FOR BUMAIN	1222
77.	BUFPPR - FOOTPRINT DESCRIPTION	1226
78.	BUKEPE - PROGRAM TO COMPUTE SIX CLASSICAL ELEMENTS	1234
79.	MFBUDC - MONITOR SUFFIX TO BUDCCP AND BUOBCP	1241
80.	MYFPPR - MONITOR FOOTPRINT PREDICTOR	1244
81.	RHBRVG - PROGRAM TO COMPUTE THE RELATIVE VELOCITY VECTOR MAGNITUDE AND RELATIVE FLIGHT PATH ANGLE	1246
82.	GINTAB - SUBROUTINE TO DETERMINE INTEGRATION TABLE	1251
83.	GEMWGT - ORBIT WEIGHT SUBROUTINE	1255
84.	GLINTS - VECTOR LINEAR INTERPOLATION SUBROUTINE	1258
85.	MYTABU - MANEUVER TABLE CONTROL PROCESSOR	1262
86.	VOMANV - PROPULSIONS PANEL COMPUTATIONS	1271
87.	A3MSCP - SUBROUTINE TO COMPUTE LATITUDE AND LONGITUDE	1285
88.	N80TST - NOCPNI END TEST	1293
89.	N9IRRB - INITIATE ROLL, REVERSE BANK TEST	1299
90.	A3TSUP - A3MSCP INPUT TIME SET UP FOR NOCPNI	1304

CONTENTS

SEC	TITLE	PAGE
91.	MYCOTO - CONFIRMATION OF MANEUVER CONTROL PROCESSOR	1306
INDEX OF PROGRAMS		1311

PART I

LAUNCH/ABORT

VOL III REAL TIME PROCESSORS PART I - LAUNCH/ABORT

1. GENERAL CONTROL DURING LAUNCH/ABORT

THE LAUNCH PHASE INCLUDES TWO SUBPHASES, NORMAL LAUNCH AND HOLD. THE NORMAL LAUNCH SUBPHASE EXTENDS FROM LIFTOFF UNTIL ABOUT 15 SECONDS AFTER SECOND STAGE CUT-OFF. THE HOLD SUBPHASE EXTENDS FROM THE END OF THE NORMAL LAUNCH SUBPHASE UNTIL THE FLIGHT DYNAMICS OFFICER AT THE MISSION CONTROL CENTER (MCC) AT CAPE KENNEDY PLACES A SWITCH INTO EITHER THE ORBIT OR THE ABORT POSITION.

THE ABORT PHASE, IF USED DURING A MISSION, WILL BE ONE OF TWO TYPES, LOW ABORT OR HIGH ABORT. IF THE FLIGHT DYNAMICS OFFICER REQUESTS THAT THE ABORT PHASE BE ENTERED PRIOR TO ABORT MODE CHANGEOVER (AMC), THE LOW ABORT PROGRAMS WOULD BE USED. OTHERWISE, THE HIGH ABORT PROGRAMS WOULD BE USED. DURING LOW ABORT, EITHER (1) THE ASTRONAUTS EJECT THEMSELVES FROM THE SPACECRAFT, AND THE BOOSTER/SPACECRAFT CONFIGURATION REENTERS STILL JOINED TOGETHER, OR (2) THE RETRO-ROCKETS ARE USED IN A POSIGRADE SENSE TO SEPARATE THE SPACECRAFT FROM THE BOOSTER. THEREFORE, DURING LOW ABORT THERE IS NO INPACT POSITION CONTROL DUE TO FIRING THE RETRO-ROCKETS AT A COMPUTED TIME. DURING HIGH ABORT, HOWEVER, THE ORBIT ATTITUDE MANEUVERING SYSTEM (OAMS) IS USED TO SEPARATE THE SPACECRAFT FROM THE BOOSTER AND IMPACT CONTROL BY FIRING THE RETROS AT A CALCULATED TIME IF FEASIBLE. THIS IS THE PRIMARY DISTINCTION BETWEEN LOW AND HIGH ABORT.

2. PROGRAM SYSTEM OPERATION DURING NORMAL LAUNCH

HIGH-SPEED TRACKING DATA FROM THE BURROUGHS-GENERAL ELECTRIC (B-GE) GUIDANCE SYSTEM COMPUTER AND THE INPACT PREDICTOR 3600 (IP 3600) COMPUTER AT CAPE KENNEDY IS RECEIVED AND PROCESSED AT GSFC. HIGH-SPEED DATA FROM THE BERMUDA RADARS IS ALSO PROCESSED AFTER SECOND-STAGE CUTOFF.

RESULTS ARE TRANSMITTED EVERY ONE-HALF SECOND TO MCC FOR EVALUATION. THE GSFC COMPUTERS PROVIDE MCC WITH CRITICAL INFORMATION, SUCH AS SPACECRAFT ALTITUDE, VELOCITY RATIO, FLIGHT PATH ANGLE, AND PREDICTED IMPACT LATITUDE AND LONGITUDE FOR IMMEDIATE ABORT CONDITIONS.

AT THE USUAL TERMINATION OF THE NORMAL LAUNCH SUBPHASE, THE GSFC COMPUTERS GIVE GO NO-GO RECOMMENDATIONS FOR FLIGHT CONTINUATION BASED ON DATA FROM THE VARIOUS SOURCES. RECOMMENDED TIME OF RETROFIRE AND OTHER PERTINENT SPACECRAFT PARAMETERS ARE ALSO SENT TO MCC AT THIS TIME.

THE HOLD PHASE IS THEN ENTERED. HOWEVER, IT IS POSSIBLE UNDER CERTAIN CONDITIONS FOR THE MISSION TO GO FROM THE NORMAL LAUNCH SUBPHASE DIRECTLY TO THE ABORT PHASE.

NORMAL LAUNCH PROGRAM ORGANIZATION

PROCESSING INITIATED BY INCOMING INPUT TRAPS

EACH HALF SECOND HIGH-SPEED DATA IS RECEIVED FROM THE B-GE COMPUTER AT CAPE KENNEDY. EACH FOUR-TENTHS OF A SECOND HIGH-SPEED DATA IS RECEIVED FROM THE IP 3600 COM-

PUTER AT THE CAPE. WHEN DATA FROM EITHER SOURCE ARRIVES IT IS -

A) CONVERTED TO A USABLE FORMAT BY AN INPUT PROGRAM (IOHSGB OR IOHS09).

B) CONVERTED TO THE GEMINI COORDINATE SYSTEM (COCOOI OR CSCOO2).

C) USED TO COMPUTE DEVIATIONS FROM NOMINAL VALUES OF v/v_r AND γ FOR THE STRIP CHARTS DATA QUALITY MONITOR (CCSTRP).

THE INPUT PROGRAMS ALSO UTILIZE THE INCOMING TELEMETRY INFORMATION TO SET CERTAIN COMMUNICATIONS CELLS THAT KEEP COGNIZANT PROGRAMS AWARE OF CERTAIN CONDITIONS IN THE OUTSIDE WORLD.

THE SUFFIX TO CCSTRP ALSO SAVES A CERTAIN NUMBER OF THE MOST RECENT B-GE AND IP 3600 VECTORS RECEIVED FOR LATER PROCESSING.

AT A CERTAIN TIME AFTER SECOND STAGE CUTOFF THE HIGH-SPEED DATA LINES FROM BERMUDA ARE ACTIVATED. EACH TENTH OF A SECOND A TRAP OCCURS ANNOUNCING THE ARRIVAL OF r,a,e, DATA FROM THE BERMUDA AN/FPS-16 AND VERLORT RADARS. THIS DATA IS CONVERTED TO A USABLE FORMAT BY THE BERMUDA INPUT PROGRAM IOHSBD AND STORED IN BUFFER TABLES TO AWAIT LATER PROCESSING.

PROCESSING INITIATED BY HALF-SECOND TRAPS

A MONITOR PROCESSOR CAUSES A TRAP EACH HALF SECOND WHICH TRIGGERS VARIOUS PROCESSING WHICH IS DONE ON A TIMED FREQUENCY.

A) MAIN NORMAL LAUNCH PROCESSING (CCMAIN) - THIS PROCESSOR COMPUTES THE BULK OF THE QUANTITIES TO BE OUTPUTTED TO MCC. IS PREFIX DETERMINES WHICH DATA SOURCE THE FLIGHT DYNAMICS OFFICER REQUESTS BE USED AS INPUT IN MAKING THE COMPUTATIONS, AND PRESENTS CCMAIN WITH THE APPROPRIATE INPUT, IF AVAILABLE. VARIOUS OUTPUT QUANTITIES ARE COMPUTED FOR DISPLAY. DATA WHICH REFLECTS THE ORBITAL CONDITIONS WHICH WOULD BE OBTAINED IF AN ABORT WERE TO BE INITIATED IMMEDIATELY ARE CALCULATED FOR POSSIBLE USE BY THE INPACT PREDICTOR CIMPRK AND THE LOW ABORT PROCESSORS.

B) IMPACT PREDICTOR (CIMPRK) - THIS PROCESSOR IS ENTERED ONCE EACH SECOND. IT COMPUTES THE MINIMUM AND MAXIMUM DELAY IMPACT POINTS TO BE SENT TO MCC FOR DISPLAY.

C) BERMUDA DATA PROCESSING (AFTER SSCO ONLY) - EACH SECOND THE r,a,e's, COMING FROM BERMUDA ARE EDITED (EOBDIT). THIS EDITED DATA IS USED TO GENERATE POSITION AND VELOCITY VECTORS USING THE SHORT ARC PROCESS. THESE SOLUTIONS ARE GENERATED EACH SECOND AND ARE ANALAGOUS TO THE OUTPUT OF COCOO1 OR CSCOO2, I.E., THEY ARE t, $\vec{r}$, $\vec{v}$'s IN GEMINI UNITS.

D) OUTPUT DISPLAY PROGRAM (OOLANA) - THIS PROCESSOR IS ENTERED EACH HALF SECOND TO REFORMAT ALL OUTPUT FOR HIGH SPEED TRANSMISSION TO MCC.

PROCESSING CARRIED OUT AT THE TERMINATION OF NORMAL LAUNCH (IF AMC HAS OCCURRED) -

THE VECTORS RECEIVED FROM THE B-GE AND IP 3600 COMPUTERS DURING A 10 SECOND INTERVAL BEGINNING AT ABOUT FIVE SECONDS AFTER SECOND STAGE CUTOFF ARE AVERAGED FOR EACH SOURCE TO OBTAIN AN AVERAGE t, $\bar{r}$, $\bar{v}$, REPRESENTING THE COMPUTERS MOST RELIABLE DEFINITION OF THE ORBIT OF THE BOOSTER. FROM THESE VECTORS THE GO NO-GO RECOMMENDATION, RECOMMENDED RETRO FIRING TIMES AND OTHER IMPORTANT QUANTITIES ARE COMPUTED AND SENT TO MCC FOR EVALUATION BY THE FLIGHT DYNAMICS OFFICER. AT THIS POINT THE MISSION ENTERS THE HOLD PHASE.

3. PROGRAM SYSTEM OPERATION DURING HOLD SUBPHASE

AT THIS TIME THE FLIGHT DYNAMICS OFFICER AT MCC QUICKLY BUT CAREFULLY EVALUATES THE VARIOUS PARAMETERS DISPLAYED AND DECIDES TO EITHER ABORT THE MISSION OR ALLOW IT TO CONTINUE. IF HE DECIDES TO ABORT, HE MUST DO SO QUICKLY IF HE HOPES TO HAVE THE SPACECRAFT LAND IN THE ATLANTIC OCEAN RATHER THAN IN AFRICA.

THE DISPLAYS PRESENTED TO THE FLIGHT DYNAMICS OFFICER ARE HELD CONSTANT UNTIL HE HAS HAD TIME TO EVALUATE THEM THOROUGHLY, HENCE, THE TERM 'HOLD PHASE.' HE HAS THE OPTION TO UPDATE THE DISPLAYS BASED ON DATA COMING FROM RADARS TRACKING THE SPACECRAFT (OR BY EXTRAPOLATING FROM THE LAST GOOD DATA RECEIVED) WHEN HE SO DESIRES. ONCE HE HAS MADE HIS DECISION, HE CAUSES THE GSFC COMPUTERS TO ENTER EITHER THE ABORT PHASE OR THE ORBIT PHASE BY PLACING A SWITCH IN THE APPROPRIATE POSITION.

PROGRAM ORGANIZATION DURING HOLD PHASE

AT THE TERMINATION OF NORMAL LAUNCH THE BULK OF THE QUANTITIES BEING DISPLAYED ARE BASED ON THE DATA RECEIVED FROM THE SOURCE THAT WAS SELECTED DURING THE DATA-COLLECTION PERIOD. THE GO/NO-GO RECOMMENDATION AND THE RETROFIRE TIMES FROM THE NONSELECTED SOURCE ARE DISPLAYED. AT THIS POINT, THE FLIGHT DYNAMICS OFFICER MAY CHANGE THE SELECTED SOURCE BACK AND FORTH BETWEEN B-GE AND IP 3600 IF HE SO DESIRES. OTHERWISE, THE DISPLAYS REMAIN STATIC DURING THE HOLD PHASE UNTIL THE BERMUDA SELECTION SWITCH IS THROWN. INTERNALLY, HOWEVER, THE

BERMUDA PROCESSING PROGRAMS DESCRIBED UNDER NORMAL LAUNCH (SECTION 1) ARE CONTINUALLY PRODUCING SOLUTIONS SO THAT A CURRENT SOLUTION WILL BE AVAILABLE WHEN NEEDED. ALSO, INCOMING VECTORS FROM THE IP LINES ARE STILL BEING SAVED FOR POSSIBLE USE.

DURING THE HOLD PHASE THE COMPUTATIONS ARE FIVE-SECOND CYCLE. THE LATEST BERMUDA VECTORS AND AVERAGED IP-SPACECRAFT VECTORS ARE UPDATED TO CURRENT TIME EACH FIVE SECONDS. IN THE EVENT OF MISSING DATA THE LAST VECTOR IS AGAIN BROUGHT UP TO CURRENT TIME. IN THIS UPDATING PROCESS THE SETTING OF THE THRUST SWITCH AT THE TIME OF THE UPDATING DETERMINES WHETHER OR NOT THE OAMS SYSTEM IS CONSIDERED TO BE BURNING DURING THE UPDATING.

IF THE BERMUDA SELECTION SWITCH IS THROWN, THE LATEST UPDATED BERMUDA VECTOR IS USED AS INPUT FOR THE HOLD PHASE COMPUTATIONS. THEREAFTER, THE LATEST UPDATED IP VECTOR WILL BE USED AS INPUT TO THE HOLD PHASE COMPUTATIONS WHENEVER IP IS SELECTED. SELECTING B-GE, HOWEVER, WILL CAUSE THE ORIGINAL B-GE OUTPUT DISPLAYS TO BE SHOWN (I.E., THOSE THAT WERE SHOWN AT THE BEGINNING OF THE HOLD PHASE.)

EACH TIME THE AVERAGE VECTOR PROCESSING IS DONE FOR DATA FROM A GIVEN SOURCE, THE VELOCITY VECTOR IS MODIFIED AS NECESSARY TO ACCOUNT FOR NOMINAL SEPARATION. THIS VECTOR IS UPDATED TO THE NEXT WHOLE MINUTE. THE RESULTANT UPDATED VECTOR IS ALWAYS READY TO BE TURNED OVER TO EITHER THE HIGH ABORT OR ORBIT PHASE WHEN ONE OF THESE PHASES IS REQUESTED.

4. PROGRAM SYSTEM OPERATION DURING LOW ABORT

IF THE FLIGHT DYNAMICS OFFICER ABORTS THE MISSION PRIOR TO ABORT MODE CHANGEOVER (AMC) BY THROWING THE ABORT/ORBIT SWITCH TO THE ABORT POSITION THE LOW ABORT PHASE IS ENTERED. IN THIS EVENT THERE WOULD BE NO HOLD SUBPHASE. THE LAST VECTOR FROM NORMAL LAUNCH (WHICH WAS MODIFIED BY CCMAIN TO REFLECT THE SPACECRAFT ORBIT) BECOMES 'THE ABORT PHASE DEFINING VECTOR.'

THE BERMUDA HIGH-SPEED INPUT LINES ARE ACTIVATED, AND THE BERMUDA INPUT PROGRAM ICHSBD IS USED TO CONVERT r,a,e's FROM THE BERMUDA RADARS TO A USUABLE FORMAT AS TRAPS OCCUR AT .1 SECOND INTERVALS. THESE ARE EDITED BY EOBDIT AT A ONE-SECOND CYCLE. THE r,a,e's ARE CONVERTED TO INERTIAL X,Y,Z'S AND A 51-POINT SMOOTHING PROGRAM IS ENTERED ON A ONE-SECOND CYCLE TO CALCULATE THE NECESSARY t, $\overline{r}$, $\overline{v}$'s.

B-GE DATA, WHICH REPRESENTS THE BOOSTER, IS AN INVALID SOURCE DURING LOW ABORT, IP DATA WHICH REPRESENTS BOOSTER TRACK IS ALSO INVALID.

IP SPACECRAFT DATA RECEIVED DURING THE PREVIOUS FIVE SECONDS IS AVERAGED TO DETERMINE RELIABLE IP VECTORS.

DURING LOW ABORT A SIX-SECOND COMPUTING CYCLE IS USED. EACH SIX SECONDS THE LATEST VALID VECTOR FROM EACH SOURCE IS UPDATED (TO CURRENT TIME) USING NO LIFT, PROGRAMMED LIFT, AND MAXIMUM LIFT. IF NO VECTOR WAS EVER RECEIVED FROM A SOURCE THE ABORT PHASE DEFINING VECTOR IS USED.

THE VECTOR FROM THE SELECTED SOURCE THAT WAS UPDATED WITH THE CURRENT LIFT SWITCH SETTING IS USED AS INPUT TO THE LOW ABORT PROCESSING.

VARIOUS OUTPUTS ARE CALCUATED AND SENT TO THE MCC ON A HALF-SECOND OUTPUT CYCLE.

ONCE THE LATEST TRAJECTORY INFORMATION INDICATES THAT THE SPACECRAFT IS BELOW 10,000 FEET, THE OUTPUTS CEASE AND THE SYSTEM WAITS FOR A MANUAL INSERTION MESSAGE, TELLING IT TO ENTER THE REENTRY PHASE.

5. PROGRAM SYSTEM OPERATION DURING HIGH ABORT

WHEN THE FLIGHT DYNAMICS OFFICER DIRECTS THAT THE ABORT PHASE BE ENTERED FROM THE HOLD PHASE, HE SPECIFIES WHICH DATA SOURCE IS TO BE USED (B-GE, IP 3600, OR BERMUDA) TO DETERMINE THE TRAJECTORY. NO FURTHER RADAR IS USED DURING THE ABORT PHASE.

UPON ENTRY TO THE HIGH ABORT PHASE, THE AVERAGE VECTORS FROM THE SELECTED SOURCE ARE USED TO GENERATE AN ORBIT TABLE AND THE COMPUTER'S RECOMMENDED TIME TO RETRO-ROCKETS IS USED TO GENERATE A REENTRY TABLE. THESE TABLES ARE USED TO PREDICT POSITION INFORMATION FOR THE RANGE STATIONS. TTY MESSAGES OF PREDICTED POINTING DATA ARE SENT TO THE RANGE STATIONS TO ENABLE ACQUISITION AND RADARS AT THE OBSERVING SITES TO ACQUIRE THE SPACECRAFT SHORTLY AFTER IT APPEARS OVER THE LOCAL HORIZON.

VARIOUS OTHER OUTPUTS ARE ALSO COMPUTED FOR DISPLAY FROM THESE TABLES. THE BASIC COMPUTING CYCLE DURING THE ABORT PHASE IS SIX SECONDS. EACH SIX SECONDS THOSE QUANTITIES WHICH ARE A FUNCTION OF TIME ARE UPDATED (POSITION, VELOCITY, FLIGHT PATH ANGLE, ETC).

THIRTY SECONDS AFTER RECEIPT OF TELEMETRY INFORMATION INDICATING THAT THE FIRST RETRO HAS FIRED, THE REENTRY TABLE AND ASSOCIATED QUANTITIES ARE RECOMPUTED TO REFLECT THE TELEMETRY INDICATED CONDITIONS OF RETROFIRE. NOMINAL ATTITUDES DURING THE FIRING ARE ASSUMED. THIS CALCULATION IS REPEATED IF CHANGES OCCUR IN TELEMETRY INFORMATION CONCERNING RETROFIRING TIME. EACH TIME A REENTRY TABLE IS GENERATED, NEW ACQUISITION INFORMATION IS COMPUTED AND SENT TO THE RANGE STATIONS.

ONCE THE FLIGHT DYNAMICS OFFICER IS SATISFIED THAT

HE KNOWS THE RETROFIRE CONDITIONS AS WELL AS POSSIBLE, HE CAUSES THE GSFC COMPUTERS TO ENTER THE ENTRY PHASE BY DIRECTING THAT AN APPROPRIATE MANUAL INSERTION BE MADE. UPON RECEIPT OF THIS MESSAGE, THE HIGH ABORT PROGRAM CALCULATES THE VECTORS AT THE BEGINNING OF RETROFIRE WHICH ARE TURNED OVER TO THE REENTRY PHASE.

6. MPCCGB - MONITOR PREFIX TO COCOO1

MPCCGB SETS UP THE INPUT FOR THE CURRENT ENTRY TO THE B/GE COORDINATE CONVERSION PROCESSOR, COCOO1. (SEE FLOWCHART)

INPUT

A) COMMUNICATION CELL -

MCLFTM	GMT OF LIFT OFF IN FLOATING POINT SECONDS.

B) TABLES -

TMH1TM	A 2-LOCATION TABLE CONTAINING THE TELEMETRY TRANSMISSION TIME IN THE FIRST WORD AND THE DISCRETE WORD INDICATORS IN THE SECOND WORD.
TMH1DB	A 7-LOCATION TABLE CONTAINING TIME IN GMT OR ELAPSED SECONDS AND $\overline{r}$ AND $\overline{v}$ IN FLOATING POINT B/GE UNITS.

OUTPUT

A) COMMUNICATION CELL -

MCDESC	DISCRETE INDICATORS

B) TABLE -

TMRV1G	A 7-LOCATION TABLE CONTAINING THE TIME IN GMT SECONDS AND $\overline{r}$ AND $\overline{v}$ IN FLOATING POINT B/GE UNITS.

METHOD

MPCCGB TURNS ON THE A INDICATOR AND TURNS OFF THE B INDICATOR FOR THE B/GE COORDINATE CONVERSION PROCESSOR. IT THEN SUPPRESSES ENTRANCE TO THE B/GE INPUT PROCESSOR, IOHSGB.

THE DISCRETE INDICATORS ARE MOVED INTO A COMMUNICATION CELL, MCDESC.

THE VECTOR, SOURCE ID AND DATA QUALITY INDICATOR ARE THEN MOVED FROM THE OUTPUT BUFFER FROM IOHSGB TO THE INPUT BUFFER FOR COCO01.

SENSE SWITCH 6 IS INTERROGATED TO DETEMINE WHETHER THE TIME OF LIFT OFF SHOULD BE ADDED TO THE VECTOR TIME.

IOHSGB IS UNSUPPRESSED AND CONTROL IS TRANSFERRED TO COCO01.

USAGE

MPCCGB IS ENTERED FROM MOPRIO WHEN PRIORITY CONDITIONS PERMIT. IT TRANSFERS CONTROL TO THE B/GE COORDINATE CONVERSION PROCESSOR, COCO01

A) PARAMETER -

A

B

C

MNCCGB

MNHSGB

B) MACROS -

TRNON

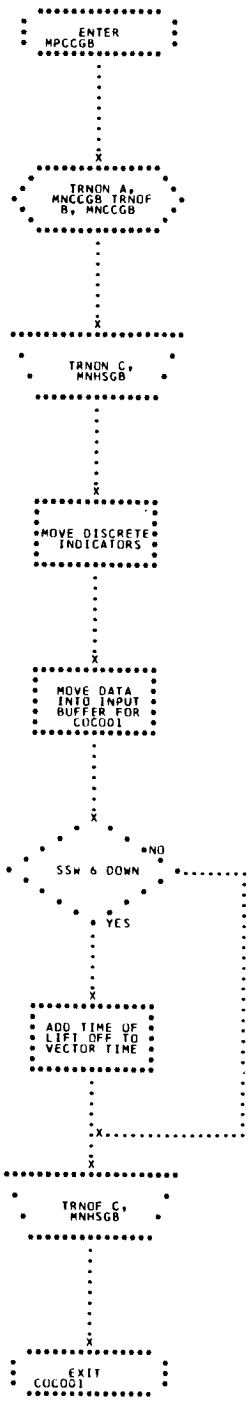
TRNOF

C) STORAGE REQUIRED -

24 LOCATIONS

D) TIMING (7094) -

.106 MILLISECONDS


ENTER MPCCGB
TRNON A, MNCCGB TRNOF B, MNCCGB
TRNON C, MNHSGB
MOVE DISCRETE INDICATORS
MOVE DATA INTO INPUT BUFFER FOR COCOOI
SSW 6 DOWN
NO
YES
ADD TIME OF LIFT OFF TO VECTOR TIME
TRNOF C, MNHSGB
EXIT COCOOI

7. COCO01 - LAUNCH COORDINATE CONVERSION OF B-GE DATA

THIS PROGRAM ROTATES THE COORDINATES OF THE INPUT VECTORS FROM THE B-GE COORDINATE SYSTEM TO THE GEMINI COORDINATE SYSTEM. THE RESULTING VECTORS ARE THEN CONVERTED TO GEMINI LENGTH AND VELOCITY UNITS. (SEE FLOWCHART)

INPUT

A) CONSTANTS -

K.OMES	ROTATION OF EARTH IN FLOATING POINT RADIANS/SECONDS, ω_e
KLAMDO	INERTIAL LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH IN FLOATING POINT RADIANS, λ_0
KLONGE	LONGITUDE OF B-GE RADAR IN FLOATING POINT RADIANS, λ_p
KRTOGE	CONVERSION OF B-GE LENGTH UNITS TO GEMINI LENGTH UNITS, K_{RG}
KVTOGE	CONVERSION OF B-GE VELOCITY UNITS TO GEMINI VELOCITY UNITS, K_{VG}

B) TABLES -

TMRV1G — A NINE-LOCATION TABLE CONTAINING IN THE FIRST SEVEN LOCATIONS THE INPUT VECTORS AND THEIR ASSOCIATED TIME

TMRV1G	t_i -TIME OF VECTOR (FLOATING POINT GMT SECONDS)
+1 - +3	$\bar{R}_{1G}$ -POSITION VECTOR (FLOATING POINT B-GE LENGTH UNITS)

+4 - +6 $\bar{V}_{1G}$ -VELOCITY VECTOR (FLOATING POINT B-GE VELOCITY UNITS)

C) LIBRARY SUBROUTINE U1SICO IS ALSO USED.

OUTPUT

A) TABLES -

TMRV3G AN 11-LOCATION TABLE CONTAINING IN THE FIRST SEVEN LOCATIONS THE CONVERTED VECTORS AND THEIR ASSOCIATED TIME.

TMRV3G t_i - (SAME AS IN TMRV1G)

+1 - +3 $\bar{R}_3$ -POSITION VECTOR (FLOATING POINT GEMINI LENGTH UNITS)

+4 - +6 $\bar{V}_3$ -VELOCITY VECTOR (FLOATING POINT GEMINI VELOCITY UNITS)

METHOD

PROGRAM COC001 USES THE FOLLOWING FORMULAS IN ITS COMPUTATIONS (SEE FLOWCHART COC001)

$$\alpha = \lambda_0 + \lambda_p + t_i \omega_e$$

$$\left.\begin{aligned} X_{3G} &= X_{1G}\cos\alpha - Y_{1G}\sin\alpha \\ Y_{3G} &= X_{1G}\sin\alpha + Y_{1G}\cos\alpha \\ Z_{3G} &= Z_{1G} \end{aligned}\right\} \quad \bar{R}_{3G}$$

$$\left.\begin{aligned} \dot{X}_{3G} &= \dot{X}_{1G}\cos\alpha - \dot{Y}_{1G}\sin \\ \dot{Y}_{3G} &= \dot{X}_{1G}\sin\alpha + \dot{Y}_{1G}\cos\alpha \\ \dot{Z}_{3G} &= \dot{Z}_{1G} \end{aligned}\right\} \quad \bar{V}_{3G}$$

$$\bar{R}_3 = K_{RG}(\bar{R}_{3G})$$

$$\bar{V}_3 = K_{VG}(\bar{V}_{3G})$$

USAGE

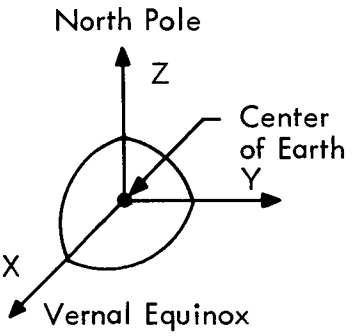
PROGRAM COCOO1 OBTAINS CONTROL FROM ITS PREFIX, MPCCGB, AFTER IT HAS BEEN DETERMINED THAT THE DATA IS OF GOOD QUALITY. IT THEN TRANSFERS CONTROL TO ITS SUFFIX, MFCCGB.

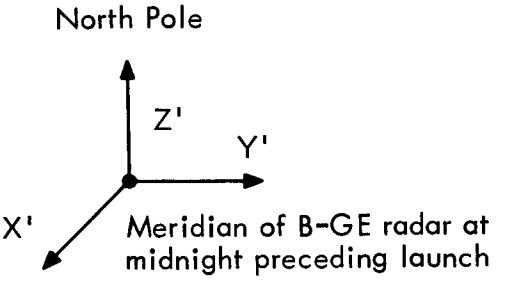
A) STORAGE REQUIRED - 47 LOCATIONS

B) TIME REQUIRED (7094) - 1.040 MILLISECONDS INCLUDING TIME REQUIRED BY SUBROUTINE U1SICO.

8. ROTATION OF B-GE COORDINATE SYSTEM TO GEMINI COORDINATE SYSTEM.

THE GEMINI COORDINATE SYSTEM IS DEFINED AS FOLLOWS - THE Z-AXIS ALWAYS POINTS TOWARD THE NORTH POLE, THE X-AXIS ALWAYS POINTS TOWARD THE VERNAL EQUINOX AND THE Y-AXIS IS SO CHOSEN THAT (X,Y,Z) FORMS A RIGHT-HANDED SET.





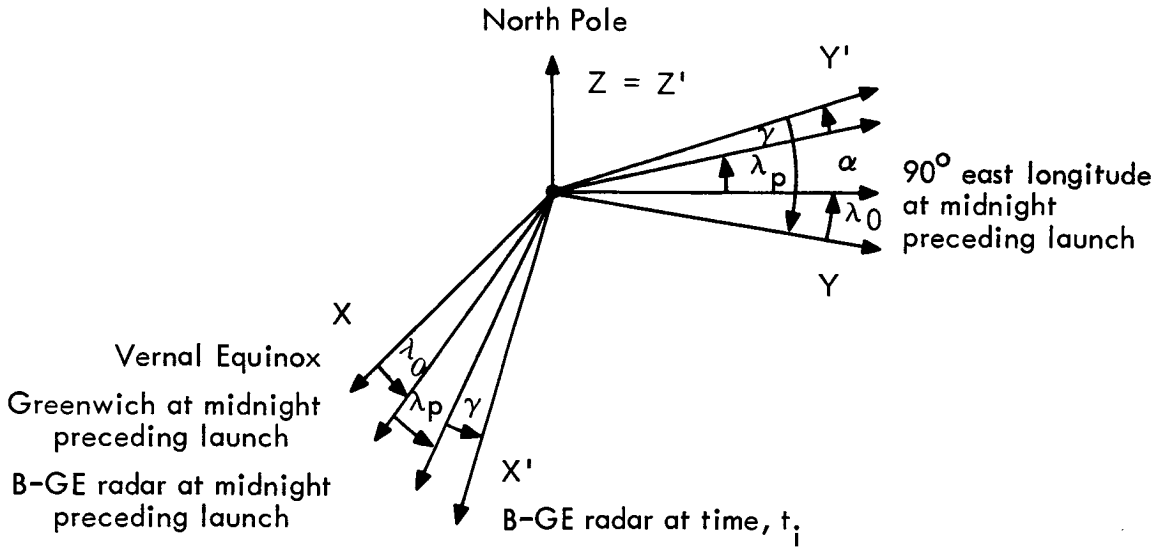
THE B-GE COORDINATE SYSTEM IS REDEFINED EVERY COMPUTING CYCLE. THE Z(AXIS ALWAYS POINTS TOWARD THE NORTH POLE. AT MIDNIGHT PRECEDING LAUNCH, THE X'-AXIS IS SO DEFINED THAT IT LIES IN THE EQUATCRIAL PLANE AND PASSES THROUGH THE MERIDIAN OF THE B-GE RADAR (THUS HAS INERTIAL LONGITUDE $\lambda_{\phi} + \lambda_{p}$ THE Y'-AXIS IS SO CHOSEN THAT (X',Y',Z') FORMS A RIGHT-HANDED SET. THEREAFTER, THE (X',Y') PLANE IS ROTATED ABOUT THE Z'-AXIS SO THAT THE X'-AXIS PASSES THROUGH A POINT WHOSE INERTIAL LCNGITUDE IS GIVEN BY

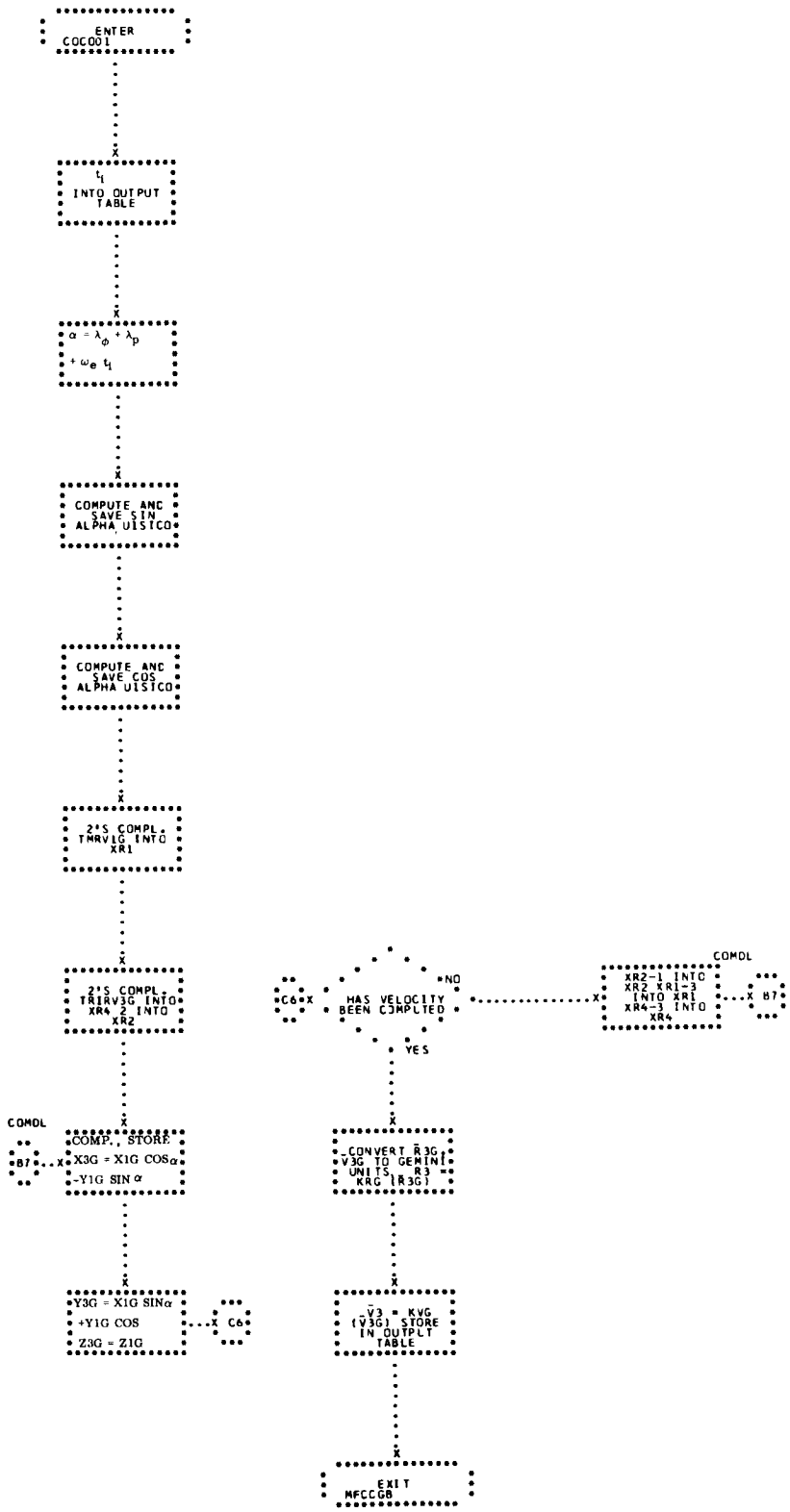
$\lambda_\phi + \lambda + \omega_e t_i$ WHERE IS THE TOTAL AMOUNT OF TIME WHICH HAS ELAPSED SINCE MIDNIGHT PRECEDING LAUNCH.

THE ANGLE THROUGH WHICH THE B-GE COORDINATE SYSTEM MUST BE ROTATED TO OBTAIN THE GEMINI COORDINATE SYSTEM IS GIVEN BY -

$$-\alpha = \lambda_\phi + \lambda_p + \omega_e t_i = \lambda_\phi + \lambda_p + \gamma$$

(THE SIGN OF α IS NEGATIVE BECAUSE THE ROTATION (CLOCKWISE) IS IN A DIRECTION OPPOSITE TO THAT IN WHICH THE ANGLE IS MEASURED (COUNTERCLOCKWISE).




ENTER COC001
t_1 INTO OUTPUT TABLE
$\alpha = \lambda_\phi + \lambda_p + \omega_e t_1$
COMPUTE AND SAVE SIN ALPHA UISICO
COMPUTE AND SAVE COS ALPHA UISICO
2'S COMPL. TMRV1G INTO XR1
2'S COMPL. TRIRV3G INTO XR4 2 INTO XR2
COMP., STORE X3G = X1G COSα -Y1G SIN α
Y3G = X1G SINα +Y1G COS Z3G = Z1G
C6
B7
COMDL
HAS VELOCITY BEEN COMPUTED
NO
YES
CONVERT $\bar{R}$3G, V3G TO GEMINI UNITS, R3 = KRG (R3G)
$\bar{V}$3 = KVG (V3G) STORE IN OUTPLT TABLE
EXIT MFCCGB
XR2-1 INTO XR2 XR1-3 INTO XR1 XR4-3 INTO XR4

9. MFCCGB - MONITOR SUFFIX TO COCOO1

MFCCGB PROCESSES THE OUTPUT FROM THE B/GE COORDINATE CONVERSICN PROCESSOR, COCOO1. (SEE FLOWCHART.)

INPUT

(NOT APPLICABLE)

OUTPUT

LOCATION OF COCOO1 OUTPUT TABLE PLACED IN QUEUE BLOCK FOR CCSTRP.

METHOD

MFCCGB QUEUES THE STRIPCHART PROCESSOR, CCSTRP, WITH THE OUTPUT VECTOR FROM COCOO1.

THE A INDICATOR IS THEN TURNED OFF AND CONTROL IS TRANSFERRED TO MOPRIO.

USAGE

MFCCGB IS ENTERED FROM COCOO1 AND EXITS TO MOPRIO.

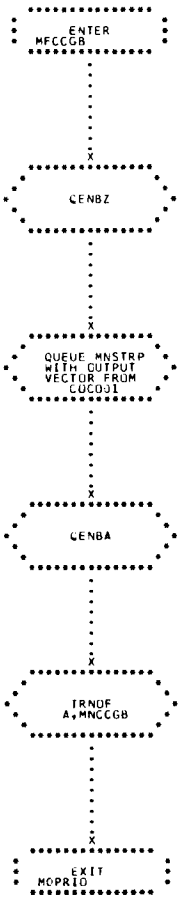
A) PARAMETERS -
   A
   MNSTRP
   MNCCGB

B) MACROS -
   QENBZ
   QENBA
   QUEUE
   TRNOF

C) STORAGE REQUIRED -
   8 LOCATIONS

D) TIMING (7094) -

.094 MILLISECONDS


ENTER
MFCCGB
GENBZ
QUEUE MNSTRP
WITH OUTPUT
VECTOR FROM
COCO01
GENBA
IRNDF
A,MNCCGB
EXIT
MOPRIO

10. MPCCIP - MONITOR PREFIX TO CSC002

MPCCIP SETS UP THE INPUT FOR THE CURRENT ENTRY TO THE IP COORDINATE CONVERSION PROCESSOR, CSC002. (SEE FLOWCHART.)

INPUT

A) COMMUNICATION CELL -

MCLFTM	GMT OF LIFT OFF IN FLOATING POINT SECONDS.

B) TABLE -

TMH2DB	A 7-LOCATION TABLE CONTAINING TIME IN GMT OR ELAPSED SECOND, AND $\overline{r}$ AND $\overline{v}$ IN FLOATING POINT IP UNITS.

OUTPUT

A) TABLES -

TMRV1I	A 7-LOCATION TABLE CONTAINING THE TIME IN GMT SECONDS AND $\overline{r}$ AND $\overline{v}$ IN FLOATING POINT IP UNITS.

METHOD

MPCCIP TURNS ON THE A INDICATOR AND TURNS OFF THE B INDICATOR FOR THE IP COORDINATE CONVERSION PROCESSOR. IT THEN SUPPRESSES THE IP INPUT PROCESSOR, IOHS09.

THE VECTOR, SOURCE ID, AND DATA QUALITY INDICATOR ARE THEN MOVED FROM THE OUTPUT BUFFER FROM IOHS09 TO THE INPUT BUFFER FOR CSC002.

SENSE SWITCH 6 IS INTERROGATED TO DETERMINE WHETHER

THE TIME OF LIFT OFF SHOULD BE ADDED TO THE VECTOR TIME.

IOHS09 IS UNSUPPRESSED AND CONTROL IS TRANSFERRED TO CSC002.

USAGE

MPCCIP IS ENTERED FROM MOPRIO WHEN PRIORITY CONDITIONS PERMIT. IT TRANSFERS CONTROL TO THE IP COORDINATE CONVERSION PROCESSOR, CSC002.

A) PARAMETERS -

A

B

C

MNCCIP

MNHS09

B) MACROS -

TRNON

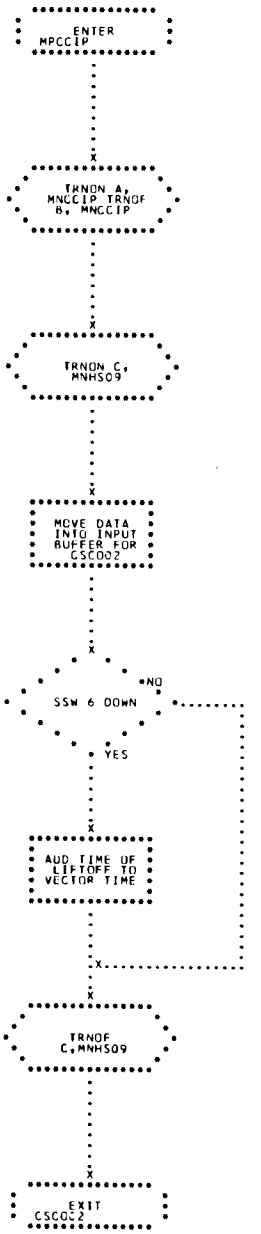
TRNOF

C) STORAGE REQUIRED -

22 LOCATIONS

D) TIMING (7094) -

.099 MILLISECONDS.


ENTER MPCCIP
TRNON A, MNCCIP TRNOF B, MNCCIP
TRNON C, MNHS09
MOVE DATA INTO INPUT BUFFER FOR CSC002
SSW 6 DOWN
NO
YES
ADD TIME OF LIFTOFF TO VECTOR TIME
TRNOF C,MNHS09
EXIT CSC002

11. CSC002 - LAUNCH COORDINATE CONVERSION OF IP-7094 DATA.

THIS PROGRAM ROTATES THE COORDINATES OF THE INPUT VECTORS FROM THE IP-7094 COORDINATE SYSTEM TO THE GEMINI COORDINATE SYSTEM. THE RESULTING VECTORS ARE THEN CONVERTED TO GEMINI LENGTH AND VELOCITY UNITS. (SEE FLOWCHART.)

INPUT

A) CONSTANTS -

K.OMES	ROTATION OF EARTH IN FLOATING POINT RADIANS/SECOND, ω_e
KLAMDO	LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH IN FLOATING POINT RADIANS, λ_0
KRITOG	CONVERSION OF IP-7094 LENGTH TO GEMINI LENGTH UNITS, K_{RI}
KVITOG	CONVERSION OF IP-7094 VELOCITY UNITS TO GEMINI VELOCITY UNITS, K_{VI}

B) TABLES -

TMRV1I — A 9-LOCATION TABLE CONTAINING IN THE FIRST SEVEN LOCATIONS THE INPUT VECTORS AND THEIR ASSOCIATED TIME

TMRV1I	t_i	- TIME OF VECTOR (FLOATING POINT GMT SECONDS)
+1 - +3	$\overline{R}_{1I}$	- POSITION VECTOR (FLOATING POINT IP-7094 LENGTH UNITS)
+4 - +6	$\overline{V}_{1I}$	VELOCITY VECTOR (FLOATING POINT IP-7094 VELO-

CITY UNITS)

C) THE LIBRARY SUBROUTINE U1SICO IS ALSO USED.

OUTPUT

A) TABLES -

TMRV3I AN 11-LOCATION TABLE CONTAINING IN THE FIRST SEVEN CELLS THE CONVERTED VECTORS AND THEIR ASSOCIATED TIME

TMRV3I	t_i	- (SAME AS IN TMRV1I)
+1 - +3	$\overline{R}_3$	- POSITION VECTOR (FLOATING POINT GEMINI LENGTH UNITS)
+4 - +6	$\overline{V}_3$	- VELOCITY VECTOR (FLOATING POINT GEMINI VELOCITY UNITS

METHOD

CSC002 USES THE FOLLOWING FORMULAS IN ITS COMPUTATIONS (SEE FLOWCHART) -

$$\left.\begin{aligned} \phi &= \lambda_0 + \omega_e t_i \\ X_{3I} &= X_{1I}\cos\phi - Y_{1I}\sin\phi \\ Y_{3I} &= X_{1I}\sin\phi + Y_{1I}\cos\phi \\ Z_{3I} &= Z_{1I} \end{aligned}\right\} \overline{R}_{3I}$$

$$\left.\begin{aligned} \dot{X}_{31} &= \dot{X}_{1I}\cos\phi - \dot{Y}_{1I}\sin\phi \\ \dot{Y}_{31} &= \dot{X}_{1I}\sin\phi + \dot{Y}_{1I}\cos\phi \\ \dot{Z}_{3I} &= \dot{Z}_{1I} \end{aligned}\right\} \overline{V}_{3I}$$

$$\overline{R}_3 = K_{RI}(\overline{R}_{3I})$$

$$\overline{V}_3 = K_{VI}(\overline{V}_{3I})$$

USAGE

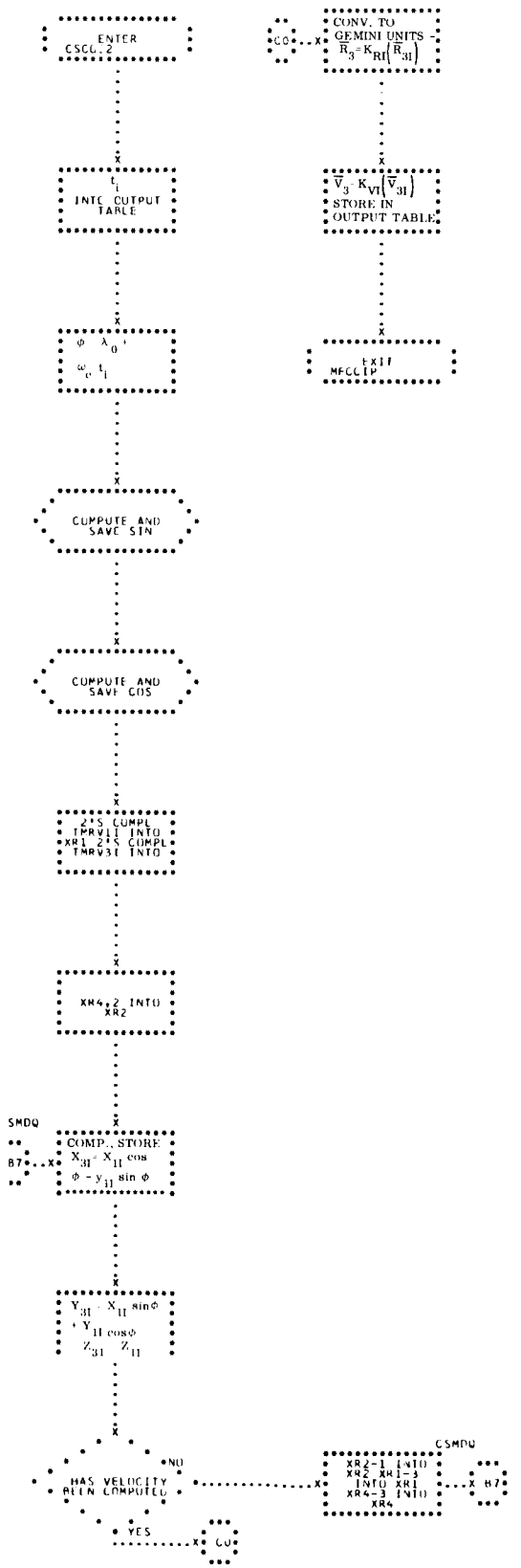
CSC00I RECEIVES CONTROL FROM ITS PREFIX, MPCCIP. IT THEN GIVES CONTROL TO ITS SUFFIX, MFCCIP.

A) STORAGE REQUIRED -

46 LOCATIONS

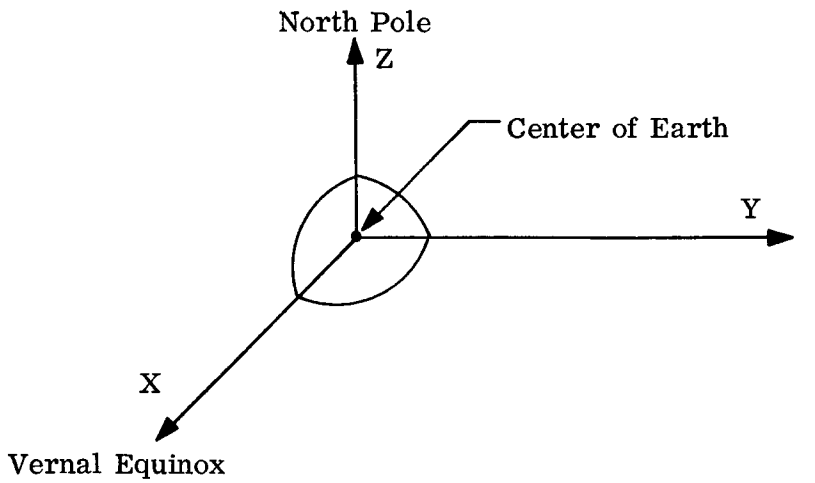
B) TIMING (7094) -

1.016 MILLISECONDS, INCLUDING TIME REQUIRED BY SUBROUTINE U1SICO

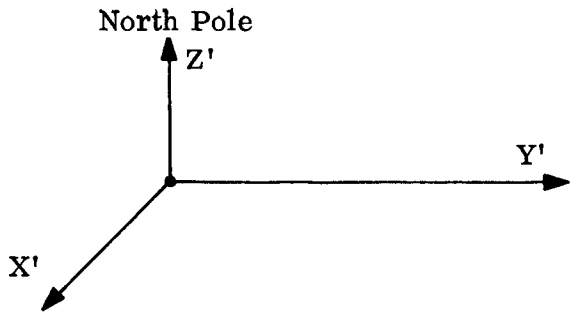

ENTER CSCG.2
CO
CONV. TO GEMINI UNITS - $\bar{R}_3 = K_{RI}(\bar{R}_{3I})$
t_i INTO OUTPUT TABLE
$\bar{V}_3 = K_{VI}(\bar{V}_{3I})$ STORE IN OUTPUT TABLE
ϕ λ_0 ω_o t_i
EXIT MEGCIP
COMPUTE AND SAVE SIN
COMPUTE AND SAVE COS
2'S COMPL TMRV1I INTO XR1 2'S COMPL TMRV3I INTO
XR4,2 INTO XR2
SMDQ
B7
COMP., STORE $X_{3I} = X_{1I} \cos \phi - y_{1I} \sin \phi$
$Y_{3I} = X_{1I} \sin \phi + Y_{1I} \cos \phi$
$Z_{3I} = Z_{1I}$
HAS VELOCITY BEEN COMPUTED
NO
YES
CSMDQ
XR2-1 INTO XR2 XR1-3 INTO XR1 XR4-3 INTO XR4
B7
CO

12. ROTATION OF IP-7094 COORDINATE SYSTEM TO GEMINI COORDINATE SYSTEM

THE GEMINI COORDINATE SYSTEM IS DEFINED AS FOLLOWS - THE Z-AXIS ALWAYS POINTS TO THE NORTH POLE, THE X-AXIS POINTS TOWARD THE VERNAL EQUINOX AND THE Y-AXIS IS SO CHOSEN THAT (X,Y,Z) FORMS A RIGHT-HANDED SET.



THE IP-7094 COORDINATE SYSTEM IS REDEFINED EVERY COMPUTING CYCLE. THE Z'-AXIS ALWAYS POINTS TOWARD THE NORTH POLE. AT MIDNIGHT PRECEDING LAUNCH, THE X'-AXIS IS DEFINED SO THAT IT LIES IN THE EQUATORIAL PLANE AND PASSES THROUGH THE GREENWICH MERIDIAN (THUS HAS INERTIAL LONGITUDE λ_0). THE Y'-AXIS IS CHOSEN SO THAT (X',Y',Z') FORMS A RIGHT-HANDED SET. THEREAFTER, THE (X',Y') PLANE IS ROTATED ABOUT THE Z'-AXIS SO THAT THE X'-AXIS PASSES THROUGH A POINT HAVING INERTIAL LONGITUDE $\lambda_0 + \omega_e t_i$ WHERE t_i IS THE TOTAL AMOUNT OF TIME WHICH HAS ELAPSED SINCE MIDNIGHT PRECEDING LAUNCH.

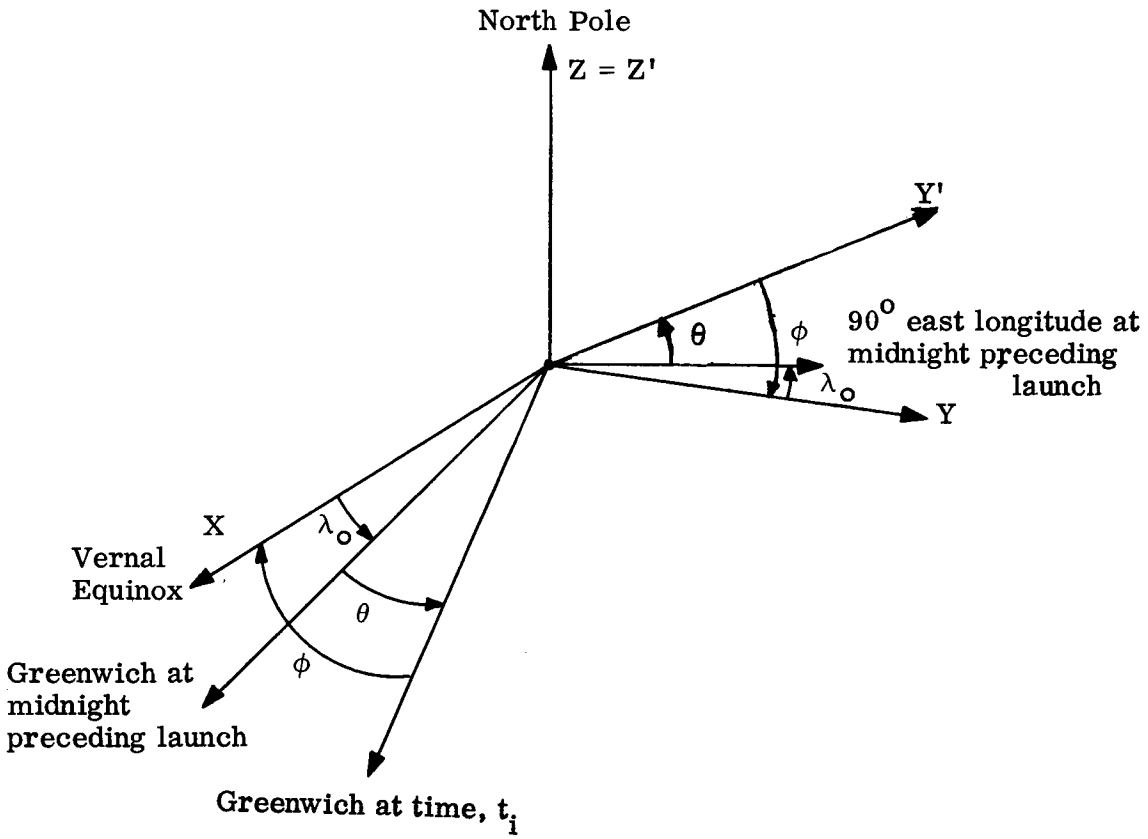


Greenwich meridian at
midnight preceding launch

THE ANGLE THROUGH WHICH THE IP-7094 COORDINATE SYSTEM MUST BE ROTATED TO OBTAIN THE GEMINI COORDINATE SYSTEM IS GIVEN BY -

$$-\varphi=\lambda_0 + \omega_e = \lambda_0+\Theta$$

(THE SIGN OF φ IS NEGATIVE BECAUSE THE ROTATION (CLOCKWISE) IS IN A DIRECTION OPPOSITE TO THAT IN WHICH THE ANGLE IS MEASURED (COUNTERCLOCKWISE).


North Pole
Z = Z'
Y'
θ
φ
90° east longitude at midnight preceding launch
λo
Y
X
λo
Vernal Equinox
θ
φ
Greenwich at midnight preceding launch
Greenwich at time, ti

13. MFCCIP - MONITOR SUFFIX TO CSC002

MFCCIP PROCESSES THE OUTPUT FROM THE IP COORDINATE CONVERSION PROCESSOR, CSC002. (SEE FLOWCHART).

INPUT

(NOT APPLICABLE)

OUTPUT

LOCATION OF CSC002 OUTPUT TABLE PLACED IN QUEUE BLOCK FOR CCSTRP.

METHOD

MFCCIP QUEUES THE STRIPCHART PROCESSOR, CCSTRP, WITH THE OUTPUT VECTOR FROM CSC002.

THE A INDICATOR IS THEN TURNED OFF AND CONTROL IS TRANSFERRED TO MOPRIO.

USAGE

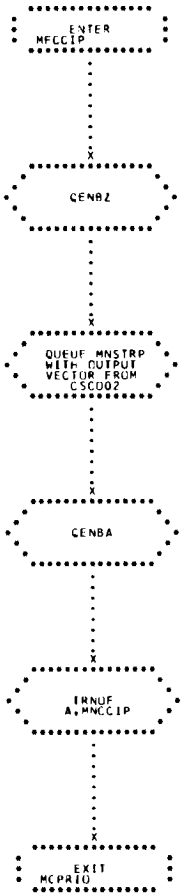
MFCCIP IS ENTERED FROM CSC002 AND EXITS TO MOPRIO.

A) PARAMETERS -
   A
   MNSTRP
   MNCCIP

B) MACROS -
   QENBZ
   QENBA
   QUEUE
   TRNOF

C) STORAGE REQUIRED -
   8 LOCATIONS

D) TIMING (7094) -
.094 MILLISECONDS


ENTER MFCCIP
CENBZ
QUEUE MNSTRP WITH OUTPUT VECTOR FROM CSC002
CENBA
IRNUE A,MNCCIP
EXIT MCPRIO

14. MPEOBD - MONITOR PREFIX TO EOBDIT

THIS PROGRAM DETERMINES WHEN AND UNDER WHAT CONDITIONS THE BERMUDA RADAR DATA EDIT PROGRAM, EOBDIT, IS TO BE ENTERED AND WHICH OF THE TWO PROCESSORS (THE SHORT ARC PROGRAM, CASARC, OR THE GENERAL LEAST SQUARES SMOOTHING PROGRAM, CSOGLS) IS TO BE USED TO PROCESS THE EDITED RADAR DATA OF RANGE, AZIMUTH, AND ELEVATION (r, a, e). (SEE FLOWCHART)

INPUT

A) TABLES -

TC.ONT	A ONE-LOCATION TABLE CONTAINING THE NOMINAL TIME OVER WHICH THRUSTING FOR OAMS BURN FOR SEPARATION OF THE SPACECRAFT IS TO TAKE PLACE IN FLOATING POINT SECONDS, N_t
TMHEDF (TMHEDV)	A 32-LOCATION TABLE CONTAINING THE PARAMETERS OF INPUT, EDIT, SMOOTHING, AND SHORT ARC FOR THE FPC-16 (VERLORT) RADAR DATA
TM8.3M	A ONE-LOCATION TABLE CONTAINING THE 8.3 MILLISECOND COUNT SINCE THE LAST HALF-SECOND IN FIXED POINT,B35

B) CONSTANTS -

K 00001	DEC 1	FIXED POINT ONE
K 00009	DEC 9	FIXED POINT NINE
K8.3M3	DEC 8.3E-3	CONVERSION OF 8.3 MILLISECOND COUNT TO SECONDS
K005.0	DEC 5.0	FLOATING POINT

		FIVE
K010.0	DEC 10.0	FLOATING POINT TEN
K020.0	DEC 20.0	FLOATING POINT TWENTY
KCH233	OCT 233000000000	CHARACTERISTIC OF 233 OCTAL
KCH234	OCT 234000000000	CHARACTERISTIC OF 234 OCTAL

C) COMMUNICATION CELLS -

MCACBD NON-ZERO IF BERMUDA HIGH-SPEED LINES ACTIVATED FOR LOW ABORT PHASE, OTHERWISE, ZERO

MCFRST ZERO ON A FIRST OR RESTART ENTRY TO EDIT, NON-ZERO OTHERWISE, e_{edit}

MCHFSC CONTAINS CURRENT GMT IN FIXED POINT HALF-SECONDS, B35

MCINFF (MCINFV) NON-ZERO IF FPS-16 (VERLORT) RADAR DATA BEING EDITED FOR A NEW FIT, ZERO OTHERWISE, i_{nff} (OR i_{nfv})

MCPROP ZERO IS NO THRUSTING GOING ON, ONE IF THRUSTING TAKING PLACE, P

MCTAST NON-ZERO IF AT LEAST ONE DATA TABLE ESTABLISHED (I.E., CONTAINS AT LEAST FOUR RADAR OBSERVATIONS). ZERO OTHERWISE, i_{sate}

MCTBAR CONTAINS THE COMPUTER'S BEST ESTIMATE OF THE TIME OF SECOND STAGE CUTOFF PLUS APPROXIMATELY 5 SECONDS IN GMT FLOATING POINT SECONDS, t_{bar}

OUTPUT

A) TABLES -

TCTSAS A ONE-LOCATION TABLE CONTAINING THE INITIALIZATION TIME FOR THE SHORT ARC PROGRAM IN FIXED POINT TENTHS OF SECONDS, B25, t_i

B) COMMUNICATION CELLS -

MCERAD CONTAINS INDICATION OF RADAR DATA BEING EDITED - ZERO IF FPS-16 DATA BEING EDITED, NON-ZERO IF VERLORT DATA BEING EDITED, P_e

MCESTL CONTAINS ADDRESSES OF EDIT PARAMETER AND CRITERIA TABLES FOR RADAR TO BE EDITED. SIGN IS NEGATIVE IF RADAR DATA MUST BE EDITED FOR A NEW FIT, OTHERWISE, SIGN IS POSITIVE

MCPROG CONTAINS THE CONTENTS OF MCPROP ON THIS ENTRY, G

MCSAET ZERO IF FIRST OR RESTART ENTRY TO CASARC MUST BE MADE, NON-ZERO OTHERWISE, e_{sa}

MCSAIP NON-ZERO IF SHORT ARC PROGRAM IS TO BE USED, ZERO OTHERWISE, i_{saip}

MCTIME CONTAINS CURRENT GMT IN FLOATING POINT TENTHS OF SECONDS, t_e

METHOD

AS SOON AS CONTROL IS GIVEN TO MPEOBD, THE A BIT IS TURNED ON AND THE B BIT IS TURNED OFF TO SHOW THAT EDIT IS NOW IN PROGRESS AND IS HANDLING THE LATEST RADAR DATA. INDICATIONS OF SPACECRAFT SEPARATION, SECOND STAGE CUT-OFF, AND SPACECRAFT PROPULSION ARE SAVED IN INTERNAL STORAGE. BECAUSE THE BERMUDA HIGH-SPEED LINES REMAIN

DEACTIVATED UNTIL t_{bar} HAS BEEN ESTABLISHED, MPEOBD WILL TRANSFER CONTROL TO ITS SUFFIX, MFEOBD, IF IT FINDS THIS TIME ZERO.

ONCE t_{bar} HAS BEEN SET, MPEOBD CONVERTS IT TO FLOATING POINT TENTHS OF SECONDS AND SAVES IT IN INTERNAL STORAGE. CURRENT TIME IS THEN COMPUTED IN FLOATING POINT TENTHS OF SECONDS AND IS COMPARED TO THE SAVED TIME OF t_{bar}. IF CURRENT TIME IS LESS, THE EDIT PROGRAM WILL NOT BE ENTERED AND CONTROL WILL BE GIVEN TO THE SUFFIX, MFEOBD.

WHEN CURRENT TIME INDICATES THAT THE EDIT PROGRAM MAY ENTERED, MPEOBD DETERMINES WHETHER THE MISSION HAS ENTERED THE HOLD PHASE OR THE LOW ABORT PHASE.

A) ON THE FIRST ENTRY TO EDIT IN THE LOW ABORT PHASE, MPEOBD INDICATES THAT THE GENERAL LEAST SQUARES SMOOTHING PROGRAM, CSOGLS, IS TO BE USED TO PROCESS THE EDITED RADAR DATA BEFORE TRANSFERRING CONTROL TO C). ON SUBSEQUENT ENTRIES TO EDIT IN THE LOW ABORT PHASE, CONTROL IS GIVEN DIRECTLY TO C).

B) ON THE FIRST ENTRY TO EDIT IN THE HOLD PHASE, MPEOBD INDICATES THAT THE SHORT ARC PROGRAM, CASARC, IS TO BE USED IN PROCESSING THE EDITED RADAR DATA AND SETS THE INITIALIZATION TIME FOR THAT PROGRAM TO THE SAVED TIME OF t_{bar}. MPEOBD THEN CONTINUES AS IN C).

ON SUBSEQUENT ENTRIES TO EDIT IN THE HOLD PHASE, MPEOBD PASSES CONTROL TO C), IF EITHER THE NOMINAL THRUSTING TIME FOR OAMS BURN FOR SPACECRAFT SEPARATION HAS NOT PASSED OR THIS TIME HAS ELAPSED, BUT THE PROPULSION INDICATOR HAS NOT CHANGED SINCE THE PREVIOUS ENTRY. WHEN

A CHANGE IS DETECTED IN THE PROPULSION INDICATOR AFTER THE NOMINAL THRUSTING TIME HAS ELAPSED, MPEOBD REINITIALIZES CASARC BY SETTING THE INITIALIZATION TIME TO CURRENT TIME. CONTROL THEN PASSES TO C).

C) FPS-16 RADAR DATA IS FIRST EDITED BY EOBDIT IF AT LEAST FOUR RADAR OBSERVATIONS OF RANGE, AZIMUTH, AND ELEVATION HAVE BEEN PLACED IN THE DATA TABLE. IF FOUR OBSERVATIONS DO NOT EXIST IN THE TABLE, CONTROL IS GIVEN TO MFEOBD.

WHILE DATA IS BEING EDITED, THE DATA TABLE MOVE IS SUPPRESSED TO PREVENT DESTRUCTION OF DATA BEFORE IT CAN BE FURTHER PROCESSED. IF THE EDITED DATA IS TO BE PROCESSED BY CSOGLS, MPEOBD DETERMINES WHETHER THE DATA IS BEING EDITED FOR A NEW FIT AND SETS THE SIGN OF MCESTL ACCORDINGLY. EOBDIT IS THEN ENTERED

USAGE

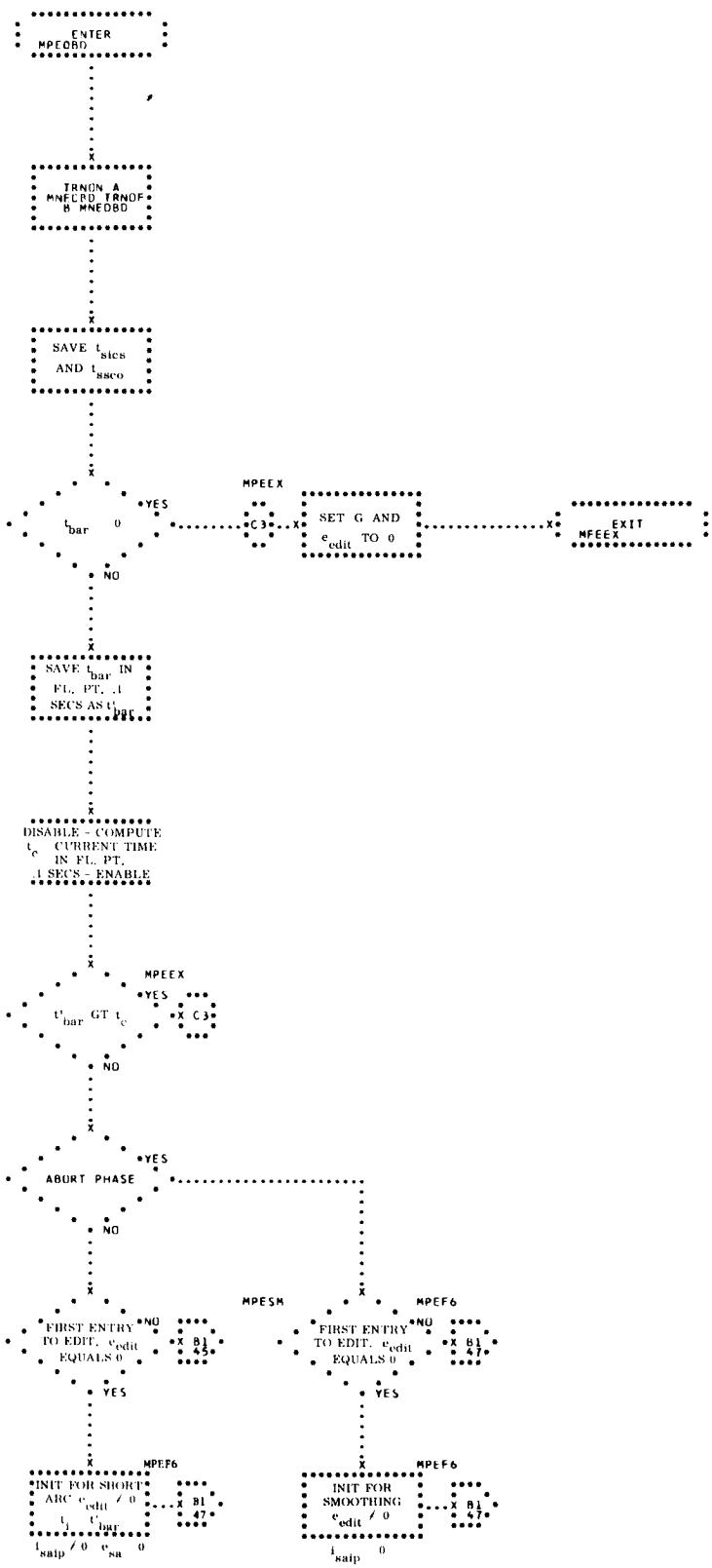
MPEOBD RECEIVES CONTROL FROM MOPRIO ONCE PER SECOND - IT TRANSFERS CONTROL EITHER TO EOBDIT OR TO MFEOBD.

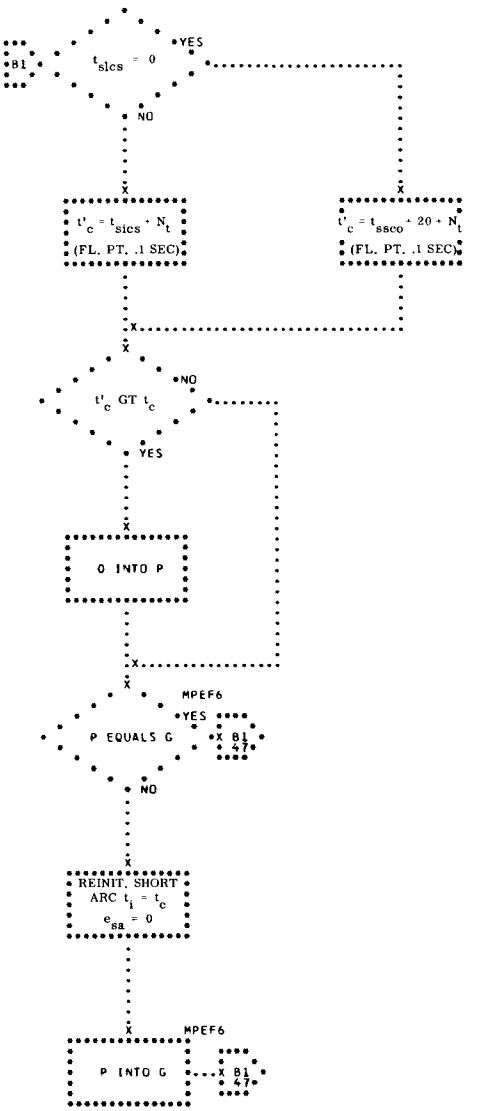
1. STORAGE REQUIRED - 109 LOCATIONS
2. TIMING (7094) - .048 MILLISECONDS, MINIMUM
   .330 MILLISECONDS, MAXIMUM

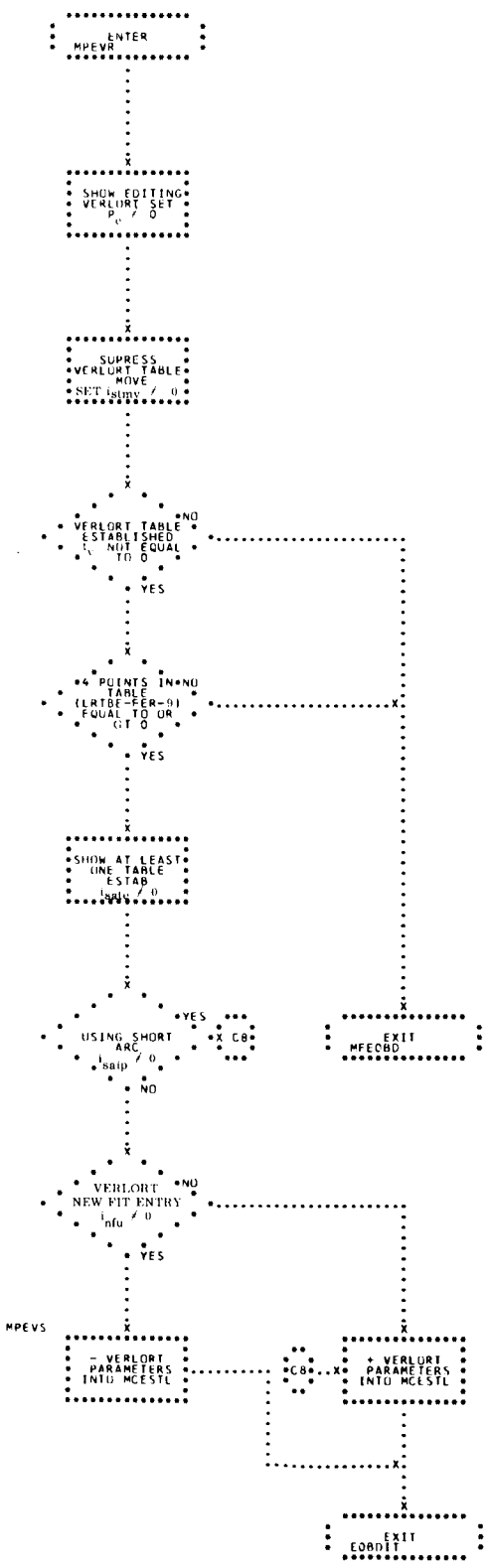
LOCATION	DESCRIPTION	
TMHEDH	LRTBE	LOCATION IN DATA TABLE OF LAST RANGE TO BE EDITED
+1	FER	LOCATION IN DATA TABLE OF FIRST EDITED RANGE
+2	LER	LOCATION IN DATA TABLE OF LAST EDITED RANGE
+3	FEA	LOCATION IN DATA TABLE OF FIRST EDITED AZIMUTH
+4	LEA	LOCATION IN DATA TABLE OF LAST EDITED AZIMUTH
+5	FEE	LOCATION IN DATA TABLE OF FIRST EDITED ELEVATION
+6	LEE	LOCATION IN DATA TABLE OF LAST EDITED ELEVATION
+7	LR	LOCATION IN DATA TABLE OF RANGE OF LAST COMPLETELY EDITED POINT
+8	LAR	LOCATION IN DATA TABLE OF RANGE OF LAST POINT USED BY CASARC
+9	i_{STMH}	MOVE SUPPRESSION INDICATOR
+10		LOCATION OF RADAR DATA TABLE
+11		GRANULARITY REFERENCE
+12		MOVE INSTRUCTION
+13		LOCATION IN DATA TABLE OF FIRST RANGE
+14		MOVE INSTRUCTION
+15		MOVE INSTRUCTION
+16		RADAR IDENTIFIER
+17		TABLE ESTABLISHED INDICATOR
+18		MOVE PARAMETER
+19		MOVE PARAMETER
+20		REINITIALIZATION PARAMETER

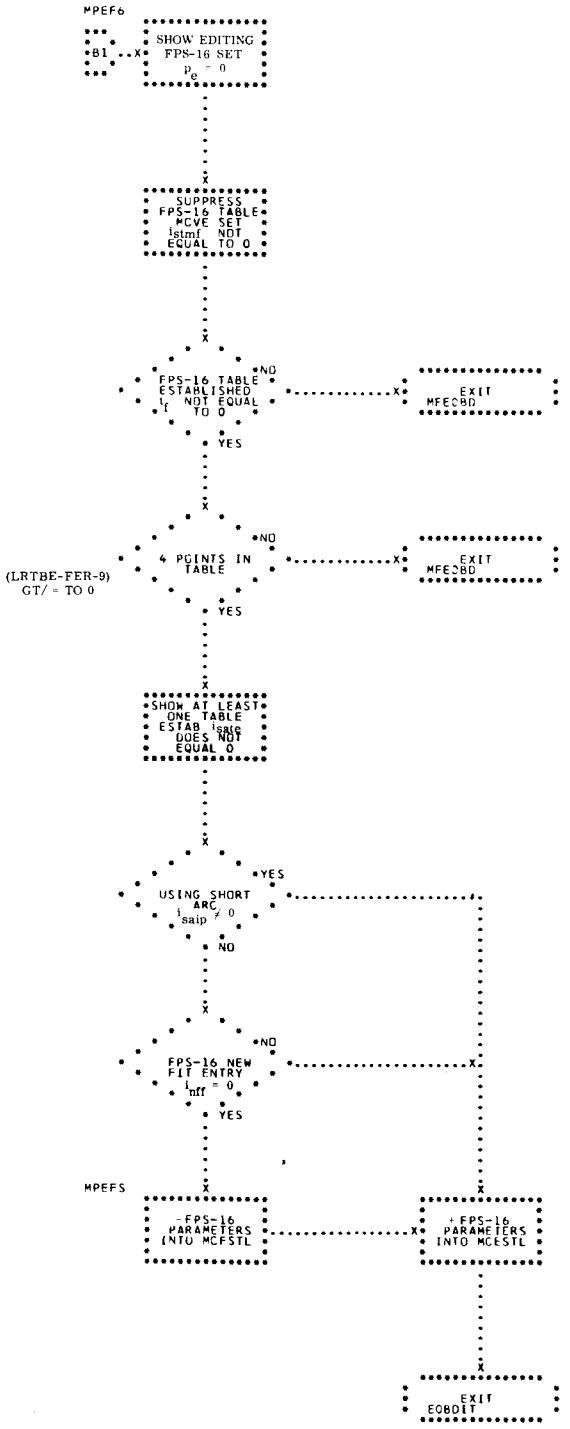
LOCATION	DESCRIPTION	
+21		REINITIALIZATION PARAMETER
+22		TIME OF LAST OBSERVATION RECEIVED
+23		FORCED MOVE SIZE
+24		LAST LOCATION IN DATA TABLE WHICH SHOULD BE USED
+25		MAXIMUM MOVE SIZE
+26	$L(L(X_0))$	LOCATION OF LOCATION IN DATA TABLE OF FIRST COMPONENT, MID-POINT OF U, V, W SUMS
+27	FR	LOCATION IN DATA TABLE OF RANGE OF FIRST COMPLETELY EDITED POINT
+28	FRTC	LOCATION IN DATA TABLE OF FIRST COMPONENT OF FIRST POINT TO BE CONVERTED
+29	L(FCR)	LOCATION OF LOCATION IN DATA TABLE OF FIRST COMPONENT OF FIRST CONVERTED POINT
+30	LRTC	LOCATION IN DATA TABLE OF FIRST COMPONENT OF LAST POINT TO BE COMVERTED
+31	L(LCR)	LOCATION OF LOCATION IN DATA TABLE OF FIRST COMPONENT OF LAST CONVERTED POINT

1. H IS EITHER F (FOR FPS-16 RADAR DATA) OR V (FOR VERLORT RADAR DATA).
2. ALL QUANTITIES ARE OCTAL UNLESS OTHERWISE SPECIFIED.


ENTER MPEOBD
TRNON A MNECRD TRNOF B MNEOBD
SAVE t_{sics} AND t_{ssco}
t_{bar} 0
YES
NO
MPEEX
C3
SET G AND e_{edit} TO 0
EXIT MFEEX
SAVE t_{bar} IN FL. PT. .1 SECS AS t'_{bar}
DISABLE - COMPUTE t_c CURRENT TIME IN FL. PT. .1 SECS - ENABLE
t'_{bar} GT t_c
ABORT PHASE
MPESM
MPEF6
FIRST ENTRY TO EDIT, e_{edit} EQUALS 0
B1 45
B1 47
INIT FOR SHORT ARC e_{edit} / 0
t_j t'_{bar}
i_{saip} / 0 e_{sa} 0
INIT FOR SMOOTHING e_{edit} / 0
i_{saip} 0


B1
$t_{slcs} = 0$
YES
NO
$t'_c = t_{slcs} + N_t$
(FL. PT. .1 SEC)
$t'_c = t_{ssco} + 20 + N_t$
(FL. PT. .1 SEC)
t'_c GT t_c
NO
YES
0 INTO P
MPEF6
P EQUALS G
YES
X B1
47
NO
REINIT. SHORT ARC $t_i = t_c$
$e_{sa} = 0$
MPEF6
P INTO G
X B1
47


ENTER
MPEVR
SHOW EDITING
VERLORT SET
P ≠ 0
SUPRESS
VERLORT TABLE
MOVE
SET isuny ≠ 0
VERLORT TABLE
ESTABLISHED
NOT EQUAL
TO 0
NO
YES
4 POINTS IN
TABLE
(LATBE-FER-9)
EQUAL TO OR
GT 0
NO
YES
SHOW AT LEAST
ONE TABLE
ESTAB
isaty ≠ 0
USING SHORT
ARC
isaip ≠ 0
YES
NO
X C8
EXIT
MFEOBD
VERLORT
NEW FIT ENTRY
infu ≠ 0
NO
YES
MPEVS
- VERLORT
PARAMETERS
INTO MCESTL
C8
+ VERLORT
PARAMETERS
INTO MCESTL
EXIT
EOBDIT


MPEF6
B1
SHOW EDITING FPS-16 SET
$p_e = 0$
SUPPRESS FPS-16 TABLE MOVE SET
i_{stmf} NOT EQUAL TO 0
FPS-16 TABLE ESTABLISHED
i_f NOT EQUAL TO 0
NO
YES
EXIT MFE2BD
4 POINTS IN TABLE
(LRTBE-FER-9) GT/ = TO 0
NO
YES
EXIT MFE2BD
SHOW AT LEAST ONE TABLE ESTAB i_{smte} DOES NOT EQUAL 0
USING SHORT ARC
$i_{saip} \neq 0$
YES
NO
FPS-16 NEW FIT ENTRY
$i_{nff} = 0$
NO
YES
MPEF5
-FPS-16 PARAMETERS INTO MCFSTL
+FPS-16 PARAMETERS INTO MCESTL
EXIT EOBOIT

15. EOBDIT - BERMUDA HIGH-SPEED EDIT

THIS PROGRAM EDITS ALL UNPROCESSED AN/FPS-16 AND VERLORT RADAR OBSERVATIONS RECEIVED BY WAY OF THE HIGH-SPEED DATA LINES FROM BERMUDA PRIOR TO THEIR USE BY THE SMOOTHING OR SHORT PROCESSORS. (SEE FLOWCHARTS.)

INPUT

A) TABLES -

TCBDF6 (OR TCBDVE)	A 601-WORD TABLE CONTAINING IN THE FIRST WORD, THE GMT TIME TAG IN 0.2 SECOND INCREMENTS, FIXED POINT, B35 OF THE FIRST OBSERVATION IN THE TABLE. THE REMAINING 600 WORDS OF THE TABLE CONTAIN UP TO 200 3-WORD AN/FPS-16 (OR VERLORT) RADAR OBSERVATIONS. THE FIRST OF THE THREE WORDS IN EACH OF THESE OBSERVATIONS CONTAINS THE SLANT RANGE (R), EXPRESSED AS FIXED POINT FEET (B35), WHILE THE SECOND AND THIRD WORDS CONTAIN AZIMUTH (A) AND ELEVATION (E), RESPECTIVELY, EXPRESSED AS FIXED POINT RADIANS (B14). THE TIME INCREMENT BETWEEN SUCCESSIVE OBSERVATIONS IN THE TABLE IS 0.1 SECOND.
TMHCRF (OR TMHCRV	A 9-WORD TABLE OF EDIT CRITERIA FOR AN/FPS-16 (OR VERLORT) RADAR DATA CONTAINING THE FOLLOWING LIMITS - $+0\sqrt{2.5}$ TIMES ALLOWABLE RANGE ERROR

+1 2.0 TIMES ALLOWABLE RANGE ERROR
+2 $\sqrt{6.0}$ TIMES ALLOWABLE RANGE ERROR
+3 $\sqrt{2.5}$ TIMES ALLOWABLE AZIMUTH ERROR
+4 2.0 TIMES ALLOWABLE AZIMUTH ERROR
+5 $\sqrt{6.0}$ TIMES ALLOWABLE AZIMUTH ERROR
+6 $\sqrt{2.5}$ TIMES ALLOWABLE ELEVATION ERROR
+7 2.0 TIMES ALLOWABLE ELEVATION ERROR
+8 $\sqrt{6.0}$ TIMES ALLOWABLE ELEVATION ERROR

TMHEDF (OR TMHEDV) A 32-WORD TABLE OF WHICH THE FIRST SEVEN WORDS ARE USED IN EDITING THE RADAR OBSERVATIONS OF THE TCBDF6 (OR TCBDVE) TABLE. THE FIRST SEVEN WORDS ARE -

+0 LOCATION OF THE LAST RANGE ENTRY LRTBE OF THE LAST OBSERVATION.
+1 LOCATION OF THE FIRST EDITED RANGE ENTRY FER
+2 LOCATION OF THE LAST EDITED RANGE ENTRY (LER)
+3 LOCATION OF THE FIRST EDITED AZIMUTH ENTRY (FEA)
+4 LOCATION OF THE LAST EDITED AZIMUTH ENTRY (LEA)
+5 LOCATION OF THE FIRST EDITED ELEVATION ENTRY (FEE)

+6 LOCATION OF THE LAST EDITED ELEVATION ENTRY (LEE)

THE ENTRIES OF THE TABLE TCBDF6 (OR TCBDVE), WITH THE FIRST WORD OF TMHEDF (OR TMHEDV), ARE STORED BY THE BERMUDA HIGH-SPEED RADAR PROCESSOR, IOHSBD. THE NEXT SIX ENTRIES OF TMHEDF (OR TMHEDV) ARE INITIALIZED BY MONITOR PROGRAMS ON THE FIRST OR RESTART EDIT ENTRIES AND ADJUSTED BY THE BERMUDA HIGH-SPEED RADAR DATA PROCESSOR WHEN THE DATA TABLE IS MOVED TO ACCOMODATE NEW DATA. EOBDIT, ITSELF, ALSO ADJUSTS THESE QUANTITIES AS EDITING PROGRESSES.

B) CONSTANTS -

KM0001	DEC	-1	FIXED POINT MINUS ONE
K00001	DEC	1	FIXED POINT ONE
K00002	DEC	2	FIXED POINT TWO
K00003	DEC	3	FIXED POINT THREE
K00006	DEC	6	FIXED POINT SIX
K00009	DEC	9	FIXED POINT NINE
KVMNRP	DEC	30	THREE TIMES THE MAXIMUM NUMBER OF BAD POINTS ALLOWED, IN FIXED POINT

C) COMMUNICATION CELLS -

MCESTL SIGN - NEGATIVE IF THE DATA IS IN A BAD FIT AND THE SMOOTHING PROGRAM, CSOGLS, IS TO BE USED. OTHERWISE POSITIVE.

ADDR - LOCATION OF THE TABLE OF EDIT PARAMETERS FOR EITHER AN/FPS-16 DATA (TMHEDF) OR VERLORT DATA (TMHEDV).

DECR - LOCATION OF TABLE OF EDIT CRITERIA FOR EITHER AN/FPS-16 DATA

(TMHCRF) OR VERLORT DATA (TMHCRV).

MCSAIP — NON-ZERO IF THE SHORT ARC PROGRAM, CASARC, IS TO BE USED. OTHERWISE ZERO.

OUTPUT

A) TABLES -

TCBDF6 (OR TCBDVE) — A 601-LOCATION TABLE CONTAINING THE EDITED RADAR OBSERVATIONS IN THE FOLLOWING FORMAT - TIME TAG (GMT) OF THE FIRST OBSERVATION IN THE TABLE IN FIXED POINT 0.1 SECOND INCREMENTS, B35.

TCBDF6 + C(TMHEDF+1)	FER
TCBDF6 + C(TMHEDF+3)	FEA
TCBDF6 + C(TMHEDF+5)	FEE
.	.
.	.
.	.
TCBDF6 + C(TMHEDF+2)	LER
TCBDF6 + C(TMHEDF+4)	LEA
TCBDF6 + C(TMHEDF+6)	LEE

THE TABLE TCBDVE IS SIMULARLY ARRANGED.

TMHEDF (OR TMHEDV) — A 32-LOCATION TABLE CONTAINING THE RANGE LOCATION OF THE LATEST OBSERVATION FOR WHICH ALL THREE COORDINATES HAVE BEEN EDITED IN THE EIGHTH LOCATION, TMHEDF+7 (OR TMHEDV+7), AND THE RANGE LOCATION OF THE EARLIEST OBSERVATION FOR WHICH ALL

THREE COORDINATES HAVE BEEN EDITED IN THE TWENTY-EIGHTH LOCATION, TMHEDF+27 (OR TMHEDV+27).

UPON EXIT FROM EOBDIT -

A) IF THERE HAVE BEEN 1 OR 2 BAD POINTS IN SUCESSION, INTERPOLATED VALUES WILL REPLACE THE TRANSMITTED VALUES OF THESE POINTS IN THE TCBDF6 OR TCBDVE TABLES.

B) IF THERE HAVE BEEN MORE THAN 2 BAD POINTS IN SUCCESSION AND CASARC IS TO BE USED, -1'S REPLACE THE TRANSMITTED VALUES OF THE POINTS.

C) IF THERE HAVE BEEN MORE THAN 2 BAD POINTS IN SUCCESSION, BUT LESS THAN THE MAXIMUM NUMBER OF BAD POINTS ALLOWED, AND CSOGLS IS TO BE USED, INTERPOLATED VALUES REPLACE THE TRANSMITTED VALUES OF THE POINTS.

METHOD

WHEN CONTROL IS FIRST GIVEN TO EOBDIT, THE TABLE OF EDIT PARAMETERS, TMHEDF (OR TMHEDV) INDICATES THAT THE RADAR DATA TABLE TCBDF6 (OR TCBDVE) CONTAINS FOUR OR MORE UNEDITED OBSERVATIONS. SUBSEQUENT ENTRIES ARE MADE AT 1.0 SECOND INTERVALS. UPON EACH ENTRY, THE CONTENTS OF MCESTL GIVE THE LOCATIONS OF THE EDIT PARAMETER AND CRITERIA TABLES TO BE USED.

THE EOBDIT PROGRAM FIRST EDITS ALL PREVIOUSLY UNEDITED OBSERVATIONS IN TCBDF6 (OR TCBDVE). THE EDIT PROCEDURE USED IN EOBDIT IS COMPOSED OF SIX DISTINCT STEPS -

A) INITIALIZATION

B) NO PREVIOUS EDITING

C) PREVIOUS EDITING

D) ONE OR TWO CONSECUTIVE REJECTED POINTS

E) THREE OR MORE CONSECUTIVE REJECTED POINTS

F) TEST FOR OTHER UNEDITED COORDINATES

INITIALIZATION

THE FOLLOWING QUANTITES ARE PLACED IN THE ADDRESS PORTIONS OF INDIVIDUAL CORE LOCATIONS -

LOCATION OF LAST RANGE TO BE EDITED (j)

LOCATION OF FIRST EDITED RANGE (h)

LOCATION OF LAST EDITED RANGE (i)

LOCATION OF FIRST EDITED ELEVATION

IF $h = i$, THE DATA TABLE CONTAINS NO EDITED OBSERVATIONS AND A TRANSFER IS MADE TO NO PREVIOUS EDITING. IF $h \neq i$ THE DATA TABLE DOES CONTAIN EDITED OBSERVATION AND TRANSFER IS MADE TO PREVIOUS EDITING.

NO PREVIOUS EDITING

IN THIS CASE, THERE EXIST AT LEAST FOUR POINTS TO BE EDITED. THE FIRST AND THIRD POINTS ARE USED TO PREDICT THE SECOND, WHICH IS THEN COMPARED WITH THE ACTUAL VALUE. IF THIS COMPARISON IS SUCCESSFUL, THE SECOND AND FOURTH POINTS ARE USED TO PREDICT THE THIRD, WHICH IS COMPARED TO ITS ACTUAL VALUE. LINEAR, ALMOST-LEAST-SQUARES FORMULAS ARE USED - $X_2 = (X_1 + X_3)/2$, $X_3 = (X_2 + X_4)/2$.

BY CONSIDERING OVERLAPPING SETS OF FOUR POINTS, THIS TEST IS CONTINUED UNTIL A SET OF FOUR POINTS IS FOUND WHICH SATISFIES BOTH THESE FORMULAS OR UNTIL NO MORE DATA IS AVAILABLE, h AND i ARE UPDATED AT EACH STEP. IF FOUR SATISFACTORY POINTS ARE FOUND, i IS FURTHER UPDATED AND TRANSFER IS MADE TO PREVIOUS EDITING. IF THE DATA IS EXHAUSTED BEFORE FOUR SUCH POINTS ARE FOUND, TRANSFER IS MADE TO TEST FOR OTHER UNEDITED COORDINATES.

PREVIOUS EDITING

THIS IS THE NORMAL CALCULATION. THE RATE OF CHANGE BETWEEN SUCCESSIVE POINTS IS STORED FOR POSSIBLE USE IN THE ''THREE OR MORE CONSECUTIVE REJECTED POINTS'' SECTION. FOUR ESTABLISHED POINTS ARE USED TO PREDICT A FIFTH AND AS YET UNEDITED POINT WHICH IS COMPARED WITH ITS ACTUAL VALUE. IF THE POINT IS ACCEPTED, i IS UPDATED AND THE PROCESS CONTINUES UNTIL A POINT IS REJECTED OR UNTIL THE DATA IS EXHAUSTED. A TRANSFER IS THEN MADE TO ONE OR TWO CONSECUTIVE REJECTED POINTS OR TO TEST FOR OTHER UNEDITED COORDINATES, RESPECTIVELY. THE FOLLOWING LINEAR, ALMOST-LEAST-SQUARE FORMULA IS USED TO PREDICT THE FIFTH POINT -

$$X_5 = X_4 + (X_3 - X_1)/2$$

ONE OR TWO CONSECUTIVE REJECTED POINTS

A SIXTH POINT IS PREDICTED FROM THE FOUR ESTABLISHED POINTS, IF ACCEPTED, THE FIFTH POINT IS INTERPOLATED AND A RETURN IS MADE TO PREVIOUS EDITING.

$$X_6 = X_4 + X_3 - X_1, \quad X_5 = (X_6 + X_4)/2$$

IF THE SIXTH POINT IS REJECTED AND NO MORE DATA IS AVAILABLE, A TRANSFER TO TEST FOR OTHER UNEDITED COORDINATES IS MADE. IF MORE DATA IS AVAILABLE, THE SEVENTH POINT IS PREDICTED FROM THE FOUR POINTS. IF THE SEVENTH IS ACCEPTED, BOTH THE FIFTH AND SIXTH POINTS ARE INTERPOLATED AND RETURN THEN MADE TO PREVIOUS EDITING. IN THIS CASE,

$$X_7 = 2X_4 - X_1$$
$$X_5 = (X_7 + X_4 - X_3 + X_2)/2$$
$$X_6 = (X_7 + X_4 + X_3 - X_2)/2$$

IF THE SEVENTH POINT IS REJECTED, A TRANSFER IS MADE TO THREE OR MORE CONSECUTIVE REJECTED POINTS.

THREE OR MORE CONSECUTIVE REJECTED POINTS

IN THIS CASE, THREE OR MORE POINTS MAY ACTUALLY BE IN ERROR, OR THE TRAJECTORY HAS BEEN DISTINCTLY ALTERED AND THESE POINTS LIE ON THIS TRAJECTORY. IF THERE IS NO DATA, A TRANSFER TO TEST FOR OTHER UNEDITED COORDINATES IS MADE. IF THERE IS MORE DATA, SUCCESSIVE SETS OF FOUR POINTS ARE AGAIN USED TO ESTABLISH A STRAIGHT LINE, AS IN NO PREVIOUS EDITING, UNTIL A SET OF FOUR ACCEPTABLE POINTS IS FOUND, DATA IS EXHAUSTED, OR A PRESET MAXIMUM NUMBER OF ATTEMPTS TO FIND SUCH A SET HAS BEEN EXCEEDED. IF AN ACCEPTABLE SET IS FOUND THE AVERAGE RATE OF CHANGE IS CHECKED AGAINST THE ONE SAVED UNDER ''PREVIOUS EDITING.'' IF THEY ARE NOT WITHIN A PRESCRIBED TOLERANCE THE SET IS REGARDED AS UNACCEPTABLE. IF THE SET IS ACCEPTABLE AND SMOOTHING IS TO BE USED THE MISSING POINTS ARE INTERPOLATED FOR ANC PLACED IN THE TABLE) MINUS ONES ARE PLACED IN THE TABLE IF SHORT ARC IS TO BE USED. IF THE DATA IS EXHAUSTED BEFORE A SATISFACTORY SET IS FOUND, A TRANSFER IS MADE TO TEST FOR @THER UNEDITED COORDINATES. IF THE PRESET MAXIMUM NUMBER OF ATTEMPTS IS EXCEEDED (PRESENTLY 10) AND THE SHORT ARC PROGRAM (CASARC) IS TO BE USED TO FURTHER PROCESS THE DATA, THE TABLE OF EDIT PARAMETERS (MHEDF OR TMHEDV IS RESET TO SHOW THE OBSERVATION CONTAINING THE OFFENDING COORDINATES AS THE FIRST EDITED OBSERVATION. EDITING OF THIS COORDINATE THEN RESUMES BY A TRANSFER TO NO PREVIOUS EDITING (IF THE DATA TABLE CONTAINS FOUR OR MORE ENTRIES OF THIS COORDINATE WHICH HAVE BEEN EDITED) OR BY A TRANSFER TO TEST FOR OTHER UNEDITED COORDINATES (IF FEWER THAN FOUR ENTRIES REMAIN).

IF THE PRESET MAXIMUM NUMBER OF ATTEMPTS IS EXCEEDED AND THE GENERAL LEAST SQUARES SMOOTHING PROGRAM (CSOGLS) IS TO BE USED TO FURTHER PROCESS THE DATA, THE SIGN OF MCESTL IS EXAMINED TO DETERMINE WHETHER AN EXCESSIVE NUMBER OF BAD POINTS WAS ALSO FOUND ON THE PREVIOUS ENTRY. IF THIS IS THE FIRST TIME AN EXCESSIVE NUMBER OF BAD POINTS HAS BEEN FOUND, THE BAD POINTS INDICATOR IS SET AND A TRANSFER IS MADE TO TEST FOR OTHER UNEDITED COORDINATES. IF THIS IS NOT THE FIRST TIME AN EXCESSIVE NUMBER OF BAD POINTS HAS BEEN FOUND (I.E., THIS ENTRY WAS TO TRY TO REESTABLISH THE FIT), THE TABLE OF EDIT PARAMETERS TMHEDF OR TMHEDV IS RESET TO SHOW THE LAST OBSERVATION CONTAINING THE OFFENDING COORDINATE AS THE FIRST EDITED OBSERVATION. TRANSFER IS THEN MADE TO TEST FOR OTHER UNEDITED COORDINATES. THE REQUIRED LINEAR, ALMOST-LEAST- SQUARES FORMULAS ARE -

$$X_{i+k-1} = (X_{i+k} + X_{i+k-2})/2$$

$$X_{i+k} = (X_{i+k+1} + X_{i+k-1})/2$$

$$X_{i+N} = X_i + N\,(X_{i+k-2} - X_i)/k-2 \qquad N=1,2,3,\ldots$$

TEST FOR OTHER UNEDITED COORDINATES

UPON COMPLETING THE EDITING OF THE RANGE OR AZIMUTH COORDINATES, THE h AND i PARAMETER LOCATIONS ARE UPDATED AND A RETURN MADE TO PROCESS THE REMAINING COORDINATES. UPON COMPLETION OF THE EDITING PROCEDURE, A TRANSFER IS MADE TO THE BAD POINTS SUFFIX, MFEBAD, IF FOR THE FIRST TIME, THE MAXIMUM NUMBER OF BAD POINTS ALLOWED HAS BEEN EXCEEDED AND THE GENERAL LEAST SQUARES SMOOTHING PROGRAM IS TO BE USED. OTHERWISE, TRANSFER IS MADE TO THE NORMAL SUFFIX, MFEOBD.

USAGE

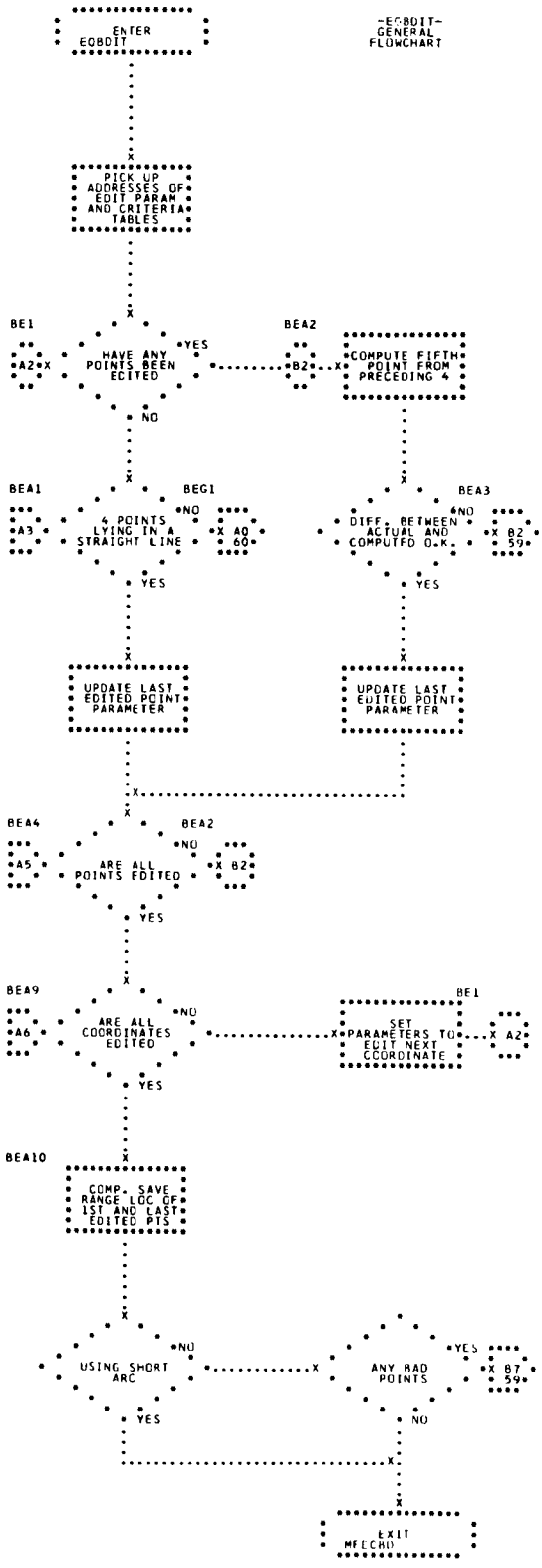
EOBDIT INITIALLY RECEIVES CONTROL FROM ITS PREFIX, MPEOBD, AS SOON AS THE LATTER HAS DETERMINED (1) THAT THE TIME FOR ACTIVATION OF THE BERMUDA HIGH-SPEED LINES HAS OCCURRED AND (2) THAT AT LEAST FOUR UNEDITED POINTS EXIST IN THE TCBDF6 (OR TCBDVE) TABLE. THEREAFTER, CONTROL IS GIVEN TO EOBDIT BY MPEOBD AT 1.0 SECOND INTERVALS.

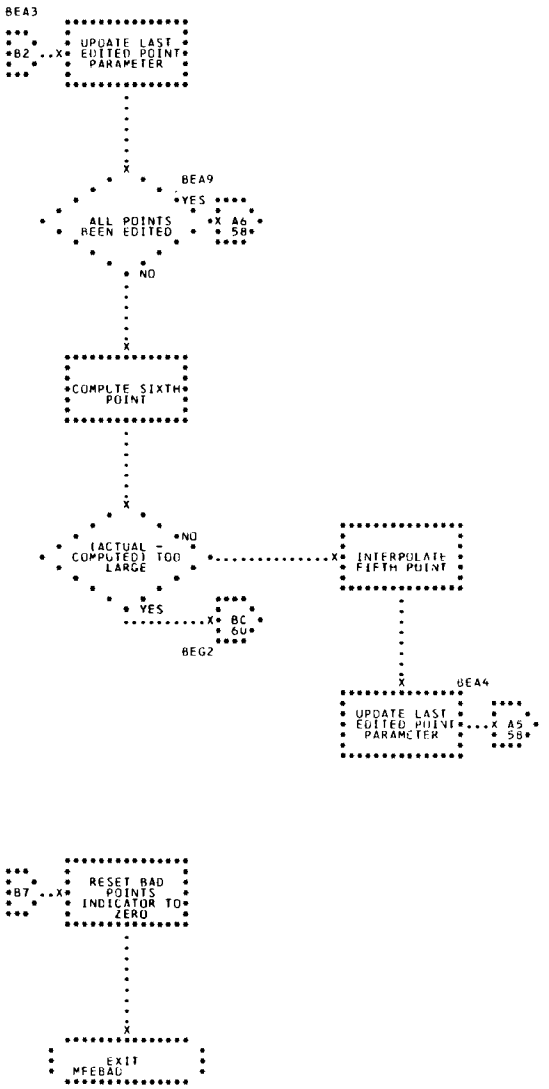
A) STORAGE REQUIRED -

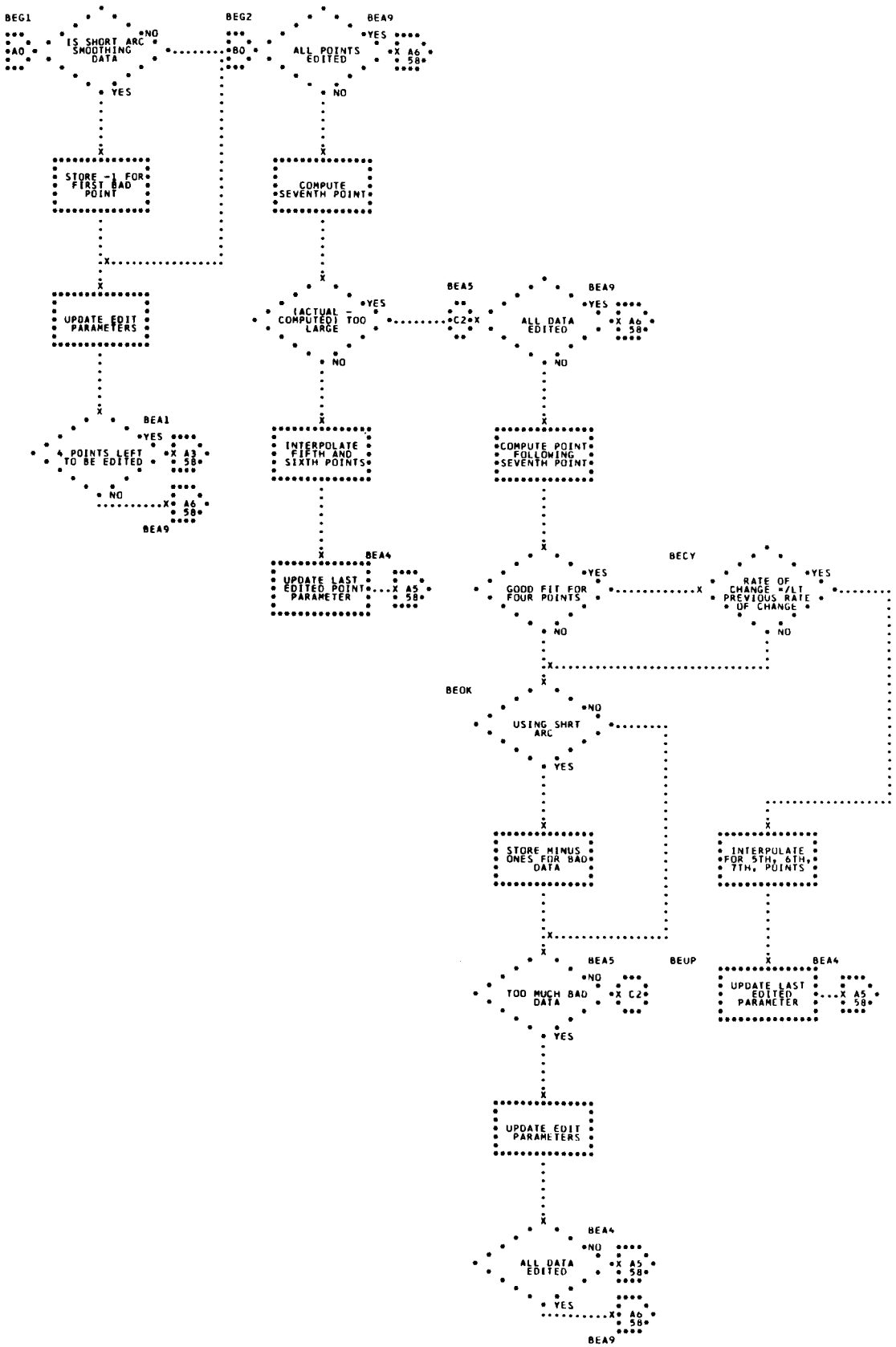
263 LOCATIONS

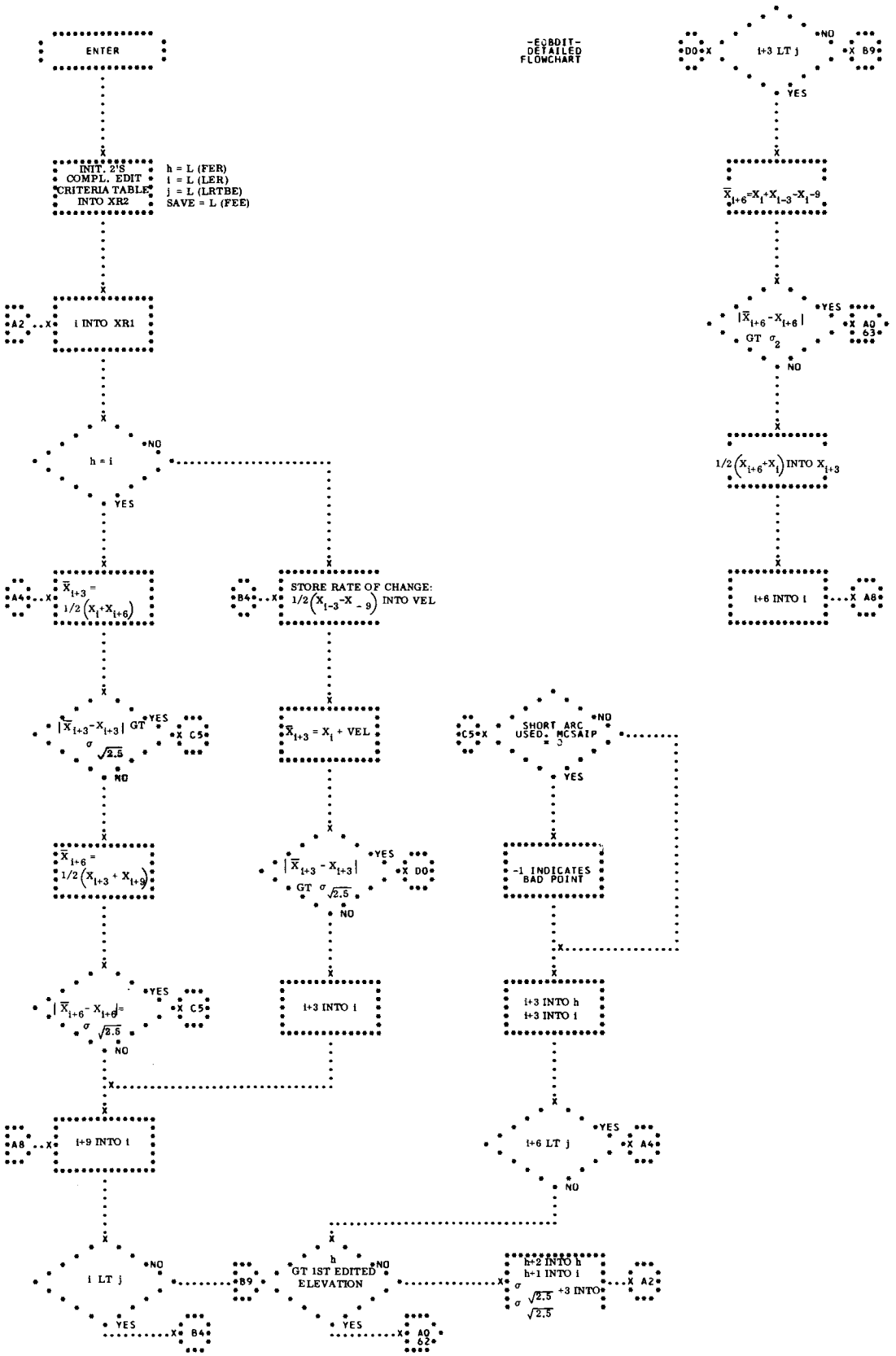
B) TIMING (7094) -

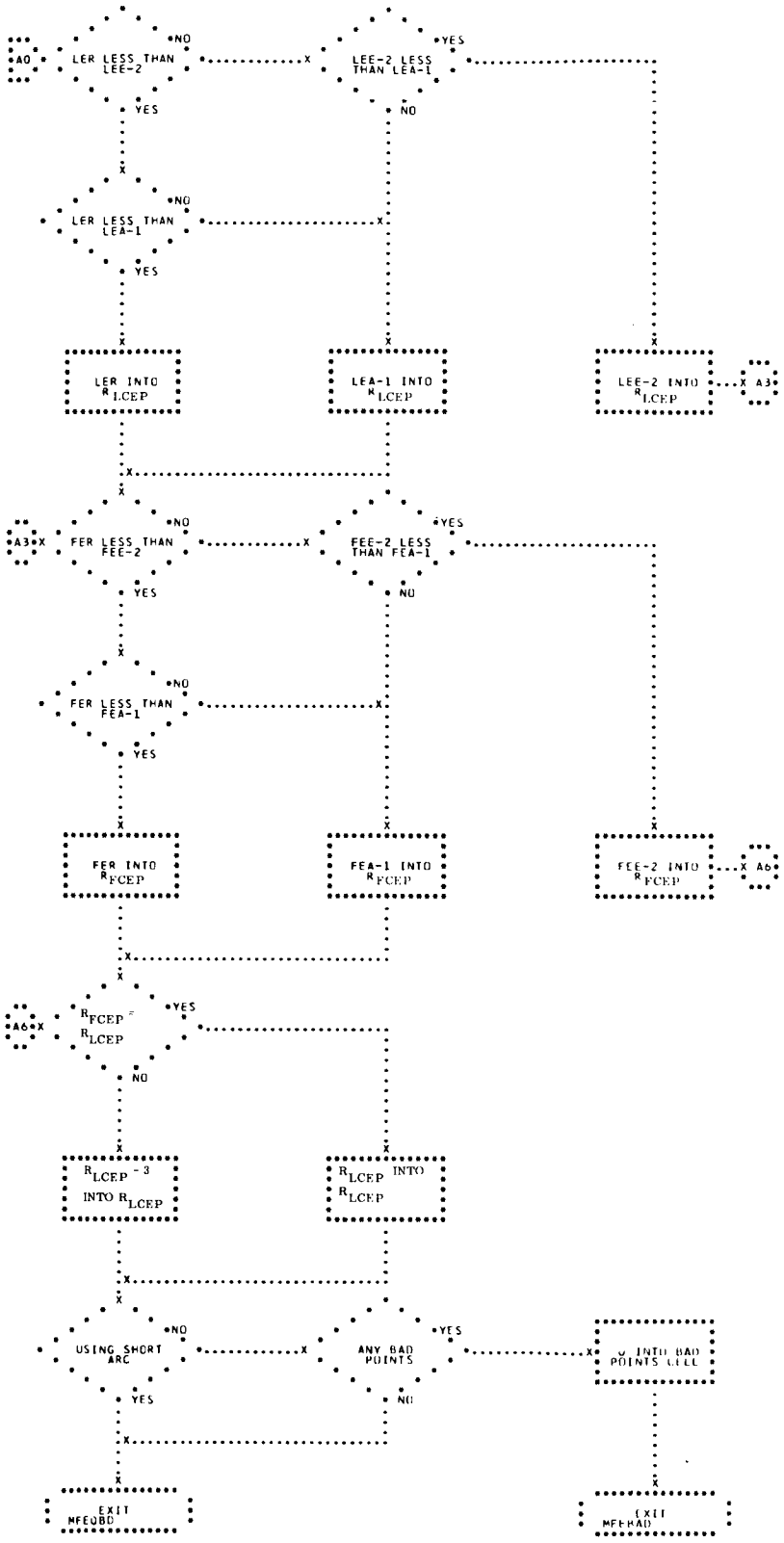
NO PREVIOUS EDITING,	10 GOOD POINTS-	1.209 MSEC.
PREVIOUS EDITING,	10 GOOD POINTS-	1.693 MSEC.
PREVIOUS EDITING,	1 BAD POINT -	
	9 GOOD POINTS-	1.953 MSEC.
PREVIOUS EDITING,	2 BAD POINTS-	
	8 GOOD POINTS-	1.678 MSEC.
PREVIOUS EDITING,	MORE THAN 10	
	BAD POINTS-	2.017 MSEC.

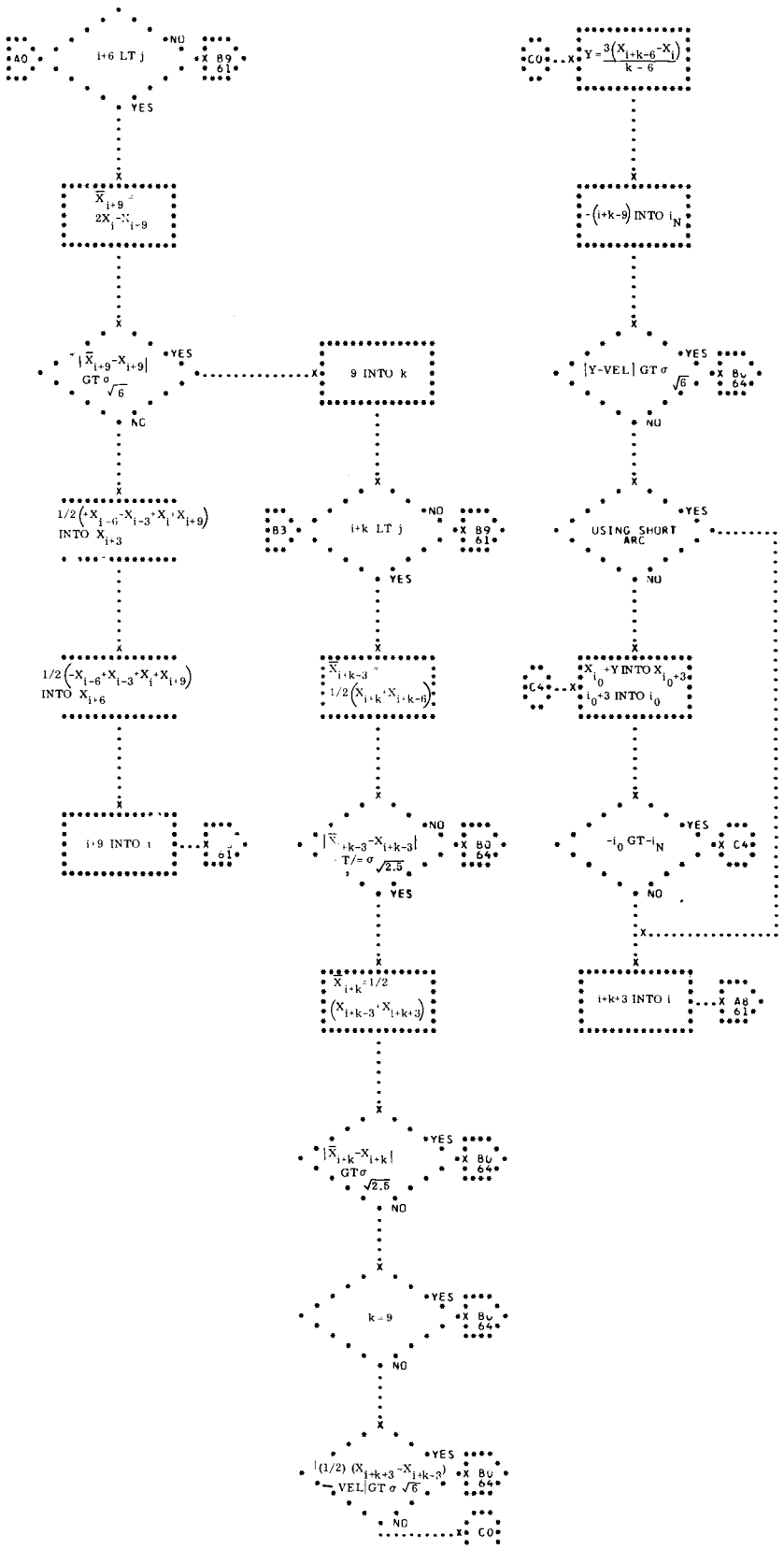

ENTER EGBDIT
-EGBDIT- GENERAL FLOWCHART
PICK UP ADDRESSES OF EDIT PARAM AND CRITERIA TABLES
BE1
HAVE ANY POINTS BEEN EDITED
YES
A2
B2
NO
BEA2
COMPUTE FIFTH POINT FROM PRECEDING 4
BEA1
4 POINTS LYING IN A STRAIGHT LINE
BEG1
NO
A3
A0
60
YES
BEA3
DIFF. BETWEEN ACTUAL AND COMPUTED O.K.
NO
B2
59
YES
UPDATE LAST EDITED POINT PARAMETER
UPDATE LAST EDITED POINT PARAMETER
BEA4
ARE ALL POINTS EDITED
BEA2
NO
A5
B2
YES
BEA9
ARE ALL COORDINATES EDITED
NO
A6
SET PARAMETERS TO EDIT NEXT COORDINATE
BE1
A2
YES
BEA10
COMP. SAVE RANGE LOC OF 1ST AND LAST EDITED PTS
USING SHORT ARC
NO
YES
ANY BAD POINTS
YES
B7
59
NO
EXIT MFECHO

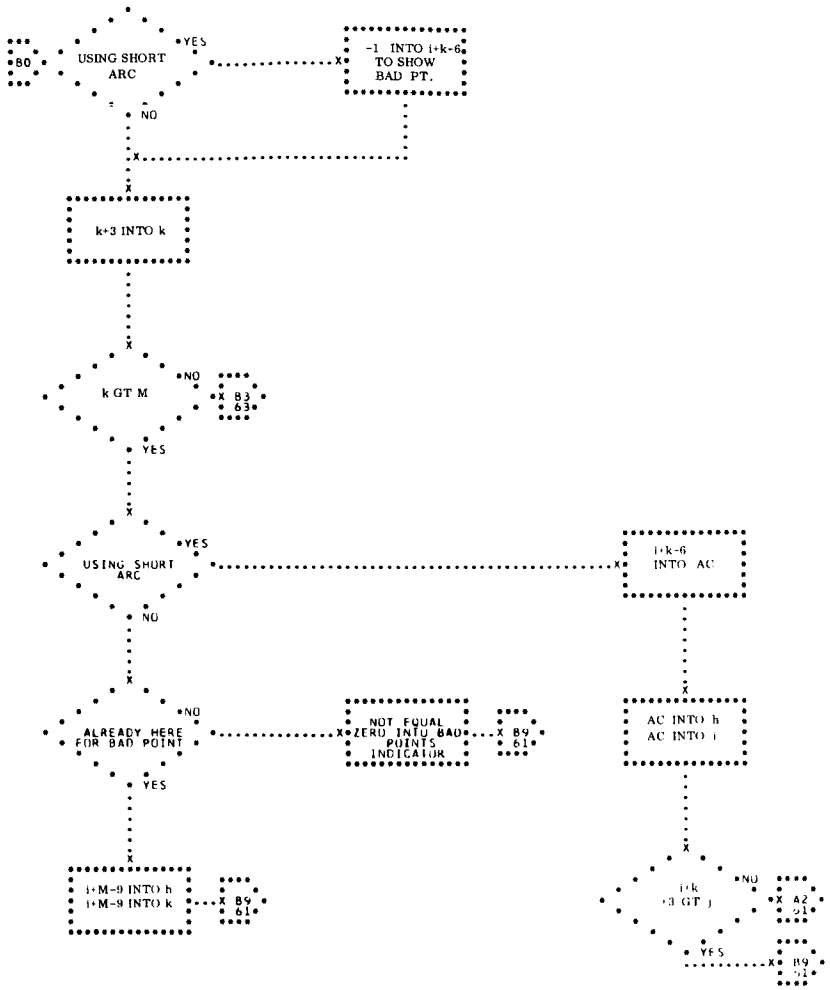

BEA3
UPDATE LAST EDITED POINT PARAMETER
B2
BEA9
YES
ALL POINTS BEEN EDITED
A6
5B
NO
COMPUTE SIXTH POINT
(ACTUAL - COMPUTED) TOO LARGE
NO
INTERPOLATE FIFTH POINT
YES
8C
6U
BEG2
BEA4
UPDATE LAST EDITED POINT PARAMETER
A5
5B
B7
RESET BAD POINTS INDICATOR TO ZERO
EXIT MFEBAD


BEG1
A0
IS SHORT ARC SMOOTHING DATA
NO
YES
STORE -1 FOR FIRST BAD POINT
BEG2
B0
UPDATE EDIT PARAMETERS
BEA1
4 POINTS LEFT TO BE EDITED
YES
X A3 58
NO
X A6 58
BEA9
BEA9
ALL POINTS EDITED
YES
X A6 58
NO
COMPUTE SEVENTH POINT
(ACTUAL - COMPUTED) TOO LARGE
YES
C2
NO
INTERPOLATE FIFTH AND SIXTH POINTS
BEA4
UPDATE LAST EDITED POINT PARAMETER
X A5 58
BEA5
BEA9
ALL DATA EDITED
YES
X A6 58
NO
COMPUTE POINT FOLLOWING SEVENTH POINT
BECY
GOOD FIT FOR FOUR POINTS
YES
RATE OF CHANGE =/LT PREVIOUS RATE OF CHANGE
YES
NO
NO
BEOK
USING SHRT ARC
NO
YES
STORE MINUS ONES FOR BAD DATA
INTERPOLATE FOR 5TH, 6TH, 7TH, POINTS
BEUP
BEA5
TOO MUCH BAD DATA
NO
X C2
YES
BEA4
UPDATE LAST EDITED PARAMETER
X A5 58
UPDATE EDIT PARAMETERS
BEA4
ALL DATA EDITED
NO
X A5 58
YES
X A6 58
BEA9


ENTER
INIT. 2'S COMPL. EDIT CRITERIA TABLE INTO XR2
h = L (FER)
i = L (LER)
j = L (LRTBE)
SAVE = L (FEE)
i INTO XR1
A2
h = i
NO
YES
$\overline{X}_{i+3} = 1/2\left(X_i + X_{i+6}\right)$
A4
STORE RATE OF CHANGE: $1/2\left(X_{i-3} - X_{-9}\right)$ INTO VEL
B4
$|\overline{X}_{i+3} - X_{i+3}|$ GT $\sigma\sqrt{2.5}$
YES
C5
NO
$\overline{X}_{i+3} = X_i + VEL$
$\overline{X}_{i+6} = 1/2\left(X_{i+3} + X_{i+9}\right)$
$|\overline{X}_{i+3} - X_{i+3}|$ GT $\sigma\sqrt{2.5}$
YES
D0
NO
$|\overline{X}_{i+6} - X_{i+6}| = \sigma\sqrt{2.5}$
i+3 INTO i
i+9 INTO i
A8
i LT j
B9
h GT 1ST EDITED ELEVATION
A0 62
-EQBDIT- DETAILED FLOWCHART
SHORT ARC USED. MCSAIP = 0
-1 INDICATES BAD POINT
i+3 INTO h
i+3 INTO i
i+6 LT j
h+2 INTO h
h+1 INTO i
$\sigma\sqrt{2.5}$ +3 INTO $\sigma\sqrt{2.5}$
$\overline{X}_{i+6} = X_i + X_{i-3} - X_{i-9}$
i+3 LT j
$|\overline{X}_{i+6} - X_{i+6}|$ GT σ_2
A0 63
$1/2\left(X_{i+6} + X_i\right)$ INTO X_{i+3}
i+6 INTO i
A8


A0
LER LESS THAN LEE-2
NO
YES
LEE-2 LESS THAN LEA-1
YES
NO
LER LESS THAN LEA-1
NO
YES
LER INTO R LCEP
LEA-1 INTO R LCEP
LEE-2 INTO R LCEP
A3
A3
FER LESS THAN FEE-2
NO
YES
FEE-2 LESS THAN FEA-1
YES
NO
FER LESS THAN FEA-1
NO
YES
FER INTO R FCEP
FEA-1 INTO R FCEP
FEE-2 INTO R FCEP
A6
A6
R FCEP = R LCEP
YES
NO
R LCEP - 3 INTO R LCEP
R LCEP INTO R LCEP
USING SHORT ARC
NO
YES
ANY BAD POINTS
YES
NO
INTO BAD POINTS CELL
EXIT MFEQBD
EXIT MFEBAD


A0
i+6 LT j
NO
X B9 61
YES
$\bar{X}_{i+9} = 2X_i - X_{i-9}$
$|\bar{X}_{i+9} - X_{i+9}|$ GT $\sigma\sqrt{6}$
YES
9 INTO k
NO
$1/2\left(+X_{i-6} - X_{i-3} + X_i + X_{i+9}\right)$ INTO X_{i+3}
B3
i+k LT j
NO
X B9 61
YES
$1/2\left(-X_{i-6} + X_{i-3} + X_i + X_{i+9}\right)$ INTO X_{i+6}
$\bar{X}_{i+k-3} = 1/2\left(X_{i+k} + X_{i+k-6}\right)$
i+9 INTO i
X A1
$|\bar{X}_{i+k-3} - X_{i+k-3}|$ $T/=\sigma\sqrt{2.5}$
NO
X B0 64
YES
$\bar{X}_{i+k} = 1/2\left(X_{i+k-3} + X_{i+k+3}\right)$
$|\bar{X}_{i+k} - X_{i+k}|$ GT $\sigma\sqrt{2.5}$
YES
X B0 64
NO
k = 9
YES
X B0 64
NO
$|(1/2)(X_{i+k+3} - X_{i+k-3})$ VEL$|$ GT $\sigma\sqrt{6}$
YES
X B0 64
NO
X C0
C0
$Y = \frac{3(X_{i+k-6} - X_i)}{k-6}$
$-(i+k-9)$ INTO i_N
$|Y\text{-VEL}|$ GT $\sigma\sqrt{6}$
YES
X B0 64
NO
USING SHORT ARC
YES
NO
C4
$X_{i_0} + Y$ INTO X_{i_0+3}
i_0+3 INTO i_0
$-i_0$ GT $-i_N$
YES
X C4
NO
i+k+3 INTO i
X A8 61


BO
USING SHORT ARC
YES
NO
-1 INTO i+k-6 TO SHOW BAD PT.
k+3 INTO k
k GT M
NO
X B3 63
YES
USING SHORT ARC
YES
NO
i+k-6 INTO AC
ALREADY HERE FOR BAD POINT
NO
YES
NOT EQUAL ZERO INTO BAD POINTS INDICATOR
X B9 61
AC INTO h
AC INTO i
i+M-9 INTO h
i+M-9 INTO k
X B9 61
i+k +3 GT j
NO
X A2 21
YES
X B9 21

16. MFEBAD - MONITOR BAD POINTS SUFFIX TO EOBDIT

THIS PROGRAM REINITIALIZES THE EDIT AND SMOOTHING PARAMETERS WHEN AN EXCESSIVE NUMBER OF BAD RADAR POINTS IS ENCOUNTERED. (SEE FLOWCHART.)

INPUT

A) TABLE -

TMHEDF (TMHEDV)	A 32-LOCATION TABLE CONTAINING THE EDIT, SMOOTHING, AND SHORT ARC PARAMETERS

B) CONSTANTS -

K00001	DEC 1 FIXED POINT ONE
K00009	DEC 9 FIXED POINT NINE

C) COMMUNICATION CELL -

MCERAD	ZERO IF FPS-16 DATA JUST EDITED, NON-ZERO IF VERLORT, P_e

OUTPUT

A) TABLES -

TMHEDF (TMHEDV)	A 32-LOCATION PARAMETER TABLE RE-INITIALIZED AS THE RESULT OF BAD POINTS
TCFSPU+1 (TCVSPT+1)	THE SECOND OF TWELVE CELLS IN THE SMOOTHING PARAMETERS TABLE. RESET TO ZERO INDICATING A FIRST OR RE-START ENTRY TO SMOOTHING, e_{smf} (OR e_{smv})
TCFSWT+1 (TCVSWT+1)	THE SECOND OF FIFTY-TWO CELLS IN THE SMOOTHING WORKING TABLE. RESET TO ZERO. THIS CELL CONTAINS THE MID-POINT OF THE LAST SMOOTHED SPAN OF

DATA, $L(X_0)_f$ (OR $L(X_0)_v$)

B) COMMUNICATION CELL -

MCINFF (MCINFV) SET NON-ZERO TO SHOW DATA MUST BE EDITED FOR A NEW FIT, i_{nff} (OR i_{nfv})

METHOD

MFEBAD IS ENTERED THE FIRST TIME A SET OF BAD RADAR DATA POINTS EXCEEDING THE MAXIMUM NUMBER IS ENCOUNTERED WHEN THE GENERAL LEAST SQUARES SMOOTHING PROGRAM (CSOGLS) IS TO BE USED TO PROCESS THE EDITED DATA.

AFTER DETERMINING THE SOURCE FROM WHICH THE BAD POINTS CAME, MFEBAD SETS THE INDICATOR MCINFF (MCINFV) TO SHOW THAT DATA FROM THE FPS-16 (VERLORT) RADAR MUST NOW BE EDITED FOR A NEW FIT. IT REINITIALIZES THE EDIT PARAMETERS FOR THE DATA AS FOLLOW -

FER = LER = LR - 3M - 9

FEA = LEA = LR - 3M - 8

FEE = LEE = LR - 3M - 7, WHERE M IS THE MAXIMUM ALLOWABLE NUMBER OF BAD POINTS.

NEXT, MFEBAD RESETS THE CELL CONTAINING THE MIDPOINT OF THE LAST SMOOTHED SPAN OF DATA TO ZERO SO THAT THE DATA TABLE MAY STILL BE MOVED TO ALLOW THE ENTRY OF NEW DATA. THE FIRST OR RESTART ENTRY INDICATOR IS ALSO SET FORCING THE SMOOTHING PROGRAM TO REINITIALIZE AT THE POINT OF GOOD DATA WHEN IT IS ENTERED AFTER THE NEW FIT HAS BEEN FOUND.

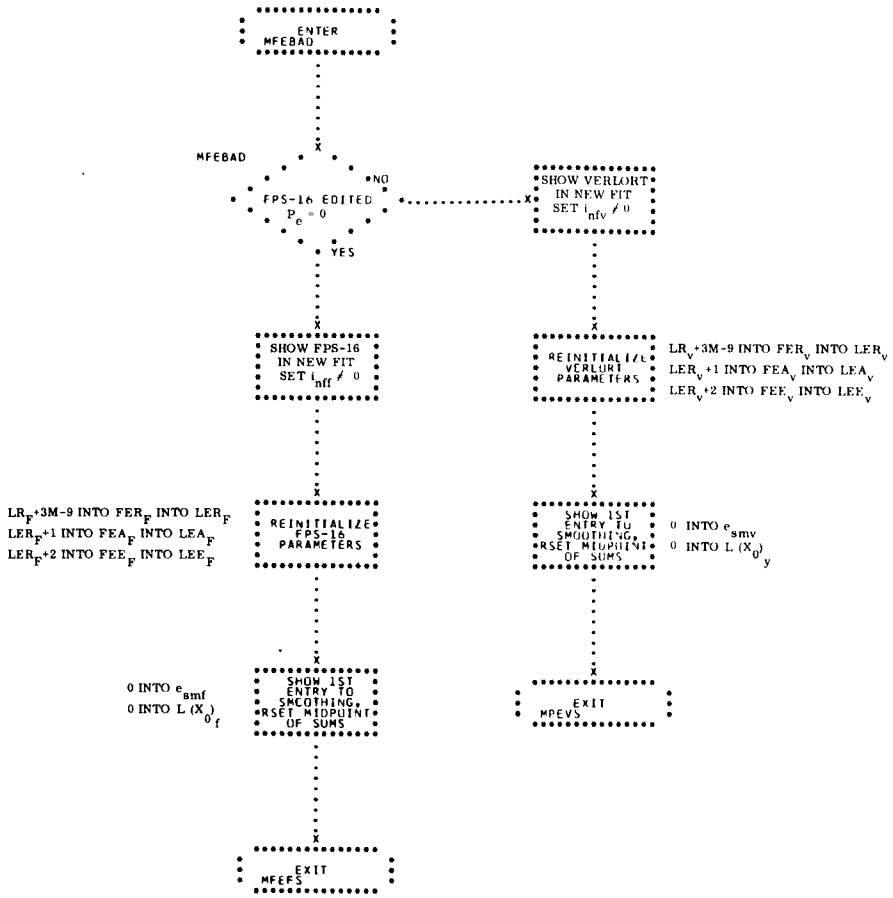
FINALLY, MFEBAD GIVES CONTROL TO MPEOBD FOR AN IMMEDIATE EDIT PASS ON THE SAME RADAR DATA IN AN ATTEMPT TO ESTABLISH THE NEW FIT.

USAGE

A) MFEBAD RECEIVES CONTROL FROM EOBDIT. IT TRANSFERS CONTROL TO MPEOBD.

B) STORAGE REQUIRED - 34 LOCATIONS

C) TIMING (7094) - .054 MILLISECONDS


ENTER MFEBAD
MFEBAD
FPS-16 EDITED $P_e = 0$
NO
YES
SHOW VERLORT IN NEW FIT SET $i_{nfv} \neq 0$
SHOW FPS-16 IN NEW FIT SET $i_{nff} \neq 0$
REINITIALIZE VERLORT PARAMETERS
LR_v+3M-9 INTO FER_v INTO LER_v
LER_v+1 INTO FEA_v INTO LEA_v
LER_v+2 INTO FEE_v INTO LEE_v
REINITIALIZE FPS-16 PARAMETERS
LR_F+3M-9 INTO FER_F INTO LER_F
LER_F+1 INTO FEA_F INTO LEA_F
LER_F+2 INTO FEE_F INTO LEE_F
SHOW 1ST ENTRY TO SMOOTHING, RSET MIDPOINT OF SUMS
0 INTO e_{smv}
0 INTO $L(X_0)_y$
SHOW 1ST ENTRY TO SMOOTHING, RSET MIDPOINT OF SUMS
0 INTO e_{smf}
0 INTO $L(X_0)_f$
EXIT MFEFS
EXIT MFEVS

17. MFEOBD - MONITOR SUFFIX TO EOBDIT

MFEOBD DETERMINES WHETHER BOTH DATA SOURCES HAVE BEEN PORCESSED BY EOBDIT AND WHETHER IT IS PERMISSIBLE TO ENTER ONE OF THE BERMUDA HIGH SPEED RADAR DATA PROCESSORS. (SEE FLOWCHART).

INPUT

A) TABLE -

TMHEDF (TMHEDV)	THE 32-LOCATION TABLE CONTAINING THE FPS-16 (VERLORT) INPUT, EDIT, SMOOTHING AND SHORT ARC PARAMETERS

B) COMMUNICATION CELLS -

MCERAD	CONTAINS INDICATION OF SOURCE OF DATA JUST PROCESSED BY EOBDIT- ZERO, IF DATA WAS FROM FPS-16 RADAR, NON-ZERO, IF DATA WAS FROM VERLORT RADAR, P_e
MCINFF (MCINFV)	NON-ZERO IF FPS-16 (VERLORT) DATA IS TO BE EDITED FOR A NEW FIT. ZERO OTHERWISE. (i_{nff} OR i_{nfv})
MCTAST	NON-ZERO IF AT LEAST ONE DATA TABLE HAS BEEN ESTABLISHED (I.E., HAS FOUR DATA POINTS IN IT). OTHERWISE, ZERO. (i_{sate})

OUTPUT

A) COMMUNICATION CELLS -

MCINFF (MCINFV)	RESET TO ZERO IF NEW FIT ESTABLISHED FOR FPS-16 (VERLORT) DATA, i_{nff} (OR i_{nfv})

METHOD

WHEN MFEOBD RECEIVES CONTROL, IT DETERMINES WHICH OF THE DATA SOURCES WAS BEING PROCESSED AND WHETHER THE DATA WAS BEING EDITED FOR A NEW FIT. IF THE DATA FROM THAT RADAR WAS GOOD, MFEOBD CONTINUES AS IN B).

A) IF THE RADAR DATA WAS BEING EDITED FOR A NEW FIT, MFEOBD CHECKS THE EDIT PARAMETERS TO DETERMINE WHETHER THE NEW FIT HAS BEEN FOUND.
IT HAS NOT BEEN FOUND IF FER=LER, FEA=LEA, OR FEE=LEE AND MFEOBD CONTINUES AS IN B).
IF FER≠LER, FEA≠LEA, AND FEE≠LEE, THE NEW FIT HAS BEEN FOUND AND MCINFF (OR MCINFV) IS RESET ACCORDINGLY.

B) SINCE EDIT HAS FINISHED PROCESSING THE DATA AND UPDATING THE NECESSARY PARAMETERS, THE DATA TABLE MOVE IS UNSUPPRESSED SO THAT OLD DATA MAY BE TAKEN OUT OF THE TABLE, IF NECESSARY.
IF FPS-16 DATA WAS EDITED ON THIS ENTRY, MFEOBD GOES BACK TO MPEOBD FOR AN EDITING PASS ON THE BERLORT DATA.

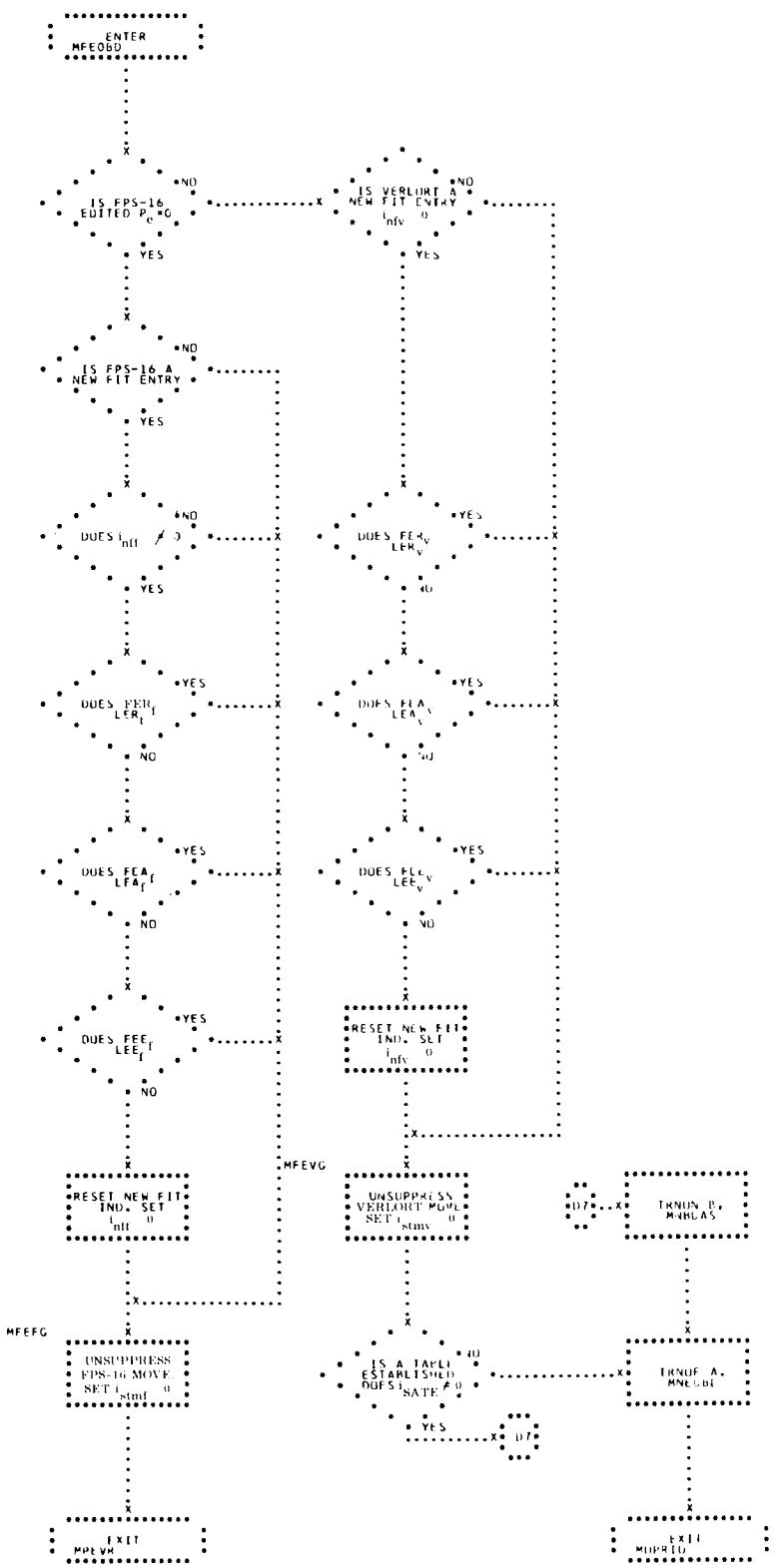
ONCE THE VERLORT DATA HAS BEEN EDITED, MFEOBD TRANSFERS CONTROL TO MOPRIO WITH ITS OWN ''IN PROCESS'' (A) BIT TURNED OFF AND, IF THE CONTENTS OF MCTAST INDICATE THAT AT LEAST ONE DATA TABLE IS ESTABLISHED, THE ''READY' (B) BIT TURNED ON FOR THE BERMUDA PROCESSORS, SHORT ARC AND SMOOTHING.

USAGE

MFEOBD IS ENTERED FROM EOBDIT AND TRANSFERS CONTROL EITHER TO MOPRIO OR TO MPEOBD.

A) STORAGE REQUIREMENTS -
36 LOCATIONS

B) TIMING (7094) - .012 MILLISECONDS, MINIMUM
.054 MILLISECONDS, MAXIMUM


ENTER MFEOBD
IS FPS-16 EDITED P_e = 0
NO
YES
IS VERLORT A NEW FIT ENTRY i_{nfv} 0
IS FPS-16 A NEW FIT ENTRY
DOES $i_{nff} \neq 0$
DOES FER_v LER_v
DOES FER_f LER_f
DOES FEA_v LEA_v
DOES FEA_f LFA_f
DOES FEE_v LEE_v
DOES FEE_f LEE_f
RESET NEW FIT IND. SET i_{nfv} 0
RESET NEW FIT IND. SET i_{nff} 0
MFEVG
UNSUPPRESS VERLORT MOVE SET i_{stmv} 0
D7
TRNON B, MNBLAS
MFEFG
UNSUPPRESS FPS-16 MOVE SET i_{stmf} 0
IS A TABLE ESTABLISHED DOES $i_{SATE} \neq 0$
TRNOF A, MNEOBI
EXIT MPEVR
EXIT MOPRID

18. MPBDAS - MONITOR PREFIX TO CSOGLS AND CASARC

THIS PROGRAM CONTROLS ENTRIES TO THE BERMUDA HIGH SPEED RADAR DATA PROCESSORS, CSOGLS (THE GENERAL SQUARES SMOOTHING PROGRAM) AND CASARC (THE SHORT ARC PROGRAM) WHICH PRODUCE THE POSITION AND VELOCITY VECTORS WITH THEIR ASSOCIATED TIME. (SEE FLOWCHART.)

INPUT

A) TABLES -

TCFSPT (OR TCVSPT)	THE 12-LOCATION PARAMETER TABLE USED BY CSOGLS IN PROCESSING THE FPS-16 (VERLORT) RADAR DATA
TCFSWT+1 (OR TCVSWT+1)	THE SECOND LOCATION OF THE FIFTY-TWO LOCATION SMOOTHING WORKING TABLE CONTAINING THE LOCATION OF THE FIRST COMPONENT OF THE MID-POINT OF THE U, V, AND W SUMS FOR FPS-16 (VERLORT) RADAR DATA, $L(X_0)_f$ (OR $L(X_0)_v$)
TMHEDF (OR TMHEDV)	THE 32-LOCATION PARAMETER TABLE CONTAINING THE INPUT, EDIT, SMOOTHING, AND SHORT ARC PARAMETERS FOR FPS-16 (VERLORT) RADAR DATA
TMSTCH+4 (OR TMSTCH)5	THE FIFTH (SIXTH) ENTRY IN THE TMSTCH TABLE CONTAINING THE LOCATION OF THE FPS-16 (VERLORT) STATION CHARACTERISTICS BLOCK - 1
TM8.3M	THE 8.3 MILLISECOND COUNT SINCE

THE LAST HALF-SECOND IN FIXED POINT, B35.

TSVRAF (OR TSVRAV) — A 12-LOCATION TABLE USED TO SAVE THE LAST FOUR OBSERVATION IN THE FPS-16 (VERLORT) RADAR DATA TABLE

TCBDF6 (OR TCBDVE) — A 601-LOCATION TABLE CONTAINING THE FPS-16 (VERLORT) RADAR OBSERVATIONS. THE FIRST CELL CONTAINS THE TIME ASSOCIATED WITH THE FIRST OBSERVATION IN THE TABLE IN FIXED POINT TENTHS OF SECONDS, B35

B) CONSTANTS -

K00001 DEC 1 FIXED POINT ONE

K00003 DEC 3 FIXED POINT THREE

KCH233 OCT 233000000000
CHARACTERISTIC OF 233 OCTAL

KCH234 OCT 234000000000
CHARACTERISTIC OF 234 OCTAL

K8.3M3 CONVERSION OF 8.3 MILLISECOND COUNT TO SECONDS

K005.0 DEC 5.0 FLOATING POINT FIVE

C. COMMUNICATION CELLS -

MCHFSC — CONTAINS CURRENT GMT IN FIXED POINT HALF-SECONDS, B35

MCINFF (OR MCINFV) — NON-ZERO IF FPS-16 (VERLORT) DATA BEING EDITED FOR A NEW FIT, ZERO OTHERWISE, i_{nff} (OR i_{nfv})

MCSAIP — NON-ZERO IF SHORT ARC PROGRAM IN PROCESS, ZERO OTHERWISE, i_{saip}

OUTPUT

A) TABLES -

TCBDF6 (OR TCBDVE)	THE 601-LOCATION TABLE CONTAINING FPS-16 (VERLORT) RADAR DATA WITH THE LAST POINTS CONVERTED TO THE PROPER FORMAT FOR USE BY THE SELECTED RADAR DATA PROCESSOR
TCFSPT+1 (OR TCVSPT+1)	THE SECOND LOCATION OF THE 12-LOCATION PARAMETER TABLE USED BY THE SMOOTHING PROGRAM IN PROCESSING FPS-16 (VERLORT) RADAR DATA, SET TO ZERO TO INDICATE A FIRST OR RESTART ENTRY UPON ERROR RETURN FROM THE LOCAL TO INERTIAL COORDINATE CONVERSION PROGRAM, e_{smf} (O
TSVRAF (OR TSVRAV)	THE 12 LOCATION TABLE CONTAINING THE LAST FOUR FPS-16 (VERLORT) RADAR OBSERVATIONS

B) COMMUNICATION CELLS -

MCINFF (OR MCINFV)	SET NON-ZERO ON AN ERROR RETURN FROM THE LOCAL TO INERTIAL COORDINATE CONVERSION PROGRAM TO SHOW THAT FPS-16 (VERLORT) DATA SHOULD BE EDITED FOR A NEW FIT, i_{nff} (OR i_{nfv}) CONTAINS CURRENT TIME
MCTIME	IN GMT FLOATING POINT TENTHS OF SECONDS, t_c

METHOD

WHEN CONTROL IS GIVEN TO MPBDAS, IT TURNS ON ITS A

BIT TO SHOW THAT IT IS NOW EXECUTING , TURNS OFF ITS B BIT (TO INDICATE THAT IT IS HANDLING THE LATEST DATA) AND TURNS ON THE E BIT OF THE EDIT PROGRAM, EOBDIT (TO PREVENT ENTRY TO EDIT UNTIL THE PROCESSORS HAVE BEEN EXECUTED).

IF THE SHORT ARC PROGRAM HAS BEEN SELECTED FOR USE ON THIS PASS, MPBDAS COMPUTED THE CURRENT TIME IN GMT FLOATING POINT TENTHS OF SECONDS, SUPPRESSES THE FPS-16 AND THE VERLORT DATA TABLE MOVES, AND TRANSFERS CONTROL TO THE SHORT ARC PROGRAM CASARC.

IF THE SMOOTHING PROGRAM IS TO BE USED ON THIS PASS AND BOTH THE FPS-16 AND BOTH THE FPS-16 AND THE VERLORT RADAR DATA ARE BEING EDITED FOR A NEW FIT, NO COMPUTATIONS CAN BE MADE UNTIL THE NEW FIT HAS BEEN FOUND. THEREFORE, MPBDAS TRANSFERS CONTROL TO ITS SUFFIX, MFBDAS.

IF THE SMOOTHING PROGRAM IS TO BE USED ON THIS PASS AND AT LEAST ONE OF THE RADAR DATA TYPES CAN BE PROCESSED, MPBDAS TAKES THE FOLLOWING STEPS -

A) ON A FIRST OR RESTART ENTRY TO THE SMOOTHING PROGRAM FOR THIS RADAR, MPBDAS DETERMINES WHETHER ANY EDITED OBSERVATIONS ARE AVAILABLE (I.E., WHETHER THE LOCATION OF THE RANGE COMPONENT OF THE LAST COMPLETELY EDITED POINT (LR) IS GREATER THAN THAT OF THE FIRST COMPLETELY EDITED POINT (FR). IF THERE IS NOT AT LEAST ONE EDITED POINT, MPBDAS CONTINUES AS IN D).

IF EDITED POINTS EXIST, HOWEVER, THE FR IS USED TO UPDATE THE CELLS CONTAINING THE LOCATIONS OF THE FIRST COMPONENTS OF (1) THE FIRST POINT TO BE CONVERTED (FRTC), (2) THE FIRST

CONVERTED POINT (FCR), AND, AFTER ADDING 3N, (3) THE MIDPOINT OF THE U, V, W SUMS ($L(X_0)$).

SIMILARLY, THE LR IS USED TO UPDATE THE CELLS CONTAINING THE LOCATIONS OF THE FIRST COMPONENT OF (1) THE LAST POINT TO BE CONVERTED (LRTC) AND (2) THE LAST CONVERTED POINT (LCR).

THEN, BASED ON THE LR, THE LAST FOUR RADAR OBSERVATIONS (r, a, e) ARE STORED IN THE SAVE TABLE FOR THIS RADAR. THIS ALLOWS MPBDAS TO CONVERT ALL OF THE LATEST OBSERVATIONS TO THE CORRECT FORMAT FOR USE BY THE SMOOTHING PROGRAM WHILE STILL MAINTAINING THE MINIMUM NUMBER OF RADAR OBSERVATIONS IN THEIR ORIGINAL FORMAT FOR SUBSEQUENT USE BY THE EDIT PROGRAM. MPBDAS THEN CONTINUES AS IN C.

B) ON ANY ENTRY OTHER THAN THE FIRST OR RESTART ENTRY, MPBDAS DETERMINES WHETHER ADDITIONAL EDITED POINTS ARE AVAILABLE (I.E., WHETHER LCR+3 (THE LOCATION OF THE FIRST COMPONENT OF THE POINT FOLLOWING THE LAST CONVERTED POINT) IS LESS THAN OR EQUAL TO LR). IF NOT, MPBDAS TRANSFERS TO D).

OTHERWISE, LCR+3 IS USED TO UPDATE FRTC. THEN BASED ON LCR+3 AND THE LR, THE CONTENTS OF THE SAVE BLOCK (CONTAINING THE LAST FOUR CONVERTED OBSERVATIONS FROM THE PREVIOUS PASS) ARE STORED INTO THE DATA TABLE IN THEIR CORRECT LOCATIONS AND THE LAST FOUR RADAR OBSERVATIONS (r, a, e) ARE PLACED IN THE SAVE BLOCK. THE LR IS USED TO UPDATE LRTC AND LCR AND CONTROL PASSES TO C).

C) THE SUBROUTINE CLSTOI IS ENTERED WITH THE LOCA-

TION OF THE DATA TABLE, THE LOCATION OF THE STATION CHARACTERISTICS TABLE, AND AN INDICATION OF WHICH POINTS IN THE TABLE ARE TO BE CONVERTED (I.E., FROM FRTC TO LRTC INCLUSIVE) TO THE FORMAT REQUIRED BY THE SMOOTHING PROGRAM. ON A NORMAL RETURN FROM THE CONVERSION ROUTINE, THE SMOOTHING PROGRAM, CSOGLS, IS ENTERED TO PRODUCE THE POSITION AND VELOCITY VECTORS AND THEIR ASSOCIATED TIME. WHEN THE SMOOTHING OPERATION IS FINISHED, THE LCR IS USED TO PLACE THE LAST FOUR CONVERTED POINTS IN THE SAVE TABLE AND TO PLACE THE CONTENTS OF THE SAVE TABLE (THE LAST FOUR RADAR OBSERVATIONS IN THEIR ORIGINAL FORMAT) BACK INTO THE DATA TABLE. THIS SWITCHING PRESERVES THE CONTINUITY OF BOTH THE LINE ESTABLISHED BY THE EDIT PROGRAM AND THE SOLUTION OBTAINED BY THE SMOOTHING PROGRAM. THE TABLE MOVE IS UNSUPPRESSED AND MPBDAS CONTINUES AS IN D).

ON AN ERROR RETURN FROM CLSTOI, THE AC CONTAINS THE ADDRESS OF THE FIRST COMPONENT OF THE POINT IN ERROR. THIS ADDRESS IS UPDATED TO THE NEXT POINT AND THE EDIT PARAMETER TABLE IS REINITIALIZED AS FOLLOWS -

C(AC) +3 = FER = LER

FER +1 = FEA = LEA

FER +2 = FEE = LEE

INDICATORS ARE THEN SET TO SHOW THAT THE DATA MUST BE EDITED FOR A NEW FIT AND THAT THE NEXT ENTRY TO SMOOTHING AFTER THE NEW FIT HAS BEEN FOUND SHOULD BE A RESTART ENTRY. MPBDAS THEN TRANSFERS TO D).

D) IF THE DATA JUST PROCESSED WAS FPS-16 DATA, THE TABLE MOVE FOR THE FPS-16 DATA TABLE IS UNSUPPRESSED AND MPBDAS RETURNS TO PROCESS THE VERLORT DATA.

IF VERLORT DATA HAS JUST BEEN PROCESSED, ITS TABLE MOVE IS UNSUPPRESSED AND CONTROL IS TRANSFERRED TO THE SUFFIX, MFBDAS.

USAGE

MPBDAS RECEIVES CONTROL FROM MCPRIO ONCE PER SECOND IT TRANSFERS CONTROL TO CSOGLS, CASARC, MFBDAS.

A) STORAGE REQUIRED -

207 LOCATIONS

B) TIMING (7094) -

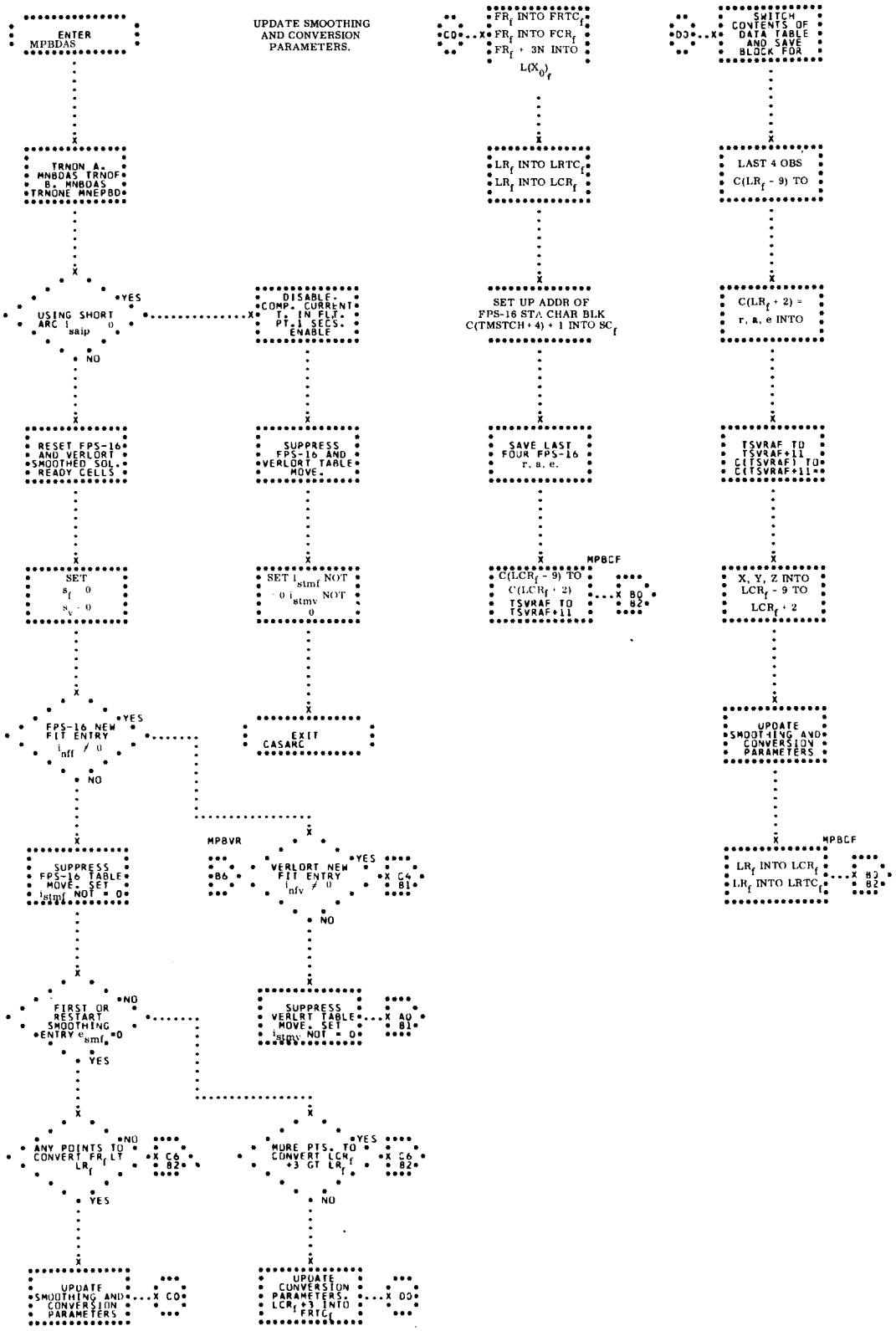
1) 0.134 MILLISECONDS FOR ENTRY TO CASARC
2) 50.528 + N(2.138) + P(1.494) MILLISECONDS FOR FIRST ENTRY TO CSOGLS FOR ONE RADAR
3) 1.670 + N(2.138) + P(1.494) MILLISECONDS FOR SUBSEQUENT ENTRY TO CSOGLS FOR ONE RADAR

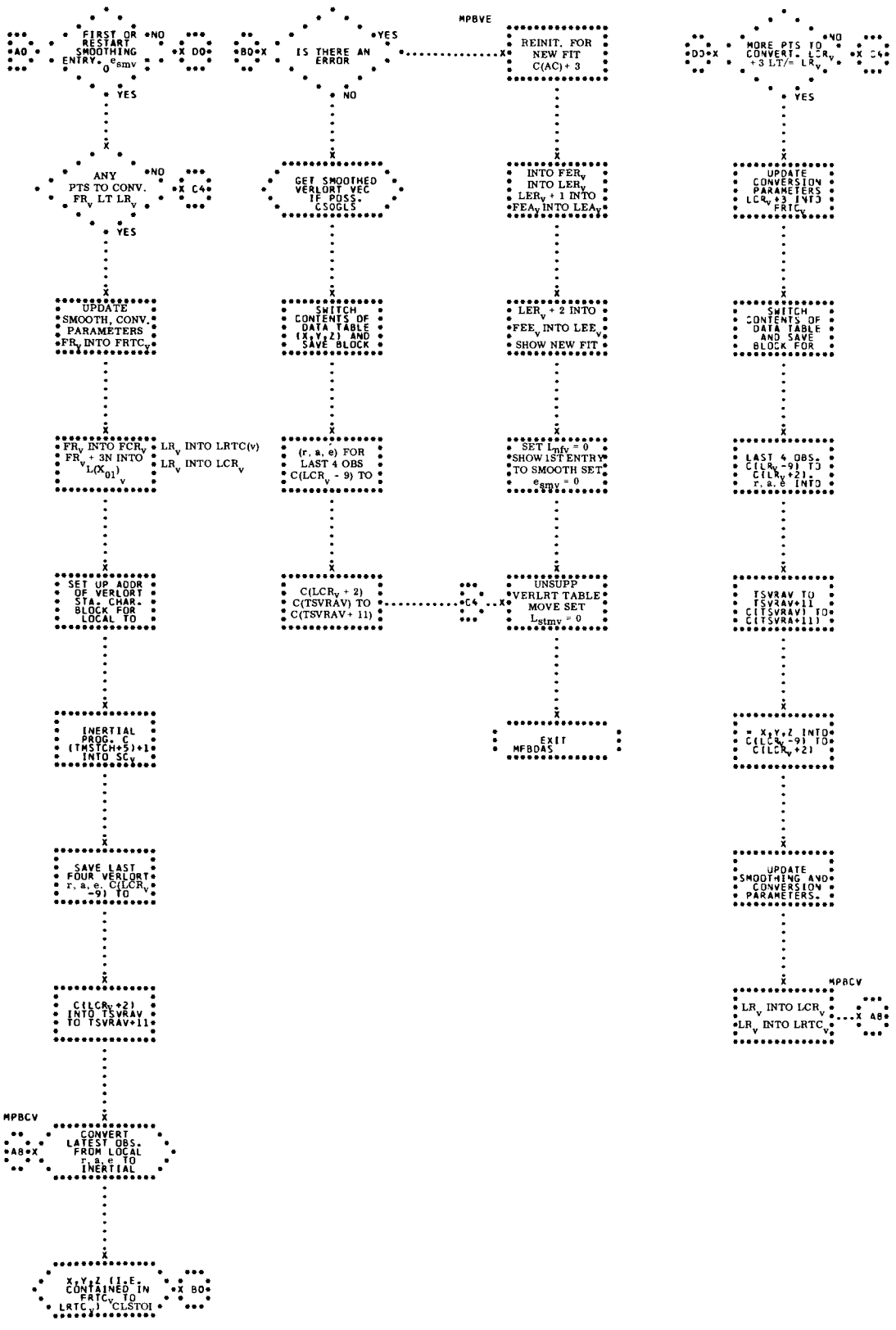
WHERE

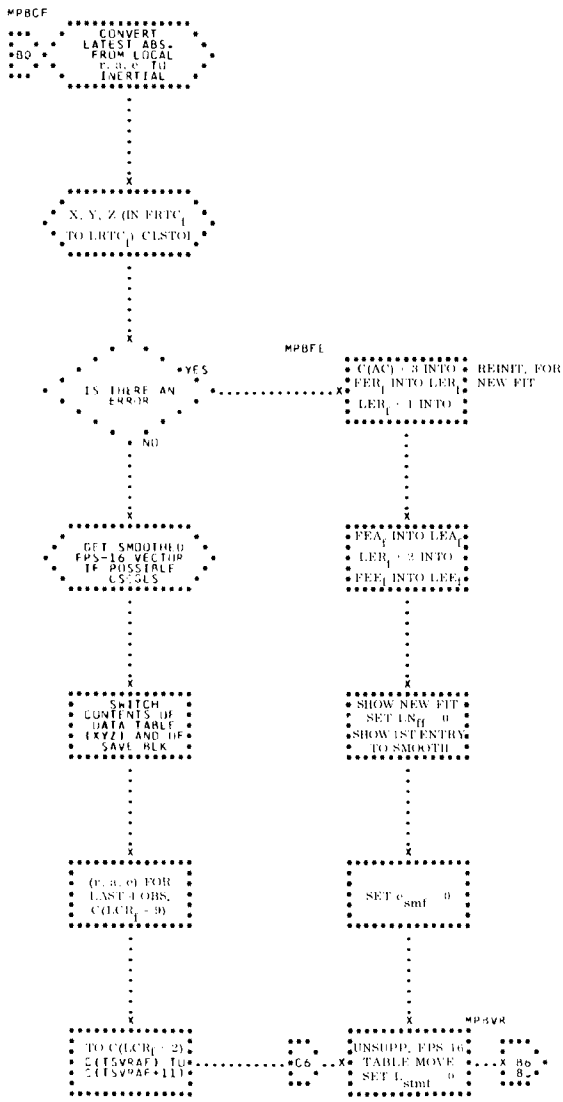
N = NUMBER OF POINTS CONVERTED

P = NUMBER OF POINTS TO BE ADDED TO U, V, W SUMS (P IS LESS THAN OR EQUAL TO N)

NOTE - ABOVE TIMES EXCLUDE TIMING FOR CASARC AND CSOGLS PROGRAMS


ENTER MPBDAS
TRNON A. MNBOAS TRNOF B. MNBOAS TRNONE MNEPBD
USING SHORT ARC $i_{saip} \neq 0$
YES
NO
RESET FPS-16 AND VERLORT SMOOTHED SOL. READY CELLS
SET $s_f = 0$, $s_v = 0$
FPS-16 NEW FIT ENTRY $i_{nff} \neq 0$
SUPPRESS FPS-16 TABLE MOVE. SET i_{stmf} NOT = 0
FIRST OR RESTART SMOOTHING ENTRY $e_{smf} = 0$
ANY POINTS TO CONVERT FR_f LT LR_f
UPDATE SMOOTHING AND CONVERSION PARAMETERS
UPDATE SMOOTHING AND CONVERSION PARAMETERS.
DISABLE COMP. CURRENT T. IN FLT. PT.-1 SECS. ENABLE
SUPPRESS FPS-16 AND VERLORT TABLE MOVE.
SET i_{stmf} NOT $= 0$, i_{stmv} NOT $= 0$
EXIT CASARC
MPBVR
VERLORT NEW FIT ENTRY $i_{nfv} \neq 0$
SUPPRESS VERLRT TABLE MOVE. SET i_{stmv} NOT = 0
MORE PTS. TO CONVERT LCR_f +3 GT LR_f
UPDATE CONVERSION PARAMETERS. LCR_f +3 INTO FRTC
FR_f INTO $FRTC_f$
FR_f INTO FCR_f
FR_f + 3N INTO $L(X_0)_f$
LR_f INTO $LRTC_f$
LR_f INTO LCR_f
SET UP ADDR OF FPS-16 STA CHAR BLK C(TMSTCH + 4) + 1 INTO SC_f
SAVE LAST FOUR FPS-16 r, a, e.
$C(LCR_f - 9)$ TO $C(LCR_f + 2)$ TSVRAF TO TSVRAF+11
MPBCF
SWITCH CONTENTS OF DATA TABLE AND SAVE BLOCK FOR
LAST 4 OBS $C(LR_f - 9)$ TO
$C(LR_f + 2)$ = r, a, e INTO
TSVRAF TO TSVRAF+11 C(TSVRAF) TO C(TSVRAF+11
X, Y, Z INTO $LCR_f - 9$ TO $LCR_f + 2$
UPDATE SMOOTHING AND CONVERSION PARAMETERS
MPBCF
LR_f INTO LCR_f
LR_f INTO $LRTC_f$


FIRST OR RESTART SMOOTHING ENTRY, $e_{smv} = 0$
NO
YES
A0
D0
ANY PTS TO CONV. FR_v LT LR_v
NO
YES
C4
UPDATE SMOOTH, CONV. PARAMETERS FR_v INTO $FRTC_v$
FR_v INTO FCR_v
FR_v + 3N INTO $L(X_{01})_v$
LR_v INTO LRTC(v)
LR_v INTO LCR_v
SET UP ADDR OF VERLORT STA. CHAR. BLOCK FOR LOCAL TO
INERTIAL PROG. C (TMSTCH+5)+1 INTO SC_v
SAVE LAST FOUR VERLORT r, a, e, $C(LCR_v$ -9) TO
$C(LCR_v+2)$ INTO TSVRAV, TO TSVRAV+11
MPBCV
A8
CONVERT LATEST OBS. FROM LOCAL r, a, e TO INERTIAL
$X_1Y_1Z_1$ (i.e. CONTAINED IN $FRTC_v$ TO $LRTC_v$) ^{CLSTOI}
B0
IS THERE AN ERROR
YES
NO
GET SMOOTHED VERLORT VEC IF POSS. CSOGLS
SWITCH CONTENTS OF DATA TABLE (X,Y,Z) AND SAVE BLOCK
(r, a, e) FOR LAST 4 OBS $C(LCR_v$ - 9) TO
$C(LCR_v + 2)$ C(TSVRAV) TO C(TSVRAV+ 11)
C4
MPBVE
REINIT. FOR NEW FIT C(AC)+3
INTO FER_v INTO LER_v LER_v + 1 INTO FEA_v INTO LEA_v
LER_v + 2 INTO FEE_v INTO LEE_v SHOW NEW FIT
SET $L_{nfv} = 0$ SHOW 1ST ENTRY TO SMOOTH SET, $e_{smv} = 0$
UNSUPP VERLRT TABLE MOVE SET $L_{stmv} = 0$
EXIT MFBDAS
D3
MORE PTS TO CONVERT, LCR_v +3 LT/= LR_v
NO
YES
C4
UPDATE CONVERSION PARAMETERS LCR_v +3 INTO $FRTC_v$
SWITCH CONTENTS OF DATA TABLE AND SAVE BLOCK FOR
LAST 4 OBS. $C(LR_v$ -9) TO $C(LR_v+2)$, r, a, e INTO
TSVRAV TO TSVRAV+11 C(TSVRAV) TO C(TSVRA+11)
= X_1Y_1Z INTO $C(LCR_v$ -9) TO $C(LCR_v+2)$
UPDATE SMOOTHING AND CONVERSION PARAMETERS.
MPBCV
LR_v INTO LCR_v LR_v INTO $LRTC_v$
A8


MPBCF
CONVERT LATEST ABS. FROM LOCAL C.S. TO INERTIAL
B0
X, Y, Z (IN FRTC$_1$) TO LRTC$_1$) CLSTOL$_1$
MPBFE
IS THERE AN ERROR
YES
NO
C(AC) + 3 INTO FER$_1$ INTO LER$_1$
LER$_1$ + 1 INTO
REINIT. FOR NEW FIT
GET SMOOTHED FPS-16 VECTOR IF POSSIBLE CSMSLS
FEA$_1$ INTO LEA$_1$
LER$_1$ + 2 INTO
FEE$_1$ INTO LEE$_1$
SWITCH CONTENTS OF DATA TABLE (XYZ) AND DP SAVE BLK
SHOW NEW FIT
SET LN$_{ff}$ 0
SHOW 1ST ENTRY TO SMOOTH
(r, a, e) FOR LAST 4 OBS, C(LCR$_1$ + 9)
SET c_{smf} 0
MP3VR
TO C(LCR$_1$ + 2) C(TSVRAF) TO C(TSVRAF+11)
C6
UNSUPP. FPS 16 TABLE MOVE
SET L$_{Stmf}$ 0
B0
B2

19. CASARC - SHORT ARC PROGRAM

THIS PROGRAM IS A QUICK, SIMPLE METHOD FOR USING FREE-FLIGHT, HIGH SPEED RADAR DATA TO DETERMINE INERTIAL POSITION AND VELOCITY VECTORS AND THEIR VARIANCES. THESE VECTORS CAN BE USED TO COMPUTE A GO, NO-GO RECOMMENDATION AND OTHER DESIRED OUTPUT INFORMATION DURING THE HOLD PHASE. INPUT TO CASARC IS .1 SECOND DATA FROM TWO RADAR SOURCES, VERLORT AND FPS-16. THE PROGRAM IS BASED ON A TAYLOR SERIES EXPANSION OF AN INERTIAL POSITION VECTOR AND APPLIES A LEAST-SQUARES CURVE FIT METHOD TO PRODUCE INERTIAL VECTORS REFLECTING THE SPACECRAFT TRAJECTORY. CASARC ITSELF IS A CONTROLLING PROGRAM WHICH SETS UP INPUT CONDITIONS, AND TRANSFERS CONTROL TO ITS SUBROUTINES CALTOI, CAMPSO, CANORM AND CASOLN, WHICH DO THE ACTUAL MATHEMATICAL PROCESSING. (SEE FLOWCHARTS.)

INPUT

A) TABLES -

TCBDF6	601 LOCATION FPS-16 DATA TABLE
TCBDVE	601 LOCATION VERLORT DATA TABLE
TCTSAS	A ONE LOCATION TABLE CONTAINING THE INITIALIZATION TIME IN FIXED POINT .1 SECONDS
TCFOLR	A ONE LOCATION TABLE CONTAINING THE LOCATION OF THE RANGE COORDINATE OF THE LAST FPS-16 DATA POINT TO BE PROCESSED BY CASARC
TCVOLR	A ONE LOCATION TABLE CONTAINING THE LOCATION OF THE RANGE COORDINATE OF THE LAST VERLORT DATA POINT TO BE PROCESSED

BY CASARC

B) CONSTANTS -

K00000	DEC	0	FIXED POINT ZERO
K00001	DEC	1	FIXED POINT ONE
K00002	DEC	2	FIXED POINT TWO
K00003	DEC	3	FIXED POINT THREE
K001.0	DEC	1.0	FLOATING POINT ONE
K002.0	DEC	2.0	FLOATING POINT TWO
K003.0	DEC	3.0	FLOATING POINT THREE
K0200	DEC	200.	FLOATING POINT TWO-HUNDRED
K0300.	DEC	300.	FLOATING POINT THREE-HUNDRED
KCH206	OCT	206000000000	OCTAL MASK
KCH233	OCT	233000000000	OCTAL MASK
KBFWGT	DEC	1.0	SQUARE ROOT OF FPS-16 WEIGHT
KBVWGT	DEC	.15	SQUARE ROOT OF VERLORT WEIGHT
KFTOER	DEC	.47788040 E-7	1 FOOT IN GEMINI LENGTH UNITS

C) PARAMETERS -

TCBF6L		CONTAINS LOCATION OF FPS-16 DATA TABLE
TCBVEL		CONTAINS LOCATION OF VERLORT DATA TABLE
TCNLOC		A THREE LOCATION TABLE CONTAINING LOCATIONS OF THE NORMAL EQUATION TABLES AS FOLLOWS -
L(TCNLOC)		LOCATION OF TCNOR1
+1		LOCATION OF TCNOR2
+2		LOCATION OF TCNOR3
TCSETN		A NINE LOCATION TABLE CONTAINING VARIOUS PARAMETERS USED TO CONTROL THE USE OF THE NORMAL EQUATION TABLES -
L(TCSETN)	1.	DENOTING NORMAL EQUATION TABLE NUMBER ONE
+1	200.	FACTOR FOR LAST TIME NORMAL EQUATION TABLE NUMBER ONE IS TO BE USED IN FLOATING POINT TENTHS OF SECONDS
+2	50.	FACTOR FOR THE SOLUTION TIME FOR NORMAL EQUATION TABLE NUMBER ONE IN FLOATING POINT TENTHS OF SECONDS
+3	2.	DENOTING NORMAL EQUATION TABLE NUMBER TWO
+4	400.	FACTOR FOR LAST TIME NORMAL EQUATION TABLE NUMBER TWO IS TO BE USED IN FLOATING POINT TENTHS OF SECONDS

	+5	150. FACTOR FOR THE SOLUTION TIME FOR NORMAL EQUATION TABLE NUMBER TWO IN FLOATING POINT TENTHS OF SECONDS
	+6	3. DENOTING NORMAL EQUATION TABLE NUMBER THREE
	+7	600. FACTOR FOR LAST TIME NORMAL EQUATION TABLE NUMBER THREE IS TO BE USED IN FLOATING POINT TENTHS OF SECONDS
	+8	250. FACTOR FOR THE SOLUTION TIME FOR NORMAL EQUATION TABLE NUMBER THREE IN FLOATING POINT TENTHS OF SECONDS
TMSTCH		A TWENTY-EIGHT LOCATION TABLE CONTAINING THE LOCATIONS OF THE FPS-16 AND VERLORT STATION CHARACTERISTICS TABLES AS SPECIFIED.
L(TMSTCH)		
	+	
	•	
	•	
	•	
	+4	LOCATION OF FPS-16 STATION CHARACTERISTICS TABLE
	+5	LOCATION OF VERLORT STATION CHARACTERISTICS TABLE
	+	
	•	
	•	
	•	
	+27	

D) COMMUNICATION CELLS -

MCSAET	ENTRY INDICATOR, SET TO ZERO FOR FIRST ENTRY OR FOR REINITIALIZATION
MCSASO	SOLUTION AVAILABLE INDICATOR, SET TO ZERO WHEN NO NEW SOLUTION FROM CASARC IS AVAILABLE
MCTIME	CONTAINS CURRENT TIME IN FLOATING POINT .1 SECONDS

E) SUBROUTINES

CAMPSO	APPROXIMATE MIDPOINT SOLUTION SUBROUTINE
CALTOI	LOCAL TO INERTIAL COORDINATE CONVERSION SUBROUTINE
CANORM	SUBROUTINE TO FORM NORMAL EQUATIONS
CASOLN	SUBROUTINE TO SOLVE NORMAL EQUATIONS

INTERMEDIARY TABLES

CASARC USES A NUMBER OF CELLS FOR INTERNAL STORAGE AND CONTROL PURPOSES AND ALSO FOR INPUT TO ITS VARIOUS SUBROUTINES. THESE TABLES AND THEIR VARIOUS FUNCTIONS ARE DESCRIBED BELOW -

TCFLAR	A ONE LOCATION TABLE CONTAINING THE LOCATION OF THE RANGE COORDINATE OF THE LAST FPS-16 POINT PREVIOUSLY PROCESSED, SET BY CASARC ON INITIALIZATION
TCVLAR	A ONE LOCATION TABLE CONTAINING THE LOCATION OF THE RANGE COORDINATE OF THE LAST VERLORT POINT PREVIOUSLY PROCESSED, SET BY CASARC ON INITIALIZATION

TCNOR1 TCNOR2 TCNOR3	TWENTY-FIVE LOCATION TABLES USED FOR NORMAL EQUATIONS 1,2 AND 3, RESPECTIVELY. THE FIRST THREE LOCATIONS OF EACH TABLE ARE PARAMETERS CONTROLLING HOW LONG THE TABLE SHOULD BE USED AND WHAT THE SOLUTION TIME FOR THE ASSOCIATED NORMAL EQUATION SHOULD BE. THEY ARE INITIALIZED USING THE CONTENTS OF TABLE TCSETN. ON SUBSEQUENT PASSES THROUGH CASACR, THE CONTENTS OF THE LOCATIONS ARE MODIFIED WHEN REINITIALIZATION TAKES PLACE. THE TABLES ARE USED BY CASARC AS INPUT TO CAMPSO, CANORM AND CASOLN. CAMPSO MODIFIES THE CONTENTS OF LOCATIONS L+3 THROUGH L+24. CANORM MODIFIES THE CONTENTS OF LOCATIONS L+3 THROUGH L+7, L+10 THROUGH L+12, L+15 THROUGH L+17 AND L+20 THROUGH L+22. CASOLN DOES NOT MODIFY THE CONTENTS OF ANY LOCATIONS.

L(TCNOR1, 2 OR 3)		SET NO.	FLOATING POINT 1., 2., OR 3. TO INDICATE THE NUMBER OF THE NORMAL EQUATION TABLE
	+1	t_1	FLOATING POINT .1 SECONDS, GMT TO INDICATE THE LAST TIME THIS NORMAL EQUATION TABLE IS TO BE USED BEFORE BEING REINITIALIZED
	+2	t_s	SOLUTION TIME FOR THE ASSOCIATED NORMAL EQUATION, FLUATING POINT .1 SECONDS, GMT
	+3	n	NUMBER OF OBSERVATIONS, FLOATING POINT
	+4	ΣW_i	SUM OF WEIGHTS, FLOATING POINT
	+5	$\Sigma W_i a_i^2$	GEMINI UNITS
	+6	$\Sigma W_i b_i^2$	
	+7	$\Sigma W_i a_i b_i$	
	+8	x_0	APPROXIAMTE SOLUTION, GEMINI LENGTH UNITS

+9	$\dot{x}_0$	APPROXIMATE SOLUTION, GEMINI VELOCITY UNITS
+10	$\Sigma W_i a_i \xi_i$	GEMINI UNITS
+11	$\Sigma W_i b_i \xi_i$	
+12	$\Sigma W_i \xi_i^2$	
+13	y_0	APPROXIMATE SOLUTION, GEMINI LENGTH UNITS
+14	$\dot{y}_0$	APPROXIMATE SOLUTION, GEMINI VELOCITY UNITS
+15	$\Sigma W_i a_i \eta_i$	GEMINI UNITS
+16	$\Sigma W_i b_i \eta_i$	
+17	$\Sigma W_i \eta_i^2$	
+18	z_0	APPROXIMATE SOLUTION, GEMINI LENGTH UNITS
+19	$\dot{z}_0$	APPROXIMATE SOLUTION, GEMINI VELOCITY UNITS

+20	$\sum W_i a_i \zeta_i$	
+21	$\sum W_i b_i \zeta_i$	GEMINI UNITS
+22	$\sum W_i \zeta_i^2$	
+23	r_0^3	BASED ON APPROXIMATE SOLUTION, GEMINI LENGTH UNITS
+24	m	ZERO IF APPROXIMATE SOLUTION IS AVAILABLE, OTHERWISE NON-ZERO

TCLOCN A THREE LOCATION TABLE INITIALIZED BY CASARC USING THE CONTENTS OF TABLE TCNLOC SO THAT IT CONTAINS THE LOCATIONS OF TABLE TCNOR1, TCNOR2 AND TNCOR3, RESPECTIVELY. ON SUBSEQUENT PASSES THROUGH CASARC, THE CONTENTS OF THE LOCATIONS ARE PERMUTED SO THAT THE FIRST LOCATIONS ALWAYS CONTAINS THE LOCATION OF THE TABLE CURRENTLY BEING USED FOR SOLUTION -

L(TCLOCN) LOATION OF NORMAL EQUATION TABLE BEING CURRENTLY USED FOR SOLUTION

+1 LOCATION OF NORMAL EQUATION TABLE TO BE USED NEXT

+2 LOCATION OF NORMAL EQUATION TABLE TO BE USED FINALLY BEFORE MODIFIED CURRENT TABLE IS USED AGAIN

TCORAE		A FOUR LOCATION TABLE FILLED IN BY CASARC USING DATA FROM TCBDF6 OR TCBDVE AND USED BY CASARC AS INPUT TO CALTOI -
L(TCORAE)	t_i	TIME IN FLOATING POINT 01 SECONDS, GMT
+1	r_i	RANGE IN FLOATING POINT GEMINI LENGTH UNITS
+2	a_i	AZIMUTH IN FLOATING POINT RADIANS
+3	e_i	ELEVATION IN FLOATING POINT RADIANS
TCOXYZ		A SIX LOCATION TABLE OUTPUT FROM CALTOI AND USED BY CASARC AS INPUT TO CANORM -
L(TCOXYZ)	t_i	TIME IN FLOATING POINT .1 SECONDS, GMT
+1	x_i	
+2	y_i	POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+3	z_i	
+4	$\sqrt{W_i}$	SQUARE ROOT OF RADAR WEIGHT IN FLOATING POINT
+5	W_i	RADAR WEIGHT IN FLOATING POINT
TCNPWC		A ONE LOCATION TABLE CONTAINING THE SUM OF THE NUMBER OF DATA POINTS USED AND THE WEIGHTS OF THESE POINTS. THE CONTENTS OF THE TABLE IS MODIFIED BY CASARC USING THE CONTENTS OF LOCATIONS L+3 AND L+4 OF THE NORMAL EQUATION TABLE CURRENTLY BEING USED FOR SOLUTION.

TCSRVT		A THIRTEEN LOCATION TABLE CONTAINING THE CASARC SOLUTION AND ITS ASSOCIATED VARIANCES, USED BY CASARC AS INPUT TO CAMPSO -
L(TVSRVT)	t_e	TIME IN FLOATING POINT .1 SECONDS, GMT
+1	x_e	
+2	y_e	POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+3	z_e	
+4	$\dot{x}_e$	
+5	$\dot{y}_e$	VELOCITY VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+6	$\dot{z}_e$	
+7	$\sigma_{x_e}^2$	
+8	$\sigma_{y_e}^2$	POSITION VECTOR VARIANCE IN FLOATING POINT GEMINI LENGTH UNITS, SQUARED
+9	$\sigma_{z_e}^2$	
+10	$\sigma_{\dot{x}_e}^2$	
+11	$\sigma_{\dot{y}_e}^2$	VELOCITY VECTOR VARIANCE IN FLOATING POINT GEMINI VELOCITY UNITS, SQUARED
+12	$\sigma_{\dot{z}_e}^2$	

OUTPUT

THE OUTPUT FROM CASARC IS FOUND IN TABLE TSCRVT, A THIRTEEN LOCATION TABLE CONTAINING THE TIME OF THE SOLUTION AND THE POSITION AND VELOCITY VECTORS AND THEIR VARIANCES. THIS TABLE IS ALSO USED BY CASARC AS AN INTERMEDIATE TABLE AS INPUT TU CAMPSO. REFER TO THE PRECEDING SECTION FOR A TABLE FORMAT DESCRIPTION, BUT NOTE

THAT THE SOLUTION TABLE CONTAINS COMPONENTS WITH SUB-SCRIPTS ''S'' DENOTING THE ACTUAL SOLUTION OF THE SHORT ARC PROCESS.

L(TCSRVT)	t_s
+1	x_s
+2	y_s
+3	z_s
•	
•	
•	
+12	$\sigma_{\dot{z}_s}$

FORMULATION

THE METHOD IS BASED ON A TAYLOR SERIES EXPANSION OF AN INERTIAL POSTION VECTOR -

$$\bar{r} = \bar{r}_s + \bar{r}'_s (t - t_s) + r''_s \frac{(t - t_s)^2}{2} + \bar{r}''_0 \frac{(t - t_s)^3}{6} + \ldots .$$

THEN THE SIMPLE CONSTRAINT OF GRAVITY -

$$r'' = -\frac{\mu \bar{r}}{r^3} = -\frac{\bar{r}}{r^3} \qquad \text{(IF UNITS ARE CHOSEN PROPERLY SO THAT } \mu = 1)$$

IS USED TO OBTAIN THE SERIES EXPANSION AS A FUNCTION OF POSITION AND VELOCITY -

$$\bar{r} = f(t)\bar{r}_s + g(t)\bar{v}_s$$

WHERE

$$f(t) = 1 - \frac{\Delta t^2}{2r_s^3} + \ldots$$

$$g(t) = \Delta t - \frac{\Delta t^3}{6r_s^3} + \ldots$$

NOTE: SEE HERGET, COMPUTATION OF ORBITS, PAGES 40-41

$$\Delta t = t - t_s$$

SINCE TERMS HIGHER THAN THE SECOND DEGREE BECOME INCREASINGLY COMPLEX AND ARE NOT SIGNIFICANT FOR THE LENGTH OF ARC TO BE USED, THEY ARE NEGLECTED AS ARE OBLATENESS AND DRAG EFFECTS. THESE SIMPLIFICATIONS DO NOT AFFECT ACCURACY FOR ARC LENGTHS UP TO A MINUTE OF TIME OR ALTITUDES GREATER THAN 75 NAUTICAL MILES.

THE FORMULAS ABOVE ARE VALID ONLY FOR AN INERTIAL SYSTEM OR COORDINATES WITH THE EARTH'S MASS AS THE UNIT OF MASS, ONE EQUATORIAL MEAN RADIUS AS THE UNIT OF LENGTH, AND WITH THE UNIT OF TIME BEING CHOSEN TO MAKE THE CONSTANT μ EQUAL TO 1. A LOCAL RADAR OBSERVATION MUST, THEREFORE, BE PROPERLY TRANSFORMED TO THE INERTIAL SYSTEM AND UNITS. (SEE CALTOI, SECTION 21) FOR A DERIVATION OF THE LOCAL TO INERTIAL COORDINATE TRANSFORMATION.)

THE VECTOR EQUATION IS, IN FACT, THREE LINEAR EQUATIONS OF CONDITION FOR EACH RADAR OBSERVATION. IF MORE THAN ONE RADAR SOURCE IS INVOLVED, THESE EQUATIONS SHOULD BE WEIGHTED TO TAKE INTO CONSIDERATION THE RELATIVE ACCURACY OF THE VARIOUS DATA SOURCES. THE FOLLOWING EQUA-

TIONS WEIGHT THE INFORMATION -

$$\sqrt{W_i}\,[a_i\,x_s + b_i\,\dot{x}_s = x_i]$$

$$\sqrt{W_i}\,[a_i\,y_s + b_i\,\dot{y}_s = y_i]$$

$$\sqrt{W_i}\,[a_i\,z_s + b_i\,\dot{z}_s = z_i]$$

WHERE

$$a_i = f(t_i)$$

$$b_i = g(t_i)$$

IN ADDITION TO THE POSITION AND VELOCITY VECTORS $\bar{r}_s$ AND $\bar{v}_s$, THEIR VARIANCES ARE ALSO REQUIRED WITHOUT CONSULTING THE CONDITION EQUATIONS TO OBTAIN THE RESIDUALS. IN USING THE FOLLOWING METHOD, ALL SIGNIFICANT DIGITS IN THE VARIANCE CALCULATION ARE LOST UNLESS AN APPROXIMATE SOLUTION IS KNOWN AND THE VARIABLE CHANGED TO ADJUST THE DECIMAL POINT AND RETAIN ACCURACY. AN ESTIMATED SOLUTION $\bar{r}_e$, $\bar{v}_e$ AT SOME TIME, t_e, WILL BE AVAILABLE AFTER THE FIRST SUCCESSFUL USE OF THE SHORT ARC PROGRAM (BY USING THE PREVIOUSLY GENERATED SOLUTION). AN APPROXIMATE POSITION SOLUTION, $\bar{r}_0$, IS OBTAINED FOR EACH NORMAL EQUATION BY THE SHORT ARC METHOD ITSELF. THE VELOCITY, $\bar{v}_0$, IS OBTAINED BY TAKING THE DERIVATIVE OF THE POSITION EXPRESSION WHICH RESULTS IN THE FOLLOWING EQUATIONS -

$$\bar{r}_0 = (1 - \frac{\mu\,\Delta t^2}{2})\,\bar{r}_e + \Delta t\,(1 - \frac{\mu\,\Delta t^2}{6})\,\bar{v}_e$$

$$\bar{v}_0 = (1 - \frac{\mu\,\Delta t^2}{2})\,\bar{v}_e + \mu\Delta t\,(\frac{3}{2}\sigma\,\Delta t - 1)\,\bar{r}_e$$

THE EXPRESSION DIFFERENTIATED TO OBTAIN $\bar{v}_0$ ACTUALLY CONTAINS ONE ADDITIONAL TERM IN THE $\bar{r}_e$ COEFFICIENT, I.E.,

$$\bar{r}_0 = \left(\frac{1 - \mu\Delta t^2}{2} + \frac{\mu_0 \Delta t^3}{2}\right)\bar{r}_e + \Delta t\left(\frac{1 - \mu\Delta t^2}{6}\right)\bar{v}_e$$

$\bar{r}_e$ AND $\bar{v}_e$ ARE HELD CONSTANT WHILE PERFORMING THE DIFFERENTIATION.

WHERE $\sigma = \dfrac{\bar{r}_e \cdot \bar{v}_e}{r_e^2}$;

$$\mu = \frac{1}{r_e^3}; \qquad \Delta t = t_0 - t_e$$

MAKING A SUBSTITUTION OF VARIABLES (IN EQUATION SET (1))

$$x_s = x_0 + \xi_0$$

$$\dot{x}_s = \dot{x}_0 + \dot{\xi}_0$$

$$x_i = \xi_i + a_i x_0 + b_i \dot{x}_0 \qquad \text{SIMILARLY FOR y AND z}$$

THE CONDITION EQUATIONS REDUCE TO THE FOLLOWING -

$$\sqrt{W_i}\,[a_i \xi_0 + b_i \dot{\xi}_0 = \xi_i]$$

$$\sqrt{W_i}\,[a_i \eta_0 + b_i \dot{\eta}_0 = \eta_i]$$

$$\sqrt{W_i}\,[a_i \zeta_0 + b_i \dot{\zeta}_0 = \zeta_i]$$

THIS PRODUCES THREE SETS OF SIMULTANEOUS EQUATIONS WITH THE SAME COEFFICIENT MATRIX. TO OBTAIN A LEAST SQUARES SOLUTION, FIRST MULTIPLY THE MATRIX ARRAY BY ITS TRANSPOSE TO OBTAIN THE NORMAL EQUATIONS -

$$\Sigma W_i a_i^2 \xi_0 + \Sigma W_i a_i b_i \dot{\xi}_0 = \Sigma W_i a_i \xi_i$$

$$\Sigma W_i a_i b_i \xi_0 + \Sigma W_i b_i^2 \dot{\xi}_0 = \Sigma W_i b_i \xi_i \qquad \text{Similarly for } \eta \text{ and } \zeta.$$

SOLVING BY DETERMINANTS, THE FOLLOWING SOLUTION IS OBTAINED -

$$\xi_0 = \frac{\Sigma W_i b_i^2 \; \Sigma W_i a_i \xi_i - \Sigma W_i a_i b_i \; \Sigma W_i b_i \xi_i}{D}$$

$$\dot{\xi}_0 = \frac{\Sigma W_i a_i^2 \; \Sigma W_i b_i \xi_i - \Sigma W_i a_i b_i \; \Sigma W_i a_i \xi_i}{D} \qquad \text{Similarly for } \eta \text{ and } \zeta.$$

WHERE

$$D = \Sigma W_i a_i^2 \, \Sigma W_i b_i^2 - [\Sigma W_i a_i b_i]^2 .$$

FROM STANDARD TEXTS ON STATISTICS, THE STANDARD ERROR OF $\sqrt{W_i \xi_i}$ IS GIVEN BY -

$$S_\xi = \sqrt{\frac{1}{n-2} \Sigma R_i^2}$$

$$R_i = \sqrt{W_i}\,[a_i \xi_0 + b_i \dot{\xi}_0 - \xi_i]$$

AND N = NUMBER OF OBSERVATIONS.

SUBSTITUTING R_i IN THE EXPRESSION FOR S_ξ AND USING THE NORMAL EQUATIONS FOR SIMPLIFIACTION, THE FOLLOWING EQUATION IS OBTAINED -

$$S_{\xi_i}^2 = \frac{1}{n-2} [\Sigma W_i \xi_i^2 - \xi_0 \Sigma W_i a_i \xi_i - \dot{\xi}_0 \Sigma W_i b_i \xi_i]$$

WITH SIMILAR EXPRESSIONS FOR $S_{\eta_i}^2$ AND $S_{\zeta_i}^2$.

USING THE ABOVE NOTATIONS, THE INDIVIDUAL COMPONENTS $\sigma_{x_s}^2, \sigma_{\dot{x}_s}^2$, ETC., WHICH ARE EQUAL TO $\sigma_{\xi_0}^2, \sigma_{\dot{\xi}_0}^2$, ETC., CAN BE DERIVED. IF THE ERRORS IN x_i ARE INDEPENDENT OF EACH OTHER AND HAVE A NORMAL DISTRIBUTION, AND IF THE FUNCTION $y = \Sigma a_i x_i$, THEN $\sigma_{y_s}^2 = \Sigma a_i \sigma_{\dot{x}_s}^2$ THE SOLUTION CAN BE REARRANGED TO OBTAIN -

$$\xi_0 = \sum_j \frac{(\Sigma W_i b_i^2 \sqrt{W_j a_j} - \Sigma W_i a_i b_i \sqrt{W_j b_j}) \sqrt{W_j \xi_j}}{D}$$

HENCE

$$\sigma_{\xi_0}^2 = \frac{[\Sigma W_i b_i^2]^2 \Sigma W_i a_i^2 + [\Sigma W_i a_i b_i]^2 \Sigma W_i b_i^2 - 2\Sigma W_i b_i^2 [\Sigma W_i a_i b_i]^2}{D^2} S_{\xi_i}$$

WHICH REDUCES TO

$$\sigma_{x_s}^2 = \sigma_{\xi_0}^2 = \frac{\Sigma W_i b_i^2}{D} S_{\xi_i}^2 \text{ and similarly } \sigma_{\dot{x}_s}^2 = \frac{\Sigma W_i a_i^2}{D} S_{\xi_i}^2$$

WITH SIMILAR EXPRESSIONS FOR y AND z.

THE ACCURACY OF THE SOLUTION DEPENDS UPON THE NUMBER AND WEIGHT OF THE OBSERVATIONS, THE LENGTH OF ARC USED, AND THE ACCURACY OF THE OBSERVATIONS.

METHOD

A) CASARC, UPON GAINING CONTROL, SETS MCSASO TO ZERO TO SHOW THAT THERE IS NO NEW SHORT ARC SOLUTION YET AVAILABLE THIS ENTRY. NEXT, IT TESTS MCSAET. IF MCSAET IS ZERO, IT IS ASSUMED THAT THERE HAVE BEEN NO PERVIOUS ENTRIES TO THE PROGRAM, OR THAT REINITIALIZATION IS TO TAKE PLACE, I.E., THAT PROCESSING OF OBSERVATIONS IN THE DATA TABLES IS TO BEGIN AT A LOCATION OTHER THAN THAT AT WHICH PREVIOUS PROCESSING STOPPED. IF MCSASO IS NOT ZERO. CONTROL IS TRANSFERRED TO .

B) NEXT, IT IS NECESSARY TO DETERMINE THE RANGE LOCATIONS OF THE FIRST OBSERVATIONS TO BE PROCESSED, THIS ENTRY, IN THE DATA TABLES CONTAINING THE VERLORT AND FPS-16 RADAR COORDINATES (TCBDVE AND TCBDF6 RESPECTIVELY).

1) TCFLAR IS TO CONTAIN THE LOCATION OF THE RANGE COORDINATE OF THE LAST FPS-16 POINT PREVIOUSLY PROCESSED BY SHORT ARC. THIS IS ACCOMPLISHED BY TESTING TCBDF6 WHICH CONTAINS THE TIME ASSOCIATED WITH THE FIRST OBSERVATION IN THE FPS-16 DATA TABLE.

2) IF TCBDF6 IS ZERO, IT IS ASSUMED THAT THERE HAVE BEEN NO POINTS PREVIOUSLY PROCESSED IN THE FPS-16 TABLE AND TCFLAR IS GIVEN THE LOCATION OF A CELL PRECEDING THE LOCATION OF THE DATA TABLE AS FOLLOWS -

$$TCFLAR = \overbrace{\text{LOCATION OF TCBDF6}}^{TCBF6L} - 2$$

3) IF TDBDF6 IS NOT ZERO, A TEST IS MADE TO SEE WHETHER OR NOT PROCESSING IS TO BEGIN WITH THE FIRST OBSERVATION IN THE TABLE, OR WITH A SUBSEQUENT OBSERVATION. UPON ENTRY TO CASARC, TCTSAS CONTAINS THE TIME (GMT) OF THE FIRST OBSERVATION TO BE PROCESSED. IF THIS TIME IS LESS THAN OR EQUAL TO THE TIME IN TCBDF6, TCFLAR IS SET AS IN B) 2. IF TCTSAS CONTAINS A TIME GREATER THAN THAT CONTAINED IN TCBDF6, TCFLAR IS SET AS FOLLOWS -

$$TCFLAR = \overbrace{\text{LOCATION OF TCBDF6}}^{TCBF6L} + 3(TCTSAS - TCBDF6) - 2$$

4) TCVLAR IS TO CONTAIN THE LOCATION OF THE RANGE COORDINATE OF THE LAST VERLORT POINT PREVIOUSLY PROCESSED BY SHORT ARC. THIS IS ACCOMPLISHED BY TESTING TCBDVE WHICH CON-

TAINS THE TIME ASSOCIATED WITH THE FIRST OBSERVATION IN THE VERLORT DATA TABLE.

5) IF TCBDVE IS ZERO, IT IS ASSUMED THAT THERE HAVE BEEN NO POINTS PREVIOUSLY PROCESSED IN THE VERLORT TABLE AND TCVLAR IS GIVEN THE LOCATION OF A CELL PRECEDING THE LOCATION OF THE DATA TABLE AS FOLLOWS -

$$TCVLAR = \overbrace{\text{LOCATION OF TCBDVE}}^{TCBVEL} - 2$$

6) IF TCBDVE IS NOT ZERO, A TEST IS MADE TO SEE WHETHER OR NOT PROCESSING IS TO BEGIN WITH THE FIRST OBSERVATION IN THE TABLE OR WITH A SUBSEQUENT OBSERVATION. IF THE TIME IN TCTSAS IS LESS THAN OR EQUAL TO THE TIME IN TCBDVE, TCVLAR IS SET AS IN 5. IF TCTSAS CONTAINS A TIME GREATER THAN THAT CONTAINED IN TCBDVE, TCVLAR IS SET AS FOLLOWS -

$$TCVLAR = \overbrace{\text{LOCATION OF TCBDVE}}^{TCBVEL} + 3(TCTSAS-TCBDVE) - 2$$

C) NEXT, THREE NORMAL EQUATION TABLES ARE INITIALIZED. CASARC USES 3 SETS OF NORMAL EQUATIONS. IN THIS WAY, THERE IS NOT A NEED FOR AS MUCH REINITIALIZATION AS THERE WOULD BE IF ONLY ONE TABLE WERE USED, AND AFTER THERE HAS BEEN ENOUGH DATA INCORPORATED INTO THE SUMS FOR AN INITIAL SOLUTION THERE CAN BE A NEW SOLUTION EACH TIME A NEW POINT IS INCORPORATED. CASARC IS PRESENTLY SET UP SO THAT AN INITIAL SOLUTION CAN BE MADE AFTER 7 SECONDS OF DATA HAVE BEEN INCORPORATED. THEREAFTER, EACH SET OF NORMAL EQUATIONS IS USED FOR A MAXIMUM OF 60 SECONDS. TABLE TCSETN CON-

TAINS THE PARAMETERS WHICH CONTROL THIS SCHEME. WHEN THE NORMAL EQUATION TABLES ARE INITIALIZED, THE NUMBER OF THE NORMAL EQUATION TABLE, THE LATEST TIME FOR WHICH THE SET OF NORMAL EQUATIONS IS TO BE USED, AND THE SOLUTION TIME FOR THE SET OF NORMAL EQUATIONS ARE PUT RESPECTIVELY INTO LOCATIONS ONE, TWO AND THREE OF EACH NORMAL EQUATION TABLE, TCNORM,M 1,2, OR 3. THE NORMAL EQUATIONS ARE INITIALIZED TO GIVE SOLUTIONS AT TIME WHICH ARE DESIGNATED AS TCTSAS 5, 15, AND 25 SECONDS. IT WILL BE FOUND THAT THEY BY THIS INITIALIZATION, DATA WHICH IS COLLECTED OVER THE TIME INTERVAL OF 0-20 SECONDS AFTER INITIALIZATION THE TCTSAS IS PROCESSED USING THE FIRST SET OF NORMAL EQUATIONS, AND THE SOLUTION IS GIVEN A TIME DESIGNATION OF TCTSAS 5 SECONDS. SIMILARLY, DATA COLLECTED OVER 0-40 SECONDS IS DESIGNATED AS TCTSAS 15 SECONDS AND THE SECOND SET OF NORMAL EQUATIONS IS USED. DATA COLLECTED OVER 0-60 SECONDS IS DESIGNATED AS TCTSAS 25 SECONDS AND THE THIRD SET OF NORMAL EQUATIONS IS USED.

D) NEXT, THE SUBROUTINE CAMPSO IS ENTERED TO GET AN APPROXIMATE SOLUTION FOR EACH SET OF NORMAL EQUATIONS. THE APPROXIMATE SOLUTION IS MADE AT THE SOLUTION TIME SPECIFIED FOR THE PARTICULAR NORMAL EQUATION. THREE ENTRIES ARE MADE TO THE PROGRAM. ON THE FIRST ENTRY, THE SIGN ASSOCIATED WITH THE SECOND LOCATION OF THE CALLING SEQUENCE IS POSITIVE, AND UPON THE TWO SUBSEQUENT ENTRIES, IT IS NEGATIVE. MCSAET IS SET NON-ZERO.

E) A TEST IS MADE TO SEE IF THE LATEST TIME THAT NORMAL EQUATION TABLE NUMBER 1 IS TO BE USED IS GREATER THAN OR EQUAL TO CURRENT TIME, MCTIME. IF IT IS, THE NORMAL EQUATIONS NEED NOT BE REINITIALIZED AND CONTROL IS TRANSFERRED TO F). IF IT IS NOT, THE NORMAL EQUATIONS ARE REINITIALIZED AS FOLLOWS. THE LAST TIME FOR WHICH NORMAL EQUATION TABLE NUMBER 1 IS TO BE USED IS UPDATED 60 SECONDS. THE SOLUTION TIME FOR NORMAL EQUATION TABLE NUMBER 1 IS UPDATED TO THIRTY SECONDS BEFORE THE LAST TIME THAT THE TABLE IS TO BE USED. THE TABLES ARE PERMUTED SO THAT TABLE NUMBER ONE BECOMES TABLE NUMBER THREE. TABLE NUMBER THREE BECOMES TABLE NUMBER TWO AND TABLE NUMBER TWO BECOMES TABLE NUMBER ONE. CAMPSO IS THEN ENTERED TO GET A NEW APPROXIMATE SOLUTION AT THE NEW SOLUTION TIME FOR THE NORMAL EQUATION THREE AND TO SET THE SUM CELLS TO ZERO. CONTROL IS THEN PASSED TO F.

F) ALL FPS-16 OBSERVATION ARE PROCESSED FIRST, AND THEN ALL VERLORT, EACH WITH APPROPRIATE WEIGHTING FACTORS. THE PARAMETERS FOR FPS-16 DATA ARE INITIALIZED AS FOLLOWS. THE LOCATION OF THE FPS-16 STATION CHARACTERISTICS TABLE IS PICKED UP FROM THE TMSTCH TABLE AND STORED FOR USE IN CALTOI, THE LOCAL TO INERTIAL COORDINATE TRANSFORMATION PROGRAM. THE FPS-16 WEIGHT, KBFWGT IS PICKED UP AND STORED. ALSO, TCBDF6, TCFOLR AND TCFLAR ARE STORED FOR USE IN G.

G) A TEST IS MADE TO SEE WHETHER THERE IS ANOTHER POINT TO BE PROCESSED, I.E., IF TCFOLR-3 IS LESS

THAN TCFLAR, OR IF TCVOLR-3 IS LESS THAN TCVLAR, DEPENDING ON WHETHER FPS-16 OR VERLORT DATA IS BEING PROCESSED. IF THERE IS NO MORE FPS-16 DATA TO BE PROCESSED THIS ENTRY, VERLORT PARAMETERS ARE PICKED UP AND CONTROL IS PASSED BACK TO F. IF PROCESSING HAS BEEN COMPLETED ON BOTH VERLORT AND FPS-16, CONTROL IS PASSED TO I.

H) EITHER TCFOLR OR TCVOLR IS SET TO THE LOCATION OF THE RANGE COORDINATE OF THE POINT PRESENTLY BEING PROCESSED. RANGE, AZIMUTH AND ELEVATION COORDINATES OF THE OBSERVATION BEING PROCESSED ARE TESTED. IF ANY ONE OF THEM IS NEGATIVE, IT IS IMPLIED THAT THERE IS MISSING DATA, AND CONTROL IS PASSED TO G. IF ALL COORDINATES ARE POSITIVE, THEY ARE CONVERTED TO FLOATING POINT AND PLACED IN TABLE TCORAE FOR USE AS INPUT TO CALTOI. THE RANGE IS CONVERTED TO GEMINI LENGTH UNITS. CALTOI IS ENTERED TO CONVERT THE LOCAL COORDINATES CONTAINED IN TCORAE TO INERTIAL COORDINATES WHICH ARE STORED IN TCOXYZ. THREE ENTRIES TO CANORM ARE MADE TO ADD THE OBSERVATION TO THE THREE SETS OF NORMAL EQUATIONS. THE SIGN ASSOCIATED WITH THE SECOND LOCATION OF THE CALLING SEQUENCE IS POSITIVE ON THE FIRST ENTRY AND NEGATIVE ON THE TWO SUBSEQUENT ENTRIES. CONTROL IS PASSED TO I.

I) TCSNPW, CONTAINING n + W (THE SUM OF THE NUMBER OF OBSERVATIONS PLUS THE SUM OF THE WEIGHTS OF THESE OBSERVATIONS), IS SET USING THE PARAMETERS SUPPLIED IN THE NORMAL EQUATION TABLE COR-

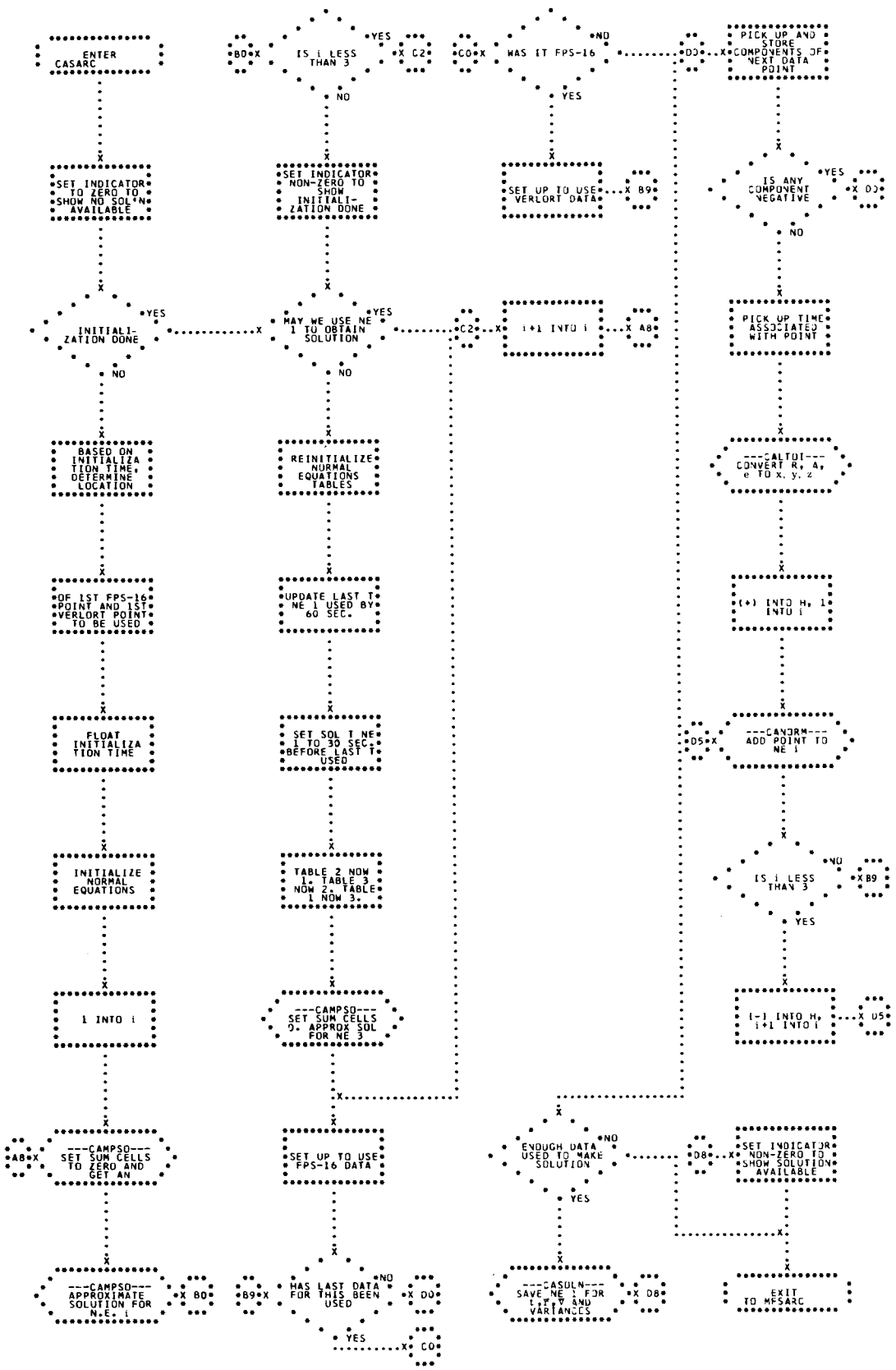
RECTLY BEING USED. A TEST IS MADE TO SEE IF THE SUM OF n + W IS GREATER THAN 200. THIS PARAMETER IS CHOSEN IN ACCORDANCE WITH OTHER PARAMETERS AS DESCRIBED IN C, SO THAT THERE WILL BE A SOLUTION MADE WHEN 20 SECONDS WORTH OF PURE VERLORT DATA HAS BEEN INCORPORATED INTO THE NORMAL EQUATIONS (ASSUMING THAT THE FPS-16 RADAR TABLE IS MISSING DATA) OR WHEN 7 SECONDS WORTH OF FPS-16 AND VERLORT DATA HAVE BEEN INCORPORATED. (FPS-16 DATA IS WEIGHTED MORE HEAVILY THAN VERLORT.)

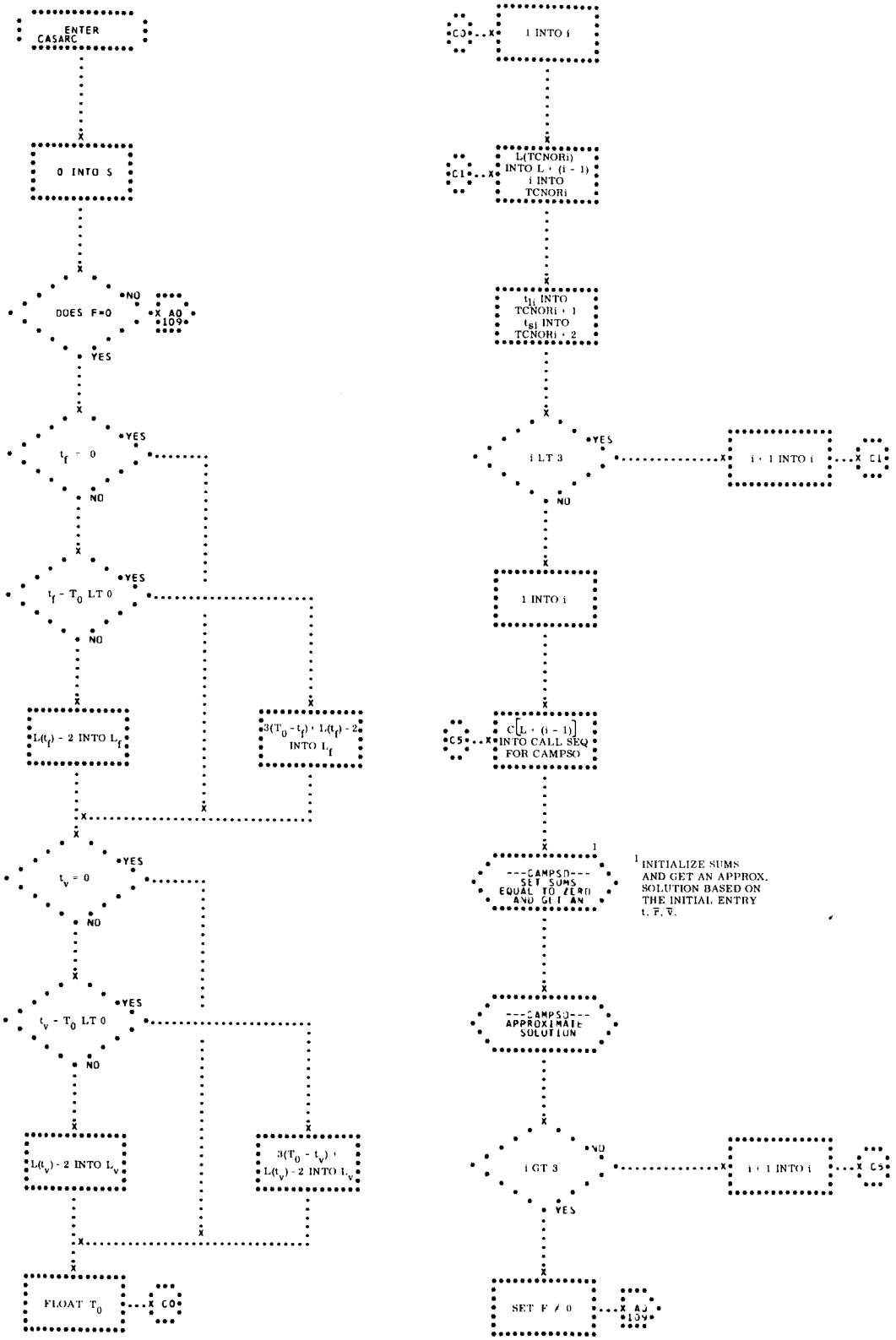
USAGE

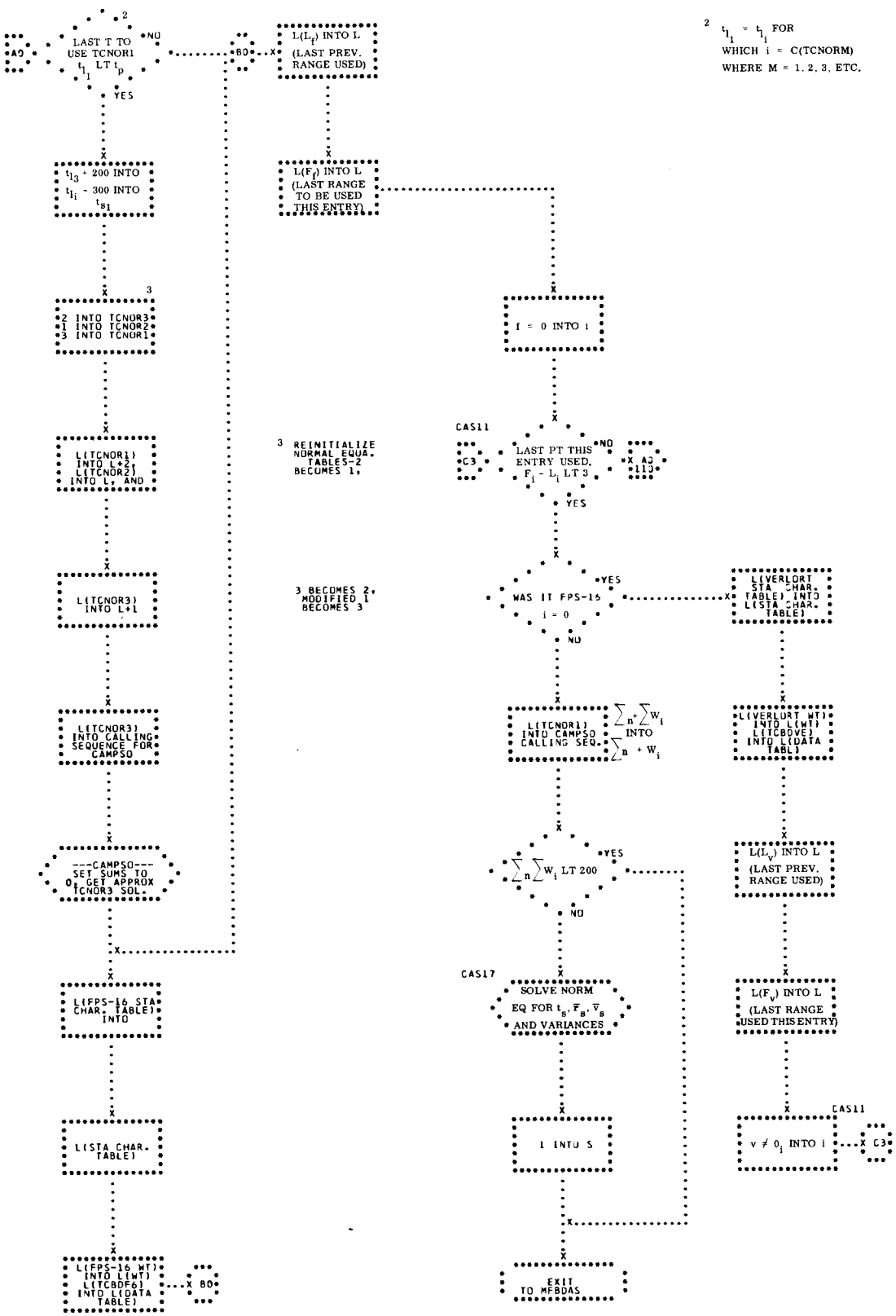
A) THE PROGRAM IS ENTERED BY TRANSFERRING TO CASARC AND ITS EXITS TO ITS SUFFIX MFSARC. NO INDEX REGISTERS ARE SAVED OR RESTORED.

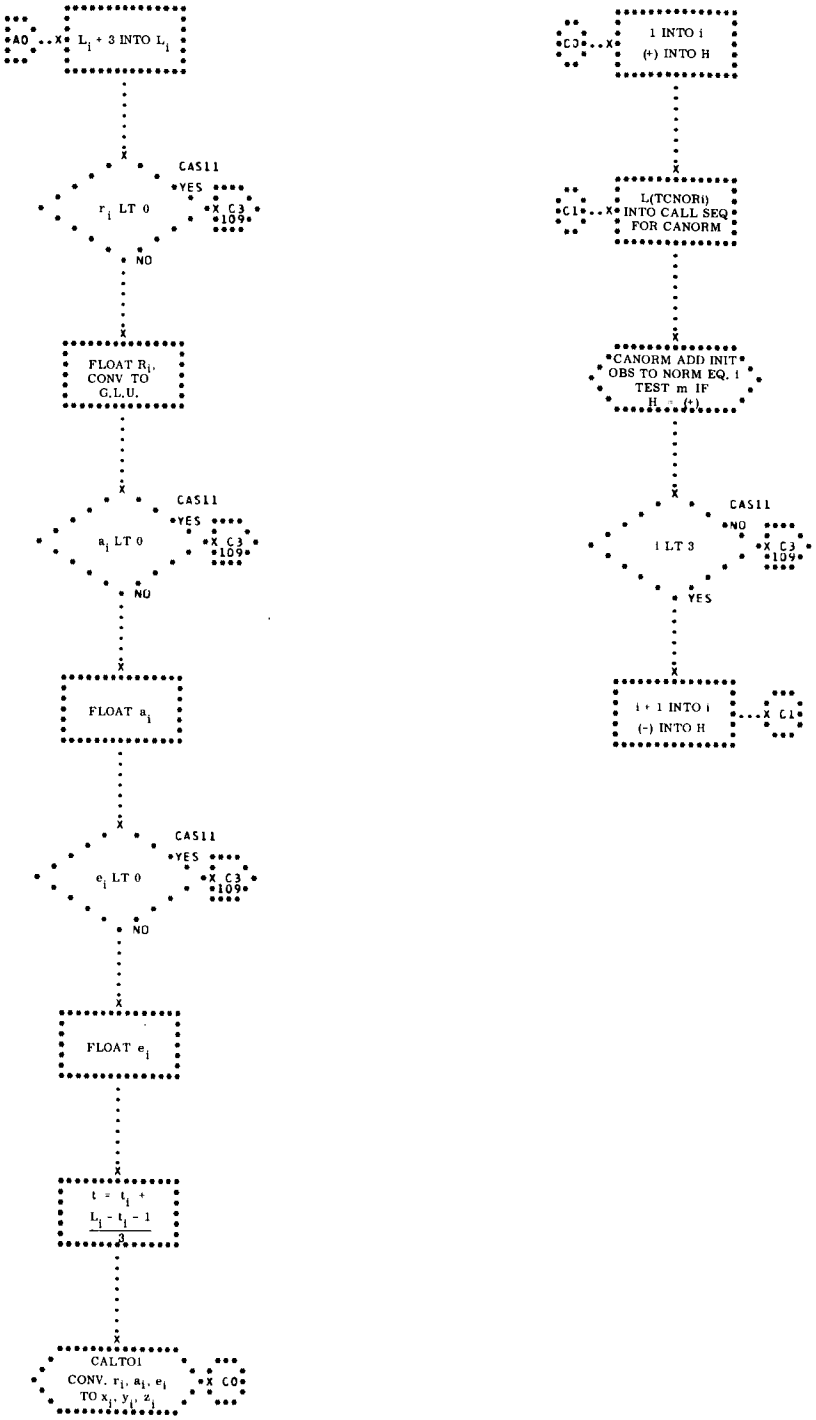
B) STORAGE REQUIREMENTS -
178 DECIMAL LOCATIONS

C) TIMING -
163 MILLISECONDS (MAXIMUM)


ENTER CASARC
SET INDICATOR TO ZERO TO SHOW NO SOL'N AVAILABLE
INITIALI-ZATION DONE
YES
NO
BASED ON INITIALIZATION TIME, DETERMINE LOCATION OF 1ST FPS-16 POINT AND 1ST VERLORT POINT TO BE USED
FLOAT INITIALIZATION TIME
INITIALIZE NORMAL EQUATIONS
1 INTO i
---CAMPSO--- SET SUM CELLS TO ZERO AND GET AN
---CAMPSO--- APPROXIMATE SOLUTION FOR N,E, i
X B0
A8
IS i LESS THAN 3
YES
NO
SET INDICATOR NON-ZERO TO SHOW INITIALI-ZATION DONE
MAY WE USE NE 1 TO OBTAIN SOLUTION
YES
NO
REINITIALIZE NORMAL EQUATIONS TABLES
UPDATE LAST T NE 1 USED BY 60 SEC.
SET SOL T NE 1 TO 30 SEC. BEFORE LAST T USED
TABLE 2 NOW 1, TABLE 3 NOW 2, TABLE 1 NOW 3.
---CAMPSO--- SET SUM CELLS 0. APPROX SOL FOR NE 3
SET UP TO USE FPS-16 DATA
HAS LAST DATA FOR THIS BEEN USED
YES
NO
X C0
B9
WAS IT FPS-16
YES
NO
SET UP TO USE VERLORT DATA
X B9
i+1 INTO i
X A8
C0
C2
ENOUGH DATA USED TO MAKE SOLUTION
YES
NO
---CASOLN--- SAVE NE 1 FOR L.S.P. AND VARIANCES
X D8
D0
PICK UP AND STORE COMPONENTS OF NEXT DATA POINT
IS ANY COMPONENT NEGATIVE
YES
NO
X D3
PICK UP TIME ASSOCIATED WITH POINT
---CALTOT--- CONVERT R, A, e TO x, y, z
(+) INTO H, 1 INTO i
---CANORM--- ADD POINT TO NE 1
D5
IS i LESS THAN 3
NO
YES
X B9
(-) INTO H, i+1 INTO i
X D5
SET INDICATOR NON-ZERO TO SHOW SOLUTION AVAILABLE
D8
EXIT TO MPSARC


ENTER CASARC
0 INTO S
DOES F=0
NO
X A0 109
YES
t_f = 0
YES
NO
$t_f - T_0$ LT 0
YES
NO
$L(t_f) - 2$ INTO L_f
$3(T_0 - t_f) + L(t_f) - 2$ INTO L_f
t_v = 0
YES
NO
$t_v - T_0$ LT 0
YES
NO
$L(t_v) - 2$ INTO L_v
$3(T_0 - t_v) + L(t_v) - 2$ INTO L_v
FLOAT T_0
X C0
C0
1 INTO i
C1
L(TCNORi) INTO L + (i - 1)
i INTO TCNORi
t_{ti} INTO TCNORi + 1
t_{si} INTO TCNORi + 2
i LT 3
YES
i + 1 INTO i
X C1
NO
1 INTO i
C5
C[L + (i - 1)] INTO CALL SEQ FOR CAMPSO
---CAMPSO--- SET SUMS EQUAL TO ZERO AND GET AN APPROXIMATE SOLUTION
1
---CAMPSO--- APPROXIMATE SOLUTION
i GT 3
NO
i + 1 INTO i
X C5
YES
SET F ≠ 0
X A0 109
1 INITIALIZE SUMS AND GET AN APPROX. SOLUTION BASED ON THE INITIAL ENTRY $t, \bar{r}, \bar{v}$.


LAST T TO USE TCNOR1 t_{l_1} LT t_p
NO
YES
A0
B0
t_{l_3} + 200 INTO t_{l_i} - 300 INTO t_{s_1}
2 INTO TCNOR3
1 INTO TCNOR2
3 INTO TCNOR1
L(TCNOR1) INTO L+2, L(TCNOR2) INTO L, AND
L(TCNOR3) INTO L+1
L(TCNOR3) INTO CALLING SEQUENCE FOR CAMPSO
---CAMPSO--- SET SUMS TO 0, GET APPROX t_{TCNOR3} SOL.
L(FPS-16 STA CHAR. TABLE) INTO
L(STA CHAR. TABLE)
L(FPS-16 WT) INTO L(WT) L(TCBDF6) INTO L(DATA TABLE)
X B0
L(L_f) INTO L (LAST PREV. RANGE USED)
L(F_f) INTO L (LAST RANGE TO BE USED THIS ENTRY)
I = 0 INTO i
CAS11
C3
LAST PT THIS ENTRY USED. F_i - L_i LT 3
NO
X A3
L13
YES
WAS IT FPS-16 i = 0
YES
NO
L(TCNOR1) INTO CAMPSO CALLING SEQ.
$\sum_n + \sum W_i$ INTO $\sum_n + W_i$
$\sum_n \sum W_i$ LT 200
YES
NO
CAS17
SOLVE NORM EQ FOR t_s, $\bar{F}_s$, $\bar{v}_s$ AND VARIANCES
1 INTO S
EXIT TO MFBOAS
L(VERLORT STA CHAR. TABLE) INTO L(STA CHAR. TABLE)
L(VERLORT WT) INTO L(WT) L(TCBOVE) INTO L(DATA TABL)
L(L_v) INTO L (LAST PREV. RANGE USED)
L(F_v) INTO L (LAST RANGE USED THIS ENTRY)
CAS11
$v \neq 0_i$ INTO i
X C3
2 $t_{l_1} = t_{l_i}$ FOR WHICH i = C(TCNORM) WHERE M = 1, 2, 3, ETC.
3 REINITIALIZE NORMAL EQUA. TABLES-2 BECOMES 1,
3 BECOMES 2, MODIFIED 1 BECOMES 3


A0
$L_i + 3$ INTO L_i
CAS11
YES
r_i LT 0
X C3 *109
NO
FLOAT R_i, CONV TO G.L.U.
CAS11
YES
a_i LT 0
X C3 *109
NO
FLOAT a_i
CAS11
YES
e_i LT 0
X C3 *109
NO
FLOAT e_i
$t = t_i + \frac{L_i - t_i - 1}{3}$
CALTOI
CONV. r_i, a_i, e_i TO x_i, y_i, z_i
X C0
C3
1 INTO i
(+) INTO H
C1
L(TCNORi) INTO CALL SEQ FOR CANORM
CANORM ADD INIT OBS TO NORM EQ. i TEST m IF H = (+)
CAS11
NO
i LT 3
X C3 *109
YES
i + 1 INTO i
(-) INTO H
X C1

20. CAMPSO - APPROXIMATE MIDPOINT SOLUTION SUBROUTINE

GIVEN THE BEST AVAILABLE SOLUTION $t_e, \bar{r}_e, \bar{v}_e$, THIS PROGRAM IS USED TO COMPUTE AN APPROXIMATE MIDPOINT SOLUTION, $t_0, \bar{r}_0, \bar{v}_0$, WHICH IS USED TO MAKE A CHANGE IN VARIABLE AND TO CALCULATE r_0^3. IT IS A SUBROUTINE OF THE GENERAL SHORT ARC PROGRAM, CASARC. (SEE FLOWCHART.)

INPUT

A) CONSTANTS -

K001.0	DEC 1.0	FLOATING POINT ONE
K001.5	DEC 1.5	FLOATING POINT THREE-HALVES
K002.0	DEC 2.0	FLOATING POINT TWO
K006.0	DEC 6.0	FLOATING POINT SIX
KTS.TU	DEC .12394436E-3	ONE TENTH OF A SECOND IN GEMINI TIME UNITS

B) SUBROUTINES -

UISQRT	FLOATING POINT SQUARE ROOT ROUTINE

C) TABLES -*

TCNORM	A TWENTY-FOUR LOCATION TABLE CONTAINING THE PARAMETERS AND SUMS FOR NORMAL EQUATION M, M=1,2, OR 3
TCSRVT	A THIRTEEN-LOCATION TABLE CONTAINING THE BEST AVAILABLE SOLUTION, $t_e, \bar{r}_e, \bar{v}_e$, AND THE LAST CALCULATED VARIANCES. IF NO SOLUTION IS AVAILABLE, THE TIME SHOULD BE SET TO ZERO.

*SEE CASARC FOR TABLE FORMAT DESCRIPTIONS.

OUTPUT

THE OUTPUT FROM CAMPSO WILL BE FOUND IN TABLE TCNORM AS DESCRIBED IN CASARC.

METHOD

WHEN CAMPSO IS GIVEN CONTROL, IT PICKS UP THE LOCATION OF THE NORMAL EQUATION TABLE TO BE USED AND SETS THE CONTENTS OF LOCATIONS FOUR THROUGH TWENTY-FIVE EQUAL TO ZERO. IT THEN TESTS TO SEE IF THERE IS A SOLUTION AVAILABLE. THIS IS DETERMINED BY TESTING THE TIME ASSOCIATED WITH THE INPUT VECTORS IN TCSRVT. IF t_e IS ZERO, IT IS ASSUMED THAT THERE IS NO SOLUTION AVAILABLE, I.E., THAT $\overline{r}_e$ AND $\overline{v}_e$ ARE MISSING AND IT SETS m (THE TWENTY-FOURTH LOCATION OF THE NORMAL EQUATION TABLE BEING USED) TO ONE AND CONTROL IS RETURNED TO THE CALLING PROGRAM. IN A SUBSEQUENT PROGRAM (CANORM) m=1 INDICATES THAT r_0^3, BASED ON THE APPROXIMATE SOLUTION, IS NOT AVAILABLE AND THAT IT WILL BE NECESSARY TO COMPUTE AND USE r_i^3 IN ITS PLACE. IT ALSO MEANS THAT THE VARIANCES CALCULATED IN CASOLN ARE USELESS SINCE ALL SIGNIFICANT DIGITS WILL BE LOST.

IF THERE IS A SOLUTION AVAILABLE, I.E., IF t_0 IS NOT ZERO, THE LOCATION OF THE NORMAL EQUATION TABLE IS PICKED UP AGAIN, AND THE LOCATION OF THE BEST AVAILABLE SOLUTION (TCSRVT) IS DETERMINED. NEXT, A TEST IS MADE ON THE SIGN OF H (SECOND LOCATION OF THE CALLING SEQUENCE) TO DETERMINE WHETHER OR NOT THE SOLUTION IN TCSRVT IS THE SAME AS THAT USED IN THE LAST PREVIOUS ENTRY TO CAMPSO. IN THIS CASE, CERTAIN QUANTITIES NEED NOT BE RECOMPUTED. INITIALLY, CASARC IS CONCERNED WITH APPROXIMATE SOLUTIONS FOR ALL THREE SETS OF NORMAL EQUATIONS FROM THE SAME INPUT VECTORS. SOME OF THE COMPUTATIONS ARE THE SAME RE-

GARDLESS OF WHICH NORMAL EQUATION IS BEING PROCESSED. TO AVOID DUPLICATION, μ AND σ ARE COMPUTED ONLY UPON THE FIRST ENTRY WHEN H IS SET POSITIVE (+). THIS PORTION OF THE PROGRAM IS BYPASSED ON SUBSEQUENT ENTRIES BY SETTING THE SIGN OF H TO NEGATIVE (-).

ONCE μ AND σ HAVE BEEN COMPUTED, THE PROGRAM COMPUTES COEFFICIENTS FOR THE INPUT VECTORS $\bar{r}_e$ AND $\bar{v}_e$ AS A FUNCTION OF TIME. THEN, BY ITERATING THREE TIMES, THE INDIVIDUAL COMPONENTS OF $\bar{r}_0$ AND $\bar{v}_0$ ARE CALCULATED. FROM THE COMPONENTS OF $\bar{r}_0$, THE VALUE OF r_0^3 IS COMPUTED AND WILL BE USED LATER IN CANORM IN FORMING THE NORMAL EQUATIONS. EXIT IS THEN MADE TO THE CALLING PROGRAM.

THE FOLLOWING EQUATIONS ARE USED IN COMPUTING THE APPROXIMATE MIDPOINT SOLUTION -

$$\bar{r}_0(t) = (1 - \tfrac{1}{2}\mu\Delta t)^2\,\bar{r}_e + \Delta t(1 - \tfrac{1}{6}\mu\Delta t^2)\,\bar{v}_e$$

$$\bar{v}_0(t) = -\mu\Delta t(1 - \tfrac{3}{2}\sigma\Delta t)\,\bar{r}_e + (1 - \tfrac{1}{2}\mu\Delta t^2)\,\bar{v}_e$$

WHERE

$$\mu = \frac{1}{r_e^3}$$

$$\sigma = \frac{\bar{r}_e\ \bar{v}_e}{r_e^3}$$

WHERE

$\Delta t = t_s - t_e$ AND t_s IS THE SOLUTION TIME FOR THE NORMAL EQUATION TABLE BEING USED.

USAGE

A) CALLING SEQUENCE -

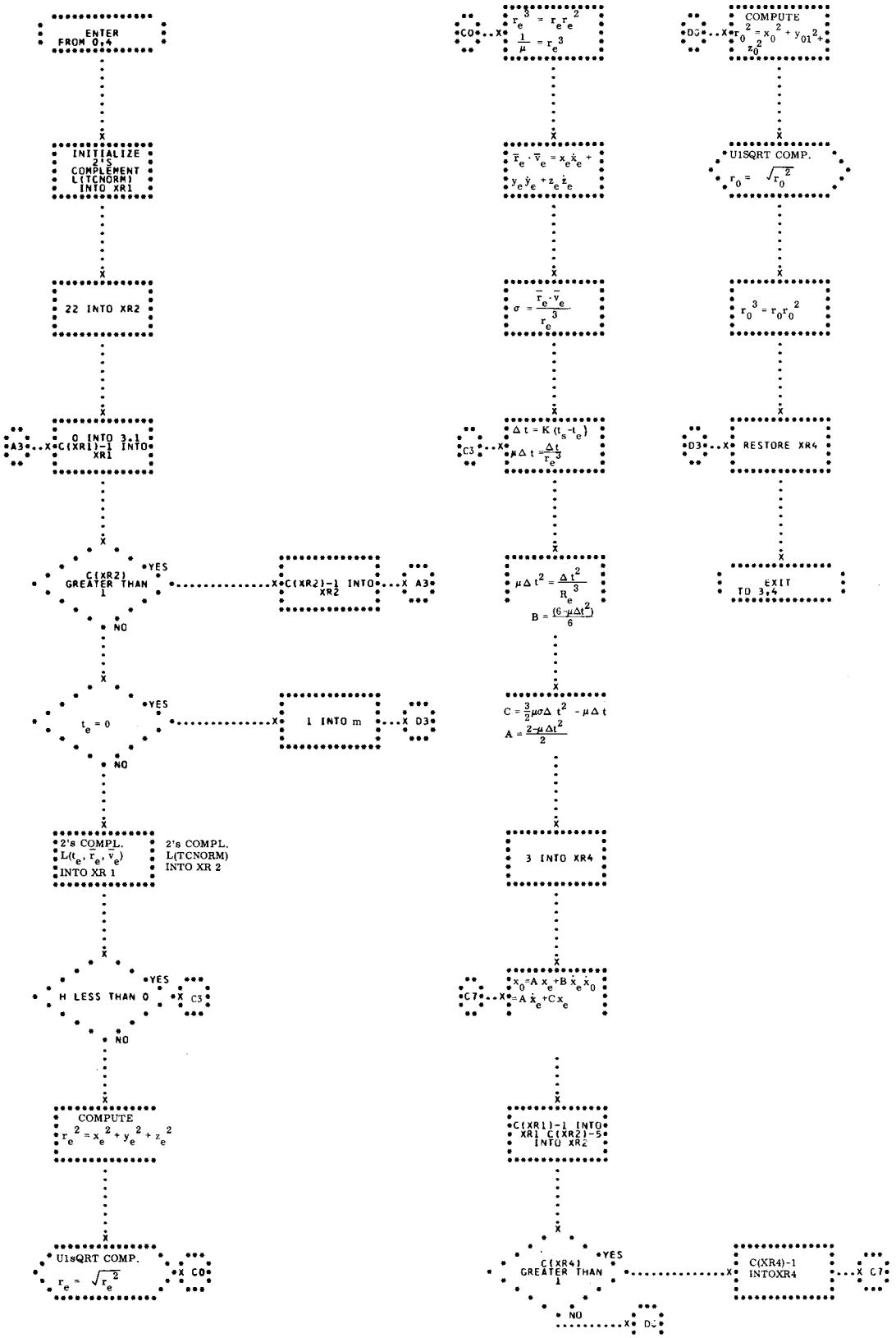
```
TSX     CAMPSO,4
H       L(TCNORM)     M = 1,2, OR 3
PZE     L(TCSRVT)
RETURN
```

B) STORAGE REQUIRED -

109 DECIMAL LOCATIONS

C) TIMING -

2.414 MILLISECONDS


ENTER FROM 0,4
INITIALIZE 2'S COMPLEMENT L(TCNORM) INTO XR1
22 INTO XR2
0 INTO 3,1
C(XR1)-1 INTO XR1
A3
C(XR2) GREATER THAN 1
YES
C(XR2)-1 INTO XR2
A3
NO
$t_e = 0$
YES
1 INTO m
D3
NO
2's COMPL. $L(t_e, \bar{r}_e, \bar{v}_e)$ INTO XR 1
2's COMPL. L(TCNORM) INTO XR 2
H LESS THAN 0
YES
C3
NO
COMPUTE $r_e^2 = x_e^2 + y_e^2 + z_e^2$
U1sQRT COMP. $r_e = \sqrt{r_e^2}$
C0
$r_e^3 = r_e r_e^2$
$\frac{1}{\mu} = r_e^3$
C0
$\bar{r}_e \cdot \bar{v}_e = x_e \dot{x}_e + y_e \dot{y}_e + z_e \dot{z}_e$
$\sigma = \frac{\bar{r}_e \cdot \bar{v}_e}{r_e^3}$
$\Delta t = K(t_s - t_e)$
$\mu \Delta t = \frac{\Delta t}{r_e^3}$
C3
$\mu \Delta t^2 = \frac{\Delta t^2}{R_e^3}$
$B = \frac{(6 - \mu \Delta t^2)}{6}$
$C = \frac{3}{2}\mu\sigma\Delta t^2 - \mu\Delta t$
$A = \frac{2 - \mu \Delta t^2}{2}$
3 INTO XR4
$x_0 = A x_e + B \dot{x}_e x_0$
$\dot{x}_0 = A \dot{x}_e + C x_e$
C7
C(XR1)-1 INTO XR1 C(XR2)-5 INTO XR2
C(XR4) GREATER THAN 1
YES
C(XR4)-1 INTOXR4
C7
NO
D0
COMPUTE $r_0^2 = x_0^2 + y_{01}^2 + z_0^2$
U1SQRT COMP. $r_0 = \sqrt{r_0^2}$
D0
$r_0^3 = r_0 r_0^2$
D3
RESTORE XR4
EXIT TO 3,4

21. CALTOI - LOCAL TO INERTIAL COORDINATE CONVERSION SUBROUTINE

THIS PROGRAM IS USED TO CONVERT A LOCAL RADAR OBSERVATION $t_i, r_i, a_i,$ AND e_i INTO AN INERTIAL OBSERVATION t_i, x_i, y_i, z_i. IT IS A SUBROUTINE OF THE GENERAL SHORT ARC PROGRAM, CASARC. (SEE FLOWCHART.)

INPUT

A) CONSTANTS -

K.OMTS	DEC .72921151E-5	ROTATION OF THE EARTH IN RADIANS PER TENTH OF A SECOND
KLAMDO		INERTIAL LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH

B) SUBROUTINES -

U1SICO	FLOATING POINT SINE - COSINE ROUTINE

C) TABLES -

TCORAE	INPUT TABLE FOR CALTOI CONTAINING LOCAL RADAR t_i, T_i, a_i, e_i
TMCHAR	STATION CHARACTERISTICS TABLE

INTERMEDIARY TABLES

TCOXYZ	A SIX-LOCATION TABLE INPUT TO CALTOI BY CASARC AND USED FOR OUTPUT BY CALTOI. UPON ENTRY TO CALTOI, TCOXYZ CONTAINS $\sqrt{W_i}$ IN LOCATION L+4. THIS QUANTITY IS USED BY CALTOI TO COMPUTE W_i.

OUTPUT

THE OUTPUT t_i, x_i, y_i, z_i FROM CALTOI IS STORED IN TABLE TCOXYZ IN LOCATIONS L THROUGH L+3 AND IN LOCATION L+5.

SEE CASARC PROGRAM WRITEUP FOR FORMAT DESCRIPTION OF TCORAE AND TCOXYZ. SEE CLSTOI PROGRAM WRITEUP FOR FORMAT DESCRIPTION OF TMCHAR.

METHOD

UPON GAINING CONTROL FROM THE CALLING PROGRAM, CALTOI PICKS UP THE LOCATIONS OF THE INPUT AND OUTPUT TABLES. LOCAL x_i, y_i, z_i COORDINATES ARE CALCULATED USING THE FOLLOWING EQUATIONS -

$$x_i'' = r_i \cos e_i \sin a_i$$

$$y_i'' = r_i \cos e_i \cos a_i$$

$$z_i'' = r_i \sin e_i$$

THE LOCATION OF THE STATION CHARACTERISTICS TABLE IS PICKED UP AND THE LOCAL COORDINATES ARE TRANSLATED TO THE CENTER OF THE EARTH USING

$$x_i' = x_i''$$

$$y_i' = y_i'' - R_E \sin (\phi' - \phi')$$

$$z_i' = z_i'' + R_E \cos (\phi - \phi') + H$$

WHERE

ϕ = GEODETIC LATITUDE OF THE STATION

ϕ' = GEOCENTRIC LATITUDE OF THE STATION

R_E = RADIUS OF THE GEMINI SPHEROID AT GEOCENTRIC LATITUDE ϕ'

H = HEIGHT OF THE STATION ABOVE THE GEMINI SPHEROID

THE LONGITUDINAL ROTATION IS THEN CALCULATED AS -

$$\lambda = \lambda_o + \lambda_{STA} + \omega t_i$$

WHERE

λ_o = INERTIAL LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH (KLAMDO)

λ_{STA} = LONGITUDE OF THE STATION

ω = SIDERICAL ROTATION OF THE EARTH IN RADIANS/.1 SECOND (K.OMTS)

t_i = GMT OF THE OBSERVATION

FOR A COMPLETE FORMULATION OF LOCAL TO INERTIAL COORDINATE CONVERSION, REFER TO PROGRAM CLSTOI DESCRIPTION. CALTOI CONVERTS ONLY ONE POINT PER ENTRY. TIME IS INPUT IN FLOATING POINT TENTHS OF SECONDS, RANGE COORDINATES ARE INPUT IN FLOATING POINT GEMINI UNITS AND AZIMUTH AND ELEVATION ARE INPUT IN FLOATING POINT RADIANS SO THEY NEED NOT BE CONVERTED BY CALTOI.

FINALLY, THE AXES ARE ALIGNED WITH THE INERTIAL AXES USING THE FOLLOWING ROTATIONS -

$$\bar{r} = R_Z(-\lambda - 90^o) R_X(\phi - 90^o) \bar{r}'$$

or

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \begin{bmatrix} -\sin\lambda & -\cos\lambda & 0 \\ \cos\lambda & -\sin\lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & \\ 0 & \sin 0 & -\cos 0 \\ 0 & \cos 0 & \sin 0 \end{bmatrix} \begin{bmatrix} x_i' \\ y_i' \\ z_i' \end{bmatrix}$$

or

$$x_i = x_i' \sin\lambda + \cos\lambda (z' \cos\phi - y_i' \sin\phi)$$

$$y_i = x_i \cos\lambda + \sin\lambda (z' \cos\phi - y_i' \sin\phi)$$

$$z_i = y_i' \cos\phi + z_i' \sin\phi$$

EXIT IS MADE TO THE CALLING PROGRAM.

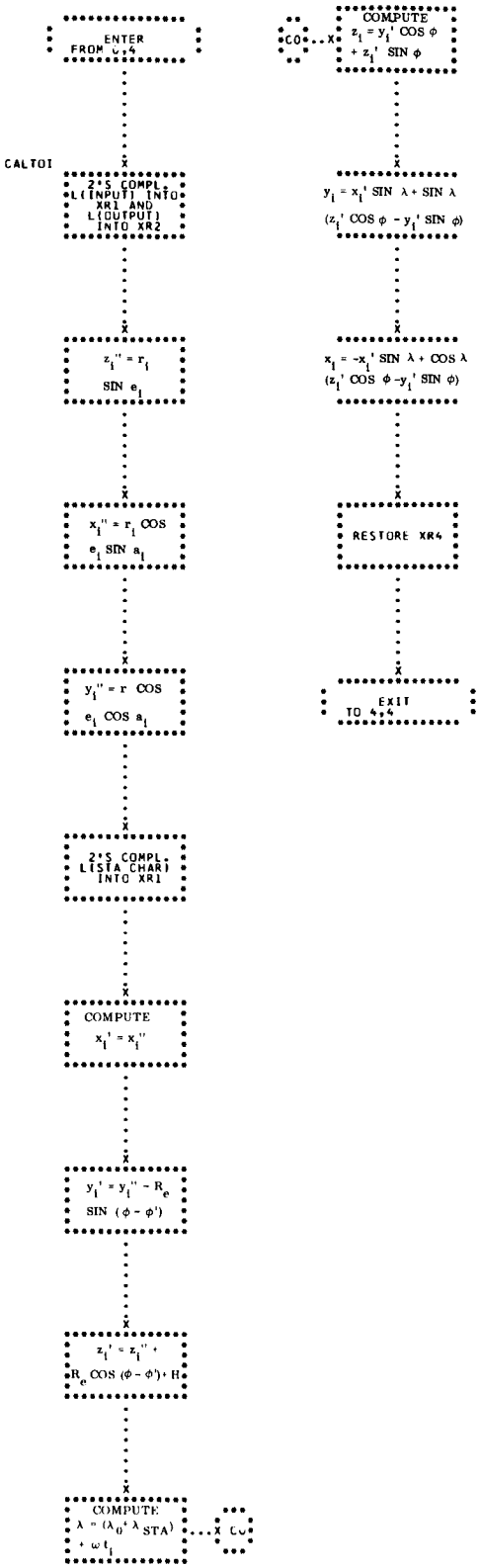
USAGE

A) CALLING SEQUENCE -

```
TSX     CALTOI,4
PZE     L(INPUT TABLE)
PZE     L(OUTPUT TABLE)
TZE     L(STATION CHARACTERISTICS TABLE)
RETURN
```

B) STORAGE REQUIRED - LOCATIONS

C) TIMING - 2.4 MILLISECONDS


ENTER FROM 4.4
CALTOI
2'S COMPL. L(INPUT) INTO XR1 AND L(OUTPUT) INTO XR2
$z_i'' = r_i$ SIN e_i
$x_i'' = r_i$ COS e_i SIN a_i
$y_i'' = r$ COS e_i COS a_i
2'S COMPL. L(STA CHAR) INTO XR1
COMPUTE $x_i' = x_i''$
$y_i' = y_i'' - R_e$ SIN $(\phi - \phi')$
$z_i' = z_i'' +$ R_e COS $(\phi - \phi') + H$
COMPUTE $\lambda = (\lambda_0 + \lambda_{STA}) + \omega t_i$
CO
COMPUTE $z_i = y_i'$ COS $\phi + z_i'$ SIN ϕ
$y_i = x_i'$ SIN $\lambda +$ SIN λ $(z_i'$ COS $\phi - y_i'$ SIN $\phi)$
$x_i = -x_i'$ SIN $\lambda +$ COS λ $(z_i'$ COS $\phi - y_i'$ SIN $\phi)$
RESTORE XR4
EXIT TO 4.4

22. CANORM - SUBROUTINE TO FORM NORMAL EQUATIONS

THIS PROGRAM IS USED TO FORM NORMAL EQUATIONS BASED ON INERTIAL t_i, x_i, y_i, z_i OBSERVATIONS WEIGHTED ACCORDING TO RADAR RELIABILITY FOR USE IN THE GENERAL SHORT ARC PROGRAM, CASARC. (SEE FLOWCHART.)

INPUT

A) CONSTANTS -

K0C1.0	DEC 1.0	FLOATING POINT ONE
K0C2.0	DEC 2.0	FLOATING POINT TWO
K0U6.0	DEC 6.0	FLOATING POINT THREE
KTS.TU	DEC. 12394436E-3	ONE-TENTH OF A SECOND IN GEMINI TIME UNITS

B) TABLES -

(SEE CASARC FOR TABLE FORMAT DESCRIPTION)

TCNORM	A TWENTY-FOUR LOCATION TABLE CONTAINING THE PARAMETERS AND SUMS FOR NORMAL EQUATION M, M=1, 2 OR 3
TCCXYZ	TABLE CONTAINING INERTIAL t_i, x_i, y_i, z_i OBSERVATION AND $\sqrt{W_i}, W_i$ TO BE ADDED TO THE CONDITICN EQUATION

C) SUBROUTINE -

U1SQRT	FLOATING POINT SQUARE ROOT ROUTINE

OUTPUT

THE OUTPUT FROM CANORM IS STORED IN TABLE TCNORM AS DESCRIBED IN CASARC.

METHOD

WHEN CONTROL IS TRANSFERRED TO CANORM, THE LOCATION OF THE NORMAL EQUATION TABLE BEING USED, AND THE LOCATION OF THE OBSERVATION BEING ADDED TO THE NORMAL EQUATION SUMS ARE PICKED UP. A TEST IS MADE TO DETERMINE WHETHER OR NOT m, THE TWENTY-FOURTH LOCATION OF THE NORMAL EQUATION TABLE, IS ZERO. IF μ IS ZERO, THE QUANTITY r_0^3, WHICH IS TO BE USED IN THE COMPUTATIONS IN CANORM, HAS ALREADY BEEN COMPUTED BASED ON THE APPROXIMATE SOLUTION FOUND BY CAMPSO. IF $m \neq 0, r_i^3$, BASED ON THE OBSERVATION BEING ADDED. MUST BE COMPUTED AND USED INSTEAD.

THE CONDITION EQUATION COEFFICIENTS ARE THEN CALCULATED AS FOLLOWS -

$$a_i = 1 - \frac{\Delta t^2}{2r^3}$$

$$b_i = \Delta t \left(1 - \frac{\Delta t^2}{6R^3}\right)$$

WHERE

$\Delta t = t_i - t_s$

t_s IS THE SOLUTION TIME FOR THE NORMAL EUQATION TABLE BEING USED

$R^3 = r_i^3$ OR $R^3 = r_0^3$ DEPENDING ON WHETHER m IS ZERO OR NON-ZERO.

NEXT, THE CALCULATIONS WHICH ARE INDEPENDENT OF THE OBSERVATION ARE MADE -

$$n = n + 1$$

$$\sum W_i a_i^2 = \sum W_i a_i^2 + W_i a_i^2$$

$$\sum W_i b_i^2 = \sum W_i b_i^2 + W_i b_i^2$$

$$\sum W_i a_i b_i = \sum W_i a_i b_i + W_i a_i b_i$$

A CHANGE OF VARIABLES IS INTRODUCED BEFORE THE TERMS INVOLVING THE INDIVIDUAL COMPONENTS OF THE OBSERVATION ARE COMPUTED - $\xi_i = x_i - a_i x_0 - b_i \dot{x}_0$
THEN

$$\sum W_i a_i \xi_i = W_i a_i \xi_i + \sum W_i a_i \xi_i$$

$$\sum W_i b_i \xi_i = W_i b_i \xi_i + \sum W_i b_i \xi_i$$

$$\sum W_i \xi_i^2 = W_i \xi_i^2 + \sum W_i \xi_i^2$$

SIMILARLY FOR η AND ζ
CONTROL IS TRANSFERED BACK TO THE CALLING PROGRAM.

USAGE

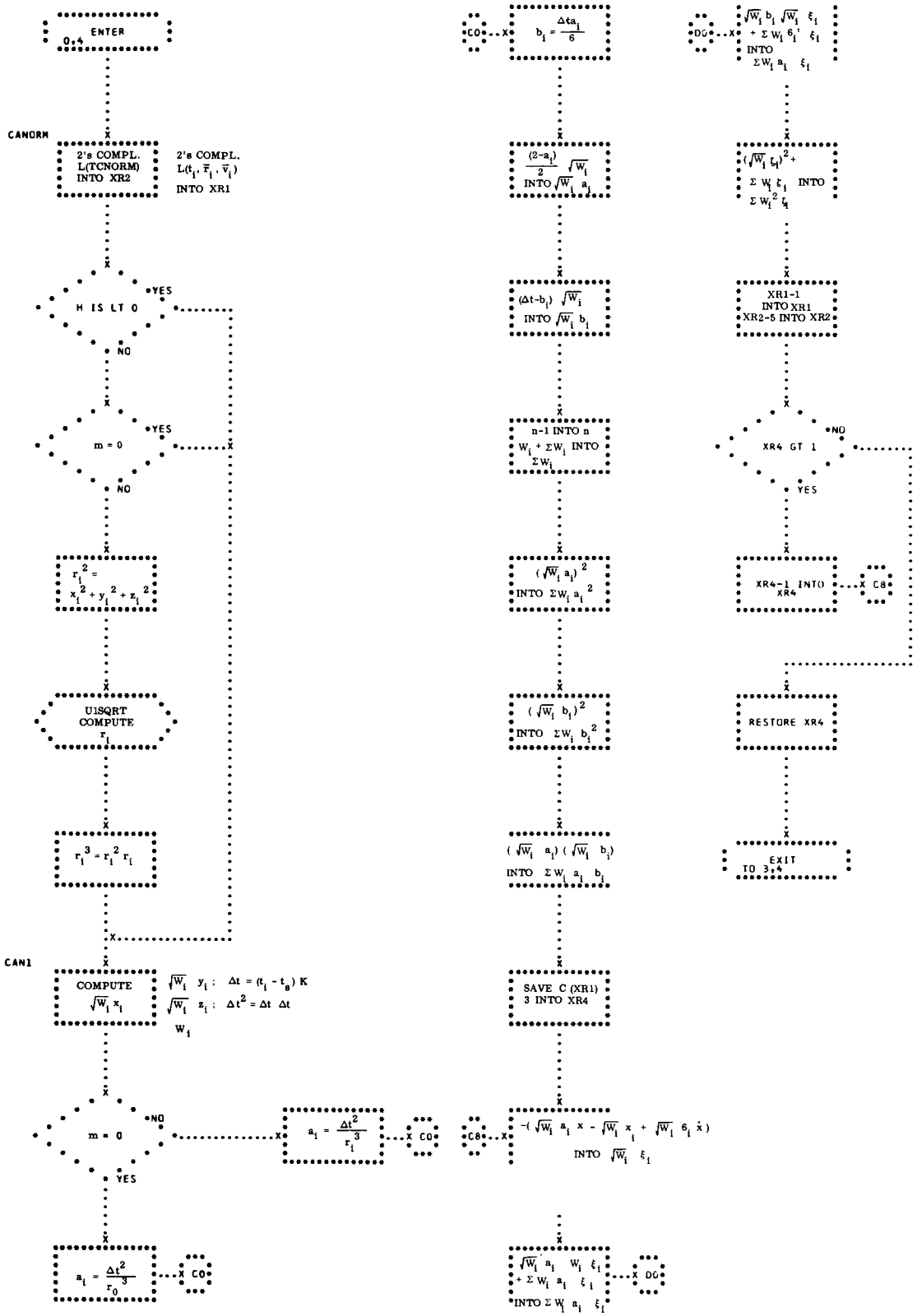
CAMPSO IS ENTERED THREE TIMES FOR EACH OBSERVATION BEING INCORPORATED INTO THE SOLUTION IN ORDER TO UPDATE EACH NORMAL EQUATION TABLE. ON THE FIRST ENTRY, H SHOULD BE SET POSITIVE AND IT SHOULD BE SET NEGATIVE ON EACH OF THE TWO SUBSEQUENT ENTRIES UPON EACH PASS THROUGH THE GENERAL SHORT ARC PROGRAM.

A) CALLING SEQUENCE -

```
TSX       CANORM,4
H         L(TCNORM)
PZE       L(TCOXYZ)
RETURN
```

B) STORAGE REQUIRED - 109 DECIMAL LOCATIONS

C) TIMING - 1.757 MILLISECONDS


ENTER
0,4
CANORM
2's COMPL. L(TCNORM) INTO XR2
2's COMPL. L(t_i, $\bar{r}_i$, $\bar{v}_i$) INTO XR1
H IS LT 0
YES
NO
m = 0
YES
NO
$r_i^2 = x_i^2 + y_i^2 + z_i^2$
U1SQRT COMPUTE r_i
$r_i^3 = r_i^2 r_i$
CAN1
COMPUTE $\sqrt{W_i}\, x_i$
$\sqrt{W_i}\, y_i$: $\Delta t = (t_i - t_s)\, K$
$\sqrt{W_i}\, z_i$: $\Delta t^2 = \Delta t\ \Delta t$
W_i
m = 0
NO
YES
$a_i = \frac{\Delta t^2}{r_i^3}$
CO
$a_i = \frac{\Delta t^2}{r_0^3}$
CO
CO
$b_i = \frac{\Delta t a_i}{6}$
$\frac{(2-a_i)}{2}\sqrt{W_i}$ INTO $\sqrt{W_i}\, a_i$
$(\Delta t - b_i)\sqrt{W_i}$ INTO $\sqrt{W_i}\, b_i$
n-1 INTO n
$W_i + \Sigma W_i$ INTO ΣW_i
$(\sqrt{W_i}\, a_i)^2$ INTO $\Sigma W_i a_i^2$
$(\sqrt{W_i}\, b_i)^2$ INTO $\Sigma W_i b_i^2$
$(\sqrt{W_i}\, a_i)(\sqrt{W_i}\, b_i)$ INTO $\Sigma W_i a_i b_i$
SAVE C (XR1) 3 INTO XR4
C8
$-(\sqrt{W_i}\, a_i\, x - \sqrt{W_i}\, x_i + \sqrt{W_i}\, \theta_i\, \dot{x})$ INTO $\sqrt{W_i}\, \xi_i$
$\sqrt{W_i}\, a_i\ W_i\, \xi_i + \Sigma W_i\, a_i\, \xi_i$ INTO $\Sigma W_i\, a_i\, \xi_i$
DO
DO
$\sqrt{W_i}\, b_i \sqrt{W_i}\, \xi_i + \Sigma W_i\, \theta_i^1\, \xi_i$ INTO $\Sigma W_i\, a_i\, \xi_i$
$(\sqrt{W_i}\, \zeta_i)^2 +$ $\Sigma W_i\, \xi_i$ INTO $\Sigma W_i^2\, \zeta_i$
XR1-1 INTO XR1
XR2-5 INTO XR2
XR4 GT 1
NO
YES
XR4-1 INTO XR4
C8
RESTORE XR4
EXIT TO 3,4

23. CASOLN - SUBROUTINE TO SOLVE NORMAL EQUATIONS

THIS PROGRAM IS USED TO SOLVE A SET OF NORMAL EQUATIONS FOR POSITION AND VELOCITY VECTORS AND THEIR VARIANCES AT TIME t_s. THESE NORMAL EQUATIONS CONSIST OF SUMS OF SQUARES AND CROSS PRODUCTS COMPUTED BY CANORM. IT IS A SUBROUTINE OF THE GENERAL SHORT ARC PROGRAM, CASARC. (SEE FLOWCHART)

INPUT

A. CONSTANTS -

K002.0	DEC 2.0

B. TABLES -

(SEE CASARC FOR FORMAT DESCRIPTION)

TCNORM	A TWENTY-FOUR LOCATION TABLE CONTAINING THE PARAMETERS AND SUMS FOR NORMAL EQUATION M, M = 1, 2 OR 3
TCSRVT	A THIRTEEN-LOCATION OUTPUT TABLE CONTAINING THE SOLUTION t_s, $\bar{r}_s$, $\bar{v}_s$ AND ITS ASSOCIATED VARIANCES.

OUTPUT

THE OUTPUT FROM CASOLN WILL BE FOUND IN TABLE TCSRVT AS DESCRIBED IN CASARC.

METHOD

UPON GAINING CONTROL FROM THE CALLING PROGRAM, CASOLN PICKS UP THE LOCATION OF THE NORMAL EQUATION BEING SOLVED AND ALSO THE LOCATION OF THE OUTPUT TABLE. USING

THE METHOD OF DETERMINANTS, IT NEXT SOLVES THREE SETS OF TWO SIMULTANEOUS NORMAL EQUATIONS, EACH WITH THE SAME COEFFICIENT MATRIX, TO FIND ξ_0, $\dot{\xi}_0$, η_0, $\dot{\eta}_0$, and ζ_0, $\dot{\zeta}_0$.
WE HAVE THREE SETS OF NORMAL EQUATIONS -

$$\sum W_i a_i^2 \xi_0 + \sum W_i a_i b_i \dot{\xi}_0 = \sum W_i a_i \xi_i$$

$$\sum W_i a_i b_i \xi_0 + \sum W_i b_i^2 \dot{\xi}_0 = \sum W_i b_i \xi_i$$

SIMILARLY FOR η AND ζ
BY THE METHOD OF DETERMINANTS -

$$\xi_0 = \frac{\sum W_i b_i^2 \sum W_i a_i \xi_i - \sum W_i a_i b_i \sum W_i a_i \xi_i}{D}$$

$$\dot{\xi}_0 = \frac{\sum W_i a_i^2 \sum W_i b_i \xi_i - \sum W_i a_i b_i \sum W_i a_i \xi_i}{D}$$

WHERE $$D = \sum W_i a_i^2 \sum W_i b_i^2 - \left[\sum W_i a_i b_i\right]^2$$

HE SOLUTION OBTAINED IS -

$$x_s = x_0 + \xi_0$$

$$\dot{x}_s = \dot{x}_0 + \dot{\xi}_0$$

SIMILARLY FOR y_s AND z_s

SUFFICIENT SUMS ARE TAKEN TO OBTAIN THE VARIANCES BY THE FOLLOWING FORMULAS -

$$\sigma_{x_s}^2 = S_{\xi_i}^2 \frac{\sum W_i b_i^2}{D}$$

$$\sigma_{\dot{x}_s}^2 = S_{\xi_i}^2 \frac{\sum W_i a_i^2}{D}$$

SIMILARLY FOR $\sigma_{y_s}^2$ AND $\sigma_{z_s}^2$

WHERE

$$S_{\xi_0}^2 = \frac{1}{n-2} \sum W_i \xi_i^2 - \xi_0 \sum W_i a_i \xi_i - \dot{\xi}_0 \sum W_i b_i \xi_i$$

SIMILARLY FOR $S_{\eta_0}^2$ AND $S_{\eta_0}^2$

IF THERE IS NO APPROXIMATE SOLUTION AVAILABLE, THE VARIANCES ARE USELESS SINCE ALL SIGNIFICANT DIGITS ARE LOST AS THE CHANGE OF VARIABLES IN CANORM IS INEFFECTIVE. AN EXIT IS THEN MADE BACK TO THE CALLING PROGRAM.

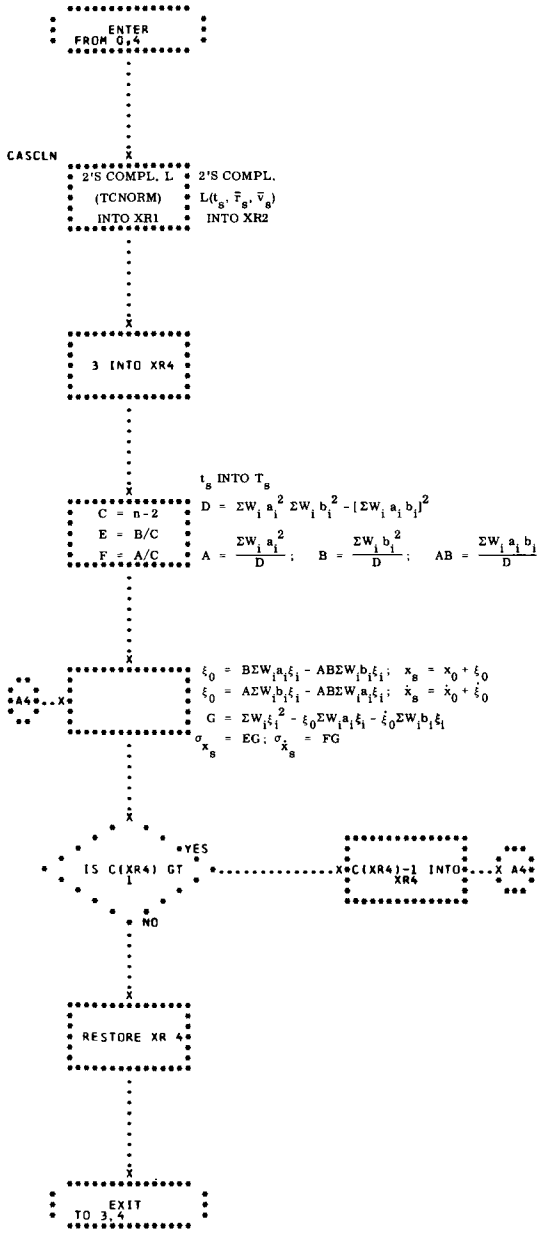
USAGE

A. CALLING SEQUENCE

```
TSX         CASOLN,4
PZE         L(TCNORM)        M = 1,2 OR 3
PZE         L(TCSRVT)
RETURN
```

B. STORAGE REQUIRED - LOCATIONS

C. TIMING - .831 MILLISECONDS


ENTER FROM Q,4
CASCLN
2'S COMPL. L (TCNORM) INTO XR1
2'S COMPL. L(t_s, r̄_s, v̄_s) INTO XR2
3 INTO XR4
t_s INTO T_s
D = ΣW_i a_i^2 ΣW_i b_i^2 - [ΣW_i a_i b_i]^2
C = n-2
E = B/C
F = A/C
A = ΣW_i a_i^2 / D ; B = ΣW_i b_i^2 / D ; AB = ΣW_i a_i b_i / D
A44
ξ_0 = BΣW_i a_i ξ_i - ABΣW_i b_i ξ_i ; x_s = x_0 + ξ_0
ξ_0 = AΣW_i b_i ξ_i - ABΣW_i a_i ξ_i ; ẋ_s = ẋ_0 + ξ̇_0
G = ΣW_i ξ_i^2 - ξ_0 ΣW_i a_i ξ_i - ξ̇_0 ΣW_i b_i ξ_i
σ_xs = EG ; σ_ẋs = FG
IS C(XR4) GT 1
YES
C(XR4)-1 INTO XR4
X A4
NO
RESTORE XR 4
EXIT TO 3,4

24. CLSTOI - LOCAL TO INERTIAL COORDINATE CONVERSION

THIS PROGRAM IS USED TO CONVERT ONE OR MORE SETS OF LOCAL COORDINATES OF POSITION, r(RANGE), a(AZIMUTH), e(ELEVATION) TO INERTIAL COORDINATES OF POSITION x, y, z. (SEE FLOWCHART.)

INPUT

A) CONSTANTS -

K00000	ZERO
K00001	FIXED POINT ONE
K00003	FIXED POINT THREE
K001.0	FLOATING POINT ONE
K.0MTS	ROTATIONAL VELOCITY OF THE EARTH IN FLOATING POINT RADIANS PER TENTH OF SECOND, .72921151 X 10^{-5}
KLAMDO	INERTIAL LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH
KGUFTT	NUMBER OF FEET PER GEMINI LENGTH UNIT IN FLOATING POINT, .20925738 X 10 OCTAL
KCH233	OCTAL MASK FOR FLOATING POINT CHARACTERISTIC 233000000000
KCH206	OCTAL MASK FOR FLOATING POINT CHARACTERISTIC 206000000000

K77777	OCTAL MASK TO KEEP ADDRESS 000000C77777

B) PARAMETERS -

$L[L(r_f)]$	THE LOCATION OF THE LOCATION OF THE RANGE COORDINATE OF THE FIRST POINT TO BE CONVERTED
$L[L(r_L)]$	THE LOCATION OF THE LOCATION OF THE RANGE COORDINATE OF THE LAST POINT TO BE CONVERTED.

C) TABLES -

A DATA TABLE OF THE FORM -

L(DATA TABLE)	t	TIME OF FIRST OBSERVATION
+1	r_1	RANGE (FIRST OBSERVATION)
+2	a_1	AZIMUTH (FIRST OBSERVATION)
+3	e_1	ELEVATION (FIRST OBSERVATION)
+4	r_2	
+5	a_2	
+6	e_2	
+		
•		
•		
•		
+		
+n-2	r_n	RANGE (n^{th} OBSERVATION)
+n-1	a_n	AZIMUTH (n^{th} OBSERVATION)
+n	e_n	ELEVATION (n^{th} OBSERVATION)

WHERE TIME IS IN FIXED POINT TENTHS OF SECONDS GMT, B35, RANGE IS IN FIXED POINT FEET, B35, AND AZIMUTH AND ELEVATION ARE IN FIXED POINT RADIANS, B14. t IS THE FIRST OBSERVATION AND SUBSEQUENT OBSERVATIONS FOLLOW AT 0.1 SECOND INTERVALS.

STATION CHARACTERISTICS TABLE -

A COMPLETE DESCRIPTION OF THE STATION CHARACTERISTICS TABLE IS NOT GIVEN HERE, BUT MAY BE FOUND IN OTHER DOCUMENTAION. ONLY THE LOCATION USED BY CLSTOI ARE DISCRIBED BELOW.

L(STATICN CHARACTERISTICS TABLE)

+5	GEODETIC LONGITUDE OF THE STATION IN FLOATING POINT RADIANS, λ_{STA}
+8	SIN ϕ IN FLOATING POINT RADIANS
+9	COS ϕ IN FLOATING POINT RADIANS
+27	R SIN (ϕ - ϕ') IN FLOATING POINT GEMINI LENGTH UNITS
+28	R COS (ϕ - ϕ') IN FLOATING POINT GEMINI LENGTH UNITS

WHERE ϕ IS THE GEODETIC LATITUDE OF THE RADAR, ϕ' IS THE GEOCENTRIC LATITUDE OF THE PROJECTION OF THE RADAR ONTO THE EARTH, AND R IS THE RADIUS OF THE EARTH MEASURED FROM THE CENTER OF THE EARTH TO THE PROJECTION OF THE RADAR ONTO THE EARTH.

D) SUBROUTINE -

UISICO - LIBRARY FLOATING POINT SINE-COSINE ROUTINE

OUTPUT

THERE IS NO DIRECT OUTPUT FROM CLSTOI, THE INERTIAL COORDINATES x_i, y_i, z_i REPLACE THE LOCAL COORDINATES IN THE INPUT DATA TABLE. INERTIAL COORDINATES ARE IN FLOATING POINT GEMINI LENGTH UNITS.

METHOD

WHEN CONTROL IS TRANSFERRED TO CLSTOI, THE CONTENTS OF THE INDEX REGISTERS 1,2,3 AND 4 ARE SAVED. THE LOCATION OF THE RANGE COORDINATE OF THE FIRST POINT TO BE CONVERTED IS PICKED UP, AND THE NUMBER OF POINTS TO BE CONVERTED IS DETERMINED USING THE FOLLOWING FORMULA -

$$N_p = - \frac{L(r_f) - L(r_L)}{3} + 1$$

WHERE N_p IS THE NUMBER OF POINTS TO BE CONVERTED

$L(r_f)$ IS THE LOCATION OF THE RANGE COORDINATE OF THE FIRST POINT TO BE CONVERTED.

$L(r_L)$ IS THE LOCATION OF THE RANGE COORDINATE OF THE LAST POINT TO BE CONVERTED.

NEXT, THE LOCATION OF THE STATION CHARACTERISTICS TABLE IS PICKED UP, AND USING THE FOLLOWING FORMULA, THE TIME OF THE FIRST POINT TO BE CONVERTED IS DETERMINED -

$$T = \frac{L(r_f) - L(\text{Data Table}) - 1}{3}$$

THE POINTS IN THE DATA TABLE ARE CONVERTED ONE AT A TIME UNTIL THE LAST POINT SPECIFIED HAS BEEN CONVERTED. THERE IS AN ERROR RETURN FROM CLSTOI IF AN ERROR CONDITION OCCURS WITHIN A LIBRARY SUBROUTINE.

THE RADAR COORDINATES ARE RECEIVED IN THE FOLLOWING FORM-

r	RANGE IN FIXED POINT FEET, B35
a	AZIMUTH IN RADIANS, B14
e	ELEVATION IN RADIANS, B14

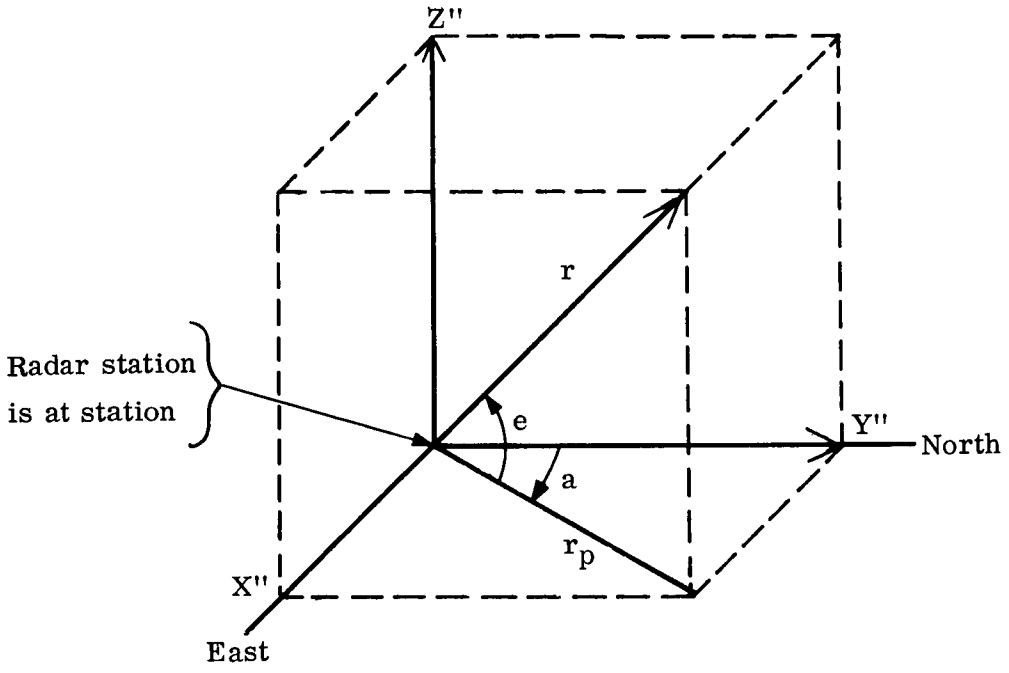


Figure 1

IN FIGURE 1 -

r_ρ IS THE PROJECTION OF $\bar{r}$ THE RADIUS VECTOR, INTO THE LOCAL HORIZONTAL PLANE x", y" PLANE)

a IS MEASURED CLOCKWISE FROM NORTH TO r_p IN THE LOCAL HORIZONTAL PLANE

e IS MEASURED FROM r_p TO $\bar{r}$ -

NOW

$$r_p = r \cos e$$
$$x'' = r_p \cos (90^o - a) = r \cos e \sin a$$
$$y'' = r_p \sin (90^o - a) = r \cos e \cos a$$
$$z'' = r \sin e$$

THE LOCAL RADAR COORDINATES r,a,e ARE CONVERTED TO FLOATING POINT FORMAT. r IS CONVERTED TO GEMINI LENGTH UNITS AND THE ABOVE FORMULAS ARE APPLIED TO OBTAIN LOCAL x'',y'',z'' IN GEMINI LENGTH UNITS. NEXT, THE ORIGIN OF THE LOCAL x'',y'',z'' SYSTEM IS TRANSLATED TO THE CENTER OF THE EARTH.

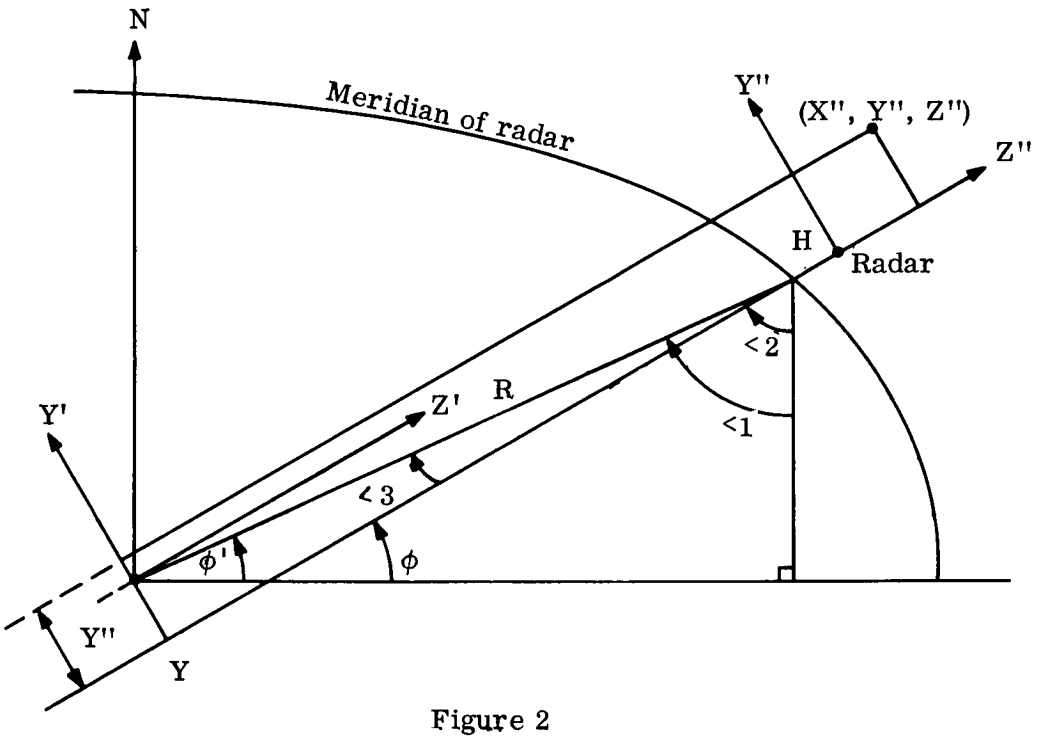


Figure 2

FIGURE 2 IS A CROSS-SECTION OF THE EARTH AT THE MERIDIAN OF THE RADAR WHERE -

ϕ IS THE GEODETIC LATITUDE OF THE RADAR

ϕ' IS THE GEOCENTRIC LATITUDE OF THE PROJECTION OF THE RADAR ONTO THE EARTH ALONG THE LOCAL VERTICAL

R IS THE RADIUS OF THE EARTH MEASURED FROM THE CENTER OF THE EARTH TO THE PORJECTION OF THE RADAR ONTO THE EARTH ALONG THE LOCAL VERTICAL

H IS THE HEIGHT OF THE RADAR ABOVE THE EARTH

FROM FIGURE 2, WE CAN SEE THAT

$$< 1 = 90^{o} - \phi'$$
$$< 2 = 90^{o} - \phi$$
$$< 3 = < 1 - < 2$$
$$= \phi - \phi'$$

AND ALSO, $x' = x''$

$$y' = y'' - R \sin < 3 = y'' - R \sin (\phi - \phi')$$
$$z' = z'' + H + R \cos < 3 = z'' + R \cos (\phi - \phi') + H$$

NOW, THIS GEOCENTRIC SYSTEM MUST BE CONVERTED TO THE GEMINI INERTIAL SYSTEM. THIS REQUIRES TWO ROTATIONS. THE FIRST IS A CLOCKWISE ROTATION ABOUT THE x' AXIS THROUGH AN ANGLE OF $(90^{o}-\phi)$. THIS CAUSES THE NEW z AXIS TO HAVE THE SAME ORIENTATION AS THE EARTH'S ROTATIONAL AXIS (FIGURE 2). THIS ROTATION IS EQUIVALENT TO A POSITIVE ROTATION ABOUT THE x AXIS OF $-(90^{o}-\phi)$, DENOTED $[-(90^{o}-\phi)]$.

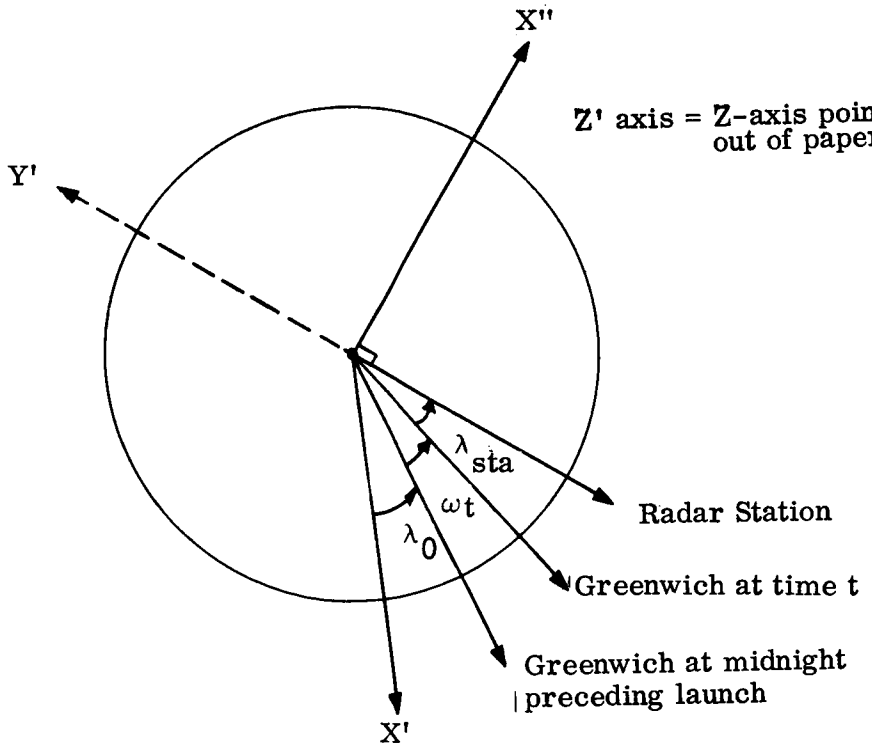


Figure 3

IN FIGURE 3 -

λ_{STA} IS THE GEODETIC LONGITUDE OF THE RADAR STATION

λ_0 IS THE INERTIAL LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH (KLAMDO)

ω IS THE ROTATIONAL VELOCITY OF THE EARTH IN RADIANS PER TENTH OF SECOND (K.UMTS)

t IS THE TIME OF POINT BEING CONVERTED IN TENTHS OF SECONDS, GMT.

THE x' AXIS IS OFFSET FROM THE x AXIS BY AN ANGLE OF $90^{\circ}+\lambda$, WHERE $\lambda = \lambda_0 + \lambda_{STA} + wt$ (FIGURE 3). THEREFORE, IT IS NECESSARY TO MAKE THE SECOND ROTATION, CLOCKWISE, ABOUT THE z AXIS THROUGH $90^{\circ} + \lambda$ WHICH IS EQUIVALENT OT A POSITIVE ROTATION ABOUT THE z AXIS OF $-(90^{\circ}+\lambda)$, DENOTED $R_z\ [-(90^{\circ}+\lambda)]$.

THUS,

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R_z[-(90^{\circ}+\lambda)]\ R_x[-(90^{\circ}-\phi)] \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos(90^{\circ}+\lambda) & -\sin(90^{\circ}+\lambda) & 0 \\ \sin(90^{\circ}+\lambda) & \cos(90^{\circ}+\lambda) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(90^{\circ}-\phi) & -\sin(90^{\circ}-\phi) \\ 0 & \sin(90^{\circ}-\phi) & \cos(90^{\circ}-\phi) \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}$$

(c.f. Danby, J. M. A.; Fundamentals of Celestial Mechanics, The Macmillan Company, New York, 1962, pg. 329)

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -\sin\lambda & -\cos\lambda & 0 \\ \cos\lambda & -\sin\lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \sin\phi & -\cos\phi \\ 0 & \cos\phi & \sin\phi \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}$$

(c.f. Burington, Richard Stevens, Handbook of Mathematical Tables and Formulas, Handbook Publishers, Inc., Sandusky, Ohio, p. 17)

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -\sin\lambda & -\cos\lambda\sin\phi & \cos\lambda\cos\phi \\ \cos\lambda & -\sin\lambda\sin\phi & \sin\lambda\cos\phi \\ 0 & \cos\phi & \sin\phi \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}$$

$$x = -x'\sin\lambda - y'(\cos\lambda\sin\phi) + z'(\cos\lambda\cos\phi)$$
$$y = x'\cos\lambda - y'(\sin\lambda\sin\phi) + z'(\sin\lambda\cos\phi)$$
$$z = y'(\cos\phi) + z'(\sin\phi)$$

$$x = -x'\sin\lambda + \cos\lambda(-y'\sin\phi + z'\cos\phi)$$
$$y = x'\cos\lambda + \sin\lambda(-y'\sin\phi + z'\cos\phi)$$
$$z = y'\cos\phi + z'\sin\phi$$

USAGE -

A) CALLING SEQUENCE -

```
TSX     CLSTOI,4
PZE  L(DATA TABLE),,L(STATION CHARACTERISTICS
                          TABLE)
PZE  L[L(r_f)],,L[L(r_L)]
ERROR RETURN
NORMAL RETURN
```

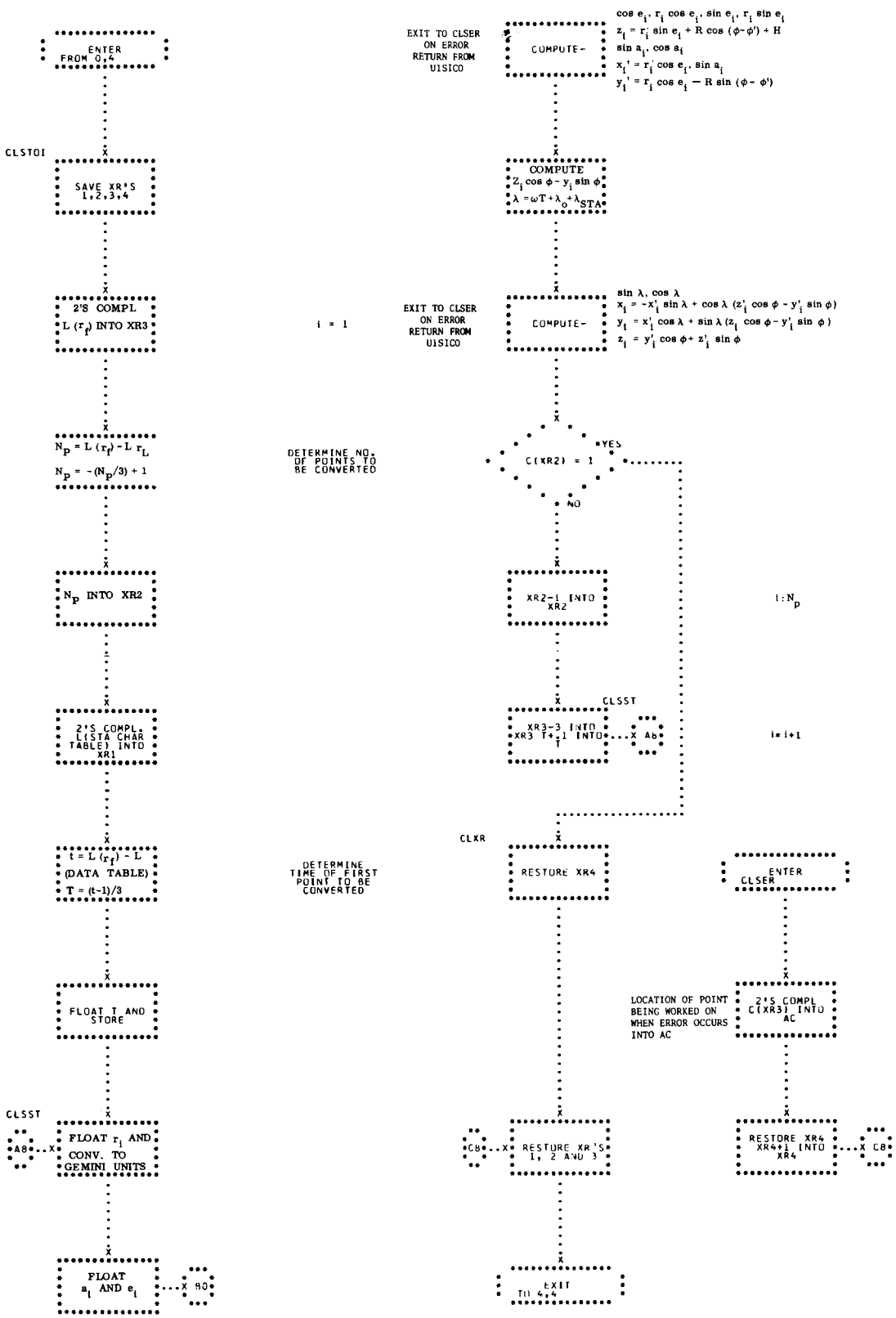
B) STORAGE REQUIRED -

131 LOCATIONS

C) TIME REQUIRED -

.0932+.583 P

WHERE P=THE NUMBER OF POINTS TO BE CONVERTED PER ENTRY


ENTER FROM 0,4
CLSTOI
SAVE XR'S 1,2,3,4
2'S COMPL L (r_f) INTO XR3
$N_P = L(r_f) - L\ r_L$
$N_P = -(N_P/3) + 1$
N_P INTO XR2
2'S COMPL. L(STA CHAR TABLE) INTO XR1
$t = L(r_f) - L$ (DATA TABLE)
$T = (t-1)/3$
FLOAT T AND STORE
CLSST
A8
FLOAT r_i AND CONV. TO GEMINI UNITS
FLOAT a_i AND e_i
GO
EXIT TO CLSER ON ERROR RETURN FROM UISICO
COMPUTE-
$\cos e_i$, $r_i \cos e_i$, $\sin e_i$, $r_i \sin e_i$
$z_i = r_i \sin e_i + R \cos(\phi - \phi') + H$
$\sin a_i$, $\cos a_i$
$x_i' = r_i' \cos e_i$, $\sin a_i$
$y_i' = r_i \cos e_i - R \sin(\phi - \phi')$
COMPUTE
$Z_i \cos \phi - y_i \sin \phi$
$\lambda = \omega T + \lambda_o + \lambda_{STA}$
i = 1
EXIT TO CLSER ON ERROR RETURN FROM UISICO
COMPUTE-
$\sin \lambda$, $\cos \lambda$
$x_i = -x_i' \sin \lambda + \cos \lambda (z_i' \cos \phi - y_i' \sin \phi)$
$y_i = x_i' \cos \lambda + \sin \lambda (z_i \cos \phi - y_i' \sin \phi)$
$z_i = y_i' \cos \phi + z_i' \sin \phi$
DETERMINE NO. OF POINTS TO BE CONVERTED
C(XR2) = 1
YES
NO
XR2-1 INTO XR2
i:N_p
CLSST
XR3-3 INTO XR3
T+1 INTO I
A8
i = i+1
CLXR
DETERMINE TIME OF FIRST POINT TO BE CONVERTED
RESTORE XR4
C8
RESTORE XR'S 1, 2 AND 3
EXIT TO 4,4
ENTER CLSER
LOCATION OF POINT BEING WORKED ON WHEN ERROR OCCURS INTO AC
2'S COMPL C(XR3) INTO AC
RESTORE XR4 XR4+1 INTO XR4
C8

25. CSOGLS - GENERAL LEAST SQUARES SMOOTHING

THE BASIC FUNCTION OF THIS PROGRAM IS TO PRODUCE SMOOTHED POSITION AND VELOCITY VECTORS ($\bar{r}_k, \bar{v}_k$) ASSOCIATED WITH TIME t_k. (SEE FLOWCHART)

INPUT

A) CONSTANTS -

KM03.0	DEC	3.0	FLOATING POINT MINUS THREE
KM30.0	DEC	30.0	FLOATING POINT MINUS THIRTY
K001.0	DEC	1.0	FLOATING POINT ONE
K002.0	DEC	2.0	FLOATING POINT TWO
K003.0	DEC	3.0	FLOATING POINT THREE
K004.0	DEC	4.0	FLOATING POINT FOUR
K009.0	DEC	9.0	FLOATING POINT NINE
K015.0	DEC	15.0	FLOATING POINT FIFTEEN

B) TABLES - (H IS PRESENTLY EITHER F, FOR FPS-16 DATA, OR V, FOR VERLORT DATA.)

TCHSPT — A 12-LOCATION TABLE CONTAINING THE PARAMETERS USED BY THE SMOOTHING PROGRAM (SEE FLOWCHART)

TCHSWT — A 52-LOCATION TABLE USED BY THE SMOOTHING PROGRAM FOR THE INTERNAL DOUBLE PRECISION COMPUTATIONS (SEE FLOWCHART)

TCBDF6 — A 601-LOCATION TABLE CONTAINING THE FPS-16 RADAR OBSERVATIONS IN THE

		REQUIRED FORMAT FOR THE SMOOTHING PROGRAM. THE FIRST CELL CONTAINS THE TIME OF THE FIRST ENTRY IN THE TABLE IN FIXED POINT TENTHS OF SECONDS B35 (SEE FLOWCHART)
	TCBOVE	A 601-LOCATION TABLE CONTAINING THE VERLORT RADAR OBSERVATIONS IN THE REQUIRED FORMAT FOR THE SMOOTHING PROGRAM. THE FIRST CELL CONTAINS THE TIME OF THE FIRST ENTRY IN THE TABLE IN FIXED POINT TENTHS OF SECONDS B35 (SEE FLOWCHART)

C) SUBROUTINES -

CS1GLS	INITIALIZATION SUBROUTINE FOR CSOGLS
CS2GLS	RECURSIVE SUBROUTINE FOR CSOGLS
CS3GLS	SOLUTION GENERATING SUBROUTINE FOR CSOGLS

OUTPUT

OUTPUT FROM CSOGLS IS DESCRIBED UNDER METHOD AND, MORE SPECIFICALLY, IN THE WRITEUPS OF CS1GLS, CS2GLS, AND CS3GLS.

METHOD

WHEN CONTROL IS TRANSFERRED TO CSOGLS, THE CONTENTS OF INDEX REGISTERS A, B, AND C AND OF THE SENSE INDICATORS ARE SAVED. THE CALLING SEQUENCES TO SUBROUTINES CS1GLS, CS2GLS, AND CS3GLS ARE INITIALIZED AND THE ADDRESSES OF THE WORKING TABLE AND THE PARAMETER TABLE FOR THIS RADAR ARE OBTAINED. IF THIS IS NOT AN INITIAL OR RESTART ENTRY TO SMOOTHING, THE REMAINING PROCEDURES ARE AS OUTLINED IN STEP B), BELOW.

A) ON AN INITIAL OR RESTART ENTRY, THE INDICATOR IS RESET TO ZERO TO SHOW NO SOLUTION AVAILABLE AND A CHECK IS MADE TO DETERMINE IF THERE ARE POINTS AVAILABLE FOR USE BY SMOOTHING. IF NO POINTS ARE PRESENT, SMOOTHING EXITS TO ITS SUFFIX, MFBDAS. WHEN POINTS ARE AVAILABLE, THE FIRST OR RESTART ENTRY CELL IS SET NONZERO, SHOWING THAT THE ENTRY HAS BEEN MADE - THE SETUP COMPLETED CELL IS RESET TO ZERO SHOWING THAT THE SETUP HAS NOT YET BEEN DONE, AND THE LOCATION OF THE FIRST COMPONENT, MIDPOINT OF U, V, W SUMS IS INITIALIZED. THE QUANTITIES n, n+1, k, AND Δt IN SINGLE PRECISION ARE THEN MOVED FROM THE PARAMETER TABLE TO THE WORKING TABLE IN DOUBLE PRECISION.

CSOGLS THEN COMPUTES THE A_{ij} FACTORS IN DOUBLE PRECISION AND STORES THEM IN THE WORKING TABLE. THE FOLLOWING FORMULAS ARE USED -

$$A_{11} = (9n(n+1)-3-15k^2)/(2n+3)(4n^2-1)$$
$$A_{12} = 3k/n(n+1)(2n+1)$$
$$A_{13} = (45k^2-15n(n+1))/n(n+1)(2n+3)(4n^2-1)$$
$$A_{21} = -30k/\Delta t(2n+3)(4n^2-1)$$
$$A_{22} = 3/\Delta t(n)(n+1)(2n+1)$$
$$A_{23} = 90k/\Delta t(n)(2n+3)(n+1)(4n^2-1)$$

B) IF THERE ARE NOT 2n+1 POINTS AVAILABLE FOR USE BY SMOOTHING, CSOGLS EXITS TO MFBDAS. AS SOON AS 2n+1 POINTS ARE PRESENT IN THE DATA TABLE, CS1GLS IS ENTERED TO COMPLETE THE SETUP OF THE INITIAL U, V, W SUMS.

IF MORE POINTS REMAIN IN THE DATA TABLE, THEN U, V, W SUMS ARE UPDATED BY CS2GLS USING THE RECURSIVE SCHEME. WHEN ALL THE DATA HAS BEEN INCORPORATED INTO THE U, V, W SUMS, CS3GLS PRODUCES THE OUTPUT $\bar{r}_k$ AND $\bar{v}_k$ ASSOCIATED WITH TIME t_k .

THE CONTENTS OF THE INDEX REGISTERS AND OF SENSE INDICATORS ARE THEN RESTORED AND CSOGLS EXITS TO ITS SUFFIX, MFBDAS.

USAGE

CSOGLS RECEIVES CONTROL FROM ITS PREFIX, MPBDAS, AND GIVES CONTROL TO ITS SUFFIX, MFBDAS.

A) STORAGE REQUIRED - 156 LOCATIONS

B) TIMING (7094) - DOES NOT INCLUDE TIMING FOR SUBROUTINES

FIRST ENTRY, $k = 0$ - 1.116 + (.012) P MILLISECONDS

FIRST ENTRY, $k \neq 0$ - 1.640 + (.012) P MILLISECONDS

SUBSEQUENT ENTRY - .068 + (.012) P MILLISECONDS, WHERE P IS THE NUMBER OF POINTS TO BE ADDED TO THE U, V, W SUMS.

C) CSOGLS AND ITS SUBROUTINES RECEIVE THEIR INPUT AND PRODUCE THEIR OUTPUT IN SINGLE PRECISION. HOWEVER, ALL INTERNAL COMPUTATIONS ARE MADE IN DOUBLE PRECISION.

D) ALL A_{ij} FACTORS ARE REQUIRED TO BE LESS THAN 1. BECAUSE OF THIS STIPULATION, THE MINIMUM VALUE THAT MAY BE USED FOR n IS 3.

TABLE 25-1. SMOOTHING PROGRAM PARAMETERS*

LOCATION		DESCRIPTION
TCHSPT		SOLUTION AVAILABLE INDICATOR, S
	+1	FIRST OR RESTART ENTRY INDICATOR, F
	+2	LOCATION OF TABLE CONTAINING DATA TO BE SMOOTHED
	+3	LOCATION OF FIRST COMPONENT, FIRST POINT USED, $L(x_f)$
	+4	LOCATION OF FIRST COMPONENT, LAST POINT USED, $L(x_1)$
	+5	LOCATION OF SMOOTHING WORKING TABLE
	+6	3n (FIXED POINT)
	+7	k (FIXED POINT)
	+8	n (FLOATING POINT)
	+9	n+1 (FLOATING POINT)
	+10	k (FLOATING POINT)
	+11	Δt TIME INTERVAL BETWEEN ENTRIES IN THE DATA TABLE (FLOATING POINT)

* H IS PRESENTLY EITHER F, FOR FPS-16 DATA, OR V, FOR VERLORT DATA.

TABLE 25-2. SMOOTHING PROGRAM WORKING TABLE ENTRIES (SHEET 1 OF 3)*

IN THIS TABLE, ALL ENTRIES, UNLESS OTHERWISE SPECIFIED, ARE IN FLOATING POINT DOUBLE PRECISION - M = MOST SIGNIFICANT PORTION OF ENTRY. L = LEAST SIGNIFICANT PORTION OF ENTRY.

LOCATION		DESCRIPTION/SIGNIFICANCE	
TCHSWT		SETUP FINISHED INDICATOR (OCTAL)	
	+1	LOCATION OF FIRST COMPONENT, MIDPOINT OF U,V, W SUMS (OCTAL), $L(x_0)$	
	+2	n	M
	+3	n	L
	+4	n+1	M
	+5	n+1	L
	+6	k	M
	+7	k	L
	+8	t	M
	+9	t	L
	+10	A_{11}	M
	+11	A_{11}	L
	+12	A_{12}	M
	+13	A_{12}	L
	+14	A_{13}	M
	+15	A_{13}	L
	+16	A_{21}	M
	+17	A_{21}	L
	+18	A_{22}	M
	+19	A_{22}	L

*H IS PRESENTLY EITHER F, FOR FPS-16 DATA, OR V FOR VERLORT DATA.

TABLE 25-2. SMOOTHING PROGRAM WORKING TABLE ENTRIES (SHEET 2 OF 3)*

LOCATION	DESCRIPTION/SIGNIFICANCE	
+20	A_{23}	M
+21	A_{23}	L
+22	U_X	M
+23	U_X	L
+24	V_X	M
+25	V_X	L
+26	W_X	M
+27	W_X	L
+28	U_Y	M
+29	U_Y	L
+30	V_Y	M
+31	V_Y	L
+32	W_Y	M
+33	W_Y	L
+34	U_Z	M
+35	U_Z	L
+36	V_Z	M
+37	V_Z	L
+38	W_Z	M
+39	W_Z	L
+40	$*T_1$	M
+41	T_1	L
+42	T_2	M
+43	T_2	L
+44	T_3	M

*TEMPORARY STORAGE

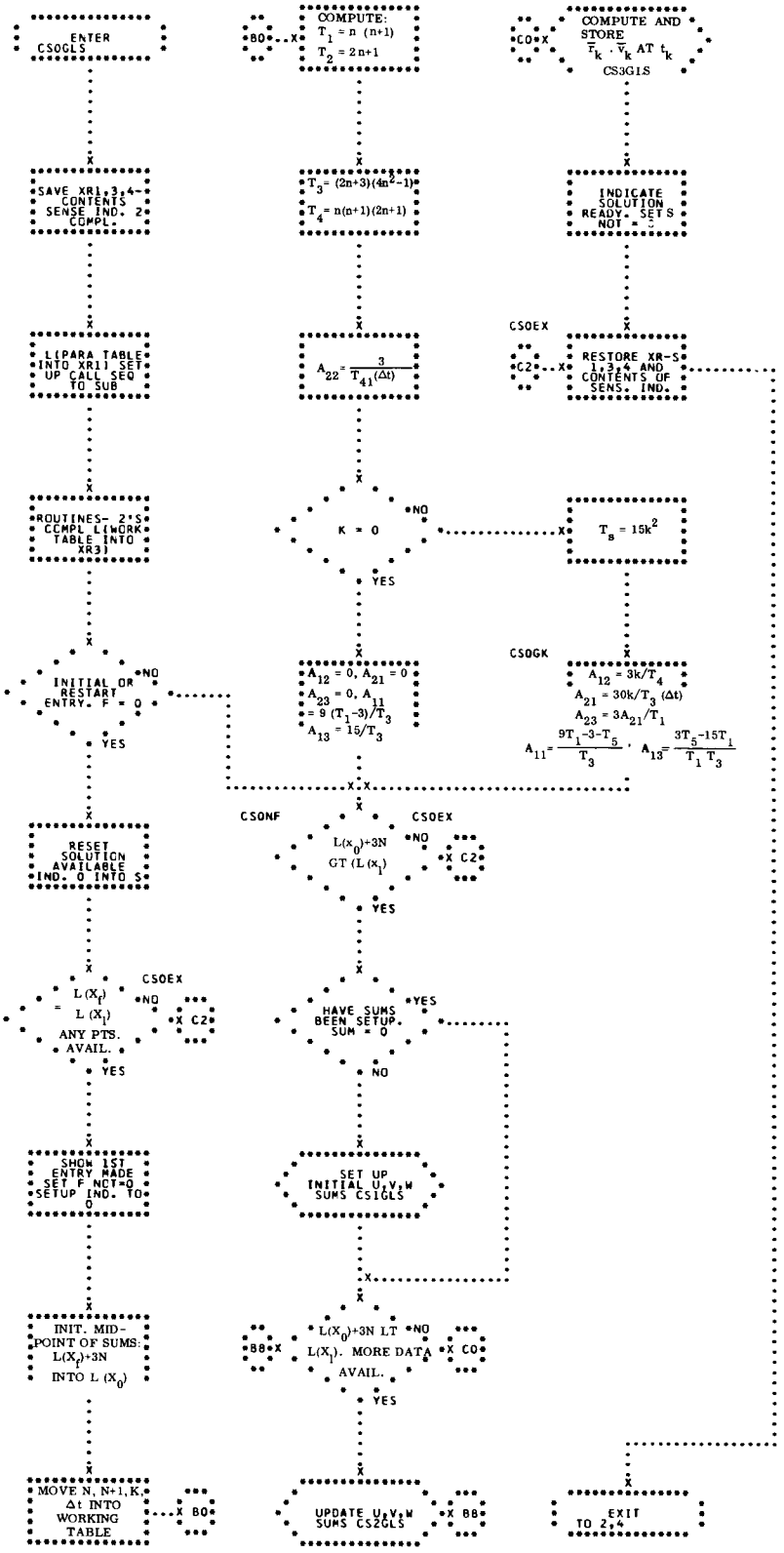
TABLE 25-2. SMOOTHING PROGRAM WORKING TABLE ENTR
(SHEET 3 OF 3)*

LOCATION	DESCRIPTION/SIGNIFICANCE	
+45	T_3	L
+46	T_4	M
+47	T_4	L
+48	T_5	M
+49	T_5	L
+50	T_6	M
+51	T_6	L

TABLE 25-3. SMOOTHING INPUT DATA TABLE

LOCATION	DESCRIPTION	
TCBDF6 CR		
TCBDVE	t (FIXED POINT TENTHS OF SECONDS, B35)	
+1	C_1	P_1
+2	C_2	
+3	C_3	
+4	C_1	P_2
+5	C_2	
+6	C_3	
•		
•		
•		
+598	C_1	P_{200}
+599	C_2	
+600	C_3	

THE ABOVE IS THE GENERAL FORMAT OF THE INPUT DATA TABLE (S). THE THREE COORDINATES OF THE POINTS OVER WHICH SMOOTHING IS TO TAKE PLACE, THE P_i, ARE DENOTED BY C_1, C_2 AND C_3. IN THE GEMINI SYSTEM, THESE COORDINATES ARE THE X, Y, AND Z COMPONENTS OF THE POSITION VECTOR AND ARE IN FLOATING POINT GEMINI LENGTH UNITS.


ENTER CSOGLS
SAVE XR1,3,4 CONTENTS SENSE IND. 2 COMPL.
L(PARA TABLE INTO XR1) SET UP CALL SEQ TO SUB
ROUTINES- 2'S COMPL L(WORK TABLE INTO XR3)
INITIAL OR RESTART ENTRY. F = 0
YES
NO
RESET SOLUTION AVAILABLE IND. 0 INTO S
L(X_f) = L(X_1) ANY PTS. AVAIL.
CSOEX
C2
SHOW 1ST ENTRY MADE SET F NOT=0 SETUP IND. TO 0
INIT. MID-POINT OF SUMS: L(X_f)+3N INTO L(X_0)
MOVE N, N+1, K, Δt INTO WORKING TABLE
B0
COMPUTE: T_1 = n(n+1) T_2 = 2n+1
T_3 = (2n+3)(4n^2-1)
T_4 = n(n+1)(2n+1)
A_22 = 3/T_41(Δt)
K = 0
A_12 = 0, A_21 = 0
A_23 = 0, A_11 = 9(T_1-3)/T_3
A_13 = 15/T_3
CSONF
L(X_0)+3N GT (L(X_1)
HAVE SUMS BEEN SETUP. SUM = 0
SET UP INITIAL U,V,W SUMS CS1GLS
L(X_0)+3N LT L(X_1). MORE DATA AVAIL.
B8
C0
UPDATE U,V,W SUMS CS2GLS
B8
COMPUTE AND STORE r̄_k · v̄_k AT t_k
CS3G1S
C0
INDICATE SOLUTION READY. SETS NOT = 0
CSOEX
C2
RESTORE XR-S 1,3,4 AND CONTENTS OF SENS. IND.
T_5 = 15k^2
CSOGK
A_12 = 3k/T_4
A_21 = 30k/T_3 (Δt)
A_23 = 3A_21/T_1
A_11 = (9T_1-3-T_5)/T_3 , A_13 = (3T_5-15T_1)/(T_1 T_3)
EXIT TO 2,4

GENERAL DESCRIPTION OF THE GENERAL LEAST SQUARES SMOOTHING PROGRAM

GIVEN -

A) 2N+1 SEQUENTIAL DATA POINTS, Q_1 WITH UNIFORM TIME SPACEING, Δt.

B) t_0, THE TIME ASSOCIATED WITH THE MIDPOINT OF THE DATA SPAN OVER WHICH TO SMOOTH.

C) k, A PARAMETER SPECIFYING THE NUMBER OF POINTS FROM THE MIDPOINT AT WHICH THE SOLUTION IS DESIRED.

FIND -

Q_k AND $\dot{Q}_k$ ASSUMING A SECOND DEGREE CURVE FIT TO THE DATA.

FORMULATION -

A) LET THE SECOND DEGREE CURVE FITTING THE DATA BE-

1) $$Q_i = a_0 + a_1 (i\Delta t)/(\Delta t) + a_2 (i\Delta t)^2/(\Delta t)^{2*}$$

IN THIS EXPRESSION, $(i\Delta t)$ IS THE INDEPENDENT VARIABLE, AND Δt IS A CONSTANT INTRODUCED IN THE DENOMINATION TO SIMPLIFY CALCULATIONS.

B) THE DERIVATIVE OF 1) WITH RESPECT TO $(i\Delta t)$ YIELDS THE GENERAL EXPRESSION FOR VELOCITY -

2) $$Q_i = a_1/(\Delta t) + 2a_2 (i\Delta t)/(\Delta t)^2$$

C) ELIMINATING Δt FROM 1) AND LETTING i VARY FROM -n TO +n, WE OBTAIN 2n+1 SIMULTANEOUS EQUATIONS OF THE FORM -

$$Q_i = a_0 + a_1 i + a_2 i^2$$

(i.e., $a_0 + a_1 (-n) + a_2 (-n)^2 = Q_{-n}$

$a_0 + a_1 (-n+1) + a_2 (-n+1)^2 = Q_{-n+1}$

$\vdots \qquad\qquad \vdots$

$a_0 + a_1 (0) + a_2 (0)^2 = Q_0$

$a_0 + a_1 (1) + a_2 (1)^2 = Q_1$

$\vdots \qquad\qquad \vdots$

$a_0 + a_1 (n) + a_2 (n)^2 = Q_n$), from which the unknown a_0, a_1, and a_2 must be obtained.

D) BY PREMULTIPLYING BOTH SIDES OF THE ARRAY BY T TRANSPOSE OF THE COEFFICIENT MATRIX, THE ARRAY IS REDUCED TO THE FOLLOWING -

$$\begin{bmatrix} 2n+1 & 0 & \sum_{-n}^{n} i^2 \\ 0 & \sum_{-n}^{n} i^2 & 0 \\ \sum_{-n}^{n} i^2 & 0 & \sum_{-n}^{n} i^4 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum_{-n}^{n} Q_i \\ \sum_{-n}^{n} iQ_i \\ \sum_{-n}^{n} i^2 Q_i \end{bmatrix} = \begin{bmatrix} U \\ V \\ W \end{bmatrix}$$

E) LET THE LEFT HAND MATRIX, WHICH REPRESENTS THE COEFFICIENT MATRIX PREMULTIPLIED BY ITS TRANSPOSE, BE DESIGNATED AS P. THE INVERSE OF THIS MATRIX, P^{-1}, MAY BE EXPRESSED AS

$$P^{-1} = \frac{\tilde{P}}{|P|}$$

WHERE $\tilde{P}$ IS THE ADJOINT OF P. TAKEN TERM BY TERM, THE ADJOINT IS

$$\tilde{p}_{ij} = (-1)^{i+j} \ |M_{ji}|,$$

WHERE M_{ji} IS THE MATRIX OBTAINED BY DELETING THE j^{TH} ROW AND i^{TH} COLUMN OF P.

$$\tilde{P} = \begin{bmatrix} \sum i^2 \sum i^4 & 0 & -\left(\sum i^2\right)^2 \\ 0 & (2n+1)\sum i^4 - \left(\sum i^2\right)^2 & \\ -\left(\sum i^2\right)^2 & 0 & (2n+1)\sum i^2 \end{bmatrix}$$

$$|P| = \sum i^2 \left[(2n+1)\sum i^4 - \left(\sum i^2\right)^2\right]$$

DIVIDING EACH TERMS IN $\tilde{P}$ BY $|P|$, P^{-1} REDUCES TO A MATRIX OF THE FORM

$$\begin{bmatrix} A & 0 & B \\ 0 & C & 0 \\ B & 0 & D \end{bmatrix}$$

WHERE

$$A = \sum_{-n}^{n} i^4 \Big/ \left[(2n+1) \sum_{-n}^{n} i^4 - \left(\sum_{-n}^{n} i^2 \right)^2 \right]$$

$$B = \sum_{-n}^{n} i^2 \Big/ \left[\left(\sum_{-n}^{n} i^2 \right)^2 - (2n+1) \sum_{-n}^{n} i^4 \right]$$

$$C = 1 \Big/ \sum_{-n}^{n} i^2$$

$$D = (2n+1) \Big/ \left[(2n+1) \sum_{-n}^{n} i^4 - \left(\sum_{-n}^{n} i^2 \right)^2 \right]$$

REFERRING TO EQUATION D), IT IS DESIRABLE TO HAVE ONLY THE UNKNOWNS a_0, a_1, AND a_2, APPEARING ON THE LEFT SIDE OF THE EQUATION. EACH SIDE IS THEREFORE PREMULTIPLIED BY P^{-1}, WHICH GIVES -

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} A & 0 & B \\ 0 & C & 0 \\ B & 0 & D \end{bmatrix} \begin{bmatrix} U \\ V \\ W \end{bmatrix}$$

F) USING THE RELATIONSHIPS -

$$\sum_{-n}^{n} i^2 = (n/3)(n+1)(2n+1), \text{ AND } \sum_{-n}^{n} i^4 = (n/15)(n+1)(2n+1)\ (3n^2 + 3n-1)$$

IT IS POSSIBLE TO EXPRESS A,B,C, AND D IN TERMS ONLY OF n. (FOR DERIVATIONS OF THE ABOVE, SEE DERIVATION OF EXPRESSIONS FOR $\sum_{-n}^{n} i^2$ and $\sum_{-n}^{n} i^4$

THUS -

A = $(9n(n+1)-3)/(\ (2n+3)(4n^2-1)\ )$

B = $-15/(2n+3)(4n^2-1)$

C = $3/n(2n+1)(n+1)$

D = $45/n(n+1)(2n+3)(4n^2-1)$

G) THEN, SOLVING THE MATRIX IN E) -

$a_0 = AU + BW$

$a_1 = CV$

$a_2 = BU + DW$

SUBSTITUTING THESE VALUES IN THE ORIGINAL EQUATIONS A) AND B), WE OBTAIN -

$$Q_k = (AU + BW) + CV(k) + (BU + DW)(k^2)$$

$$= (A + Bk^2)U + (Ck)V + (B + Dk^2)W$$

$$\dot{Q}_k = CV(1/\Delta t) + 2k(BU + DW)/(\Delta t)$$

$$= (2kB/\Delta t)U + (C/\Delta t)V + (2kD/\Delta t)W$$

LETTING -

$$A_{11} = A + Bk^2$$

$$A_{12} = Ck$$

$$A_{13} = B + Dk^2$$

$$A_{21} = 2kB/\Delta t$$

$$A_{22} = C/\Delta t$$

$$A_{23} = 2kD/\Delta t,$$

THE FOLLOWING EQUATIONS ARE OBTAINED -

$$Q_k = A_{11}U + A_{12}V + A_{13}W$$

$$\dot{Q}_k = A_{21}U + A_{22}V + A_{23}W$$

H) IT IS DESIRABLE TO SMOOTH ABOUT THE MIDPOINT (k = 0), SINCE THE EQUATIONS ARE THEN SIMPLIFI AND THE PROBABLE ERROR IS LEAST.

I) THE COEFFICIENTS OF U, V, AND W(I.E., THE A_{ij} FACTORS) ARE FUNCTIONS OF n, k, AND Δt ONLY AND NOT OF THE POINTS BEING SMOOTHED. THUS, ONCE THESE PARAMETERS ARE KNOWN, THE A_{ij} FACTORS CAN BE EVALUATED AT ONCE AND IT IS NECESSARY ONLY TO DETERMINE THE U, V, AND W SUMS. BECAUSE OF THEIR FORM, THE U, V, W SUMS LEND THEMSELVES READILY TO A RECURSIVE SCHEME WHEREBY THEY CAN BE UPDATED BY ADDING THE EFFECTS OF THE NEWEST POINT AND SUBTRACTING THE EFFECTS OF THE OLDEST ONE. THE RECURSIVE FORMULAS AND THEIR DERIVATION ARE GIVEN BELOW -

1) $$U_0 = \sum_{-n}^{n} Q_i$$

$$= \sum_{-n+1}^{n+1} Q_{i-1}$$

$$= \left[\sum_{-n}^{n} Q_{i-1}\right] + Q_n - Q_{n-1}$$

$$= U_{-1} + Q_n - Q_{n-1}$$

2) $$V_0 = \sum_{-n}^{n} iQ_i$$

$$= \sum_{-n+1}^{n+1} (i-1)Q_{i-1}$$

$$= \left[\sum_{-n}^{n} (i-1)Q_{i-1}\right] + nQ_n - (-n-1)Q_{-n-1}$$

$$= \left[\sum_{-n}^{n} iQ_{i-1}\right] - \left[\sum_{-n}^{n} Q_{i-1}\right] + nQ_n + (n+1)Q_{-n-1}$$

$$= V_{-1} - U_{-1} + nQ_n + (n+1)Q_{-n-1}$$

3) $$W_0 = \sum_{-n}^{n} i^2 Q_i$$

$$= \sum_{-n+1}^{n+1} (i-1)^2 Q_{i-1}$$

$$= \left[\sum_{-n}^{n} (i-1)^2 Q_{i-1}\right] + n^2 Q_n - (-n-1)^2 Q_{-n-1}$$

$$= \left[\sum_{-n}^{n} i^2 Q_{i-1}\right] - 2\left[\sum_{-n}^{n} iQ_{i-1}\right] + \left[\sum_{-n}^{n} Q_{i-1}\right] + n^2 Q_n - (n+1)^2 Q_{-n-1}$$

$$= W_{-1} - 2V_{-1} + U_{-1} + n^2 Q_n - (n+1)^2 Q_{-n-1}$$

DERIVATION OF EXPRESSIONS FOR $\sum_{-n}^{n} i^2$ AND $\sum_{-n}^{n} i^4$

A) UPPER FACTORIALS - $i^{(k)}$

1) $i^{(k)} = i(i-1)\ (i-2)\ldots(i-(k-1))$

2) $\sum_{i=a}^{b-1} i^{(n)} = (b^{(n+1)} - a^{(n+1)})/(n+1)$

3) USING SYNTHETIC DIVISION AND THE ABOVE DEFINITION OF $i^{(k)}$ IT IS POSSIBLE TO EXPRESS ANY POLYNOMIAL IN TERMS OF UPPER FACTORIALS- GIVEN - $P(x) = x^4 + 6x^2 - 7x + 8$

USING DIVISORS 0, 1, 2, AND 3 (SINCE THIS IS A QUARTIC) -

1	1	0	6	-7	8
		1	1	7	
2	1	1	7	0	
		2	6		
3	1	3	13		
		3			
	1	6			

THE RESULTS OF THE SYNTHETIC DIVISION MAY BE EXPRESSED AS

$$P(x) = [\{[(x-3)+6](x-2)+13\}(x-1)+0](x-0)+8.$$

UPON MULTIPLYING AND REARRANGING TERMS, THE FOLLOWING IS OBTAINED -

$$P(x) = x(x-1)(x-2)(x-3) + 6x(x-1)(x-2) + 13x(x-1) + 8.$$

THEREFORE, $P(x) = x^{(4)} + 6x^{(3)} + 13x^{(2)} + 8$

(THIS EXPRESSION MAY BE CHECKED BY SUBSTITUTING FOR EACH FACTORIAL THE DEFINITION GIVEN IN A)1).)

B) $\sum_{-n}^{n} i^2$

1) $\sum_{-n}^{n} i^2 = 2 \sum_{1}^{n} i^2$

2) EXPRESSING i^2 IN TERMS OF UPPER FACTORIALS -

$P(i) = i^2$

USING DIVISORS 0 AND 1 (SINCE THIS IS A SECOND DEGREE EQUATION) -

$$\begin{array}{r|l} 1 & 1 \quad 0 \quad 0 \\ & \quad 1 \\ \hline & 1 \quad 1 \end{array}$$

Therefore, $P(i) = i^{(2)} + i^{(1)}$

3) $\sum_{-n}^{n} i^2 = 2\sum_{1}^{n} i^2 = 2\sum_{1}^{n}(i^{(2)} + i^{(1)}) = 2\sum_{1}^{n} i^{(2)} + 2\sum_{1}^{n} i^{(1)}$

4) $\sum_{1}^{n} i^{(2)} = ((n+1)^{(3)} - 1^{(3)})/3$ (See a)2).) (See A)2).)

$1^{(3)} = 0$

Thus, $\sum_{1}^{n} i^{(2)} = ((n+1)^{(3)})/3 = ((n+1)(n)(n-1))/3$ (See A)1).)

5) $\sum_{1}^{n} i^{(1)} = ((n+1)^{(2)} - 1^{(2)})/2$ (See A)2).)

$1^{(2)} = 0$

Therefore, $\sum_{1}^{n} i^{(1)} = ((n+1)^{(2)})/2 = ((n+1)(n))/2$ (See A)1).)

6) Combining the results of 4) and 5):

$$\sum_{-n}^{n} i^2 = 2\sum_{1}^{n} i^{(2)} + 2\sum_{1}^{n} i^{(1)}$$

$$= 2((n+1)(n)(n-1))/3 + 2((n+1)(n))/2$$

$$= ((n)(n+1)(2n+1))/3$$

c) $\sum_{-n}^{n} i^4$

1) $\sum_{-n}^{n} i^4 = 2\sum_{1}^{n} i^4$

2) Expressing i^4 in terms of upper factorials: $P(i) = i^4$.

With divisors of 0, 1, 2, and 3 (since this is a quartic equation):

$$
\begin{array}{r|cccc|c}
1 & 1 & 0 & 0 & 0 & 0 \\
 & & 1 & 1 & 1 & \\
\hline
2 & 1 & 1 & 1 & 1 & \\
 & & 2 & 6 & & \\
\hline
3 & 1 & 3 & 7 & & \\
 & & 3 & & & \\
\hline
 & 1 & 6 & & &
\end{array}
$$

Therefore, $P(i)' = i^{(4)} + 6i^{(3)} + 7i^{(2)} + i^{(1)}$

3) $$\sum_{-n}^{n} i^4 = 2\sum_{1}^{n} i^4 = 2\sum_{1}^{n}(i^{(4)} + 6i^{(3)} + 7i^{(2)} + i^{(1)})$$

$$= 2\sum_{1}^{n} i^{(4)} + 12\sum_{1}^{n} i^{(3)} + 14\sum_{1}^{n} i^{(2)} + 2\sum_{1}^{n} i^{(1)}$$

4) Noting that for $q > 1$, $1^{(q)} = 0$,

$$\sum_{1}^{n} i^{(4)} = ((n+1)^{(5)} - 1^{(5)})/5$$

$$= ((n+1)(n)(n-1)(n-2)(n-3))/5 \qquad \text{(See A)1).)}$$

$$\sum_{1}^{n} i^{(3)} = ((n+1)^{(4)} - 1^{(4)})/4$$

$$= ((n+1)(n)(n-1)(n-2))/4 \qquad \text{(See A)1).)}$$

$$\sum_{1}^{n} i^{(2)} = ((n+1)^{(3)} - 1^{(3)})/3$$

$$= ((n+1)(n)(n-1))/3 \qquad \text{(See A)1).)}$$

$$\sum_{1}^{n} i^{(1)} = ((n+1)^{(2)} - 1^{(2)})/2$$

$$= ((n+1)(n))/2 \qquad \text{(See A)1).)}$$

5) Combining the results of 4):

$$\sum_{-n}^{n} i^4 = 2\sum_{1}^{n} i^{(4)} + 12\sum_{1}^{n} i^{(3)} + 14\sum_{1}^{n} i^{(2)} + 2\sum_{1}^{n} i^{(1)}$$

$$= 2((n+1)(n)(n-1)(n-2)(n-3))/5 +$$

$$12((n+1)(n)(n-1)(n-2))/4 + 14((n+1)(n)(n-1))/3$$

$$+ 2((n+1)(n))/2$$

$$= ((n)(n+1)(2n+1)(3n^2 + 3n-1))/15$$

26. CS1GLS - INITIALIZATION SUBROUTINE FOR CSOGLS

THIS PORGRAM COMPUTES THE INITIAL U,V,W SUMS TO BE USED IN THE GENERAL LEAST SQUARES SMOOTHING PROGRAM CSOGLS. IT IS ENTERED THROUGH CSOGLS AFTER AT LEAST 2n+1 POINTS ARE AVAILABLE FOR SMOOTHING. (SEE FLOWCHART.)

INPUT

A) CONSTANTS -

K00000	ZERO
K001.0	FLOATING POINT ONE

B) TABLES -

TCHSPT	A 12-LOCATION TABLE CONTAINING PARAMETERS AND LOCATIONS OF OTHER TABLES TO BE USED IN CS1GLS AND BY CSOGLS AND ITS OTHER SUBROUTINES.
TCHSWT	A 52-LOCATION WORKING TABLE USED FOR DOUBLE PRECISION COMPUTATIONS BY CS1GLS AND BY CSOGLS AND ITS OTHER SUBROUTINES
TCBDVE	DATA TABLES, PRESENTLY 601 LOCATION TABLES
TCBDF6	FOR VERLORT OR FPS-16 DATA. ONLY ONE TABLE IS SPECIFIED PER ENTRY.

IN THE TABLES TCHSPT AND TCHSWT, H IS PRESENTLY F (FOR FPS-16 DATA) OR V (FOR VERLORT DATA). ONLY TCHSPT IS DIRECTLY INPUT TO CS1GLS, AND THE OTHER TABLES ARE PICKED UP FROM THE LOCATIONS SPECIFIED IN TCHSPT. THE ABOVE TABLES ARE USED, IN GENERAL, BY CSOGLS AND ITS SUBROUTINES. (SEE CSOGLS TABLE FORMAT DESCRIPTION)

OUTPUT

THE OUTPUT FROM CS1GLS IS STORED IN TCHSWT. IT CONSISTS OF NINE DOUBLE-PRECISION, FLOATING POINT U,V,W SUMS FOUND IN LOCATION AS SPECIFIED IN THE TABLE FORMAT DESCRIPTICN (SEE FLOWCHARTS)

METHOD

WHEN CONTROL IS TRANSFERRRED TO CS1GLS, THE CONTENTS OF INDEX REGISTERS 1, 2,3,4,5, AND 6 ARE SAVED. THE LOCATIONS OF THE SMOOTHING WORKING TABLE (TCHSWT) AND OF THE X-COMPONENT OF THE MIDPOINT OF THE DATA TABLE (TCBDVE OR TCBDF6) CURRENTLY BEING USED ARE PICKED UP. THE INITIAL SUMS ARE COMPUTED IN DOUBLE-PRECISION, USING 2n+1 POINTS FROM THE SPECIFIED DATA TABLE. UPON EXIT FROM CS1GLS, THE SETUP FINISHED INDICATOR IN THE TCHSWT TABLE IS SET TO NON-ZERO TO SHOW THAT THE INITIAL SUMS HAVE BEEN COMPUTED. THE INDEX REGISTERS ARE RESTORED, AND CONTROL IS TRANSFERRED BACK TO CSOGLS. THE FOLLOWING FORMULAS ARE USED -

$$U_{Q_o} = \sum_{j=-n}^{n} Q_j = Q_o + \sum_{j=1}^{n} (Q_j + Q_{-j})$$

$$V_{Q_o} = \sum_{j=-n}^{n} jQ_j = \sum_{j=1}^{n} j\ (Q_j - Q_{-j})$$

$$W_{Q_o} = \sum_{j=-n}^{n} j^2 Q_j = \sum_{j=1}^{n} j^2 (Q_j - Q_{-j})$$

WHERE Q = X, Y, Z

USAGE

A) CS1GLS IS ENTERD ONLY THROUGH CSOGLS, AND O AFTER $2n+1$ POINTS ARE AVAILABLE IN EITHER T OR TCBOVE. IT IS ENTERED ONCE AFTER EACH R INITIALIZATION OF CSOGLS.

B) CALLING SEQUENCE -

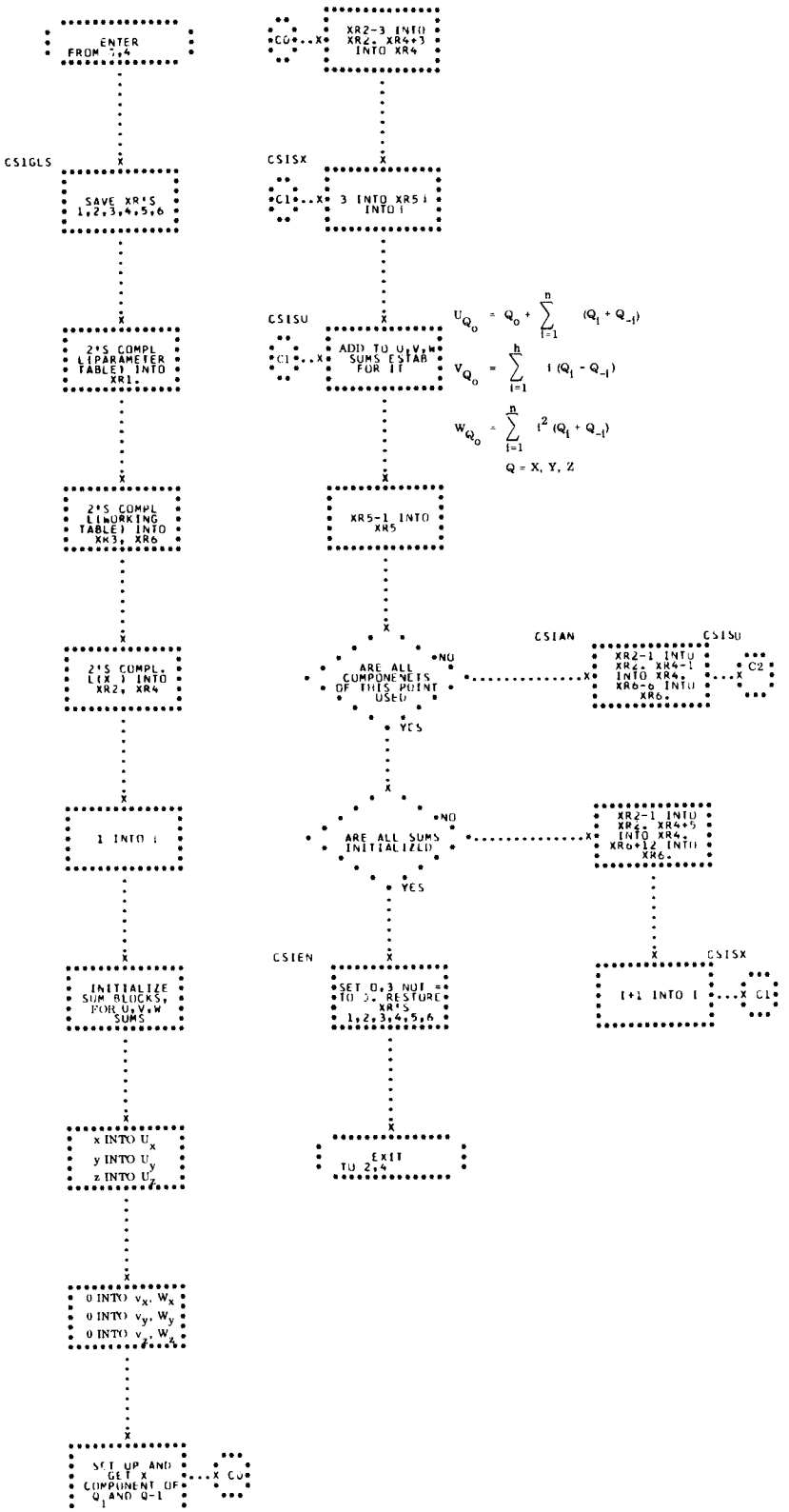
```
TSX     CS1GLS,4
PZE     L(TCHSPT)
RETURN
```

C) STORAGE REQUIRED -

86 LOCATIONS

D) TIMING -

.140+.957N MILLISECONDS WHERE $N = \frac{P-1}{2}$ AND P THE NUMBER OF POINTS TO BE INCLUDED IN THE SUMS.


ENTER FROM 7,4
CSIGLS
SAVE XR'S 1,2,3,4,5,6
2'S COMPL (PARAMETER TABLE) INTO XR1,
2'S COMPL (WORKING TABLE) INTO XR3, XR6
2'S COMPL. L(X) INTO XR2, XR4
1 INTO I
INITIALIZE SUM BLOCKS, FOR U,V,W SUMS
x INTO U_x
y INTO U_y
z INTO U_z
0 INTO v_x, W_x
0 INTO v_y, W_y
0 INTO v_z, W_z
SET UP AND GET X COMPONENT OF Q AND Q-1
C0
C0
XR2-3 INTO XR2, XR4+3 INTO XR4
CSISX
C1
3 INTO XR5 INTO I
CSISU
C1
ADD TO U,V,W SUMS ESTAB FOR IT
XR5-1 INTO XR5
ARE ALL CUMPONENETS OF THIS POINT USED
NO
YES
CSIAN
XR2-1 INTO XR2, XR4-1 INTO XR4, XR6-8 INTO XR6,
CSISU
C2
ARE ALL SUMS INITIALIZED
NO
YES
XR2-1 INTO XR2, XR4+5 INTO XR4, XR6+12 INTO XR6,
CSIEN
SET Q,3 NOT = TO 0. RESTORE XR'S 1,2,3,4,5,6
I+1 INTO I
CSISX
C1
EXIT TO 2,4
$U_{Q_o} = Q_o + \sum_{i=1}^{n} (Q_i + Q_{-i})$
$V_{Q_o} = \sum_{i=1}^{n} i\,(Q_i - Q_{-i})$
$W_{Q_o} = \sum_{i=1}^{n} i^2\,(Q_i + Q_{-i})$
Q = X, Y, Z

27. CS2GLS - RECURSIVE SUBROUTINE FOR CSOGLS

THIS PROGRAM GAINS CONTROL FROM CSOGLS, THE GENERAL LEAST SQUARES SMOOTHING PROGRAM. IT IS ENTERED AS ADDITICNAL DATA IS AVAILABLE IN THE DATA TABLE. USING RECURSIVE FORMULAS, CS2GLS UPDATES THE DOUBLE - PRECISION U,V,W SUMS IN THE TCHSWT TABLE. (SEE FLOWCHART)

INPUT

A) CONSTANTS -

K00003	FIXED POINT THREE
K002.0	FLOATING POINT TWO

B) TABLES -

TCHSPT	A 12-LOCATION TABLE CONTAINING PARAMETERS AND LOCATIONS OF OTHER TABLES TO BE USED IN CS2GLS AND BY CSOGLS AND ITS OTHER SUBROUTINES.
TCHSWT	A 52-LOCATION WORKING TABLE USED FOR DOUBLE PRECISION COMPUTATIONS BY CS2GLS AND BY CSOGLS AND ITS OTHER SUBROUTINES.
TCBDVE TCBDF6	DATA TABLES, PRESENTLY 601 LOCATION TABLES FOR VERLORT OR FPS-16 DATA. ONLY ONE TABLE IS SPECIFIED PER ENTRY.

IN THE TABLES TCHSPT AND TCHSWT, H IS PRESENTLY F (FOR FPS-16 DATA) OR V (FOR VERLORT DATA). AS IN CS1GLS, ONLY TCHSPT IS DIRECTLY INPUT TO THE PROGRAM. OTHER TABLES ARE PICKED UP FROM LOCATIONS SPECIFIED IN TCHSPT. THE ABOVE TABLES ARE USED IN GENERAL, BY CSOGLS AND ITS SUBROUTINES. SEE CSOGLS TABLES 2-1 THROUGH 2-3 FOR

TABLE FORMAT DESCRIPTION.)

OUTPUT

THERE IS NO DIRECT OUTPUT FROM CS2GLS. THE PROGRAM UPDATES THE DOUBLE-PRECISION, FLOATING POINT U,V,W, SUMS IN TCHSWT.

METHOD

WHEN CONTROL IS GIVEN TO CS2GLS, THE CONTENTS OF INDEX REGISTERS 1, 2, 3, 5 AND 6 ARE SAVED. THE LOCATION OF THE SMOOTHING WORKING TABLE (TCHSWT) IS PICKED UP. USING RECURSIVE FORMULAS FOR A SECOND DEGREE GENERAL LEAST SQUARES CURVE FIT THE U,V,W SUMS ARE UPDATED, I.E., A NEW POINT IS ADDED IN AND THE OLDEST POINT IS SUBTRACTED OUT SO THAT THE SUMS STILL INCLUDE ONLY 2n+1 TENTHS OF SECONDS WORTH OF DATA. RECURSIVE COMPUTATIONS ARE CARRIED OUT IN DOUBLE-PRECISION, FLOATING POINT ARITHMETIC. THE INDEX REGISTERS ARE RESTORED AND CONTROL IS TRANSFERRED BACK TO CSOGLS. THE FOLLOWING FORMULAS ARE USED -

$$A = nQ_n$$

$$B = nA$$

$$C = (n+1)Q_{-n-1}$$

$$W_{Q_0} = W_{Q_{-1}} - 2V_{Q_{-1}} + U_{Q_{-1}} + B - (n+1)C \qquad Q = (X, Y, Z)$$

$$V_{Q_0} = V_{Q_{-1}} - U_{Q_{-1}} + A + C$$

$$U_{Q_0} = U_{Q_{-1}} + Q_n - Q_{-n-1}$$

WHERE Q_n IS THE POINT BEING ADDED TO THE U,V,W SUMS AND Q_{-n-1} IS THE POINT BEING SUBTRACTED FROM THE SUMS.

USAGE

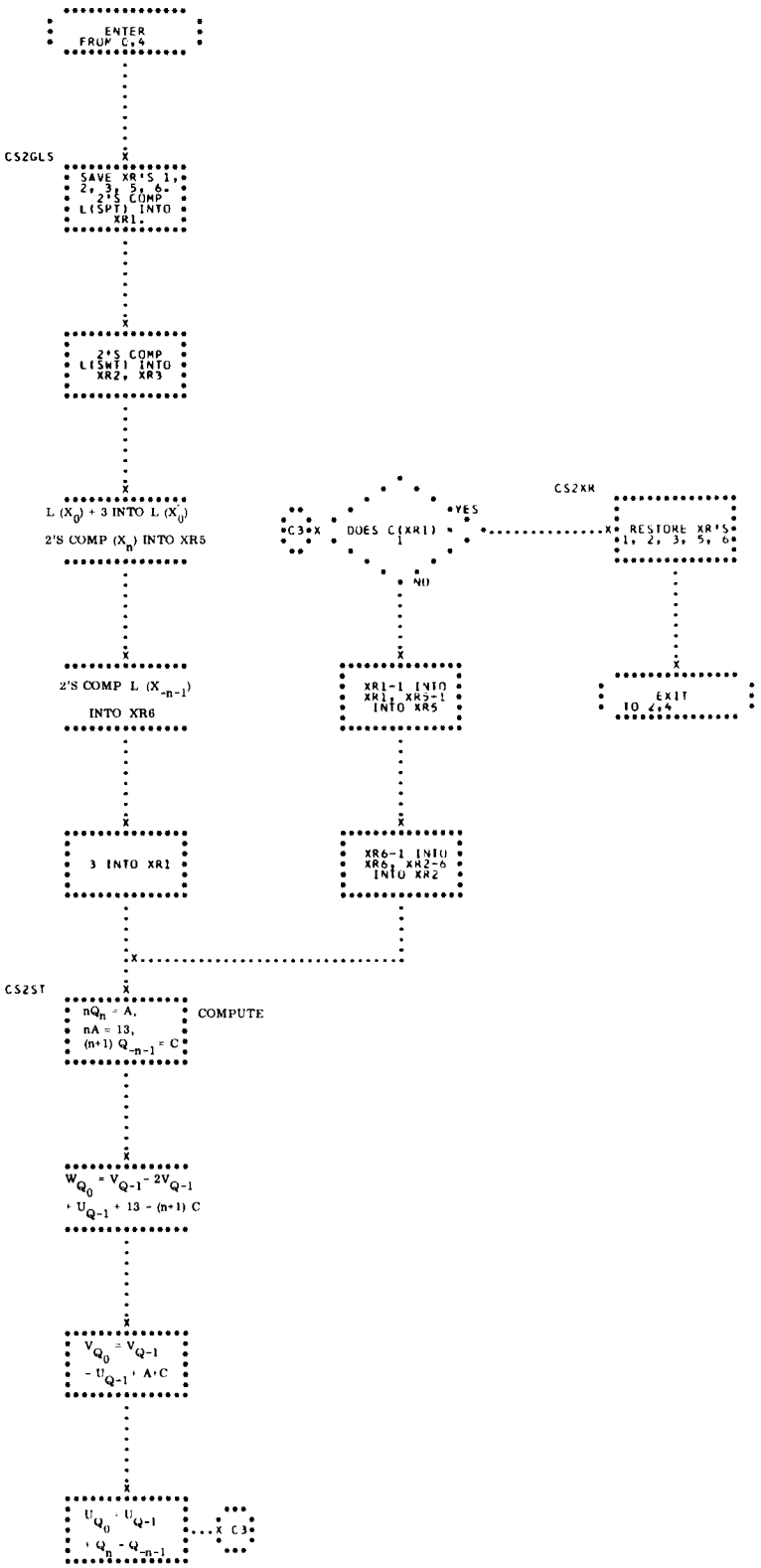
CS2GLS IS ENTERED THROUGH CSOGLS EACH TIME A NEW POINT IS TO BE ADDED OT THE U,V,W SUMS SO THAT A NEW CURVE FIT BASED ON THE LAST CONTINUOUS $2n + 1$ POINTS MAY BE OBTAINED.

A) CALLING SEQUENCE -

```
TSX     CS2GLS,4
PZE     L(TCHSPT)
RETURN
```

B) STORAGE REQUIRED - 64 LOCATIONS

C) TIMING - 1.47 MILLISECONDS


ENTER FROM C;4
CS2GLS
SAVE XR'S 1, 2, 3, 5, 6- 2'S COMP L(SPT) INTO XR1.
2'S COMP L(SWT) INTO XR2, XR3
L (X0) + 3 INTO L (X'0)
2'S COMP (Xn) INTO XR5
2'S COMP L (X-n-1) INTO XR6
3 INTO XR1
C3
DOES C(XR1) = 1
YES
NO
CS2XR
RESTORE XR'S 1, 2, 3, 5, 6
EXIT TO 2;4
XR1-1 INTO XR1, XR5+1 INTO XR5
XR6-1 INTO XR6, XR2-6 INTO XR2
CS2ST
nQn = A, nA = 13, (n+1) Q-n-1 = C
COMPUTE
WQ0 = VQ-1 - 2VQ-1 + UQ-1 + 13 - (n+1) C
VQ0 = VQ-1 - UQ-1 + A+C
UQ0 = UQ-1 + Qn - Q-n-1
C3

28. CS3GLS - SUBROUTINE TO GENERATE SOLUTIONS FOR CSOGLS

THIS PROGRAM IS ENTERED THROUGH CSOGLS, THE GENERAL LEAST SQUARES SMOOTHING PROGRAM, WHEN THERE ARE 2n + 1 DATA POINTS INCLUDED IN THE U, V, W SUMS AND WHEN THE SET UP FINISHED INDICATOR IS SET IN THE SMOOTHING WORKING TABLE. ITS PURPOSE IS TO COMPUTE AN OUTPUT SOLUTION $\overline{r}_k$, $\overline{v}_k$ AT TIME t_k.

INPUT

A) CONSTANTS -

K00001	FIXED POINT ONE
K00003	FIXED POINT THREE
K00022	FIXED POINT TWENTY-TWO
K010.0	FLOATING POINT TEN
K00.UT	NUMBER OF SECONDS PER GEMINI TIME UNIT FLOATING POINT 806.81364
KCH233	MASK FOR FLOATING POINT CHARACTERISTIC 233000000000

B) TABLES -

TCHSPT	A 12-LOCATION TABLE CONTAINING PARAMETERS AND LOCATIONS OF OTHER TABLES TO BE USED BY CS3GLS AND BY CSOGLS AND ITS OTHER SUBROUTINES.
TCHSWT	A 52-LOCATION WORKING TABLE USED FOR DOUBLE-PRECISION COMPUTATIONS BY CS3GLS AND BY CSOGLS AND ITS OTHER SUBROUTINES.
TCBDVE	DATA TABLES, PRESENTLY 601 LOCATION TABLES FOR VERLORT OR FPS-16 DATA.

TCBDF6 ONLY ONE TABLE IS SPECIFIED PER ENTRY.

IN THE TABLES TCHSPT AND TCHSWT, H IS PERSENTLY F FOR FPS-16 DATA OF V FOR VERLORT DATA . ONLY TCHSPT IS DIRECTLY INPUT TO CS3GLS, AND THE OTHER TABLES ARE PICKED UP FROM THE LOCATIONS SPECIFIED IN TCHSPT. THE ABOVE TABLES ARE USED, IN GENERAL, BY CSOGLS AND ITS SUBROUTINES. SEE CSOGLS TABLES 2-1 THRU 2-3 FOR TABLE FORMAT DESCRIPTION.)

OUTPUT

THE OUTPUT IS IN FLOATING POINT IN A 7-LOCATION TABLE HAVING THE FOLLOWING FORMAT -

L(OUTPUT VECTORS)		t_k TIME IN GMT SECS
+1	X	
+2	Y	POSITION VECTOR IN $\overline{r}_k$ GEMINI LENGTH UNITS
+3	Z	
+4	$\dot{X}$	
+5	$\dot{Y}$	VELOCITY VECTOR IN $\overline{v}_k$ GEMINI VELOCITY UNITS
+6	Z	

METHOD

WHEN CONTROL IS TRANSFERRED TO CS3GLS, THE CONTENTS OF INDEX REGISTERS 1, 2, 3 AND 4 ARE SAVED. THE LOCATIONS OF THE SMOOTHING PARAMETER TABLE (TCHSPT), THE SMOOTHING WORKING TABLE (TCHSWT) AND THE OUTPUT TABLE ARE PICKED UP. THE OUTPUT TIME, t_k, IS CALCULATED USING THE FORMULA - $t_k = t + k$ WHERE t IS THE TIME ASSOCIATED WITH THE MIDPOINT OF THE 2n + 1 DATA SPAN CURRENTLY INCOR-

PORATED IN THE U, V, W SUMS, AND k IS A PARAMETER DETERMINING THE NUMBER OF SECONDS AWAY FROM THE MIDPOINT TIME THAT THE SOLUTION WILL BE MADE. THE $\overline{r}$ AND $\overline{v}$ VECTORS ARE CALCULATED IN DOUBLE PRECISION, BUT STORED IN THE OUTPUT TABLE AS SINGLE PRECISION NUMBERS. THE FOLLOWING EQUATIONS ARE USED -

$$Q_k = A_{11} U_{Q_0} + A_{12} V_{Q_0} + A_{13} W_{Q_0}$$

$$\dot{Q}_k = A_{21} U_{Q_0} + A_{22} V_{Q_0} + A_{23} W_{Q_0} \qquad Q = (X, Y, Z)$$

THE CONTENTS OF THE INDEX REGISTERS ARE STORED AND CONTROL IS TRANSFERRED BACK TO CSOGLS.

USAGE

CS3GLS IS ENTERED THROUGH CSOGLS TO OBTAIN AN OUTPUT SOLUTION $t_k, \overline{r}_k, \overline{v}_k$.

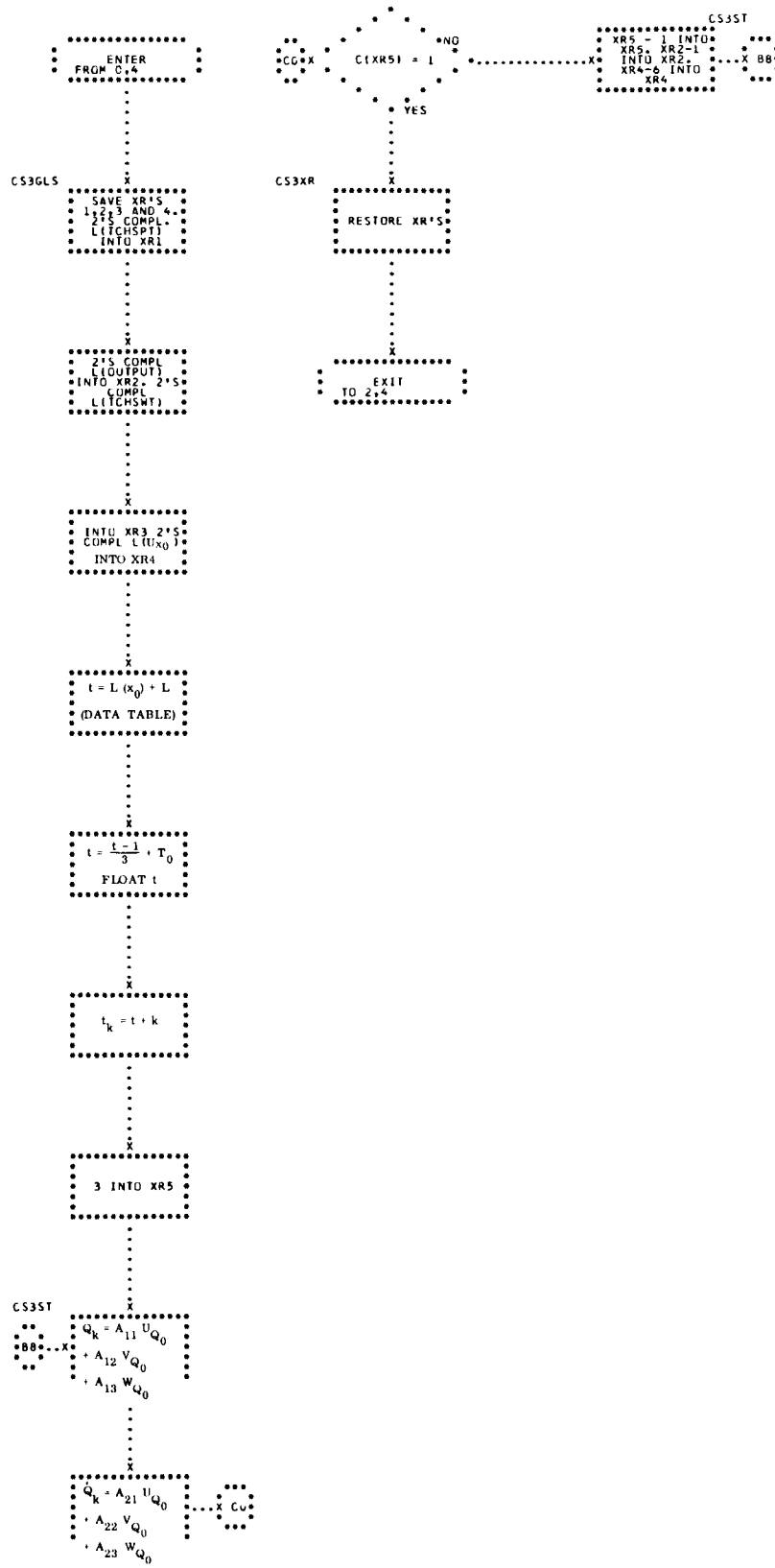
IT IS TO BE ENTERED AFTER 2n + 1 POINTS HAVE BEEN INCORPORATED INTO THE U, V, W SUMS

A) CALLING SEQUENCE -

```
TSX     CS3GLS,4
PZE     L(TSHSPT),,L(OUTPUT t_k, r_k, v_k )
RETURN
```

B) STORAGE REQUIRED - 50 LOCATIONS

C) TIMING - 1.106 MILLISECONDS


ENTER FROM C.4
CS3GLS
SAVE XR'S 1,2,3 AND 4. 2'S COMPL. L(TCHSPT) INTO XR1
2'S COMPL L(OUTPUT) INTO XR2. 2'S COMPL L(TCHSWT)
INTO XR3 2'S COMPL L(U$_{X_0}$) INTO XR4
t = L (x_0) + L (DATA TABLE)
$t = \frac{t-1}{3} + T_0$ FLOAT t
$t_k = t + k$
3 INTO XR5
CS3ST
BB
$Q_k = A_{11} U_{Q_0} + A_{12} V_{Q_0} + A_{13} W_{Q_0}$
$\dot{Q}_k = A_{21} U_{Q_0} + A_{22} V_{Q_0} + A_{23} W_{Q_0}$
CC
C(XR5) = 1
NO
YES
CS3ST
XR5 - 1 INTO XR5. XR2-1 INTO XR2. XR4-6 INTO XR4
BB
CS3XR
RESTORE XR'S
EXIT TO 2.4

29. MFBDAS - MONITOR SUFFIX TO CASARC AND CSOGLS

THIS PROGRAM PLACES THE LATEST VECTORS IN THE BERMUDA VECTOR TABLE AND INDICATES FROM WHICH SOURCE AND PROCESSOR THEY CAME. (SEE FLOWCHART.)

INPUT

A) TABLES -

TCFSPT (OR TCVSPT)	THE FIRST OF THE TWELVE-LOCATION SMOOTHING PARAMETER TABLE WHICH IS SET NON-ZERO WHEN A NEW FPS-16 (VERLORT) SMOOTHED SOLUTION IS AVAILABLE, S_f (OR S_v)
TCNPWC	A ONE-LOCATION TABLE CONTAINING THE SUM OF THE NUMBER OF POINTS (N) AND THE WEIGHT ASSIGNED TO EACH POINT(W) INCORPORATED INTO THE SHORT ARC SOLUTION
TCSRVT	A THIRTEEN-LOCATION TABLE CONTAINING THE VECTORS PRODUCED BY THE SHORT ARC PROGRAM, THEIR ASSOCIATED TIME, AND THE VARIANCES OF EACH COMPONENT
MTFPS (OR MTVER)	A SEVEN-LOCATION TABLE CONTAINING THE SMOOTHED

		FPS-16 (VERLORT) SOLUTION
B)	CONSTANT -	
	K010.0 DEC 10.0	FLOATING POINT TEN
C)	COMMUNICATION CELLS -	
	MCSAIP	NON-ZERO IF SHORT ARC IS IN PROCESS, ZERO OTHERWISE, i_{saip}
	MCSASO	NON-ZERO IF A SHORT ARC SOLUTION IS AVAILABLE, ZERO OTHERWISE, S_{sa}

OUTPUT

A)	TABLE -		
	TMBDAV		A SEVEN-LOCATION TABLE CONTAINING THE MOST RECENT BERMUDA VECTORS AND THEIR ASSOCIATED TIME IN THE FOLLOWING FORMAT -
	TMBDAV	t	TIME OF VECTORS IN FLOATING POINT GMT SECONDS
	+1	x	$\overline{r}$, POSITION VECTOR IN GEMINI LENGTH UNITS
	+2	y	
	+3	z	
	+4	$\dot{x}$	$\overline{v}$, VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
	+5	$\dot{y}$	
	+6	$\dot{z}$	
B)	COMMUNICATION CELLS -		
	MCMBDA		BERMUDA SOLUTION AVAIL-

	ABLE INDICATOR, S_{BDA} HAVING ONE OF THE FOLLOWING VALUES -
	0 NO SOLUTION AVAILABLE
	1 SHORT ARC SOLUTION
	2 FPS-16 SMOOTHED SOLUTION
	3 VERLORT SMOOTHED SOLUTION
MCSUPF(OR MCSUPV)	RESET TO ZERO TO UNSUPPRESS FPS-16 (VERLORT) DATA TABLE MOVE, i_{stmf} (OR i_{stmv})

METHOD

WHEN CONTROL IS GIVEN TO MFBDAS, IT DETERMINES FIRST, WHICH OF THE TWO PROCESSORS (CASARC OR CSOGLS) WAS USED IN THIS PASS. ONCE THIS HAS BEEN ESTABLISHED, MFBDAS CHECKS TO SEE IF THE PROCESSOR HAS PRODUCED NEW OUTPUT VECTORS. IF NO NEW SOLUTION IS AVAILABLE, MCMBDA IS SET TO ZERO TO INDICATE MISSING DATA ON BERMUDA BEFORE MFBDAS RETURNS TO MOPRIO.

IF NEW VECTORS HAVE BEEN PRODUCED, THE CELL MCMBDA IS SET TO SHOW THE SOURCE OF THE VECTORS AND THEY ARE THEN MOVED TO THE BERMUDA VECTOR TABLE, TMBDAV.

BEFORE RETURNING CONTROL TO MOPRIO, MFBDAS TURNS OFF THE SUPPRESSION OF THE EDIT ROUTINE AND TURNS OFF ITS OWN A BIT.

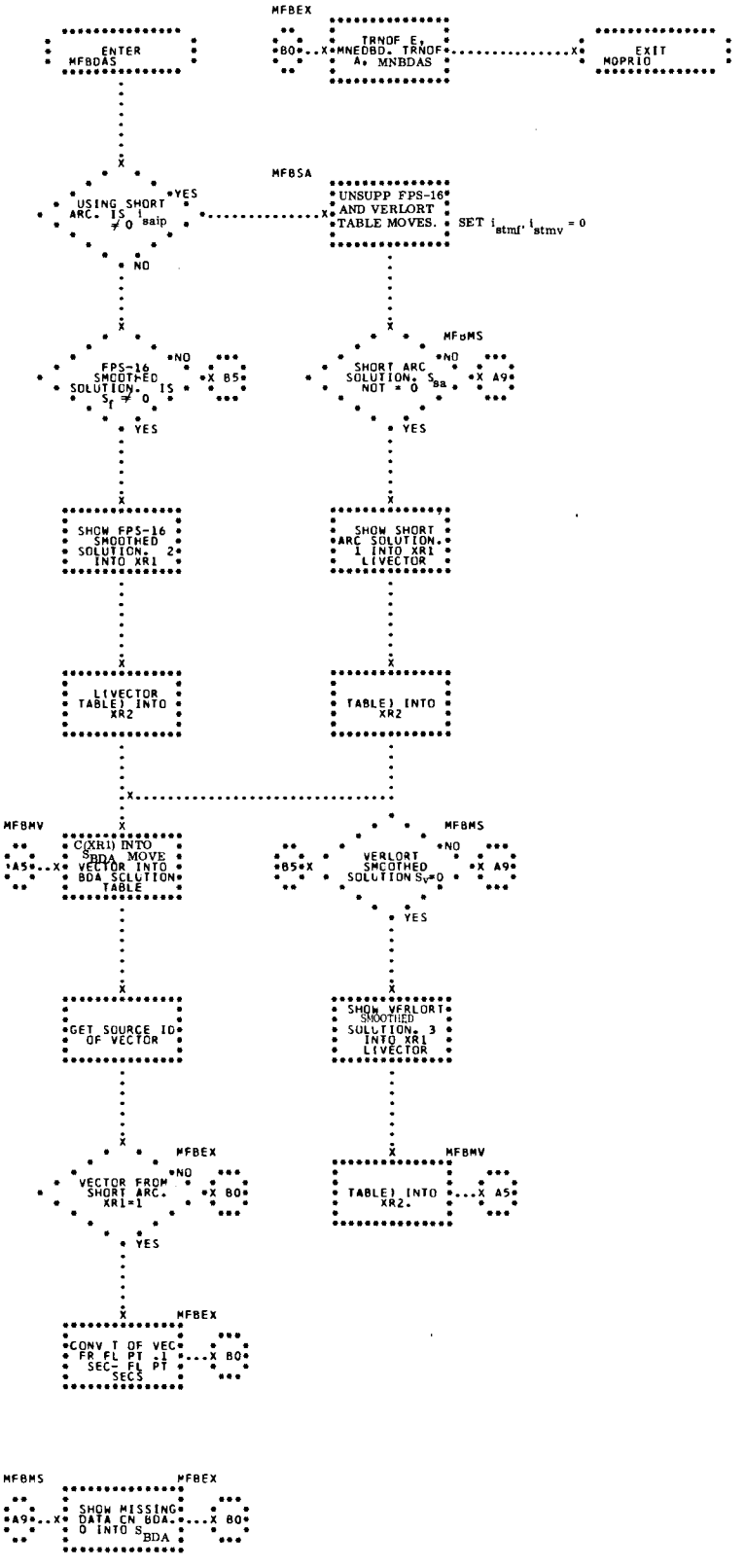
USAGE

MFBDAS IS ENTERED FROM EITHER CASARC, CSOGLS, OR

MPBDAS. IT EXITS TO MOPRIO.

A) STORAGE REQUIRED - 33 LOCATIONS

B) TIMING (7094) - 042 MILLISECONDS, MINIMUM
132 MILLISECONDS, MAXIMUM


ENTER MFBDAS
MFBEX
B0
TRNOF E, A, MNEDBD, TRNOF A, MNBDAS
EXIT MOPRIO
MFBSA
USING SHORT ARC. IS $i_{saip} \neq 0$
YES
NO
UNSUPP FPS-16 AND VERLORT TABLE MOVES. SET $i_{stmf}, i_{stmv} = 0$
FPS-16 SMOOTHED SOLUTION. IS $S_f \neq 0$
NO
B5
YES
MFBMS
SHORT ARC SOLUTION, S_{sa} NOT = 0
NO
A9
YES
SHOW FPS-16 SMOOTHED SOLUTION, 2 INTO XR1
SHOW SHORT ARC SOLUTION, 1 INTO XR1 LIVECTOR
LIVECTOR TABLE) INTO XR2
TABLE) INTO XR2
MFBMV
C(XR1) INTO S_{BDA} MOVE VECTOR INTO BDA SOLUTION TABLE
A5
VERLORT SMOOTHED SOLUTION $S_v \neq 0$
GET SOURCE ID OF VECTOR
SHOW VERLORT SMOOTHED SOLUTION, 3 INTO XR1 LIVECTOR
VECTOR FROM SHORT ARC. XR1=1
TABLE) INTO XR2.
CONV T OF VEC FR FL PT TO SEC- FL PT SECS
SHOW MISSING DATA ON BDA- 0 INTO S_{BDA}

30. MPSTRP - MONITOR PREFIX TO CCSTRP

MPSTRP SETS UP THE INPUT FOR THE CURRENT ENTRY TO THE STRIP CHARTS PROCESSOR, CCSTRP. (SEE FLOWCHART.)

INPUT (NOT APPLICABLE)

OUTPUT

A) COMMUNICATION CELL -

MCSTRO	CONTAINS LOCATION OF THE INPUT FOR THE CURRENT ENTRY TO CCSTRP

METHOD

UPON ENTRY, MPSTRP TURNS ON THE A INDICATOR FOR THE STRIP CHARTS PROCESSOR AND UNQUEUES THE CURRENT INPUT INTO A TEMPORARY LOCATION. THE ADDRESS OF THE INPUT LOCATION IS THEN TRANSFERRED TO THE CELL MCSTRO AND CONTROL IS GIVEN TO CCSTRP.

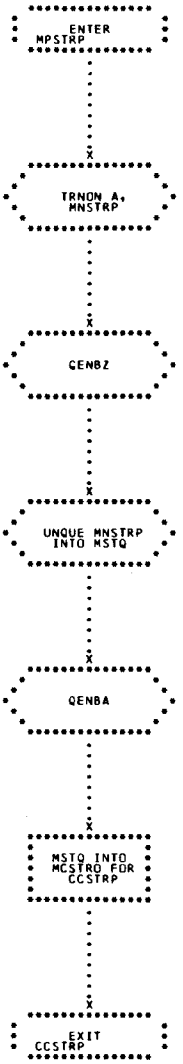
USAGE

MPSTRP IS ENTERED FROM MOPRIO WHEN PRIORITY CONDITIONS PERMIT, IT TRANSFERS CONTROL TO THE STRIP CHARTS PROCESSOR, CCSTRP.

A) STORAGE REQUIRED - 12 LOCATIONS

B) TIMING (7094) - .103 MILLISECONDS

C) MPSTRP USED THE PARAMETERS A AND MNSTRP, THE MACROS TRNON, QENBZ, QENBA, AND UNQUE ARE ALSO USED.


ENTER MPSTRP
TRNON A, MNSTRP
QENBZ
UNQUE MNSTRP INTO MSTQ
QENBA
MSTQ INTO MCSTRQ FOR CCSTRP
EXIT CCSTRP

31. CCSTRP - LAUNCH STRIP CHARTS PROCESSOR

GIVEN A $t, \bar{r}$, and $\bar{v}$ THIS PROGRAM COMPUTES THE DEVIATIONS OF v/v_r AND γ (FLIGHT PATH ANGLE) FROM THEIR NOMINAL VALUES. THIS PROGRAM IS USED EVERY COMPUTING CYCLE DURING NORMAL LAUNCH TO COMPUTE THESE DEVIATIONS FOR BOTH IP-7090 AND B-GE VECTORS. (SEE FLOWCHART.)

INPUT

A) THE ADDRESS OF AN ELEVEN-LOCATION TABLE IS CONTAINED IN THE ACCUMULATOR (AC) UPON THE ENTRY TO THIS PROGRAM. THE FIRST SEVEN LOCATIONS OF THE TABLE CONTAIN t IN FLOATING POINT SECONDS AND THE COMPONENTS OF $\bar{r}$ AND $\bar{v}$ IN FLOATING POINT GEMINI UNITS.

B) LIBRARY SUBROUTINES U1SQRT, U3VMAG AND U1ASCO ARE USED.

C) CONSTANTS -

K00001	FIXED POINT ONE
K00003	FIXED POINT THREE
KGUFTT	FLOATING POINT GEMINI LENGTH UNIT IN FEET
KFTSCG	FLOATING POINT GEMINI VELOCITY UNIT IN FEET/SECOND

D) COMMUNICATION CELLS -

MCTSEC	CONTAINS THE BEST ESTIMATE OF SSCO IN GMT FLOATING POINT SECONDS.
MCLFTM	CONTAINS GMT OF LIFTOFF IN FLOATING POINT SECONDS.

E) TABLES -

TCSTAG	CONTAINS NOMINAL TIME OF STAGING IN

FLOATING POINT SECONDS.

TPVRQD CONTAINS THE ADDRESS OF CONSTANTS AND DEGREE OF CURVE FIT USED TO COMPUTE v_R

TPVVRB CONTAINS THE ADDRESS OF CONSTANTS AND DEGREE OF CURVE FIT USED TO COMPUTE $(v/v_R)_{nom}$ BEFORE STAGING.

TPYNMB CONTAINS THE ADDRESS OF CONSTANTS AND DEGREE OF CURVE FIT USED TO COMPUTE $(\gamma)_{nom}$ BEFORE STAGING.

TPVVRA CONTAINS THE ADDRESS OF CONSTANTS AND DEGREE OF CURVE FIT TO COMPUTE $(v/v_R)_{nom}$ AFTER STAGING.

TPYNMA CONTAINS THE ADDRESS OF CONSTANTS AND DEGREE OF CURVE FIT USED TO COMPUTE $(\gamma)_{nom}$ AFTER STAGING.

OUTPUT

THE TENTH AND ELEVENTH LOCATIONS OF THE INPUT TABLE CONTAIN $(v/v_R)-(v/v_R)_{nom}$ AND $\gamma - \gamma_{nom}$ UPON EXIT FROM THE PROGRAM. OUTPUT IS IN FLOATING POINT DIMENSIONLESS FORMAT AND FLOATING POINT RADIANS, RESPECTIVELY.

METHOD

FIRST, THE VELOCITY REQUIRED (v_R) IS COMPUTED USING

$$v_R = \frac{k_6 \mu_e^{1/2} |r| + k_7 \mu_e^{1/2}}{\sqrt{|r|}}$$

WHERE

$k_6\mu_e^{1/2}$ and $k_7\mu_e^{1/2}$ ARE CURVE FIT CONSTANTS

$|r|$ IS THE MAGNITUDE OF THE POSITION VECTOR IN FEET

$\sqrt{|r|}$ IS THE SQUARE ROOT OF THE POSITION VECTOR MAGNITUDE IN FEET

THUS, v/v_R IS EASILY COMPUTED, WHERE v IS THE MAGNITUDE OF THE VELOCITY VECTOR IN FEET/SECOND.

THE FLIGHT PATH ANGLE (γ) IS THEN COMPUTED USING

$$\gamma = \arcsin \frac{\bar{r} \cdot \bar{v}}{|r|\,|v|}$$

(where $\bar{r}$, $\bar{v}$, $|r|$, and $|v|$ are all in Gemini units.)

BOTH $(v/v_R)_{nom}$ AND γ_{nom} ARE COMPUTED USING POLYNOMIAL CURVE FITS. THE SPECIFIC CURVE FIT IS CHOSEN IN THE FOLLOWING MANNER -

A) IF THE TIME OF THE INPUT VECTOR IS LESS THAN THE NOMINAL TIME OF STAGING, THE ''BEFORE STAGING'' CURVE FITS ARE USED TO COMPUTE BOTH $(v/v_R)_{nom}$ AND γ_{nom} THE INDEPENDENT VARIABLE IS $t_{INPUT} - t_{LIFTOFF}$.

B) IF THE TIME OF THE INPUT VECTOR IS EQUAL TO OR GREATER THAN THE NOMINAL TIME OF STAGING, THE ''AFTER STAGING'' CURVE FITS ARE USED TO COMPUTE BOTH $(v/v_R)_{nom}$ AND γ_{nom}.

1) IF t_{SSCO} HAS NOT BEEN SET, THE INDEPENDENT VARIABLE USED IN THE CURVE FITS IS $t_{INPUT} - t_{LIFTOFF}$.

2) IF t_{SSCO} HAS BEEN SET, THE INDEPENDENT VARIABLE IS $t_{SSCO} - t_{LIFTOFF}$.

PLOTS OF THE NOMINAL VALUES OF v/v_R AND γ VS TIME SINCE LIFTOFF ARE SHOWN IN CCSTRP FIGURES 1 THROUGH 4 FOR BOTH GT-2 AND GT-3.

USAGE

A) CCSTRP IS ENTERED THROUGH ITS PREFIX MPSTRP AND TRANSFER CONTROL TO ITS SUFFIX MFSTRP.

B) STORAGE REQUIRED -
104 LOCATIONS EXCLUDING SPACE REQUIRED BY EXTERNAL CONSTANTS, TABLES AND SUBROUTINES.

C) TIME REQUIRED (MAXIMUM 7094) -
1.414 MILLISECONDS, EXCLUDING TIME REQUIRED BY U1ASCO.

D) ERROR CONDITIONS -
CCSTRP HAS NO ERROR RETURN.

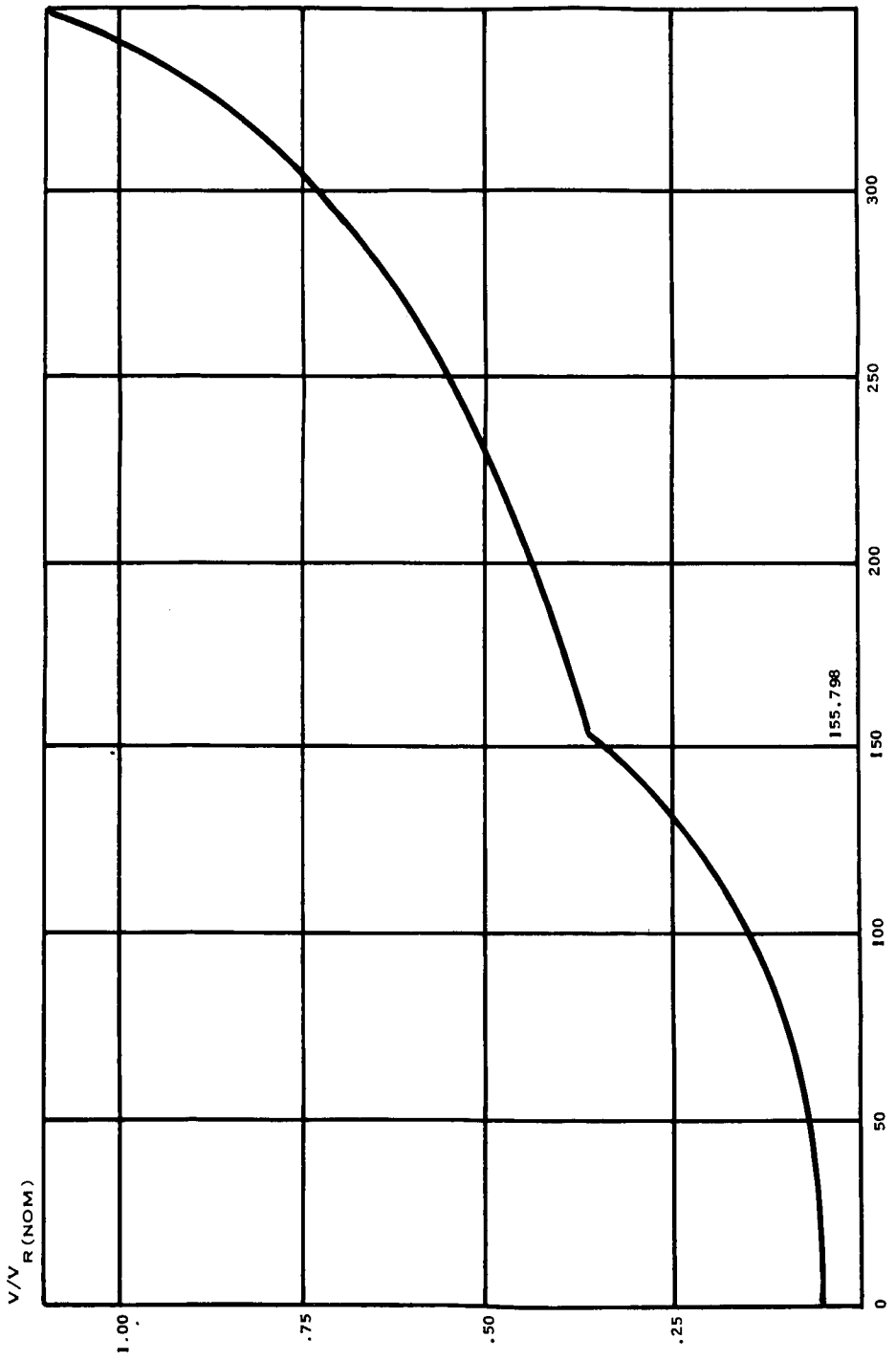


Figure 1. $v/v_{R(nom)}$ Versus Time Since Liftoff for GT-2

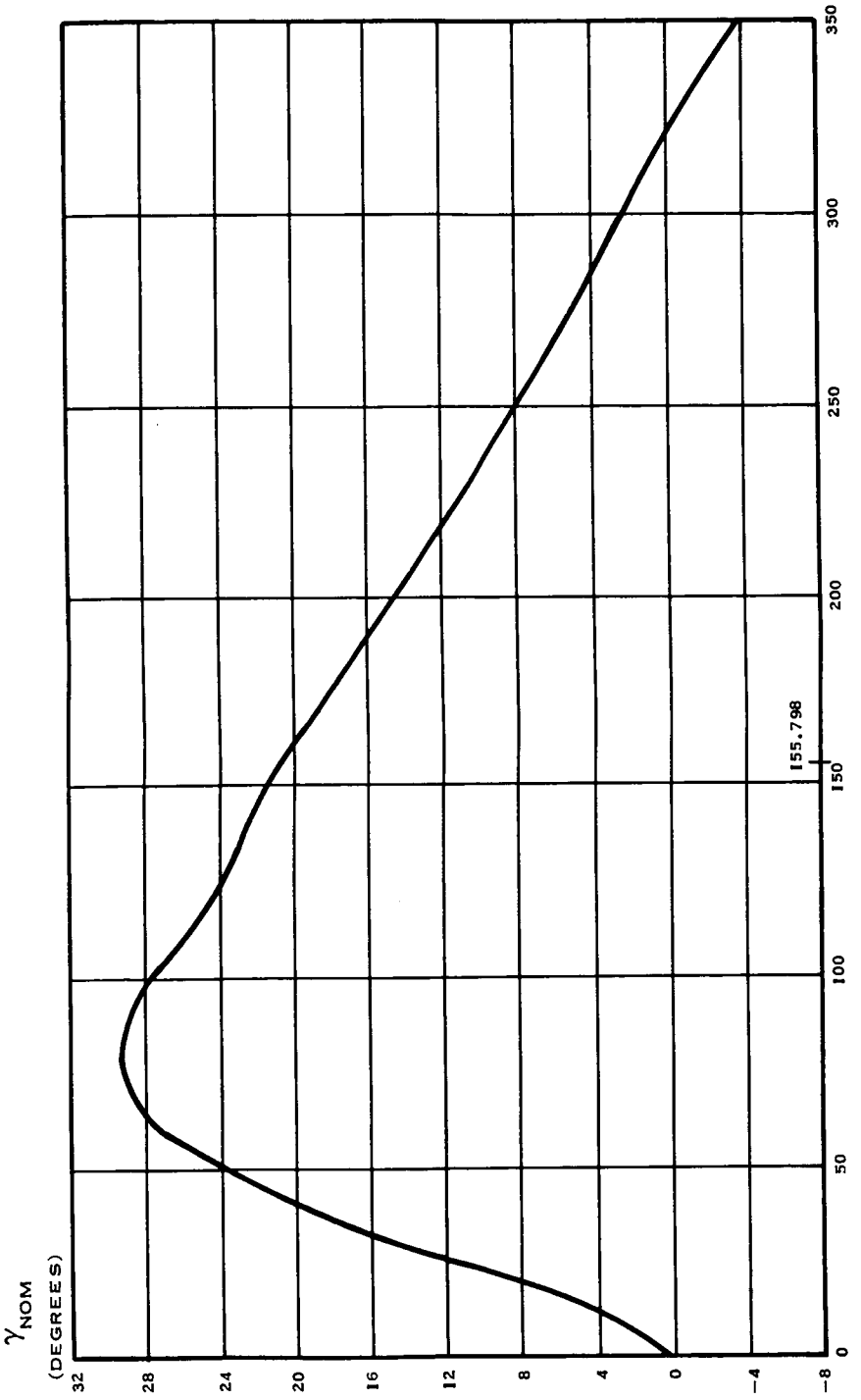


Figure 2. γ_{nom} Versus Time Since Liftoff for GT-2

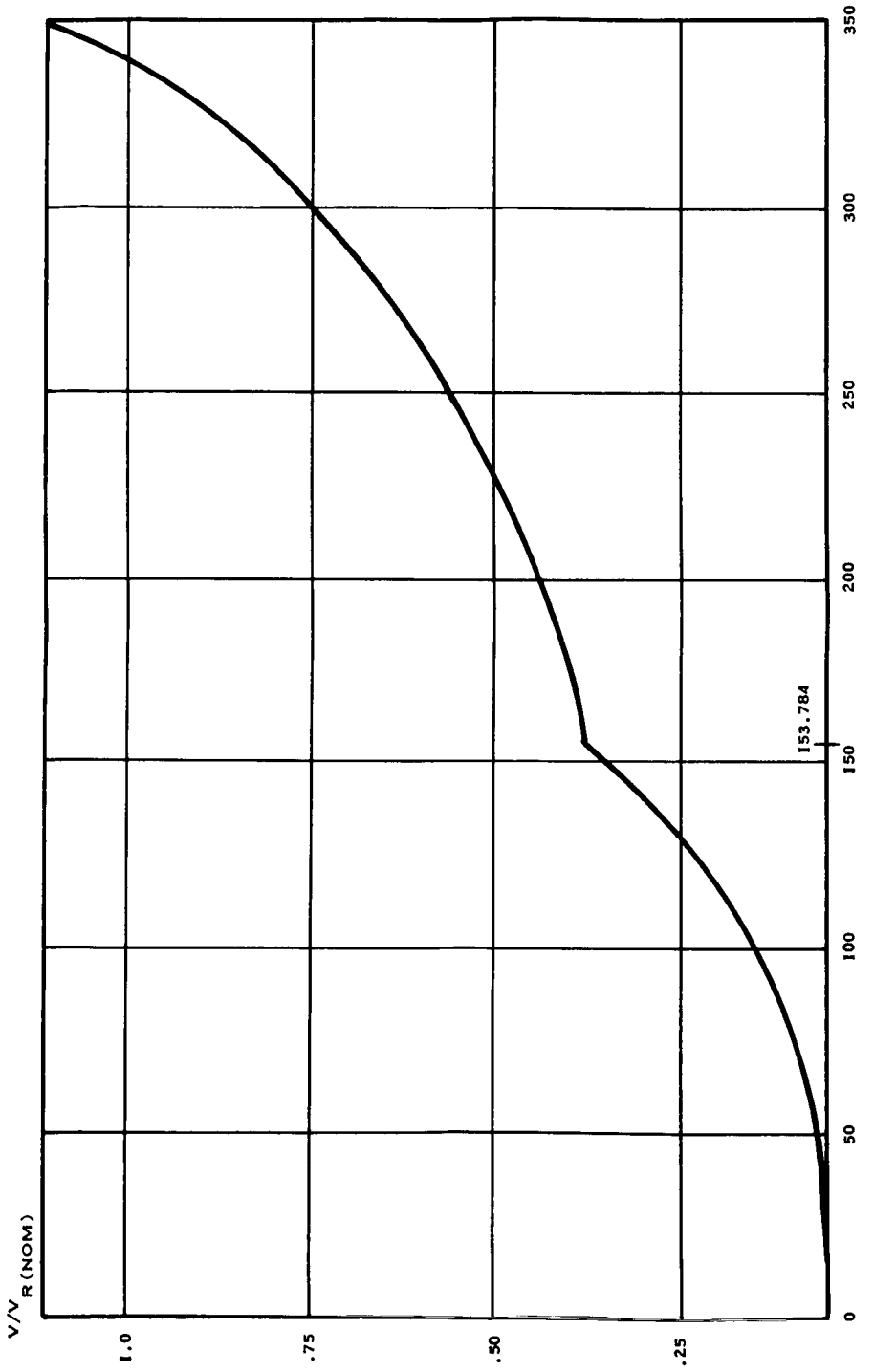


Figure 3. $v/v_{R(nom)}$ Versus Time Since Liftoff for GT-3

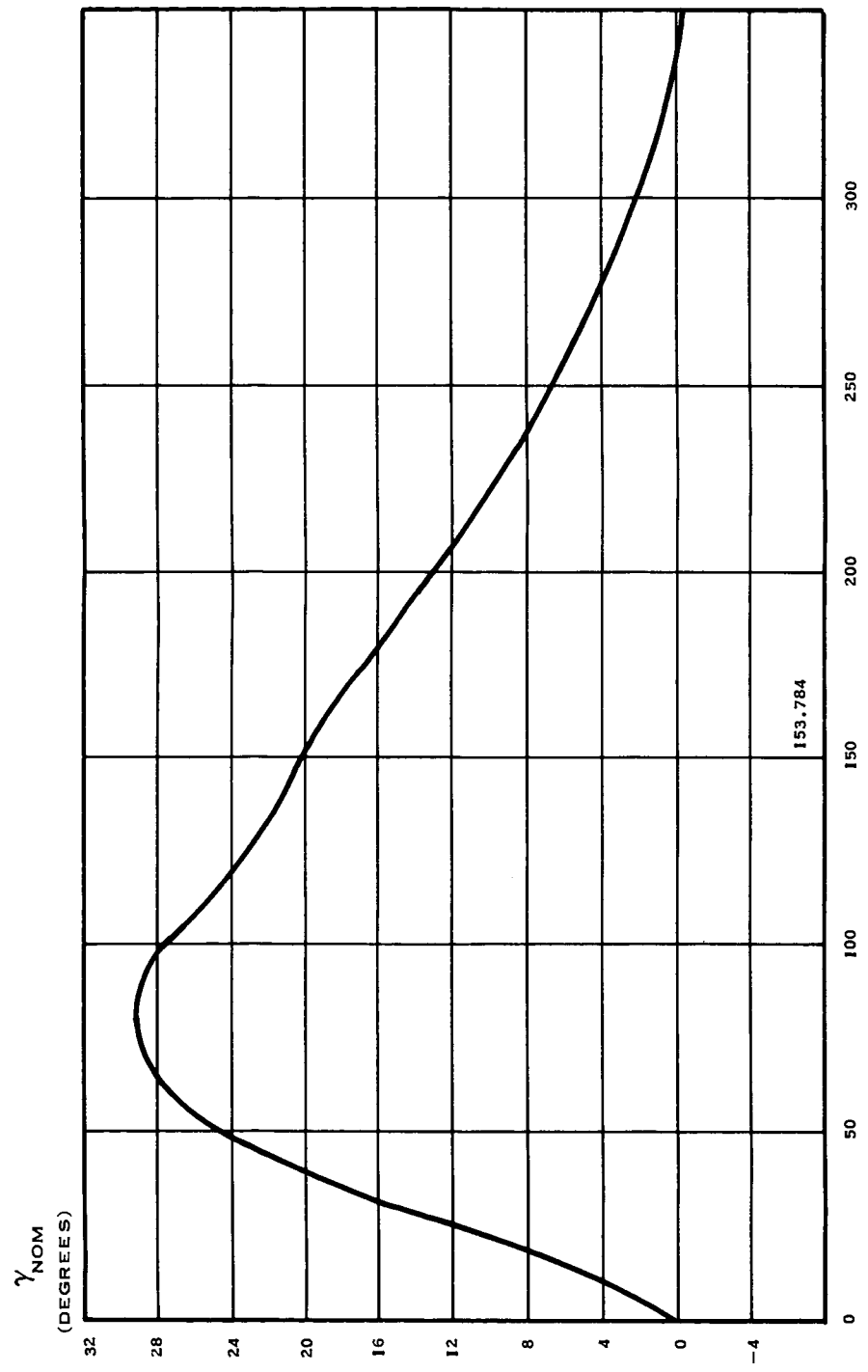
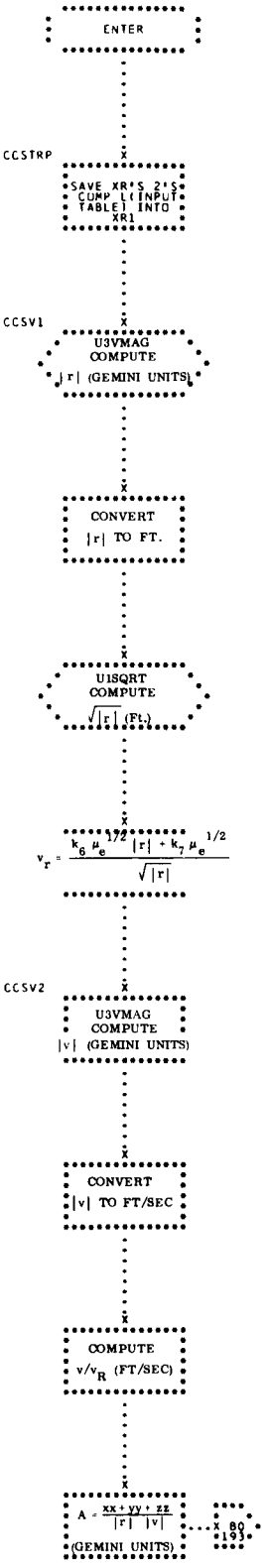
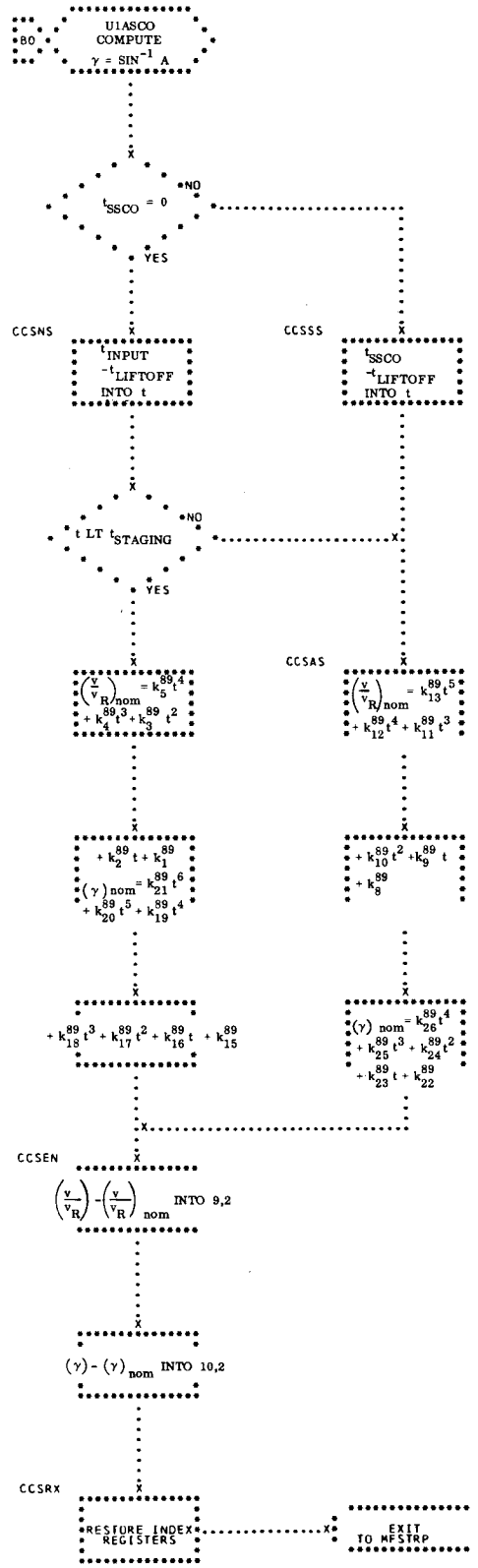


Figure 4. γ_{nom} Versus Time Since Liftoff for GT-3


ENTER
CCSTRP
SAVE XR'S 2'S COMP L(INPUT TABLE) INTO XR1
CCSV1
U3VMAG COMPUTE |r| (GEMINI UNITS)
CONVERT |r| TO FT.
U1SQRT COMPUTE √|r| (Ft.)
$v_r = \frac{k_6 \mu_e^{1/2} |r| + k_7 \mu_e^{1/2}}{\sqrt{|r|}}$
CCSV2
U3VMAG COMPUTE |v| (GEMINI UNITS)
CONVERT |v| TO FT/SEC
COMPUTE v/v_R (FT/SEC)
$A = \frac{xx + yy + zz}{|r| \ |v|}$
(GEMINI UNITS)
X 80
193


BO
U1ASCO
COMPUTE
$\gamma = \sin^{-1} A$
$t_{SSCO} = 0$
NO
YES
CCSNS
$t_{INPUT} - t_{LIFTOFF}$ INTO t
CCSSS
$t_{SSCO} - t_{LIFTOFF}$ INTO t
t LT $t_{STAGING}$
NO
YES
CCSAS
$\left(\frac{v}{v_R}\right)_{nom} = k_5^{89} t^4 + k_4^{89} t^3 + k_3^{89} t^2$
$+ k_2^{89} t + k_1^{89}$
$(\gamma)_{nom} = k_{21}^{89} t^6 + k_{20}^{89} t^5 + k_{19}^{89} t^4$
$+ k_{18}^{89} t^3 + k_{17}^{89} t^2 + k_{16}^{89} t + k_{15}^{89}$
$\left(\frac{v}{v_R}\right)_{nom} = k_{13}^{89} t^5 + k_{12}^{89} t^4 + k_{11}^{89} t^3$
$+ k_{10}^{89} t^2 + k_9^{89} t$
$+ k_8^{89}$
$(\gamma)_{nom} = k_{26}^{89} t^4 + k_{25}^{89} t^3 + k_{24}^{89} t^2$
$+ k_{23}^{89} t + k_{22}^{89}$
CCSEN
$\left(\frac{v}{v_R}\right) - \left(\frac{v}{v_R}\right)_{nom}$ INTO 9,2
$(\gamma) - (\gamma)_{nom}$ INTO 10,2
CCSRx
RESTORE INDEX REGISTERS
EXIT TO MFSTRP

32. MFSTRP - MONITOR SUFFIX TO CCSTRP

MFSTRP EXAMINES AND CONTROLS THE INPUT TO THE COLLECTION PROCESS AND TO THE MAIN LAUNCH CALCULATIONS PROGRAM, CCMAIN. (SEE FLOWCHART)

INPUT

A) TABLES -

TMRV3G		AN ELEVEN-LOCATION TABLE CONTAINING THE B-GE INPUT VECTOR ARRANGED AS FOLLOWS -
TMRV3G	t	VECTOR TIME TAG IN FLOATING POINT GMT SECONDS
+1	x	POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+2	y	
+3	z	
+4		
+5	$\dot{x}$	VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
	$\dot{y}$	
+6	$\dot{z}$	
+7		DATA SOURCE
+8		DATA QUALITY
+9		DEVIATION OF VELOCITY RATIO FROM NOMINAL
+10		DEVIATION OF FLIGHT PATH ANGLE FROM NOMINAL

TMRV31	AN ELEVEN-LOCATION TABLE CONTAINING THE IP-7094 INPUT VECTOR AND ARRANGED IN THE SAME MANNER AS TMRV3G.

B) COMMUNICATION CELLS -

MCGMLM	CONTAINS MAXIMUM ALLOWABLE FLIGHT PATH ANGLE DEVIATION FROM NOMINAL.
MCVRLM	CONTAINS MAXIMUM ALLOWABLE VELOCITY RATION DEVIATION FROM NOMINAL.

OUTPUT

A) TABLES -

TMSCGB	A SEVEN-LOCATION TABLE CONTAINING THE LATEST B-GE VECTOR PROCESSED AND ITS ASSOCIATED TIME TAG.
TMSCIP	A SEVEN-LOCATION TABLE CONTAINING THE LATEST IP 3600 VECTOR PROCESSED AND ITS ASSOCIATED TIME TAG.

B) COMMUNICATION CELLS -

MCSCGB	NON-ZERO IF B-GE VECTOR HAS BEEN PROCESSED.
MCSCIP	NON-ZERO IF IP 3600 VECTOR HAS BEEN PROCESSED.
MCSOUR	CONTAINS DATA SOURCE IDENTIFICATION FOR SOURCE TO BE USED IN COLLECTION PROCESS.

METHOD

MFSTRP DETERMINES WHETHER EITHER THE COMPUTED DEVIATION OF THE VELOCITY RATIO OR OF THE FLIGHT PATH ANGLE FALLS WITHOUT ITS LIMIT. IF IT DOES, MFSTRP SETS AN INDICATOR NON-ZERO TO FORCE EXCLUSION OF THIS VECTOR FROM THE COLLECTION TABLE.

AFTER DETERMINING THE SOURCE OF THE CURRENT VECTOR, MFSTRP MOVES THE VECTOR TO THE INTERMEDIATE BUFFER USED BY THE HALF-SECOND TRAP PROCESSOR, MTHSFC. (THE INTERMEDIATE BUFFER IS TMSCGB, IF THE VECTOR IS FROM B-GE, OR TMSCIP, IF THE VECTOR IS FROM IP 3600.)

IF THE VECTOR IS TO BE ADDED TO THE COLLECTION TABLE, IT IS MOVED TO THE APPROPRIATE COLLECTION TABLE BY THE SUBROUTINE COCOLLECT. MFSTRP THEN RETURNS CONTROL TO MOPRIO.

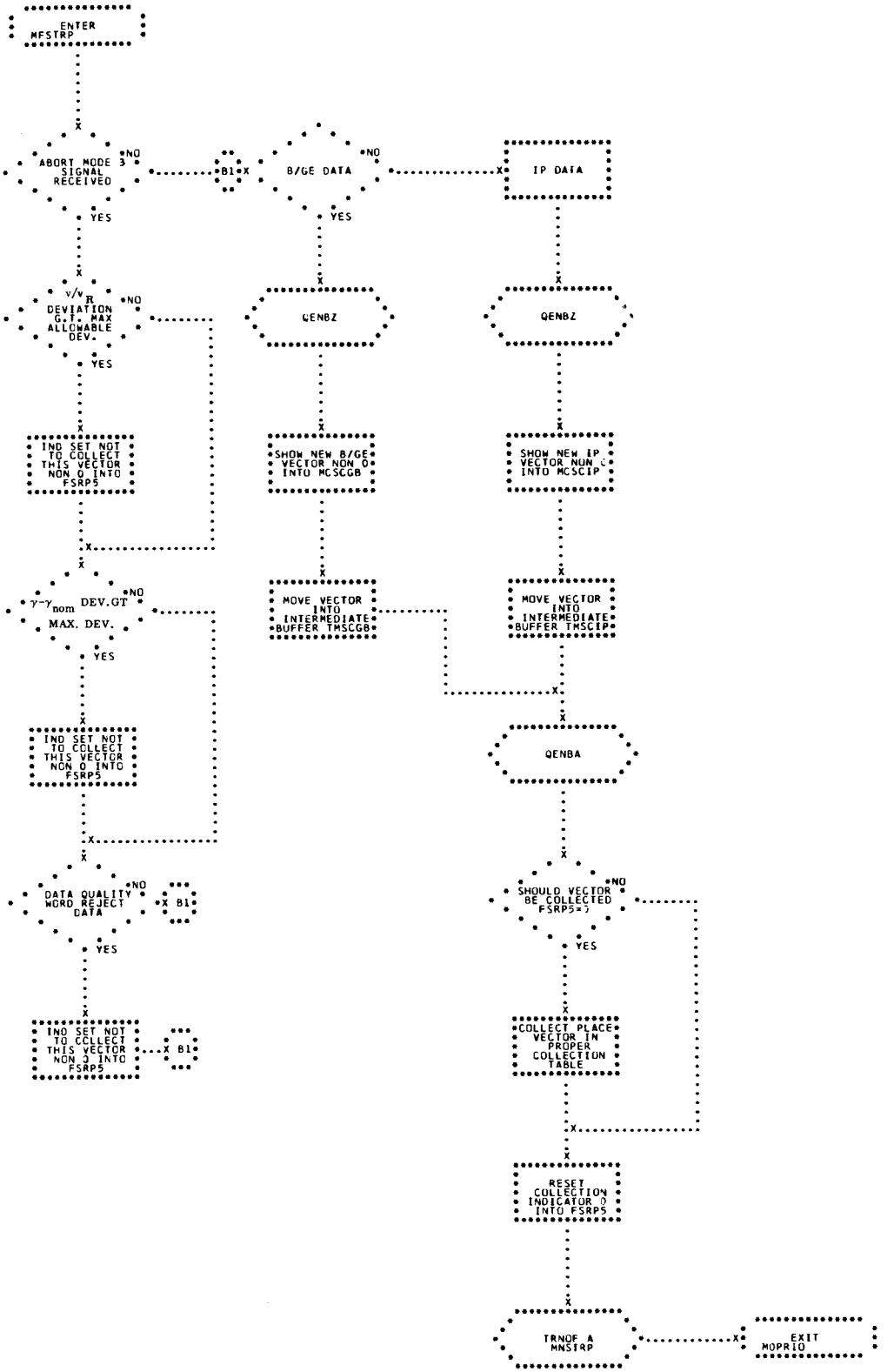
USAGE

MFSTRP IS ENTERED FROM THE STRIP CHARTS PROCESSOR, CCSTRP, IT EXITS TO MOPRIO.

A) STORAGE REQUIRED - 41 LOCATIONS

B) TIMING (7094) - .144 MILLISECONDS, MINIMUM
.227 MILLISECONDS, MAXIMUM

C) MFSTRP USES THE PARAMETERS A AND MNSTRP, THE MACROS QENBZ, QENBA, TRNUF ARE ALSO USED.


ENTER MFSTRP
ABORT MODE 3 SIGNAL RECEIVED
NO
YES
B1
B/GE DATA
NO
YES
IP DATA
V/V_R DEVIATION G.T. MAX ALLOWABLE DEV.
NO
YES
QENBZ
QENBZ
IND SET NOT TO COLLECT THIS VECTOR NON 0 INTO FSRP5
SHOW NEW B/GE VECTOR NON 0 INTO MCSCGB
SHOW NEW IP VECTOR NON 0 INTO MCSCIP
$\gamma-\gamma_{nom}$ DEV.GT MAX. DEV.
NO
YES
MOVE VECTOR INTO INTERMEDIATE BUFFER TMSCGB
MOVE VECTOR INTO INTERMEDIATE BUFFER TMSCIP
IND SET NOT TO COLLECT THIS VECTOR NON 0 INTO FSRP5
QENBA
DATA QUALITY WORD REJECT DATA
NO
YES
B1
SHOULD VECTOR BE COLLECTED FSRP5=0
NO
YES
IND SET NOT TO COLLECT THIS VECTOR NON 2 INTO FSRP5
B1
COLLECT PLACE VECTOR IN PROPER COLLECTION TABLE
RESET COLLECTION INDICATOR 0 INTO FSRP5
TRNOF A MNSTRP
EXIT MOPRIO

33. MPCCMN - MONITOR PREFIX TO CCMAIN.

INPUT

INPUT TO MPCCMN CONSISTS OF TELEMETRY AND OTHER SIGNALS NECESSARY FOR THE CHOICE OF THE PROPER INPUT TO CCMAIN AND PROPER CONTROL OF LAUNCH COMPUTATIONS AND DATA FLOW. (SEE FLOWCHART.)

OUTPUT

UPON EXIT, MPCCMN HAS INDICATED TO CCMAIN OR MFCCMN THE APPROPRIATE VECTOR AND TELEMETRY SIGNALS.

USAGE

MPCCMN IS ENTERED FROM MOPRIO AND EXITS TO CCMAIN OR MFCCMN.

A) STORAGE REQUIRED -
   1380 LOCATIONS

B) MACROS -
   QENBZ
   QENBA
   TRNON
   TRNOF
   QUEUE

C) PARAMETERS -

A	B
C	MNCCMN
MNEDBD	MNMPRK
MNSENS	MNNSRS
MNMESS	

D) SUBROUTINE -
   SUMATE

CUPDAT

E) CONSTANTS -

K00000 K00001 K00002
K00003 K00010 K00015
K000.5 K002.0 K004.5
K005.0 K010.0 K010.0
K0020. K8.3M3 KCH232
KCH233

F) TABLES -

TMSCGB TMLANA TMSCIP TMSIGV TM8.3M TCCOUT
TMCYCL TMCYNO TMOUT1 TMOUT3 TMOUT4 TMOUT5
TMOUT6 TMOUT7 TMLHGB TMLHBD TMLIPB TMLIPC
TMBDAV TCGMT1 TCNPWC TMHEDV TMHEDF TCVSWT
TCFSWT TCBDVE TCBDF6 TABEDT TMGLVC TMILVB
TMILVC TMBLVC TC.ONT TMTHTI TPRAM6 TMSUM1
TMSUM4 TMSUM5 TMSUM6 TMSUM7 TMSLMN TMSLMF

G) COMMUNICATION CELLS -

MCREST MCLFTM MCCFHP MCFVEC MCPSEL MCABMC
MCPABM MCLIFT MCELFT MCSCUT MCPSSC MCAAAB
MCPAAA MCSECO MCPSEC MCSCTI MCTBEG MCENDT
MCTEND MCSPSE MCPSPS MCORPS MCPORP MCABPS
MCPABS MC4RRF MCP4RR MCNRRF MCPNRE MC1RRF
MCP1RR MC2RRF MCP2RR MCP3RR MCERRS MCPERS
MCLTTL MCSBDA MCSSOR MCSINP MCHFSC MCCURT
MCUABM MCSGNG MCBSUM MCTSEC MCTBAR MCTSUM
MCCCMD MCLAST MCBGNG MCLIMP MCDTCL MCACTV
MCFRST MCPROG MCMBDA MCPBDA MCMISD MCMIDA
MCCURV MCLIDT MCLLIN MCLLOT MCMINT MCFSIN
MCMAXT MCSUMS MCIPBS MCXXXX MCBSUM MCDISP
MCEABM MCESSC MCEAAA MCESEC MCE4RR MCE3RR
MCE2RR MCE1RR MCEERS MCERTS MCESPS MCEORP
MCEABS

METHOD -

A) IF LIFTOFF HAS NOT OCCURED OR THERE IS NO DATA, CONTROL IS RETURNED TO MOPRIO. IF AVERAGE VECTOR COMPUTATIONS ARE BEING DONE (THE BOOSTER GO/NO-GO RECOMMENDATIONS HAVE BEEN COMPLETED.) CONTROL IS PASSED TO PARAGRAPH (B). OTHERWISE, STRIP CHARTS DEVIATION QUANTITIES ARE MOVED INTO THE PROPER OUTPUT LOCATIONS FOR DISPLAY.

B) THE INPUT VECTOR IS SAVED IN TEMPORARY STORAGE AND INPUT TELEMETRY SIGNALS ARE SAVED IN INTERMEDIATE STORAGE. CURRENT TIME IS COMPUTED AND STORED. SELECTED TELEMETRY EVENTS ARE CHECKED AND A MESSAGE IS PRINTED FOR EACH EVENT WHICH HAS OCCURED.

IF ABORT MODE THREE (AM3) HAS NOT OCCURED, CONTROL IS PASSED TO PARAGRAPH (C). IF AM3 HAS OCCURRED BUT NEITHER SUSTAINER ENGINE CUTOFF (SECO), SECOND STAGE CUTOFF (SSCO), NOR ASTRONAUT ACTUATED ABORT (AAA) HAS OCCURED, CONTROL IS PASSED TO PARAGRAPH (C). IF AM3 AND EITHER SECO, SSCO, OR AAA HAS OCCURRED AND THE TIME OF THE END OF POWERED FLIGHT HAS BEEN SET, CONTROL IS PASSED TO PARAGRAPH (E). IF AM3 AND EITHER SECO, SSCO, OR AAA HAS OCCURRED BUT THE TIME OF THE END OF POWERED FLIGHT HAS NOT BEEN SET, THE TIME OF THE END OF POWERED FLIGHT IS ESTABLISHED AND CONTROL IS PASSED TO PARAGRAPH (E).

C) IF THE TIME OF THE END OF POWERED FLIGHT HAS NOT BEEN ESTABLISHED, CONTROL IS PASSED TO PARAGRAPH (D). IF THE TIME OF THE END OF POWERED FLIGHT HAS BEEN ESTABLISHED, ALL INDICATORS WHICH REFLECT POWER CUTOFF ARE RESET TO POWERED FLIGHT STATUS. THE BERMUDA HIGH-SPEED LINES ARE DE-ACTIVATED AND CONTROL

IS PASSED TO PARAGRAPH (D).

D) THE PROPER VECTOR IS STORED AS INPUT, TO CCMAIN WITH ALL INPUT AND OUTPUT PARAMETERS PROPERLY SET. EFFECTIVE TELEMETRY SIGNALS ARE SET FOR USE BY CCMAIN AND MFCCMN. IF THERE IS NO NEW VECTOR FOR CCMAIN, PROPER INDICATORS ARE SET TO USE THE LAST INPUT VECTOR AS INPUT TO CCMAIN. CONTROL IS PASSED TO CCMAIN.

E) THE STATUS OF THE SPACECRAFT THRUSTING INDICATOR IS RECORDED FOR USE IN UPDATING AVERAGED VECTORS. AT THE PROPER TIME THE BERMUDA HIGH-SPEED LINES ARE ACTIVATED.

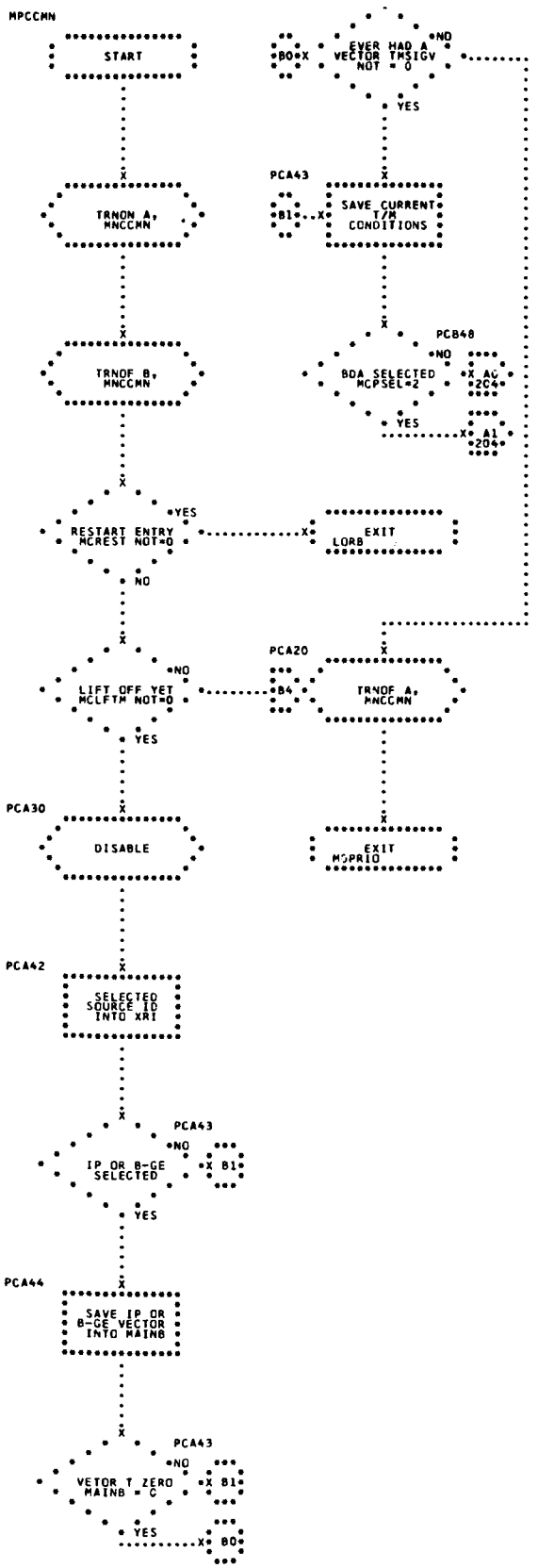
IF THE BOOSTER GO/NO-GO RECOMMENDATIONS HAVE BEEN COMPLETED, CONTROL IS PASSED TO PARAGRAPH (H). IF THE BOOSTER GO/NO-GO RECOMMENDATIONS HAVE NOT BEEN COMPLETED, BUT THE RECOMMENDATIONS SHOULD BE COMPUTED, CONTROL IS PASSED TO PARAGRAPH (F). OTHERWISE CONTROL IS PASSED TO PARAGRAPH (D).

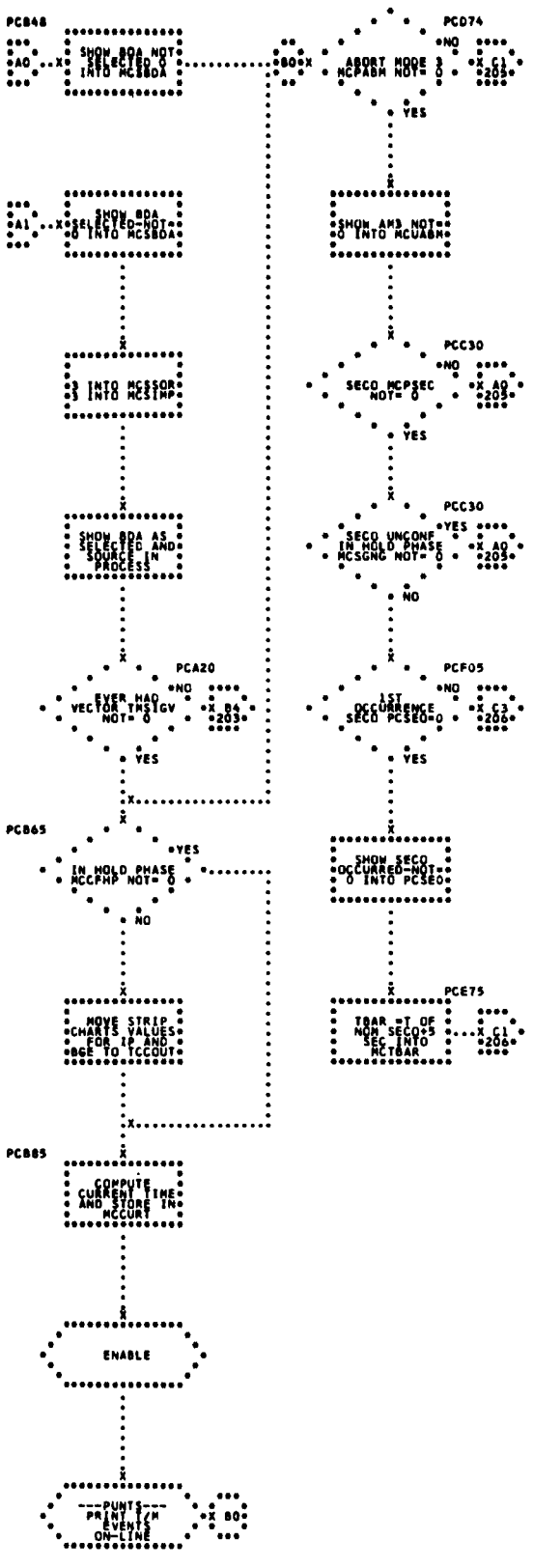
F) IF THERE ARE VECTORS FROM GE/B IN THE COLLECTION PERIOD, THE GE/B VECTORS ARE AVERAGED FOR USE IN THE GO/NO-GO COMPUTATIONS. OTHERWISE THE LAST VECTOR FROM GE/B IS USED AS INPUT. IF THERE ARE VECTORS FROM IP-3600 IN THE COLLECTION PERIOD, THE IP-3600 VECTORS FROM THE MISTRAM, AZUSA, FPQ-6, OR FPS-16 RADARS ARE AVERAGED TO OBTAIN A VECTOR FOR USE IN THE GO/NO-GO COMPUTATIONS. IF THERE ARE NO VECTORS FROM IP-3600 IN THE COLLECTION PERIOD, THE LAST VECTOR FROM IP-3600 IS USED AS INPUT. THE ORDER OF COMPUTING THE GO/NO-GO RECOMMENDATIONS IS ESTABLISHED. IF BERMUDA IS NOT THE SELECTED SOURCE, CONTROL IS PASSED TO PARAGRAPH (G). IF BERMUDA IS THE SELECTED SOURCE, THE LATEST VECTOR OBTAINED FROM

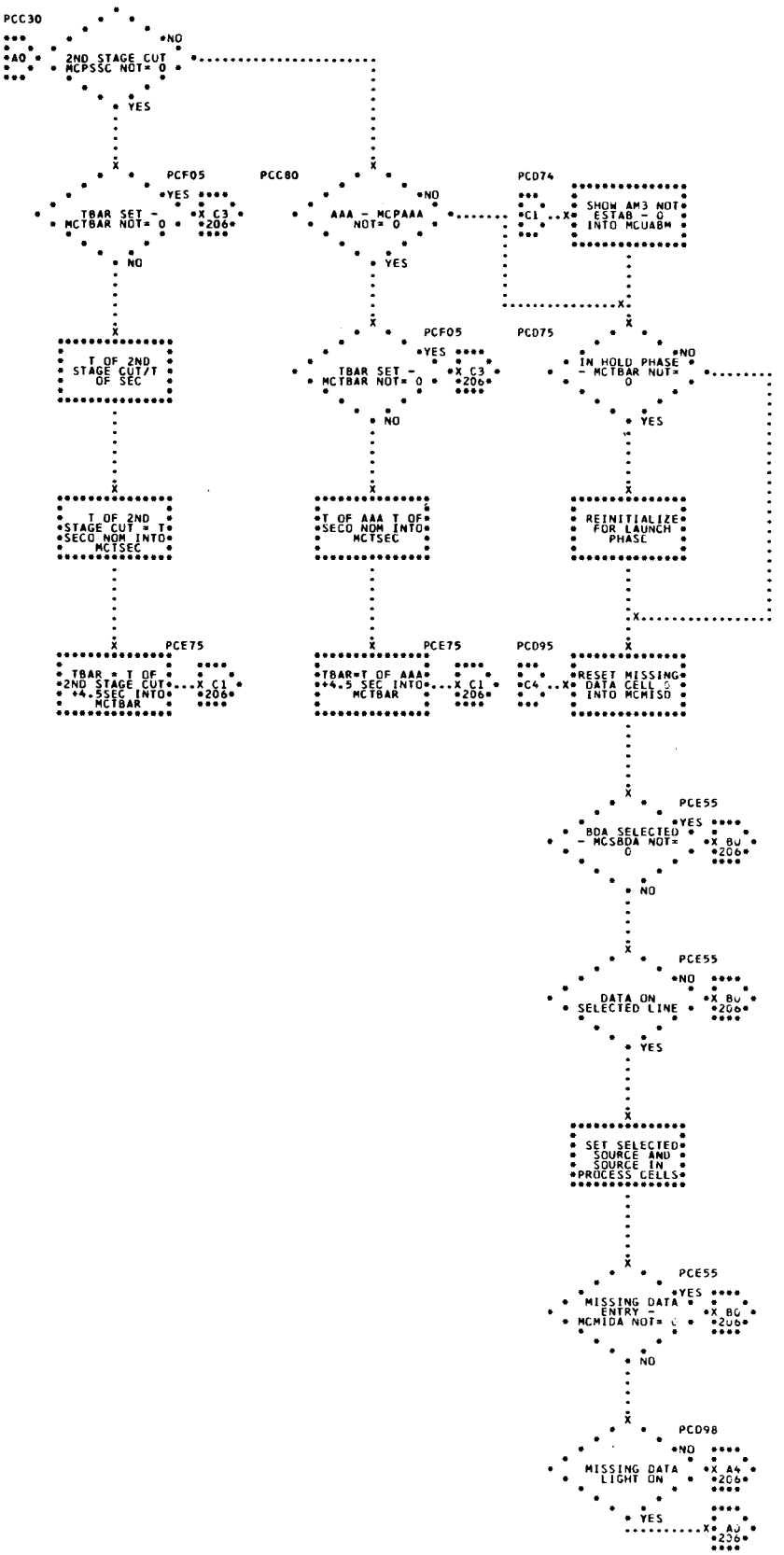
THE BERMUDA DATA IS USED AS INPUT FOR THE BERMUDA GO/NO-GO COMPUTATIONS.

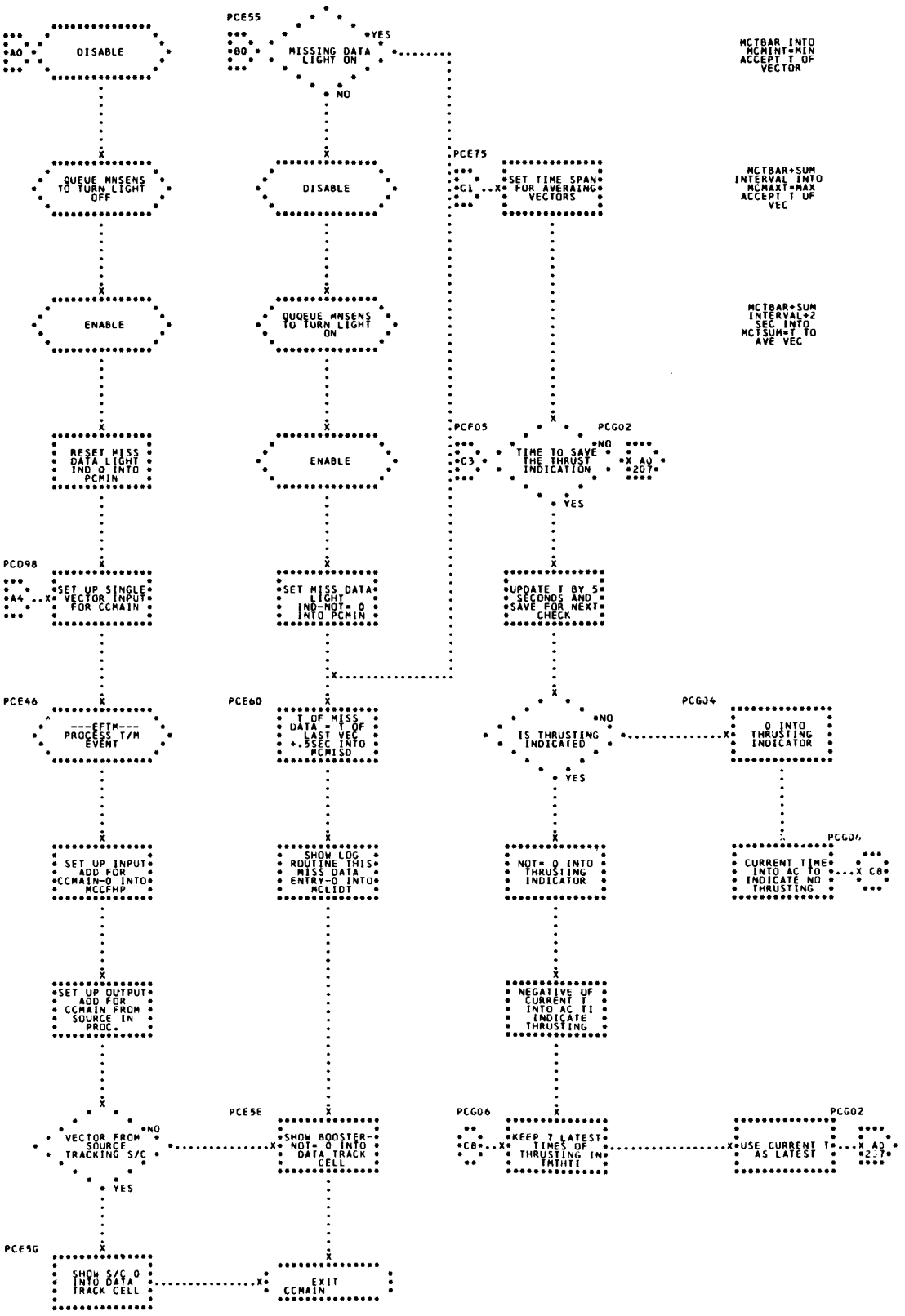
G) ALL INPUT AND OUTPUT PARAMETERS FOR CCMAIN AND MFCCMN ARE PROPERLY SET. MISSING DATA INDICATORS ARE PROPERLY SET OR RESET. IF THE INPUT DATA IS ZERO, CONTROL IS PASSED TO MFCCMN. OTHERWISE CONTROL IS PASSED TO CCMAIN.

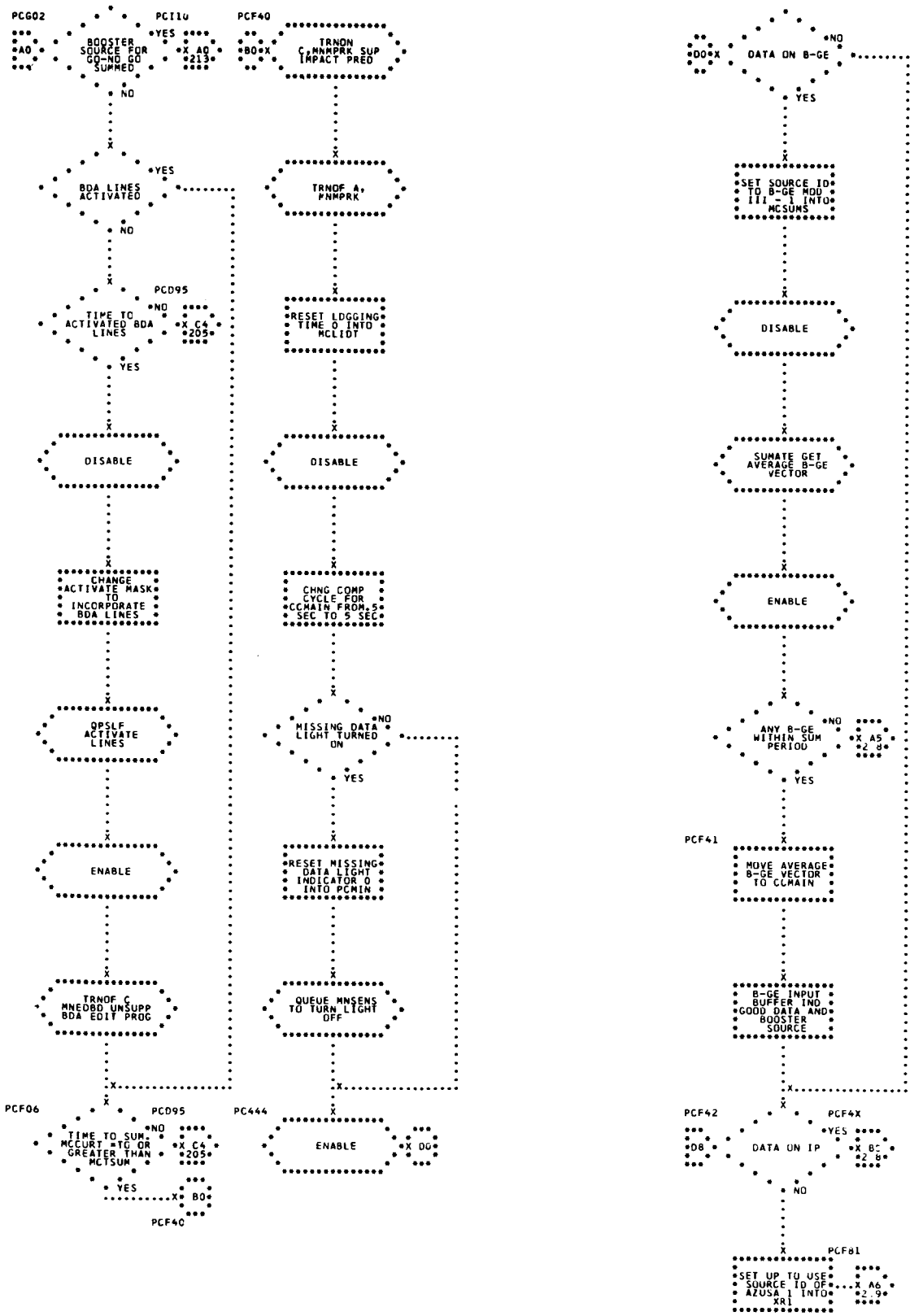
H) IF A VECTOR HAS EVER BEEN OBTAINED FROM BERMUDA DATA, THE LATEST BERMUDA VECTOR IS INTEGRATED UP TO CURRENT TIME. IF NO BERMUDA VECTOR HAS EVER BEEN RECEIVED, THE OUTPUT BLOCK FOR BERMUDA IS SET TO ZERO. IF THERE ARE SPACECRAFT VECTORS FROM IP-3600 IN THE COLLECTION PERIOD, THE IP-3600 VECTORS FROM THE FPQ-6 OR FPS-16 RADARS ARE AVERAGED AND INTEGRATED TO CURRENT TIME. IF THERE ARE NO SPACECRAFT VECTORS IN THE COLLECTION PERIOD, THE LAST VECTOR RECEIVED IS INTEGRATED TO CURRENT TIME. IF NO CALCULATIONS SHOULD BE PERFORMED THE PROPER OUTPUT PARAMETERS ARE SET, THE MISSING DATA INDICATORS ARE SET OR RESET, AND CONTROL IS PASSED TO MFCCMN. IF CALCULATIONS SHOULD OCCUR CONTROL IS PASSED TO PARAGRAPH (G).

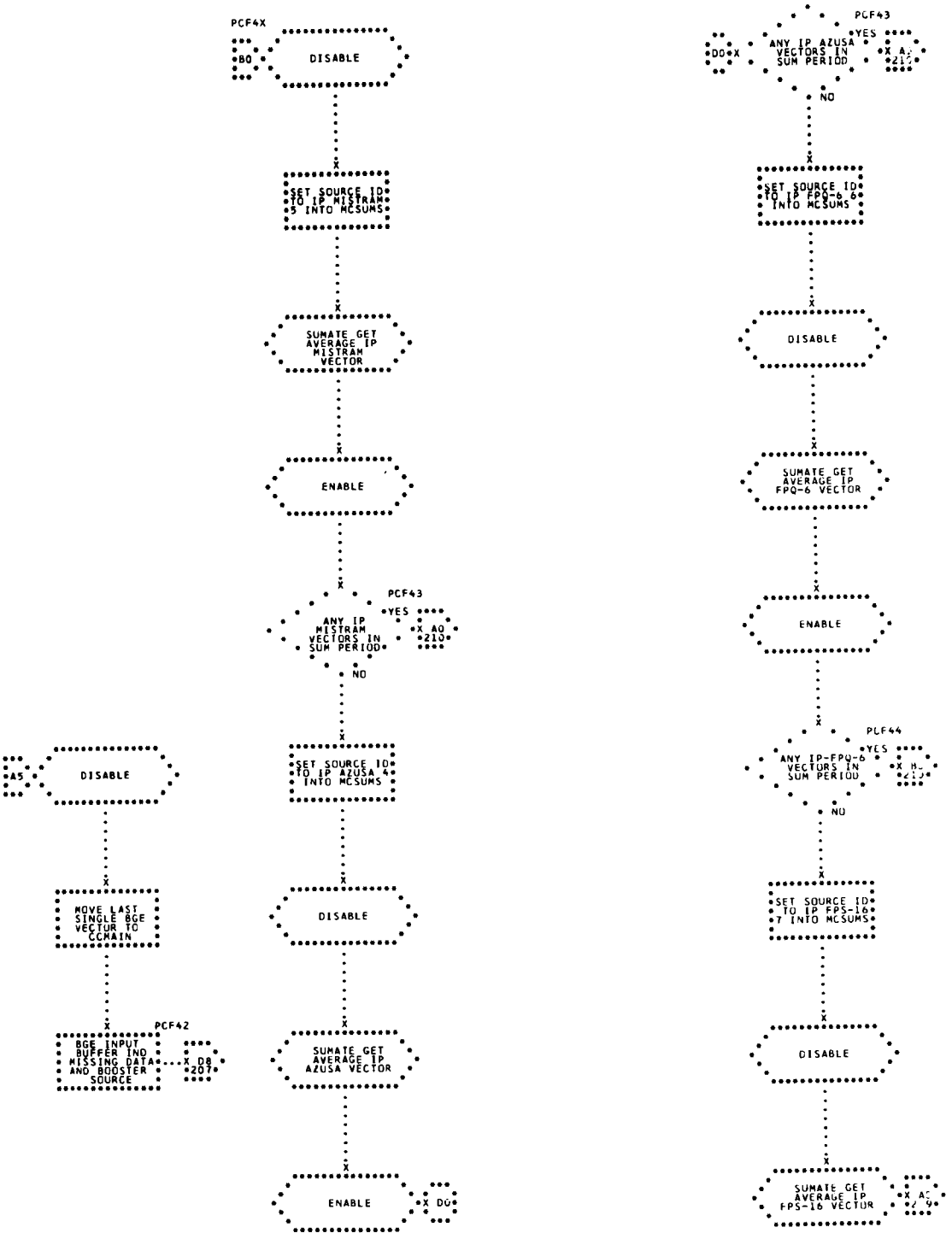

MPCCMN
START
TRNON A, MNCCMN
TRNOF B, MNCCMN
RESTART ENTRY MCREST NOT=0
YES
NO
LIFT OFF YET MCLFTM NOT=0
NO
YES
PCA30
DISABLE
PCA42
SELECTED SOURCE ID INTO XRI
PCA43
IP OR B-GE SELECTED
NO
YES
PCA44
SAVE IP OR B-GE VECTOR INTO MAINB
PCA43
VETOR T ZERO MAINB = C
NO
YES
B0
EXIT
LORB
PCA20
TRNOF A, MNCCMN
EXIT
MSPRIO
B4
EVER HAD A VECTOR TMSIGV NOT = 0
NO
YES
B0
PCA43
SAVE CURRENT T/M CONDITIONS
B1
PCB48
BDA SELECTED MCPSEL=2
NO
YES
AC +2C4
A1 +2D4

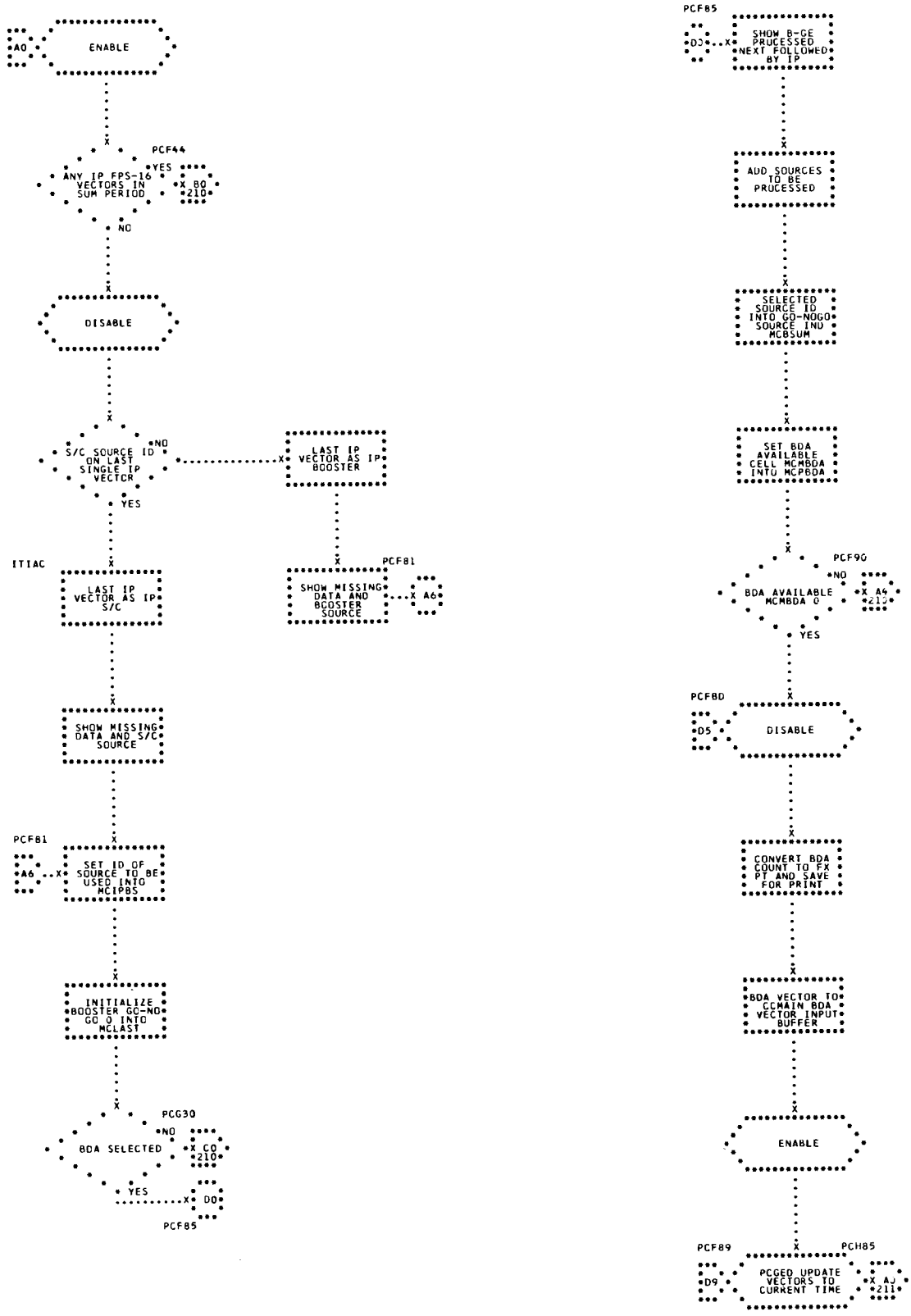

PCB48
SHOW BDA NOT SELECTED 0 INTO MCSBDA
A0
PCD74
NO
ABORT MODE 3 MCPABM NOT= 0
B0
X C1
203
YES
SHOW BDA SELECTED-NOT-0 INTO MCSBDA
A1
SHOW AM3 NOT-0 INTO MCUABM
3 INTO MCSSOR
3 INTO MCSIMP
PCC30
NO
SECO MCPSEC NOT= 0
X A0
203
YES
SHOW BDA AS SELECTED AND SOURCE IN PROCESS
PCC30
YES
SECO UNCONF IN HOLD PHASE MCSGNC NOT= 0
NO
PCA20
NO
EVER HAD VECTOR TMSTGV NOT= 0
X B4
203
YES
PCF05
NO
1ST OCCURRENCE SECO PCSEO=0
X C3
206
YES
PCB65
YES
IN HOLD PHASE MCCPHP NOT= 0
NO
SHOW SECO OCCURRED-NOT-0 INTO PCSEO
PCE75
MOVE STRIP CHARTS VALUES FOR IP AND AGE TO TCOUT
TOAR =T OF NON SECO*5 SEC INTO MCTOAR
X C1
206
PCB85
COMPUTE CURRENT TIME AND STORE IN MCURT
ENABLE
---PUNTS--- PRINT T/M EVENTS ON-LINE
X B0

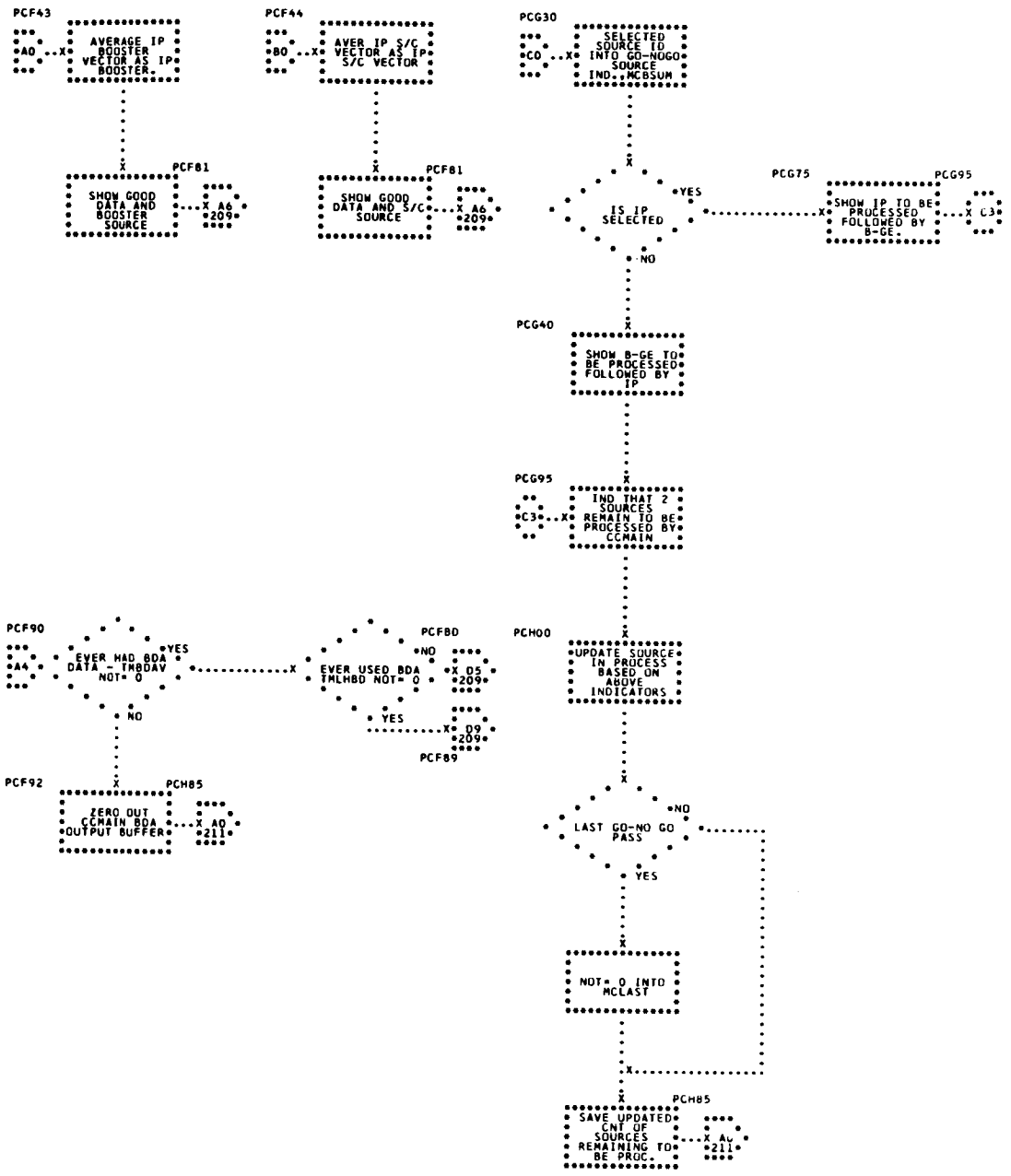

PCC30
2ND STAGE CUT MCPSSC NOT= 0
NO
YES
PCF05
TBAR SET - MCTBAR NOT= 0
YES
X C3 206
NO
T OF 2ND STAGE CUT/T OF SEC
T OF 2ND STAGE CUT = T SECO NOM INTO MCTSEC
PCE75
TBAR = T OF 2ND STAGE CUT + 4.5SEC INTO MCTBAR
X C1 206
PCC80
AAA - MCPAAA NOT= 0
NO
YES
PCF05
TBAR SET - MCTBAR NOT= 0
YES
X C3 206
NO
T OF AAA T OF SECO NOM INTO MCTSEC
PCE75
TBAR=T OF AAA +4.5 SEC INTO MCTBAR
X C1 206
PCD74
C1
SHOW AM3 NOT ESTAB - 0 INTO MCUABM
PCD75
IN HOLD PHASE - MCTBAR NOT= 0
NO
YES
REINITIALIZE FOR LAUNCH PHASE
PCD95
C4
RESET MISSING DATA CELL 2 INTO MCMISD
PCE55
BDA SELECTED - MCSBDA NOT= 0
YES
X B4 206
NO
PCE55
DATA ON SELECTED LINE
NO
X B4 206
YES
SET SELECTED SOURCE AND SOURCE IN PROCESS CELLS
PCE55
MISSING DATA ENTRY - MCMIDA NOT= 0
YES
X B6 206
NO
PCD98
MISSING DATA LIGHT ON
NO
X A4 206
YES
X A0 206

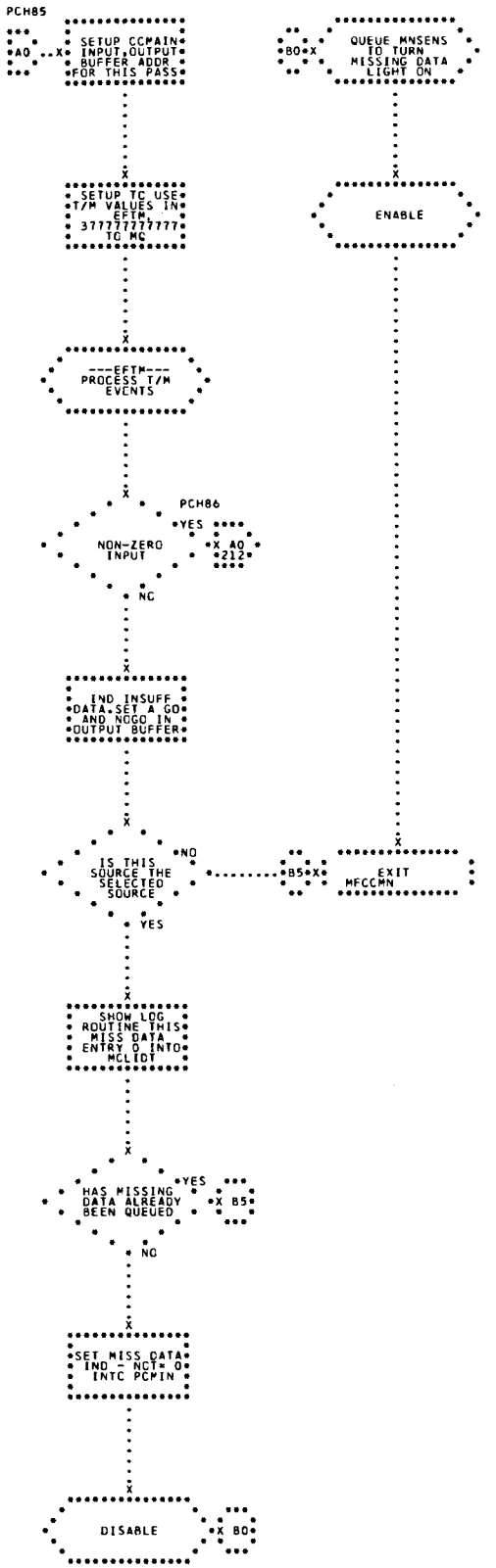

DISABLE
QUEUE MNSENS TO TURN LIGHT OFF
ENABLE
RESET MISS DATA LIGHT IND 0 INTO PCMIN
PCD98
SET UP SINGLE VECTOR INPUT FOR CCMAIN
PCE46
---EFTM--- PROCESS T/M EVENT
SET UP INPUT ADD FOR CCMAIN-0 INTO MCCFHP
SET UP OUTPUT ADD FOR CCMAIN FROM SOURCE IN PROC.
VECTOR FROM SOURCE TRACKING S/C
YES
NO
PCE5G
SHOW S/C 0 INTO DATA TRACK CELL
PCE55
MISSING DATA LIGHT ON
YES
NO
DISABLE
QUEUE MNSENS TO TURN LIGHT ON
ENABLE
PCE60
T OF MISS DATA = T OF LAST VEC +.5SEC INTO MCMISD
SHOW LOG ROUTINE THIS MISS DATA ENTRY-0 INTO MCLIDT
PCE5E
SHOW BOOSTER-NOT= 0 INTO DATA TRACK CELL
EXIT CCMAIN
PCE75
SET TIME SPAN FOR AVERAING VECTORS
PCF05
PCG02
TIME TO SAVE THE THRUST INDICATION
NO
YES
UPDATE T BY 5 SECONDS AND SAVE FOR NEXT CHECK
IS THRUSTING INDICATED
NO
YES
NOT= 0 INTO THRUSTING INDICATOR
NEGATIVE OF CURRENT T INTO AC TI INDICATE THRUSTING
PCG06
KEEP 7 LATEST TIMES OF THRUSTING IN THTHTI
PCG04
0 INTO THRUSTING INDICATOR
PCG0%
CURRENT TIME INTO AC TO INDICATE NO THRUSTING
PCG02
USE CURRENT T AS LATEST
MCTBAR INTO MCMINT=MIN ACCEPT T OF VECTOR
MCTBAR+SUM INTERVAL INTO MCMAXT=MAX ACCEPT T OF VEC
MCTBAR+SUM INTERVAL+2 SEC INTO MCTSUM=T TO AVE VEC

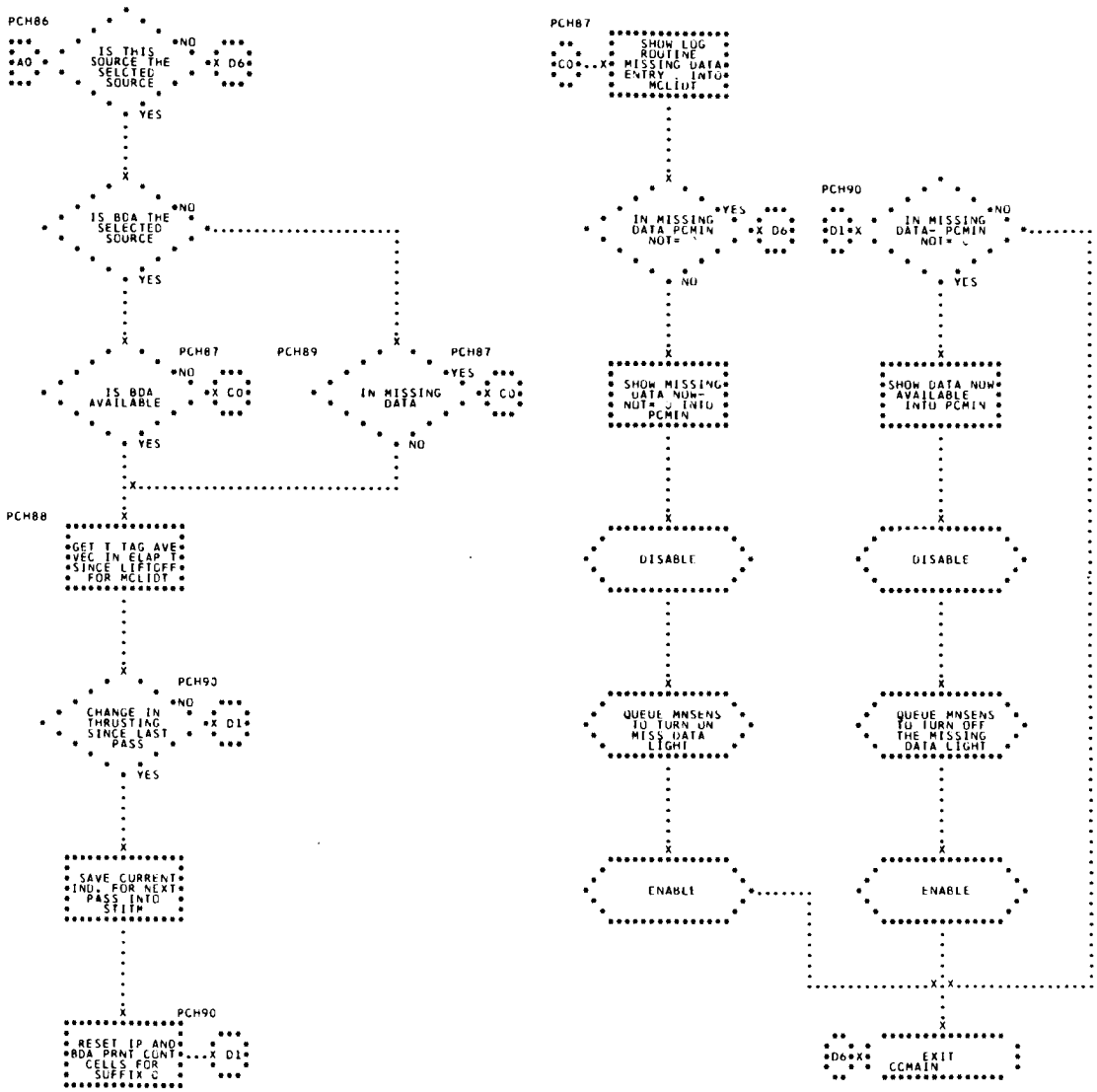

PCG02
PCI10
PCF40
BOOSTER SOURCE FOR GO-NO GO SUMMED
YES
X A0 213
A0
NO
TRNON C,MNMPRK SUP IMPACT PRED
B0
BDA LINES ACTIVATED
YES
NO
TRNOF A, PNMPRK
PCD95
TIME TO ACTIVATED BDA LINES
NO
X C4 205
YES
RESET LDGGING TIME 0 INTO MCLIDT
DISABLE
DISABLE
CHANGE ACTIVATE MASK TO INCORPORATE BDA LINES
CHNG COMP CYCLE FOR CCMAIN FROM 5 SEC TO 5 SEC
OPSLF ACTIVATE LINES
MISSING DATA LIGHT TURNED ON
NO
YES
ENABLE
RESET MISSING DATA LIGHT INDICATOR 0 INTO PCMIN
TRNOF C MNEDBD UNSUPP BDA EDIT PROG
QUEUE MNSENS TO TURN LIGHT OFF
PCF06
PCD95
PC444
TIME TO SUM MCCURT =TO OR GREATER THAN MCTSUM
NO
X C4 205
YES
X B0
PCF40
ENABLE
X D0
DO
DATA ON B-GE
NO
YES
SET SOURCE ID TO B-GE MOD III - 1 INTO MCSUMS
DISABLE
SUMATE GET AVERAGE B-GE VECTOR
ENABLE
ANY B-GE WITHIN SUM PERIOD
NO
X A5 2 8
YES
PCF41
MOVE AVERAGE B-GE VECTOR TO CCMAIN
B-GE INPUT BUFFER IND GOOD DATA AND BOOSTER SOURCE
PCF42
PCF4X
DATA ON IP
YES
X B5 2 8
D8
NO
PCF81
SET UP TO USE SOURCE ID OF AZUSA 1 INTO XR1
X A6 2 9

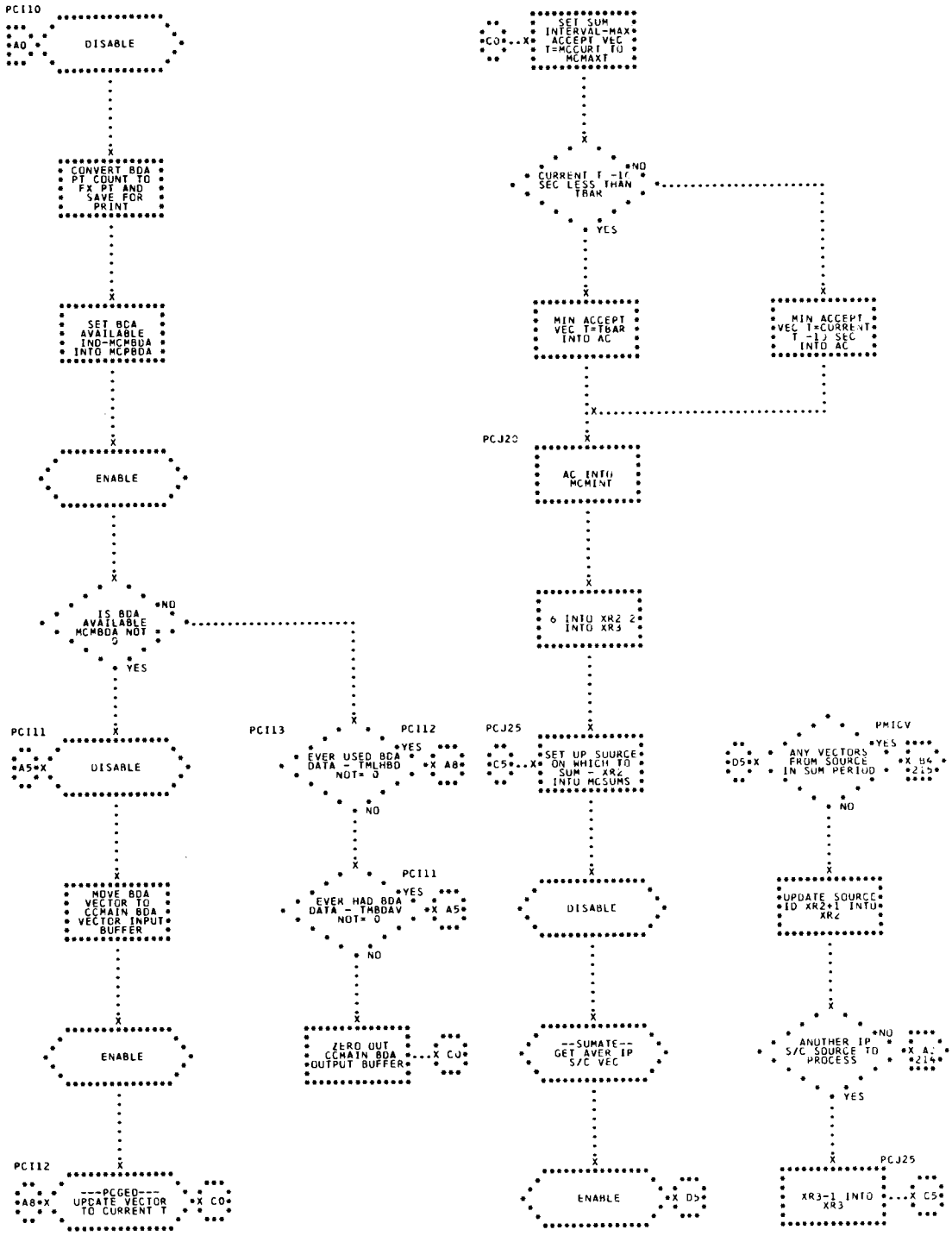

PCF4X
B0
DISABLE
SET SOURCE ID TO IP MISTRAM 5 INTO MCSUMS
SUMATE GET AVERAGE IP MISTRAM VECTOR
ENABLE
PCF43
ANY IP MISTRAM VECTORS IN SUM PERIOD
YES
X A0 210
NO
SET SOURCE ID TO IP AZUSA 4 INTO MCSUMS
DISABLE
SUMATE GET AVERAGE IP AZUSA VECTOR
ENABLE
X D0
A5
DISABLE
MOVE LAST SINGLE BGE VECTOR TO CCMAIN
PCF42
BGE INPUT BUFFER IND MISSING DATA AND BOOSTER SOURCE
X D8 207
PCF43
D0 X
ANY IP AZUSA VECTORS IN SUM PERIOD
YES
NO
SET SOURCE ID TO IP FPQ-6 6 INTO MCSUMS
DISABLE
SUMATE GET AVERAGE IP FPQ-6 VECTOR
ENABLE
PCF44
ANY IP-FPQ-6 VECTORS IN SUM PERIOD
YES
NO
SET SOURCE ID TO IP FPS-16 7 INTO MCSUMS
DISABLE
SUMATE GET AVERAGE IP FPS-16 VECTOR

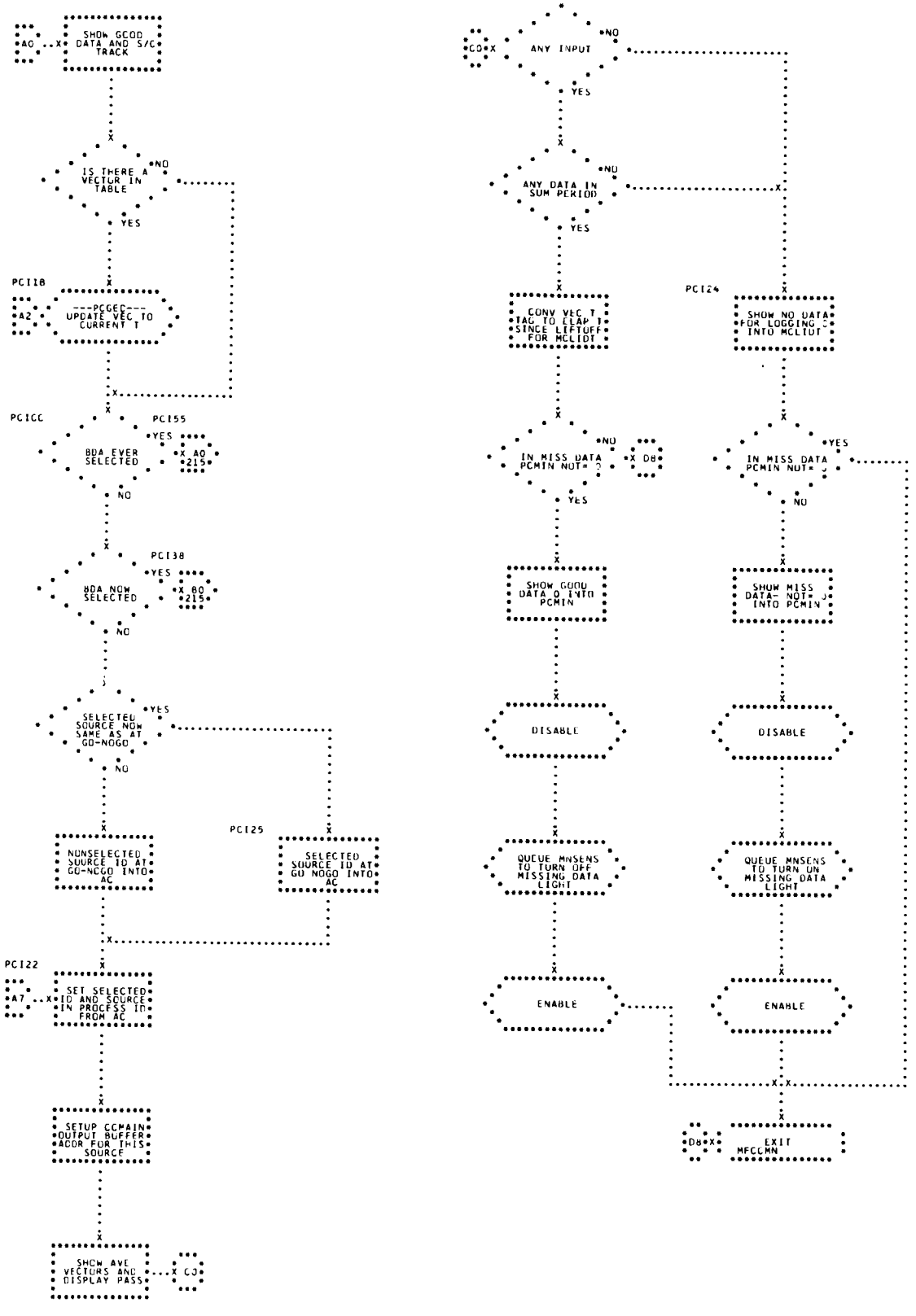

A0
ENABLE
PCF44
ANY IP FPS-16 VECTORS IN SUM PERIOD
YES
X BO
210
NO
DISABLE
S/C SOURCE ID ON LAST SINGLE IP VECTOR
NO
YES
LAST IP VECTOR AS IP BOOSTER
ITIAC
LAST IP VECTOR AS IP S/C
PCF81
SHOW MISSING DATA AND BOOSTER SOURCE
X A6
SHOW MISSING DATA AND S/C SOURCE
PCF81
A6
SET ID OF SOURCE TO BE USED INTO MCIPBS
INITIALIZE BOOSTER GO-NO GO 0 INTO MCLAST
PCG30
BDA SELECTED
NO
X CO
210
YES
DO
PCF85
PCF85
D3
SHOW B-GE PROCESSED NEXT FOLLOWED BY IP
ADD SOURCES TO BE PROCESSED
SELECTED SOURCE ID INTO GO-NOGO SOURCE IND MCBSUM
SET BDA AVAILABLE CELL MCMBDA INTO MCPBDA
PCF90
BDA AVAILABLE MCMBDA 0
NO
X A4
210
YES
PCF8D
D5
DISABLE
CONVERT BDA COUNT TO FX PT AND SAVE FOR PRINT
BDA VECTOR TO CCMAIN BDA VECTOR INPUT BUFFER
ENABLE
PCF89
D9
PCG6D UPDATE VECTORS TO CURRENT TIME
PCH85
X AJ
211

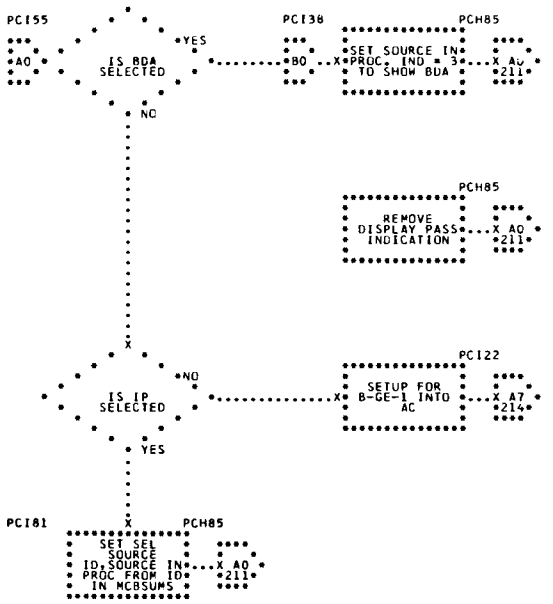

PCF43
AVERAGE IP BOOSTER VECTOR AS IP BOOSTER.
A0
PCF81
SHOW GOOD DATA AND BOOSTER SOURCE
A6
209
PCF44
AVER IP S/C VECTOR AS IP S/C VECTOR
B0
PCF81
SHOW GOOD DATA AND S/C SOURCE
A6
209
PCG30
SELECTED SOURCE ID INTO GO-NOGO SOURCE IND., MCBSUM
C0
IS IP SELECTED
YES
PCG75
PCG95
SHOW IP TO BE PROCESSED FOLLOWED BY B-GE.
C3
NO
PCG40
SHOW B-GE TO BE PROCESSED FOLLOWED BY IP
PCG95
IND THAT 2 SOURCES REMAIN TO BE PROCESSED BY CCMAIN
C3
PCH00
UPDATE SOURCE IN PROCESS BASED ON ABOVE INDICATORS
PCF90
EVER HAD BDA DATA - TMBDAV NOT= 0
A4
YES
NO
PCF80
EVER USED BDA TMLHBD NOT= 0
NO
D5
209
YES
D9
209
PCF89
PCF92
ZERO OUT CCMAIN BDA OUTPUT BUFFER
PCH85
A0
211
LAST GO-NO GO PASS
NO
YES
NOT= 0 INTO MCLAST
PCH85
SAVE UPDATED CNT OF SOURCES REMAINING TO BE PROC.
A4
211

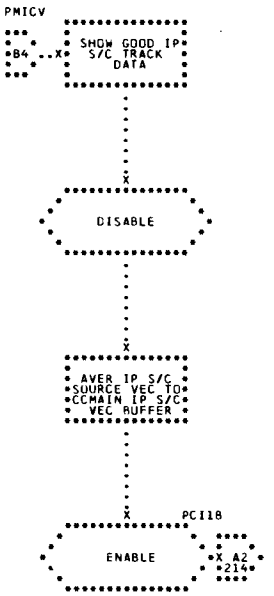

PCH85
SETUP CCMAIN INPUT,OUTPUT BUFFER ADDR FOR THIS PASS
AO
X
SETUP TC USE T/M VALUES IN EFTM 377777777777 TO MC
---EFTM--- PROCESS T/M EVENTS
PCH86
NON-ZERO INPUT
YES
X AO 212
NO
IND INSUFF DATA-SET A GO AND NOGO IN OUTPUT BUFFER
IS THIS SOURCE THE SELECTED SOURCE
NO
B5
X
EXIT MFCCMN
YES
SHOW LOG ROUTINE THIS MISS DATA ENTRY 0 INTO MCLIDT
HAS MISSING DATA ALREADY BEEN QUEUED
YES
X B5
NO
SET MISS DATA IND - NCT- 0 INTC PCMIN
DISABLE
X B0
B0
X
QUEUE MNSENS TO TURN MISSING DATA LIGHT ON
ENABLE

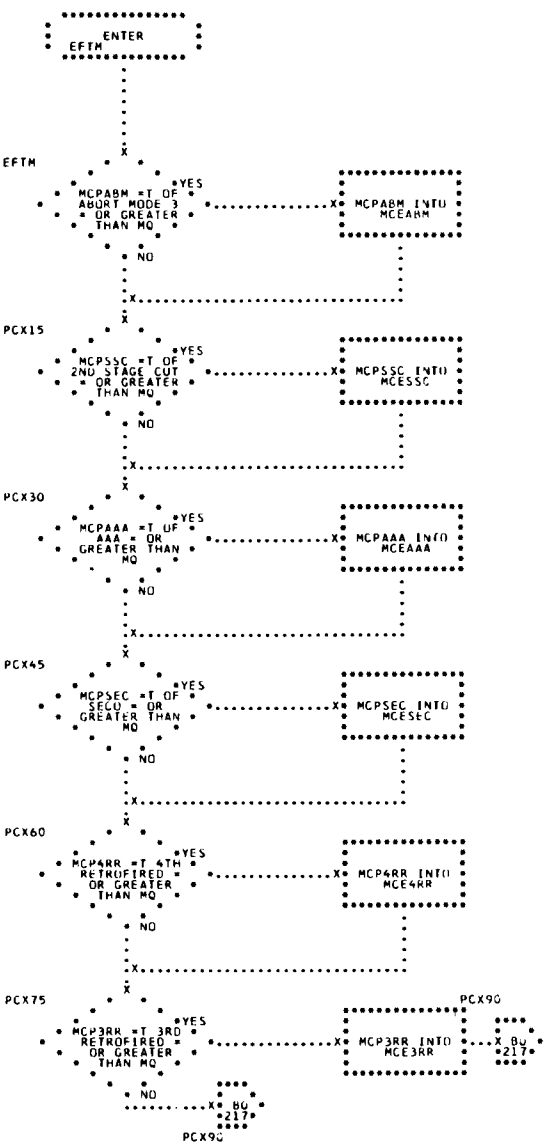

PCH86
IS THIS SOURCE THE SELCTED SOURCE
NO
A0
D6
YES
IS BDA THE SELECTED SOURCE
NO
YES
PCH87
IS BDA AVAILABLE
NO
C0
YES
PCH89
PCH87
IN MISSING DATA
YES
C0
NO
PCH88
GET T TAG AVE VEC IN ELAP T SINCE LIFTOFF FOR MCLIDT
PCH90
CHANGE IN THRUSTING SINCE LAST PASS
NO
D1
YES
SAVE CURRENT IND. FOR NEXT PASS INTO STIM
PCH90
RESET IP AND BDA PRNT CONT CELLS FOR SUFFIX C
D1
PCH87
SHOW LOG ROUTINE MISSING DATA ENTRY INTO MCLIDT
C0
IN MISSING DATA PCMIN NOT=
YES
D6
NO
SHOW MISSING DATA NOW- NOT= INTO PCMIN
DISABLE
QUEUE MNSENS TO TURN ON MISS DATA LIGHT
ENABLE
PCH90
IN MISSING DATA= PCMIN NOT=
D1
NO
YES
SHOW DATA NOW AVAILABLE INTO PCMIN
DISABLE
QUEUE MNSENS TO TURN OFF THE MISSING DATA LIGHT
ENABLE
D6
EXIT CCMAIN

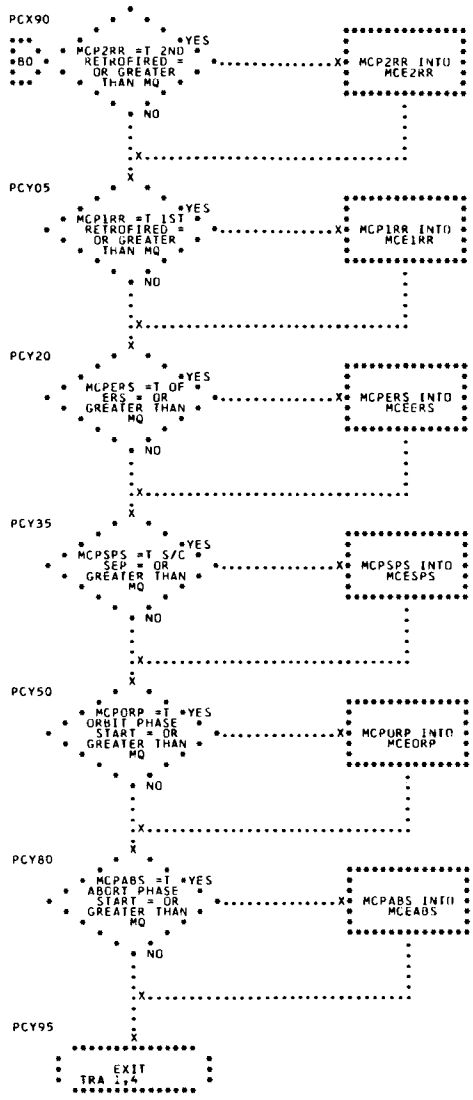

PCI10
A0
DISABLE
CONVERT BDA PT COUNT TO FX PT AND SAVE FOR PRINT
SET BDA AVAILABLE IND=MCMBDA INTO MCPBDA
ENABLE
IS BDA AVAILABLE MCMBDA NOT 0
NO
YES
PCI11
A5
DISABLE
MOVE BDA VECTOR TO CCMAIN BDA VECTOR INPUT BUFFER
ENABLE
PCI12
A8
---PCGED---
UPDATE VECTOR TO CURRENT T
C0
PCI13
PCI12
EVER USED BDA DATA - TMLHBD NOT= 0
YES
A8
NO
PCI11
EVER HAD BDA DATA - TMBDAV NOT= 0
YES
A5
NO
ZERO OUT CCMAIN BDA OUTPUT BUFFER
C0
C0
SET SUM INTERVAL=MAX ACCEPT VEC T=MCCURT TO MCMAXT
CURRENT T -1C SEC LESS THAN TBAR
NO
YES
MIN ACCEPT VEC T=TBAR INTO AC
MIN ACCEPT VEC T=CURRENT T -1C SEC INTO AC
PCJ20
AC INTO MCMINT
6 INTO XR2 2 INTO XR3
PCJ25
C5
SET UP SOURCE ON WHICH TO SUM - XR2 INTO MCSUMS
DISABLE
--SUMATE-- GET AVER IP S/C VEC
ENABLE
D5
PMICV
D5
ANY VECTORS FROM SOURCE IN SUM PERIOD
YES
B4
215
NO
UPDATE SOURCE ID XR2+1 INTO XR2
ANOTHER IP S/C SOURCE TO PROCESS
NO
A3
214
YES
PCJ25
XR3-1 INTO XR3
C5

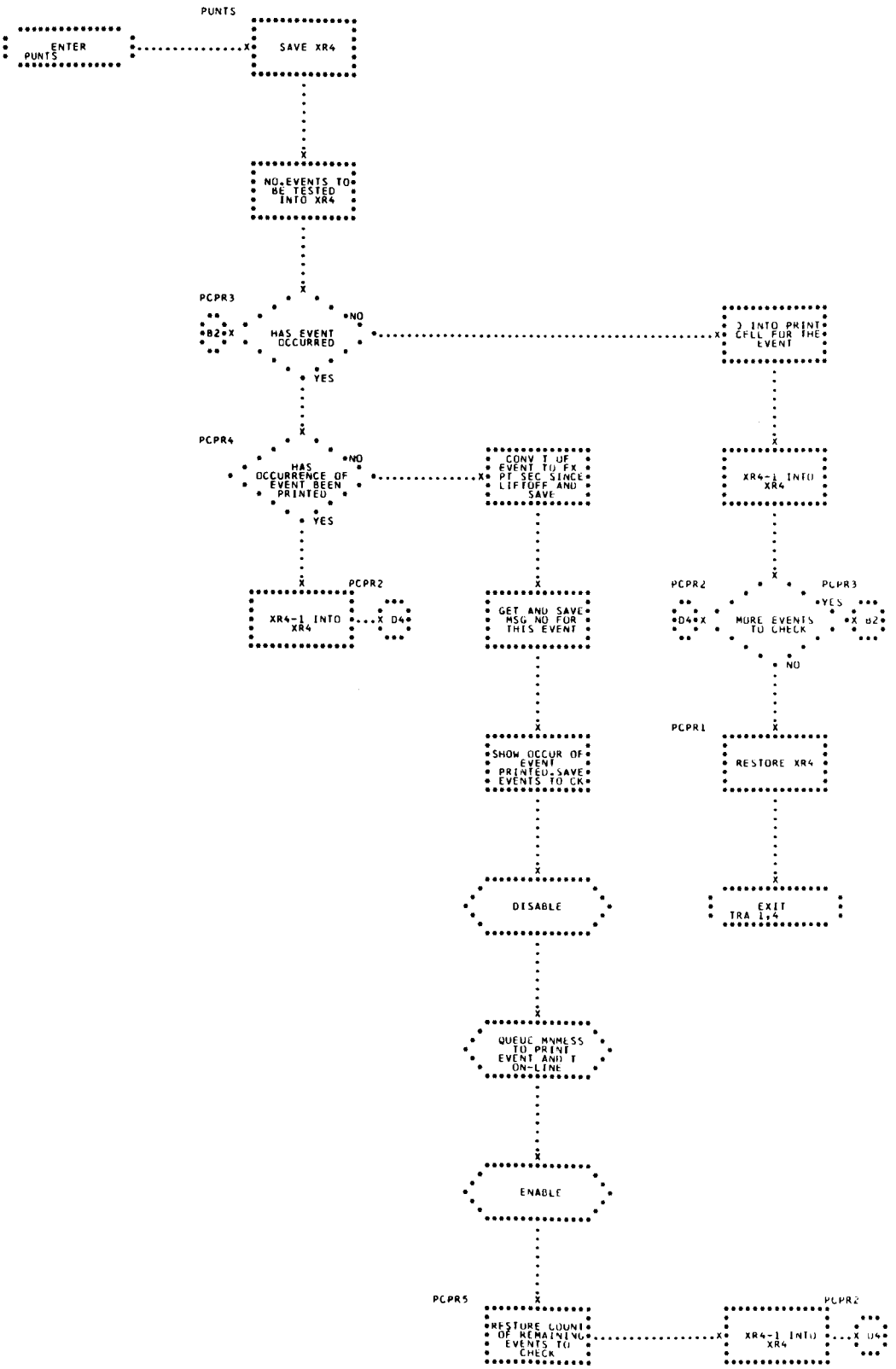

SHOW GOOD DATA AND S/C TRACK
IS THERE A VECTOR IN TABLE
NO
YES
PCI18
---PCGEE--- UPDATE VEC TO CURRENT T
PCICC
PCI55
BDA EVER SELECTED
YES
NO
PCI38
BDA NOW SELECTED
SELECTED SOURCE NOW SAME AS AT GO-NOGO
NONSELECTED SOURCE ID AT GO-NOGO INTO AC
PCI25
SELECTED SOURCE ID AT GO NOGO INTO AC
PCI22
SET SELECTED ID AND SOURCE IN PROCESS IN FROM AC
SETUP GCMAIN OUTPUT BUFFER ADDR FOR THIS SOURCE
SHOW AVE VECTORS AND DISPLAY PASS
ANY INPUT
ANY DATA IN SUM PERIOD
CONV VEC T TAG TO ELAP T SINCE LIFTOFF FOR MCLIDT
PCI24
SHOW NO DATA FOR LOGGING INTO MCLIDT
IN MISS DATA PCMIN NOT= 0
SHOW GOOD DATA 0 INTO PCMIN
SHOW MISS DATA- NOT= 0 INTO PCMIN
DISABLE
QUEUE MNSENS TO TURN OFF MISSING DATA LIGHT
QUEUE MNSENS TO TURN ON MISSING DATA LIGHT
ENABLE
EXIT MFCCMN

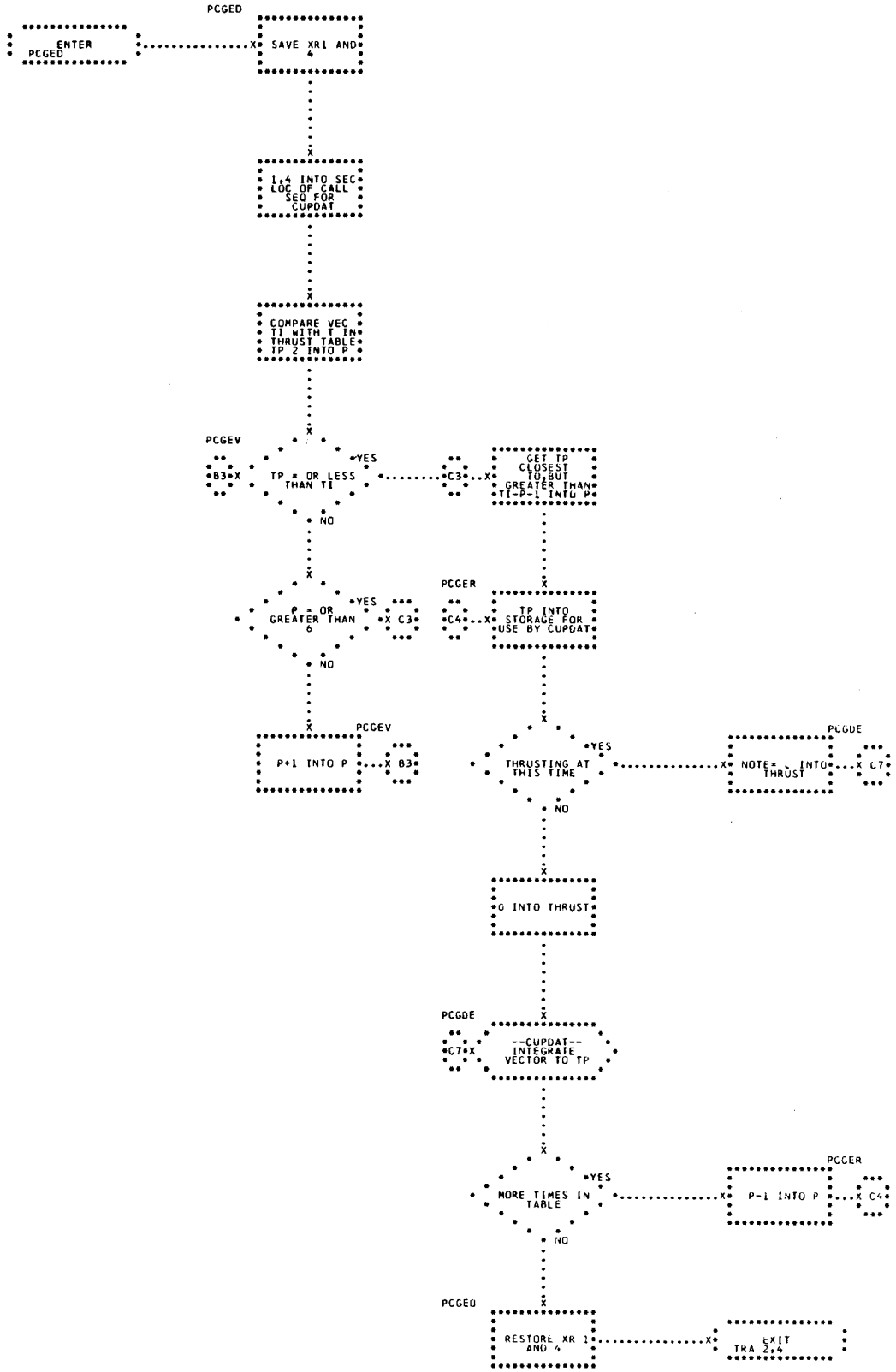

PCI55
PCI38
PCH85
YES
AO
IS BDA SELECTED
BO
SET SOURCE IN PROC. IND = 3 TO SHOW BDA
X AO 211
NO
PCH85
REMOVE DISPLAY PASS INDICATION
X AO 211
PCI22
NO
IS IP SELECTED
SETUP FOR B-GE-1 INTO AC
X A7 214
YES
PCI81
PCH85
SET SEL SOURCE ID, SOURCE IN PROC FROM ID IN MCBSUMS
X AO 211


PMICV
B4
SHOW GOOD IP S/C TRACK DATA
DISABLE
AVER IP S/C SOURCE VEC TO CCMAIN IP S/C VEC BUFFER
PCI18
ENABLE
X A2 214


ENTER EFTM
EFTM
MCPABM =T OF ABORT MODE 3 = OR GREATER THAN MQ
YES
NO
MCPABM INTO MCEABM
PCX15
MCPSSC =T OF 2ND STAGE CUT = OR GREATER THAN MQ
YES
NO
MCPSSC INTO MCESSC
PCX30
MCPAAA =T OF AAA = OR GREATER THAN MQ
YES
NO
MCPAAA INTO MCEAAA
PCX45
MCPSEC =T OF SEC0 = OR GREATER THAN MQ
YES
NO
MCPSEC INTO MCESEC
PCX60
MCP4RR =T 4TH RETROFIRED = OR GREATER THAN MQ
YES
NO
MCP4RR INTO MCE4RR
PCX75
MCP3RR =T 3RD RETROFIRED = OR GREATER THAN MQ
YES
NO
MCP3RR INTO MCE3RR
PCX90
X 80 *217*
PCX90
X 80 *217*


PCX90
MCP2RR =T 2ND RETROFIRED = OR GREATER THAN MQ
YES
MCP2RR INTO MCE2RR
NO
PCY05
MCP1RR =T 1ST RETROFIRED = OR GREATER THAN MQ
YES
MCP1RR INTO MCE1RR
NO
PCY20
MCPERS =T OF ERS = OR GREATER THAN MQ
YES
MCPERS INTO MCEERS
NO
PCY35
MCPSPS =T S/C SEP = OR GREATER THAN MQ
YES
MCPSPS INTO MCESPS
NO
PCY50
MCPORP =T ORBIT PHASE START = OR GREATER THAN MQ
YES
MCPORP INTO MCEORP
NO
PCY80
MCPABS =T ABORT PHASE START = OR GREATER THAN MQ
YES
MCPABS INTO MCEABS
NO
PCY95
EXIT
TRA 1,4


PUNTS
ENTER PUNTS
SAVE XR4
NO. EVENTS TO BE TESTED INTO XR4
PCPR3
B2
HAS EVENT OCCURRED
NO
YES
) INTO PRINT CELL FOR THE EVENT
PCPR4
HAS OCCURRENCE OF EVENT BEEN PRINTED
NO
YES
CONV T OF EVENT TO FX PT SEC SINCE LIFTOFF AND SAVE
XR4-1 INTO XR4
PCPR2
XR4-1 INTO XR4
D4
GET AND SAVE MSG NO FOR THIS EVENT
PCPR2
D4
MORE EVENTS TO CHECK
PCPR3
YES
B2
NO
SHOW OCCUR OF EVENT PRINTED-SAVE EVENTS TO CK
PCPR1
RESTORE XR4
DISABLE
EXIT TRA 1,4
QUEUE MNMESS TO PRINT EVENT AND T ON-LINE
ENABLE
PCPR5
RESTORE COUNT OF REMAINING EVENTS TO CHECK
XR4-1 INTO XR4
PCPR2
D4


PCGED
ENTER PCGED
SAVE XR1 AND 4
1,4 INTO SEC LOC OF CALL SEQ FOR CUPDAT
COMPARE VEC TI WITH T IN THRUST TABLE TP 2 INTO P
PCGEV
B3
TP = OR LESS THAN TI
YES
C3
GET TP CLOSEST TO BUT GREATER THAN TI-P-1 INTO P
NO
P = OR GREATER THAN 6
YES
C3
PCGER
C4
TP INTO STORAGE FOR USE BY CUPDAT
NO
PCGEV
P+1 INTO P
B3
THRUSTING AT THIS TIME
YES
PCGDE
NOTE= INTO THRUST
C7
NO
0 INTO THRUST
PCGDE
C7
--CUPDAT-- INTEGRATE VECTOR TO TP
MORE TIMES IN TABLE
YES
PCGER
P-1 INTO P
C4
NO
PCGEO
RESTORE XR 1 AND 4
EXIT TRA 1,4

34. CCMAIN - LAUNCH MAIN CALCULATIONS PROGRAM

THIS PROGRAM IS USED DURING THE LAUNCH PHASE TO PRODUCE THE REQUIRED QUANTITIES FOR OUTPUT DISPLAYS. (SEE FLOWCHART.)

INPUT

A) TABLES

TCSTAG	A ONE-CELL TABLE CONTAINING THE NOMINAL TIME OF STAGING IN FLOATING POINT SECONDS SINCE LIFT-OFF.
TC.ON1	A ONE-CELL TABLE CONTAINING THE NOMINAL TIME FROM SECOND STAGE CUTOFF TO MINIMUM DELAY GMT IN FLOATING POINT SECONDS, N_1
TC.ONA	A ONE-CELL TABLE CONTAINING THE NOMINAL OAMS ACCELERATION IN FLOATING POINT GEMINI ACCELERATION UNITS, N_a
TC.ONT	A ONE-CELL TABLE CONTAINING THE NOMINAL OAMS THRUSTING TIME IN FLOATING SECONDS, N_t
TPVVRB	A THREE-LOCATION TABLE CONTAINING THE PARAMETERS USED TO COMPUTE THE NOMINAL VELOCITY RATIO BEFORE STAGING OCCURS
TPVVRA	A THREE-LOCATION TABLE CONTAINING THE PARAMETERS USED TO COMPUTE THE NOMINAL VELOCITY RATIO AFTER STAGING OCCURS
TPYNMB	A THREE-LOCATION TABLE CONTAINING THE PARAMETERS USED TO COMPUTE THE

		NOMINAL FLIGHT PATH ANGLE BEFORE STAGING OCCURS
	TPYNMA	A THREE-LOCATION TABLE CONTAINING THE PARAMETERS USED TO COMPUTE THE NOMINAL FLIGHT PATH ANGLE AFTER STAGING OCCURS

B) CONSTANTS

K00001	DEC 1 FIXED POINT ONE
K00002	DEC 2 FIXED POINT TWO
K00003	DEC 3 FIXED POINT THREE
K001.0	DEC 1.0 FLOATING POINT ONE
K00.SG	CONVERSION FROM SECONDS TO GEMINI TIME UNITS
K.70KM	70,000 FEET IN FLOATING POINT GEMINI LENGTH UNITS

C) COMMUNICATION CELLS

MCEAAA	CONTAINS TIME OF ASTRONAUT ACTUATED ABORT (AAA) IN GMT FLOATING POINT SECONDS
MCEABM	CONTAINS TIME OF ABORT MODE CHANGE-OVER (AMC) IN GMT FLOATING POINT SECONDS
MCERTS	CONTAINS TIME OF EMERGENCY RETRO SALVO IN GMT FLOATING POINT SECONDS
MCESPS	CONTAINS TIME OF SPACECRAFT SEPARATION IN GMT FLOATING POINT SECONDS
MCLFTM	CONTAINS TIME OF LIFT-OFF IN FLOATING POINT SECONDS
MCLLIN	CONTAINS ADDRESS OF INPUT TABLE FOR THIS PASS
MCLLOT	CONTAINS ADDRESS OF OUTPUT TABLE FOR THIS PASS

MCMISD	CONTAINS TIME OF MISSING DATA IN GMT FLOATING POINT SECONDS
MCTSEC	CONTAINS COMPUTER'S BEST ESTIMATE OF SUSTAINER ENGINE CUTOFF TIME IN GMT FLOATING POINT SECONDS
MCCFHP	NON-ZERO IF AVERAGE VECTORS COMPUTATIONS ARE DESIRED

OUTPUT

A) TABLE

L(OUTPUT)	THE OUTPUT TABLE SPECIFIED BY THE ADDRESS OF MCLLOT CONTAINS THE QUANTITIES COMPUTED FOR OUTPUT DISPLAYS FROM THE INPUT VECTORS AND THEIR ASSOCIATED TIME (SEE CCMAIN OUTPUT TABLE.)

METHOD

AFTER SAVING THE CONTENTS OF INDEX REGISTERS 1,2 AND 4 AND OF THE SENSE INDICATORS, CCMAIN PICKS UP THE ADDRESSES OF THE INPUT AND OUTPUT TABLES TO BE USED. IT THEN TESTS THE EVENT CELLS AND PROCEEDS ACCORDINGLY.

A) IF MCCFHP IS NON-ZERO, THE HOLD PHASE OF LAUNCH PROCESSING HAS BEEN ENTERED. CONTROL IS THEN TRANSFERRED TO THE CFCFAV SUBROUTINE WHICH PERFORMS THE AVERAGE VECTORS COMPUTATIONS ON THE INPUT VECTORS AND STORES THE RESULTS IN THE SPECIFIED OUTPUT TABLE. WHEN CONTROL IS RETURNED TO CCMAIN, THE INDEX REGISTERS AND SENSE INDICATORS ARE RESTORED AND TRANSFER IS MADE TO THE SUFFIX, MFCCMN.

B) IF MCCFHP IS ZERO, CONTROL IS GIVEN TO THE SUBROUTINE, CPRVCM, WHICH PERFORMS ALL PRESENT VECTORS COMPUTATIONS. THE RESULTS ARE THEN PLACED IN THE INTERNAL STORAGE TABLE OF THE SUBROUTINE. CONTROL PASSES TO C).

C) IF MCMISD IS ZERO, CONTROL IS PASSED TO D). IF MCMISD IS NON-ZERO, THIS IS A MISSING DATA ENTRY AND CCMAIN DETERMINES WHETHER OR NOT SUSTAINER ENGINE CUTOFF HAS OCCURRED. IF IT HAS AND THE TIME OF THE MISSING DATA ENTRY IS GREATER THAN THE TIME OF SUSTAINER ENGINE CUTOFF, NO COMPUTATIONS OF THE VELOCITY RATIO AND THE FLIGHT PATH ANGLE ARE MADE SINCE THE CURVE FITS ARE FROZEN TO SUSTAINER ENGINE CUTOFF TIME.

IF, HOWEVER, EITHER SUSTAINER ENGINE CUTOFF HAS NOT OCCURRED OR THE TIME OF THE MISSING DATA ENTRY IS EARLIER THAN THE TIME OF SUSTAINER ENGINE CUTOFF, THE MISSING DATA TIME IS CONVERTED TO ELAPSED TIME SINCE LIFT-OFF AND IS COMPARED TO THE NOMINAL TIME OF STAGING.

THE RESULTS OF THIS COMPARISON DETERMINE WHICH SET OF NOMINAL CURVE FITS WILL BE USED IN THE COMPUTATION OF THE VELOCITY RATIO AND THE FLIGHT PATH ANGLE. THE RESULTS OF THE COMPUTATION REPLACE THE VELOCITY RATIO AND THE FLIGHT PATH ANGLE PRODUCED BY THE CPRVCM SUBROUTINE.

D) CPRVCM ARE NOW TRANSFEERED TO THE SPECIFIED OUTPUT TABLE.

E) IF MCEABM IS NON-ZERO, ABORT MODE CHANGEOVER (AMC) HAS OCCURRED AND CCMAIN SETS THE ABORT MODE INDICATOR TO SHOW THAT ABORT MODE 3 IS NOW IN EFFECT (I.E., IF AN ABORT OCCURS, THE RETRO-ROCKETS WILL BE RIPPLE FIRED TO SLOW THE SPACE-CRAFT DOWN RATHER THAN SALVO FIRED TO ACHIEVE SEPARATION OF SPACECRAFT AND BOOSTER).

THE MINIMUM DELAY TIME TO RETROFIRE DELTA $t_{r\,min}$ IS THEN COMPUTED AS FOLLOWS -

DELTA $t_{r\,min} = N_1$ IF EITHER SUSTAINER ENGINE CUTOFF HAS NOT OCCURRED OR HAS OCCURRED BUT HAS AN ASSOCIATED TIME EQUAL TO OR GREATER THAN THE TIME OF THE INPUT VECTORS.

DELTA $t_{r\,min} = (t_{ssco} + N_1 - t_i)$, IF SUSTAINER ENGINE CUTOFF HAS OCCURRED AND ITS ASSOCIATED TIME IS LESS THAN THE TIME OF THE INPUT VEC-TORS.

FOLLOWING THIS COMPUTATION, CCMAIN DECIDES WHETHER AND BY HOW MUCH THE GIVEN VELOCITY VEC-TOR MUST BE MODIFIED TO REFLECT OAMS BURN FOR SPACECRAFT SEPARATION. THE FINAL VELOCITY VEC-TOR IS PLACED IN THE OUTPUT TABLE AND CCMAIN EXITS TO ITS SUFFIX, MFCCMN, AFTER RESTORING IN-DEX REGISTERS AND THE SENSE INDICATORS.

F) IF AMC HAS NOT OCCURRED, BUT MCEAAA IS NON-ZERO (I.E., ASTRONAUT ACTUATED ABORT (AAA) HAS OCCURRED), CCMAIN CHECKS THE TIME OF EMERGENCY RETRO SALVO ($t_{e.r.s.}$).

IF EMERGENCY RETRO SALVO HAS NOT YET OCCURRED, THIS TIME HAS NOT BEEN SET, AND THE BOSTER AND SPACECRAFT ARE STILL JOINED. THUS, THE GIVEN VELOCITY VECTOR PROVIDES THE MOST ACCURATE DESCRIPTION OF THE SPACECRAFT'S SPEED AND IS PLACED IN THE OUTPUT TABLE WITHOUT MODIFICATION, AND CONTROL PASSES TO 3).

IF EMERGENCY RETRO SALVO HAS OCCURRED, BUT THE SOURCE OF THE VECTORS IS TRACKING THE SPACECRAFT, THE GIVEN VELOCITY VECTOR REFLECTS THE EFFECT OF RETRO SALVO. THUS, IN THIS CASE ALSO, THE VECTOR MAY BE PLACED IN THE OUTPUT TABLE WITHOUT MODIFICATION. CONTROL IS THEN TRANSFERRED TO 3).

IF, HOWEVER, EMERGENCY RETRO SALVO HAS OCCURRED AND THE VECTORS ARE FROM A SOURCE TRACKING THE BOOSTER, THE FOLLOWING STEPS ARE TAKEN -

1) THE TIME OF EMERGENCY RETRO SALVO IS SAVED IN INTERNAL STORAGE.
2) THE SUBROUTINE CDELTV IS THEN ENTERED TO UPDATE THE INPUT VELOCITY VECTOR TO REFLECT RETRO SALVO BASED ON THE FIRING OF FOUR RETROROCKETS AND THE HEIGHT OF THE SPACECRAFT. THE INPUT TIME AND POSITION VECTOR AND THE MODIFIED VELOCITY VECTOR ARE PLACED IN THE OUTPUT TABLE AND ARE SAVED IN INTERNAL STORAGE.

3) THE ABORT MODE TYPE INDICATOR IS SET TO INDICATE THAT ABORT MODE 2 IS IN EFFECT.

4) THE MINIMUM DELAY TIME TO RETROFIRE IS SET TO ZERO AND CCMAIN EXITS TO ITS SUFFIX, MFCCMN.

5) THE LATEST TIME OF EMERGENCY RETRO SALVO IS CHECKED AGAINST THE SAVED TIME. IF THE NEWER TIME IS GREATER, THE STEPS FROM 1) TO 4) ARE EXECUTED. IF THE LATEST TIME DOES NOT WARRANT THE ABOVE PROCESS, THE LAST VECTORS SAVED ARE AGAIN PLACED IN THE OUTPUT TABLE AND CONTROL GOES TO 3).

G) IF MCEAAA IS ZERO, THE SUBROUTINE CDELTV IS USED TO UPDATE THE GIVEN VECTOR. THE MODIFIED VELOCITY VECTOR IS PLACED IN THE OUTPUT TABLE AND CONTROL IS PASSED TO 3).

USAGE

CCMAIN IS ENTERED FROM ITS PREFIX, MPCCMN. IT TRANSFERS CONTROL TO ITS SUFFIX, MFCCMN.

A) STORAGE REQUIRED -
225 LOCATIONS

B) TIMING (7094)

1) MCMISD NON-ZERO -
6.169 MILLISECONDS, MAXIMUM

2) MCEABM NON-ZERO -
5.914 MILLISECONDS, MAXIMUM

3) MCEAAA NON-ZERO -
6.128 MILLISECONDS, MAXIMUM

4) MCCFHP NON-ZERO -
.046 MILLISECONDS, MAXIMUM

5) MCEAGM, MCEAAA, MCCFHP ZERO -
6.006 MILLISECONDS, MAXIMUM

TO THE TIME OF MCMISD MUST BE ADDED THE TIMING FOR ANY OTHER EVENT.

THE TIMING REQUIRED FOR MCCFHP NON-ZERO DOES NOT INCLUDE THE TIME NECESSARY TO EXECUTE THE CFCFAV PROGRAM ITSELF.

TABLE 34-1. INPUT TABLE TO CCMAIN (SHEET 1 OF 2)

MCLLIN CONTAINS IN THE ADDRESS THE ADDRESS OF THE INPUT TABLE WHICH IS ARRANGED AS FOLLOWS -

L(INPUT)	t	TIME OF VECTORS IN GMT FLOATING POINT SECONDS
+1	x	POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+2	y	
+3	z	
+4	$\dot{x}$	VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+5	$\dot{y}$	
+6	$\dot{z}$	
+7	S.I	SOURCE INDICATOR - 0, IF SPACECRAFT SOURCE 1, IF BOOSTER SOURCE
+8	Q.I	QUALITY OF DATA INDICATOR - 0, IF VECTORS ACCEPTABLE FOR AVERAGE VECTORS COMPUTATIONS, 1, IF VECTORS UNACCEPTABLE FOR THE ABOVE COMPUTATIONS
+9		VELOCITY RATIO DIFFERENCE (I.E., ACTUAL VELOCITY RATIO MINUS NOMINAL VELOCITY RATIO) - NOT USED DURING AVERAGE VECTORS COMPUTATIONS

TABLE 34-1. INPUT TABLE TO CCMAIN (SHEET 2 OF 2)

+10	FLIGHT PATH ANGLE DIFFERENCE (I.E., ACTUAL FLIGHT PATH ANGLE MINUS NOMINAL FLIGHT PATH ANGLE) - NOT USED DURING AVERAGE VECTORS COMPUTATIONS

OUTPUT TABLE FROM CCMAIN (SHEET 1 OF 6)

MCLLOT CONTAINS THE ADDRESS OF THE OUTPUT TABLE WHICH IS ARRANGED AS FOLLOWS -

DURING NORMAL LAUNCH -

L(OUTPUT)			
	t_u		THE TIME OF THE VECTORS IN FLOATING POINT GMT SECONDS
+1	x	$\bar{r}_u$	x-COMPONENT OF THE POSTION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+2	y		y-COMPONENT OF THE POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+3	z		z-COMPONENT OF THE POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+4	$\dot{x}$	$\bar{v}_u$	$\dot{x}$-COMPONENT OF THE VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+5	$\dot{y}$		y-COMPONENT OF THE VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+6	$\dot{z}$		$\dot{z}$-COMPONENT OF THE VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+7	γ		FLIGHT PATH ANGLE IN FLOATING POINT RADIANS
+8	v/v_r		VELOCITY RATIO IN FLOATING POINT (DIMENSIONLESS)

OUTPUT TABLE FROM CCMAIN (SHEET 2 OF 6)

+9	i	INCLINATION ANGLE IN FLOATING POINT RADIANS
+10	$r-\bar{R}$	HEIGHT OF SPACECRAFT ABOVE A SPHERICAL EARTH IN FLOATING POINT NAUTICAL MILES
+11	ϕ_{pp}	LATITUDE OF PRESENT POSITION IN FLOATING POINT RADIANS
+12	λ_{pp}	LONGITUDE OF PRESENT POSITION IN FLOATING POINT RADIANS
+13	s	CROSSRANGE DISTANCE IN FLOATING POINT NAUTICAL MILES
+14	d	DOWNRANGE DISTANCE IN FLOATING POINT NAUTICAL MILES
+15	AMI	ABORT MODE INDICATOR IN FIXED POINT, B35
+16	$\Delta t_{r\ min}$	MINIMUM DELAY TIME TO RETROFIRE IN FLOATING POINT SECONDS
+17	v'	VELOCITY IN FEET PER SECOND DURING THE HOLD PHASE
L(OUTPUT)	t_u	THE TIME OF THE VECTORS IN FLOATING POINT GMT SECONDS
+1	$x\ \bar{r}_u$	a-COMPONENT OF POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS

OUTPUT TABLE FROM CCMAIN (SHEET 3 OF 6)

+2	y	$\bar{r}_u$	y-COMPONENT OF POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+3	z		z-COMPONENT OF POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+4	$\dot{x}$	$\bar{v}_u$	$\dot{x}$-COMPONENT OF VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+5	$\dot{y}$		$\dot{y}$-COMPONENT OF VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+6	$\dot{z}$		$\dot{z}$-COMPONENT OF VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+7	γ		FLIGHT PATH ANGLE IN FLOATING POINT RADIANS
+8	v/v_r		VELOCITY RATIO IN FLOATING POINT (DIMENSIONLESS)
+9	i		INCLINATION ANGLE IN FLOATING POINT RADIANS
+10	$r-\overline{R}$		HEIGHT OF SPACECRAFT ABOVE A SPHERICAL EARTH IN FLOATING POINT NAUTICAL MILES
+11	φ_{pp}		LATITUDE OF PRESENT POSITION IN FLOATING POINT RADIANS
+12	λ_{pp}		LONGITUDE OF PRESENT POSITION IN FLOATING POINT RADIANS

OUTPUT TABLE FROM CCMAIN (SHEET 4 OF 6)

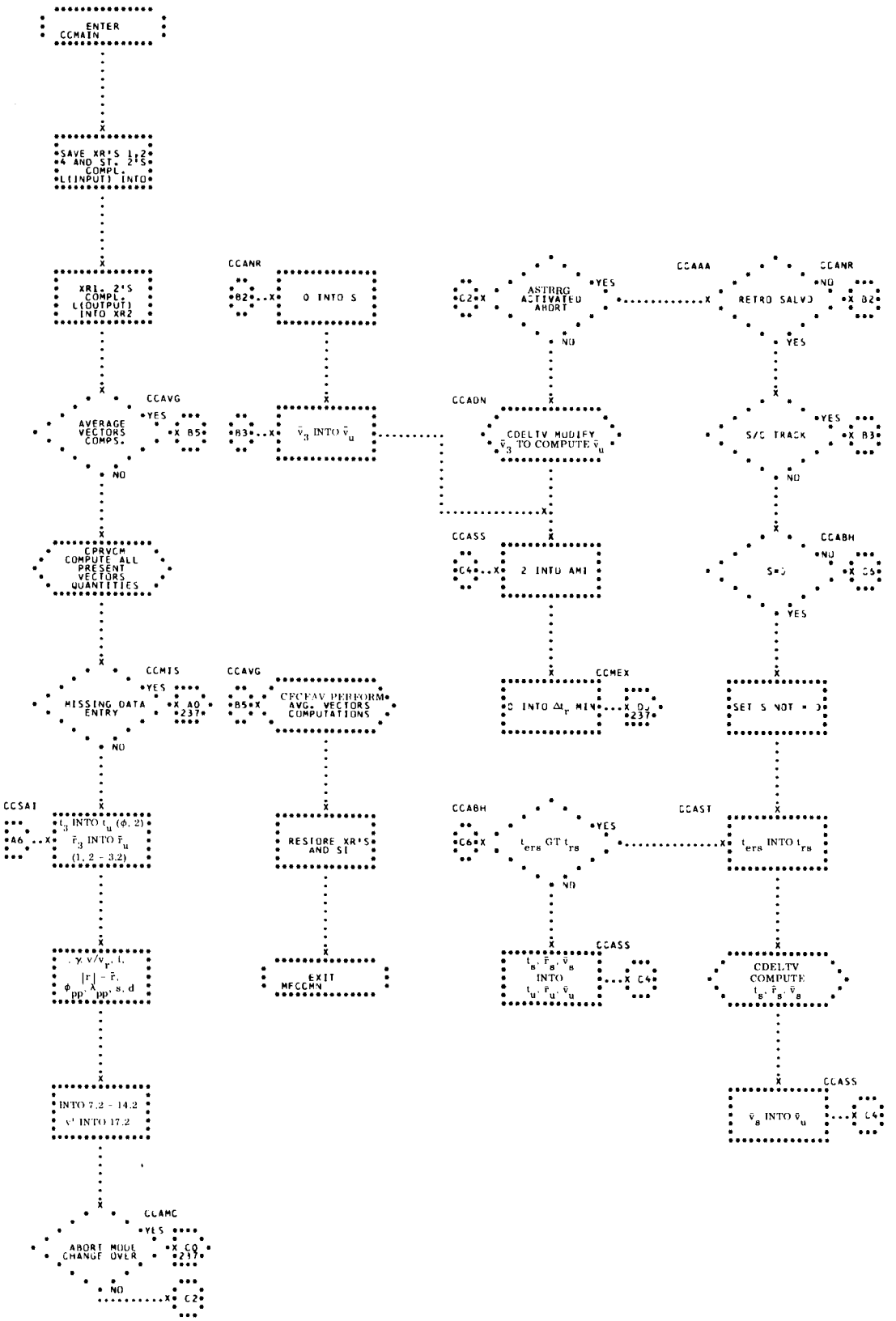
+13		NOT USED
+14		NOT USED
+15	Go/No-Go	ABORT RECOMMENDATION IN OCTAL
+16	$-\Delta V_{G'}$	NEGATIVE OF INCERMENT TO SPACECRAFT VELOCITY TO ACHIEVE 1.5 REVOLUTION ORBIT LIFETIME IN FLOATING POINT FEET PER SECOND
+17	$-\Delta V$	NEGATIVE OF TOTAL INCREMENT TO SPACECRAFT VELOCITY TO ACHIEVE OPTIMUM ORBITAL CONDITIONS IN FLOATING POINT FEET PER SECOND
+18	$\phi_{\text{min delay}}$	LATITUDE OF MINIMUM DELAY IMPACT IN FLOATING POINT RADIANS
+19	$\lambda_{\text{min delay}}$	LONGITUDE OF MINIMUM DELAY IMPACT IN FLOATING POINT RADIANS
+20	ϕ_{IP}	LATITUDE OF IMPACT IN FLOATING POINT RADIANS
+21	λ_{IP}	LONGITUDE OF IMPACT IN FLOATING POINT RADIANS
+22	GMTRC	RECOMMENDED RETROFIRE TIME IN GMT FLOATING POINT SECONDS

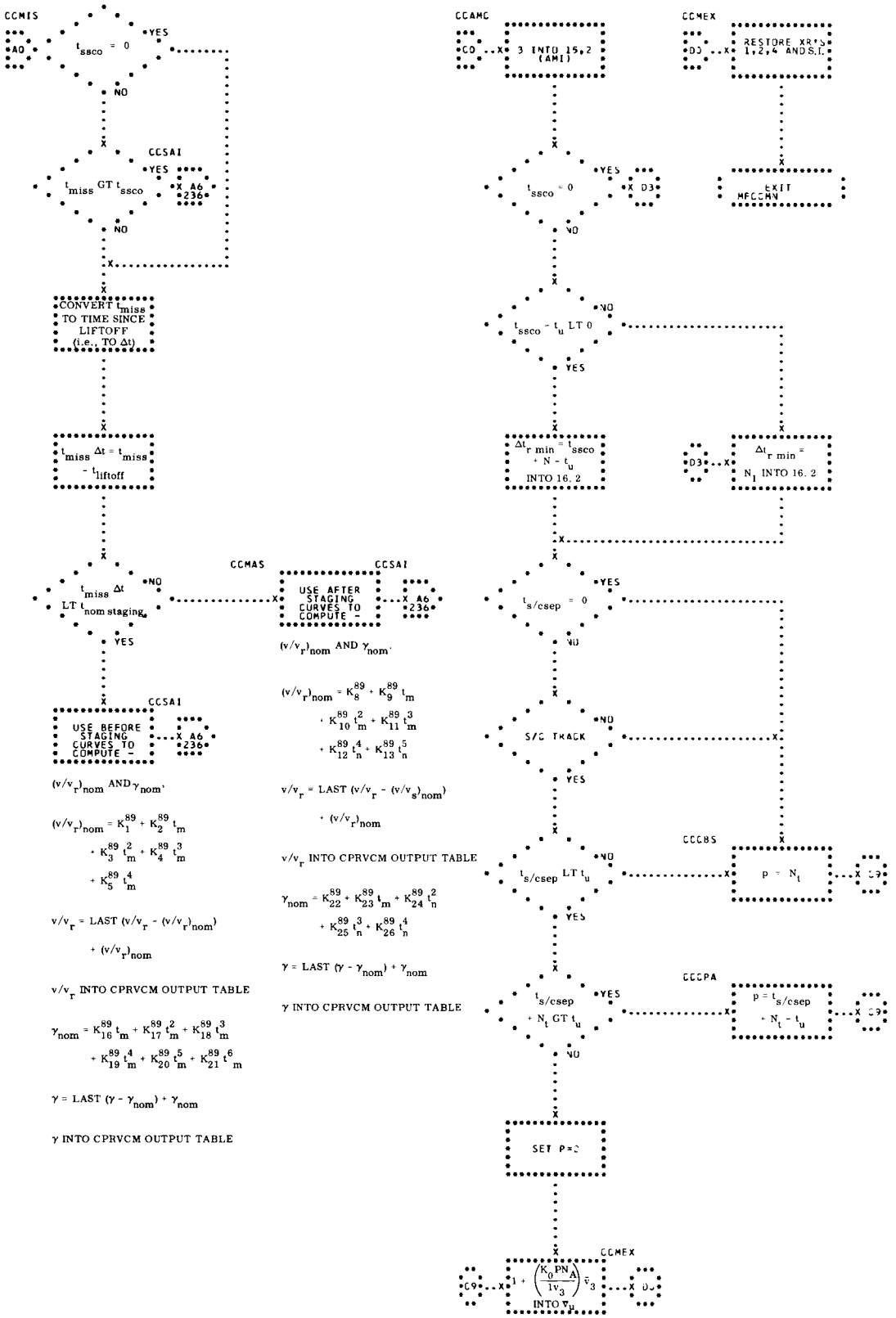
OUTPUT TABLE FROM CCMAIN (SHEET 5 OF 6)

+23	ESTRC	RECOMMENDED RETROFIRE TIME IN ELAPSED TIME SINCE LIFT-OFF, FLOATING POINT SECONDS
+24	R. A.	RECOVERY AREA IN OCTAL
+25	$V_{act.}$	ACTUAL SPACECRAFT VELOCITY IN FLOATING POINT FEET PER SECOND
+26	GMTPC	RECOMMENDED TIME TO COMPLETE PROPULSION TO ACHIEVE 1.5 ORBITS IN GMT FLOATING POINT SECONDS
+27	V_{350K}	RELATIVE VELOCITY MAGNITUDE AT 350 THOUSAND FEET IN FLOATING POINT FEET PER SECOND
+28	γ_{350K}	RELATIVE FLIGHT PATH ANGLE AT 350 THOUSAND FEET IN FLOATING POINT RADIANS
+29	e	ECCENTRICITY IN FLOATING POINT, DIMENSIONLESS
+30	H_{ap}	KEPLERIAN HEIGHT AT APOGEE IN FLOATING POINT NAUTICAL MILES
+31	H_{per}	KEPLERIAN HEIGHT AT PERIGEE IN FLOATING POINT NAUTICAL MILES

OUTPUT TABLE FROM CCMAIN (SHEET 6 OF 6)

+32	ΔV_D	VELOCITY INCREMENT TO BE APPLIED AT APOGEE TO RAISE THE HEIGHT AT THE FOLLOWING PERIGEE TO 75 NAUTICAL MILES FEET PER SECOND
+33	Δt_D	TIME DURATION OF NECESSARY OAMS BURN TO ACHIEVE FLOATING POINT SECONDS
+34	GMTPI	GREENWICH MEAN TIME AT WHICH TO INITIATE OAMS BURN, FLOATING POINT SECONDS
+35	GETPI	ELAPSED TIME FROM LIFT-OFF TO TIME TO INITIATE OAMS BURN, FLOATING POINT SECONDS


ENTER CCMAIN
SAVE XR'S 1,2 4 AND ST. 2'S COMPL. L(INPUT) INTO
XR1, 2'S COMPL. L(OUTPUT) INTO XR2
CCAVG
AVERAGE VECTORS COMPS.
YES
NO
B5
CPRVCM COMPUTE ALL PRESENT VECTORS QUANTITIES
CCMIS
MISSING DATA ENTRY
A0 237
CCSAI
A6
t_3 INTO t_u (ϕ, 2)
$\bar{r}_3$ INTO $\bar{r}_u$ (1, 2 - 3.2)
γ, v/v_r, i, $|r| - \bar{r}$, ϕ_{pp}, λ_{pp}, s, d
INTO 7.2 - 14.2
v' INTO 17.2
CCAMC
ABORT MODE CHANGE OVER
C0 237
C2
CCANR
B2
0 INTO S
B3
$\bar{v}_3$ INTO $\bar{v}_u$
CCAVG
B5
CFCFAV PERFORM AVG. VECTORS COMPUTATIONS
RESTORE XR'S AND SI
EXIT MFCCMN
C2
ASTRRG ACTIVATED ABORT
CCADN
CDELTV MODIFY $\bar{v}_3$ TO COMPUTE $\bar{v}_u$
CCASS
C4
2 INTO AMI
CCMEX
0 INTO Δt_r MIN
D0 237
CCABH
C6
t_{ers} GT t_{rs}
CCASS
t_s, $\bar{r}_s$, $\bar{v}_s$ INTO t_u, $\bar{r}_u$, $\bar{v}_u$
C4
CCAAA
RETRO SALVO
CCANR
B2
S/C TRACK
B3
CCABH
S=0
C5
SET S NOT = 0
CCAST
t_{ers} INTO t_{rs}
CDELTV COMPUTE t_s, $\bar{r}_s$, $\bar{v}_s$
CCASS
$\bar{v}_s$ INTO $\bar{v}_u$
C4


CCMIS
A0
$t_{ssco} = 0$
YES
NO
CCSAI
t_{miss} GT t_{ssco}
YES
A6 236
NO
CONVERT t_{miss} TO TIME SINCE LIFTOFF (i.e., TO Δt)
$t_{miss}\ \Delta t = t_{miss} - t_{liftoff}$
$t_{miss}\ \Delta t$ LT $t_{nom\ staging}$
NO
YES
CCMAS
CCSAI
USE BEFORE STAGING CURVES TO COMPUTE -
A6 236
$(v/v_r)_{nom}$ AND γ_{nom}
$(v/v_r)_{nom} = K_1^{89} + K_2^{89} t_m + K_3^{89} t_m^2 + K_4^{89} t_m^3 + K_5^{89} t_m^4$
v/v_r = LAST $(v/v_r - (v/v_r)_{nom}) + (v/v_r)_{nom}$
v/v_r INTO CPRVCM OUTPUT TABLE
$\gamma_{nom} = K_{16}^{89} t_m + K_{17}^{89} t_m^2 + K_{18}^{89} t_m^3 + K_{19}^{89} t_m^4 + K_{20}^{89} t_m^5 + K_{21}^{89} t_m^6$
γ = LAST $(\gamma - \gamma_{nom}) + \gamma_{nom}$
γ INTO CPRVCM OUTPUT TABLE
CCSAI
USE AFTER STAGING CURVES TO COMPUTE -
A6 236
$(v/v_r)_{nom}$ AND γ_{nom}
$(v/v_r)_{nom} = K_8^{89} + K_9^{89} t_m + K_{10}^{89} t_m^2 + K_{11}^{89} t_m^3 + K_{12}^{89} t_n^4 + K_{13}^{89} t_n^5$
v/v_r = LAST $(v/v_r - (v/v_s)_{nom}) + (v/v_r)_{nom}$
v/v_r INTO CPRVCM OUTPUT TABLE
$\gamma_{nom} = K_{22}^{89} + K_{23}^{89} t_m + K_{24}^{89} t_n^2 + K_{25}^{89} t_n^3 + K_{26}^{89} t_n^4$
γ = LAST $(\gamma - \gamma_{nom}) + \gamma_{nom}$
γ INTO CPRVCM OUTPUT TABLE
CCAMC
C0
3 INTO 15,2 (AMI)
$t_{ssco} = 0$
YES
D3
NO
$t_{ssco} - t_u$ LT 0
NO
YES
$\Delta t_{r\ min} = t_{ssco} + N - t_u$ INTO 16.2
CCMEX
D3
RESTORE XR'S 1,2,4 AND S.I.
EXIT MPCCMN
D3
$\Delta t_{r\ min} = N_1$ INTO 16.2
$t_{s/csep} = 0$
YES
NO
S/C TRACK
NO
YES
CCCBS
$t_{s/csep}$ LT t_u
NO
YES
$p = N_1$
C9
CCCPA
$t_{s/csep} + N_t$ GT t_u
YES
NO
$p = t_{s/csep} + N_t - t_u$
C9
SET P=2
CCMEX
C9
$1 + \left(\frac{K_0 PN_A}{1v_3}\right) \bar{v}_3$ INTO $\bar{v}_p$
D0

35. CFCFAV - LAUNCH COMPUTATIONS FOR HOLD PHASE

THIS PROGRAM IS USED TO PROCESS THE AVERAGE VECTORS AND TO GENERATE THE DISPLAYS DURING THE HOLD PHASE. (SE FLOWCHART.)

INPUT

A) TABLES -

L(INPUT)	AN ELEVEN-LOCATION TABLE CONTAINING THE TIME AND THE DATA SOURCE AND QUALITY INDICATORS. SEE CCMAIN TABLE.
TC.ON1	THE NOMINAL TIME FROM SECOND STAGE CUTOFF TO MINIMUM DELAY GMT IN FLOATING POINT SECONDS, N_1
TC.ON2	THE NOMINAL TIME FROM COMPLETION OF THRUSTING TO MINIMUM DELAY GMT IN FLOATING POINT SECONDS, N_2
TC.ONA	THE NOMINAL ACCELERATION DUE TO OAMS BURN FOR SEPARATION IN FLOATING POINT GEMINI ACCELERATION UNITS, N_a
TC.ONT	THE NOMINAL THRUSTING TIME FOR OAMS BURN FOR SPACE CRAFT SEPARATION IN FLOATING POINT SECONDS, N_t
TC.NTF	THE NOMINAL ACCELERATION DUE TO OAMS BURN FOR SEPARATION IN FLOATING POINT FEET/SECOND^2, N_f
TCGMT1	CONTAINS NOMINAL TIME OF MINIMUM DELA GMT IN FLOATING POINT SECONDS BASED (THE TIME OF SECOND STAGE CUTOFF (COMPUTED ON FIRST PASS THROUGH CFCFAV

WITH A BOOSTER SOURCE, USED AS INPUT THEREAFTER)

TCICRT — CONTAINS INCLINATION ANGLE CRITERION IN FLOATING POINT RADIANS

TCRADF — CODE OF RECOVERY AREA F IN OCTAL

TCRADG — CODE OF RECOVERY AREA G IN OCTAL

TCRALF — LONGITUDE OF RECOVERY AREA F IN FLOATING POINT RADIANS

TCRALG — LONGITUDE OF RECOVERY AREA G IN FLOATING POINT RADIANS

TCRCAD — A SEVEN-LOCATION TABLE CONTAINING THE CODES OF THE RECOVERY AREAS IN OCTAL

TCRCAL — A NINE-LOCATION TABLE CONTAINING THE LONGITUDES OF THE RECOVERY AREAS IN FLOATING POINT RADIANS

TPVEGO — A THREE-LOCATION TABLE OF PARAMETERS USED IN COMPUTING V_{go}

B) COMMUNICATION CELLS -

MCCURT — CONTAINS CURRENT TIME IN GMT FLOATING POINT SECONDS, $t_{current}$

MCELFT — CONTAINS SETTING OF LIFT SWITCH IN OCTAL

0 - NO LIFT

1 - PROGRAMMED LIFT

2 - MAX. LIFT

MCESPS — CONTAINS TIME OF SPACECRAFT SEPARATION IN GMT FLOATING POINT SECONDS, $t_{s/c\ sep}$

MCLFTM — CONTAINS TIME OF LIFTOFF IN FLOATING POINT SECONDS, $t_{L.O.}$

MCTBEG — CONTAINS TIME OF BEGINNING OF THRUSTING IN GMT FLOATING POINT SECONDS, $t_{begin\ thrust}$

MCTEND	CONTAINS TIME OF ENDING OF THRUSTING IN GMT FLOATING POINT SECONDS, $t_{\text{end thrust}}$	
MCTSEC	CONTAINS COMPUTER'S BEST ESTIMATE OF OCCURRENCE OF SECOND STAGE CUTOFF IN GMT FLOATING POINT SECONDS, t_{SSCO}	

C) CONSTANTS -

K00001	DEC 1	FIXED POINT ONE
K00001	DEC 2	FIXED POINT TWO
K00003	DEC 3	FIXED POINT THREE
K000 4	DEC 4	FIXED POINT FOUR
K00007	DEC 7	FIXED POINT SEVEN
K001.0	DEC 1.0	FLOATING POINT ONE
K0020.	DEC 20.0	FLOATING POINT TWENTY
K00.PI		PI IN FLOATING POINT RADIANS
K00.SG		CONVERSION OF SECONDS TO GEMINI TIME UNITS
K00.UT		CONVERSION TO GEMINI TIME UNITS TO SECONDS
KFTSCG		CONVERSION OF GEMINI VELOCITY UNITS TO FEET/ SECOND
KGAMLL		LOWER LIMIT ON FLIGHT PATH ANGLE IN FLOATING POINT DEGREES
KGAMUL		UPPER LIMIT ON FLIGHT PATH ANGLE IN FLOATING POINT DEGREES
KVELLL		LOWER LIMIT ON VELOCITY IN FLOATING POINT FEET/SECOND
KVELUL		UPPER LIMIT ON VELOCITY IN FLOATING POINT FEET/SECOND
KMAXDV		MAXIMUM VELOCITY REQUIRED TO

ACHIEVE V_{go} IN FLOATING POINT FEET/SECOND

OUTPUT

TABLE

L(OUTPUT) CONTAINS QUANTITIES FOR OUTPUT DISPLAYS COMPUTED COMPUTED FROM THE AVERAGE VECTORS (SEE CCMAIN TABLE.)

METHOD

A) AFTER SAVING THE CONTENTS OF THE INDEX REGISTERS, CFCFAV PICKS UP THE ADDRESS OF THE INPUT AND OF THE OUTPUT TABLES TO BE USED THIS PASS. THE GIVEN TIME AND POSITION VECTOR COMPONENTS ARE THEN PLACED IN THE OUTPUT TABLE. BEFORE THE VELOCITY VECTOR COMPONENTS MAY BE USED, CFCFAV DECIDES WHETHER THE GIVEN VELOCITY VECTOR SHOULD BE MODIFIED TO REFLECT OAMS BURN FOR SPACECRAFT SEPARATION AND, AS A RESULT, WHAT VALUE SHOULD BE ASSIGNED TO THE VALID TIME OF THE ACTUAL VELOCITY. IN CASE VECTOR MODIFICATION IS NECESSARY, CFCFAV COMPUTES AND SAVES AS THE MAGNITUDE OF THE ACTUAL VELOCITY VECTOR, THE MAGNITUDE OF THE INPUT VELOCITY VECTOR. THEN, BASED ON ITS DECISIONS, CFCFAV ADDS THE EFFECT OF OAMS BURN TO THE INPUT VELOCITY VECTOR COMPONENTS AND PLACES THE RESULTING COMPONENTS IN THE OUTPUT TABLE. FROM THIS POINT ON, THE ORIGINAL TIME AND POSITION VECTOR, WITH THE MODIFIED VELOCITY VECTOR ARE USED AS THE MOST ACCURATE DESCRIPTION OF EVENTS TAKING PLACE IN THE OUTSIDE WORLD.

USING THE ABOVE VECTORS, THE SUBROUTINE CPRVCM IS ENTERED FOR THE COMPUTATION OF ALL PRESENT VECTORS QUANTITIES. THESE RESULTS ARE PLACED IN THE OUTPUT TABLE. THE INCLINATION ANGLE COMPUTED BY CPRVCM IS USED TO SELECT THE CODE AND LONGITUDE OF EITHER RECOVERY ARE F OR G FOR TCRCAD AND TCRCAL TABLES, RESPECTIVELY. WHEN THE INCLINATION ANGLE IS LESS THAN THE CRITERION, AREA F IS USED, OTHERWISE, AREA G IS USED. THIS PREVENTS THE SPACECRAFT FROM LANDING AMONG A SMALL GROUP OF ISLANDS OFF THE WEST COAST OF AFRICA IN AN ABORT SITUATION WHERE AREA F WOULD NORMALLY BE SELECTED.*

B) THE ABORT RECOMMENDATION IS THEN MADE AS FOLLOWS -

1) IF THE VECTORS ARE FROM A SOURCE DEEMED ACCEPTABLE FOR THE COMPUTATION OF THE MINIMUM VELOCITY NEEDED TO ACHIEVE ORBIT AND BOTH THE FLIGHT PATH ANGLE AND THE EXPECTED VELOCITY FALL WITH THEIR RESPECTIVE LIMITS, THE MINIMUM VELOCITY NEEDED TO ACHIEVE ORBIT IS COMPUTED AS A FUNCTION OF THE LAST TWO QUANTITIES.

THE DIFFERENCE BETWEEN THE EXPECTED VELOCITY AND THE MINIMUM VELOCITY NEEDED TO ACHIEVE ORBIT IS THEN COMPUTED AND COMPARED TO ZERO. IF NEGATIVE, A NO-GO RECOMMENDATION IS GIVEN AND CONTROL PASSES TO 2). IF POSITIVE OR ZERO, A GO RECOMMENDATION IS MADE AND CONTROL IS GIVEN TO 2).

*FOR GT-3, THE INCLINATION ANGLE TEST IS OMITTED AND AREA F IS ALWAYS USED.

2) THE SUBROUTINE C1DVGS IS THEN ENTERED TO COMPUTE AND OUTPUT THE NEGATIVE OF THE INCREMENT TO THE SPACECRAFT VELOCITY TO ACHIEVE A 1.5 REVOLUTION ORBIT LIFE-TIME, $-\Delta V_{G''}$ AND THE NEGATIVE OF THE TOTAL INCREMENT TO THE SPACECRAFT VELOCITY TO ACHIEVE OPTIMUM ORBITAL CONDITIONS, $-\Delta V_S$.

IF $\Delta V_{G'}$ IS NEGATIVE OR ZERO, THE GREENWICH MEAN TIME TO COMPLETE PROPULSION (GMTPC) IS SET TO ZERO. OTHERWISE, THIS TIME IS COMPUTED AS A FUNCTION OF THE VALID TIME OF THE ACTUAL VELOCITY, THE INCREMENT TO THE SPACECRAFT VELOCITY TO ACHIEVE A 1.5 REVOLUTION ORBIT LIFETIME, AND THE NOMINAL ACCELERATION DUE TO OAMS BURN FOR SPACECRAFT SEPARATION. AFTER THE RESULTS OF THIS COMPUTATION ARE PLACED IN THE OUTPUT TABLE, CONTROL PASSES TO C).

3) IF THE SOURCE OF THE VECTORS IS DEEMED UNACCEPTABLE FOR THE ABORT RECOMMENDATION, BOTH A GO AND A NO-GO RECOMMENDATION ARE SHOWN IMPLYING LACK OF ACCEPTABLE DATA. THE QUANTITIES COMPUTED BASED ON THIS RECOMMENDATION ARE SET TO ZERO AND CONTROL IS GIVEN TO C).

4) IF THE SOURCE OF THE VECTORS IS ACCEPTABLE FOR THE ABORT RECOMMENDATION, BUT EITHER THE FLIGHT PATH ANGLE OR THE EXPECTED VELOCITY FALLS OUTSIDE OF ITS LIMITS, A NO-GO RECOMMENDATION IS SHOWN. THE INCREMENT TO THE VELOCITY NEEDED TO ACHIEVE A 1.5 REVOLUTION ORBIT LIFETIME IS SET TO MAXIMUM

VALUE WHOSE SIGN INDICATES WHICH LIMIT, UPPER OR LOWER, WAS EXCEEDED AND THE TOTAL INCREMENT TO THE SPACECRAFT VELOCITY TO ACHIEVE OPTIMUM ORBITAL CONDITIONS IS SET TO ZERO. CONTROL IS THEN GIVEN TO C).

C) THE PROGRAM CELEVP IS ENTERED TO COMPUTE THE DISPLAY QUANTITIES ASSOCIATED WITH RAISING THE HEIGHT AT PERIGEE TO 75 NAUTICAL MILES. THE OUTPUT FROM CELEVP IS PLACED IN THE CFCFAV OUTPUT TABLE. SEE CCMAIN TABLE.

D) CFCFAV THEN PREPARES TO COMPUTE THE LATITUDE AND THE LONGITUDE OF IMPACT ASSOCIATED WITH THE MINIMUM AND THE MAXIMUM DELAY TIMES TO RETRO-FIRE.

1) THE MINIMUM DELAY TIME TO RETROFIRE IS BASED ON THE DIFFERENCE BETWEEN THE GREATER OF TWO GMT TIMES AND THE TIME ASSOCIATED WITH THE INPUT VECTORS. THE FIRST GMT TIME IS COMPUTED AS THE SUM OF THE NOMINAL TIME OF SECOND STAGE CUTOFF AND THE FACTOR N_1, IF EITHER THE VECTORS ARE FROM A SOURCE TRACKING THE BOOSTER OR SPACECRAFT THRUSTING HAS NOT YET BEGUN.

THE SECOND GMT TIME MAY HAVE ONE OF TWO VALUES. IF THE VECTORS ARE FROM A SOURCE TRACKING THE SPACECRAFT, SPACECRAFT THRUSTING HAS BEGUN, AND THE TIME OF THE END OF THRUSTING IS GREATER THAN THE TIME AT THE BEGINNING OF THRUSTING, ITS VALUE IS THE SUM OF THE FACTOR N_2 AND THE TIME AT THE END OF THRUSTING. IF THE VECTORS ARE FROM A SOURCE TRACKING THE SPACECRAFT, SPACECRAFT

THRUSTING HAS BEGUN, AND THE TIME IS THE BEGINNING OF THRUSTING IS GREATER THAN THE TIME AT THE END OF THRUSTING, ITS VALUE IS THE SUM OF THE FACTOR N_2 AND CURRENT TIME.

2) THE MAXIMUM DELAY TIME TO RETROFIRE IS DEFINED TO BE THE TIME OF ARRIVAL AT A SPECIFIED HEIGHT LESS A FACTOR INCORPORATING THE BURNING TIME OF THE RETROROCKETS AND THE TIME REQUIRED TO PLACE THE SPACECRAFT IN THE REENTRY ATTITUDE.

3) BASED ON THE SETTING OF THE LIFT SWITCH, CFCFAV SELECTS A BANK ANGLE WHICH SPECIFIES THE LIFT CONFIGURATION TO BE USED IN COMPUTING THE MINIMUM AND THE MAXIMUM DELAY IMPACT POINTS. WITH THIS INFORMATION, CFCFAV ENTERS THE SUBROUTINE CBIPRE FOR THE REQUIRED COMPUTATIONS. AN ERROR RETURN FROM CBIPRE SENDS CONTROL TO I)

D) ON A NORMAL RETURN, THE SUBROUTINE CRADES USES THE OUTPUT FROM CBIPRE TO SELECT THE SPACECRAFT RECOVERY AREA AND TO SET THE RECOVERY AREA INDICATOR. THIS INDICATOR MAY HAVE ONE OF THE FOLLOWING VALUES -

1) +0, IF THE LONGITUDE OF THE SELECTED RECOVERY AREA FALLS BETWEEN THE LONGITUDES OF MINIMUM AND MAXIMUM DELAY IMPACT.

2) -0, IF THE LONGITUDE OF THE SELECTED RECOVERY AREA LIES CLOSER TO THE LONGITUDE OF MINIMUM DELAY IMPACT.

3) -1, IF THE LONGITUDE OF THE SELECTED RECOVERY AREA LIES CLOSER TO THE LONGITUDE OF MAXIMUM DELAY IMPACT.

E) IF THE SECOND CASE OCCURS, THE TIME MINIMUM DELAY RETROFIRE BECOMES THE TIME IN GMT OF RETROFIRE, THE LONGITUDE OF MINIMUM DELAY IMPACT BECOMES THE LONGITUDE ASSOCIATED WITH THE TIME IN GMT OF RETROFIRE, AND THE LATITUDE OF MINIMUM DELAY IMPACT BECOMES THE LATITUDE ASSOCIATED WITH THE TIME IN GMT OF RETRFIRE. THE VECTORS AT THE END OF MINIMUM DELAY RETRO-FIRE THEN SERVE AS INPUT TO THE SUBROUTINE CVAFPA AND CONTROL PASSES TO H.

F) IF THE THIRD CASE OCCURS, THE STEPS OUTLINED ABOVE ARE EXECUTED USING THE QUANTITIES ASSO-CIATED WITH MAXIMUM DELAY IMPACT. THE VECTORS AT THE END OF MAXIMUM DELAY RETROFIRE SERVE AS INPUT TO CVAFPA AND CONTROL IS GIVEN TO H).

G) IF THE FIRST CASE OCCURS, CFCFAV ATTEMPTS TO REFINE THE TIME OF RETROFIRE TO LAND AT THE SELECTED RECOVERY AREA A MAXIMUM OF SEVEN TIMES. THE INITIAL INPUT TO THE SUBROUTINE C9TDRF IS THE TIME AND LONGITUDE OF MINIMUM DELAY RETRO-FIRE AND THE LONGITUDE OF THE SELECTED RECOVERY AREA. IF, ON THE FIRST ENTRY TO THE SUBROUTINE, THE CORRECTION TO THE TIME OF RETROFIRE IS LESS THAN .5 SECOND, THIS IMPLIES THAT THE MINIMUM DELAY RETROFIRE QUANTITIES ARE THE BEST CHOICES OF THE RETROFIRE GMT QUANTITIES AND THE PROCEDURE IN E) IS FOLLOWED.
OTHERWISE, THE CORRECTION IS ADDED TO THE LAST TIME OF RETROFIRE AND THE SUBROUTINE CGFTCI IS ENTERED TO OBTAIN THE LONGITUDE AND LATITUDE ASSOCIATED WITH THE NEW TIME. AN ERROR RETURN FROM CGFTCI SENDS CONTROL TO I). A NORMAL

RETURN CAUSES THE NEW LONGITUDE ASSOCIATED WITH THE TIME OF RETROFIRE AND THE ORIGINAL LONGITUDE OF THE SELECTED RECOVERY AREA TO BE INPUTTED TO C9TDRF AND THE PROCESS IS REPEATED. THE LAST COMPUTED VECTORS AT THE END OF RETRO-FIRE FROM CGFTCI THEN SERVE AS INPUT TO CVAFPA.

H) THE SUBROUTINE CVAFPA PRODUCES THE MAGNITUDE OF THE RELATIVE VELOCITY AND THE RELATIVE FLIGHT PATH ANGLE AT A GIVEN HEIGHT CRITERION USING THE SELECTED INPUT VECTORS. ON AN ERROR RETURN THESE QUANTITIES ARE SET TO ZERO IN THE OUTPUT TABLE. A NORMAL RETURN CAUSES THE RESULTS OF THE COMPUTATIONS TO BE STORED IN THE OUTPUT TABLE. THE REMAINING QUANTITIES - THE LONGI-TUDES AND LATITUDES ASSOCIATED WITHTHE MINIMUM DELAY RETROFIRE TIME AND THE TIME OF RETROFIRE TO LAND AT THE SELECTED RECOVERY AREA, THE TIME OF RETROFIRE TO LAND AT THE SELECTED RECOVERY AREA IN BOTH GMT AND ELAPSED TIME SINCE LIFTOFF, AND THE CODE OF THE SELECTED AREA IN OCTAL - ARE OUTPUTTED BEFORE CFCFAV RETURNS TO THE CONTROL PROGRAM.

I) AN ERROR RETURN FROM EITHER CBIPRE OR CGFTCI CAUSES ALL LATITUDES AND TIMES ASSOCIATED WITH IMPACT POINTS TO BE SET TO ZERO. THE CORRE-SPONDING LONGITUDES ARE SET TO PI RADIANS (THUS THROWING THEIR PLOTS OUT OF RANGE OF THE SCALE COVERED ON THE PLOTBOARD), AND THE CODE OF THE RECOVERY AREA IS SET DESIGNATING A FICITIOUS AREA. CONTROL IS THEN PASSED FROM CFCFAV TO THE MAIN PROGRAM, AFTER THE LOCATIONS IN THE OUTPUT TABLE CONTAINING THE RELATIVE VELOCITY AND THE

RELATIVE FLIGHT PATH ANGLE HAVE BEEN SET TO ZERO.

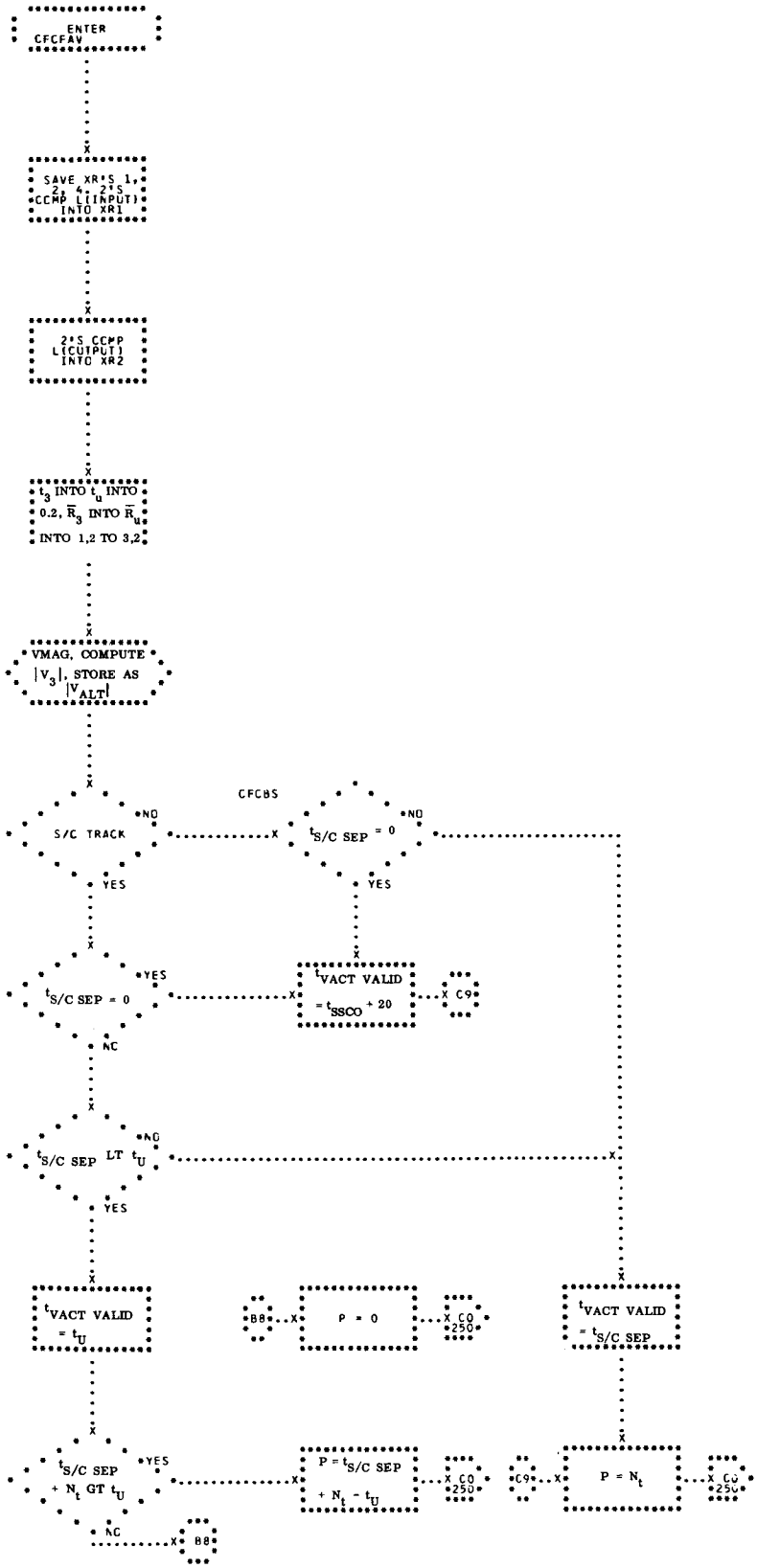
USAGE

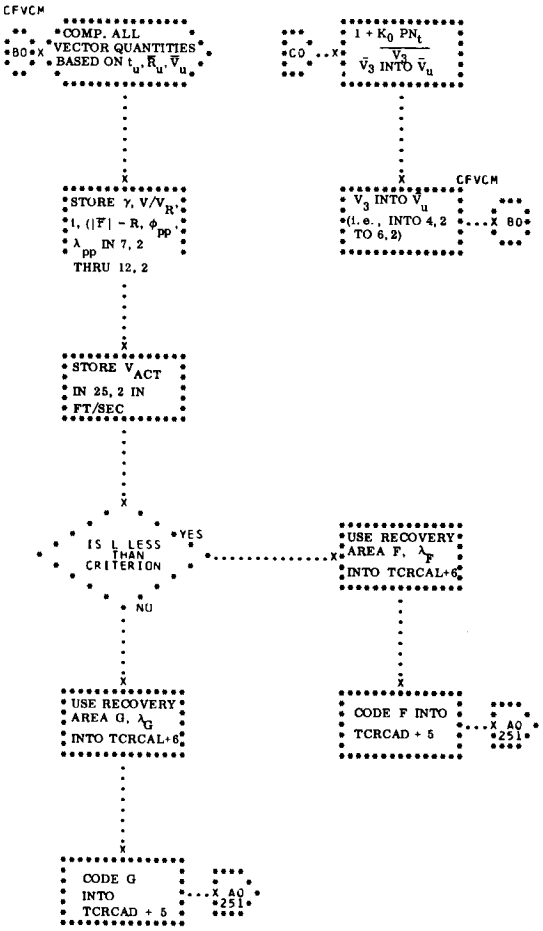
A) CALLING SEQUENCE -

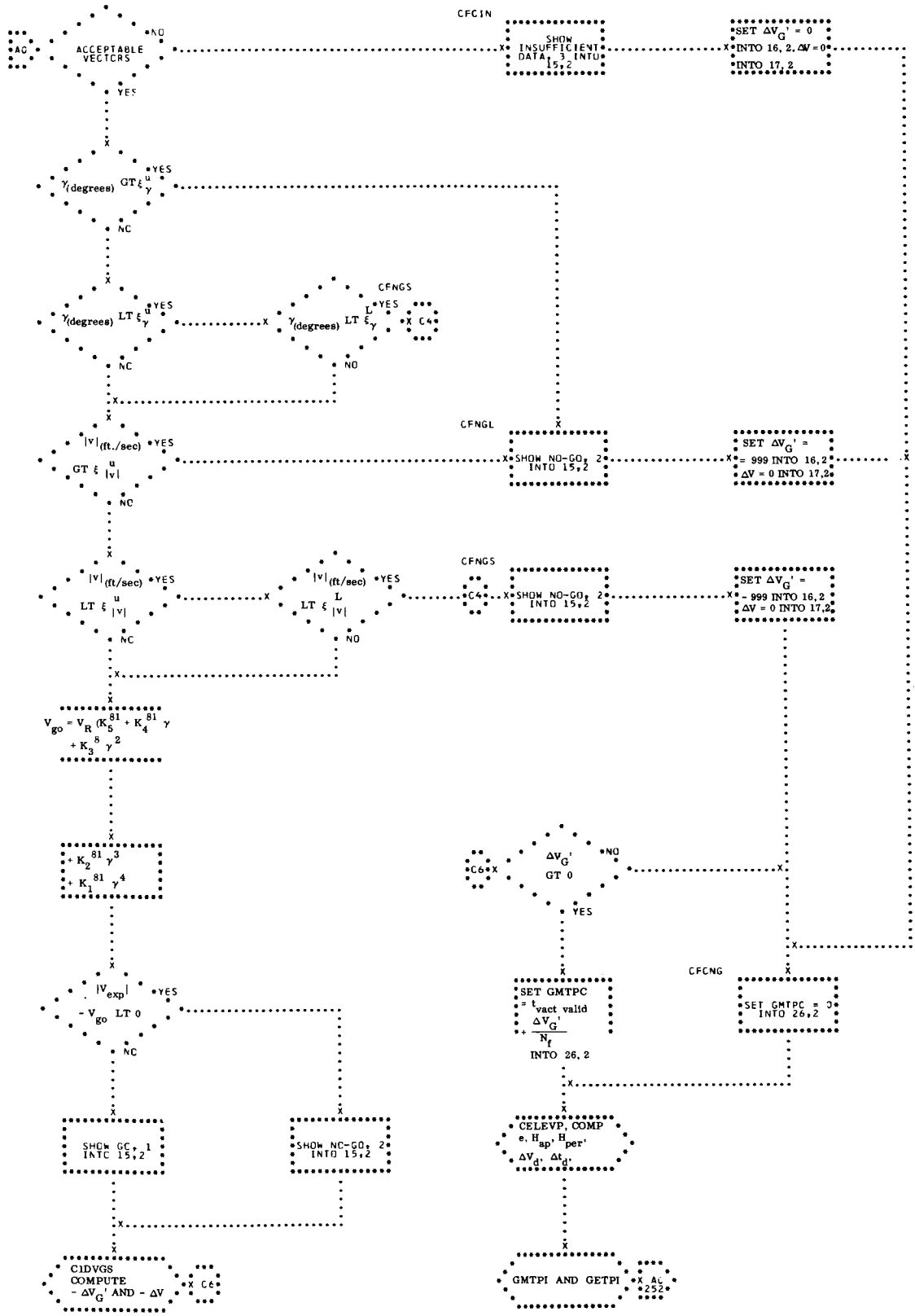
```
TSX       CFCFAV,4
PZE       L(INPUT),,L(OUTPUT)
RETURN
```

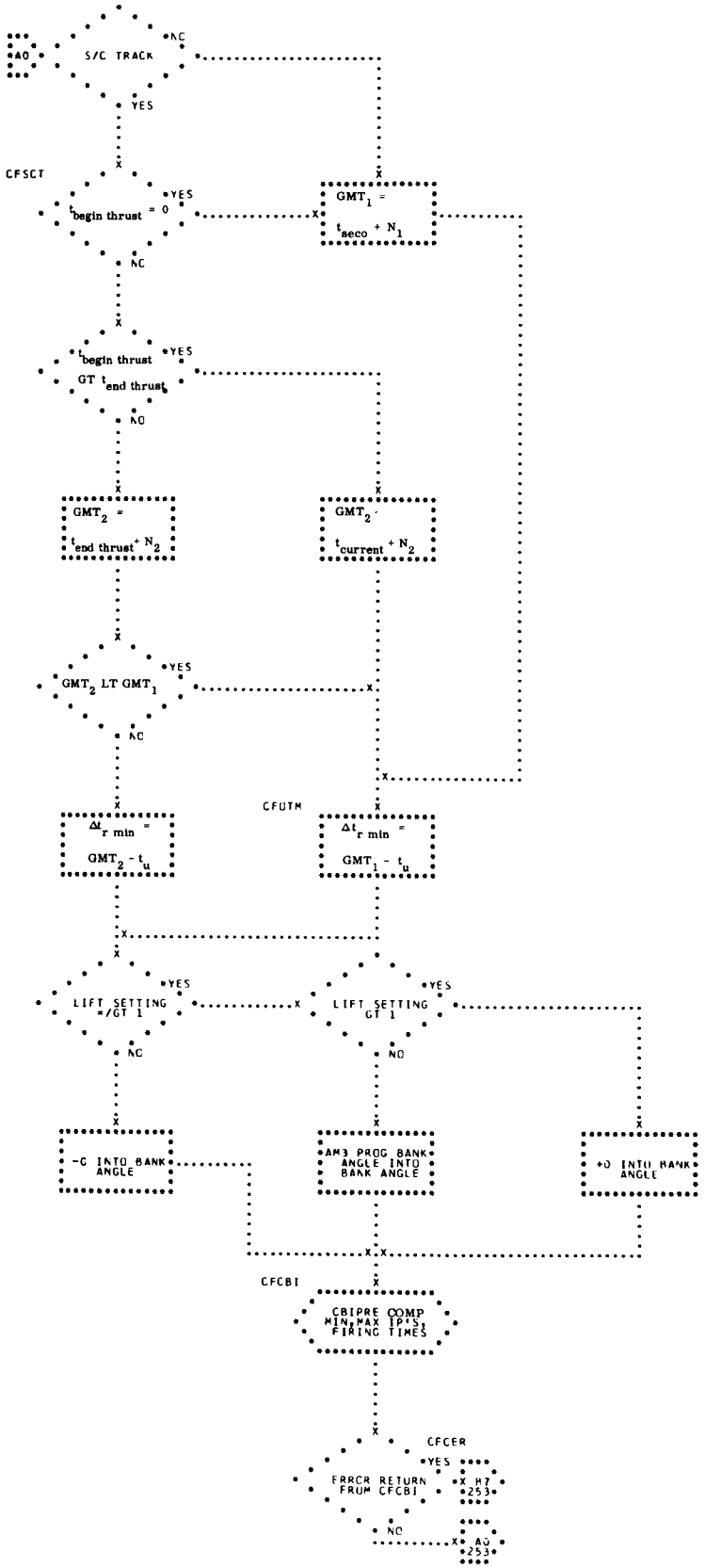
B) STORAGE REQUIRED - 321 LOCATIONS

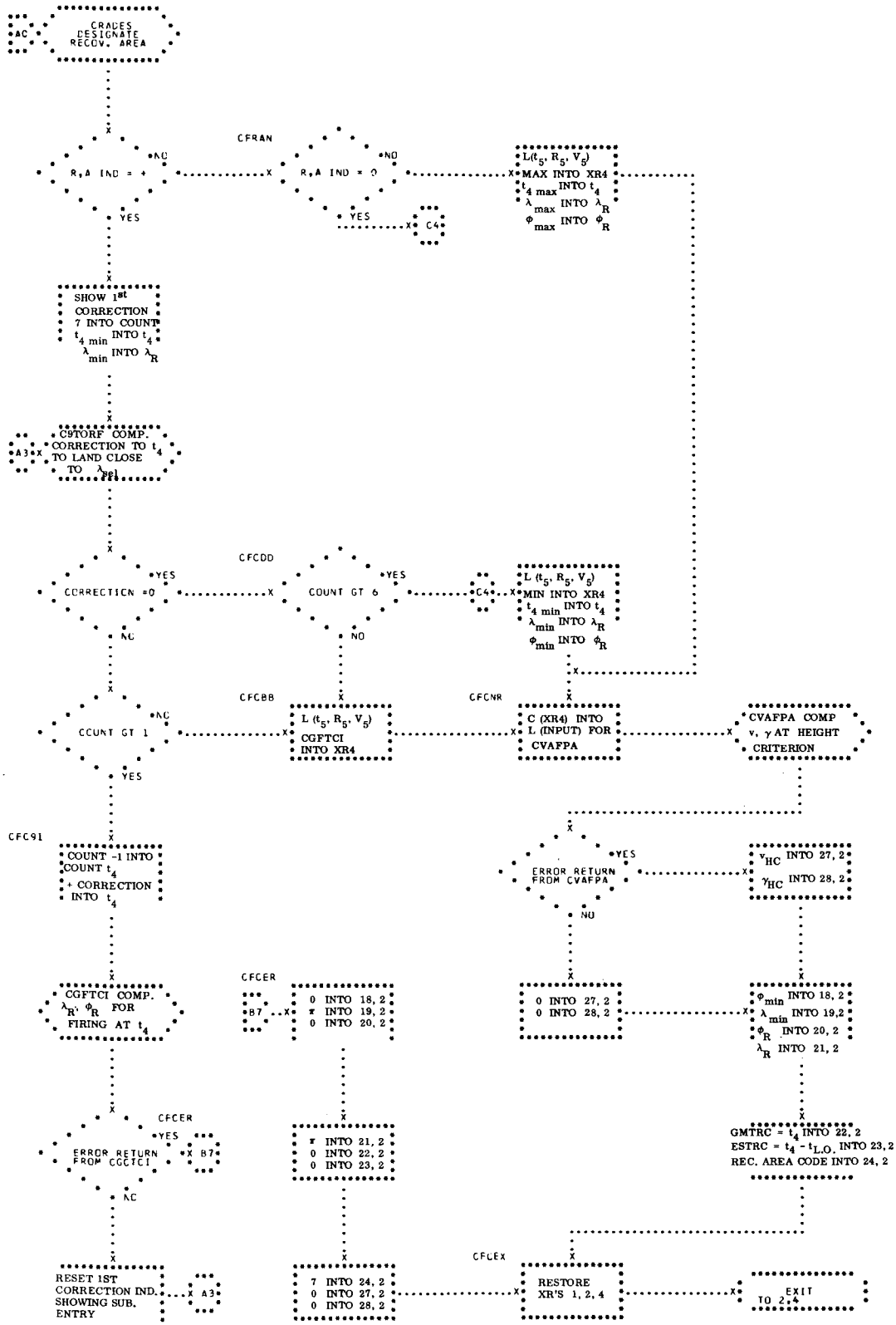
C) TIMING (7094) - 547 MILLISECONDS EXCLUDING TI REQUIRED FOR CVAFPA, CBIPRE, AND CGFTCI SUBROUTINES


ENTER CFCFAV
SAVE XR*S 1, 2* 4, 2*S CCMP L(INPUT) INTO XR1
2*S CCMP L(OUTPUT) INTO XR2
t_3 INTO t_u INTO 0.2, $\bar{R}_3$ INTO $\bar{R}_u$ INTO 1,2 TO 3,2
VMAG, COMPUTE $|V_3|$, STORE AS $|V_{ALT}|$
S/C TRACK
NO
YES
CFCBS
$t_{S/C\ SEP} = 0$
NO
YES
$t_{S/C\ SEP} = 0$
YES
NC
$t_{VACT\ VALID} = t_{SSCO} + 20$
C9
$t_{S/C\ SEP}$ LT t_U
NO
YES
$t_{VACT\ VALID} = t_U$
BB
P = 0
C0 250
$t_{VACT\ VALID} = t_{S/C\ SEP}$
$t_{S/C\ SEP} + N_t$ GT t_U
YES
NC
BB
$P = t_{S/C\ SEP} + N_t - t_U$
C0 250
C9
$P = N_t$
C0 250


CFVCM
COMP. ALL VECTOR QUANTITIES BASED ON t_u, $\bar{R}_u$, $\bar{V}_u$
B0
STORE γ, V/V_R, $|\bar{F}| - R$, ϕ_{pp}, λ_{pp} IN 7, 2 THRU 12, 2
STORE V_{ACT} IN 25, 2 IN FT/SEC
IS L LESS THAN CRITERION
YES
NO
USE RECOVERY AREA G, λ_G INTO TCRCAL+6
CODE G INTO TCRCAD + 5
A0 251
C0
$1 + K_0 PN_t$ / $\bar{V}_3$ INTO $\bar{V}_u$
$\bar{V}_3$ INTO $\bar{V}_u$ (i.e., INTO 4, 2 TO 6, 2)
CFVCM
B0
USE RECOVERY AREA F, λ_F INTO TCRCAL+6
CODE F INTO TCRCAD + 5
A0 251


AC
ACCEPTABLE VECTORS
NO
YES
CFCIN
SHOW INSUFFICIENT DATA, 3 INTO 15,2
SET $\Delta V_G' = 0$ INTO 16, 2, $\Delta V = 0$ INTO 17, 2
$\gamma_{(degrees)}$ GT ξ^u_γ
NC
$\gamma_{(degrees)}$ LT ξ^u_γ
CFNGS
$\gamma_{(degrees)}$ LT ξ^L_γ
C4
CFNGL
$|v|_{(ft./sec)}$ GT $\xi^u_{|v|}$
SHOW NO-GO, 2 INTO 15,2
SET $\Delta V_G' =$ - 999 INTO 16,2 $\Delta V = 0$ INTO 17,2
$|v|_{(ft/sec)}$ LT $\xi^u_{|v|}$
$|v|_{(ft/sec)}$ LT $\xi^L_{|v|}$
$V_{go} = V_R (K_5^{81} + K_4^{81} \gamma + K_3^8 \gamma^2$
$+ K_2^{81} \gamma^3$
$+ K_1^{81} \gamma^4$
C6
$\Delta V_G'$ GT 0
$|V_{exp}|$ - V_{go} LT 0
SET GMTPC = $t_{vact\ valid} + \frac{\Delta V_G'}{N_f}$ INTO 26,2
CFCNG
SET GMTPC = 0 INTO 26,2
SHOW GC, 1 INTO 15,2
SHOW NC-GO, 2 INTO 15,2
CELEVP, COMP e, H_{ap}, H_{per}, ΔV_d, Δt_d,
C1DVGS COMPUTE - $\Delta V_G'$ AND - ΔV
GMTPI AND GETPI
AC 252


S/C TRACK
NO
YES
CFSCT
begin thrust = 0
YES
NO
GMT1 = t seco + N1
begin thrust GT t end thrust
YES
NO
GMT2 = t end thrust + N2
GMT2 = t current + N2
GMT2 LT GMT1
YES
NO
CFDTM
Δt r min = GMT2 - tu
Δt r min = GMT1 - tu
LIFT SETTING =/GT 1
YES
NO
LIFT SETTING GT 1
YES
NO
-0 INTO BANK ANGLE
AM3 PROG BANK ANGLE INTO BANK ANGLE
+0 INTO BANK ANGLE
CFCBI
CBIPRE COMP MIN, MAX IPI'S, FIRING TIMES
CFCER
ERROR RETURN FROM CFCBI
YES
X H7 +253+
NO
X A9 +253+


AC
CRADES DESIGNATE RECOV. AREA
R,A IND = +
NO
YES
CFRAN
R,A IND = 0
NO
YES
C4
L(t5, R5, V5) MAX INTO XR4
t4 max INTO t4
λmax INTO λR
φmax INTO φR
SHOW 1st CORRECTION 7 INTO COUNT
t4 min INTO t4
λmin INTO λR
A3
CSTORF COMP. CORRECTION TO t4 TO LAND CLOSE TO λpad
CFCDD
CORRECTION =0
YES
NO
COUNT GT 6
YES
NO
C4
L (t5, R5, V5) MIN INTO XR4
t4 min INTO t4
λmin INTO λR
φmin INTO φR
CFCBB
COUNT GT 1
NO
YES
L (t5, R5, V5) CGFTCI INTO XR4
CFCNR
C (XR4) INTO L (INPUT) FOR CVAFPA
CVAFPA COMP v, γ AT HEIGHT CRITERION
CFC91
COUNT -1 INTO COUNT t4
+ CORRECTION INTO t4
ERROR RETURN FROM CVAFPA
YES
NO
vHC INTO 27, 2
γHC INTO 28, 2
CGFTCI COMP. λR, φR FOR FIRING AT t4
CFCER
B7
0 INTO 18, 2
π INTO 19, 2
0 INTO 20, 2
0 INTO 27, 2
0 INTO 28, 2
φmin INTO 18, 2
λmin INTO 19,2
φR INTO 20, 2
λR INTO 21, 2
ERROR RETURN FROM CGFTCI
CFCER
YES
NO
B7
π INTO 21, 2
0 INTO 22, 2
0 INTO 23, 2
GMTRC = t4 INTO 22, 2
ESTRC = t4 - tL.O. INTO 23, 2
REC. AREA CODE INTO 24, 2
RESET 1ST CORRECTION IND. SHOWING SUB. ENTRY
A3
7 INTO 24, 2
0 INTO 27, 2
0 INTO 28, 2
CFCEX
RESTORE XR'S 1, 2, 4
EXIT TO 2.4

36. C1DVGS - COMPUTE $-\Delta V_G'$ AND $-\Delta V$

GIVEN THE ACTUAL VELOCITY OF A SPACECRAFT, THE FLIGHT PATH ANGLE, THE HEIGHT OF THE SPACECRAFT ABOVE A SPHERICAL EARTH, AND THE MINIMUM VELOCITY NEEDED TO ACHIEVE ORBIT, C1DVGS WILL COMPUTE THE INCREMENT TO THE SPACECRAFT VELOCITY TO ACHIEVE A 1.5 REVOLUTION ORBIT LIFETIME, $\Delta V_{G'}$ AND THE TOTAL INCREMENT TO THE SPACECRAFT VELOCITY TO ACHIEVE OPTIMUM ORBITAL CONDITIONS, ΔV. C1DVGS IS USED DURING THE HOLD PHASE BY THE AVERAGE VECTORS PROGRAM, CFCFAV. $\Delta V_{G'}$ IS USED TO COMPUTE THE GREENWICH MEAN TIME TO COMPLETE PROPULSION, GMTPC.

INPUT

A) TABLES -

L(INPUT) THE INPUT TABLE TO C1DVGS IS THE THIRTEEN-LOCATION OUTPUT TABLE FROM CPRVCM. THE LOCATIONS OF THE TABLE USED BY C1DBGS ARE AS FOLLOWS -

L(INPUT)

+3 FLIGHT PATH ANGLE IN FLOATING POINT DEGREES, γ

•
•
•

+7 HEIGHT OF THE SPACECRAFT ABOVE A SPHERE IN FLOATING POINT NAUTICAL MILES, $(r - \bar{R})$

$L(V_{ACT})$ A ONE-LOCATION TABLE CONTAINING THE MAGNITUDE OF THE ACTUAL VELOCITY OF

THE SPACECRAFT IN FLOATING POINT FEET PER SECOND

$L(V_{GO})$ A ONE-LOCATION TABLE CONTAINING THE MINIMUM VELOCITY NEEDED TO ACHIEVE ORBIT IN FLOATING POINT FEET PER SECOND. (THIS QUANTITY IS COMPUTED BY CFCFAV AS A FUNCTION OF THE MAGNITUDE OF THE POSITION VECTOR AND THE FLIGHT PATH ANGLE SHORTLY AFTER SSCO.)

B) PARAMETER -

TCVINS NOMINAL INSERTION VELOCITY TO ACHIEVE OPTIMUM ORBITAL CONDITIONS IN FLOATING POINT FEET PER SECOND

OUTPUT

THE OUTPUT IS IN FLOATING POINT IN A TWO-LOCATION TABLE HAVING THE FOLLOWING FORMAT -

L(OUTPUT)-$\Delta V_{G'}$. THE NEGATIVE OF THE VELOCITY INCREMENT NEEDED TO ACHIEVE A 1.5 REVOLUTION ORBIT LIFETIME IN FLOATING POINT FEET PER SECOND

$-\Delta V$ THE NEGATIVE OF THE TOTAL VELOCITY INCREMENT NEEDED TO ACHIEVE OPTIMUM ORBITAL CONDITIONS IN FLOATING POINT FEET PER SECOND

METHOD

AFTER SAVING THE CONTENTS OF INDEX REGISTERS 1, 2, 3 AND 5 AND PICKING UP THE LOCATIONS OF THE INPUT AND OUTPUT TABLES, C1DVGS COMPUTES ΔV_G WHICH IS USED TO COMPUTE $\Delta V_{G'}$.

V_G IS COMPUTED AS FOLLOWS -

1) $a_i = b_{i1} + b_{i2}H + b_{i3}H^2 \quad i = 1, 2, \ldots, 9$

WHERE

b_{i3} , b_{i2} , b_{i3} ARE PROGRAM CONSTANTS AND H IS THE HEIGHT OF THE SPACECRAFT ABOVE A SPHERICAL EARTH IN FLOATING POINT NAUTICAL MILES.

2) $V_1 = a_1 + a_2\gamma + a_3\gamma^2$

$V_2 = a_4 + a_5\gamma + a_6\gamma^2$

$V_3 = a_7 + a_8\gamma + a_9\gamma^2$

WHERE

γ IS THE FLIGHT PATH ANGLE IN FLOATING POINT DEGREES

3) ΔV_G IS NOW DETERMINED BY INTERPOLATION - IF $V_{ACT} \geq V_2$, THEN $\Delta V_G = \Delta V_2 + (V_{ACT} - V_2) \frac{\Delta V_3 - \Delta V_2}{V_3 - V_2}$

IF $V_{ACT} < V_2$, THEN $\Delta V_G = \Delta V_1 + (V_{ACT} - V_1) \frac{\Delta V_2 - \Delta V_1}{V_2 - V_1}$

WHERE

ΔV_1, ΔV_2 AND ΔV_3 ARE PROGRAM CONSTANTS.

4) $\Delta V_{G'}$ IS COMPUTED BY ADDING A VELOCITY INCREMENT FOR OAMS SEPARATION, ΔV_{OS}, TO ΔV_G -

$$\Delta V_{G'} = \Delta V_G + \Delta V_{OS}$$

ΔV, THE VELOCITY INCREMENT NEEDED TO ACHIEVE OPTIMUM ORBITAL CONDITIONS IS COMPUTED AS FOLLOWS -

1) IF THE ACTUAL SPACECRAFT VELOCITY IS GREATER THAN THE NOMINAL INSERTION VELOCITY TO ACHIEVE OPTIMUM ORBITAL CONDITIONS, TCVINS, THEN -

$$\Delta V = V_{INS} - V_{ACT}$$

2) IF THE ACTUAL SPACECRAFT VELOCITY IS LESS THAN OR EQUAL TO TCVINS, A TEST IS MADE ON $\Delta V_{G'}$.

A) IF IT IS NEGATIVE, MEANING THAT THE ACTUAL VELOCITY IS SUCH THAT A 1.5 ORBITS MAY BE ACHIEVED WITHOUT MODIFICATION OF THE SPACECRAFT VELOCITY -

$$\Delta V = V_{INS} - V_{ACT}$$

B) IF $\Delta V_{G'}$ IS POSITIVE OR ZERO, ΔV IS COMPUTED BY SUBTRACTING V_{GO}, THE MINIMUM VELOCITY NEEDED TO ACHIEVE ORBIT FROM TCVINS, AND ADDING THE VELOCITY INCREMENT NEEDED TO ACHIEVE A 1.5 ORBIT REVOLUTION LIFETIME, $\Delta V_{G'}$ -

$$\Delta V = V_{INS} - V_{GO} + V_{G'}$$

THE NEGATIVES OF ΔV_G AND ΔV ARE PLACED IN THE OUT- IS MADE FROM THE PROGRAM.

THE VALUES OF THE PROGRAM CONSTANTS USED ARE AS FOLLOWS -

b_{11} = .26189446E5	b_{12} =-.12190013E2	b_{13} = .28135904E-1
b_{21} = .45033371E3	b_{22} =-.14732181E2	b_{23} = .94591081E-1
b_{31} = .47257040E3	b_{32} =-.25337494E1	b_{33} =-.78762051E-2
b_{41} = .26420661E5	b_{42} =-.15140652E2	b_{43} = .48815426E-1
b_{51} =-.70969714E3	b_{52} = .13995599E2	b_{53} =-.71819475E-1
b_{61} = .33811703E4	b_{62} =-.65284140E2	b_{63} = .33140294
b_{71} = .26914478E5	b_{72} =-.22960395E2	b_{73} = .92525252E-1
b_{81} = .12067728E4	b_{82} =-.28086933E2	b_{83} = .16171096
b_{91} = .13988601E4	b_{92} =-.19183242E2	b_{93} = .67145029E-1

$\Delta V_1 = 300.$ and $\Delta V_{OS} = 5.$

$\Delta V_2 = 150.$

$\Delta V_3 = 0.$

FOR ADDITIONAL INFORMATION, SEE UNITED STATES GOVERNMENT MEMORANDUM, ''METHOD FOR COMPUTING THE ΔV REQUIRED BY THE GEMINI SPACECRAFT AFTER LAUNCH,'' YORK AND PHILLIPS, JULY 7, 1964, REFERENCE - OM(LAP/WY - SVM/ 3301.

USAGE

A) CALLING SEQUENCE -

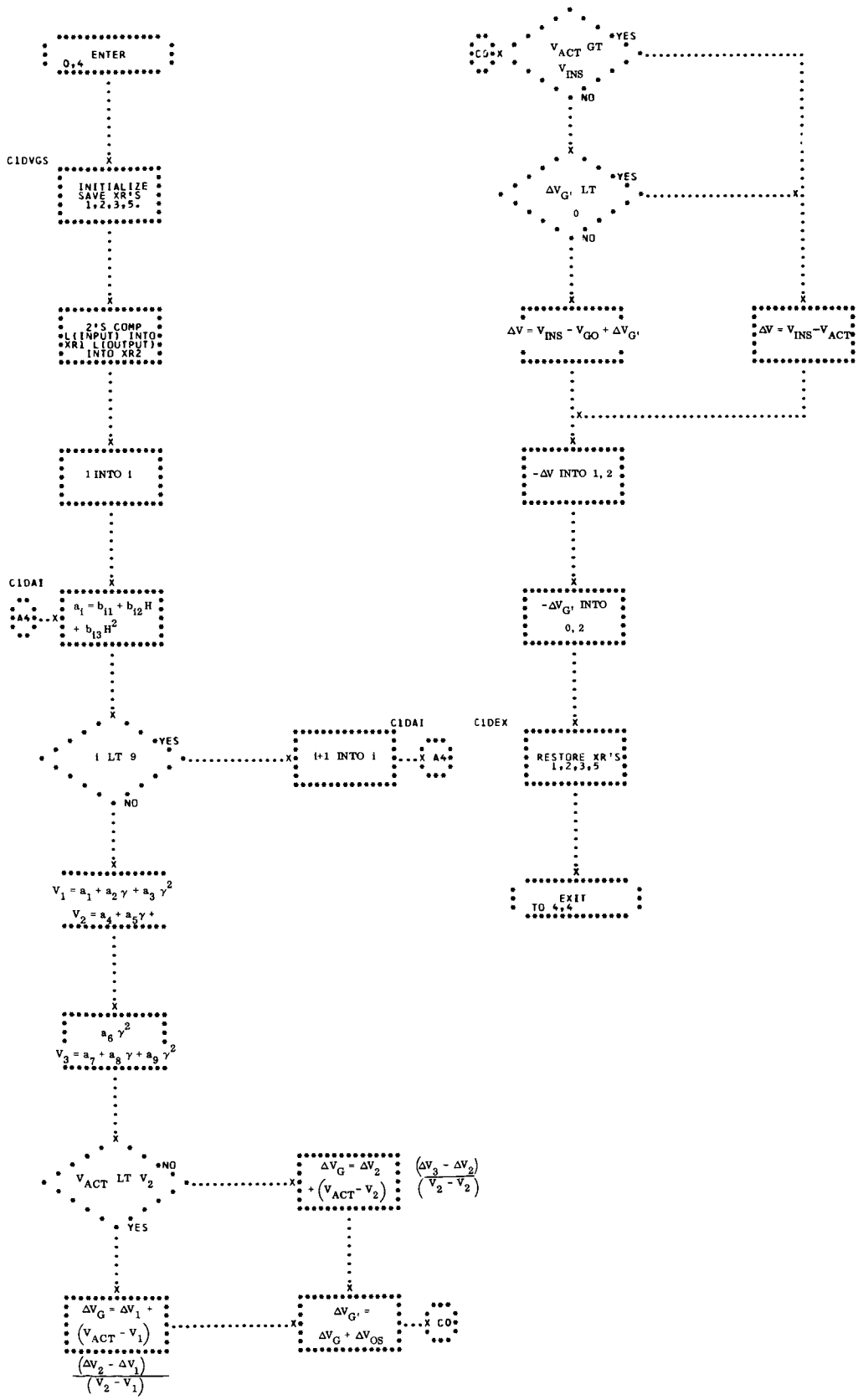
```
TSX     C1DVGS,4
PZE     L(INPUT),,L(OUTPUT)
PZE     L(V   )
PZE     L(V   )
RETURN
```

B) STORAGE REQUIRED -

107 DECIMAL LOCATIONS

C) TIMING -

1.201 MILLISECONDS


ENTER
0,4
C1DVGS
INITIALIZE
SAVE XR'S
1,2,3,5.
2'S COMP
L(INPUT) INTO
XR1 L(OUTPUT)
INTO XR2
1 INTO I
C1DAI
A4
$a_i = b_{i1} + b_{i2} H + b_{i3} H^2$
i LT 9
YES
NO
C1DAI
i+1 INTO i
A4
$V_1 = a_1 + a_2 \gamma + a_3 \gamma^2$
$V_2 = a_4 + a_5 \gamma +$
$a_6 \gamma^2$
$V_3 = a_7 + a_8 \gamma + a_9 \gamma^2$
V_{ACT} LT V_2
NO
YES
$\Delta V_G = \Delta V_1 + (V_{ACT} - V_1) \frac{(\Delta V_2 - \Delta V_1)}{(V_2 - V_1)}$
$\Delta V_G = \Delta V_2 + (V_{ACT} - V_2) \frac{(\Delta V_3 - \Delta V_2)}{(V_2 - V_2)}$
$\Delta V_{G'} = \Delta V_G + \Delta V_{OS}$
C0
C0
V_{ACT} GT V_{INS}
YES
NO
$\Delta V_{G'}$ LT 0
YES
NO
$\Delta V = V_{INS} - V_{GO} + \Delta V_{G'}$
$\Delta V = V_{INS} - V_{ACT}$
-ΔV INTO 1, 2
$-\Delta V_{G'}$ INTO 0, 2
C1DEX
RESTORE XR'S
1,2,3,5
EXIT
TO 4,4

37. CELEVP - ELEVATE PERIGEE

GIVEN A $t, \bar{r}, \bar{v}$ DEFINING THE TRAJECTORY OF A SPACECRAFT, CELEVP DETERMINES THE HEIGHT AT PERIGEE AND COMPUTES THE DISPLAY QUANTITIES TO BE USED IN RAISING THE HEIGHT AT PERIGEE TO SEVENTY-FIVE NAUTICAL MILES. CELEVP IS USED DURING THE HOLD PHASE BY THE AVERAGE VECTORS PROGRAM, CFCFAV. (SEE FLOWCHART.)

INPUT

A) THE INPUT $t, \bar{x}, \bar{v}$, IS IN FLOATING POINT IN A SEVEN-LOCATION TABLE FORMATTED AS FOLLOWS -

L(INPUT VECTORS)		
	t	TIME IN GMT SECONDS
+1	x	$\bar{r}$, POSITION VECTOR IN GEMINI LENGTH UNITS
+2	y	
+3	z	
+4	$\dot{x}$	v, VELOCITY VECTOR IN GEMINI VELOCITY UNITS
+5	$\dot{y}$	
+6	$\dot{z}$	

B) CONSTANTS -

KM00.5	DEC - .5	FLOATING POINT MINUS ONE-HALF
K001.0	DEC + 1.0	FLOATING POINT ONE
K002.0	DEC + 2.0	FLOATING POINT TWO

K8U.NM	DEC + .21777423E-1	SEVENTY-FIVE NAUTICAL MILES IN GEMINI LENGTH UNITS, FLOATING POINT
KFTSCG	DEC + .25936272E5	CONVERTS GEMINI VELOCITY UNITS TO FEET/SECOND, FLOATING POINT
K.ERGU	DEC + .99924461	RADIUS OF THE SPHERICAL EARTH IN GEMINI LENGTH UNITS, FLOATING POINT
KCGUNM	DEC + .34439336E4	CONVERTS GEMINI LENGTH UNITS TO NAUTICAL MILES, FLOATING POINT

C) PARAMETERS -

TC.NTF	NOMINAL OAMS ACCELERATION IN FEET/SECOND^2

D) COMMUNICATION CELL -

MCLFTM	CONTAINS TIME OF LIFT-OFF IN FLOATING POINT SECONDS, GMT

E) PROGRAM -

CEAPRV	PROGRAM TO COMPUTE THE VECTORS AT APOGEE OR PERIGEE

F) SUBROUTINES -

U3VMAG	FLOATING POINT VECTOR MAGNITUDE ROUTINE
U1SQRT	FLOATING POINT SQUARE ROOT ROUTINE
U3DOTP	FLOATING POINT VECTOR DOT PRODUCT

ROUTINE

OUTPUT

THE OUTPUT IS IN FLOATING POINT IN A SEVEN-LOCATION TABLE HAVING THE FOLLOWING FORMAT -

L(OUTPUT)		
	e	ECCENTRICITY
+1	H_{apogee}	KEPLERIAN HEIGHT AT APOGEE IN FLOATING POINT NAUTICAL MILES
+2	$H_{perigee}$	KEPLERIAN HEIGHT AT PERIGEE IN FLOATING POINT NAUTICAL MILES
+3	ΔV_d	VELOCITY INCREMENT TO BE APPLIED AT APOGEE TO RAISE THE FOLLOWING PERIGEE TO SEVENTY-FIVE NAUTICAL MILES, FLOATING POINT FEET/SECOND
+4	Δt_d	TIME DURATION OF NECESSARY OAMS BURN TO ACHIEVE ΔV_d FLOATING POINT SECONDS
+5	GMTPI	GMT AT WHICH TO INITIATE BURN, FLOATING POINT SECONDS
+6	GETPI	ELAPSED TIME FROM LIFT-OFF TO TIME TO INITIATE BURN, FLOATING POINT SECONDS

METHOD

THE METHOD USED IS BASICALLY AS FOLLOWS -

A) THE INPUT $t, \bar{r}, \bar{v}$ ARE USED AS INPUT TO THE PROGRAM CEAPRV TO COMPUTE THE VECTORS AT APOGEE, $t_a, \bar{r}_a, \bar{v}_a$.

B) USING THE VECTORS AT APOGEE, THE ECCENTRICITY AND THE HEIGHTS OF APOGEE AND PERIGEE WHICH ARE

TO BE PLACED IN THE OUTPUT TABLE ARE DETERMINED. IT IS TO BE NOTED HERE, THAT THE COMPUTED HEIGHTS OF APOGEE AND PERIGEE ARE ESTIMATED HEIGHTS, ABOVE THE GEMINI SPHEROID, BASED ON KEPLERIAN EQUATIONS.
THEY DO NOT REFLECT THE DRAG FORCE AND THE OBLATENESS OF THE EARTH WHICH TEND TO DECAY THE ORBIT. THUS THESE ESTIMATED HEIGHTS ARE, IN GENERAL, LARGER THAN THE TRUE HEIGHTS.

$$e \cos E = r_a v_a^2 - 1 \qquad 1$$

$$a = \frac{r_a}{1 - e \cos E} \qquad 2$$

$$(e \sin E)^2 = \frac{(\bar{r}_a \cdot \bar{v}_a)^2}{a} \qquad 1$$

$$e = (e \sin E)^2 + (e \cos E)^2$$

$$H_{apogee} = K_1 [a(1 + e) - R_{spherical\ earth}] \qquad 3$$

$$H_{perigee} = K_1 [a(1 - e) - R_{spherical\ earth}] \qquad 3$$

WHERE
K_1 IS THE FACTOR TO CONVERT GEMINI LENGTH UNITS TO NAUTICAL MILES, KCGUNM
THE ECCENTRICITY AND THE KEPLERIAN HEIGHTS AT APOGEE AND PERIGEE ARE STORED IN THE OUTPUT TABLE.

C) THE KEPLERIAN HEIGHT AT PERIGEE IS COMPARED AGAINST SEVENTY-FIVE NAUTICAL MILES.

1) IF THE KEPLERIAN HEIGHT AT PERIGEE IS LESS THAN SEVENTY-FIVE NAUTICAL MILES, A HEIGHT INCREMENT IS COMPUTED USING -

$$\Delta H = 75 \text{ n. mi.} - H_{perigee}$$

THE VELOCITY WHICH ONE MUST HAVE AT APOGEE IN ORDER THAT PERIGEE WILL BE SEVENTY-FIVE NAUTICAL MILES IS COMPUTED USING -

$$V_d = \sqrt{\frac{2}{r_a} - \frac{1}{(a + \frac{\Delta H}{2})}}$$

A VELOCITY INCREMENT TO BE APPLIED INSTANTANEOUSLY AT APOGEE IS COMPUTED USING -

$$\Delta V_d = V_d - V_a$$

THE APOGEE VELOCITY IS INCREMENTED USING -

$$\bar{V}_a' = (1 + \frac{\Delta V_d}{V_a})\bar{V}_a$$

2) IF THE KEPLERIAN HEIGHT AT PERIGEE IS GREATER THAN OR EQUAL TO SEVENTY-FIVE NAUTICAL MILES, THE APOGEE VELOCITY IS NOT MODIFIED.

D) NOW, THE PROGRAM CEAPRV IS ENTERED TO FIND THE VECTORS AT PERIGEE $t_p, \bar{r}_p, \bar{v}_p$ AND a' (SEMI-MAJOR AXIS), USING THE VECTORS AT APOGEE AS INPUT. IF THE ORIGINAL APOGEE VELOCITY HAS BEEN MODIFIED, IT WILL BE POSSIBLE TO TELL WHETHER THE VELOCITY INCREMENT COMPUTED AND APPLIED INSTANTANEOUSLY TO SEVENTY-FIVE NAUTICAL MILES. IF THE APOGEE VELOCITY HAS NOT BEEN MODIFIED, IT WILL BE

POSSIBLE TO DETERMINE THE ACCURACY OF THE ESTIMATED PERIGEE HEIGHT. DUE TO THE FACT THAT THE PERIGEE HEIGHT WAS AN ESTIMATE, THE VELOCITY INCREMENT APPLIED WILL NOT HAVE BEEN SUFFICIENT, IN GENERAL, TO RAISE THE TRUE HEIGHT OF PERIGEE TO SEVENTY-FIVE NAUTICAL MILES.

E) THE HEIGHT AT PERIGEE IS DETERMINED USING -

$$H'_{\text{perigee}} = r_p - R_{\text{spherical earth}}$$

THIS NEW HEIGHT IS SUBTRACTED FROM SEVENTY-FIVE NAUTICAL MILES TO OBTAIN A $\Delta H'$. IF $\Delta H'$ IS POSITIVE, THE CORRECTION APPLIED TO THE APOGEE VELOCITY WAS NOT SUFFICIENT TO RAISE THE TRUE HEIGHT AT PERIGEE TO SEVENTY-FIVE NAUTICAL MILES. IF $\Delta H'$ IS NEGATIVE, EITHER THE TRUE HEIGHT AT PERIGEE WILL BE GREATER THAN SEVENTY-FIVE NAUTICAL MILES OR THE CORRECTION APPLIED TO THE APOGEE VELOCITY WAS TOO LARGE. IN ANY CASE, FURTHER COMPUTATIONS ARE MADE TO OBTAIN A NEW ESTIMATE OF THE VELOCITY DESIRED AT APOGEE -

$$V_d = \sqrt{\frac{2}{r_a} - \frac{1}{a' + \frac{\Delta H'}{2}}}$$

WHERE

a' HAS BEEN COMPUTED FROM THE APOGEE VECTOR USED AS INPUT TO CEAPRV IN FINDING THE PERIGEE VECTORS, AND r_a IS THE MAGNITUDE OF THE APOGEE POSITION VECTOR. ΔV_d IS COMPUTED USING -

$$\Delta V_d = V_d - V_a$$

A TEST IS MADE ON THE SIGN OF ΔV_d -

1) IF IT IS NEGATIVE, IT IS KNOWN THAT THE TRUE HEIGHT AT PERIGEE WILL BE GREATER THAN SEVENTY-FIVE NAUTICAL MILES WITHOUT MODIFICATION OF THE APOGEE VELOCITY AND NO FURTHER COMPUTATIONS NEED BE MADE. THE REMAINING LOCATIONS IN THE OUTPUT TABLE ARE SET TO ZERO, AND AN EXIT IS MADE FROM THE PROGRAM.
2) IF ΔV_d IS POSITIVE, CCOMPUTATIONS ARE CONTINUED AS DESCRIBED BELOW.

F) DURING AN ACTUAL MISSION, THE SPACECRAFT VELOCITY CANNOT BE INCREMENTED INSTANTANEOUSLY, BUT MUST BE INCREASED GRADUALLY BY FIRING THE OAMS SYSTEM. IT IS NECESSARY, THEREFORE, TO COMPUTE Δt_d , THE TIME PERIOD OVER WHICH OAMS THRUST MUST BE APPLIED IN ORDER TO INCREMENT THE ORIGINAL APOGEE VECTOR BY ΔV_d. THE FOLLOWING EQUATION IS USED -

$$\Delta t_d = \frac{\Delta V_d}{\text{OAMS acceleration}}$$

SINCE IT IS ACTUALLY DESIRED THAT THE VELOCITY BE INCREMENTED INSTANTANEOUSLY AT APOGEE, THE BEST ALTERNATIVE SEEMS TO BE TO FIRE THE OAMS SYSTEM FOR THE SAME AMOUNT OF TIME IMMEDIATELY PRECEDING AND FOLLOWING APOGEE SO THAT THE SPACECRAFT VELOCITY WILL BE INCREMENTED EVENLY ON EITHER SIDE OF APOGEE. THE GMT AT WHICH TO INITIATE OAMS BURN IS COMPUTED USING -

$$\text{GMTPI} = t_a - \frac{\Delta t_d}{2}$$

THE TIME TO INITIATE BURN IN ELAPSED TIME SINCE LIFT-OFF IS COMPUTED USING -

$$GETPI = GMTPI - t_{lift\ off}$$

$\Delta V_d, \Delta t_d$, GMTPI AND GETPI ARE PLACED IN THE OUTPUT TABLE AND AN EXIT IS MADE FROM THE PROGRAM.

G) IF THE GMTPI COMPUTED IS LESS THAN CURRENT TIME (THE TIME TAG ASSOCIATED WITH THE INPUT VECTORS), IT WILL NOT BE POSSIBLE TO INITIATE OAMS BURN AT THE GMTPI RECOMMENDED. IN THIS CASE, GMTPI AND GETPI ARE SET TO ZERO, INDEX REGISTERS ARE RESTORED AND AN EXIT IS MADE FROM THE PROGRAM.

H) IT IS POSSIBLE THAT ECCENTRICITY WILL BE CLOSE TO ZERO, IN WHICH CASE APOGEE AND PERIGEE ARE POORLY DEFINED. IF APOGEE IS POORLY DEFINED, AN ERROR RETURN WILL RESULT FROM THE FIRST ENTRY TO CEAPRV. IN THIS CASE, SINCE IT IS KNOWN THAT THE TRUE ORBIT IS NEARLY CIRCULAR, THE INPUT VECTORS ARE USED TO COMPUTE THE ECCENTRICITY AND TRUE KEPLERIAN HEIGHTS OF APOGEE AND PERIGEE BY THE FOLLOWING -

$$e \cos E = rv^2 - 1 \qquad (1)$$

$$a = \frac{r}{1 - e \cos E} \qquad (2)$$

$$(e \sin E)^2 = \frac{(\bar{r} \cdot \bar{v})^2}{a} \qquad (3)$$

[1] Fundamentals of Celestial Mechanics, J. M. A. Danby, The Macmillian Com ny New York, 1962, p. 162.

[2] ibid., p. 125

[3] ibid., p. 124

$$e = (e \sin E)^2 + (e \cos E)^2$$

$$H_{apogee} = K_1[a(1 + e) - R_{spherical\ earth}]$$

$$H_{perigee} = K_1[a(1 - e) - R_{spherical\ earth}]$$

THE ECCENTRICITY AND THE HEIGHTS OF APOGEE AND PERIGEE ARE STORED IN THE OUTPUT TABLE. SINCE THE APOGEE VECTORS CANNOT BE DETERMINED $\Delta V_d, \Delta t_d$, GMTPI AND GETPI CANNOT BE COMPUTED. THE LOCATIONS IN THE OUTPUT TABLE WHICH THEY OCCUPY ARE SET TO ZERO AND AN EXIT IS MADE FROM THE PROGRAM.

ON THE OTHER HAND, THE CASE MAY ARISE WHERE THE ORBIT IS SUFFICIENTLY ELLIPTICAL SO THAT AN APOGEE VECTOR MAY BE FOUND, BUT THE HEIGHT AT APOGEE MAY BE FOUND TO BE ONLY SLIGHTLY GREATER THAN SEVENTY-FIVE NAUTICAL MILES. IN THIS CASE, WHEN THE APOGEE VELOCITY IS INCREMENTED, A NEW ORBIT IS DEFINED WHOSE ECCENTRICITY WILL, IN GENERAL, BE CLOSE TO ZERO SINCE THE INCREMENTED VELOCITY WILL RAISE THE HEIGHT OF PERIGEE TO NEARLY SEVENTY-FIVE MILES. IN THIS CASE, THERE WILL BE AN ERROR RETURN FROM THE SECOND ENTRY TO CEAPRV BECAUSE IT WILL NOT BE POSSIBLE TO FIND THE VECTORS AT PERIGEE. IF THIS HAPPENS, THE LOCATIONS OCCUPIED IN THE OUTPUT TABLE BY $\Delta V_d, \Delta t_d$, GMTPI AND GETPI ARE SET TO ZERO AND AN EXIT IS MADE FROM THE PROGRAM.

I) IF THE GMT OF THE INPUT VECTORS IS GREATER THAN THE GMT OF THE VECTORS AT APOGEE, NORMAL RETURN 1 WILL RESULT FROM THE FIRST ENTRY TO CEAPRV. IN THIS CASE, IT IS TOO LATE TO MAKE A RECOMMENDATION TO FIRE THE OAMS SYSTEM BEFORE ARRIVING AT APOGEE. THE ENTIRE OUTPUT TABLE IS SET TO ZEROS, AND AN EXIT IS MADE FROM THE PROGRAM.

USAGE

A) CALLING SEQUENCE -

```
        TSX     CELEVP,4
        PZE     L(INPUT t, r̄, v̄),,L(OUTPUT)
        RETURN
```

B STORAGE REQUIRED - 172 DECIMAL LOCATIONS

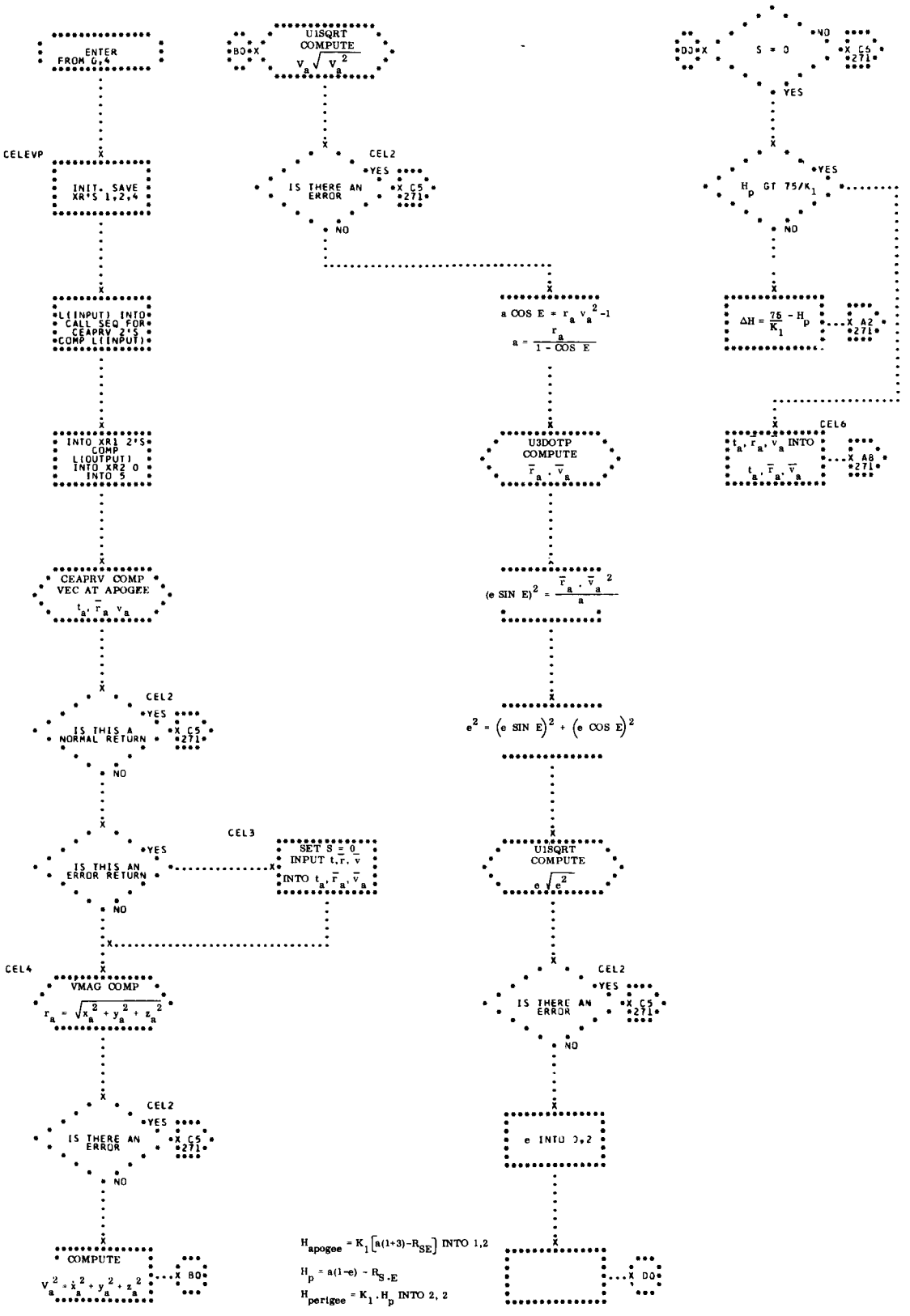
C) TIMING - 536 MILLISECONDS, MAXIMUM (INCLUDES ALL SUBROUTINES)

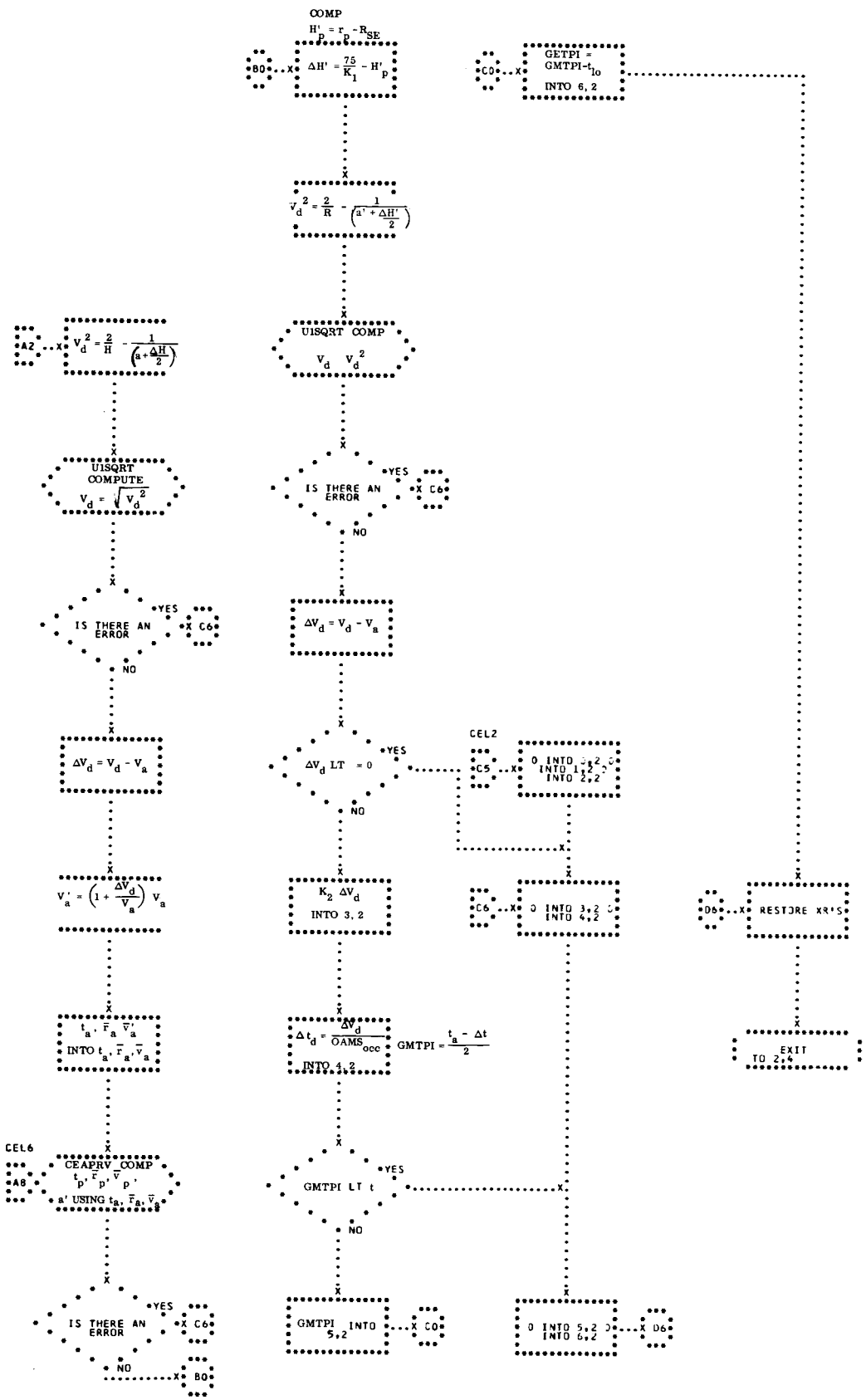
ACCURACY

IT WAS FOUND THAT -

A) IF COMPUTED H_p (KEPLERIAN) $\geq$ 0, APPLYING THE NOMINAL OAMS ACCELERATION FOR Δt_d SECONDS BEGINNING AT GMTPI RESULTED IN RAISING THE HEIGHT OF PERIGEE TO WITHIN $\pm$ .1 N.MI. OF 75 N. MI.

B) IF COMPUTED H_p (KEPLERIAN) < 0, APPLYING THE NOMINAL OAMS ACCELERATION FOR Δt_d SECONDS BEGINNING AT GMTPI RESULTED IN RAISING THE HEIGHT OF PERIGEE TO WITH $\pm$ .7 N.MI. OF 75 N.MI.


ENTER FROM G,4
CELEVP
INIT. SAVE XR'S 1,2,4
L(INPUT) INTO CALL SEQ FOR CEAPRV 2'S COMP L(INPUT)
INTO XR1 2'S COMP L(OUTPUT) INTO XR2 0 INTO 5
CEAPRV COMP VEC AT APOGEE
$t_a, \bar{r}_a, \bar{v}_a$
CEL2
IS THIS A NORMAL RETURN
YES
X C5 271
NO
CEL3
IS THIS AN ERROR RETURN
SET S = 0 INPUT t,r, v INTO $t_a, \bar{r}_a, \bar{v}_a$
CEL4
VMAG COMP
$r_a = \sqrt{x_a^2 + y_a^2 + z_a^2}$
IS THERE AN ERROR
COMPUTE
$V_a^2 = \dot{x}_a^2 + \dot{y}_a^2 + \dot{z}_a^2$
X B0
B0 X
U1SQRT COMPUTE
$V_a\sqrt{V_a^2}$
$a \cos E = r_a v_a^2 - 1$
$a = \frac{r_a}{1 - \cos E}$
U3DOTP COMPUTE
$\bar{r}_a \cdot \bar{v}_a$
$(e \sin E)^2 = \frac{\bar{r}_a \cdot \bar{v}_a}{a}^2$
$e^2 = (e \sin E)^2 + (e \cos E)^2$
U1SQRT COMPUTE
$e\sqrt{e^2}$
e INTO 0,2
$H_{apogee} = K_1[a(1+3)-R_{SE}]$ INTO 1,2
$H_p = a(1-e) - R_{S-E}$
$H_{perigee} = K_1 \cdot H_p$ INTO 2, 2
X D0
D0 X
S = 0
H_p GT $75/K_1$
$\Delta H = \frac{75}{K_1} - H_p$
X A2 271
CEL6
$t_a, \bar{r}_a, \bar{v}_a$ INTO $t_a, \bar{r}_a, \bar{v}_a$
X A8 271


COMP
$H'_p = r_p - R_{SE}$
B0
$\Delta H' = \frac{75}{K_1} - H'_p$
C0
GETPI = GMTPI-t_{lo}
INTO 6,2
$v_d^2 = \frac{2}{R} - \frac{1}{\left(a' + \frac{\Delta H'}{2}\right)}$
A2
$V_d^2 = \frac{2}{H} - \frac{1}{\left(a + \frac{\Delta H}{2}\right)}$
UISQRT COMP
V_d V_d^2
UISQRT COMPUTE
$V_d = \sqrt{V_d^2}$
IS THERE AN ERROR
YES
C6
NO
$\Delta V_d = V_d - V_a$
ΔV_d LT = 0
CEL2
C5
0 INTO 3,2
INTO 1,2
INTO 2,2
$V'_a = \left(1 + \frac{\Delta V_d}{V_a}\right) V_a$
$K_2 \Delta V_d$
INTO 3,2
C6
0 INTO 3,2
INTO 4,2
D6
RESTORE XR'S
$t_a, \bar{r}_a \bar{v}'_a$
INTO $t_a, \bar{r}_a, \bar{v}_a$
$\Delta t_d = \frac{\Delta V_d}{OAMS_{occ}}$
INTO 4,2
$GMTPI = \frac{t_a - \Delta t}{2}$
EXIT
TO 2,4
CEL6
A8
CEAPRV COMP
$t_p, \bar{r}_p, \bar{v}_p$,
a' USING $t_a, \bar{r}_a, \bar{v}_a$
GMTPI LT t
GMTPI INTO 5,2
0 INTO 5,2
INTO 6,2

38. CEAPRV - COMPUTE THE VECTORS AT APOGEE OR PERIGEE.

GIVEN A $t, \bar{r}, \bar{v}$ DEFINING THE TRAJECTORY OF A SPACECRAFT, CEAPRV DETERMINES THE $t, \bar{r}, \bar{v}$ AT APOGEE OR PERIGEE. IT IS A SUBROUTINE OF CELEVP, WHICH COMPUTES THE VELOCITY INCREMENT TO BE APPLIED AT APOGEE TO RAISE THE HEIGHT AT THE FOLLOWING PERIGEE TO 75 NAUTICAL MILES. (SEE FLOWCHART.)

INPUT

A) THE INPUT $t, \bar{r}, \bar{v}$ IS IN FLOATING POINT IN A SEVEN-LOCATION TABLE. WHEN AN ENTRY IS MADE TO COMPUTE THE VECTORS AT PERIGEE, THE INPUT $t, \bar{r}, \bar{v}$ MUST BE THE TIME, POSITION AND VELOCITY AT APOGEE. THE TABLE IS FORMATTTED AS FOLLOWS -

L(INPUT VECTORS)		
	t	TIME IN GMT SECONDS
+1	x	$\bar{r}$ POSITION VECTOR IN GEMINI LENGTH UNITS
+2	y	
+3	z	
+4	$\dot{x}$	$\bar{v}$ VELOCITY VECTOR IN GEMINI VELOCITY UNITS
+5	$\dot{y}$	
+6	$\dot{z}$	

B) CONSTANTS

KM0000	DEC -0	MINUS ZERO
K00000	DEC 0	ZERO
K00001	DEC 1	FIXED POINT ONE
K00002	DEC 2	FIXED POINT TWO
K00004	DEC 4	FIXED POINT FOUR

K00006	DEC 6	FIXED POINT SIX
K00007	DEC 7	FIXED POINT SEVEN
K00022	DEC 22	FIXED POINT TWEN-TY-TWO
KM0.UT	DEC-806.81364	CONVERTS GEMINI TIME UNITS TO SEC-ONDS (NEGATIVE)
K000.5	DEC .5	FLOATING POINT ON-HALF
K001.0	DEC 1.0	FLOATING POINT ONE
K00.PI	DEC 3.1415927	FLOATING POINT PI (π)
KKK2PI	DEC 6.2831853	FLOATING POINT TWO PI (2π)
G022.0	DEC 22.	FLOATING POINT TWENTY-TWO
K032.0	DEC 32.	FLOATING POINT THIRTY-TWO
K128.0	DEC 128.	FLOATING POINT ONE-HUNDRED AND TWENTY-EIGHT
K00.UT	DEC 806.81364	CONVERTS GEMINI TIME UNITS TO SEC-ONDS
KCH233	DEC 233000000000	MASK FOR FLOATING POINT CHARACTER-ISTIC

C) PARAMETERS -

TCWF.B	CAPSULE ORBIT WEIGHT IN FLOATING POINT POUNDS

D) PROGRAM -

CORKNI	LAUNCH NUMERICAL INTEGRATION PRO-GRAM

E) SUBROUTINE -

U1ATAB	FLOATING-POINT ARCTANGENT ROUTINE
U3VMAG	FLOATING-POINT VECTOR MAGNITUDE ROUTINE
U1SQRT	FLOATING-POINT SQUARE ROOT ROUTINE
U3DOTP	FLOATING-POINT DOT PRODUCT ROUTINE

INTERMEDIATE OUTPUT

A) TABLES

TCRKOH	SEE CHABRT FOR DESCRIPTION
TCRKOT	SEE CHABRT FOR DESCRIPTION

(THESE TABLES ARE USED ONLY FOR INTERMEDIATE OUTPUT BY CEAPRV IN ORDER TO AVOID DEFINING EXTRA INTERMEDIATE STORAGE WITHIN CEAPRV ITSELF. THE FIRST 8 DECIMAL LOCATIONS OF TABLE TCRKOH ARE USED BY CEAPRV AND THE FIRST 50 DECIMAL LOCATIONS OF TABLE TCRKOT ARE USED.)

OUTPUT

THE OUTPUT IS IN FLOATING POINT FORMATTED IN AN EIGHT-LOCATION TABLE AS FOLLOWS -

L(OUTPUT)	t	TIME OF APOGEE OR PERIGEE IN GMT SECONDS
+1	x	$\bar{r}$ POSITION VECTOR AT APOGEE OR PERIGEE IN GEMINI LENGTH UNITS
+2	y	
+3	z	
+4	$\dot{x}$	$\bar{v}$ VELOCITY VECTOR AT APOGEE OR PERIGEE IN GEMINI VELOCITY UNITS
+5	$\dot{y}$	
+6	$\dot{z}$	
+7	a	SEMI MAJOR AXIS OF THE ELLIPSE IN GEMINI LENGTH UNITS COMPUTED

FROM THE VECTORS INPUT TO CEAPRV.

METHOD

UPON ENTRY TO THE PROGRAM, INDEX REGISTERS ONE THROUGH SIX ARE SAVED AND THE LOCATIONS OF THE INPUT AND OUTPUT TABLES ARE PICKED UP. THE GENERAL METHOD USED IS TO COMPUTE AN APPROXIAMTE TIME FOR APOGEE OR PERIGEE USING KEPLERIAN EQUATIONS -

$$e \cos E = rv^2 - 1$$

$$a = \frac{r}{1 - e \cos E}$$

$$\frac{P}{2} = K_1 \pi a^{3/2}$$

WHERE

P IS THE PERIOD OF MOTION FOR THE ELLIPSE IN SECONDS AND K_1 IS THE FACTOR TO CONVERT GEMINI TIME UNITS TO SECONDS, KOO.UT.

B) IF THE APPROXIMATE TIME OF PERIGEE IS BEING COMPUTED, THE INPUT TIME IS THE TIME OF APOGEE. THE TIME OF PERIGEE OS COMPUTED BY THE FOLLOWING -

$$t_p = t_a + \frac{P}{2}$$

C) IF THE APPROXIMATE TIME OF APOGEE IS BEING COMPUTED, THE FOLLOWING EQUATIONS ARE APPLIED -

$$e \sin E = \frac{\bar{r} \cdot \bar{v}}{a} \qquad 1$$

$$E = \tan^{-1} \frac{e \sin E}{e \cos E} \quad 0 \leq E \leq 2\pi$$

$$M = E - e \sin E \qquad 2$$

$$(t-T) = K_1 M a^{3/2}$$

$$t_1 = t - (t-T)$$

$$t_a = t_1 + \frac{P}{2}$$

THE SEMI-MAJOR AXIS OF THE ELLIPSE, a, IS PLACED IN THE OUTPUT TABLE. THE ACTUAL TIME OF APOGEE OR PERIGEE WILL DIFFER FROM THE COMPUTED TIME BECAUSE THE DRAG FUNCTION AND THE OBLATENESS OF THE EARTH WHICH TEND TO DECAY THE ORBIT ARE NOT TAKEN INTO ACCOUNT IN THE ABOVE EQUATIONS. IF THE APPROXIMATE GMT OF APOGEE IS LESS THAN THE GMT OF THE INPUT VECTORS, THE VECTORS AT APOGEE WILL NOT BE DETERMINED. THE INDEX REGISTERS ARE RESTORED AND NORMAL RETURN 1 IS MADE FROM THE PROGRAM.

KNOWING THE APPROXIMATE TIME OF APOGEE OR PERIGEE, AN ITERATIVE PROCEDURE IS USED TO FIND THE TRUE TIME OF APOGEE OR PERIGEE AND THE ASSOCIATED POSITION AND VELOC-

1. FUNDAMENTALS OF CELESTIAL MECHANICS, J.M.A.DANBY, THE MACMILLAN CO., NY, 1962, P.162.
2. IBID., P.125

ITY VECTORS. A FACTOR OF ONE-HUNDRED AND TWENTY-EIGHT SECONDS IS SUBTRACTED FROM THIS APPROXIMATE TIME, AND THE INPUT VECTORS ARE INTEGRATED AHEAD TO THIS NEW TIME USING CORKNI TYPE 2 WITH TWENTY-TWO STEPS AND NO LIFT AND NO THRUST. THE STEP SIZE FOR THIS INTEGRATION IS COMPUTED USING -

$$\text{step size} = \frac{t_a - 128}{22} \text{ (for apogee) or step size} = \frac{t_p - 128}{22} \text{ (for perigee).}$$

USING THE VECTORS OBTAINED FROM THIS INTEGRATION, A TABLE OF VECTORS IS GENERATED USING CORKNI TYPE 4 TO INTEGRATE SEVEN STEPS FORWARD AT A THIRTY TWO SECOND STEP SIZE. THE TABLE CONTAINS EIGHT POSITION AND VELOCITY VECTORS REFLECTING THE TRAJECTORY ONE-HUNDRED AND TWENTY EIGHT SECONDS BEFORE, AND NINETY-SIX SECONDS AFTER THE TIME OF APOGEE OR PERIGEE WHICH WAS FIRST COMPUTED.

SINCE APOGEE IS THAT POINT ON THE ORBIT WHICH IS FARTHEST FROM THE CENTER OF THE EARTH, AND PERIGEE IS THAT POINT WHICH IS CLOSEST TO THE CENTER OF THE EARTH, A BETTER APPROXIMATION OF THE TRUE APOGEE OR PERIGEE CAN BE DETERMINED BY CHOOSING THE t AND $\bar{v}$ ASSOCIATED WITH THE LARGEST OR SMALLEST $|r|^2$ DEPENDING ON WHETHER APOGEE OR PERIGEE IS BEING DETERMINED. IF THE CRITICAL $|r|^2$ IS THE FRIST OR LAST ENTRY IN THE TABLE, IT IS NOT POSSIBLE TO DETERMINE, WITHOUT FURTHER INTEGRATION BACKWARDS OR FORWARD, WHETHER OR NOT THE $|r|^2$ CHOSEN IS THE LARGEST OR SMALLEST THAT EXISTS FOR THE TRAJECTORY. IN THIS CASE, OUR ORIGINAL ESTIMATE OF THE APPROXIMATE TIME OF APOGEE OR PERIGEE WAS TOO INACCURATE AND COMPUTATIONS CANNOT BE FINISHED. INDEX REGISTERS ARE RESTORED AND THERE IS AN ERROR RETURN FROM THE PROGRAM.

ON THE OTHER HAND, IF THE POSITION VECTOR MAGNITUDES ARE ALL OF THE SAME SIZE, IT IS LIKELY THAT THE TRAJECTORY IS NEARLY CIRCULAR AND HAS NO TRUE APOGEE OR PERIGEE SINCE ALL POINTS ARE THE SAME DISTANCE FROM THE CENTER OF THE ERATH. IN THIS CASE, ALSO, THERE IS AN ERROR RETURN FROM THE PROGRAM. IN GENERAL, HOWEVER, UNLESS ECCENTRICITY IS SMALL MEANING THAT THE ORBIT IS NEARLY CIRCULAR AND THAT APOGEE AND PERIGEE ARE POORLY DEFINED, THE MAXIMUM OR MINIMUM $|r|^2$ WILL NOT BE THE FIRST OR LAST ENTRY IN THE TABLE.

NOW, ASSUMING THAT A MAXIMUM OR MINIMUM $|r|^2$ HAS BEEN FOUND SUCCESSFULLY, THE ASSOCIATED TIME IS COMPUTED KNOWING THE TIMES OF THE FIRST AND LAST ENTRIES IN THE TABLE, AND THE ASSOCIATED VELOCITY IS PICKED UP FROM THE TABLE. THE $t, \bar{r}$ AND $\bar{v}$ ARE USED AS INPUT TO CORKNI TYPE 4 TO INTEGRATE ONE STEP FORWARD AND ONE STEP BACKWARD AT A SIXTEEN SECOND STEP SIZE WHICH IS HALF OF THE STEP SIZE ORIGINALLY USED. THE MAXIMUM OR MINIMUM $|r|^2$ IS AGAIN CHOSEN AND THE ASSOCIATED $t, \bar{r}$ AND $\bar{v}$ ARE PICKED UP.

THE ITERATIVE PROCESS CONTINUES. EACH TIME A NEW $t, \bar{r}$ AND $\bar{v}$ ARE DETERMINED, THE STEP SIZE IS HALVED BEFORE THE NEXT ENTRY TO CORKNI TYPE 4 IS MADE. THE $t, \bar{r}$, AND $\bar{v}$ ASSOCIATED WITH THE MAXIMUM OR MINIMUM $|r|^2$ ARE INTEGRATED ONE STEP FORWARD AND ONE STEP BACKWARDS. THE PROCESS IS COMPLETED WHEN THE $t, \bar{r}$, AND $\bar{v}$ ASSOCIATED WITH A FOUR SECOND STEP SIZE HAVE BEEN FOUND. AT THIS POINT, THE APOGEE OR PERIGEE t, r AND v HAVE BEEN DETERMINED TO WITHIN APPOXIMATELY TWO SECONDS OF THE TRUE APOGEE OR PERIGEE.

THE $t, \bar{r}$ AND $\bar{v}$ AT APOGEE OR PERIGEE ARE PLACED IN THE OUTPUT TABLE, INDEX REGISTERS ARE RESTORED, AND AN EXIT IS MADE FROM THE PROGRAM.

USAGE

A) CALLING SEQUENCE -

TSX CEAPRV,4

H L(INPUT t,$\bar{r}$,$\bar{v}$),,L(OUTPUT)

ERROR RETURN

NORMAL RETURN 1

NORMAL RETURN 2

H = PZE MEANS COMPUTE THE VECTORS AT APOGEE

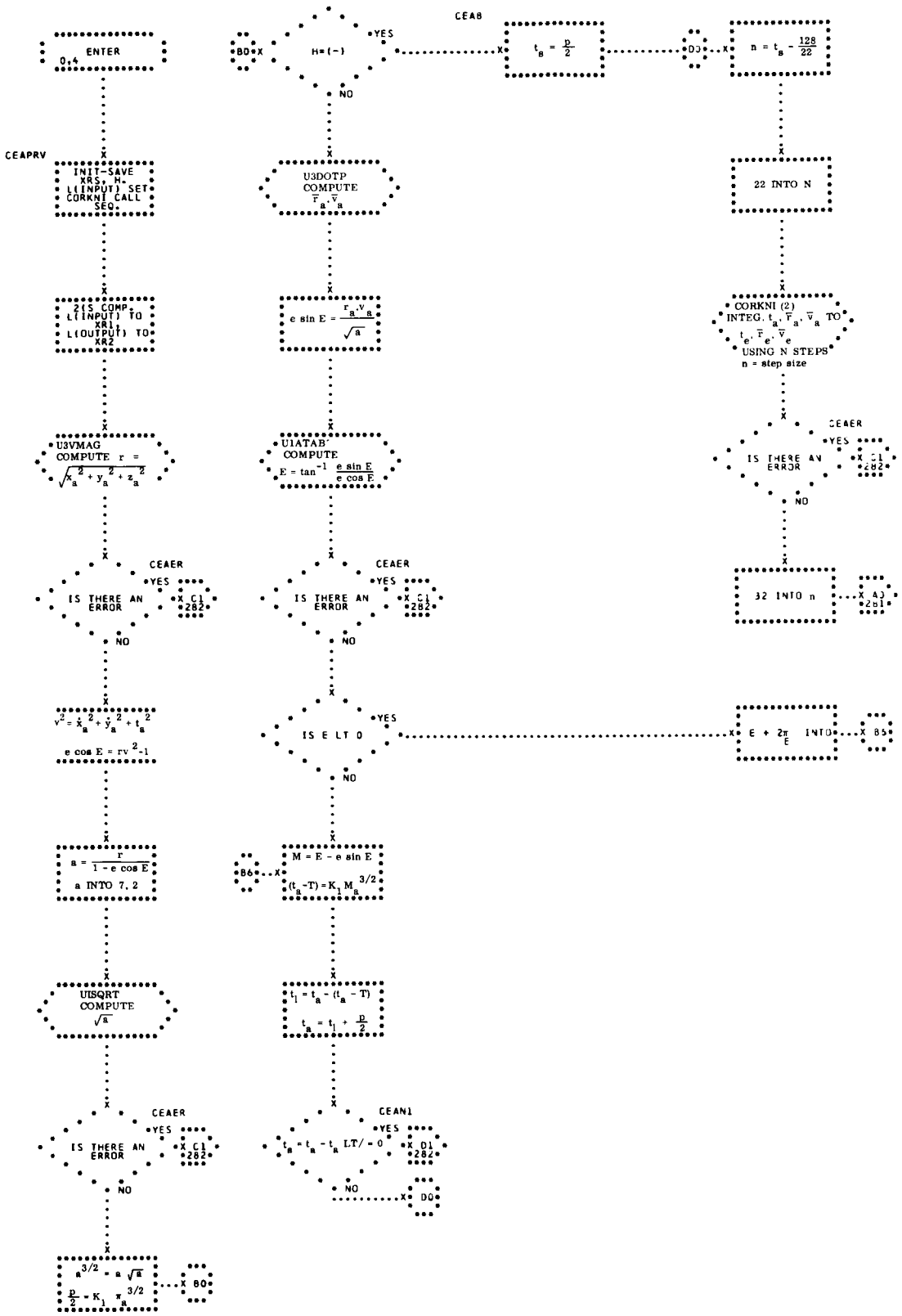
H = MZE MEANS COMPUTE THE VECTORS AT PERIGEE

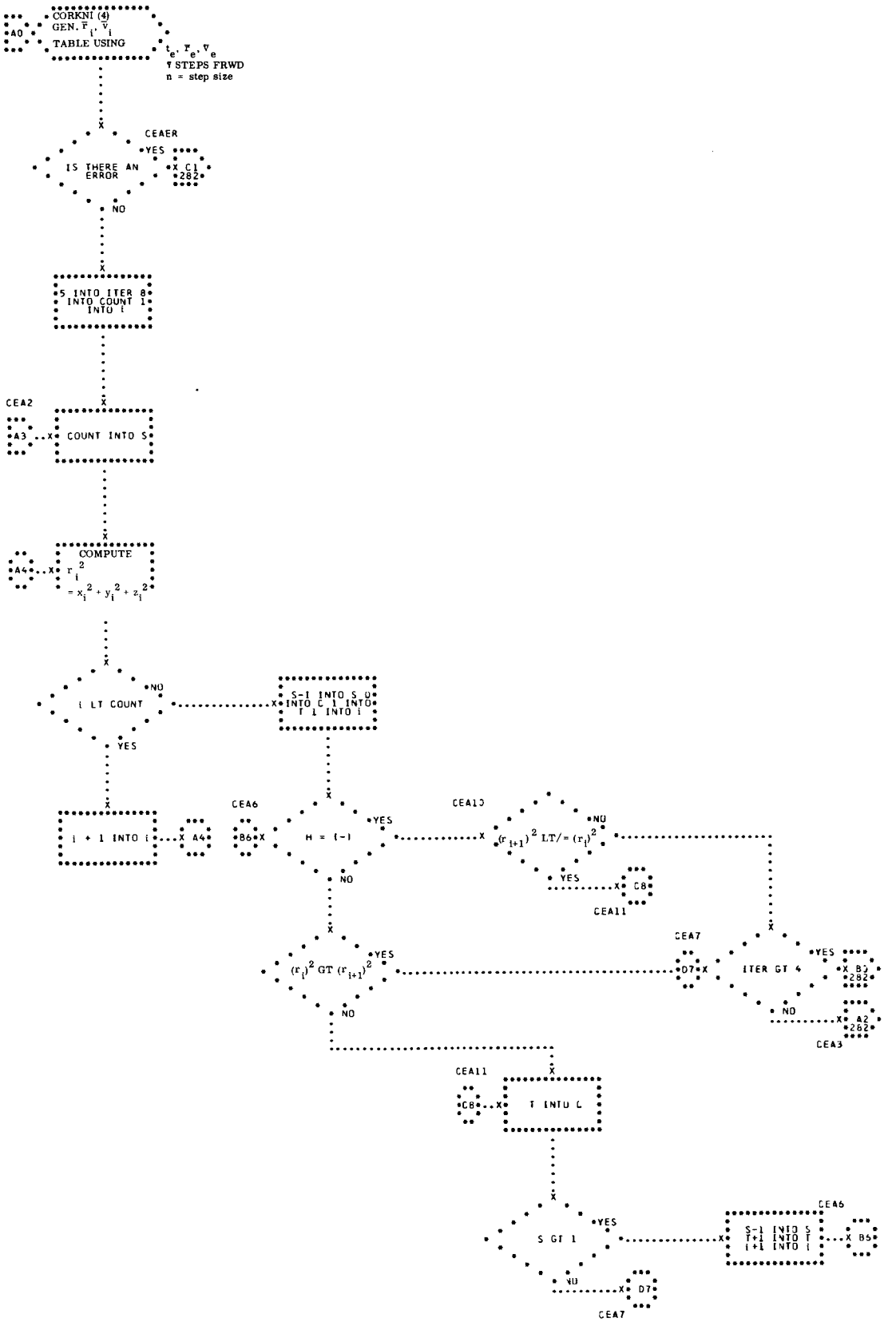
B) STORAGE REQUIRED -

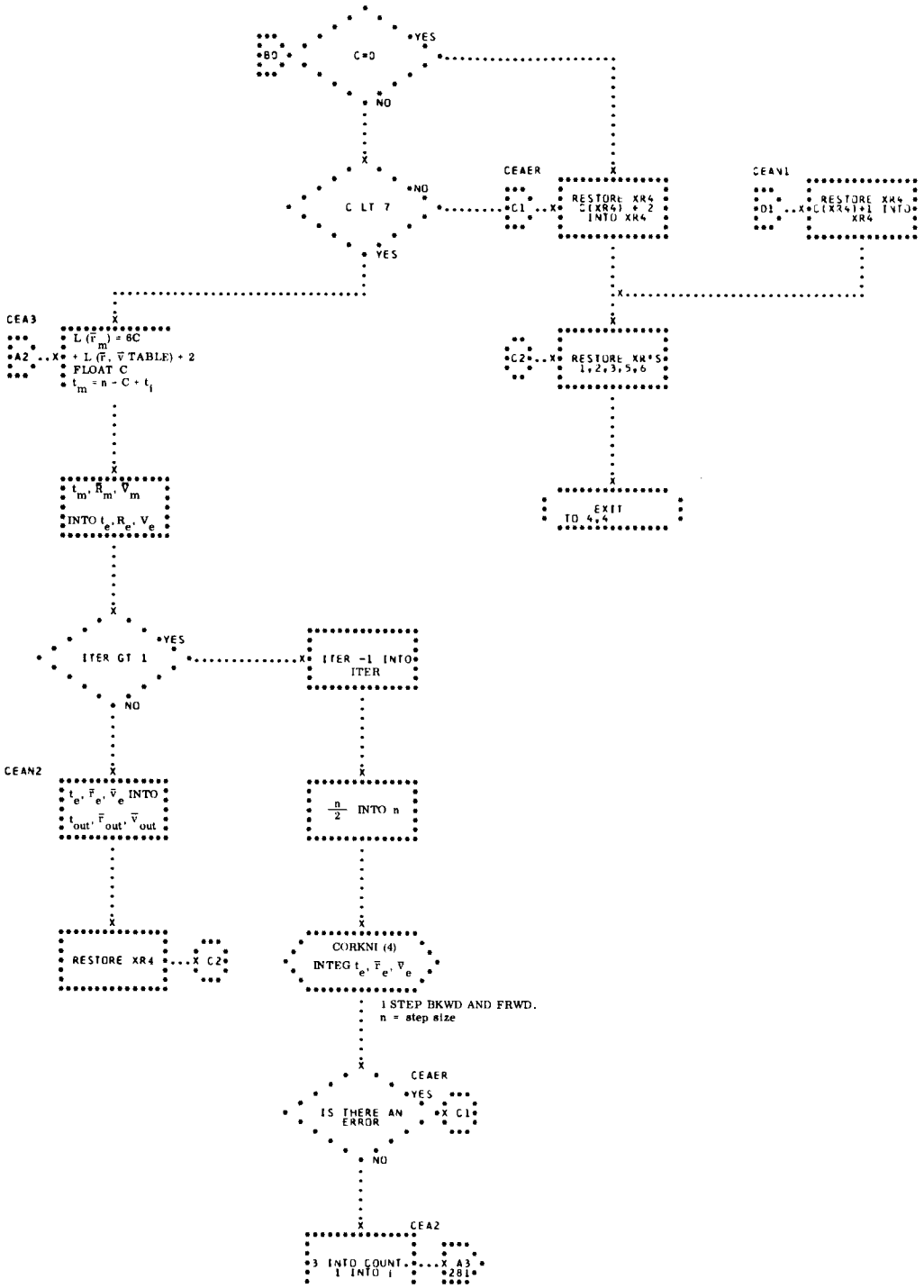
197 DECIMAL LOCATIONS

C) TIMING -

268 MILLISECONDS (MAXIMUM) INCLUDES ALL SUB-ROUTINES


CEAB
ENTER
0.4
YES
H=(-)
B0
NO
$t_B = \frac{p}{2}$
D0
$n = t_B - \frac{128}{22}$
CEAPRV
INIT-SAVE XRS, H, L(INPUT) SET CORKNI CALL SEQ.
U3DOTP COMPUTE $\bar{r}_a, \bar{v}_a$
22 INTO N
2(S COMP L(INPUT) TO XR1 L(OUTPUT) TO XR2
$e \sin E = \frac{\bar{r}_a \cdot \bar{v}_a}{\sqrt{a}}$
CORKNI (2) INTEG. $t_a, \bar{r}_a, \bar{v}_a$ TO $t_e, \bar{r}_e, \bar{v}_e$ USING N STEPS
n = step size
U3VMAG COMPUTE $r = \sqrt{x_a^2 + y_a^2 + z_a^2}$
U1ATAB COMPUTE $E = \tan^{-1} \frac{e \sin E}{e \cos E}$
CEAER
IS THERE AN ERROR
YES
X C1 282
NO
CEAER
IS THERE AN ERROR
YES
X C1 282
NO
CEAER
IS THERE AN ERROR
YES
X C1 282
NO
32 INTO n
X A1 281
$v^2 = \dot{x}_a^2 + \dot{y}_a^2 + \dot{z}_a^2$
$e \cos E = rv^2 - 1$
IS E LT 0
YES
NO
$E + 2\pi$ INTO E
X B5
$a = \frac{r}{1 - e \cos E}$
a INTO 7.2
M = E - e sin E
B6
$(t_a - T) = K_1 M_a^{3/2}$
U1SQRT COMPUTE $\sqrt{a}$
$t_1 = t_a - (t_a - T)$
$t_a = t_1 + \frac{p}{2}$
CEAER
IS THERE AN ERROR
YES
X C1 282
NO
CEAN1
$t_a = t_a - t_a$ LT/= 0
YES
X D1 282
NO
D0
$a^{3/2} = a\sqrt{a}$
$\frac{p}{2} = K_1 \pi a^{3/2}$
X B0


A0
CORKNI (4)
GEN. $\bar{r}_i$, $\bar{v}_i$
TABLE USING
t_e, $\bar{r}_e$, $\bar{v}_e$
¶ STEPS FRWD
n = step size
CEAER
IS THERE AN ERROR
YES
X C1 282
NO
S INTO ITER B
INTO COUNT 1
INTO i
CEA2
A3
COUNT INTO S
COMPUTE
A4
$r_i^2 = x_i^2 + y_i^2 + z_i^2$
i LT COUNT
NO
YES
S-1 INTO S D
INTO C 1 INTO
I 1 INTO i
i + 1 INTO i
A4
CEA6
B6
H = (-)
YES
NO
CEA10
$(r_{i+1})^2$ LT/= $(r_i)^2$
NO
YES
C8
CEA11
$(r_i)^2$ GT $(r_{i+1})^2$
YES
NO
CEA7
D7
ITER GT 4
YES
NO
X B3 282
X A2 282
CEA3
CEA11
C8
T INTO C
S GT 1
YES
NO
D7
CEA7
CEA6
S-1 INTO S
T+1 INTO T
i+1 INTO i
X B5


B0
C=0
YES
NO
C LT 7
NO
YES
CEAER
C1
RESTORE XR4
C(XR4) + 2
INTO XR4
CEAN1
D1
RESTORE XR4
C(XR4)+1 INTO
XR4
CEA3
A2
L (r_m) = 6C
+ L (r, v TABLE) + 2
FLOAT C
t_m = n - C + t_i
C2
RESTORE XR*S
1,2,3,5,6
EXIT
TO 4,4
t_m, R_m, V_m
INTO t_e, R_e, V_e
ITER GT 1
YES
NO
ITER -1 INTO
ITER
CEAN2
t_e, r_e, v_e INTO
t_out, r_out, v_out
n/2 INTO n
RESTORE XR4
C2
CORKNI (4)
INTEG t_e, r_e, v_e
1 STEP BKWD AND FRWD.
n = step size
CEAER
IS THERE AN
ERROR
YES
C1
NO
CEA2
3 INTO COUNT.
1 INTO I
A3
281

39. CPRVCM - LAUNCH PRESENT VECTORS COMPUTATIONS

THIS PROGRAM COMPUTES CERTAIN QUANTITIES FOR DISPLAY FROM THE INPUT VECTORS AND THEIR ASSOCIATED TIME. (SEE FLOWCHART)

INPUT

A) CONSTANTS -

K00001	DEC 1 FIXED POINT ONE
K00003	DEC 3 FIXED POINT THREE
K.ERGU	EQUATORIAL RADIUS OF THE EARTH IN FLOATING POINT GEMINI LENGTH UNITS, $\bar{R}$
K.OMES	ROTATION OF THE EARTH IN FLOATING POINT RADIANS PER SECOND, ω_e
KCGUNM	NAUTICAL MILES PER GEMINI LENGTH UNIT IN FLOATING POINT, k_1
KFTSCG	FEET PER SECOND PER GEMINI VELOCITY UNIT IN FLOATING POINT, k_1
KGUFTT	FEET PER GEMINI LENGTH UNIT IN FLOATING POINT, k_3
KLAMDO	LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH IN FLOATING POINT RADIANS, λ_0
KLAMSP	LONGITUDE OF PAD IN FLOATING POINT RADIANS, λ_p (THROUGHOUT THE PROGRAM THE LOCATION OF THE CENTRAL RATE TRACKER OF THE GE RADAR IS USED

AS THE LOCATION OF THE PAD. THIS THIS WAS DONE TO HAVE THE OUTPUT CONSISTANT WITH THAT OF THE BURROUGHS GE COMPLEX AT CAPE KENNEDY.)

KSNLPP — SINE OF GEODETIC LATITUDE OF THE PAD ($\sin L'_p$)

KCSLPP — COSINE OF GEODETIC LATITUDE OF THE PAD ($\cos L'_p$)

KRPCLP — $(R'_0)(\cos L_p)$ WHERE R'_0 IS THE DISTANCE FROM THE GEOCENTER TO THE PAD IN FLOATING POINT GEMINI LENGTH UNITS AND L_p IS THE GEOCENTRIC LATITUDE OF

KRPSLP — $(R'_0)(\sin L_p)$ THE (d, s, w) ORIGIN IN FLOATING POINT RADIANS

B) TABLES -

TMCSBO — A 1-LOCATION TABLE CONTAINING THE COSINE OF β_0 IN FLOATING POINT RADIANS

TMSNBO — A 1-LOCATION TABLE CONTAINING THE SINE OF β_0 IN FLOATING POINT RADIANS (β_0 IS THE NOMINAL AZIMUTH OF THE SPACECRAFT FROM THE GE RADAR AT CAPE KENNEDY AT THE NOMINAL TIME OF SECOND-STAGE CUTOFF

TPVRQD — A 3-LOCATION PARAMETER TABLE USED IN COMPUTING THE REQUIRED

VELOCITY, V_r

L(INPUT) THE ADDRESS OF A 7-LOCATION TABLE CONTAINING THE INPUT VECTORS AND THEIR ASSOCIATED TIME

L(INPUT	t_i	TIME OF VECTORS IN FLOATING POINT GMT SECONDS
+1	x	$\bar{r}_3$ - POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+2	y	
+3	z	
+4	$\dot{x}$	$\bar{v}_3$ - VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+5	$\dot{y}$	
+6	$\dot{z}$	

C) THE SUBROUTINE A3GSCP AND THE LIBRARY ROUTINES U1ASCO, U1ATNA, U1SICO, U1SQRT, AND U3VMAG ARE ALSO USED.

OUTPUT

A) TABLES -

L(OUTPUT) A 13-LOCATION TABLE CONTAINING THE COMPUTED DISPLAY QUANTITIES (SEE FLOWCHART)

METHOD

CPRVCM USES THE FOLLOWING FORMULAS IN ITS COMPUTATIONS -

A) $r = (X^2 + Y^2 + Z^2)^{1/2}$

B) $r(\text{feet}) = k_3 r$

C) $v = (\dot{X}^2 + \dot{Y}^2 + \dot{Z}^2)^{1/2}$

D) $V_r = (k_6 r + k_7)/r^{1/2}$

E) $\gamma = \sin^{-1}\left(\frac{\bar{r}\cdot\bar{v}}{r\ v}\right)$

F) $i = \tan^{-1}((h_x^2 + h_y^2)^{1/2}/h_z)$

WHERE

$$h_x^2 = ((Y)(\dot{Z}) - (Z)(\dot{Y}))^2$$

$$h_y^2 = ((Z)(\dot{X}) - (X)(\dot{Z}))^2$$

$$h_z = ((X)(\dot{Y}) - (Y)(\dot{X}))$$

G) $R_2 = A^T B^T (C^T R_3 - R_c)$

WHERE -

$$R_2 = \begin{bmatrix} d \\ s \\ w \end{bmatrix}$$

$$R_3 = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$R_c = \begin{bmatrix} r'_0 \cos L_p \\ 0 \\ r'_0 \sin L_p \end{bmatrix}$$

$$A^T = \begin{bmatrix} -\cos \beta_0 & \sin \beta_0 & 0 \\ -\sin \beta_0 & -\cos \beta_0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$B^T = \begin{bmatrix} \sin L'_p & 0 & -\cos L'_p \\ 0 & 1 & 0 \\ \cos L'_p & 0 & \sin L'_p \end{bmatrix}$$

$$C^T = \begin{bmatrix} \cos \Omega & \sin \Omega & 0 \\ -\sin \Omega & \cos \Omega & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

FOR DERIVATION OF E), SEE ''DERIVATION OF FLIGHT PATH ANGLE FORMULA.''
FOR DERIVATION OF F), SEE ''DERIVATION OF INCLINATION ANGLE FORMULA.''
FOR DERIVATION OF G), SEE ''DERIVATION OF INERTIAL TO PAD COORDINATE CONVERSION.''

USAGE

A) CALLING SEQUENCE -

```
TSX       CPRVCM,4
H         L(INPUT),,L(OUTPUT)
RETURN
```

WHERE h MAY BE EITHER A PZE OR AN MZE. IF PZE, s AND d ARE INCLUDED IN THE COMPUTATIONS. IF MZE, s AND d ARE NOT INCLUDED IN THE COMPUTATIONS.

B) STORAGE REQUIRED -

178 LOCATIONS

C) TIMING (7094) -

H A PZE, 5.5604 MILLISECONDS

H AN MZE, 4.6084 MILLISECONDS

D) ERROR CONDITIONS -

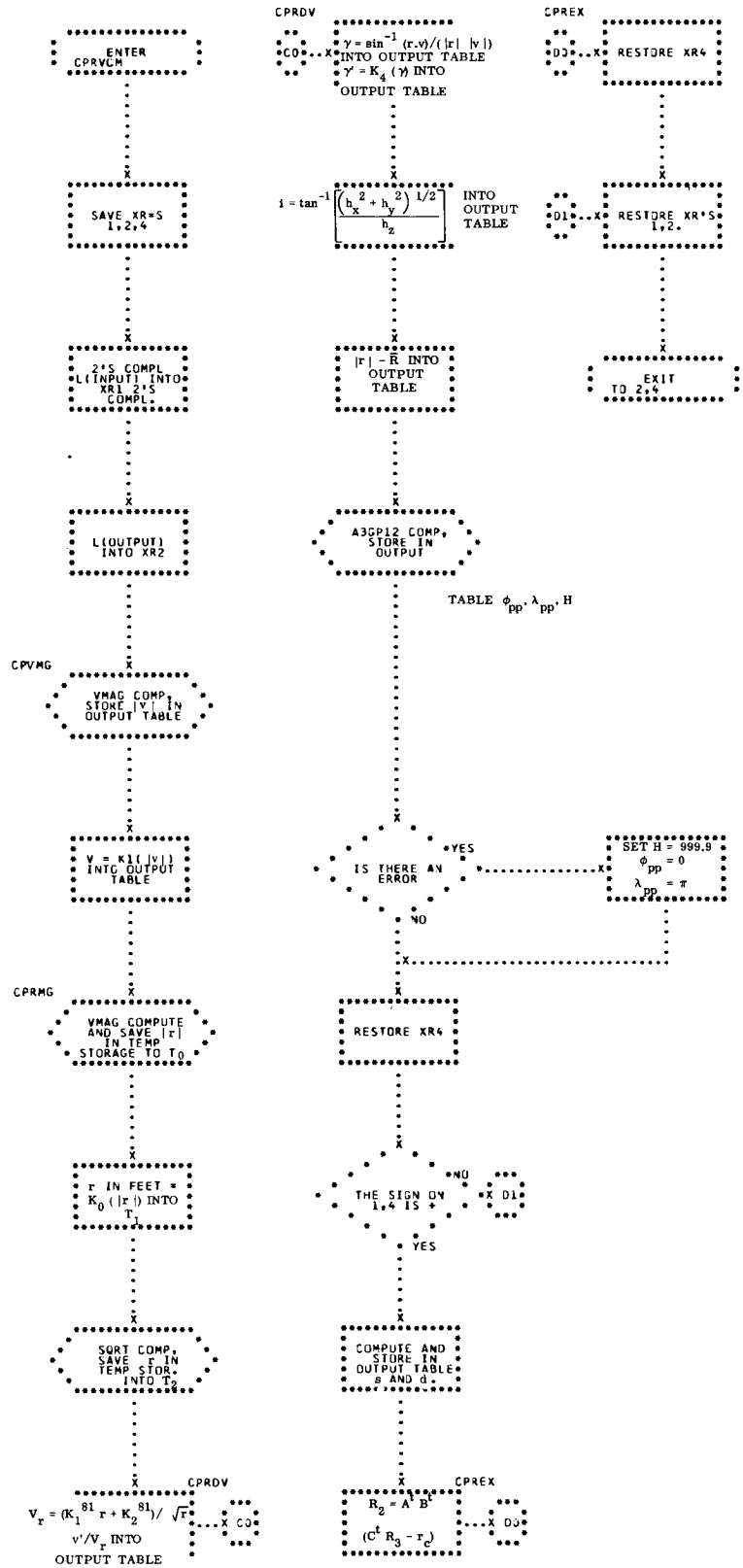
IF A3GSCP RETURNS VIA THE ERROR EXIT, THE QUANTITY H (THE HEIGHT OF THE SPACECRAFT ABOVE AN OBLATE EARTH) IS SET TO 999.9 NAUTICAL MILES, THE LATITUDE OF PRESENT POSITION IS SET TO ZERO, AND THE LONGITUDE OF PRESENT POSITION IS SET TO π RADIANS. ALL OTHER ERROR RETURNS (I.E., FROM LIBRARY SUBROUTINES) ARE IGNORED BY THE PROGRAM.

OUTPUT TABLE FROM CPRVCM

LOCATION L(OUTPUT)	DESCRIPTION
	MAGNITUDE OF INPUT VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS, $\|\bar{v}\|$
+1	MAGNITUDE OF INPUT VELOCITY VECTOR IN FLOATING POINT FEET PER SECOND, v'
+2	VELOCITY REQUIRED IN FLOATING POINT FEET PER SECOND, V_r
+3	FLIGHT PATH ANGLE IN FLOATING POINT DEGREES, γ
+4	FLIGHT PATH ANGLE IN FLOATING POINT RADIANS, γ
+5	RATIO OF ACTUAL VELOCITY TO VELOCITY REQUIRED FOR ORBIT, v'/V_r (DIMENSIONLESS)
+6	INCLINATION ANGLE IN FLOATING POINT RADIANS, i
+7	HEIGHT OF SPACECRAFT ABOVE A SPHERICAL EARTH IN FLOATING POINT NAUTICAL MILES, $(\|\bar{r}\| - \bar{R})$
+8	LATITUDE OF PRESENT POSITION IN FLOATING POINT RADIANS, ϕ_{pp}
+9	LONGITUDE OF PRESENT POSITION IN FLOATING POINT RADIANS, λ_{pp}

OUTPUT TABLE FROM CPRVCM

LOCATION	DESCRIPTION
+10	CROSSRANGE DISTANCE IN FLOATING POINT NAUTICAL MILES, s
+11	DOWNRANGE DISTANCE IN FLOATING POINT NAUTICAL MILES, d
+12	HEIGHT OF SPACECRAFT ABOVE AN OBLATE EARTH IN FLOATING POINT GEMINI LENGTH UNITS, H


ENTER CPRVCM
SAVE XR*S 1,2,4
2'S COMPL L(INPUT) INTO XR1 2'S COMPL.
L(OUTPUT) INTO XR2
CPVMG
VMAG COMP, STORE |V| IN OUTPUT TABLE
V = K1(|V|) INTO OUTPUT TABLE
CPRMG
VMAG COMPUTE AND SAVE |r| IN TEMP STORAGE TO T0
r IN FEET = K0(|r|) INTO T1
SQRT COMP, SAVE r IN TEMP STOR. INTO T2
CPRDV
Vr = (K1 81 r + K2 81)/√r
v'/Vr INTO OUTPUT TABLE
CO
CPRDV
γ = sin-1 (r.v)/(|r| |v|) INTO OUTPUT TABLE
γ' = K4 (γ) INTO OUTPUT TABLE
i = tan-1 [(hx2 + hy2)1/2 / hz] INTO OUTPUT TABLE
|r| - R INTO OUTPUT TABLE
A3GP12 COMP, STORE IN OUTPUT
TABLE φpp, λpp, H
IS THERE AN ERROR
YES
SET H = 999.9 φpp = 0 λpp = π
NO
RESTORE XR4
THE SIGN OF 1,4 IS +
NO
D1
YES
COMPUTE AND STORE IN OUTPUT TABLE a AND d.
CPREX
R2 = A t B t (Ct R3 - rc)
D0
CPREX
RESTORE XR4
D1
RESTORE XR*S 1,2.
EXIT TO 2,4

DERIVATION OF INERTIAL TO LAUNCH PAD COORDINATE CONVERSION

THIS SECTION DERIVES THE COORDINATE CONVERSION FORMULA -

IN COMPUTING THE DOWNRANGE DISTANCE (d) AND THE CROSSRANGE DISTANCE (s), IT IS NECESSARY TO CONVERT FROM THE GEMINI INERTIAL COORDINATE SYSTEM TO A SYSTEM WHICH HAS THE LAUNCH PAD AS ITS ORIGIN. THE PROCESS CONSISTS OF FOUR TRANSFORMATIONS WITH A NEW COORDINATE SYSTEM RESULTING FROM EACH TRANSFORMATION. THEREFORE, THE INERTIAL SYSTEM IS DESIGNATED AS THE X, Y, Z SYSTEM, THE SYSTEM RESULTING FROM THE FIRST TRANSFORMATION IS DESIGNATED AS THE X', Y', Z' SYSTEM, ETC. EACH SUBSECTION FOLLOWING COVERS ONE OF THE TRANSFORMATIONS AND THE RESULTING COORDINATE SYSTEM.

ROTAION OF A RECTANGULAR COORDINATE SYSTEM IN A PLANE, TRANSFORMATION 1

THE EQUATIONS FOR ROTATING A RECTANGULAR COORDINATE SYSTEM IN A PLANE (X, Y) TO A NEW SYSTEM (X', Y') IN THE SAME PLANE ARE GIVEN BELOW.

THE SYSTEM IS ROTATED IN A COUNTERCLOCKWISE DIRECTION THROUGH AN ANGLE Θ .

$$X = X' \cos \Theta - Y' \sin \Theta$$

$$Y = X' \sin \Theta + Y' \cos \Theta$$

(NOTE - SEE ANALYTICAL GEOMETRY. SMITH, GALE, AND NEELLEY GINN + CO., BOSTON, P. 117.)

BY USING CRAMER'S RULE FOR DETERMINANTS, ONE MAY EXPRESS THE NEW COORDINATE SYSTEM (X', Y') IN TERMS OF THE OLD SYSTEM (X,Y) AS FOLLOWS -

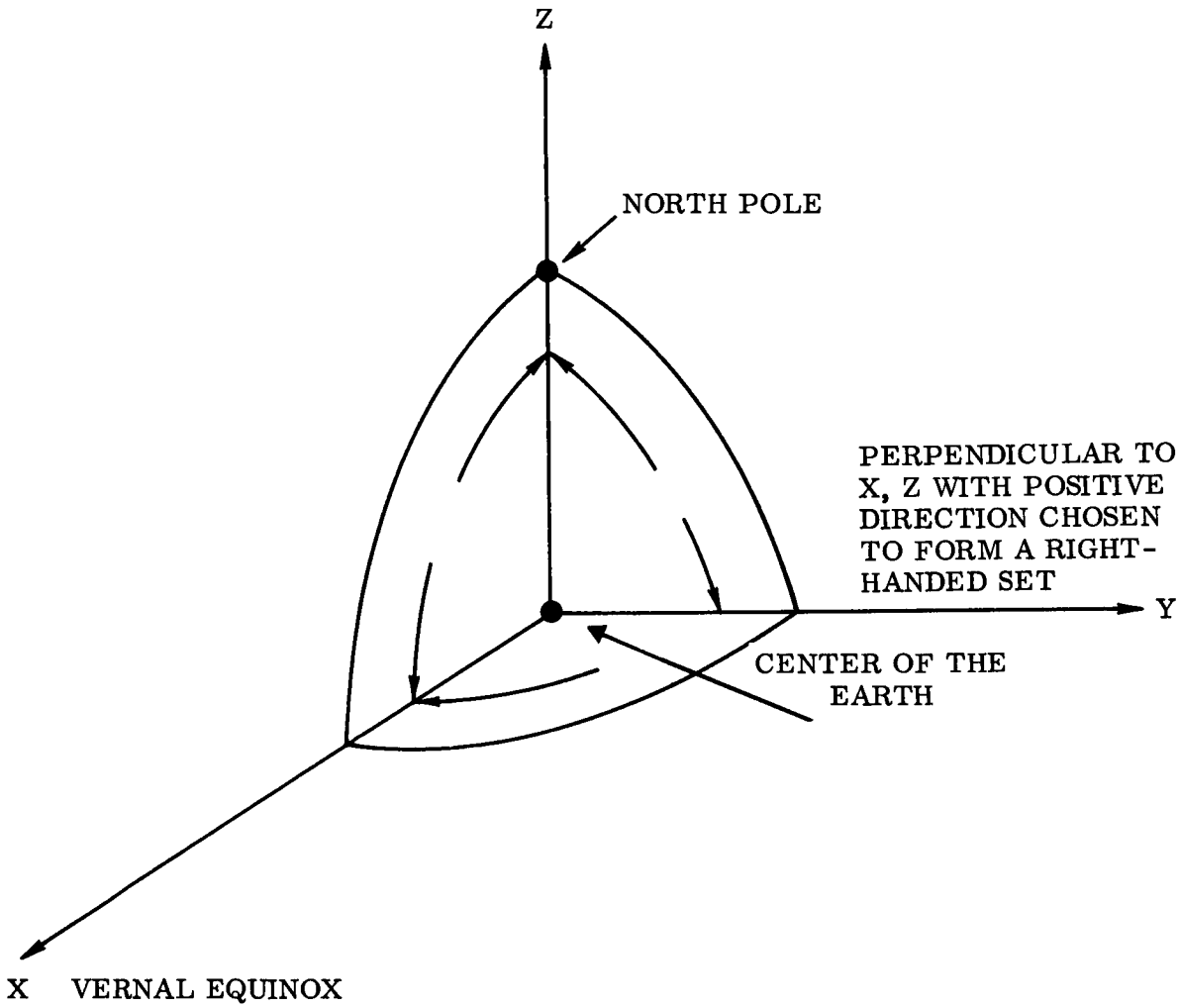
$$X' = \frac{\begin{vmatrix} X & -\sin\theta \\ Y & +\cos\theta \end{vmatrix}}{\begin{vmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{vmatrix}} = \frac{X\cos\theta + Y\sin\theta}{\cos^2\theta + \sin^2\theta} = X\cos\theta + Y\sin\theta$$

$$Y' = \frac{\begin{vmatrix} \cos\theta & X \\ \sin\theta & Y \end{vmatrix}}{\begin{vmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{vmatrix}} = \frac{Y\cos\theta - X\sin\theta}{\cos^2\theta + \sin^2\theta} = -X\sin\theta + Y\cos\theta$$

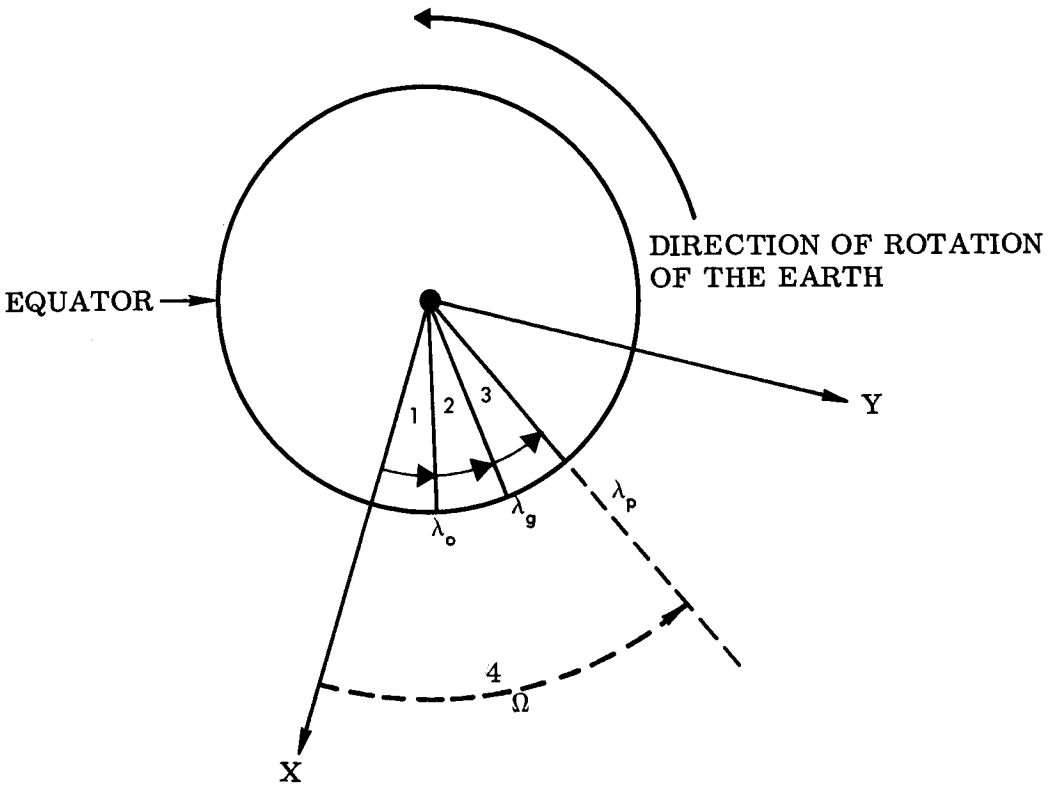
THEREFORE, THE BASIC EQUATIONS FOR ROTATING A COORDINATE SYSTEM THROUGH THE ANGLE Θ (Θ IS MEASURED IN A COUNTERCLOCKWISE DIRECTION) ARE -

$$\begin{aligned} X' &= X\cos\theta + Y\sin\theta \\ Y' &= -X\sin\theta + Y\cos\theta \end{aligned} \qquad (2)$$

THE ORIGINAL SYSTEM (X,Y,Z) IS THE GEMINI INERTIAL COORDINATE SYSTEM SHOWN ON THE NEXT PAGE.



THE (X,Y) PLANE IS THE EQUATORIAL PLANE SINCE THE ORIGIN OF THE SYSTEM IS THE CENTER OF THE EARTH.



THE PRECEDING FIGURE SHOWS THE (X,Y) PLANE AT ANY GIVEN TIME ON THE DAY OF A LAUNCH AS VIEWED FROM A POINT HIGH ON THE POSITIVE Z AXIS.

ANGLE 1 = (λ_0) REPRESENTS THE INERTIAL LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING THE LAUNCH

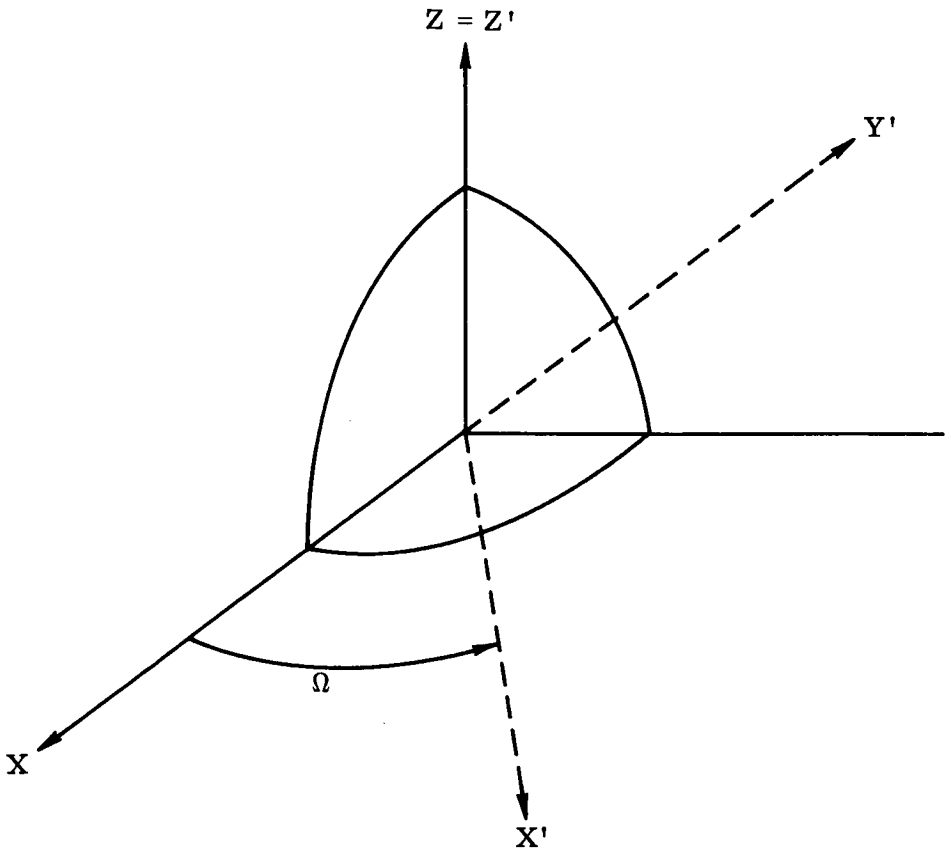
ANGLE 2 = (ωt), WHERE ω IS ANGULAR VELOCITY DUE TO THE EARTH'S ROTATION AND t IS THE ELAPSED TIME SINCE GREENWICH MIDNIGHT PRECEDING LAUNCH, REPRESENTS THE ANGLE THROUGH WHICH THE EARTH HAS ROTATED FROM MIDNIGHT UNTIL

THE PRESENT TIME.

ANGLE 3 = (λ_p) REPRESENTS THE TERRESTRIAL LONGITU
OF THE LAUNCH PAD.

ANGLE 4 = (Ω) IS THE SUM OF ANGLES 1, 2, AND 3.

TO TRANSFORM THE INERTIAL COORDINATES TO THOSE OF THE LAUNCH PAD, FIRST ROTATE THE SYSTEM ABOUT THE Z AX THROUGH THE ANGLE Ω .



THEREFORE, THE NEW SYSTEM (X', Y', Z') HAS Z' PASSING THROUGH THE NORTH POLE, X' INTERSECTING THE MERIDIAN OF THE LAUNCH PAD, AND Y' PERPENDICULAR TO THE OTHER TWO AXES WITH POSITIVE DIRECTION SUCH THAT A RIGHT-HANDED S OF AXES IS MAINTAINED. THE TRANSFORMATION OF THE COORDINATES IS GIVEN BY -

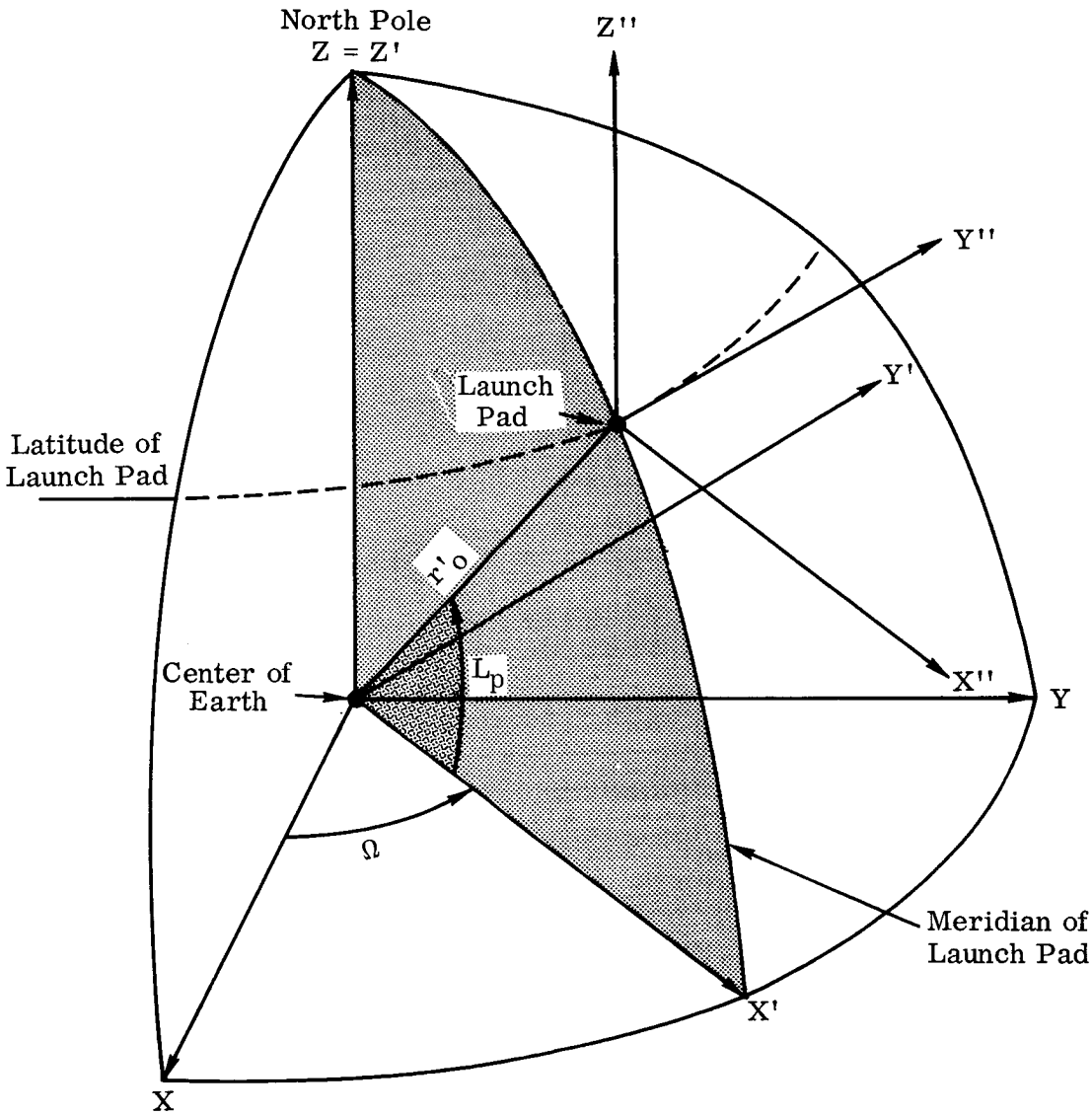
$$X' = X \cos \Omega + Y \sin \Omega$$
$$Y' = -X \sin \Omega + Y \cos \Omega$$

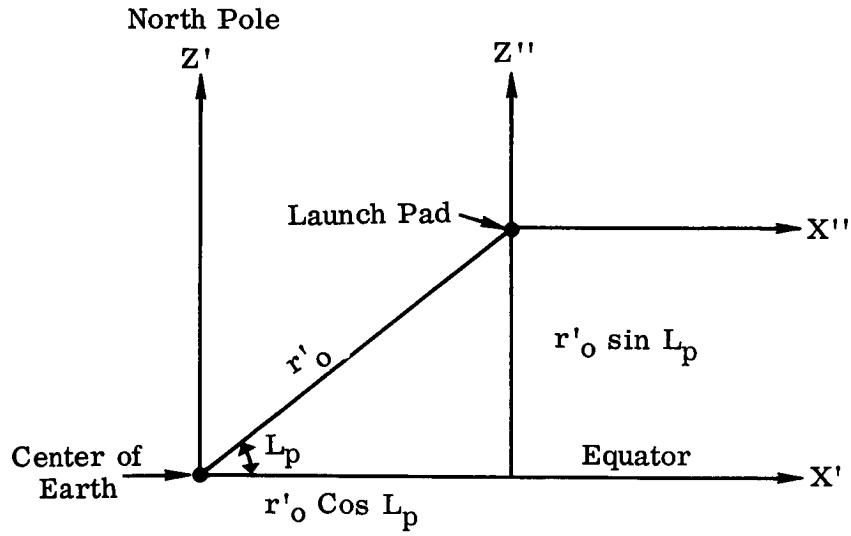
OR EXPRESSED IN MATRIX NOTATION -

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} \cos \Omega & \sin \Omega & 0 \\ -\sin \Omega & \cos \Omega & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

TRANSLATION OF ORIGIN, TRANSFORMATION 2

ONCE THE ORIGINAL SYSTEM HAS BEEN ROTATED THROUGH Ω, THE LAUNCH PAD WILL LIE IN THE (X' Z') PLANE. IT IS THEN NECESSARY TO TRANSLATE THE ORIGIN OF THE SYSTEM FROM THE CENTER OF THE EARTH TO THE LAUNCH PAD. THE DIAGRAMS BELOW SHOW THE TRANSLATION AND THE RELATIONSHIP OF THE OLD SYSTEM (EARTH CENTERED X', Y', Z') TO THE NEW SYSTEM (LAUNCH PAD CENTERED X" , Y" , Z").


North Pole
Z = Z'
Z''
Y''
Y'
Launch
Pad
Latitude of
Launch Pad
r'o
Center of
Earth
L_p
X''
Y
Ω
Meridian of
Launch Pad
X'
X



L_p = GEOCENTRIC LATITUDE OF THE LAUNCH PAD

r'_o = DISTANCE FROM THE CENTER OF THE EARTH TO THE LAUNCH PAD

SINCE THIS TRANSFORMATION INVOLVES TRANSLATING THE ORIGIN OF THE SYSTEM, THE PLANE DETERMINED BY THE X'', Y'' AXIS IS PARALLEL TO THE EQUATORIAL PLANE AND ALL OF THE AXES IN THE X'', Y'', Z'' SYSTEM ARE PARALLEL TO THE X', Y', Z' AXES. THE NEW COORDINATES ARE GIVEN BY -

$$X'' = X' - r'_o \cos L_p$$

$$Z'' = Z' - r'_o \sin L_p$$

$$Y'' = Y'$$

THE NEGATIVE SIGNS OCCUR IN THE ABOVE EQUATIONS SINCE THE ANGLE L_p IS POSITIVE.

THE ABOVE EQUATIONS MAY BE WRITTEN IN MATRIX NOTATION AS FOLLOWS -

$$\begin{pmatrix} X'' \\ Y'' \\ Z'' \end{pmatrix} = \begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} - \begin{pmatrix} r'_o \cos L_p \\ 0 \\ r'_o \sin L_p \end{pmatrix}$$

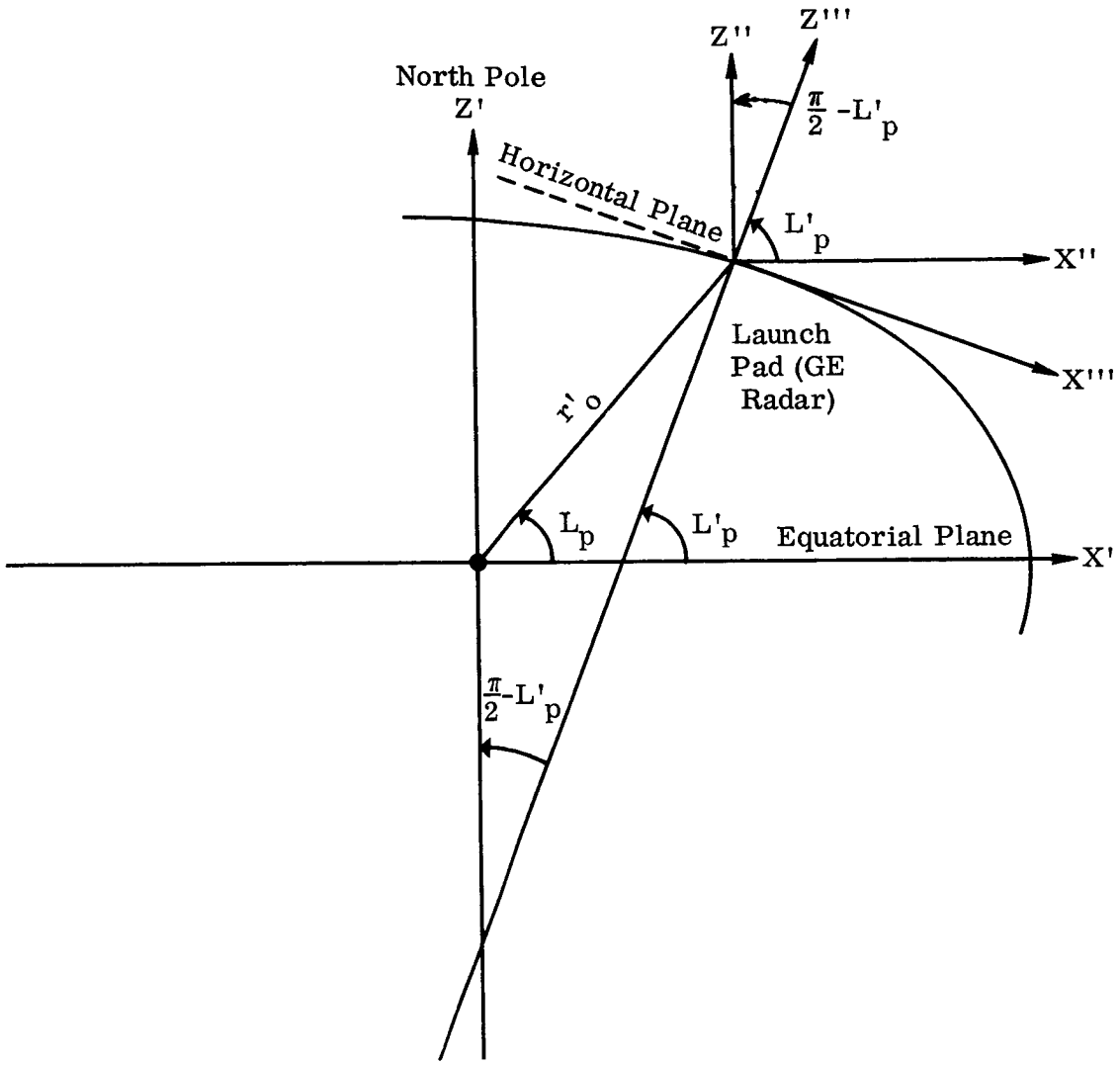
ROTATION OF A RECTANGULAR COORDINATE SYSTEM IN A PLANE, TRANSFORMATION 3

THE THIRD TRANSFORMATION ROTATES THE X'', Z'' AXIS ABOUT THE AXIS SO THAT THE NEW Z''' AXIS COINCIDES WITH THE LOCAL ZENITH AT THE LAUNCH PAD. THE DIAGRAM BELOW ILLUSTRATES THE ROTATION FROM THE OLD (X'',Y'', Z'') SYSTEM TO THE NEW SYSTEM (X''',Y''',Z''').

WHERE -

L'_p = GEODETIC LATITUDE OF THE LAUNCH PAD

IN THE FOLLOWING DIAGRAM, THE AXES Y', Y'', AND Y''' ARE ALL PERPENDICULAR TO THE PLANE OF THE PAGE, DIRECTED DOWNWOARD INTO THE PAGE, AND THEREFORE, APPEAR AS POINTS. SINCE THE (X''',Y''') PLANE COINCIDES WITH THE HORIZON PLANE, A PERSON STANDING AT THE LAUNCH PAD WOULD OBSERVE THE X''' AXIS TO BE DIRECTED DUE SOUTH, THE Y'''AXIS TO BE DIRECTED DUE EAST, AND THE Z''' AXIS TO BE DIRECTED STRAIGHT UP THROUGH THE LOCAL ZENITH AT THE PAD.



THE DIAGRAM SHOWS THE ANGLE THROUGH WHICH Z'' IS ROTATED TO BE $\pi/2-L'_p$ WHICH IS THE GEODETIC COLATITUDE OF THE LAUNCH PAD. HOWEVER, TO ROTATE IN A CLOCKWISE DIRECTION, THE ANGLE MUST BE SET TO $-(\pi/2-L'_p)$ SINCE THE FORMULAS HAVE BEEN CONSTRUCTED FOR A COUNTERCLOCKWISE DIRECTION.

SUBSTITUTING IN EQUATIONS (2), THE FOLLOWING FORMULAS ARE OBTAINED -

$$X''' = X'' \cos\left[-(\pi/2 - L'_p)\right] + Z'' \sin\left[-(\pi/2 - L'_p)\right]$$
$$Z''' = -X'' \sin -\left[(\pi/2 - L'_p)\right] + Z'' \cos\left[-(\pi/2 - L'_p)\right]$$

THESE EQUATIONS CAN BE SIMPLIFIED BY SUBSTITUTING IN THE TRIGNOMETRIC IDENTITIES $\cos(-\theta)=\cos\theta$, $\sin(-\theta)= -\sin\theta$ TO OBTAIN -

$$X''' = X'' \cos(\pi/2 - L'_p) - Z'' \sin(\pi/2 - L'_p)$$
$$Z''' = X'' \sin(\pi/2 - L'_p) + Z'' \cos(\pi/2 - L'_p)$$

THEN BY SUBSTITUTING ACCORDING TO THE IDENTITIES COS $(\pi/2 - \theta) = \sin\theta$ AND SIN $(\pi/2 - \theta) = \cos\theta$ WHERE $0 \leq \theta \leq \frac{\pi}{2}$ —

$$X''' = X'' \sin L'_p - Z'' \cos L'_p$$

$$Z''' = X'' \cos L'_p + Z'' \sin L'_p$$

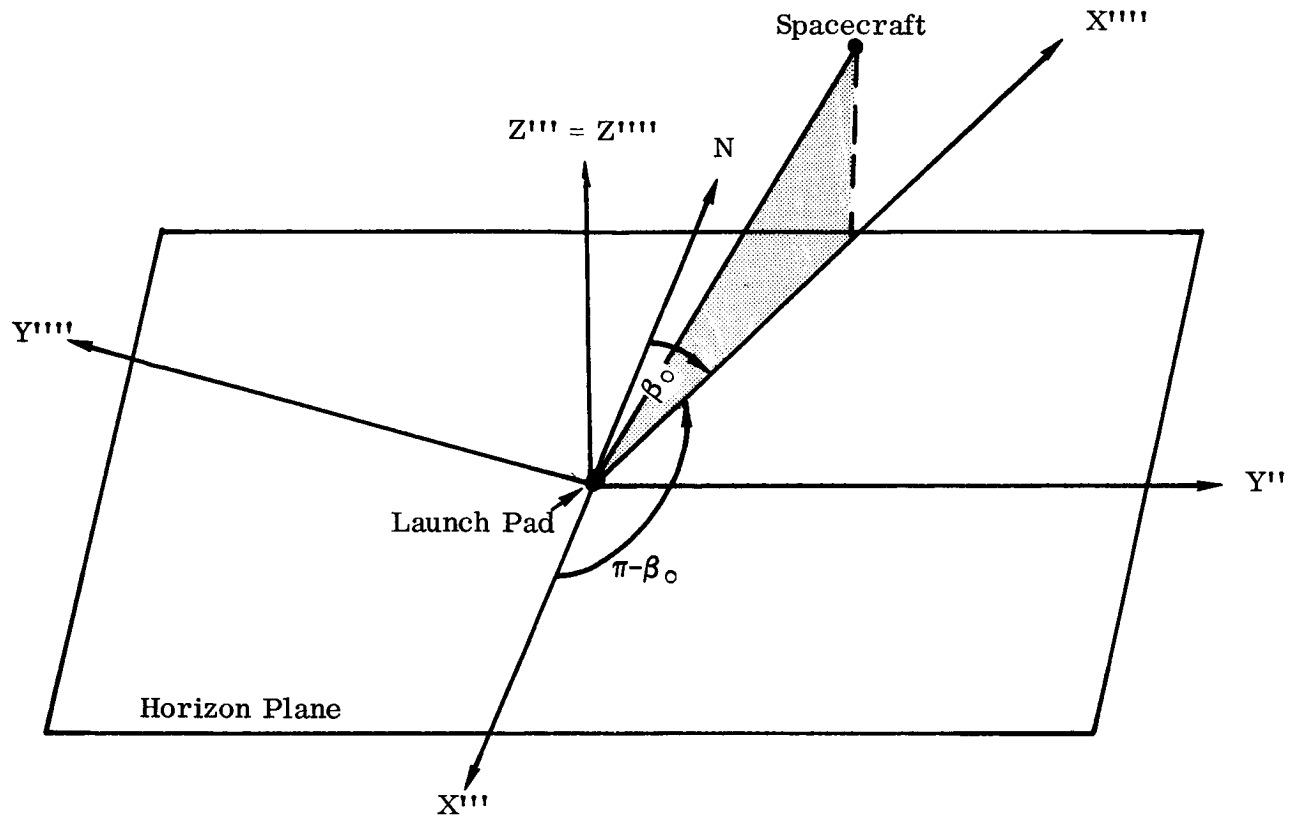
and $Y''' = Y''$ since the system was rotated about Y''

THESE EQUATIONS MAY BE WRITTEN IN MATRIX NOTATION AS FOLLOWS -

$$\begin{pmatrix} X''' \\ Y''' \\ Z''' \end{pmatrix} = \begin{pmatrix} \sin L'_p & 0 & -\cos L'_p \\ 0 & 1 & 0 \\ \cos L'_p & 0 & \sin L'_p \end{pmatrix} \begin{pmatrix} X'' \\ Y'' \\ Z'' \end{pmatrix}$$

ROTATION OF A RECTANGULAR COORDINATE SYSTEM IN A PLANE, TRANSFORMATION 4

THE FOURTH AND FINAL TRANSFORMATION ROTATES THE (X''', Y''') PLANE ABOUT THE Z''' AXIS IN SUCH A MANNER THAT THE NEW AXIS COINCIDES WITH THE BURNOUT AZIMUTH ANGLE β_o. THE DIAGRAM BELOW ILLUSTRATES THE CONCEPT OF AZIMUTH AND THE FINAL COORDINATE SYSTEM - THE LAUNCH PAD COORDINATE SYSTEM.



THE BURNOUT AZIMUTH ANGLE β_o IS DEFINED AS THE NOMINAL AZIMUTH ANGLE AT THE TIME OF SUSTAINER ENGINE CUTOFF. THE ANGLE β_o IS MEASURED IN THE (X''', Y''') PLANE IN A CLOCKWISE DIRECTION FROM NORTH. THEREFORE, THE ANGLE THROUGH WHICH X''' MUST BE ROTATED IS $\pi - \beta_o$. SUBSTITUTING THIS ANGLE INTO THE BASIC EQUATIONS (2) FOR ROTATING A COORDINATE SYSTEM PRODUCES -

$$X'''' = X''' \cos(\pi - \beta_o) + Y''' \sin(\pi - \beta_o)$$

$$Y'''' = -X''' \sin(\pi - \beta_o) + Y''' \cos(\pi - \beta_o).$$

THEN USING THE IDENTITIES COS$(\pi - \Theta)$= - COS Θ AND SIN $(\pi - \Theta)$ = SIN Θ -

$$X'''' = -X''' \cos \beta_o + Y''' \sin \beta_o$$

$$Y'''' = -X''' \sin \beta_o - Y''' \cos \beta_o.$$

REWRITING IN MATRIX NOTATION PRODUCES -

$$\begin{pmatrix} X'''' \\ Y'''' \\ Z'''' \end{pmatrix} = \begin{pmatrix} -\cos \beta_o & \sin \beta_o & 0 \\ -\sin \beta_o & -\cos \beta_o & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X''' \\ Y''' \\ Z''' \end{pmatrix} = \begin{pmatrix} d \\ s \\ w \end{pmatrix}$$

SIMPLIFICATION OF NOTATION

Let: $R_2 = \begin{pmatrix} d \\ s \\ w \end{pmatrix}$

$R_3 = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$

$R_c = \begin{pmatrix} r'_o \cos L_p \\ 0 \\ r'_o \sin L_p \end{pmatrix}$

$A^T = \begin{pmatrix} -\cos \beta_o & \sin \beta_o & 0 \\ -\sin \beta_o & -\cos \beta_o & 0 \\ 0 & 0 & 1 \end{pmatrix}$

$$B^T = \begin{pmatrix} \sin L'_p & 0 & -\cos L'_p \\ 0 & 1 & 0 \\ \cos L'_p & 0 & \sin L'_p \end{pmatrix}$$

$$C^T = \begin{pmatrix} \cos \Omega & \sin \Omega & 0 \\ -\sin \Omega & \cos \Omega & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

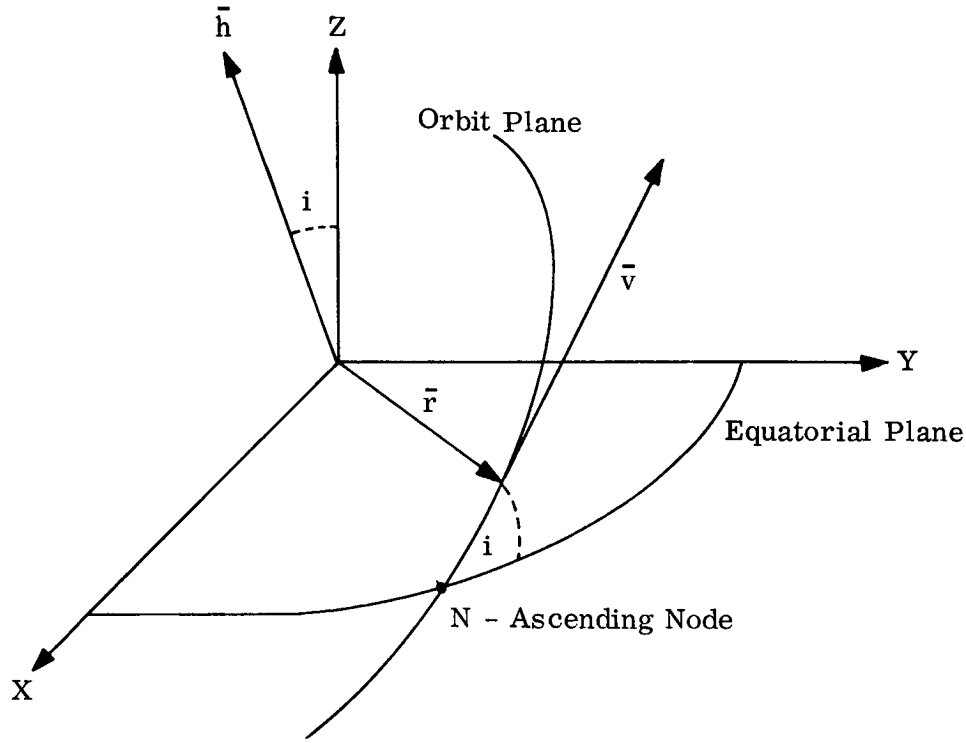
THEN, THE ENTIRE TRANSFORMATION FROM THE GEMINI INERTIAL SYSTEM TO THE LAUNCH PAD COORDINATE SYSTEM MAY BE EXPRESSED BY A SINGLE MATRIX EQUATION -

$$R_2 = A^T B^T (C^T R_3 - R_C).$$

DERIVATION OF INCLINATION ANGLE FORMULA

THE INCLINATION ANGLE, i, IS THE ANGLE FORMED BY THE INTERSECTION OF THE ORBIT PLANE ($\bar{r}, \bar{v}$ PLANE) AND THE EQUATORIAL PLANE (X,Y PLANE). THE POINT OF INTERSECTION OF THE TWO PLANES IS CALLED THE NODE, N. IF THE CAPSULE CROSSES THE EQUATOR TRAVELLING FROM SOUTH TO NORTH, THEN N IS CALLED THE ASCENDING NODE. IF THE CAPSULE CROSSES THE EQUATOR TRAVELLING FROM NORTH TO SOUTH, N IS CALLED THE DESCENDING NODE.

THE VECTORS $\bar{h}$, PERPENDICULAR TO THE ORBIT PLANE, AND THE Z-AXIS, PERPENDICULAR TO THE EQUATORIAL PLANE, INCLUDE AN ANGLE EQUAL TO THE INCLINATION ANGLE.



A) $\bar{h}$ IS DEFINED TO BE THE CROSS PRODUCT OF r AND v, $h = r \times v$

THUS -

$$\bar{h} = \bar{r}\times\bar{v} = \begin{vmatrix} \bar{i} & \bar{j} & \bar{k} \\ X & Y & Z \\ \dot{X} & \dot{Y} & \dot{Z} \end{vmatrix}$$

WHERE $\bar{i}, \bar{j}, \bar{k}$ ARE THE BASE VECTORS, X, Y, Z ARE THE COMPONENTS OF THE POSITION VECTOR AND X, Y, ZARE THE COMPONENTS OF THE VELOCITY VECTOR.

EVALUATING THE ABOVE, WE OBTAIN, IN MATRIX NOTATION

$$\begin{pmatrix} YZ - ZY \\ ZX - XZ \\ XY - YX \end{pmatrix} = \begin{pmatrix} h_x \\ h_y \\ h_z \end{pmatrix}$$

WHERE h_x, h_y, h_z ARE THE COEFFICIENTS OF THE COMPONENTS OF $\bar{h}$ IN THE $\bar{i}, \bar{j}$, AND k DIRECTIONS, RESPECTIVELY.

B) i

THE ANGLE BETWEEN TWO VECTORS IS DEFINED AS FOLLOWS-

$$\cos i = (\bar{h}\cdot\bar{k})/(/\bar{h}//\bar{k}/)$$

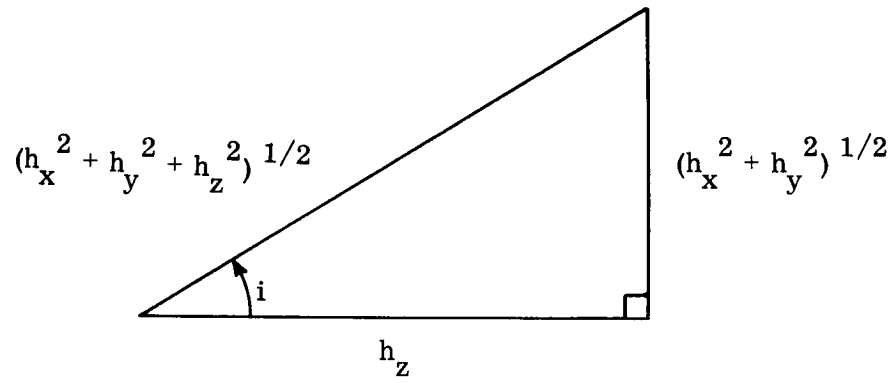
Since $\bar{h} = \begin{pmatrix} h_x \\ h_y \\ h_z \end{pmatrix}$, with $\bar{k} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$, $\bar{h}\cdot\bar{k} = h_z$

APPLYING THE DISTANCE FORMULA ON $|\bar{h}|$ AND WITH $|\bar{k}|$ = 1 -

$$(|\bar{h}|, |\bar{k}|) = (h_x^2 + h_y^2 + h_z^2)^{1/2}$$

$$\text{Thus, } \cos i = h_z / (h_x^2 + h_y^2 + h_z^2)^{1/2}$$

APPLYING THIS RESULT TO THE RIGHT TRIANGLE BELOW, WE FIND THAT tan i ALLOWS FOR USE OF ALL COMPONENTS IN A MORE SIMPLIFIED WAY THEN DID COS i -



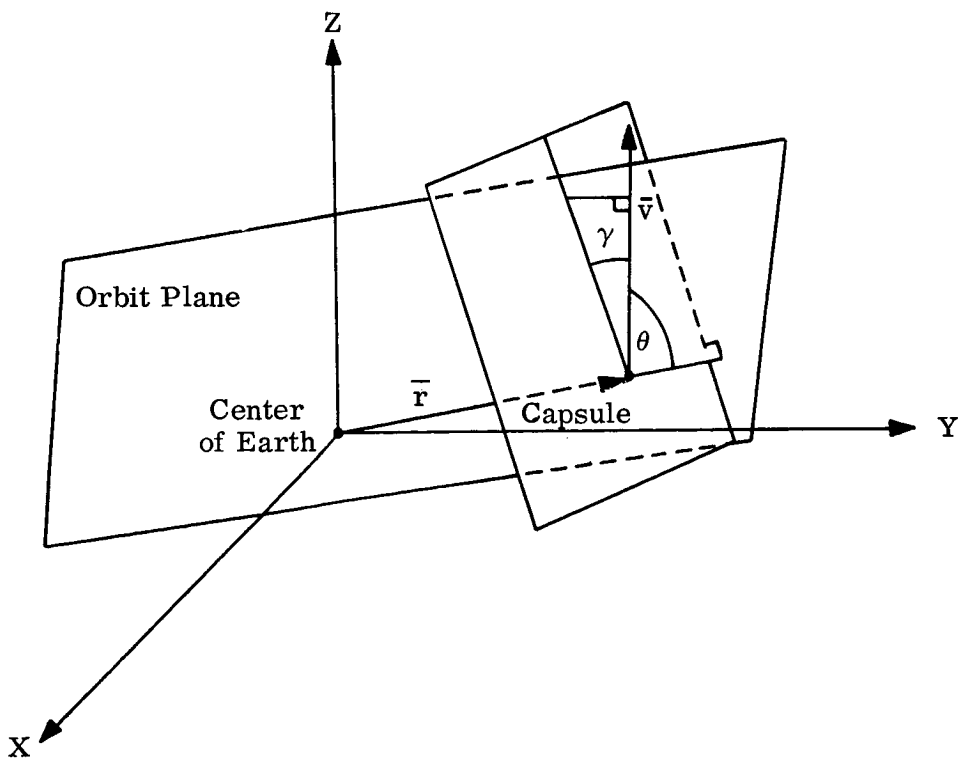
$$\tan i = (h_x^2 + h_y^2)^{1/2} / h_z$$

$$i = \tan^{-1} [(h_x^2 + h_y^2)^{1/2} / h_z]$$

DERIVATION OF FLIGHT PATH ANGLE FORMULA

THE FLIGHT PATH ANGLE, γ, IS DEFINED TO BE THAT ANGLE FORMED BY THE VELOCITY VECTOR, $\bar{v}$ AND ITS PROJECTION ONTO A PLANE PERPENDICULAR TO THE POSITION VECTOR, $\bar{r}$, AND PASSING THROUGH THE CAPSULE.

THE LATTER PLANE DIVIDES THE REFERENCE SPACE INTO TWO PARTS WITH THE CENTER OF THE EARTH (I.E., THE ORIGIN OF THE INERTIAL COORDINATE SYSTEM) ON ONE SIDE OF THE PLANE. IF THE DIRECTION OF $\bar{v}$ IS TOWARD THE SIDE ON WHICH THE ORIGIN LIES, THE SIGN OF γ IS NEGATIVE - OTHERWISE, IT IS POSITIVE.



THE ANGLE BETWEEN TWO VECTORS IS DEFINED BY -

$$\cos\theta = (\bar{r}\cdot\bar{v})/(/\bar{r}//\bar{v}/) \text{ where } 0 \leq \theta \leq \pi \text{ radians}$$

$$\text{Since } (\gamma + \theta) = \pi/2,\ \gamma = ((\pi/2) - \theta) \text{ and}$$

$$\sin\gamma = \sin((\pi/2) - \theta) = \cos\theta$$

$$\text{thus, } \sin\gamma = (\bar{r}\cdot\bar{v})/(/\bar{r}//\bar{v}/) \text{ and, finally,}$$

$$\gamma = \sin^{-1}((\bar{r}\cdot\bar{v})/(/\bar{r}//\bar{v}/))$$

40. CDELTV - LAUNCH PROGRAM TO INCREMENT THE VELOCITY VECTOR BY FIRING RETROROCKETS AS POSIGRADES

GIVEN AN INPUT $t_3, \bar{r}_3, \bar{v}_3$ CDELTV WILL COMPUTE AN INCREMENTED VELOCITY VECTOR BY COMPUTING THE INCREMENT TO THE INPUT VELOCITY VECTOR MAGNITUDE DUE TO FIRING THE RETROROCKETS AS POSIGRADES AND APPROPRIATELY ADJUSTING EACH COMPONENT OF THE GIVEN VELOCITY VECTOR. IT IS USED BEFORE THE OCCURRENCE OF ABORT MODE CHANGEOVER (AMC). (SEE FLOWCHART)

INPUT

A) TABLE -

L $(t_3, \bar{r}_3, \bar{v}_3)$	A SEVEN-LOCATION TABLE CONTAINING TIME IN FLOATING POINT SECONDS, GMT, THE POSITION VECTOR COMPONENTS IN FLOATING POINT GEMINI LENGTH UNITS, AND THE COMPONENTS OF THE VELOCITY VECTOR TO BE INCREMENTED IN FLOATING POINT GEMINI VELOCITY UNITS.

B) CONSTANT -

K001.0	DEC +1.0	FLOATING POINT ONE

OUTPUT

A) TABLE -

L (OUTPUT $t_u, \bar{r}_u, \bar{v}_u$)	A SEVEN-LOCATION TABLE CONTAINING THE TIME IN FLOATING POINT SECONDS, GMT, THE

POSITION VECTOR COMPONENTS IN FLOATING POINT GEMINI LENGTH UNITS, AND THE COMPONENTS OF THE INCREMENTED VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS.

METHOD

UPON ENTRY, CDELTV SAVES THE CONTENTS OF INDEX REGISTERS 1,2,3 AND 4, PICKS UP THE LOCATIONS OF THE INPUT AND OUTPUT TABLES AND SETS UP THE CALLING SEQUENCES FOR A3GSCP (SEE A3GSCP AND U3VMAG).

IT THEN ENTERS PROGRAM A3GSCP TO FIND H, THE HEIGHT OF THE INPUT VECTORS ABOVE AN OBLATE EARTH. IF THERE IS AN ERROR RETURN FROM THIS PROGRAM, IT IS ASSUMED THAT NO VELOCITY INCREMENT CAN BE COMPUTED, THE INPUT $t_3, \bar{r}_3, \bar{v}_3$ ARE STORED IN THE OUTPUT TABLE, THE INDEX REGISTERS ARE RESTORED, AND AN EXIT IS MADE FROM THE PROGRAM.

IF THERE IS A NORMAL RETURN FROM A3GSCP, TESTS ARE MADE ON H TO DETERMINE WHICH OF FOUR, THIRD-ORDER CURVE FITS WILL BE USED TO COMPUTE THE INCREMENT TO THE INPUT VELOCITY VECTOR.

A) IF $H \geq 505{,}000$ FT., $\Delta v = 339$ FT./SEC.

B) IF $355{,}000$ FT $\leq H < 505{,}000$ FT., CURVE 4 IS USED

C) IF $245{,}000$ FT $\leq H < 355{,}000$ FT., CURVE 3 IS USED

D) IF $100{,}000$ FT $\leq H < 245{,}000$ FT., CURVE 2 IS USED

E) IF $80{,}000$ FT $\leq H < 100{,}000$ FT., CURVE 1 IS USED

F) IF $H < 80{,}000$ FT., $\Delta v = 0$

IF Δv IS TO BE DETERMINED BY A CURVE FIT, I.E., IF Δv IS NOT COMPUTED TO BE EITHER 0 OR 339 FT./SEC. BY THE ABOVE TESTS ON H, THE VELOCITY VECTOR INCREMENT Δv IS

DETERMINED AS FOLLOWS -

$$\Delta v = a_i H^3 + b_i H^2 + c_i H + d_i$$

WHERE

a_i, b_i, c_i AND d_i ARE THE COEFFICIENTS FOR THE CURVE SELECTED, $i = 1,2,3$ OR 4. IF $i = 1$, COEFFICIENTS FOR CURVE 1 ARE USED, IF $i = 2$, COEFFICEENTS FOR CURVE 2 ARE USED, ETC.

USING A LINEAR APPROXIMATION, THE COMPUTED Δv IS ADDED TO THE COMPONENTS OF THE INPUT VELOCITY VECTOR.

$$\bar{v}_u = (1 + \frac{\Delta v}{|v_3|}) \bar{v}_3$$

WHERE

$|v_3|$ IS THE MAGNITUDE OF THE INPUT VELOCITY VECTOR (x_3, y_3, z_3) COMPUTED USING U3VMAG, AND $\bar{v}_u$ REPRESENTS THE INCREMENTED VELOCITY VECTOR (x_u, y_u, z_u). THE INPUT TIME AND POSITION VECTOR COMPONENTS ARE STORED IN THE OUTPUT TABLE AS $t_u, \bar{r}_u$ AND THE FINAL INCREMENTED VELOCITY, $\bar{v}_u$ IS STORED IN THE OUTPUT TABLE. INDEX REGISTERS ARE RESTORED AND CONTROL IS TRANSFERRED TO THE CALLING PROGRAM.

THE COEFFICIENTS FOR THE FOUR CURVES WERE DERIVED USING LEAST-SQUARES POLYNOMIAL CURVE FITS.

(LEAST SQUARES POLYNOMIAL FIT, CLLSQ1, R HARRISON, SHARE DISTRIBUTION #1817.)

IN EACH CASE, A THIRD-ORDER CURVE WAS PASSED THROUGH A PORTION OF THE GIVEN DATA. THE DISTRIBUTION OF POINTS WAS CHOSEN SO AS TO ALLOW THE BEST DATA FIT POSSIBLE USING THIRD-ORDER CURVES. FIGURE 1 IS A PLOT OF THE GIVEN DATA, H VS Δv.

FIGURE 2 SHOWS THE FOUR CURVES REPRESENTED BY THE FOUR SETS OF COEFFICIENTS AND THE STANDARD DEVIATION, σ, OF EACH CURVE FROM THE GIVEN DATA. FOLLOWING IS A LIST OF THE DATA DISTRIBUTION USED TO GENERATE EACH OF THE FOUR CURVES -

CURVE 1	80,000 FT $\leq$ H $\leq$ 110,000 FT
CURVE 2	98,000 FT $\leq$ H $\leq$ 248,000 FT
CURVE 3	240,000 FT $\leq$ H $\leq$ 360,000 FT
CURVE 4	350,000 FT $\leq$ H $\leq$ 505,000 FT

THE COEFFICIENTS DERIVED ARE AS FOLLOWS -

a_1 =	.46064813E7	a_3 =	.92621658E4
b_1 =	-.66794214E5	b_3 =	-.29558097E3
c_1 =	.32347721E3	c_3 =	.19377614E1
d_1 =	-.50509483	d_3 =	.29251129E-1
a_2 =	.80470117E4	a_4 =	-.47658149E-1
b_2 =	-.34884610E3	b_4 =	.38501305E2
c_2 =	.40552077E1	c_4 =	-.10016800E1
d_2 =	.61213470E-2	d_4 =	.21507380E-1

USAGE

A) CALLING SEQUENCE -

```
TSX CDELTV,4
PZE L (INPUT t3, r̄3, v̄3),,
L (OUTPUT tu, ru, vu)

RETURN
```

B) STORAGE REQUIRED - 92 LOCATIONS

C) TIMING - 2,586 MILLISECONDS

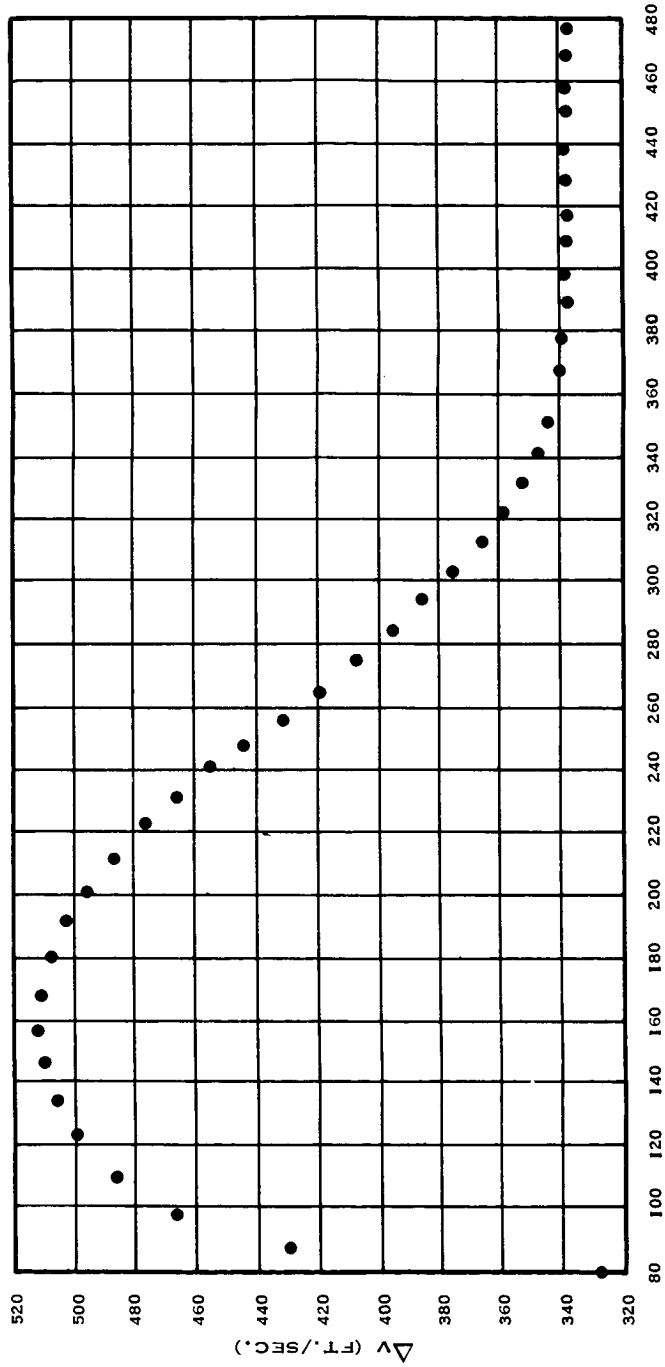


Figure 1. H vs True Δv

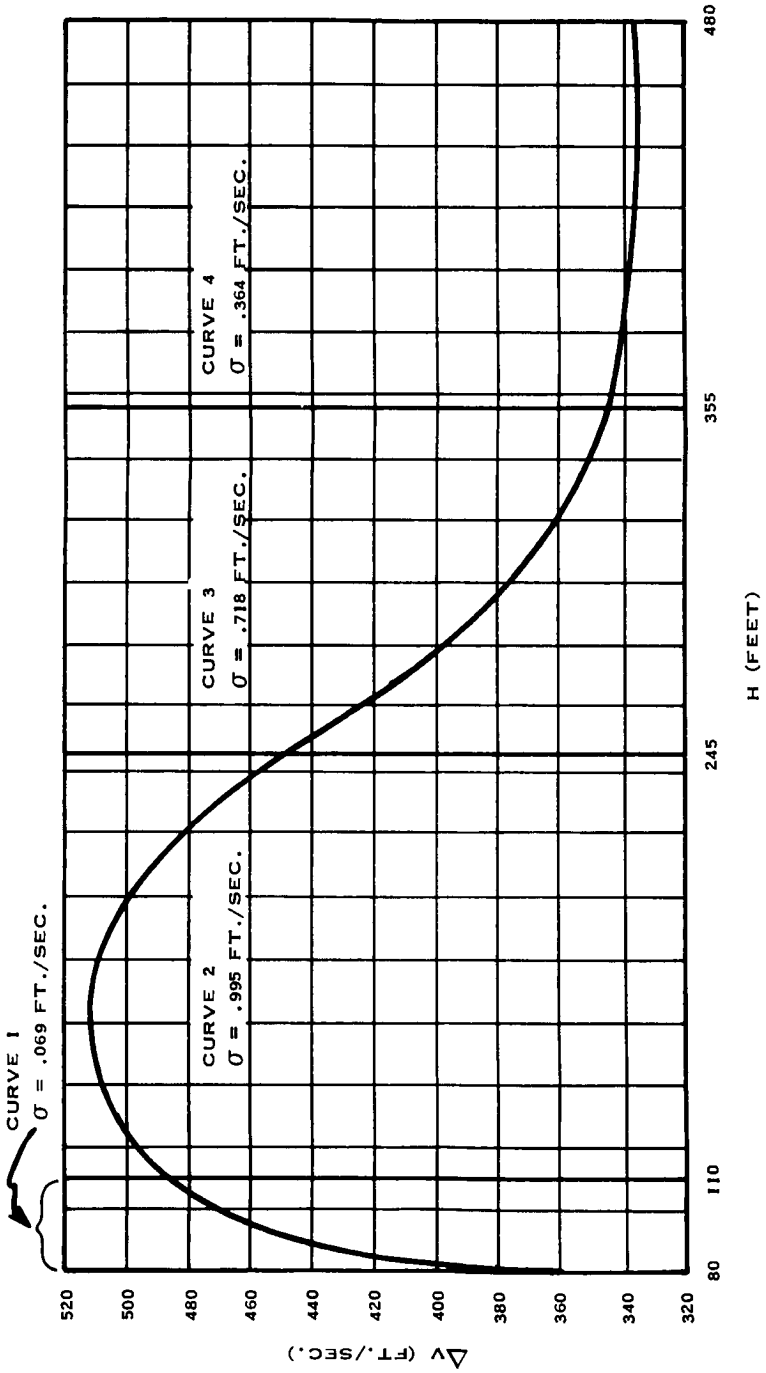
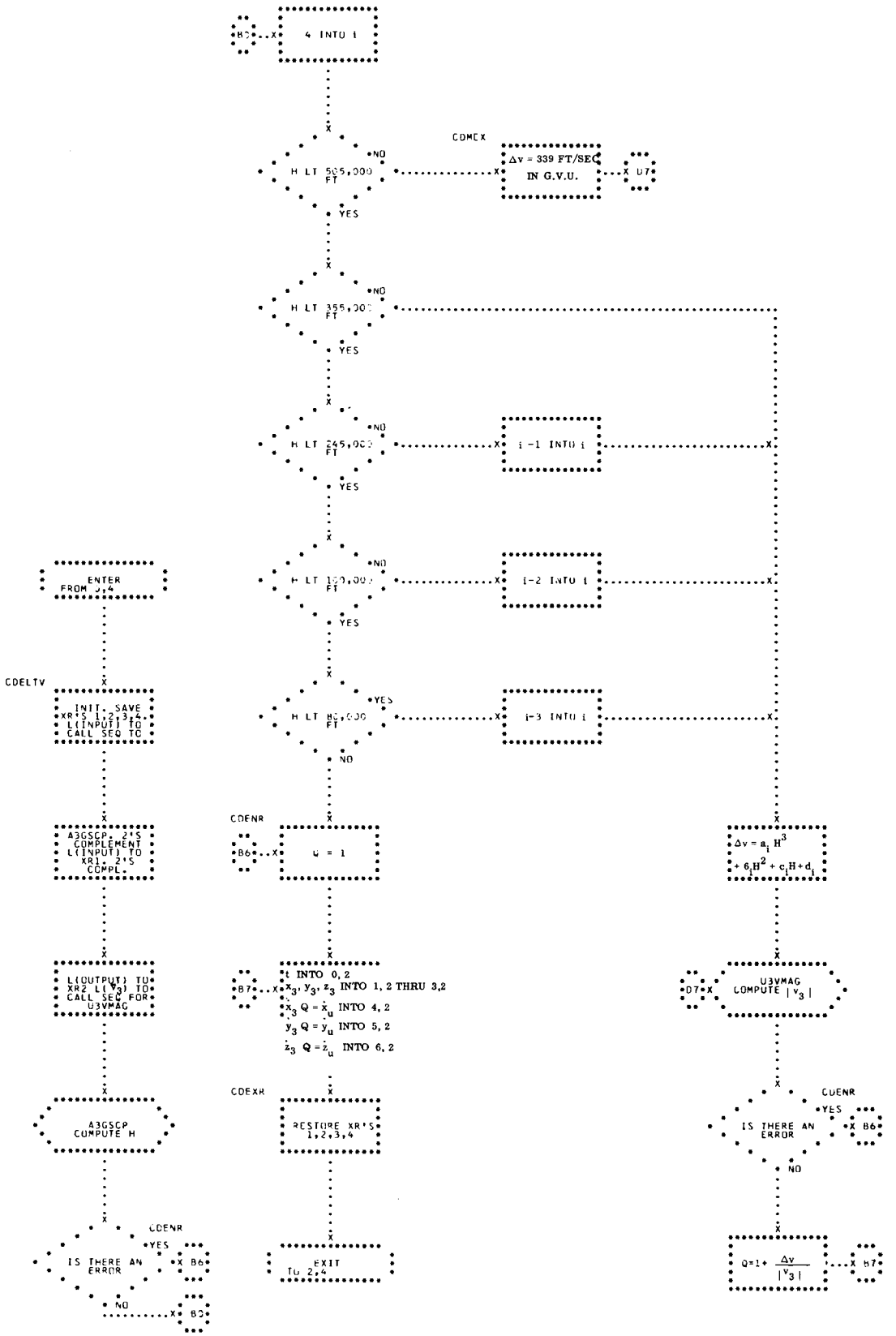


Figure 2. H vs Computed Δv


B5
4 INTO i
CDMEX
H LT 505,000 FT
NO
YES
Δv = 339 FT/SEC IN G.V.U.
B7
H LT 355,000 FT
H LT 245,000 FT
i -1 INTO i
H LT 100,000 FT
i-2 INTO i
H LT 80,000 FT
YES
i-3 INTO i
NO
ENTER FROM J,4
CDELTV
INIT. SAVE XR'S 1,2,3,4. L(INPUT) TO CALL SEQ TO
A3GSCP, 2'S COMPLEMENT L(INPUT) TO XR1, 2'S COMPL.
L(OUTPUT) TO XR2 L(V3) TO CALL SEQ FOR U3VMAG
A3GSCP COMPUTE H
CDENR
IS THERE AN ERROR
YES
B6
NO
B3
CDENR
Q = 1
t INTO 0, 2
x3, y3, z3 INTO 1, 2 THRU 3,2
x3 Q = xu INTO 4, 2
y3 Q = yu INTO 5, 2
z3 Q = zu INTO 6, 2
CDEXR
RESTORE XR'S 1,2,3,4
EXIT TO 2,4
Δv = ai H3 + biH2 + ciH + di
U3VMAG COMPUTE |V3|
CDENR
IS THERE AN ERROR
YES
B6
NO
Q=1+ Δv / |V3|
B7

41. CBIPRE - IMPACT PREDICTOR FOR AVERAGE VECTOR COMPUTATION

THIS PROGRAM COMPUTES LATITUDE AND LONGITUDE OF BOTH MINIMUM AND MAXIMUM DELAY IMPACT DURING AVERAGE VECTOR COMPUTATIONS. IT ALSO COMPUTES THE MINIMUM AND MAXIMUM DELAY RETROFIRE TIMES. IT IS A SUBROUTINE OF THE AVERAGE VECTOR PROGRAM CFCFAV. (SEE FLOWCHART)

INPUT

A) CALLING SEQUENCE REQUIREMENTS -

L(INPUT) - THE ADDRESS OF A SEVEN-LOCATION TABLE CONTAINING t IN FLOATING POINT SECONDS AND THE COMPONENTS OF $\bar{r}$ AND $\bar{v}$ IN FLOATING POINT GEMINI UNITS

L($\Delta t_{r\text{-}min}$) - THE ADDRESS OF A CELL CONTAINING THE MINIMUM DELAY TIME TO RETROFIRE IN FLOATING POINT SECONDS

L(BANK ANGLE) - THE ADDRESS OF A CELL CONTAINING THE BANK ANGLE TO BE USED IN REENTRY IN FLOATING POINT RADIANS, AND WHOSE RANGE IS SUCH THAT $+0 \leq$ BANK ANGLE $\leq 2\pi$. A NEGATIVE ANGLE REPRESENTS A CONSTANT ROLL RATE AND HENCE, NO LIFT.

B) SUBROUTINES -

CORKNI, C6RETC, C6PTLU, C6PTLI, CTSFT3

C) CONSTANTS -

KM0000	MINUS ZERO
K00000	ZERO
K00001	FIXED POINT ONE
K00002	FIXED POINT TWO
K00003	FIXED POINT THREE

K00006	FIXED POINT SIX
K00030	FIXED POINT THIRTY
K00050	FIXED POINT FIFTY
K006.0	FLOATING POINT SIX
K045.0	FLOATING POINT FORTY-FIVE
K00.PI	FLOATING POINT VALUE OF PI (π)

D) TABLES -

TCMXCH	CONTAINS THE CRITERION HEIGHT FOR MAXIMUM DELAY IMPACT IN FLOATING POINT GEMINI UNITS
TCWF.B	CONTAINS THE ORBIT WEIGHT OF THE SPACECRAFT IN FLOATING POINT POUNDS
TCWF.R	CONTAINS THE REENTRY WEIGHT OF THE SPACECRAFT IN FLOATING POINT POUNDS
TCAHC2	CONTAINS THE NOMINAL IMPACT HEIGHT FOR AVERAGE VECTORS AND ABORT COMPUTATIONS IN FLOATING POINT GEMINI UNITS
TCRFTQ	CONTAINS TORQUE SETTING FOR C6RETC =0 INTERTIAL PLATFORM OF SPACECRAFT NOT TORQUED DURING RETROFIRE ≠0 INERTIAL PLATFORM OF SPACECRAFT IS TORQUED DURING RETROFIRE

OUTPUT

A) A SIX-LOCATION TABLE CONTAINING THE LATITUDE AND LONGITUDE OF IMPACT ASSOCIATED WITH FIRING THE RETROS WITH MINIMUM DELAY, THE LATITUDE AND LONGITUDE OF IMPACT ASSOCIATED WITH FIRING THE RETROS WITH MAXIMUM DELAY, THE MINIMUM DELAY

FIRING TIME, AND THE MAXIMUM DELAY FIRING TIME, RESPECTIVELY, IS OUTPUT FROM THE PROGRAM. LATITUDE AND LONGITUDE ARE IN FLOATING POINT RADIANS, AND TIME IS IN FLOATING POINT SECONDS.

B) TABLES -

TCMNRV	CONTAINS THE t, $\bar{r}$, $\bar{v}$ AT THE BEGINNING OF MINIMUM DELAY RETROFIRE IN FLOATING POINT SECONDS AND FLOATING POINT GEMINI UNITS
TCBMN5	CONTAINS THE t, $\bar{r}$, $\bar{v}$ AT THE END OF MINIMUM DELAY RETROFIRE IN FLOATING POINT SECONDS AND FLOATING POINT GEMINI UNITS
TCBMX5	CONTAINS THE t, $\bar{r}$, $\bar{v}$ AT THE END OF MAXIMUM DELAY RETROFIRE IN FLOATING POINT SECONDS AND FLOATING POINT GEMINI UNITS
TRKO1H	CONTAINS THE STEP SIZE FOR USE IN TYPE 3 INTEGRATION IN FLOATING POINT SECONDS
TCRKOT	CONTAINS ORBIT TABLE OF $\bar{r}$ AND $\bar{v}$ VECTORS GENERATED BY TYPE 3 INTEGRATION IN FLOATING POINT GEMINI UNITS

METHOD

AFTER INITIALIZATION THE INPUT VECTOR IS INTEGRATED USING CORKNI, TYPE 2, WITH SIX STEPS EQUAL TO $(\Delta t_{r\text{-}min}/6)$ SECONDS. THIS RESULTS IN A VECTOR $t_{4\ min}, \bar{r}_{4\ min}$ $\bar{v}_{min}$ AT MINIMUM DELAY RETROFIRE TIME. C6RETC USES THIS VECTOR TO COMPUTE THE BURNOUT VECTOR $t_{5\ min}, \bar{r}_{5\ min}, \bar{v}_{5\ min}$ WHICH IS INTEGRATED TO IMPACT BY CORKNI, TYPE 1. THE BANK

ANGLE FOR THIS REENTRY INTEGRATION IS SPECIFIED IN THE CALLING SEQUENCE TO CBIPRE.

WITH $t_4, \bar{r}_4, \bar{v}_4$ AS INPUT, CORKNI, TYPE 3, GENERATES AN ORBIT TABLE OF $\bar{r}$ AND $\bar{v}$ VECTORS AND A TABLE OF HEIGHTS (H) CORRESPONDING TO THE VECTOR TABLE. THE INTERPOLATION PROGRAMS C6PTLU AND C6PTLI USE THE HEIGHT TABLE AND THE $\bar{r}$ v TABLE TO COMPUTE THE VECTOR ASSOCIATED WITH FIRING THE RETROS WITH MAXIMUM DELAY. THIS VECTOR IS $t_{4\ max}, \bar{r}_{4\ max}, \bar{v}_{4\ max}$. AGAIN C6RETC USES THIS VECTOR TO COMPUTE THE BURNOUT VECTOR $t_{5\ max}, \bar{r}_{5\ max}, \bar{v}_{5\ max}$ WHICH IS IN TURN INTEGRATED TO IMPACT BY CORKNI, TYPE 1. THE BANK ANGLE TO BE USED IN REENTRY INTEGRATION IS SPECIFIED IN THE CALLING SEQUENCE TO CBIPRE.

THERE ARE SEVERAL EXCEPTIONS TO THIS GENERAL FLOW OF LOGIC -

A) IF THE TRAJECTORY NEVER REACHES THE CRITERION HEIGHT CONTAINED IN LOCATION TCMXCH, CORKNI, TYPE 3, GIVES AN INDICATION OF THIS UPON EXIT FROM TYPE 3. THUS, A MAXIMUM IMPACT IS NOT POSSIBLE AND THE OUTPUT CELLS FOR MAXIMUM DELAY IMPACT ARE CORRESPONDINGLY SET EQUAL TO THOSE OF MINIMUM DELAY IMPACT.

B) IF THE TRAJECTORY IS SUCH THAT CORKNI, TYPE 3, CANNOT GENERATE THREE VECTORS FROM THE DESCENDING PORTION OF THE TRAJECTORY BELOW THE CRITERION HEIGHT CONTAINED IN LOCATION TCMXCH, MAXIMUM IMPACT IS UNDEFINED, AND THE OUTPUT CELLS FOR MAXIMUM DELAY IMPACT ARE APPROPRIATELY SET OFF-SCALE TO THE RIGHT.

C) IF $t_{4\ max} \leq t_{4\ min}$, A MAXIMUM DELAY IMPACT IS IMPOSSIBLE AND THE OUTPUT PROCEDURES ARE THE SAME AS IN A).

USAGE

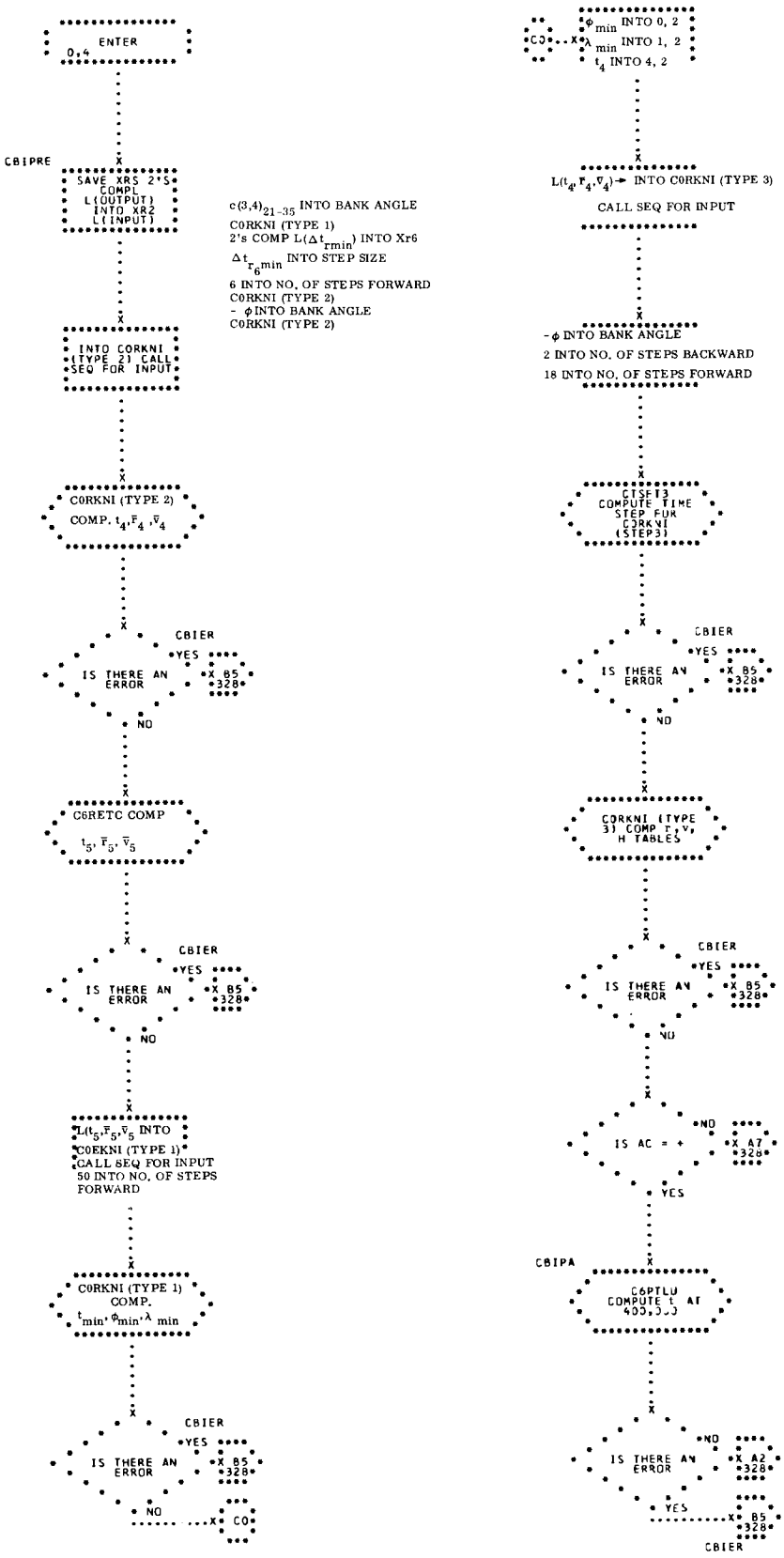
A) CALLING SEQUENCE -

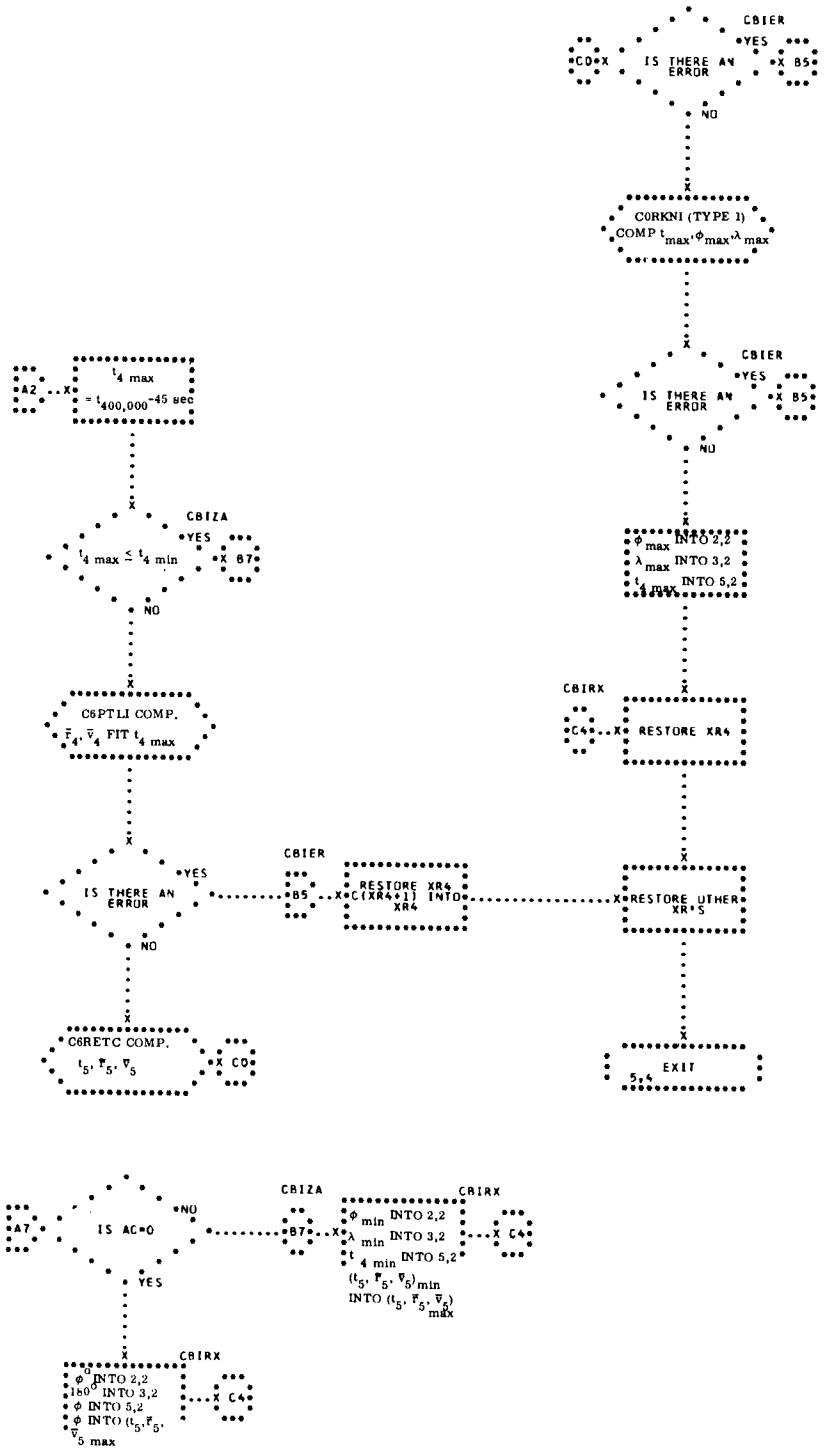
```
TSX       CBIPRE,4
PZE       L(INPUT),,L(OUTPUT)
PZE       **,,(Δt r-min)
PZE       L(BANK ANGLE),,**
ERROR     RETURN
NORMAL    RETURN
```

B) STORAGE REQUIRED - 113 LOCATIONS EXCLUDING THAT REQUIRED BY EXTERNAL CONSTANTS, TABLES, AND SUBROUTINES

C) TIME REQUIRED (7094) - CBIPRE COULD POSSIBLY TAKE A MAXIMUM OF 1400 MILLISECONDS IF CORKNI WOULD TAKE THE MAXIMUM NUMBER OF INTEGRATION STEPS SPECIFIED IN EVERY CASE. HOWEVER, A COMPUTED MAXIMUM MACHINE TIME IS 600 MILLISECONDS FOR A VECTOR USING A 60° BANK ANGLE DURING REENTRY AND WHERE MAXIMUM DELAY IMPACT IS SET OFF-SCALE TO THE RIGHT. FOR A VECTOR WHERE BOTH MINIMUM AND MAXIMUM DELAY IMPACTS ARE DEFINED AND WHERE A 60° BANK ANGLE IS USED DURING REENTRY, A COMPUTED MAXIMUM MACHINE TIME IS 625 MILLISECONDS.

D) ERROR CONDITIONS - AN ERROR RETURN IS A RESULT OF AN ERROR RETURN FROM ONE OF THE SUBROUTINES.


ENTER 0,4
CBIPRE
SAVE XRS 2*S COMPL L(OUTPUT) INTO XR2 L(INPUT)
INTO CORKNI (TYPE 2) CALL SEQ FOR INPUT
CORKNI (TYPE 2) COMP. $t_4, \bar{r}_4, \bar{v}_4$
CBIER
IS THERE AN ERROR
YES
X 85 *328*
NO
C6RETC COMP $t_5, \bar{r}_5, \bar{v}_5$
CBIER
IS THERE AN ERROR
YES
X 85 *328*
NO
$L(t_5, \bar{r}_5, \bar{v}_5$ INTO C0EKNI (TYPE 1) CALL SEQ FOR INPUT 50 INTO NO. OF STEPS FORWARD
CORKNI (TYPE 1) COMP. $t_{min}, \phi_{min}, \lambda_{min}$
CBIER
IS THERE AN ERROR
YES
X 85 *328*
NO
X CO
$c(3,4)_{21-35}$ INTO BANK ANGLE
CORKNI (TYPE 1)
2's COMP $L(\Delta t_{r_{min}})$ INTO Xr6
$\Delta t_{r_{6min}}$ INTO STEP SIZE
6 INTO NO. OF STEPS FORWARD
CORKNI (TYPE 2)
$-\phi$ INTO BANK ANGLE
CORKNI (TYPE 2)
CO
X
ϕ_{min} INTO 0, 2
λ_{min} INTO 1, 2
t_4 INTO 4, 2
$L(t_4, \bar{r}_4, \bar{v}_4)$ INTO CORKNI (TYPE 3) CALL SEQ FOR INPUT
$-\phi$ INTO BANK ANGLE
2 INTO NO. OF STEPS BACKWARD
18 INTO NO. OF STEPS FORWARD
CTSET3 COMPUTE TIME STEP FOR CORKNI (STEP3)
CBIER
IS THERE AN ERROR
YES
X 85 *328*
NO
CORKNI (TYPE 3) COMP $\bar{r}, \bar{v}$, H TABLES
CBIER
IS THERE AN ERROR
YES
X 85 *328*
NO
IS AC = +
NO
X A7 *328*
YES
CBIPA
C6PTLU COMPUTE t AT 400,000
IS THERE AN ERROR
NO
X A2 *328*
YES
X 85 *328*
CBIER


CBIER
YES
CO
IS THERE AN ERROR
B5
NO
CORKNI (TYPE 1)
COMP $t_{max}, \phi_{max}, \lambda_{max}$
CBIER
YES
IS THERE AN ERROR
B5
NO
ϕ_{max} INTO 2,2
λ_{max} INTO 3,2
$t_{4\,max}$ INTO 5,2
CBIRX
C4
RESTORE XR4
RESTORE OTHER XR'S
EXIT
5,4
A2
$t_{4\,max} = t_{400,000}$-45 sec
CBIZA
YES
$t_{4\,max} \leq t_{4\,min}$
B7
NO
C6PTLI COMP.
$\bar{r}_4, \bar{v}_4$ FIT $t_{4\,max}$
CBIER
YES
IS THERE AN ERROR
B5
RESTORE XR4
C(XR4+1) INTO XR4
NO
C6RETC COMP.
$t_5, \bar{r}_5, \bar{v}_5$
CO
A7
IS AC=0
NO
YES
CBIZA
B7
CBIRX
ϕ_{min} INTO 2,2
λ_{min} INTO 3,2
$t_{4\,min}$ INTO 5,2
$(t_5, \bar{r}_5, \bar{v}_5)_{min}$
INTO $(t_5, \bar{r}_5, \bar{v}_5)_{max}$
C4
CBIRX
ϕ^o INTO 2,2
180° INTO 3,2
ϕ INTO 5,2
ϕ INTO $(t_5, \bar{r}_5, \bar{v}_{5\,max}$
C4

42. C6RETC - LAUNCH RETROFIRE PROGRAM

THE LAUNCH RETROFIRE PROGRAM, C6RETC, IS USED TO COMPUTE THE TIME, POSITION AND VELOCITY VECTORS AT THE END OF RETROFIRE GIVEN THE TIME, POSITION AND VELOCITY VECTORS AT THE BEGINNING OF RETROFIRE, THE NUMBER OF RETROS FIRED, THE FIRING TIMES OF THE INDIVIDUAL ROCKETS, AND THE FIRING ATTITUDE.

INPUT

A) THE INPUT $t_o, \bar{r}_o, \bar{v}_o$ ARE THE VECTORS AT THE INITIATION OF RETROFIRING. THEY ARE IN FLOATING POINT IN A SEVEN LOCATION TABLE WHICH HAS THE FOLLOWING FORMAT -

L(	INPUT $t_o, \bar{r}_o, \bar{v}_o$	t	TIME IN GMT SECONDS
	+1	x	$\bar{r}_o$, POSITION VECTOR IN GEMINI LENGTH UNITS
	+2	y	
	+3	z	
	+4	$\dot{x}$	$\bar{v}_o$, VELOCITY VECTOR IN GEMINI VELOCITY UNITS
	+5	$\dot{y}$	
	+6	$\dot{z}$	

B) CONSTANTS -

K00000	DEC	0	ZERO
KM00.5	DEC	-.5	FLOATING POINT MINUS

K001.0	DEC	1.0	ONE-HALF FLOATING POINT ONE
K002.0	DEC	2.0	FLOATING POINT TWO
K022.0	DEC	22.0	FLOATING POINT TWENTY-TWO
K00.UT	DEC	.806813464E3	1 GEMINI TIME UNIT, IN SECONDS
KCH233	OCT	233000000000	OCTAL MASK
KM7777	OCT	377777777777	OCTAL MASK

C) PARAMETERS -

TC6RCW	WEIGHT OF SPACECRAFT IN FLOATING POINT POUNDS

D) SUBROUTINES -

UISICO	FLOATING POINT SINE, COSINE SUB-ROUTINE.
UISQRT	FLOATING POINT SQUARE ROOT SUB-ROUTINE.

E) L(TYPE OF FIRING INDICATOR) THIS LOCATION CONTAINS ZERO FOR A NOMINAL RETRO-FIRING SEQUENCE AND A

NON-ZERO QUANTITY FOR A NON-NOMINAL RETRO-FIRING SEQUENCE.

F) L(TYPE TORQUE SETTING) THIS LOCATION CONTAINS ZERO IF THE INERTIAL PLATFORM IS NOT TORQUED DURING THE RETRO BURNING PERIOD AND A NON-ZERO QUANTITY IF IT IS TORQUED.

G) FOR THE NON-NOMINAL CASE, A NON-NOMINAL INPUT TABLE MUST BE DEFINED FOR THE PROGRAM. THIS TABLE HAS THE FOLLOWING FORMAT -
L(NON-NOMINAL INPUT TABLE)
(IF ANY OF THE FOUR ROCKETS DOES NOT FIRE, ITS ASSOCIATED TIME MUST BE SET TO ZERO.)

n_R NUMBER OF RETRO ROCKETS FIRED, FIXED POINT B35

+1	t_1	TIME OF FIRING OF RETRO NUMBER 1 IN FLOATING POINT SECONDS
+2	t_3	TIME OF FIRING OF RETRO NUMBER 3 IN FLOATING POINT SECONDS
+3	t_2	TIME OF FIRING OF RETRO NUMBER 2 IN FLOATING POINT SECONDS
+4	t_4	TIME OF FIRING OF RETRO NUMBER 4 IN FLOATING POINT SECONDS

	+5	ROLL ANGLE AT THE BEGINNING OF RETROFIRE IN FLOATING POINT RADIANS
	+6	PITCH ANGLE AT THE BEGINNING OF RETROFIRE IN FLOATING POINT RADIANS
	+7	YAW ANGLE AT THE BEGINNING OF RETROFIRE IN FLOATING POINT RADIANS

OUTPUT

A) OUTPUT VECTORS, $t, \bar{r}, \bar{v}$ ARE IN FLOATING POINT IN A SEVEN LOCATION TABLE HAVING THE FOLLOWING FORMAT -

L(OUTPUT VECTORS)	t	BURNOUT TIME IN GMT SECONDS
+1	x	$\bar{r}$, POSITION VECTOR IN GEMINI LENGTH UNITS
+2	y	
+3	z	

+4	$\dot{x}$	$\bar{V}$, VELOCITY VECTOR IN GEMINI
+5	$\dot{y}$	VELOCITY UNITS
+6	$\dot{z}$	

FORMULATION

IN ORDER TO DERIVE THE EQUATIONS FOR COMPUTING THE TIME AND POSITION AND VELOCITY VECTORS AT THE END OF RETROFIRING, WE MUST FIRST ESTABLISH A REFERENCE COORDINATE SYSTEM. SEVERAL COORDINATE SYSTEMS HAVE BEEN ADAPTED FOR ESTABLISHING THE CAPSULE ATTITUDE AT RETROFIRE TIME. THREE SUCH SYSTEMS ARE DESCRIBED BELOW, INCLUDING THAT UPON WHICH THE DERIVATION TO FOLLOW WILL BE BASED.

CONSIDER A RIGHT-HANDED, ORTHOGONAL, INERTIAL SYSTEM, x_p, y_p, z_p, (THIS SYSTEM HAS BEEN CALLED THE INERTIAL PLATFORM SYSTEM BY GSFC PERSONNEL AND IS REFERRED TO AS THE COMPUTER SYSTEM BY SOME OF THE MSC, HOUSTON, PERSONNEL.) AT THE BEGINNING OF RETROFIRE, THE y_p AXIS POINTS (POSITIVELY) FROM THE SPACECRAFT CENTER OF GRAVITY TOWARD THE CENTER OF THE EARTH. THE z_p AXIS IS PERPENDICULAR TO y_p AND THE SPACECRAFT INERTIAL VELOCITY VECTOR AND POINTS POSITIVELY IN THE DIRECTION A RIGHT-HANDED SCREW WOULD MOVE WHEN THE SPACECRAFT INERTIAL VELOCITY VECTOR IS ROTATED INTOy_p. THE x_p AXIS IS PERPENDICULAR TO y_p AND z_p THE SPACECRAFT IS ALIGNED SO THAT ITS SMALL END POINTS IN THE DIRECTION OF FLIGHT. (FIGURE 1)

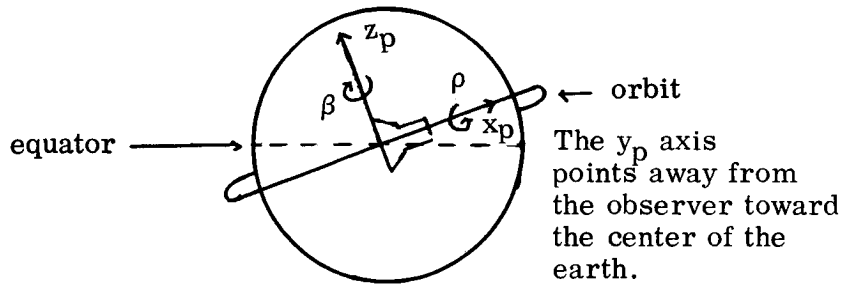


FIGURE 1

WITH REFERENCE TO THIS SYSTEM, THE QUANTITIES PITCH, YAW AND ROLL ARE DEFINED AS FOLLOWS. YAW IS ALWAYS WITHIN THE LIMITS OF $\pm 180^{o}$ AND IS A POSITIVE ROTATION COUNTERCLOCKWISE ABOUT y_p LOOKING FROM THE POSITIVE DIRECTION OF THE y_p AXIS TOWARD THE CENTER OF GRAVITY OF THE SPACECRAFT. PITCH IS ALWAYS WITHIN THE LIMITS OF $\pm 90^{o}$ AND IS A POSITIVE ROTATION CLOCKWISE ABOUT z_p LOOKING FROM THE POSITIVE DIRECTION OF THE z_p AXIS TOWARD THE CENTER OF THE SPACECRAFT. ROLL IS ALWAYS WITHIN THE LIMITS OF $\pm 180^{o}$ AND IS A POSITIVE ROTATION COUNTERCLOCKWISE ABOUT x_p LOOKING FROM THE POSITIVE DIRECTION OF THE AXIS TOWARD THE CENTER OF GRAVITY OF THE SPACECRAFT. WITH REFERENCE TO FIGURE 1, POSITIVE PITCH TILTS THE SMALL END OF THE SPACECRAFT UP, AND POSITIVE YAW TILTS THE SMALL END OF THE SPACECRAFT TO THE RIGHT OF THE ORBIT PLANE.

CONSIDER A ROTATING COORDINATE SYSTEM, x_b, y_b, z_b[2] (THIS SYSTEMS HAS BEEN CALLED THE BODY COORDINATE SYSTEM BY GSFC PERSONNEL AND IS REFERRED TO AS THE PLATFORM SYSTEM BY SOME OF THE MSC, HOUSTON, PERSONNEL.) WHERE, INITIALLY, x_b IS IDENTICAL WITH x_p, y_b LIES ALONG THE SAME LINE AS z_p BUT ITS DIRECTION IS OPPOSITE, AND z_b IS IDENTICAL WITH y_p. (FIGURE 2.)

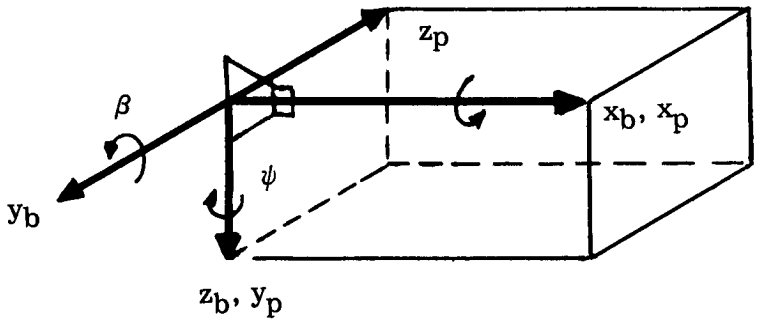


FIGURE 2

IN THIS SYSTEM THE AXES ROTATE WITH THE SPACECRAFT AS MANEUVERS ARE PERFORMED. IT IS ASSUMED THAT INITIALLY THE YAW, PITCH, AND ROLL ANGLES ARE ALL EQUAL TO ZERO. MANEUVERS ARE PERFORMED IN THE ORDER YAW, PITCH, ROLL. YAW IS A POSITIVE ROTATION COUNTERCLOCKWISE ABOUT z_b LOOKING FROM THE POSITIVE DIRECTION OF z_b TOWARD THE CENTER OF GRAVITY OF THE SPACECRAFT. PITCH IS A POSITIVE ROTATION COUNTERCLOCKWISE ABOUT y_p LOOKING FROM THE POSITIVE DIRECTION OF y_b TOWARD THE CENTER OF GRAVITY OF THE SPACECRAFT. ROLL IS A POSITIVE ROTATION COUNTERCLOCKWISE ABOUT x_b LOOKING FROM THE POSITIVE DIRECTION OF x_b TOWARD THE CENTER OF GRAVITY OF THE SPACECRAFT.

FOR USE IN THE FORMULATION TO FOLLOW, WE CONSIDER STILL ANOTHER ROTATION COORDINATE SYSTEM, x,y,z. (FIGURE 3.) THE x AXIS IS IDENTICAL WITH THE x_b AXIS, y LIES ALONG THE SAME LINE AS y_b BUT ITS DIRECTION IS OPPOSITE AND z LIES ALONG THE SAME LINE AS z_b BUT ITS DIRECTION IS OPPOSITE.

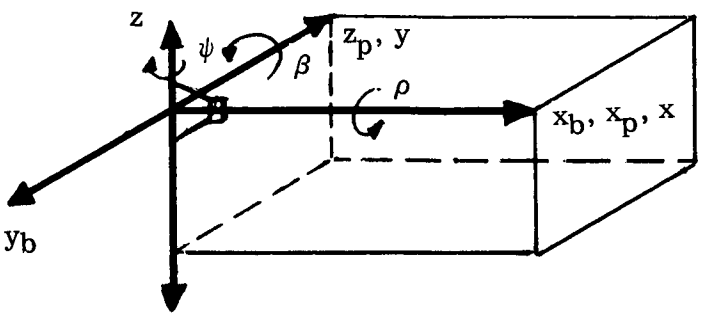


FIGURE 3

IN THIS SYSTEM, THE AXES ROTATE WITH THE SPACECRAFT AS MANEUVERS ARE PERFORMED. IT IS ASSUMED THAT, INITIALLY, THE YAW, PITCH, AND ROLL ANGLES ARE ALL EQUAL TO ZERO. MANEUVERS ARE ASSUMED TO BE PERFORMED IN THE ORDER YAW, PITCH, ROLL. YAW IS DEFINED AS A ROTATION CLOCKWISE ABOUT z LOOKING FROM THE POSITIVE z DIRECTION TOWARD THE CENTER OF GRAVITY OF THE SPACECRAFT. PITCH IS A ROTATION CLOCKWISE ABOUT y LOOKING FROM THE POSITIVE DIRECTION OF y TOWARD THE CENTER OF GRAVITY OF THE SPACECRAFT. ROLL IS A ROTATION COUNTERCLOCKWISE ABOUT x LOOKING FROM POSITIVE x TOWARD THE SPACECRAFT CENTER OF GRAVITY.

THE FOUR RETROROCKETS ARE PLACED SYMMETRICALLY ABOUT THE HEAT SHIELD 90° APART. THEY ARE OFFSET 22.5^o FROM THE AXES. THE ANGLES Φ_i, $1 \leq i \leq 4$ ARE MEASURED CLOCKWISE FROM THE NEGATIVE y AXIS. (FIGURE 4.) THE ROCKETS ARE CANTED BY AN ANGLE, $c = 15^o$ FROM THE LONGITUDINAL CENTER LINE OF THE SPACECRAFT (x AXIS). (FIGURE 5.)

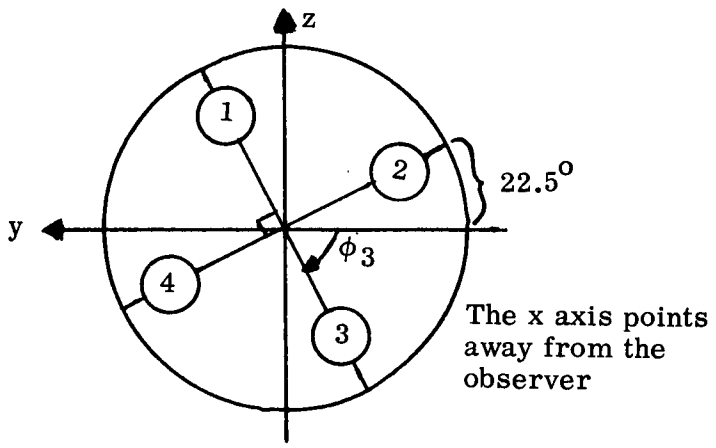


Figure 4

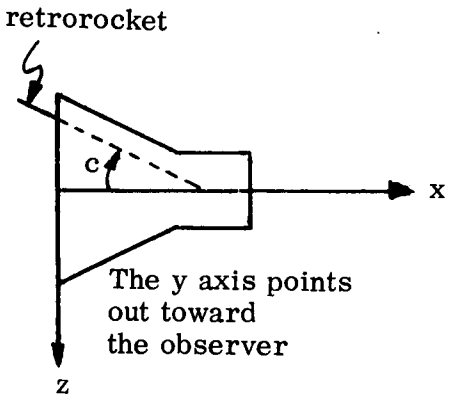


Figure 5

NOW, GIVEN THE POSITION VECTOR, VELOCITY VECTOR, YAW ANGLE, PITCH ANGLE AND ROLL ANGLE AT THE INITIATION OF RETROFIRE AND THE FIRING TIMES OF THE INDIVIDUAL ROCKETS (A ZERO TIME IS GIVEN FOR ANY RETRO THAT DOES NOT FIRE), IT IS DESIRED THAT THE TIME AND THE POSITION AND VELOCITY VECTORS AT THE TERMINATION OF RETROFIRE BE FOUND.

THE FOLLOWING FORMULATION ASSUMES THAT THE RETROS FIRE WITH A CONSTANT THRUST, T, THAT THE ACCELERATION DUE TO GRAVITY IS $-1/r_o^2$, THAT DRAG AND OBLATENESS EFFECTS ARE NEGLIGIBLE AND THAT THE RETROS FIRE THROUGH THE CENTER OF GRAVITY OF THE SPACECRAFT SO THAT TUMBLING EFFECTS CAN BE IGNORED.

THE METHOD CONSISTS, BASICALLY, OF COMPUTING THE ACCELERATION VECTORS DUE TO THE THRUST AND GRAVITY VECTORS AND CALCULATING A MEAN ORIENTATION OF THE ACCELERATION VECTORS. AN OPTION IN THE PROGRAM ALLOWS ONE TO ASSUME EITHER THAT THE INERTIAL PLATFORM OF THE SPACECRAFT IS TORQUED DURING RETROFIRE (IN WHICH CASE THE ORIENTATION OF THE THRUST VECTOR IN INERTIAL SPACE IS CONTINUALLY CHANGING DURING THE FIRING PERIOD), OR THAT IT IS NO BEING TORQUED (IN WHICH CASE THE THRUST VECTOR IS INVARIANT IN INERTIAL SPACE). TORQUEING THE PLATFORM MAY BE THOUGHT OF AS BEING EQUIVALENT TO THE ASTRONAUT'S MAINTAINING HIS ATTITUDE DURING THE FIRING PERIOD BY

USING THE HORIZON AS A REFERENCE THROUGH THE WINDOW OF THE SPACECRAFT.

FROM NEWTON'S SECOND LAW, THE ACCELERATION COMPONENTS DUE TO T, THE THRUST OF THE RETRO ROCKETS IS THE FOLLOWING -

$$A = \frac{T}{m}$$

THE ACCELERATION COMPONENTS ALONG THE $\bar{x}, \bar{y}, \bar{z}$ AXES DUE TO THRUST CAN BE FOUND WITH REFERENCE TO FIGURE 6.

The heat shield lies in the y-z plane

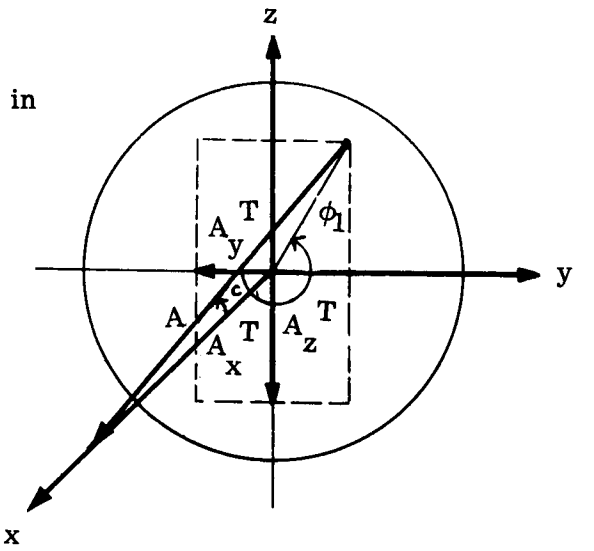


FIGURE 6

WE HAVE

$$\begin{aligned} A_x^T &= A \cos c \\ A_y^T &= A \sin c \cos \phi_i \\ A_z^T &= A \sin c \sin \phi_i \end{aligned} \tag{1}$$

SO THAT,

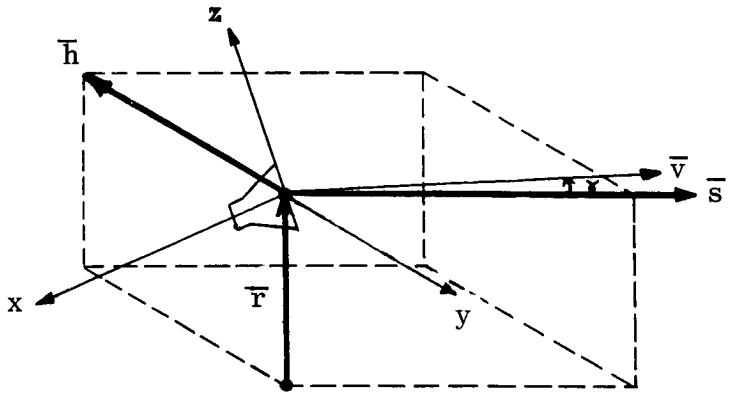
$$\begin{pmatrix} A_x^T \\ A_y^T \\ A_z^T \end{pmatrix} = \begin{pmatrix} \frac{T}{m} \cos c \\ \frac{T}{m} \sin c \cos \phi_i \\ \frac{T}{m} \sin c \sin \phi_i \end{pmatrix} \tag{2}$$

THE MEAN ORIENTATION OF THE THRUST AND GRAVITY ACCELERATION VECTORS IS COMPUTED WITH RESPECT TO THE COORDINATE SYSTEM WHICH IS DEFINED AS FOLLOWS -

$\bar{h} = \bar{r} \times \bar{v}$ (normal to the $\bar{r}$, $\bar{v}$ plane)

$\bar{s} = \bar{h} \times \bar{r}$ (lying in the $\bar{r}$, $\bar{v}$ plane)

WHERE $\bar{r}$ AND $\bar{v}$ ARE THE POSITION AND VELOCITY VECTORS AT TIME t. (FIGURE 7.)



This figure shows the capsule at a nominal retrofire attitude where yaw is 180° and pitch is negative.

Figure 7

IF WE WISH TO FIND THE COMPONENNTS OF THE ACCELERATION DUE TO THRUST PROJECTED ON THE r, h, AND s AXES, WE WILL NEED THREE ROTATIONS IN THE FOLLOWING ORDER - $-\rho$ ABOUT THE x AXIS, $+\beta$ ABOUT THE NEW y AXIS, AND $+\psi$ ABOUT THE NEW z AXIS. OUR ROTATION MATRIX IS THE PRODUCT OF THREE POSITIVE ROTATIONS, $R_z(\psi)\, R_y(\beta)\, R_x(-\rho)$ -

$$\begin{pmatrix} A_{s_T} \\ A_{h_T} \\ A_{r_T} \end{pmatrix} = \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(-\rho) & \sin(-\rho) \\ 0 & -\sin(-\rho) & \cos(-\rho) \end{pmatrix} \begin{pmatrix} A_x^T \\ A_y^T \\ A_z^T \end{pmatrix}$$

$$= \begin{pmatrix} \cos\psi\cos\beta & \sin\psi & -\cos\psi\sin\beta \\ -\sin\psi\cos\beta & \cos\psi & \sin\psi\sin\beta \\ \sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\rho & -\sin\rho \\ 0 & \sin\rho & \cos\rho \end{pmatrix} \begin{pmatrix} A_x^T \\ A_y^T \\ A_z^T \end{pmatrix}$$

$$= \begin{pmatrix} \cos\psi\cos\beta & \sin\psi & -\cos\psi\sin\beta \\ -\sin\psi\cos\beta & \cos\psi & \sin\psi\sin\beta \\ \sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} A_X^T \\ A_Y^T \\ A_Z^T \end{pmatrix} \qquad (3)$$

WHERE USING (2)

$$\begin{pmatrix} A_X^T \\ A_Y^T \\ A_Z^T \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\rho & -\sin\rho \\ 0 & \sin\rho & \cos\rho \end{pmatrix} \begin{pmatrix} A_x^T \\ A_y^T \\ A_z^T \end{pmatrix} = \begin{pmatrix} \frac{T}{m}\cos c \\ \frac{T}{m}\sin c \quad \cos(\rho + \phi_i) \\ \frac{T}{m}\sin c \quad \sin(\rho + \phi_i) \end{pmatrix}$$

THE ACCELERATION DUE TO GRAVITY ALONG THE RADIUS VECTOR TO THE CENTER OF THE EARTH IS -

$$\begin{matrix} A_{s_G} \\ A_{h_G} \\ A_{r_G} \end{matrix} = \begin{matrix} 0 \\ 0 \\ -\frac{1}{r_o^2} \end{matrix} \tag{4}$$

SO THAT AS THE TOTAL ACCELERATION WE HAVE -

$$\begin{pmatrix} A_s \\ A_h \\ A_r \end{pmatrix} = \begin{pmatrix} A_{s_T} \\ A_{h_T} \\ A_{r_T} \end{pmatrix} + \begin{pmatrix} A_{s_G} \\ A_{h_G} \\ A_{r_G} \end{pmatrix}$$

$$= \begin{pmatrix} \cos\psi \cos\beta\, A_X{}^T + \sin\psi\, A_Y{}^T - \cos\psi \sin\beta\, A_Z{}^T \\ -\sin\psi \cos\beta\, A_X{}^T + \cos\psi\, A_Y{}^T + \sin\psi \sin\beta\, A_Z{}^T \\ \sin\beta\, A_X{}^T + \cos\beta\, A_Z{}^T - \frac{1}{r_o^2} \end{pmatrix} \tag{5}$$

FIGURE 8, SHOWS THE VECTORS $\bar{r}, \bar{h}, \bar{s}$ AT TIME t WITH RESPECT TO THE ORIGINAL VECTORS $\bar{r}_o, \bar{h}_o, \bar{s}_o$ AT TIME t_o. IN THE GENERAL PROBLEM WHEN THE YAW ANGLE IS NOT 180° OR WHEN LESS THAN FOUR RETROROCKETS FIRE AT OTHER THAN ZERO DEGREE ROLL, THERE IS A COMPONENT OF ACCELERATION DIRECTED ALONG THE NORMAL TO THE $\bar{r}, \bar{v}$ PLANE. THIS COMPONENT WILL CAUSE THE SPACECRAFT TO TIP OUT OF THE ORIGINAL $\bar{r}_o, \bar{v}_o$ PLANE. THE ANGLE δ IN FIGURE 8 REPRESENTS THE ANGLE BETWEEN THE ORIGINAL $\bar{r}_o, \bar{v}_o$ PLANE AND THE $\bar{r}, \bar{v}$ PLANE AT TIME t. THE ANGLE ϕ SWEPT BY THE RADIUS VECTOR, $\bar{r}$, AT TIME t IS MEASURED FROM $\bar{r}_o$ TO $\bar{r}$.

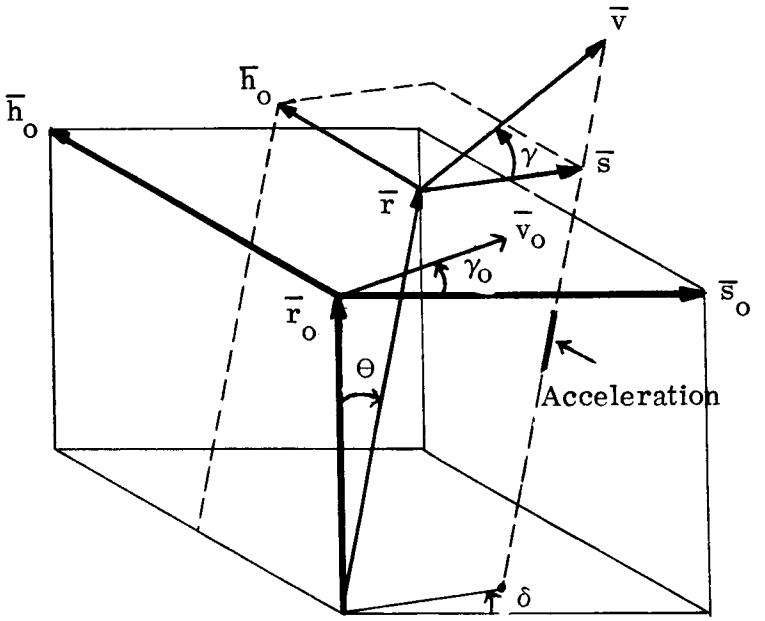


FIGURE 8

IN ORDER TO FIND THE COMPONENTS OF THE ACCELERATION AT TIME t PROJECTED ON THE ORIGINAL AXES, $\bar{r}_o$, $\bar{s}_o$, AND $\bar{h}_o$, TWO ROTATIONS ARE REQUIRED. THE FIRST ROTATION IS ABOUT THE h AXIS AN ANGLE OF $-\phi$ WHERE -

$$\theta \cong \frac{v_o \cos \gamma_o \, \Delta t + \iint A_s \, dt \, dt}{r_o}$$

THE SECOND IS ABOUT THE NEW r AXIS AN ANGLE OF $-\delta$ WHERE -

$$\tan \delta \cong \frac{\iint A_h \, dt \, dt}{v_o \cos \gamma_o \, \Delta t + \iint A_s \, dt \, dt}$$

$$\begin{pmatrix} A_{s_o} \\ A_{h_o} \\ A_{r_o} \end{pmatrix} = R_r(-\delta)\, R_h(-\Theta) \begin{pmatrix} A_s \\ A_h \\ A_r \end{pmatrix}$$

$$\begin{pmatrix} A_{s_o} \\ A_{h_o} \\ A_{r_o} \end{pmatrix} = \begin{pmatrix} \cos\delta & -\sin\delta & 0 \\ \sin\delta & \cos\delta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\Theta & 0 & \sin\Theta \\ 0 & 1 & 0 \\ -\sin\Theta & 0 & \cos\Theta \end{pmatrix} \begin{pmatrix} A_s \\ A_h \\ A_r \end{pmatrix}$$

$$\begin{pmatrix} A_{s_o} \\ A_{h_o} \\ A_{r_o} \end{pmatrix} = \begin{pmatrix} \cos\delta\cos\Theta & -\sin\delta & \cos\delta\sin\Theta \\ \sin\delta\cos\Theta & \cos\delta & \sin\delta\sin\Theta \\ -\sin\Theta & 0 & \cos\Theta \end{pmatrix} \begin{pmatrix} A_s \\ A_h \\ A_r \end{pmatrix} \qquad (6)$$

$$\begin{pmatrix} A_{s_o} \\ A_{h_o} \\ A_{r_o} \end{pmatrix} = \begin{pmatrix} A_s\cos\delta\cos\Theta - A_h\sin\delta + A_r\cos\delta\sin\Theta \\ A_s\sin\delta\cos\Theta + A_h\cos\delta + A_r\sin\delta\sin\Theta \\ -A_s\sin\Theta + A_r\cos\Theta \end{pmatrix}$$

TO OBTAIN AN AVERAGE ACCELERATION DURING THE FIRING PERIOD PROJECTED ON THE ORIGINAL AXES, WE TAKE AVERAGE OF THE ACCELERATION FUNCTIONS -

$$A_{s_o av} = \frac{A_s}{\delta\Theta}\int_0^{\delta}\int_0^{\Theta}\cos\delta(\cos\Theta d\Theta)d\delta - \frac{A_h}{\delta}\int_0^{\delta}\sin\delta d\delta + \frac{A_r}{\delta\Theta}\int_0^{\delta}\int_0^{\Theta}\cos\delta(\sin \quad d\epsilon$$

$$A_{h_o av} = \frac{A_s}{\delta\Theta}\int_0^{\delta}\sin\delta\left(\int_0^{\Theta}\cos\Theta d\Theta\right)d\delta + \frac{A_h}{\delta}\int_0^{\delta}\cos\delta d\delta + \frac{A_r}{\delta\Theta}\int_0^{\delta}\sin\delta\left(\int_0^{\Theta}si\right.$$

$$A_{r_o av} = -\frac{A_s}{\Theta}\int_0^{\Theta}\sin\Theta d\Theta + \frac{A_r}{\Theta}\int_0^{\Theta}\cos\Theta d\Theta$$

UPON INTEGRATION, WE HAVE -

$$A_{s_o av} = A_s\left(\frac{\sin\delta\ \sin\Theta}{\delta\Theta}\right) - A_h\left(\frac{1-\cos\delta}{\delta}\right) + A_r\frac{\sin\delta(1-\cos\Theta)}{\delta\Theta}$$

$$A_{h_o av} = A_s\frac{\sin\Theta(1-\cos\delta)}{\Theta\delta} + A_h\frac{\sin\delta}{\delta} + A_r\frac{(1-\cos\delta)(1-\cos\Theta)}{\delta\Theta}$$

$$A_{r_o av} = -A_s\cdot\frac{(1-\cos\Theta)}{\Theta} + A_r\frac{\sin\Theta}{\Theta}$$

IF WE CAN CONSIDER δ AS SMALL FOR $A_{s_o av}$, AND FOR $A_{h_o av}$, WE OBTAIN -

$$A_{s_o av} \cong \frac{\sin\Theta}{\Theta}A_s + \frac{1-\cos\Theta}{\Theta}A_r$$

$$A_{h_s av} \cong A_h$$

$$A_{r_o av} = -A_s \frac{(1 - \cos \Theta)}{\Theta} + A_r \frac{\sin \Theta}{\Theta}$$

FOR THE NO TORQUE CASE, THE ATTITUDES ARE ASSUMED TO BE HELD CONSTANT WITH RESPECT TO AN UNCHANGING INERTIAL PLATFORM. THE THRUST VECTOR IS THUS INVARIANT IN INERTIAL SPACE. THEREFORE THE ACCELERATION DUE TO THRUST IS CONSTANT AND IS SO TO BE CONSIDERED IN OBTAINING THE AVERAGE ACCELERATION, I.E., THE COMPONENTS OF THE ACCELERATION DUE TO GRAVITY AT TIME t PROJECTED ON THE ORIGINAL $\bar{r}_o, \bar{h}_o, \bar{s}_o$ AXES ARE GIVEN BY -

$$\begin{pmatrix} A_{s_o} \\ A_{h_o} \\ A_{r_o} \end{pmatrix} = R_r(-\delta)\ R_h(-\Theta) \begin{pmatrix} A_{s_G} \\ A_{h_G} \\ A_{r_G} \end{pmatrix}$$

USING (4) AND THE ROTATION MATRIX OF (6) -

$$\begin{pmatrix} A_{s_o} \\ A_{h_o} \\ A_{r_o} \end{pmatrix} = -\frac{1}{r_o^2} \begin{pmatrix} \cos \delta \ \sin \Theta \\ \sin \delta \ \sin \Theta \\ \cos \Theta \end{pmatrix}$$

WE OBTAIN THE AVERAGE ACCELERATION DURING THE FIRING PERIOD PROJECTED ON THE ORIGINAL AXES BY AVERAGING THE ABOVE ACCELERATION FUNCTIONS. AND WE MUST ALSO INCLUDE THE CONSTANT ACCELERATION DUE TO THRUST -

$$\begin{pmatrix} A_{s_o av} \\ A_{h_o av} \\ A_{r_o av} \end{pmatrix} = -\frac{1}{\Theta r_o^2} \begin{pmatrix} \frac{1}{\delta}\int_0^{\delta}\int_0^{\Theta} \cos\delta \sin\Theta d\Theta\, d\delta \\ \frac{1}{\delta}\int_0^{\Theta}\int_0^{\Theta} \sin\delta \sin\Theta d\Theta\, d\delta \\ \int_0^{\Theta} \cos\Theta\, d\,\Theta \end{pmatrix} + \begin{pmatrix} A_{s_T} \\ A_{h_T} \\ A_{r_T} \end{pmatrix}$$

UPON INTEGRATING, WE HAVE -

$$\begin{pmatrix} A_{s_o av} \\ A_{h_o av} \\ A_{r_o av} \end{pmatrix} = -\frac{1}{\Theta r_o^2} \begin{pmatrix} \frac{1}{\delta} \sin\delta\,(1 - \cos\Theta) \\ \frac{1}{\delta}(1 - \cos\delta)(1 - \cos\Theta) \\ \sin\Theta \end{pmatrix} + \begin{pmatrix} A_{s_T} \\ A_{h_T} \\ A_{r_T} \end{pmatrix}$$

AND IGNORING δ AS SMALL, WE HAVE -

$$\begin{pmatrix} A_{s_o av} \\ A_{h_o av} \\ A_{r_o av} \end{pmatrix} \cong \begin{pmatrix} -\frac{1-\cos\Theta}{\Theta r_o^2} \\ 0 \\ \frac{-\sin\Theta}{\Theta r_o^2} \end{pmatrix} + \begin{pmatrix} A_{s_T} \\ A_{h_T} \\ A_{r_T} \end{pmatrix}$$

SO THAT THE ABOVE EQUATIONS (7) REDUCE TO -

$$A_{s_o av} \cong \left(\frac{1-\cos\Theta}{\Theta}\right)\left(-\frac{1}{r_o^2}\right) + A_{s_T}$$

$$A_{h_o av} \cong A_{h_T}$$

$$A_{r_o av} \cong \left(-\frac{1}{r_o^2}\right)\left(\frac{\sin\Theta}{\Theta}\right) + A_{r_T}$$

THE NEXT STEP IN THE DEVELOPMENT WILL BE TO COMPUTE THE $\bar{r}$, $\bar{v}$, AND t AT BURNOUT AS A FUNCTION OF THE ORIGINAL VECTORS, $\bar{r}_o$, $\bar{v}_o$, $\bar{h}_o$.
WE MAY WRITE THE EQUATION OF MOTION AS -

$$\bar{a} = \ddot{\bar{r}} = A_{r_o av}\,\bar{r}_o^* + A_{s_o av}\,\bar{s}_o^* + A_{h_o av}\,\bar{h}_o^*$$

WHERE $\bar{r}_o^*$, $\bar{s}_o^*$, and $\bar{h}_o^*$ ARE UNIT VECTORS ALONG THE $\bar{r}_o$, $\bar{h}_o$ AND $\bar{s}_o$ AXES. WE KNOW THAT -

$$\bar{s}_o^* = \frac{(\bar{r}_o \times \bar{v}_o) \times \bar{r}_o}{r_o v_o \cos\gamma_o \, r_o} = \frac{(\bar{r}_o \cdot \bar{r}_o)\bar{v}_o - (\bar{r}_o \cdot \bar{v}_o)\bar{r}_o}{r_o^2 v_o \cos\gamma_o}$$

$$= \frac{r_o^2 \bar{v}_o - r_o v_o \sin\gamma_o \bar{r}_o}{r_o^2 v_o \cos\gamma_o} = \frac{\bar{v}_o^* - \sin\gamma_o \bar{r}_o^*}{\cos\gamma_o}$$

$$= \frac{\bar{v}_o^*}{\cos\gamma_o} - \tan\gamma_o \bar{r}_o^*$$

SO,

$$\bar{a} = \ddot{\bar{r}} = (A_{r_o av} - \tan\gamma_o A_{s_o av})\bar{r}_o^* + \frac{A_{s_o av}}{\cos\gamma_o}\bar{v}_o^* + A_{h_o av}\bar{h}_o^* \qquad (9)$$

WHERE

$$\bar{h}_o^* = \frac{\bar{r}_o \times \bar{v}_o}{r_o v_o \cos\gamma_o}$$

WE NOW WANT TO INTEGRATE THE EQUATION OF MOTION (9) TO OBTAIN $\bar{v}$ AND $\bar{r}$ AND WE ALSO WANT TO DETERMINE THE CONSTANTS OF INTEGRATION -

$$\bar{v} = \dot{\bar{r}} = \int_{t_o}^{t} \ddot{\bar{r}}\, dt + \bar{c}_1$$

$$= \int_{t_o}^{t} \left[\left(A_{r_o av} - \tan\gamma_o A_{s_o av}\right)\bar{r}_o^* + \frac{A_{s_o av}}{\cos\gamma_o}\bar{v}_o^* + A_{h_o av}\bar{h}_o^* \right] dt + \bar{c}_1 \qquad (10)$$

FOR SIMPLICITY IN PERFORMING THE NEXT INTEGRATION TO OBTAIN $\bar{r}$, WE NOW LET -

$$\left.\begin{aligned} A_r' &= A_{r_o av} - \tan\gamma_o A_{s_o av} \\ A_v' &= \frac{A_{s_o av}}{\cos\gamma_o} \\ A_h' &= A_{h_o av} \end{aligned}\right\} \qquad (11)$$

SETTING UP INITIAL CONDITIONS -

If $t = t_o$, then $\bar{v} = \bar{v}_o$ (at initial time t_o)

$$\bar{v}_o = \int_{t_o}^{t_o} \ddot{\bar{r}}\, dt + \bar{c}_1$$

$$\bar{v}_o = 0 + \bar{c}_1$$

NOW, HAVING OBTAINED $\bar{c}_1 = \bar{v}_o$ (A CONSTANT), WE MAY NOW USE (10) AND (11) TO OBTAIN THE GENERAL EXPRESSION -

$$\bar{v} = \bar{r}_o * \int_{t_o}^{t} A_r'\, dt + \bar{v}_o * \int_{t_o}^{t} A_v'\, dt + \bar{h}_o * \int_{t_o}^{t} A_h'\, dt + \bar{v}_o$$

WHICH WE MAY USE TO OBTAIN $\bar{v}$ FOR ANY TIME t.

UPON INTEGRATING AGAIN -

$$\bar{r} = \int_{t_o}^{t} \bar{v}\,dt + \bar{c}_2$$

$$= \bar{r}_o * \int_{t_o}^{t}\int_{t_o}^{t} A_r'\,dt\,dt + \bar{v}_o * \int_{t_o}^{t}\int_{t_o}^{t} A_v'\,dt\,dt + \bar{h}_o * \int_{t_o}^{t}\int_{t_o}^{t} A_h'\,dt\,dt + \bar{v}_o \int_{t_o}^{t} dt + \bar{c}_2 \qquad (11.1)$$

TO OBTAIN $\bar{c}_2$ WE SET UP INITIAL CONDITIONS AGAIN -

If $t = t_o$, then $r = r_o$ (at initial time)

$$\bar{r}_o = \int_{t_o}^{t_o} \bar{v}\,dt + \bar{c}_2$$

$$\bar{r}_o = 0 + \bar{c}_2$$

SO WE OBTAIN $\bar{r}_o = \bar{c}_o$ (A CONSTANT) AND WE SUBSTITUTE DIRECTLY INTO (11.1) TO OBTAIN -

$$\bar{r} = \bar{r}_o * \int_{t_o}^{t}\int_{t_o}^{t} A_r'\,dt\,dt + \bar{v}_o * \int_{t_o}^{t}\int_{t_o}^{t} A_v'\,dt\,dt + \bar{h}_o * \int_{t_o}^{t}\int_{t_o}^{t} A_h'\,dt\,dt + \bar{v}_o \int_{t_o}^{t} dt + \bar{r}_o$$

AND SINCE $r_o^* r_o = r_o$ AND $v_o^* v_o = v_o$ WE MAY WRITE -

$$\bar{r} = \bar{r}_o * \left[\int_{t_o}^{t}\int_{t_o}^{t} A_r'\,dt\,dt + r_o\right] + \bar{v}_o * \left[\int_{t_o}^{t}\int_{t_o}^{t} A_v'\,dt\,dt + v_o \int_{t_o}^{t} dt\right] + \bar{h}_o * \int_{t_o}^{t}\int_{t_o}^{t} A_h'\,dt\,dt$$

NOW, SUBSTITUTING BACK USING (11), AND LEAVING OFF THE LIMITS OF INTEGRATION, FOR SIMPLICITY, WE OBTAIN -

$$\bar{v} = \bar{r}_o * \int \left(A_{r_o av} - \tan \gamma_o A_{s_o av} \right) dt + \bar{v}_o * \int \frac{A_{s_o av}}{\cos \gamma_o} dt + \bar{h}_o * \int A_{h_o av} dt + \bar{v}_o$$

$$= \bar{r}_o * \left[\int A_{r_o av} dt - \tan \gamma_o \int A_{s_o av} dt \right] + \frac{\bar{v}_o *}{\cos \gamma_o} \int A_{s_o av} dt + \bar{h}_o * \int A_{h_o av} dt + \bar{v}_o$$

$$= \bar{r}_o * \int A_{r_o av} dt + \left[\frac{\bar{v}_o *}{\cos \gamma_o} - \bar{r}_o * \tan \gamma_o \right] \int A_{s_o av} dt + \bar{h}_o * \int A_{h_o av} dt + \bar{v}_o$$

AND

$$\bar{r} = \bar{r}_o * \left[\int \int \left(A_{r_o av} - \tan \gamma_o A_{s_o av} \right) dt\, dt + r_o \right] + \bar{v}_o * \left[\int \int \frac{A_{s_o av}}{\cos \gamma_o} dt\, dt + v_o \int dt \right]$$

$$+ \bar{h}_o * \int \int A_{h_o av} dt\, dt$$

$$= \bar{r}_o * \left[\int \int A_{r_o av} dt\, dt - \tan \gamma_o \int \int A_{s_o av} dt\, dt + r_o \right] + \bar{v}_o * \left[\frac{1}{\cos \gamma_o} \int \int A_{s_o av} dt\, dt + v_o \int dt \right]$$

$$+ \bar{h}_o * \int \int A_{h_o av} dt\, dt$$

$$= \bar{r}_o * \int \int A_{r_o av} dt\, dt - \bar{r}_o * \tan \gamma_o \int \int A_{s_o av} dt dt + \bar{r}_o * r_o + \frac{v_o *}{\cos \gamma_o} \int \int A_{s_o av} dt\, dt$$

$$+ \bar{v}_o * v_o \int dt + \bar{h}_o * \int \int A_{h_o av} dt\, dt$$

$$= \bar{r}_o * \iint A_{r_o av}\, dt\, dt + \left[\frac{\bar{v}_o *}{\cos \gamma_o} - \bar{r}_o * \tan \gamma_o\right] \iint A_{s_o av}\, dt\, dt + \bar{h}_o * \iint A_{h_o av}\, dt\, dt$$

$$+ \bar{v}_o * v_o \int dt + \bar{r}_o * r_o$$

LET

$$\left.\begin{aligned} V_r &= \int A_{r_o av}\, dt \\ V_\Theta &= \int A_{s_o av}\, dt \\ V_N &= \int A_{h_o av}\, dt \end{aligned}\right\} \qquad (12)$$

THEN,

$$\bar{v} = \bar{r}_o * V_r + \left[\frac{\bar{v}_o *}{\cos \gamma_o} - \bar{r}_o * \tan \gamma_o\right] V_\Theta + \bar{h}_o * V_N + \bar{v}_o \qquad (13)$$

AND LET

$$R_r = \iint A_{r_o av}\, dt\, dt$$

$$R_\Theta = \iint A_{s_o av}\, dt\, dt$$

$$R_N = \iint A_{h_o av}\, dt\, dt$$

THEN, -

$$\bar{r} = \bar{r}_o^* R_r + \left[\frac{\bar{v}_o^*}{\cos \gamma_o} - \bar{r}_o^* \tan \gamma_o \right] R_\Theta + \bar{h}_o^* R_N + \bar{v}_o^* v_o \int dt + \bar{r}_o^* r_o$$

$$= \bar{r}_o^* R_r + \left[\frac{\bar{v}_o^*}{\cos \gamma_o} - \bar{r}_o^* \tan \gamma_o \right] R_\Theta + \bar{h}_o^* R_N + \bar{v}_o \int dt + \bar{r}_o \qquad (14)$$

WE MUST NOW PROCEED TO EVALUATE V_r, V_Θ, V_N , AND R_r, R_Θ, R_N .

WE FIRST EVALUATE V_r , V_Θ , AND V_N .

FOR THE TORQUE CASE WE MAY WRITE USING (7)

$$\left.\begin{aligned}
\int A_{r_o av}\, dt &= \int \left[-\left(\frac{1 - \cos \Theta}{\Theta} \right) A_s + \left(\frac{\sin \Theta}{\Theta} \right) A_r \right] dt \\
&= \frac{\sin \Theta}{\Theta} \int A_r\, dt - \left(\frac{1 - \cos \Theta}{\Theta} \right) \int A_s\, dt \\
\int A_{h_o av}\, dt &= \int A_h\, dt \\
\int A_{s_o av}\, dt &= \int \left[\left(\frac{\sin \Theta}{\Theta} \right) A_s + \left(\frac{1 - \cos \Theta}{\Theta} \right) A_r \right] dt \\
&= \frac{\sin \Theta}{\Theta} \int A_s\, dt + \frac{1 - \cos \Theta}{\Theta} \int A_r\, dt
\end{aligned}\right\} \qquad (15)$$

WHERE (USING (5)):

$$\left.\begin{aligned}
\int A_r\,dt &= \int\left[\sin\beta\ A_X^{\,T} + \cos\beta\ A_Z^{\,T} + \left(-\frac{1}{r_o^2}\right)\right]dt \\
&= \sin\beta\int A_X^{\,T}\,dt + \cos\beta\int A_Z^{\,T}\,dt - \frac{1}{r_o^2}\int dt \\
\int A_s\,dt &= \int\left[\cos\psi\cos\beta\ A_X^{\,T} + \sin\psi\ A_Y^{\,T} - \cos\psi\sin\beta\ A_Z^{\,T}\right]dt \\
&= \cos\psi\cos\beta\int A_X^{\,T}\,dt + \sin\psi\int A_Y^{\,T}\,dt - \cos\psi\sin\beta\int A_Z^{\,T}\,dt \\
&= \cos\psi\left\{\cos\beta\int A_X^{\,T}\,dt - \sin\beta\int A_Z^{\,T}\,dt\right\} + \sin\psi\int A_Y^{\,T}\,dt
\end{aligned}\right\} \quad (16)$$

$$\left.\begin{aligned}
\int A_h\,dt &= \int\left[-\sin\psi\cos\beta\ A_X^{\,T} + \cos\psi\ A_Y^{\,T} + \sin\psi\sin\beta\ A_Z^{\,T}\right]dt \\
&= -\sin\psi\cos\beta\int A_X^{\,T}\,dt + \cos\psi\int A_Y^{\,T}\,dt + \sin\psi\sin\beta\int A_Z^{\,T}\,dt \\
&= \sin\psi\left\{-\cos\beta\int A_X^{\,T}\,dt + \sin\beta\int A_Z^{\,T}\,dt\right\} + \cos\psi\int A_Y^{\,T}\,dt
\end{aligned}\right\} \quad (16.1)$$

SINCE THE LEVEL OF THE THRUSTING FORCE IS CONSIDERED CONSTANT AND THE AVERAGE MASS IS USED THROUGHOUT THE BURNING PERIOD, THE INTEGRALS IN (16) AND (16.1) MAY BE REPLACED BY SUMMATIONS. WE THUS HAVE -

$$\int A_r\,dt = \sin\beta \sum_i A_{X_i}{}^T \Delta t_i + \cos\beta \sum_i A_{Z_i}{}^T \Delta t_i - \frac{1}{r_o^2} \sum_i \Delta t_i$$

$$\int A_s\,dt = \cos\psi \left\{ \cos\beta \sum_i A_{X_i}{}^T \Delta t_i - \sin\beta \sum_i A_{Z_i}{}^T \Delta t_i \right\} + \sin\psi \sum_i A_{Y_i}{}^T \Delta t_i$$

$$\int A_h\,dt = \sin\psi \left\{ -\cos\beta \sum_i A_{X_i}{}^T \Delta t_i + \sin\beta \sum_i A_{Z_i}{}^T \Delta t_i \right\} + \cos\psi \sum_i A_{Y_i}{}^T \Delta t_i$$

If we let:

$$\left.\begin{aligned} \sum_i A_{X_i}{}^T \Delta t_i &= V_x{}^T \\ \sum_i A_{Y_i}{}^T \Delta t_i &= V_y{}^T \\ \sum_i A_{Z_i}{}^T \Delta t_i &= V_z{}^T \end{aligned}\right\} \qquad (17)$$

AND ALSO

$$\sum_i \Delta t_i = A_4 \tag{18}$$

THEN

$$\left.\begin{aligned} \int A_r\,dt &= \sin\beta\, V_x{}^T + \cos\beta\, V_z{}^T - \frac{A_4}{r_o^2} \\ \int A_s\,dt &= \cos\psi \left\{ \cos\beta\, V_x{}^T - \sin\beta\, V_z{}^T \right\} + \sin\psi\, V_y{}^T \\ \int A_h\,dt &= \sin\psi \left\{ -\cos\beta\, V_x{}^T + \sin\beta\, V_z{}^T \right\} + \cos\psi\, V_y{}^T \end{aligned}\right\} \tag{19}$$

AND SUBSTITUTING (19) BACK INTO (15), WE USE (12) TO OBTAIN, FINALLY: -

$$V_r = \int A_{r_o av}\,dt = \left(\frac{\sin\Theta}{\Theta}\right)\int A_r\,dt - \left(\frac{1-\cos\Theta}{\Theta}\right)\int A_s\,dt$$

$$= \left(\frac{\sin\Theta}{\Theta}\right)\left(\sin\beta\, V_x{}^T + \cos\beta\, V_z{}^T - \frac{A_4}{r_o^2}\right) - \left(\frac{1-\cos\Theta}{\Theta}\right)\left(\cos\psi\left\{\cos\beta\, V_x{}^T - \sin\beta\, V_z{}^T\right\} + \sin\psi\, V_y{}^T\right)$$

$$V_\Theta = \int A_{s_o av}\, dt = \frac{\sin\Theta}{\Theta}\int A_s\, dt + \left(\frac{1-\cos\Theta}{\Theta}\right)\int A_r\, dt$$

$$= \left(\frac{\sin\Theta}{\Theta}\right)\left(\cos\psi\left\{\cos\beta\, V_x{}^T - \sin\beta\, V_z{}^T\right\} + \sin\psi\, V_y{}^T\right) + \left(\frac{1-\cos\Theta}{\Theta}\right)$$

$$\left(\sin\beta\, V_x{}^T + \cos\beta\, V_z{}^T - \frac{A_4}{r_o^2}\right)$$

$$V_N = \int A_{h_o av}\, dt = \int A_h\, dt$$

$$= \sin\psi\left\{-\cos\beta\, V_x{}^T + \sin\beta\, V_z{}^T\right\} + \cos\psi\, V_y{}^T$$

FOR THE TORQUE CASE.

NOW WE MUST EVALUATE V_r, V_Θ AND V_N FOR THE NO TORQUE CASE. USING (12), (8), AND (3), WE OBTAIN:

$$V_r = \int A_{r_o av}\, dt = \int\left[\left(\frac{\sin\Theta}{\Theta}\right)\left(-\frac{1}{r_o^2}\right) + A_{r_T}\right]dt$$

$$= \left(\frac{\sin\Theta}{\Theta}\right)\left(-\frac{1}{r_o^2}\right)\int dt + \int A_{r_T}\, dt$$

$$= \left(\frac{\sin\Theta}{\Theta}\right)\left(-\frac{1}{r_o^2}\right)\int dt + \int\left[A_X{}^T \sin\beta + A_Z{}^T \cos\beta\right]dt$$

$$= \left(\frac{\sin\Theta}{\Theta}\right)\left(-\frac{1}{r_o^2}\right)\int dt + \sin\beta\int A_X{}^T\, dt + \cos\beta\int A_Z{}^T\, dt$$

$$V_\Theta = \int A_{s_o av}\, dt = \int \left[\left(\frac{1 - \cos\Theta}{\Theta} \right) \left(- \frac{1}{r_o^2} \right) + A_{s_T} \right] dt$$

$$= \left(\frac{1 - \cos\Theta}{\Theta} \right) \left(- \frac{1}{r_o^2} \right) \int dt + \int A_{s_T}\, dt$$

$$= \left(\frac{1 - \cos\Theta}{\Theta} \right) \left(- \frac{1}{r_o^2} \right) \int dt + \int \left[A_X{}^T \cos\psi \cos\beta + A_Y{}^T \sin\psi - A_Z{}^T \cos\psi \sin\beta \right] dt$$

$$= \left(\frac{1 - \cos\Theta}{\Theta} \right) \left(- \frac{1}{r_o^2} \right) \int dt + \cos\psi \cos\beta \int A_X{}^T\, dt + \sin\psi \int A_Y{}^T\, dt - \cos\psi \sin\beta \int A_Z{}^T\, dt$$

$$V_N = \int A_{h_o av}\, dt = \int A_h\, dt$$

$$= \int \left[A_X{}^T \left(- \sin\psi \cos\beta \right) + A_Y{}^T \cos\psi + A_Z{}^T \sin\psi \sin\beta \right] dt$$

$$= - \sin\psi \cos\beta \int A_X{}^T\, dt + \cos\psi \int A_Y{}^T\, dt + \sin\psi \sin\beta \int A_Z{}^T\, dt$$

AGAIN WE REPLACE THE INTEGRALS BY SUMMATIONS, AND USING (17) AND (18) WE OBTAIN:

$$V_r = \left(\frac{\sin\Theta}{\Theta}\right)\left(-\frac{1}{r_o^2}\right)A_4 + \sin\beta\ V_x^T + \cos\beta\ V_z^T$$

$$V_\Theta = \left(\frac{1-\cos\Theta}{\Theta}\right)\left(-\frac{1}{r_o^2}\right)A_4 + \cos\psi\cos\beta\ V_x^T + \sin\psi\ V_y^T - \cos\psi\sin\beta\ V_z^T$$

$$V_N = -\sin\psi\cos\beta\ V_x^T + \cos\psi\ V_y^T + \sin\psi\sin\beta\ V_z^T$$

FOR THE NO TORQUE CASE.
IF WE NOW LET

$$\alpha_1 = \cos\beta\ V_x^T - \sin\beta\ V_z^T$$

$$A_1 = \sin\beta\ V_x^T + \cos\beta\ V_z^T$$

$$A_2 = \cos\psi\alpha_1 + \sin\psi\ V_y^T$$

$$A_3 = -\sin\psi\alpha_1 + \cos\psi\ V_y^T$$

WE MAY NOW WRITE:

$$V_r = \left(\frac{\sin\Theta}{\Theta}\right)\left(A_1 - \frac{A_4}{r_o^2}\right) - \left(\frac{1-\cos\Theta}{\Theta}\right)A_2$$

$$V_\Theta = \left(\frac{\sin\Theta}{\Theta}\right)A_2 + \left(\frac{1-\cos\Theta}{\Theta}\right)\left(A_1 - \frac{A_4}{r_o^2}\right)$$

$$V_N = A_3$$

FOR THE TORQUE CASE, AND FOR THE NO TORQUE CASE, WE MAY WRITE:

$$V_r = \left(\frac{\sin\Theta}{\Theta}\right)\left(-\frac{A_4}{r_o^2}\right) + A_1$$

$$V_\Theta = \left(\frac{1-\cos\Theta}{\Theta}\right)\left(-\frac{A_4}{r_o^2}\right) + A_2$$

$$V_N = A_3$$

NOW WE MUST EVALUATE R_r , R_Θ , AND R_N FOR BOTH THE TORQUE AND NO TORQUE CASES. WE FIRST CONSIDER THE TORQUE CASE:

$$
\left.
\begin{aligned}
R_r &= \iint A_{r_o av}\, dt\, dt = \iint \left[-\left(\frac{1-\cos\Theta}{\Theta}\right) A_s + \left(\frac{\sin\Theta}{\Theta}\right) A_r \right] dt\, dt \\
&= \frac{\sin\Theta}{\Theta} \iint A_r\, dt\, dt - \left(\frac{1-\cos\Theta}{\Theta}\right) \iint A_s\, dt\, dt \\
R_\Theta &= \iint A_{s_o av}\, dt\, dt = \iint \left[\left(\frac{\sin\Theta}{\Theta}\right) A_s + \left(\frac{1-\cos\Theta}{\Theta}\right) A_r \right] dt\, dt \\
&= \left(\frac{\sin\Theta}{\Theta}\right) \iint A_s\, dt\, dt + \left(\frac{1-\cos\Theta}{\Theta}\right) \iint A_r\, dt\, dt \\
R_N &= \iint A_{h_o av}\, dt\, dt = \iint A_h\, dt\, dt
\end{aligned}
\right\} \qquad (20)
$$

WHERE (USING (5)),

$$\iint A_r\, dt\, dt = \sin\beta \iint A_X{}^T\, dt\, dt + \cos\beta \iint A_Z{}^T\, dt\, dt - \frac{1}{r_o^2} \iint dt\, dt$$

$$\iint A_s\, dt\, dt = \cos\psi \cos\beta \iint A_X{}^T\, dt\, dt + \sin\psi \iint A_Y{}^T\, dt\, dt - \cos\psi \sin\beta \iint A_Z{}^T\, dt\, dt$$

$$\iint A_h\, dt\, dt = -\sin\psi \cos\beta \iint A_X{}^T\, dt\, dt + \cos\psi \iint A_Y{}^T\, dt\, dt + \sin\psi \sin\beta \iint A_Z{}^T\, dt\, dt$$

AGAIN WE REPLACE THE INTEGRALS BY SUMMATIONS SINCE THE LEVEL OF THRUST IS CONSIDERED CONSTANT AND THE AVERAGE MASS IS USED THROUGHOUT THE BURNING PERIOD. WE OBTAIN:

$$\iint A_r \, dt \, dt = \sin\beta \sum_i \left(A_{X_i}{}^T \Delta t_i \right) f \Delta t_i + \cos\beta \sum_i \left(A_{Z_i}{}^T \Delta t_i \right) f \Delta t_i - \frac{\left(\sum_i \Delta t_i \right)^2}{2 r_o^2}$$

$$\iint A_s \, dt \, dt = \cos\psi \cos\beta \sum_i \left(A_{X_i}{}^T \Delta t_i \right) f \Delta t_i + \sin\psi \sum_i \left(A_{Y_i}{}^T \Delta t_i \right) f \Delta t_i$$

$$- \cos\psi \sin\beta \sum_i \left(A_{Z_i}{}^T \Delta t_i \right) f \Delta t_i$$

$$\iint A_h \, dt \, dt = - \sin\psi \cos\beta \sum_i \left(A_{X_i}{}^T \Delta t_i \right) f \Delta t_i + \cos\psi \sum_i \left(A_{Y_i}{}^T \Delta t_i \right) f \Delta t_i$$

$$+ \sin\psi \sin\beta \sum_i \left(A_{Z_i}{}^T \Delta t_i \right) f \Delta t_i$$

LETTING

$$\left.\begin{aligned} \sum_i \left(A_{X_i}{}^T \Delta t_i \right) f \Delta t_i &= R_x{}^T \\ \sum_i \left(A_{Y_i}{}^T \Delta t_i \right) f \Delta t_i &= R_y{}^T \\ \sum_i \left(A_{Z_i}{}^T \Delta t_i \right) f \Delta t_i &= R_z{}^T \end{aligned}\right\} \qquad (21)$$

WE OBTAIN USING (19):

$$\left.\begin{aligned}
\iint A_r\,dt\,dt &= \sin\beta\ R_x^T + \cos\beta\ R_z^T - \frac{(A_4)^2}{2r_o^2} \\
\iint A_s\,dt\,dt &= \cos\psi\cos\beta\ R_x^T + \sin\psi\ R_y^T - \cos\psi\sin\beta\ R_z^T \\
\iint A_h\,dt\,dt &= -\sin\psi\cos\beta\ R_x^T + \cos\psi\ R_y^T + \sin\psi\sin\beta\ R_z^T
\end{aligned}\right\} \quad (22)$$

NOW, SUBSTITUTING (22) BACK INTO (20):

$$R_r = \iint A_{r_o av}\,dt\,dt = \left(\frac{\sin\Theta}{\Theta}\right)\left(\sin\beta\ R_x^T + \cos\beta\ R_z^T - \frac{(A_4)^2}{2r_o^2}\right)$$

$$\left(-\frac{1-\cos\Theta}{\Theta}\right)\left(\cos\psi\cos\beta\ R_x^T + \sin\psi\ R_y^T - \cos\psi\sin\beta\ R_z^T\right)$$

$$R_\Theta = \iint A_{s_o av}\,dt\,dt = \left(\frac{\sin\Theta}{\Theta}\right)\left(\cos\psi\cos\beta\ R_x^T + \sin\psi\ R_y^T - \cos\psi\sin\beta\ R_z^T\right)$$

$$+\left(\frac{1-\cos\Theta}{\Theta}\right)\left(\sin\beta\ R_x^T + \cos\beta\ R_z^T - \frac{(A_4)^2}{2r_o^2}\right)$$

$$R_N = \iint A_{h_o av}\,dt\,dt = -\sin\psi\cos\beta\ R_x^T + \cos\psi\ R_y^T + \sin\psi\sin\beta\ R_z^T$$

FOR THE TORQUE CASE.

NOW TO FIND R_r , R_Θ , AND R_N FOR THE NO TORQUE CASE,

WE USE (8) AND (3) TO OBTAIN:

$$R_r = \iint A_{r_o av}\, dt\, dt = \iint \left[\left(\frac{\sin \Theta}{\Theta} \right) \left(- \frac{1}{r_o^2} \right) + A_{r_T} \right] dt\, dt$$

$$= \left(\frac{\sin \Theta}{\Theta} \right) \left(- \frac{1}{r_o^2} \right) \iint dt\, dt + \sin \beta \iint A_X{}^T\, dt\, dt + \cos \beta \iint A_Z{}^T\, dt\, dt$$

$$R_\Theta = \iint A_{s_o av}\, dt\, dt = \iint \left[\left(\frac{1 - \cos \Theta}{\Theta} \right) \left(- \frac{1}{r_o^2} \right) + A_{s_T} \right] dt\, dt$$

$$= \left(\frac{1 - \cos \Theta}{\Theta} \right) \left(- \frac{1}{r_o^2} \right) \iint dt\, dt + \cos \psi \cos \beta \iint A_X{}^T\, dt\, dt$$

$$+ \sin \psi \iint A_Y{}^T\, dt\, dt - \cos \psi \sin \beta \iint A_Z{}^T\, dt\, dt$$

$$R_N = \iint A_{h_o av}\, dt\, dt = \iint A_h\, dt\, dt$$

$$= - \sin \psi \cos \beta \iint A_X{}^T\, dt\, dt + \cos \psi \iint A_Y{}^T\, dt\, dt$$

$$+ \sin \psi \sin \beta \iint A_Z{}^T\, dt\, dt$$

NOW, REPLACING THE INTEGRALS BY SUMMATIONS, AND USING (21) AND (18), WE HAVE: -

$$R_r = \left(\frac{\sin\Theta}{\Theta}\right)\left(-\frac{A_4^2}{2r_o^2}\right) + \sin\beta\, R_x^T + \cos\beta\, R_z^T$$

$$R_\Theta = \left(\frac{1-\cos\Theta}{\Theta}\right)\left(-\frac{A_4^2}{2r_o^2}\right) + \cos\psi\cos\beta\, R_x^T + \sin\psi\, R_y^T - \cos\psi\sin\beta\, R_z^T$$

$$R_N = -\sin\psi\cos\beta\, R_x^T + \cos\psi\, R_y^T + \sin\psi\sin\beta\, R_z^T$$

FOR THE NO TORQUE CASE.
IF WE NOW LET

$$\alpha_2 = \cos\beta\, R_x^T - \sin\beta\, R_z^T$$

$$B_1 = \sin\beta\, R_x^T + \cos\beta\, R_z^T$$

$$B_2 = \cos\psi\alpha_2 + \sin\psi R_y^T$$

$$B_3 = -\sin\psi\alpha_2 + \cos\psi\, R_y^T$$

$$A_5 = \frac{A_4}{2}$$

WE MAY NOW WRITE: —

$$R_r = \left(\frac{\sin\Theta}{\Theta}\right)\left(B_1 - \frac{A_4 A_5}{r_o^2}\right) - \left(\frac{1-\cos\Theta}{\Theta}\right)\left(B_2\right)$$

$$R_\Theta = \left(\frac{\sin\Theta}{\Theta}\right)\left(B_2\right) + \left(\frac{1-\cos\Theta}{\Theta}\right)\left(B_1 - \frac{A_4 A_5}{r_o^2}\right)$$

$$R_N = B_3$$

FOR THE TORQUE CASE, AND FOR THE NO TORQUE CASE, WE MAY WRITE -

$$R_r = \left(\frac{\sin\Theta}{\Theta}\right)\left(-\frac{A_4 A_5}{r_o^2}\right) + B_1$$

$$R_\Theta = \left(\frac{1-\cos\Theta}{\Theta}\right)\left(-\frac{A_4 A_5}{r_o^2}\right) + B_2$$

$$R_N = B_3$$

HAVING OBTAINED V_r , V_Θ , V_N , AND R_r , R_Θ , AND R_N FOR BOTH THE TORQUE AND NO TORQUE CASES, WE MAY NOW SUBSTITUTE THEM DIRECTLY INTO (13) AND (14) TO OBTAIN $\bar{v}$ AND $\bar{r}$ RESPECTIVELY. NOTE THAT WE MAY NOW REWRITE (14) AS -

$$\bar{r} = \bar{r}_o^* R_r + \left[\frac{\bar{v}_o^*}{\cos\gamma_o} - \bar{r}_o^* \tan\gamma_o\right] R_\Theta + \bar{h}_o^* R_N + \bar{v}_o A_4 + \bar{r}_o$$

where we have replaced the integral $\int dt$ in (14) by the summation $\sum_i \Delta t_i = A_4$.

WRITING THE EQUATION FOR THE BURNOUT VECTORS $\bar{r}$ AND $\bar{v}$ IN TERMS OF THE INPUT VECTORS $\bar{r}_o$, $\bar{v}_o$, AND $\bar{h}_o$ -

$$\bar{r} = D_1 \bar{r}_0 + D_2 \bar{v}_0 + D_3 \bar{h}_0$$

$$\bar{v} = C_1 \bar{r}_0 + C_2 \bar{v}_0 + C_3 \bar{h}_0$$

WHERE

$$D_1 = \frac{R_r - R_\Theta \tan \gamma_o}{r_o} + 1$$

$$D_2 = \frac{R_\Theta}{V_o \cos \gamma_o} + A_4$$

$$D_3 = \frac{R_N}{h_o}$$

AND

$$C_1 = \frac{V_r - V_\Theta \tan \gamma_o}{r_o}$$

$$C_2 = \frac{V_\Theta}{v_o \cos \gamma_o} + 1$$

$$C_3 = \frac{V_N}{h_o}$$

THE EVALUATION OF THE EQUATIONS IN (17) AND (21) FOR V_xT, V_yT, V_zT, AND R_x^T, R_y^T, R_z^T IS DEPENDENT UPON THE ACCELERATION DUE TO THRUST, THE NUMBER OF RETROROCKETS FIRED, THE LENGTH OF THE FIRING PERIODS OF THESE ROCKETS. FURTHERMORE, R_x^T, R_y^T, AND R_z^T WILL BE SEEN TO BE DEPENDENT UPON THE FIRING TIME OF EACH INDIVIDUAL ROCKET. THE EVALUATION OF V_xT, V_yT, AND V_zT IS STRAIGHTFORWARD -

$$V_xT = \frac{n_R \Delta t_D T \cos C}{m_{n_R}}$$

$$V_yT = \frac{\Delta t_D T \sin C}{m_{n_R}} \sum_{i=1}^{n_R} \cos(\phi_i + \rho) \tag{23}$$

$$V_zT = \frac{\Delta t_D T \sin C}{m_{n_R}} \sum_{i=1}^{n_R} \sin(\phi_i + \rho)$$

WHERE Δt_D IS THE LENGTH OF THE FIRING PERIOD FOR EACH RETROROCKET, T IS THE THRUST FOR EACH RETROROCKET, n_R IS THE NUMBER OF ROCKETS FIRED AND m_{n_R} IS THE AVERAGE MASS DURING THE BURNING PERIOD. IN ORDER TO EVALUATE R_x^T, R_y^T, AND R_z^T, WE MUST EVALUATE $f\Delta t_i$ WHICH WILL BE SHOWN TO BE A FUNCTION OF THE LENGTH OF THE FIRING AND THE TIME OF FIRING OF EACH INDIVIDUAL ROCKET.

THE NOMINAL RETROFIRING SEQUENCE IS 1, 3, 2, 4 WITH REFERENCE TO FIGURE 4 EACH ROCKET BURNS FOR Δt_D SECONDS, AND THE RETROS FIRE Δt_D SECONDS APART. THIS REPRESENTS A SEQUENCE OF THRUSTING PERIODS WHICH CAN BE CONSIDERED AS A CONTINUOUS THRUST IF THRUST TRANSIENTS ARE SMALL. SINCE THE ACCELERATION DUE TO THRUST IS NOT A FUNCTION OF THE INDIVIDUAL FIRING TIMES OF THE RETROS, BUT ONLY OF THE LENGTH OF THE INDIVIDUAL FIRING PERIODS AND THE THURST AND MASS OF THE ROCKETS, WE MAY IMMEDIATELY WRITE AN EXPRESSION FOR THE MEAN ACCELERATION DUE TO THRUST -

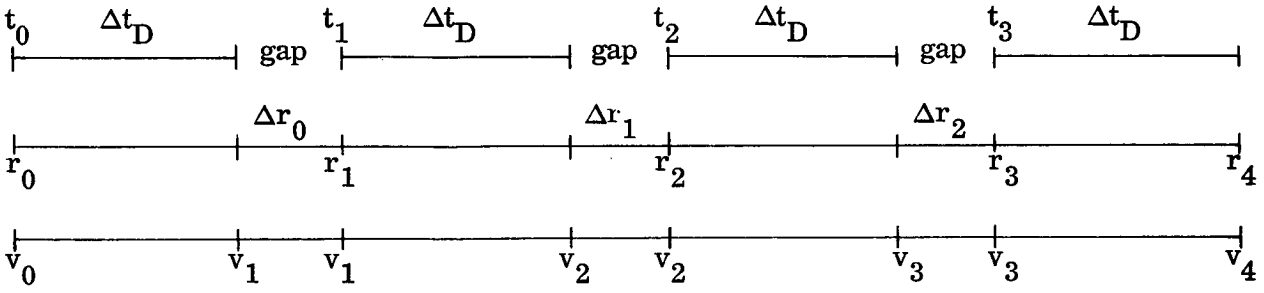
$$A = \frac{n_R \Delta t_D T}{\Delta t_{B_{n_R}} m_{n_r}} \tag{24}$$

WHERE ΔtB_{n_R} IS THE LENGTH OF THE BURNING PERIOD FOR THE TOTAL NUMBER OF RETROS FIRED.

SINCE THE RETROFIRING IS CONTROLLED BY A TIMER SEQUENCING THROUGH COMMUTATORS ASSOCIATED WITH THE INDIVIDUAL ROCKETS, AN ERROR IN THE TIMER COULD CAUSE A DEVIATION FROM THE NOMINAL SEQUENCE. OVERLAPPED FIRING COULD RESULT FROM A FAST TIMER. THRUST GAPS COULD RESULT FROM A SLOW TIMER, OR FROM THE MISFIRING OF ROCKET NO.3, OR NO.2. A CONTINUOUS THRUST STILL RESULTS IF ROCKET NO.1 OR NO. 4 MISFIRES SINCE THEY ARE, RESPECTIVELY, THE FIRST AND LAST ROCKETS TO FIRE.

LET US NOW CONSIDER THE POSITION AND VELOCITY DETERMINATION DUE TO THE THRUST OF THE ROCKETS WHEN THE FIRING

SEQUENCE DEVIATES FROM THE NORMAL. LET t_0, t_1, t_2, t_3 BE THE FIRING TIMES OF THE FIRST, SECOND, THIRD, AND FOURTH ROCKETS TO FIRE, AND LET r_0, r_1, r_2, AND r_3, v_0, v_1, v_2, AND v_3 BE THE POSTIIONS AND VELOCITIES ASSOCIATED WITH THOSE TIMES. THE POSTION AND VELOCITY AT BURNOUT (THE COMPLETION OF RETROFIRING) ARE PRESENTED BY r_4 AND v_4. AGAIN Δt_D REPRESENTS THE LENGTH OF THE BURNING PERIOD FOR EACH ROCKET. THE QUANTITY m IS USED TO REPRESENT THE AVERAGE MASS DURING THE BURNING PERIOD. WE CONSIDER THE CASE WHERE THRUST GAPS OCCUR, I.E., WHERE $t_{i+1} - t_i > \Delta t_D$.



WE KNOW THAT

$$v = v_0 + at$$

$$r = r_0 + v_0 t + \frac{1}{2} a t^2 \qquad (24.1)$$

SO,

$$v_1 = v_0 + \frac{T}{m}\Delta t_D$$

$$r_1 = r_0 + v_0\Delta t_D + \frac{1}{2}\frac{T}{m}\Delta t_D^{\,2} + \Delta r_0 \qquad \text{where } \Delta r_0 = v_1\left[t_1 - (t_0 + \Delta t_D)\right]$$

$$r_1 = r_0 + v_0\Delta t_D + \frac{1}{2}\frac{T}{m}\Delta t_D^{\,2} + v_1\left[t_1 - (t_0 + \Delta t_D)\right].$$

SUBSTITUTING FOR v_1 WE OBTAIN -

$$r_1 = r_0 + v_0(t_1 - t_0) + \frac{T}{m}\left[\Delta t_D(t_1 - t_0) - \frac{1}{2}\Delta t_D^{\,2}\right]$$

$$v_2 = v_1 + \frac{T}{m}\Delta t_D$$

$$= v_0 + 2\frac{T}{m}\Delta t_D$$

$$r_2 = r_1 + v_1\Delta t_D + \frac{1}{2}\frac{T}{m}\Delta t_D^{\,2} + \Delta r_1 \quad \text{where} \quad r_1 = v_2\left[t_2 - (t_1 + \Delta t_D)\right]$$

$$r_2 = r_1 + v_1\Delta t_D + \frac{1}{2}\frac{T}{m}\Delta t_D^{\,2} + v_2\left[t_2 - (t_1 + \Delta t_D)\right]$$

SUBSTITUTING FOR r_1 , v_1 , AND v_2 WE OBTAIN -

$$r_2 = r_0 + v_0(t_2 - t_0) + \frac{T}{m}\left[\Delta t_D(t_1 - t_0) + 2\,\Delta t_D(t_2 - t_1) - \Delta t_D{}^2\right]$$

$$v_3 = v_2 + \frac{T}{m}\Delta t_D$$

$$= v_0 + 3\,\frac{T}{m}\Delta t_D$$

$$r_3 = r_2 + v_2\Delta t_D + \frac{1}{2}\,\frac{T}{m}\,\Delta t_D{}^2 + \Delta r_2 \quad \text{where} \quad \Delta r_2 = v_3\left[t_3 - (t_2 + \Delta t_D)\right]$$

$$r_3 = r_2 + v_2\Delta t_D + \frac{1}{2}\,\frac{T}{m}\,\Delta t_D{}^2 + v_3\left[t_3 - (t_2 + \Delta t_D)\right]$$

SUBSTITUTING FOR r_2 , v_2 , AND v_3 WE OBTAIN

$$r_3 = r_0 + v_0(t_3 - t_0) + \frac{T}{m}\left[\Delta t_D(t_1 - t_0) + 2\,\Delta t_D(t_2 - t_1) + 3\,\Delta t_D(t_3 - t_2) - \frac{3}{2}\Delta t_D{}^2\right]$$

$$\left.\begin{aligned} v_4 &= v_3 + \frac{T}{m}\Delta t_D \\ &= v_0 + 4\,\frac{T}{m}\,\Delta t_D \\ r_4 &= r_3 + v_3\,\Delta t_D + \frac{1}{2}\,\frac{T}{m}\Delta t_D{}^2 \end{aligned}\right\} \qquad (25)$$

SUBSTITUTING FOR r_3 AND v_3 WE OBTAIN

$$r_4 = r_0 + v_0(t_3 - t_0 + \Delta t_D) + \frac{T}{m}$$

$$\left[\Delta t_D(t_1 - t_0) + 2\Delta t_D(t_2 - t_1) + 3\Delta t_D(t_3 - t_2) + 2\Delta t_D{}^2\right] \tag{26}$$

NOW WE LET $t_0 = 0, t_1 = 10.5, t_2 = 21, t_3 = 31.5$
SO THAT WE HAVE EQUAL THRUST GAPS, AND WE LET $\Delta t_D = 5.5$.
USING (25) AND (26) AS EXPRESSIONS FOR THE VELOCITY AND POSITION DETERMINATION AT BURNOUT, WE OBTAIN:

$$v_4 = v_0 + 22\,\frac{T}{m}$$

$$r_4 = r_0 + v_0(37) + \frac{T}{m}\left[5.5(10.5) + 11(10.5) + 16.5(10.5) + 60.5\right]$$

$$= r_0 + 37\,v_0 + 407\,\frac{T}{m}$$

IF WE USED THE MEAN ACCELERATION (24) AND THE EXPRESSIONS IN (24.1) TO DETERMINE THE POSITION AND VELOCITY AT BURN-OUT, WE OBTAIN -

$$v_4 = v_0 + \Delta t_{B_{n_R}} \quad (A)$$

$$= v_0 + \Delta t_{B_{n_R}} \frac{n_R \, \Delta t_D \, T}{t_{B_{n_R}} \, m_{n_R}}$$

$$= v_0 + n_R \, \Delta t_D \frac{T}{m_{n_R}}$$

$$= v_0 + 4(5.5) \frac{T}{m_{n_R}}$$

$$= v_0 + 22 \frac{T}{m_{n_R}}$$

$$r_4 = r_0 + v_0 \Delta t_{B_{n_R}} + \frac{1}{2} (A) \left(\Delta t_{B_{n_R}} \right)^2$$

$$= r_0 + v_0 \Delta t_{B_{n_R}} + \frac{1}{2} \frac{n_R \, \Delta t_D \, T}{\Delta t_{B_{n_R}} \, m_{n_R}} \left(\Delta t_{B_{n_R}} \right)^2$$

$$= r_0 + v_0 \Delta t_{B_{n_R}} + \frac{1}{2} \Delta t_{B_{n_R}} \, n_R \, \Delta t_D \, \frac{T}{m_{n_R}}$$

$$= r_0 + v_0 (37) + \frac{1}{2} (37) (4) (5.5) \frac{T}{m_{n_R}}$$

$$= r_0 + 37 \, v_0 + 407 \frac{T}{m_{n_R}}$$

THUS, WE SEE THAT WE HAVE OBTAINED THE SAME RESULTS FOR v_4 AND r_4 USING THE MEAN ACCELERATION (NOT TAKING INTO ACCOUNT THE INDIVIDUAL FIRING TIMES OF THE RETROS, BUT ONLY THE LENGTH OF THE ENTIRE BURNING PERIOD, $\Delta t_{B_{n_R}}$ AND USING THE DERIVED EXPRESSIONS WHICH TAKE INTO ACCOUNT THE INDIVIDUAL FIRING TIMES OF THE RETROS.

NOW WE LET $t_0 = 0$, $t_1 = 10.5$, $t_2 = 26$, $t_3 = 46.4$ SO THAT WE HAVE UNEQUAL THRUST GAPS AND LET Δt_D 5.5 AGAIN. USING OUR DERIVED EXPRESSIONS, (25) AND (26), WE OBTAIN -

$$v_4 = v_0 + 22 \frac{T}{m}$$

$$r_4 = r_0 + 52 v_0 + 627 \frac{T}{m}$$

USING THE MEAN ACCCLERATION (24) AND THE EXPRESSIONS IN (24.1) WE OBTAIN -

$$v_4 = v_0 + 22 \frac{T}{m}$$

$$r_4 = r_0 + 52 v_0 + 572 \frac{T}{m}$$

THUS, WE SEE THAT WHEN NON-SYMMETRIC FIRING GAPS EXIST, THE SOLUTION FOR THE POSITION USING THE MEAN ACCELERATION WILL DIFFER FROM THE DERIVED SOLUTION WHERE THE INDIVIDUAL FIRING TIMES OF THE RETROS ARE TAKEN INTO ACCOUNT.

THE SAME ERROR WILL BE FOUND TO EXIST FOR THE CASE WHEN NON-SYMMETRIC FIRING OVERLAPS OCCUR. TO ACCOUNT FOR THESE NON-SYMMETRIC CASES, WE HAVE DEVELOPED AN EXPRESSION FOR $f(\Delta t_i)$ WHICH TAKES INTO ACCOUNT THE INDIVIDUAL FIRING TIMES OF THE RETROROCKETS AND R_x^T, R_y^T, R_z^T ARE COMPUTED AS FOLLOWS -

$$R_x^T = \sum_{i=1}^{n_R} V_{x_i}^T \left[\frac{\Delta t_D}{2} + (t_L - t_i) \right]$$

$$R_y^T = \sum_{i=1}^{n_R} V_{y_i}^T \left[\frac{\Delta t_D}{2} + (t_L - t_i) \right]$$

$$R_z^T = \sum_{i=1}^{n_R} V_{z_i}^T \left[\frac{\Delta t_D}{2} + (t_L - t_i) \right]$$

WHERE

$$V_{x_i}^T = \frac{\Delta t_D \, T \cos c}{m_{n_R}}$$

$$V_{y_i}^T = \frac{\Delta t_D \, T \sin c}{m_{n_R}} \cos(\phi_i + \rho)$$

$$V_{z_i}^T = \frac{\Delta t_D \, T \sin c}{m_{n_R}} \sin(\phi_i + \rho)$$

AND t_1 IS THE FIRING TIME OF THE LAST RETRO TO FIRE. THE SAME EXPRESSIONS ARE USED FOR SYMMETRIC CASES.

METHOD

WHEN CONTROL IS TRANSFERRED TO C6RETC, THE CONTENTS OF INDEX REGISTERS 1 THROUGH 6 ARE SAVED. THE LOCATIONS OF THE INPUT AND OUTPUT VECTORS TABLES ARE PICKED UP AND THE TORQUE SETTING IS SAVED. THE PROGRAM HAS AN OPTION TO COMPUTE THE BURNOUT VECTOR FOR A NOMINAL OR NON-NOMINAL FIRING SEQUENCE. THE BRANCH FOR THE NOMINAL FIRING SEQUENCE ASSUMES THAT ALL FOUR RETROS HAVE FIRED IN ANTICIPATED ORDER (1, 3, 2, 4) AND WITH EQUAL FIRING TIMES OF 5.5 SECONDS PER RETRO. IT FURTHER ASSUMES A NOMINAL ATTITUDE AT THE BEGINNING OF RETROFIRE WHERE THE ROLL, PITCH AND YAW ANGLES ARE FIXED. THE NON-NOMINAL BRANCH ALLOWS FOR LESS THAN FOUR RETROS FIRING, FOR FIRING IN OTHER THAN NOMINAL SEQUENCE, AND ALSO FOR UNEQUALLY SPACED FIRING TIMES AND NON-NOMINAL ROLL, PITCH, AND YAW ANGLES. A TEST IS MADE ON A PARAMETER SPECIFIED IN THE CALLING SEQUENCE TO DETERMINE WHICH BRANCH IS TO BE TAKEN. IF NO RETROS HAVE FIRED, THE INPUT VECTORS ARE SUBSTITUTED FOR THE OUTPUT VECTORS. IF ANY RETRO HAS FIRED, THE BURNOUT VECTORS ARE COMPUTED UNLESS THERE IS AN ERROR RETURN FROM AM SUBROUTINE. THE BURNOUT VECTORS ARE PLACED IN THE OUTPUT VECTORS TABLE, THE INDEX REGISTERS ARE RESTORED, AND CONTROL IS TRANSFERRED BACK TO THE CALLING PROGRAM.

THE FOLLOWING EQUATIONS ARE USED FOR THE NON-NOMINAL CASE -

$$
\left.
\begin{aligned}
V_x^T &= n_R \frac{T_c}{m_n} \\
V_y^T &= \left[\sum_{i=1}^{n_R} \cos(\phi_i + \rho) \right] \frac{T_s}{m_{n_R}} g \\
V_z^T &= \left[\sum_{i=1}^{n_R} \sin(\phi_i + \rho) \right] \frac{T_s}{m_{n_R}} g \\
R_x^T &= K_1 \left[\sum_{i=1}^{n_R} \left(t_L - t_i + \frac{\Delta t_D}{2} \right) \right] \frac{T_c}{m_{n_R}} g \\
R_y^T &= K_1 \left[\sum_{i=1}^{n_R} \cos(\phi_i + \rho) \left(t_L - t_i + \frac{\Delta t_D}{2} \right) \right] \frac{T_s}{m_{n_R}} g \\
R_z^T &= K_1 \left[\sum_{i=1}^{n_R} \sin(\phi_i + \rho) \left(t_L - t_i + \frac{\Delta t_D}{2} \right) \right] \frac{T_s}{m_{n_R}} g
\end{aligned}
\right\} \quad (28)
$$

WHERE

$$m_{n_R} = w_c - \left(\frac{n_R}{2}\right)(w_f)$$

w_c IS THE WEIGHT OF THE CAPSULE AT RETROFIRE IN FLOATING POINT, POUNDS

w_c IS THE WEIGHT OF THE FUEL PER RETRO IN FLOATING POINT, POUNDS AND

g IS THE FACTOR TO CONVERT FT/SEC^2 TO GEMINI ACCELERATION UNITS.

T_c AND T_s ARE PROGRAM CONSTANTS DEFINED AS FOLLOWS -

$$T_c = K_1 \Delta t_D T \cos c$$

$$T_s = K_1 \Delta t_D T \sin c$$

T IS THE THRUST OF THE RETROROCKETS IN FLOATING POINT POUNDS, c IS THE CANT ANGLE OF THE RETROROCKETS, K_1 IS THE FACTOR TO CONVERT SECONDS TO GEMINI TIME UNITS, n_R IS THE TOTAL NUMBER OF ROCKETS, Δt_D IS THE LENGTH OF THE FIRING PERIOD FOR EACH RETRO IN FLOATING POINT SECONDS, t_L IS THE FIRING TIME OF THE LAST RETRO TO FIRE IN FLOATING POINTS SECONDS, GMT, t_f IS THE FIRING TIME OF THE FIRST RETRO TO FIRE IN FLOATING POINTS SECONDS, GMT, AND t_i IS THE FIRING TIME OF THE Lth RETROROCKET TO FIRE IN FLOATING POINTS SECONDS, GMT.

$$\alpha_1 = \cos\beta\, V_x^T - \sin\beta\, V_z^T$$

$$\alpha_2 = \cos\beta\, R_x^T - \sin\beta\, R_z^T$$

$$A_1 = \sin\beta\, V_x^T + \cos\beta\, V_z^T$$

$$A_2 = \cos\psi\,\alpha_1 + \sin\psi\; V_y{}^T$$

$$A_3 = -\sin\psi\,\alpha_1 + \cos\psi\; V_y{}^T$$

$$A_4 = K_1\,(t_L - t_f + \Delta t_D)$$

$$A_5 = \frac{K_1}{2}(t_L - t_f + \Delta t_D)$$

$$A_6 = t_L - t_f + \Delta t_D$$

$$B_1 = \sin\beta\; R_x{}^T + \cos\beta\; R_z{}^T$$

$$B_2 = \cos\psi\,\alpha_2 + \sin\psi\; R_y{}^T$$

$$B_3 = \sin\psi\,\alpha_2 + \cos\psi\; R_y{}^T$$

FOR THE NOMINAL CASE, THE ABOVE EQUATIONS (28) REDUCE TO THE FOLLOWING —

$$V_x{}^T = 4\,\frac{T_c}{m_{n_R}}\,g$$

$$V_y{}^T = 0$$

$$V_z{}^T = 0$$

$$R_x{}^T = 44K_1\,\frac{T_c}{m_{n_R}}\,g = 11K_1\,V_x{}^T$$

$$R_y^T = 0$$

$$R_z^T = 0$$

$$\alpha_1 = \cos\beta \; V_x^T$$

$$\alpha_2 = \cos\beta \; R_x^T$$

THEREFORE, WE NEED USE ONLY THE FOLLOWING -

$$B = V_x^T = 4 \frac{T_c}{m_{n_R}} g$$

$$A_1 = B \sin\beta$$

$$A_2 = - B \cos\beta$$

$$A_3 = 0$$

$$A_4 = 22K_1$$

$$A_5 = 11K_1$$

$$A_6 = 22$$

$$B_1 = \sin\beta \; R_x^T = \frac{A_1}{B}(11K_1 B) = 11K_1 A_1 = A_5 A_1$$

$$B_2 = - \cos\beta \; R_x^T = + \frac{A_2}{B}(11K_1 B) = 11K_1 A_2 = A_5 A_2$$

$$B_3 = 0$$

IT SHOULD BE NOTED HERE, THAT IF A NOMINAL RETRO-FIRING SEQUENCE IS USED IN THE NON-NOMINAL LOOP OF THE PROGRAM, RESULTS WILL DIFFER, SLIGHTLY, FROM THOSE WHICH WOULD BE OBTAINED BY RUNNING THE NOMINAL CASE THROUGH THE NOMINAL LOOP. THIS CAN BE ATTRIBUTED TO THE FACT THAT SMALL ERRORS ARE BUILT UP IN THE SUMMATIONS FOR V_y^T, V_z^T, AND R_y^T AND R_z^T WHICH SHOULD BE EXACTLY ZERO FOR A NOMINAL FIRING SEQUENCE.

THEN, FOR BOTH THE NOMINAL AND NON-NOMINAL CASES, THE FOLLOWING ARE COMPUTED.

$$r_o^2 = \bar{r}_o \cdot \bar{r}_o$$

$$|r_o| = \sqrt{r_o^2}$$

$$v_o^2 = \bar{v}_o \cdot \bar{v}_o$$

$$|v_o| = \sqrt{v_o^2}$$

$$\sin \gamma_o = \frac{(r_o \cdot v_o)}{|r_o||v_o|}$$

$$\cos \gamma_o = \sqrt{1 - \sin^2 \gamma_o}$$

$$\tan \gamma_o = \frac{\sin \gamma_o}{\cos \gamma_o}$$

$$\bar{h}_o = \bar{r}_o \times \bar{v}_o$$

$$h_o = |r_o||v_o| \cos \gamma_o$$

$$\Theta = \frac{v_o A_4 \cos \gamma_o + B_2}{r_o}$$

$$\Theta = \frac{v_o A_4 \cos \gamma_o + B_2}{r_o}$$

$$Q_1 = \frac{\sin \Theta}{\Theta}$$

$$Q_2 = \frac{(1 - \cos \Theta)}{\Theta}$$

$$F_1 = \frac{-A_4}{r_o^2}$$

$$F_2 = \frac{(-A_4)(A_5)}{r_o^2}$$

IF THE TORQUE OPTION IS BEING USED, THE FOLLOWING EQUATIONS ARE APPLIED TO OBTAIN -

$$L_1 = A_1 + F_1$$

$$L_2 = B_1 + F_2$$

$$V_\Theta = A_2 Q_1 + L_1 Q_2$$

$$V_r = L_1 Q_1 - A_2 Q_2$$

$$R_\Theta = B_2 Q_1 + L_2 Q_2$$

$$R_r = L_2 Q_1 - B_2 Q_2$$

AND IF THE NO TORQUE OPTION IS BEING USED -

$$W_\Theta = A_2 + F_1 Q_2$$

$$V_r = A_1 + F_1 Q_1$$

$$R_\Theta = B_2 + F_2 Q_2$$

$$R_R = B_1 + F_2 Q_1$$

FOR BOTH CASES, $R_N = B_3$ AND $V_N = A_3$.

FINALLY THE FOLLOWING EQUATIONS ARE APPLIED -

$$C_1 = \frac{V_r - V_\Theta \tan\gamma_o}{r_o}$$

$$C_2 = \frac{V_\Theta}{v_o \cos\gamma_o} + 1$$

$$C_3 = \frac{V_N}{h_o}$$

$$D_1 = \left(\frac{R_r - R_\Theta \tan\gamma_o}{r_o}\right) + 1$$

$$D_2 = A_4 + \frac{R_\Theta}{v_o \cos\gamma_o}$$

$$D_3 = \frac{R_N}{h_o}$$

THE BURNOUT VECTORS ARE OBTAINED USING -

$$\bar{r} = D_1\bar{r}_o + D_2\bar{v}_o + D_3\bar{h}_o$$

$$\bar{v} = C_1\bar{r}_o + C_2\bar{v}_o + C_3\bar{h}_o$$

USAGE

A) CALLING SEQUENCE

L	TSX	C6RETC,4
L+1	PZE	L(INPUT $t_o, \bar{r}_o, \bar{v}_o$,,L(OUTPUT $t, \bar{r}, \bar{v}$)
L+2	PZE	L(TYPE OF FIRING INDICATOR),, L(TORQUE SETTING)
L+3	PZE	L(NON-NOMINAL INPUT TABLE)
L+4	ERROR RETURN	
L+5	NORMAL RETURN	

A LOCATION NEED BE SPECIFIED IN L+3 OF THE CALLING SEQUENCE ONLY IF A NON-NOMINAL FIRING SEQUENCE IS SPECIFIED IN L+2.

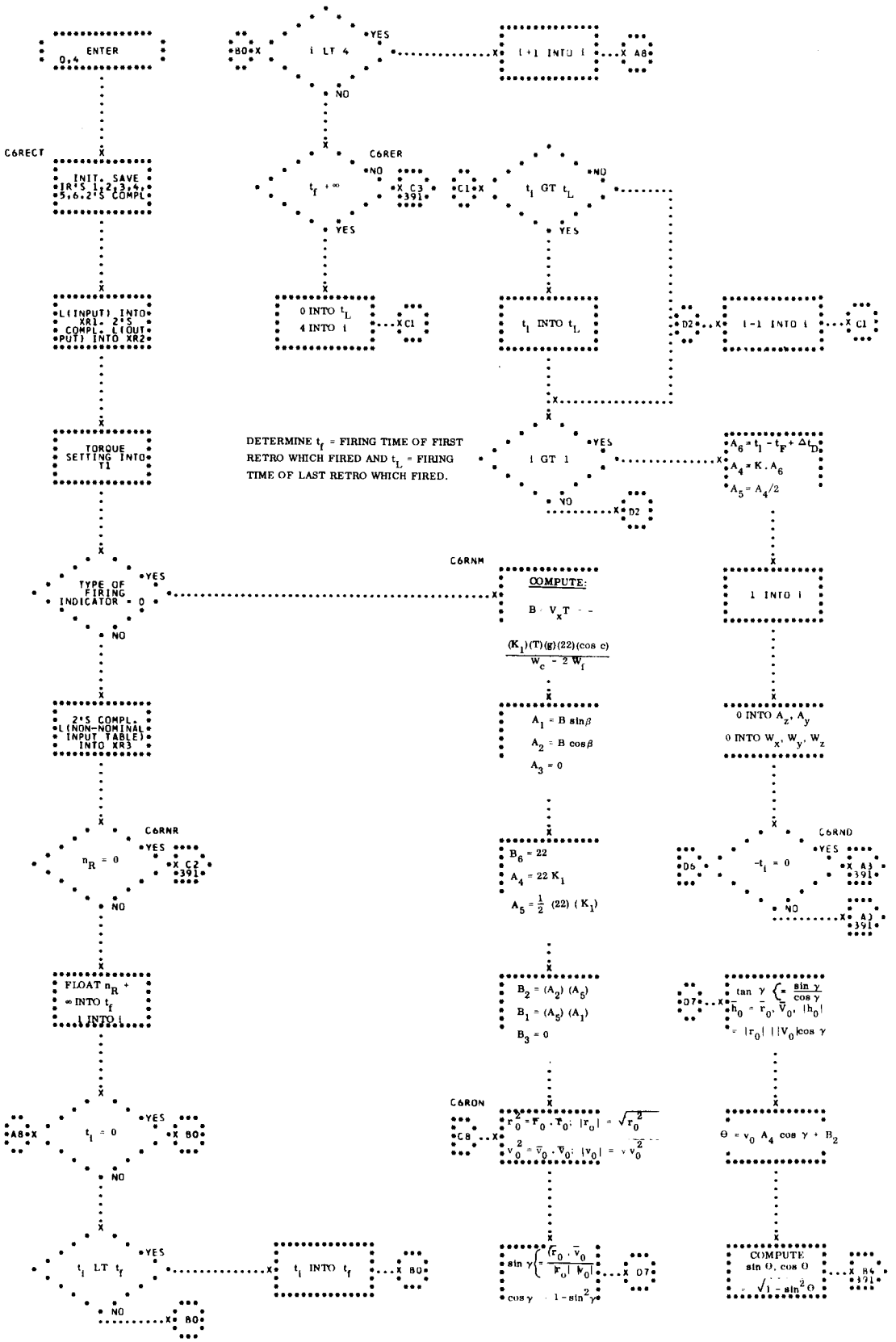
B) STORAGE REQUIRED -

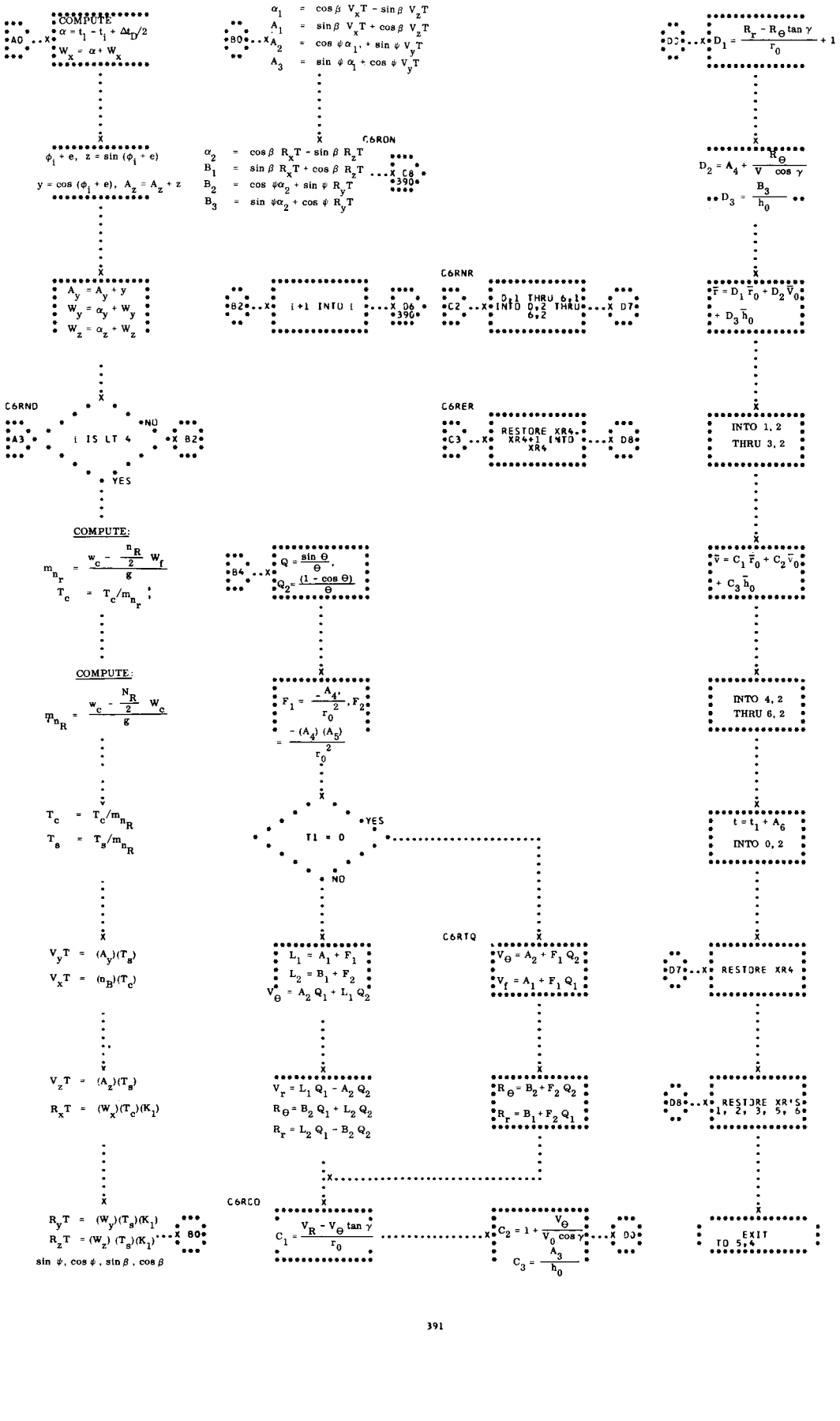
493 DECIMAL LOCATIONS EXCLUDING SUBROUTINES, AND EXTERNAL CONSTANTS AND PARAMTERS.

C) TIME REQUIRED (7094 MOD 1 MAXIMUM) -

NOMINAL (NO TORQUE) -	2.18 MILLISECONDS
NOMINAL (WITH TORQUE) -	2.26 MILLISECONDS
NON-NOMINAL (NO TORQUE) -	6.12 MILLISECONDS

NON-NOMINAL (WITH TORQUE) - 6.22 MILLISECONDS


ENTER 0,4
i LT 4
YES
NO
i+1 INTO i
A8
C6RECT
INIT. SAVE IR'S 1,2,3,4, 5,6,2'S COMPL
C6RER
$t_f = \infty$
C3 391
t_l GT t_L
C1
L(INPUT) INTO XR1. 2'S COMPL. L(OUTPUT) INTO XR2
0 INTO t_L
4 INTO i
C1
t_l INTO t_L
D2
i-1 INTO i
C1
DETERMINE t_f = FIRING TIME OF FIRST RETRO WHICH FIRED AND t_L = FIRING TIME OF LAST RETRO WHICH FIRED.
TORQUE SETTING INTO Y1
i GT 1
$A_6 = t_l - t_F + \Delta t_D$
$A_4 = K \cdot A_6$
$A_5 = A_4/2$
D2
C6RNM
TYPE OF FIRING INDICATOR = 0
COMPUTE:
$B \cdot V_x T - -$
$\frac{(K_1)(T)(g)(22)(\cos c)}{W_c - 2\dot{W}_f}$
1 INTO i
2'S COMPL. L(NON-NOMINAL INPUT TABLE) INTO XR3
$A_1 = B \sin\beta$
$A_2 = B \cos\beta$
$A_3 = 0$
0 INTO A_z, A_y
0 INTO W_x, W_y, W_z
C6RNR
$n_R = 0$
C2 391
$B_6 = 22$
$A_4 = 22 K_1$
$A_5 = \frac{1}{2}$ (22) (K_1)
C6RND
$-t_l = 0$
D5
A3 391
A3 391
FLOAT n_R
∞ INTO t_f
1 INTO i
$B_2 = (A_2)(A_5)$
$B_1 = (A_5)(A_1)$
$B_3 = 0$
D7
$\tan\gamma \left(= \frac{\sin\gamma}{\cos\gamma}\right) = \frac{\bar{r}_0 \cdot \bar{V}_0, |\bar{h}_0|}{|\bar{r}_0|\,|\bar{V}_0|\cos\gamma}$
A8
$t_l = 0$
B0
C6RON
C8
$r_0^2 = \bar{r}_0 \cdot \bar{r}_0$; $|r_0| = \sqrt{r_0^2}$
$v_0^2 = \bar{v}_0 \cdot \bar{v}_0$; $|v_0| = \sqrt{v_0^2}$
$\Theta = v_0 A_4 \cos\gamma + B_2$
t_l LT t_f
t_l INTO t_f
B0
$\sin\gamma \left(= \frac{\bar{r}_0 \cdot \bar{v}_0}{|\bar{r}_0|\,|\bar{v}_0|}\right)$
D7
$\cos\gamma = \sqrt{1-\sin^2\gamma}$
COMPUTE sin Θ, cos Θ
$= \sqrt{1-\sin^2\Theta}$
B4 391


A0
COMPUTE
$\alpha = t_1 - t_i + \Delta t_D/2$
$W_x = \alpha + W_x$
$\phi_i + e$, $z = \sin(\phi_i + e)$
$y = \cos(\phi_i + e)$, $A_z = A_z + z$
$A_y = A_y + y$
$W_y = \alpha_y + W_y$
$W_z = \alpha_z + W_z$
C6RND
A3
i IS LT 4
NO
YES
B2
COMPUTE:
$m_{n_r} = \frac{w_c - \frac{n_R}{2} W_f}{g}$
$T_c = T_c/m_{n_r}$
COMPUTE:
$m_{n_R} = \frac{w_c - \frac{N_R}{2} W_c}{g}$
$T_c = T_c/m_{n_R}$
$T_s = T_s/m_{n_R}$
$V_yT = (A_y)(T_s)$
$V_xT = (m_B)(T_c)$
$V_zT = (A_z)(T_s)$
$R_xT = (W_x)(T_c)(K_1)$
$R_yT = (W_y)(T_s)(K_1)$
$R_zT = (W_z)(T_s)(K_1)$
X 80
$\sin\psi$, $\cos\psi$, $\sin\beta$, $\cos\beta$
B0
$\alpha_1 = \cos\beta\ V_xT - \sin\beta\ V_zT$
$A_1 = \sin\beta\ V_xT + \cos\beta\ V_zT$
$A_2 = \cos\psi\alpha_1 + \sin\psi\ V_yT$
$A_3 = \sin\psi\alpha_1 + \cos\psi\ V_yT$
C6RON
$\alpha_2 = \cos\beta\ R_xT - \sin\beta\ R_zT$
$B_1 = \sin\beta\ R_xT + \cos\beta\ R_zT$
$B_2 = \cos\psi\alpha_2 + \sin\psi\ R_yT$
$B_3 = \sin\psi\alpha_2 + \cos\psi\ R_yT$
X C8
B2
i+1 INTO i
X 06
390
B4
$Q = \frac{\sin\Theta}{\Theta}$,
$Q_2 = \frac{(1 - \cos\Theta)}{\Theta}$
$F_1 = \frac{-A_4}{r_0^2}$, $F_2 = \frac{-(A_4)(A_5)}{r_0^2}$
T1 = 0
YES
NO
$L_1 = A_1 + F_1$
$L_2 = B_1 + F_2$
$V_\Theta = A_2 Q_1 + L_1 Q_2$
$V_r = L_1 Q_1 - A_2 Q_2$
$R_\Theta = B_2 Q_1 + L_2 Q_2$
$R_r = L_2 Q_1 - B_2 Q_2$
C6RCO
$C_1 = \frac{V_R - V_\Theta \tan\gamma}{r_0}$
C6RNR
C2
D+1 THRU 6,1
INTO D,2 THRU 6,2
X D7
C6RER
C3
RESTORE XR4
XR4+1 INTO XR4
X D8
C6RTQ
$V_\Theta = A_2 + F_1 Q_2$
$V_f = A_1 + F_1 Q_1$
$R_\Theta = B_2 + F_2 Q_2$
$R_r = B_1 + F_2 Q_1$
$C_2 = 1 + \frac{V_\Theta}{V_0 \cos\gamma}$
$C_3 = \frac{A_3}{h_0}$
X D0
D0
$D_1 = \frac{R_r - R_\Theta \tan\gamma}{r_0} + 1$
$D_2 = A_4 + \frac{R_\Theta}{V \cos\gamma}$
$D_3 = \frac{B_3}{h_0}$
$\bar{r} = D_1 \bar{r}_0 + D_2 \bar{v}_0 + D_3 \bar{h}_0$
$\bar{v} = C_1 \bar{r}_0 + C_2 \bar{v}_0 + C_3 \bar{h}_0$
INTO 1, 2
THRU 3, 2
INTO 4, 2
THRU 6, 2
$t = t_1 + A_6$
INTO 0, 2
D7
RESTORE XR4
D8
RESTORE XR'S
1, 2, 3, 5, 6
EXIT
TO 5,4

43. CTSFT3 - COMPUTATION OF TIME STEP FOR TYPE 3 INTEGRATION.

GIVEN A t, $\overline{r}$, $\overline{v}$, THIS PROGRAM COMPUTES THE TIME STEP, h, TO BE USED IN TYPE 3 INTEGRATION FOR THE LAUNCH IMPACT PREDICTORS CIMPRK AND CBIPRE. IT IS ALSO USED BY ABORT PROGRAM CHABRT. (SEE FLOWCHART.)

INPUT

A) A SEVEN-LOCATION TABLE CONTAINING T IN FLOATING POINT SECONDS AND THE COMPONENTS OF $\overline{r}$ AND $\overline{v}$ IN FLOATING POINT GEMINI UNITS IS INPUT TO THE PROGRAM.

B) SUBROUTINE BOSCER AND LIBRARY SUBROUTINES U3VMAG AND U1SQRT ARE USED.

C) CONSTANTS -

K00.07	DEC	.07	FLOATING POINT SEVEN-HUNDREDTHS
K000.2	DEC	.2	FLOATING POINT TWO-TENTHS
K00.95	DEC	.95	FLOATING POINT NINETY-FIVE HUNDREDTHS
K001.0	DEC	1.0	FLOATING POINT ONE
K002.0	DEC	2.0	FLOATING POINT TWO
K010.0	DEC	10.0	FLOATING POINT TEN
K030.0	DEC	30.0	FLOATING POINT THIRTY

K040.0	DEC	40.0	FLOATING POINT FORTY
K0060.	DEC	60.0	FLOATING POINT SIXTY
K06.E4	DEC	.0028672824	FLOATING POINT 60,000 FEET IN GEMINI LENGTH UNITS

OUTPUT

TRKO1H CONTAINS THE TIME STEP, h, AND TRKO2H CONTAINS TWICE THE TIME STEP, 2h. ALL OUTPUT IS IN FLOATING POINT SECONDS.

METHOD

IT IS NECESSARY TO TAKE INTO ACCOUNT THE GENERAL ''SHAPE'' (ECCENTRICITY) OF THE TRAJECTORY IN COMPUTING A TIME STEP FOR THE SUCCESFUL EXECUTION OF NUMERICAL INTEGRATION. AT TIMES WHEN THE ECCENTRICITY IS CLOSE TO 1.0, THE TRAJECTORY APPROACHES A PARABOLA, AND A SMALL TIME STEP IS NEEDED TO GET AROUND THE BEND (VERTEX) OF THE TRAJECTORY. HOWEVER, WHEN THE ECCENTRICITY APPROACHES ZERO AND THE TRAJECTORY IS NEARLY CIRCULAR, IT IS POSSIBLE TO USE A LARGER TIME STEP.

TYPE 3 INTEGRATION AS USED IN CIMPRK AND CBIPRE REQUIRES THAT THREE VECTORS BELOW A CRITERION HEIGHT (CURRENTLY 400,000 FEET) BE COMPUTED IF THE TRAJECTORY DECENDS BELOW THIS HEIGHT. IF TOO LARGE A TIME STEP IS CHOSEN FOR A TRAJECTORY HAVING A FAIRLY LARGE ECCENTRICITY, THESE THREE VECTORS BELOW THE CRITERION HEIGHT COULD EXTEND FAR DOWN INTO THE ATMOSPHERE RESULTING IN POOR ACCURACY. THUS, ANOTHER CONSTRAINT IS PUT ON THE CHOICE

OF TIME STEPS.

FIRST, A CHECK IS MADE ON THE HEIGHT H, OF THE VECTOR WITH WHICH THE PROGRAM IS ENTERED. IF THIS HEIGHT ($H = |r| - R_e$) IS LESS THAN OR EQUAL TO 60,000 FEET, A TWO-SECOND TIME STEP IS CHOSEN AND AN EXIT IS MADE FROM THE PROGRAM.

FOR THOSE VECTORS WHOSE HEIGHTS ARE ABOVE 60,000 FEET, THE ECCENTRICITY OF THE TRAJECTORY IS COMPUTED USING THE FOLLOWING FORMULA -

$$e = \sqrt{\frac{(\bar{r} \cdot \bar{v})^2}{2 - |\bar{r}|\,|\bar{v}|^2} + (1 - |\bar{r}|\,|\bar{v}|^2)^2}$$

THE TIME STEP h IS CHOSEN AS FOLLOWS - IF

$e \geq .95$, h = 10 seconds

$.95 > e \geq .2$, h = 30 seconds

$.2 > e \geq .07$, h = 40 seconds

$.07 > e$, h = 60 seconds

USAGE

A) CALLING SEQUENCE -

```
TSX     CTSFT3,4
PZE     L(INPUT),,**
ERROR   RETURN
NORMAL  RETURN
```

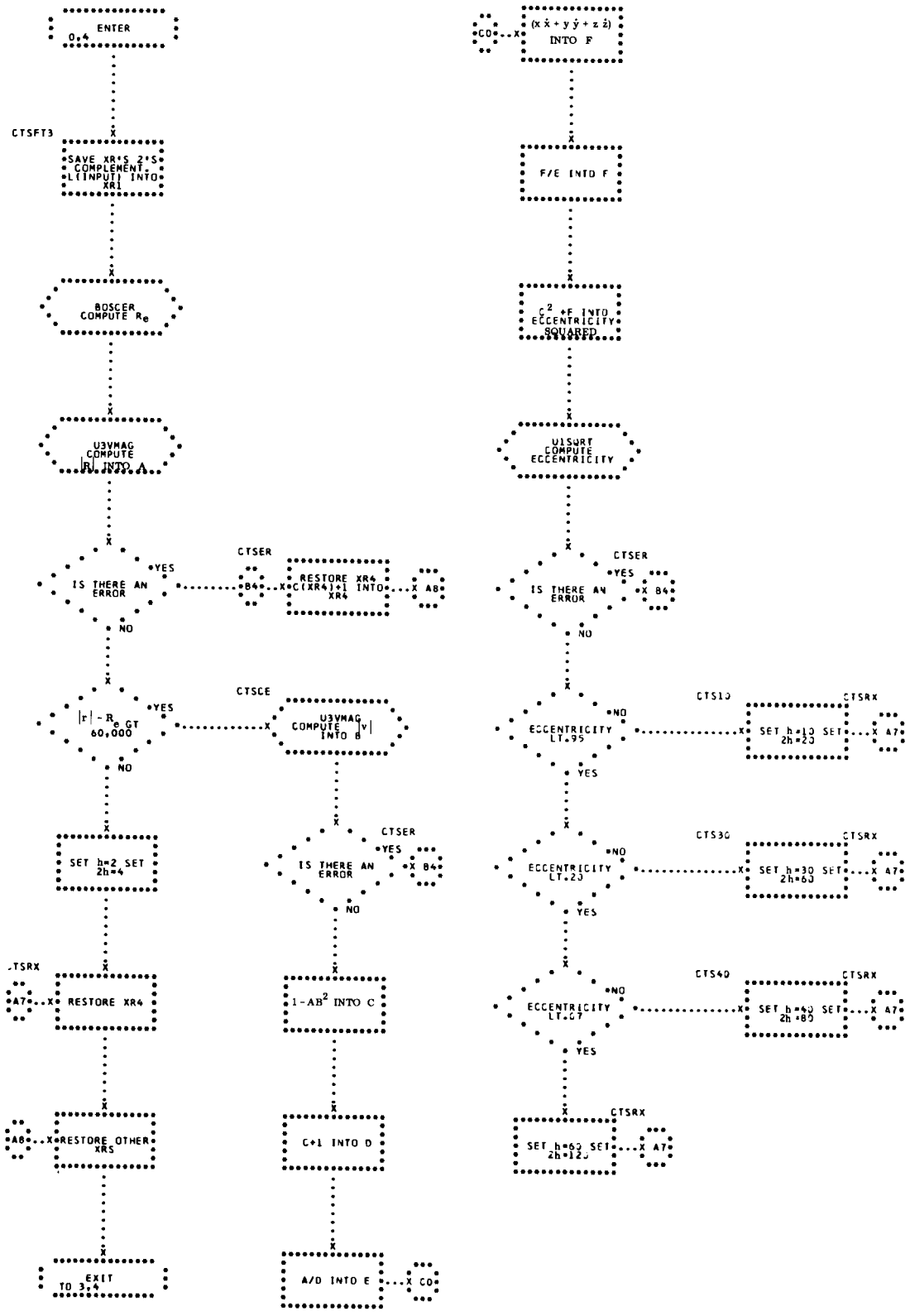
B) STORAGE REQUIRED -

85 LOCATIONS EXLCUDING STORAGE REQUIRED BY EXTERNAL CONSTANTS AND SUBROUTINES.

C) TIME REQUIRED (7094 MAXIMUM) -

1.390 MILLISECONDS INCLUDING MAXIMUM TIME REQUIRED BY ALL SUBROUTINES.

D) ERROR CONDITIONS -
AN ERROR RETURN IS A RESULT OF AN ERROR FROM A LIBRARY SUBROUTINE.


ENTER 0,4
CTSFT3
SAVE XR4;S 2*S COMPLEMENT L(INPUT) INTO XR1
BOSCER COMPUTE R_e
U3VMAG COMPUTE |R| INTO A
IS THERE AN ERROR
YES
CTSER
B4
RESTORE XR4 C(XR4)+1 INTO XR4
A8
NO
|r| - R_e GT 60,000
YES
CTSCE
U3VMAG COMPUTE |v| INTO B
CTSER
IS THERE AN ERROR
YES
B4
NO
SET h=2 SET 2h=4
CTSRX
A7
RESTORE XR4
A8
RESTORE OTHER XRS
EXIT TO 3,4
1-AB^2 INTO C
C+1 INTO D
A/D INTO E
C0
C0
(x ẋ + y ẏ + z ż) INTO F
F/E INTO F
c^2 *F INTO ECCENTRICITY SQUARED
U1SQRT COMPUTE ECCENTRICITY
CTSER
IS THERE AN ERROR
YES
B4
NO
CTS10
ECCENTRICITY LT.95
NO
SET h=10 SET 2h=20
CTSRX
A7
YES
CTS30
ECCENTRICITY LT.23
NO
SET h=30 SET 2h=60
CTSRX
A7
YES
CTS40
ECCENTRICITY LT.07
NO
SET h=40 SET 2h=80
CTSRX
A7
YES
CTSRX
SET h=60 SET 2h=120
A7

44. C6PTLU - LAUNCH PROGRAM FOR FOUR POINT LAGRANGIAN INTERPOLATION OF TIME

GIVEN A HEIGHT FOR WHICH THE ASSOCIATED TIME IS DESIRED, THIS PROGRAM USES THE VALUES IN THE LAUNCH NUMERICAL INTEGRATION HEIGHT TABLE TO DETERMINE THE REQUIRED TIME AND PLACES IT IN THE OUTPUT CELL SPECIFIED. (SEE FLOWCHART)

INPUT

A) TABLES

TCRKOH		THE LAUNCH NUMERICAL INTEGRATION HEIGHT TABLE ARRANGED AS FOLLOWS -
	TCRKOH	t_{START}- THE TIME ASSOCIATED WITH THE FIRST HEIGHT IN THE TABLE IN GMT FLOATING POINT SECONDS
	+1	t_{STOP} - THE TIME ASSOCIATED WITH THE LAST HEIGHT IN THE TABLE IN GMT FLOATING POINT SECONDS
	+2	H_1
	+3	H_2
	+4	H_3, ETC. - WHERE H_1 IS THE GEODETIC HEIGHT OF THE SPACECRAFT IN FLOATING POINT GEMINI LENGTH UNITS
TCRK1H		PZE TCRKOH+6
TRKO1H		INTERVAL BETWEEN HEIGHT ENTRIES IN TCRKOH IN FLOATING POINT SECONDS

B) CONTAINS -

K001.0	DEC 1.0 FLOATING POINT ONE
KM03.0	DEC -3.0 FLOATING POINT MINUS THREE

KCH233 OCT 233000000000

CHARACTERISTIC OF 233 OCTAL

C) UPON ENTRY TO C6PTLU, THE AC MUST CONTAIN THE HEIGHT FOR WHICH THE ASSOCIATED TIME IS DESIRED IN FLOATING POINT GEMINI LENGTH UNITS.

OUTPUT

A) TABLE

TC6PTD A ONE-CELL TABLE CONTAINING THE TIME ASSOCIATED WITH THE GIVEN HEIGHT IN GMT FLOATING POINT SECONDS

METHOD

LAGRANGE'S INTERPOLATION FORMULA FOR UNEQUAL INTERVALS

THE LAUNCH NUMERICAL INTEGRATION HEIGHT TABLE, TCRKOH, IS BASED ON TIME. THE HEIGHTS ARE ENTERED AT EQUAL TIME INTERVALS. HOWEVER, THE DISTANCE IN GEMINI LENGTH UNITS BETWEEN ANY TWO CONSECUTIVE HEIGHTS IS NOT CONSTANT. SINCE, TO INTERPOLATE FOR TIME, THE TABULATED VALUES OF THE HEIGHTS MUST BE USED, IT IS NECESSARY TO APPLY LAGRANGE'S INTERPOLATION FORMULA FOR UNEQUAL INTERVALS OF THE INDEPENDENT VARIABLE. THE FORMULA PASSES AN nth DEGREE POLYNOMIAL THROUGH (N+1) CONSECUTIVE TABULATED POINTS. IT MAY BE WRITTEN AS FOLLOWS -

$$f(X) = \frac{(X-X_1)(X-X_2)\ldots(X-X_n)}{(X_0-X_1)(X_0-X_2)\ldots(X_0-X_n)} f(X_0) + \frac{(X-X_0)(X-X_2)\ldots(X-X_n)}{(X_1-X_0)(X_i-X_2)\ldots(X_1-X_n)} f(X_1)$$

$$+ \ldots + \frac{(X-X_0)(X-X_1)\ldots(X-X_{n-1})}{(X_n-X_0)(X_n-X_1)\ldots(X_n-X_{n-1})} f(X_n)$$

WHERE

X IS THE GIVEN VALUE OF THE INDEPENDENT VARIABLE FOR WHICH THE ASSOCIATED DEPENDENT VARIABLE IS BEING INTERPOLATED.

X_i ARE THE TABULAR VALUES OF THE INDEPENDENT VARIABLE.

$F(X_i)$ ARE THE ASSOCIATED TABULAR VALUES OF THE DEPENDENT VARIABLE.

$F(X)$ IS THE FINAL INTERPOLATED DEPENDENT VARIABLE.

APPLICATION OF THE INTERPOLATION FORMULA IN C6PTLU

IN C6PTLU, THE ATTEMPT IS MADE TO PASS A 3RD DEGREE POLYNOMIAL THROUGH THE LAST FOUR POINTS IN THE TABLE. THUS, THE FOLLOWING REVISION IN DEFINITIONS APPLIES TO THE VARIABLES -

$X = H_d$, THE DESIRED HEIGHT

$X_i = H_i$, THE HEIGHTS ENTERED IN THE TCRKOH TABLE

$F(X_i) = t_i$, THE TIMES ASSOCIATED WITH THE HEIGHTS IN THE TCRKOH TABLE $n = 3$, THE ORDER OF THE POLYNOMIAL BEING USED

i VARIES FROM 1 TO 4

DETAILS OF METHOD OF C6PTLU

AS SOON AS C6PTLU RECEIVES CONTROL, IT SAVES THE DESIRED HEIGHT, H_d, IN INTERNAL STORAGE. THEN, USING THE TIME ASSOCIATED WITH THE LAST HEIGHTS IN THE TCRKOH TABLE, C6PTLU DETERMINES THE TIME OF THE FIRST HEIGHT TO BE USED IN THE INTERPOLATION AS FOLLOWS -

$t_1 = t_{stop} - 3h$ WHERE h IS THE TIME INTERVAL BETWEEN ENTRIES IN THE TABLE.

HAVING COMPUTED THIS TIME, IT IS POSSIBLE TO DETERMINE HOW MANY HEIGHTS PRECEDE THE FIRST HEIGHT TO BE USED -

N = THE INTEGRAL PORTION, IN FIXED POINT, OF $(t_1 - t_{start})/h$

THE PRESUMPTIVE ADDRESS (I.E., THE ADDRESS OF THE LOCATION FOLLOWING THE LAST HEIGHT IN THE TABLE) IS COMPUTED USING N -

P.A. = TCRKOH + 6 + N

C6PTLU THEN DETERMINES WHETHER THE DESIRED HEIGHT FALLS WITHIN THE RANGE OF THE FIRST AND SECOND HEIGHTS TO BE USED. IF THE DESIRED HEIGHT IS GREATER THAN THE FIRST HEIGHT TO BE USED OR LESS THAN THE SECOND HEIGHT TO BE USED, THE PROGRAM EXITS VIA THE ERROR RETURN. IF THE DESIRED HEIGHT IS EQUAL TO THE FIRST OR SECOND HEIGHT, THE TIME ASSOCIATED WITH THIS HEIGHT IS COMPUTED -

$H_d = H_1$, then t_1 already computed

$H_d = H_2$, then $t_2 = t_1 + h$

THIS TIME IS THEN STORED AS THE TIME ASSOCIATED WITH THE DESIRED HEIGHT IN TC6PTD AND C6PTLU EXITS VIA THE NORMAL RETURN.

IF NEITHER OF THE ABOVE CASES HOLDS, THE FOLLOWING COMPUTATIONS ARE MADE -

A) $P_1 = (H_d - H_1)(H_d-H_2)(H_d-H_3)(H_d-H_4)$

B) FOR EACH H_i (i VARYING FROM 1 TO 4) -

C) SINCE $(H_i - H_i = 0$, THIS TERM MUST BE EXCLUDED FROM P_2. THE CORRESPONDING TERM, $(H_d - H_i)$, MUST BE EXCLUDED, THEREFORE, FROM P_1. -

D) THE TIME ASSOCIATED WITH H_i IS OBTAINED -
$t_i = t_i + (i - 1)\,h$

E) TO OBTAIN THE LATEST ESTIMATE OF THE DESIRED TIME - t_d

$$t_{d\ new} = t_{d\ old} + (P_3/P_2)\ (t_i)$$

AFTER STEPS B) THROUGH E) HAVE BEEN COMPLETED FOR EACH HEIGHT USED IN THE INTERPOLATION PROCESS, THE FINAL TIME OBTAINED IS STORED AS THE TIME ASSOCIATED WITH THE DESIRED HEIGHT AND C6PTLU EXITS VIA THE NORMAL RETURN.

USAGE

CALLING SEQUENCE -

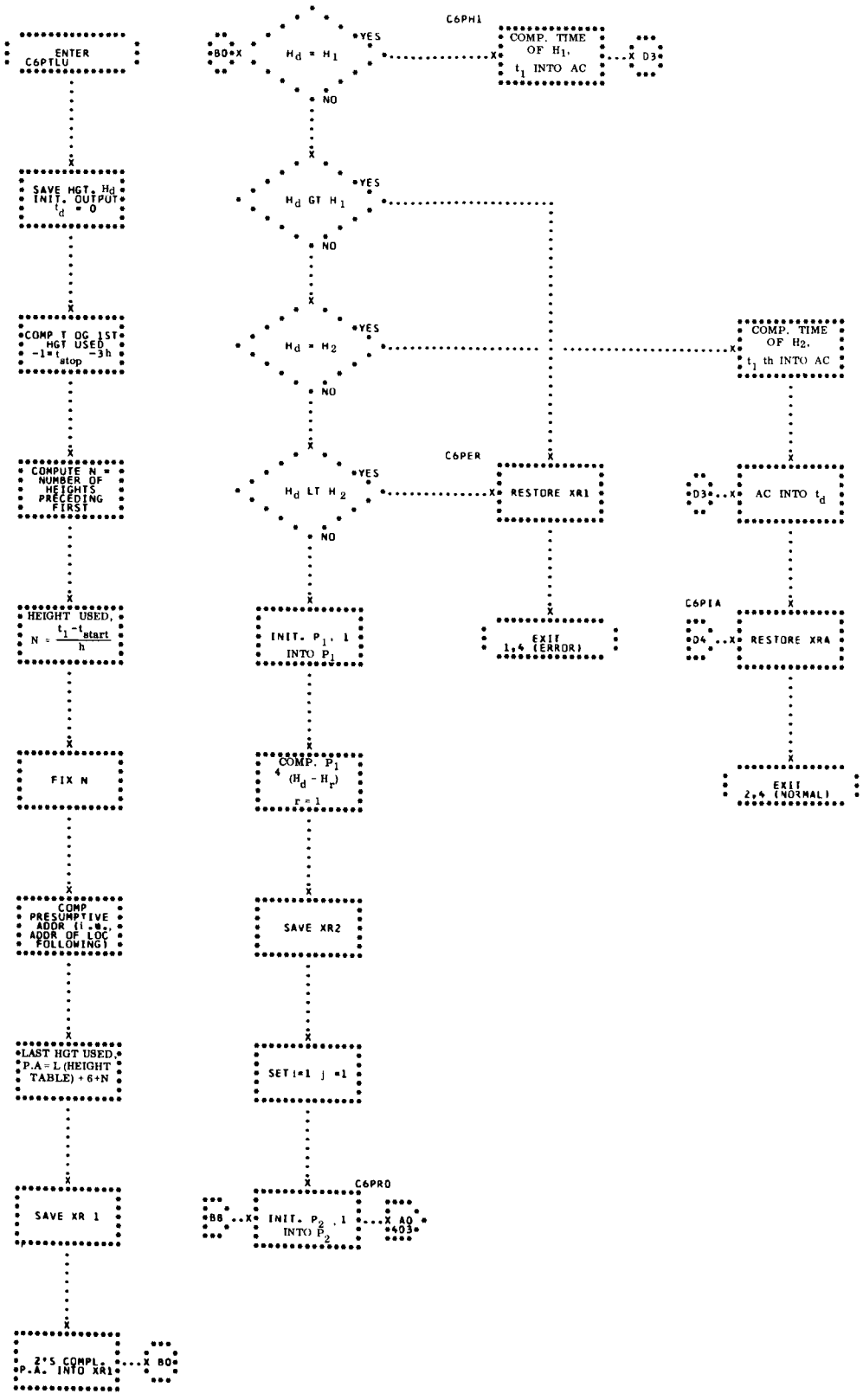
C(AC) = H_d

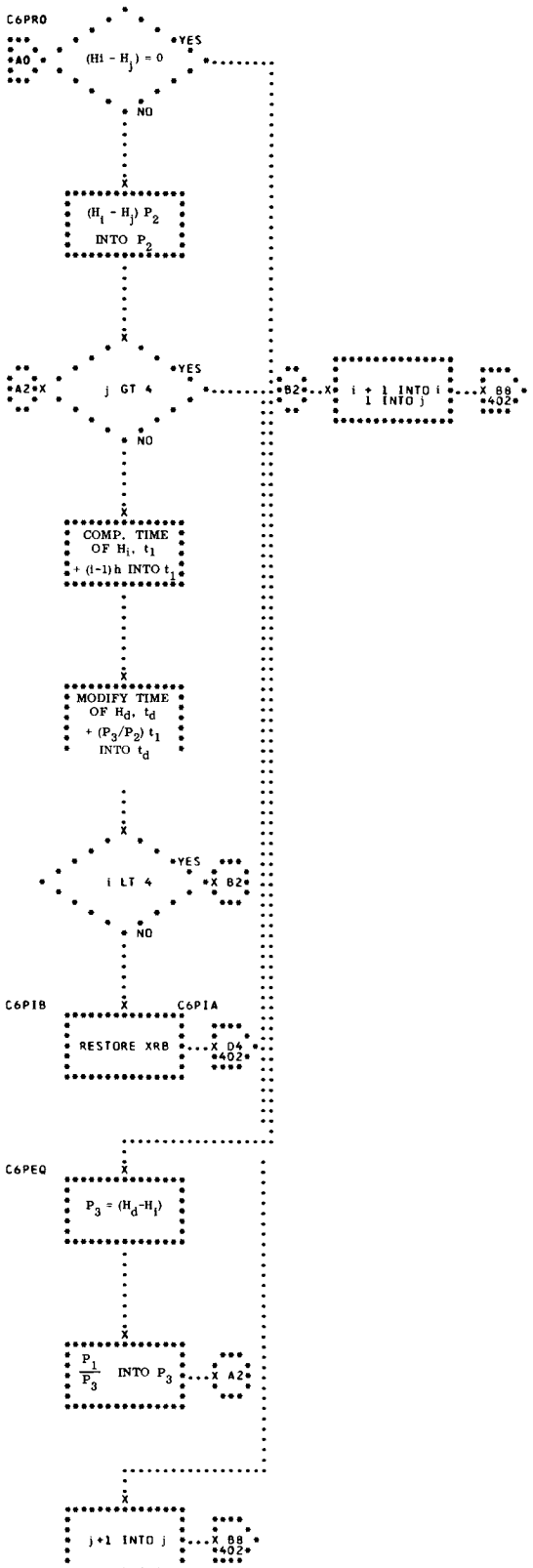
```
TSX     C6PTLU,4
ERROR RETURN
NORMAL RETURN
```

B) AN ERROR RETURN IMPLIES THAT THE DESIRED HEIGHT FELL OUTSIDE THE RANGE OF THE FIRST AND SECOND HEIGHTS TO BE USED IN THE INTERPOLATION PROCESS.

C) STORAGE REQUIRED - 84 LOCATIONS

D) TIMING (7094) - .164 MILLISECONDS, MINIMUM
1.264 MILLISECONDS, MAXIMUM


ENTER C6PTLU
SAVE HGT. Hd INIT. OUTPUT td = 0
COMP T OG 1ST HGT USED -1=tstop -3h
COMPUTE N = NUMBER OF HEIGHTS PRECEDING FIRST
HEIGHT USED, N = (t1 - tstart)/h
FIX N
COMP PRESUMPTIVE ADDR (i.e., ADDR OF LOC FOLLOWING)
LAST HGT USED, P.A. = L (HEIGHT TABLE) + 6 + N
SAVE XR 1
2'S COMPL. P.A. INTO XR1
B0
Hd = H1
YES
NO
C6PH1
COMP. TIME OF H1, t1 INTO AC
D3
Hd GT H1
Hd = H2
Hd LT H2
C6PER
RESTORE XR1
EXIT 1,4 (ERROR)
INIT. P1, 1 INTO Pi
COMP. P1 4 (Hd - Hr) r=1
SAVE XR2
SET i=1 j=1
B8
INIT. P2, 1 INTO P2
C6PRO
A0
403
COMP. TIME OF H2, t1 th INTO AC
D4
AC INTO td
C6PIA
RESTORE XR4
EXIT 2,4 (NORMAL)


C6PRO
A0
(H1 - Hj) = 0
YES
NO
(Hi - Hj) P2
INTO P2
A2
j GT 4
YES
NO
B2
i + 1 INTO i
1 INTO j
X B8
402
COMP. TIME
OF Hi, t1
+ (i-1)h INTO t1
MODIFY TIME
OF Hd, td
+ (P3/P2) t1
INTO td
i LT 4
YES
X B2
NO
C6PIB
C6PIA
RESTORE XRB
X D4
402
C6PEQ
P3 = (Hd-Hi)
P1/P3 INTO P3
X A2
j+1 INTO j
X B8
402

45. C6PTLI - LAUNCH PROGRAM FOR SIX POINT INTERPOLATION OF POSITION AND VELOCITY VECTORS

GIVEN A TIME AT WHICH THE INTERPOLATED POSITION AND VELOCITY VECTORS ARE DESIRED, C6PTLI USES THE VALUES IN THE NUMERICAL INTEGRATION VECTOR TABLE TO DETERMINE THE REQUIRED VECTORS AND PLACES THEM IN THE SPECIFIED OUTPUT TABLE. (SEE FLOWCHART.)

C6PTLI IS AN ADAPTATION OF THE LIBRARY SUBROUTINE, U7INTP.

INPUT

A) CONSTANTS -

K00000	DEC 0	FIXED POINT ZERO
K00006	DEC 6	FIXED POINT SIX
K00038	DEC 38	FIXED POINT THIRTY-EIGHT
KCH233	OCT 233000000000	CHARACTERISTIC OF 233 OCTAL
K002.0	DEC 2.0	FLOATING POINT TWO

B) TABLES

L(r,v, TABLE) THE ADDRESS OF THE NUMERICAL INTEGRATION TABLE OF POSITION AND VELOCITY VECTORS ARRANGED AS FOLLOWS - T_{start} , T_{stop} , $\dot{x}, \dot{y}, \dot{z}$, $\dot{x}, \dot{y}, \dot{z}$, x, y, z, $\dot{x}, \dot{y}, \dot{z}$, etc., WHERE t_{start} IS THE TIME ASSOCIATED WITH THE FIRST POSITION, VELOCITY VECTOR ENTRIES IN THE TABLE AND IS IN GMT FLOATING

POINT SECONDS.

t_{stop} IS THE TIME ASSOCIATED WITH THE LAST POSITION, VELOCITY VECTOR ENTRIES IN THE TABLE AND IS IN GMT FLOATING POINT SECONDS

$x, y, z, \dot{x}, \dot{y}, \dot{z}$ ARE THE COMPONENTS OF THE POSITION AND VELOCITY VECTORS IN THE TABLE IN FLOATING POINT GEMINI UNITS

L (h_n) A ONE-LOCATION TABLE CONTAINING THE TIME INTERVAL BETWEEN POSITION, VELOCITY VECTOR ENTRIES IN THE TABLE IN FLOATING POINT SECONDS.

L (t_d) A ONE-LOCATION TABLE CONTAINING THE TIME AT WHICH THE INTERPOLATED VECTORS ARE DESIRED IN GMT FLOATING POINT SECONDS.

OUTPUT

A) TABLE

L(OUTPUT) THE ADDRESS OF A SEVEN-LOCATION TABLE ARRANGED AS FOLLOWS -

L(OUTPUT)	t_d	DESIRED TIME IN GMT FLOATING POINT SECONDS
+1	x	POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+2	y	
+3	z	

+4	$\dot{x}$	VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+5	$\dot{y}$	
+6	$\dot{z}$	

METHOD

A) LAGRANGE'S INTERPOLATION FORMULA FOR EQUAL INTERVALS

THE POSITION AND THE VELOCITY VECTOR ENTRIES OF THE NUMERICAL INTEGRATION VECTOR TABLE ARE MADE AT EQUAL INTERVALS OF TIME. THEREFORE, THE INTERPOLATION FORMULA FOR EQUAL INTERVALS OF THE INDEPENDENT VARIABLE MAY BE APPLIED. THE FORMULA FITS AN nth DEGREE POLYNOMIAL THROUGH (n+1) CONSECUTIVE POINTS. IT MAY BE WRITTEN AS FOLLOWS - (NUMERICAL ANALYSIS, KUNZ, MCGRAW-HILL BOOK COMPANY, INC., NEW YORK, 1957, PP. 91-94.)

$$f(x) = (((-1)^{n}(u-1)(u-2)\ldots(u-n))/n!)(f(x_0)) +$$
$$(((-1)^{n-1}(u)(u-2)\ldots(u-n))/1!(n-1)!)(f(x_1)) + \ldots +$$
$$(((u)(u-1)(u-2)\ldots(u-n+1))/n!)(f(x_n))$$

WHERE
- $f(x)$ = FINAL INTERPOLATED VALUE
- $f(x_i)$ = VALUES OF THE (N+1) TABULATED POINTS BEING USED
- u = DISTANCE OF DESIRED POINT BETWEEN TABULATED VALUES OF THE DEPENDENT VARIABLE
- n = DEGREE OF POLYNOMIAL BEING FITTED TO THE DATA

B) APPLICATION OF THE INTERPOLATION FORMULA IN C6PTLI

IN C6PTLI, THE ATTEMPT IS MADE TO PASS A DEGREE POLYNOMIAL THROUGH SIX CONSECUTIVE POINTS IN THE TABLE. THE SIX POINTS MUST BE EQUALLY DISTRIBUTED ABOUT THE DESIRED TIME, t_d (I.E., THREE POINTS MUST FALL BEFORE t_d AND THREE POINTS AFTER t_d). THUS, THE FOLLOWING REVISION IN DEFINITION APPLIES TO THE VARIABLES -

$f(x)$ BECOMES, IN TURN $x, y, z, \dot{x}, \dot{y}, \dot{z}$

$f(x_i)$ BECOMES, IN TURN $x_i, y_i, z_i, \dot{x}_i, \dot{y}_i, \dot{z}_i$

WHERE i VARIES FROM -2 TO +3

$n = 5$

$u = T$, THE FRACTIONAL DISTANCE OF t_d BETWEEN TABULATED VALUES IN THE TABLE

THE FOLLOWING ADDITIONAL DEFINITIONS ALSO HOLD -

h_n IS THE TIME INTERVAL BETWEEN ENTRIES IN THE TABLE

L_i IS THE LAGRANGIAN COEFFICIENT OF $f(x_i)$

BECAUSE THE INTERVAL BETWEEN ENTRIES IN THE TABLE IS CONSTANT AND THE NUMBER OF POINTS TO BE USED IS KNOWN, PARTIAL EVALUATION OF THE L_i MAY BE MADE OUTISDE THE PROGRAM. THE RESULTING L_i ARE GIVEN BELOW -

$$L_{-2} = (-1/120)((T+1)(T)(T-1)(T-2)(T-3))$$
$$L_{-1} = (1/24)((T+2)(T)(T-1)(T-2)(T-3))$$
$$L_{0} = (-1/12)((T+1)(T+2)(T-1)(T-2)(T-3))$$
$$L_{+1} = (1/12)((T+1)(T+2)(T)(T-2)(T-3))$$
$$L_{+2} = (-1/24)((T+1)(T+2)(T)(T-1)(T-3))$$
$$L_{+3} = (1/120)((T+1)(T+2)(T)(T-1)(T-2))$$

C) DETAILS OF METHOD OF C6PTLI

AS SOON AS C6PTLI IS GIVEN CONTROL, IT SAVES INDEX REGISTERS AND PICKS UP THE ADDRESSES OF THE INPUT AND OUTPUT TABLES TO BE USED, THE INTERVAL BETWEEN ENTRIES IN THE TABLE, AND THE DESIRED TIME.

IT THEN DETERMINES WHETHER THERE ARE AT LEAST THREE VECTORS BEFORE AND AFTER t_d. IF ONE OF THESE CONDITIONS IS NOT MET, C6PTLI RETURNS TO THE CONTROL PROGRAM VIA THE ERROR RETURN. IF BOTH CONDITIONS ARE SATISFIED, THE DESIRED TIME, t_d IS PLACED IN THE OUTPUT TABLE.

C6PTLI THEN COMPUTES N, THE NUMBER OF VECTORS PRECEDING THE FIRST VECTOR TO BE USED IN THE INTERPOLATION PROCESS, AND T, THE FRACTIONAL DISTANCE OF t_d BETWEEN TABULATED VALUES IN THE TABLE, AS FOLLOWS -

N = INTEGRAL PORTION OF $(t_d - 2h_n - t_{start})/h_n$ IN FIXED POINT

T = FRACTIONAL PORTION $(t_d - 2h_n - t_{start})/h_n$ IN FLOATING POINT

C6PTLI ALSO COMPUTES THE PRESUMTIVE ADDRESS (I.E., THE ADDRESS OF THE X-COMPONENT OF THE VECTOR FOLLOWING THE LAST ONE TO BE USED IN THE INTERPOLATION PROCESS) -

P.A. = L(r,v TABLE) + 38 + 6N

IF T IS ZERO, THE TIME AT WHICH THE INTERPOLATED VECTORS ARE DESIRED COINCIDES WITH THE TIME OF ONE OF THE ENTRIES IN THE TABLE. THUS, THE PROCESS OF INTERPOLATION IS REDUCED TO ONE OF EXTRACTING THE CORRECT VECTORS AND PLACING THEM IN THE OUT-

PUT TABLE. ONCE THIS HAS BEEN DONE, C6PTLI RESTORES THE INDEX REGISTERS AND RETURNS TO THE MAIN PROGRAM VIA THE NORMAL RETURN.

IF T IS NON-ZERO, INTERPOLATION FOR THE DESIRED VECTORS TAKES PLACE. FIRST, THE COMPLETE FUNCTION, G, IS COMPUTED -

$$G = (T+2)(T+1)(T)(T-1)(T-2)(T-3)$$

THEN, L_i THE COEFFICIENT OF $f(x_i)$ IS COMPLETELY EVALUATED -

$L_i = G/(T-i)(K_i)$ WHERE K_i IS THE INVERSE OF THE LEADING COEFFICIENT OF L_i .

FOR EACH COMPONENT OF THE POSITION AND THE VELOCITY VECTORS, L_i IS MULTIPLIED BY THE CORRECT TABULAR VALUE AND ADDED TO THE PREVIOUS SUM FOR THAT COMPONENT. THE FINAL SUM FOR EACH COMPONENT IS STORED IN THE SECOND THROUGH THE SEVENTH LOCATIONS OF THE OUTPUT TABLE. C6PTLI THEN RESTORES INDEX REGISTERS AND RETURNS TO THE MAIN PROGRAM VIA THE NORMAL EXIT.

USAGE

A) CALLING SEQUENCE -

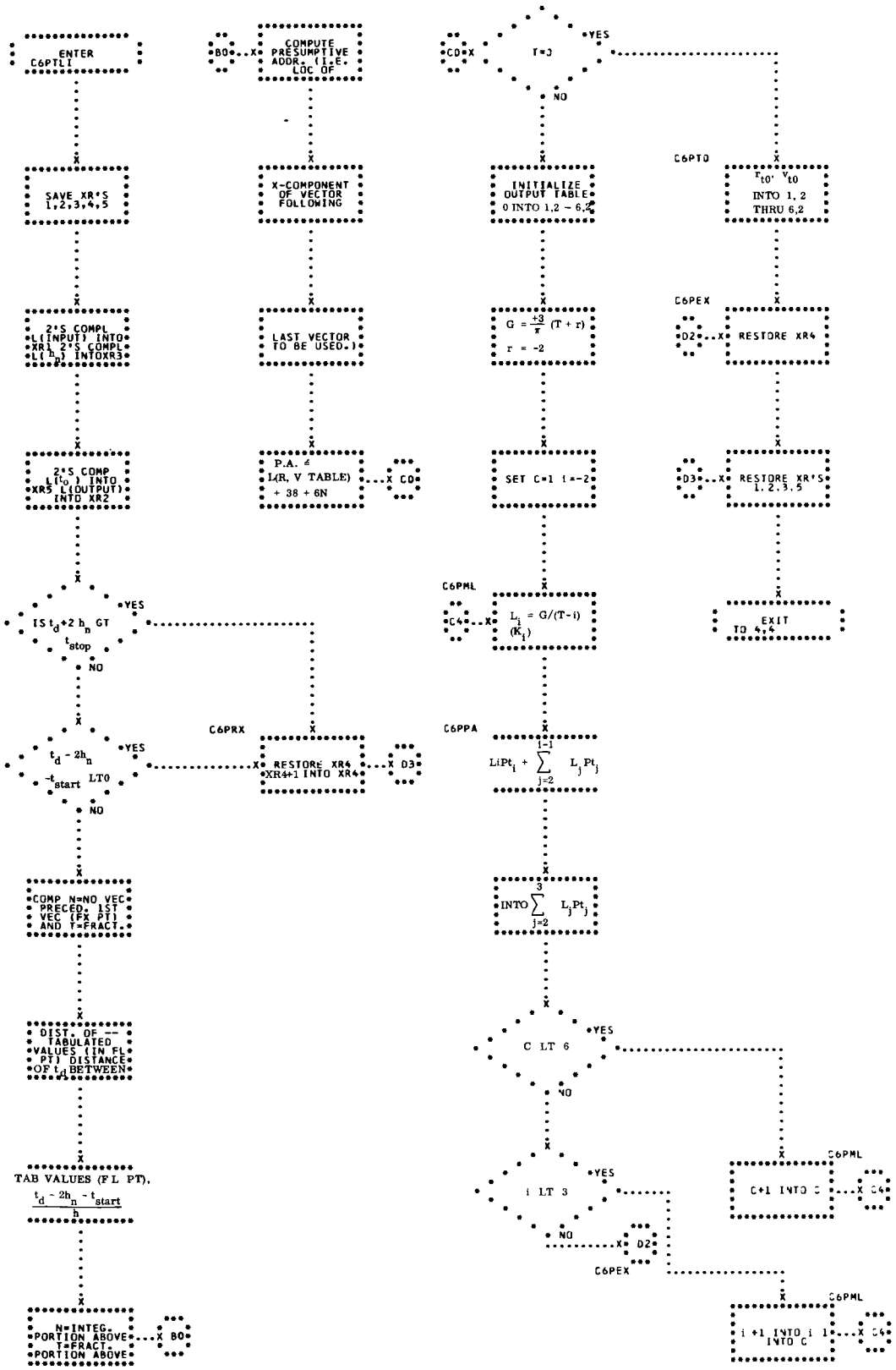
```
TSX     C6PTLI,4
PZE     L(r,v TABLE),, L(h_n)
PZE     L(t_d) ,, L(OUTPUT)
ERROR RETURN
NORMAL RETURN
```

B) ERROR CONDITIONS -

AN ERROR RETURN FROM C6PTLI IMPLIES THAT THERE WERE NOT THREE VECTORS BEFORE AND AFTER THE DESIRED TIME, t_d

C) STORAGE REQUIRED -
112 LOCATIONS

D) TIMING (7094) -
.312 MILLISECONDS, IF VECTOR AT DESIRED TIME IS IN TABLE
1.310 MILLISECONDS, IF VECTOR AT DESIRED TIME MUST BE INTERPOLATED.


ENTER C6PTLI
SAVE XR'S 1,2,3,4,5
2'S COMPL (INPUT) INTO XR1 2'S COMPL (h_n) INTOXR3
2'S COMP L(t_0) INTO XR5 L(OUTPUT) INTO XR2
IS t_d+2 h_n GT t_stop
YES
NO
t_d - 2h_n - t_start LT0
YES
NO
COMP N=NO VEC PRECED. LST VEC (FX PT) AND T=FRACT.
DIST. OF -- TABULATED VALUES (IN FL PT) DISTANCE OF t_d BETWEEN
TAB VALUES (F L PT), (t_d - 2h_n - t_start)/h
N=INTEG. PORTION ABOVE T=FRACT. PORTION ABOVE
B0
COMPUTE PRESUMPTIVE ADDR. (i.E. LOC OF
X-COMPONENT OF VECTOR FOLLOWING
LAST VECTOR TO BE USED.)
P.A. = L(R, V TABLE) + 38 + 6N
C0
C6PRX
RESTORE XR4 XR4+1 INTO XR4
D3
C0
T=0
YES
NO
INITIALIZE OUTPUT TABLE 0 INTO 1,2 - 6,2
G = -3/π (T + r)
r = -2
SET C=1 i=-2
C6PML
L_i = G/(T-i) (K_1)
C6PPA
L_iPt_i + Σ_{j=2}^{i-1} L_jPt_j
INTO Σ_{j=2}^{3} L_jPt_j
C LT 6
YES
NO
i LT 3
YES
NO
D2
C6PEX
C6PML
C+1 INTO C
C6PML
i +1 INTO i INTO C
C6PTD
t_t0, v_t0 INTO 1, 2 THRU 6,2
C6PEX
RESTORE XR4
D2
D3
RESTORE XR'S 1, 2, 3, 5
EXIT TO 4,4
C4
C4
C4

46. CRADES - LAUNCH PROGRAM TO SELECT RECOVERY AREA

GIVEN THE WESTERNMOST AND THE EASTERNMOST IMPACT LONGTIUDES THAT CAN BE ACHIEVED, THIS PROGRAM SELECTS THE RECOVERY AREA WHICH SHOULD BE USED. IN THE EVENT THAT NO RECOVERY AREA LIES BETWEEN THE ABOVE LONGITUDES, CRADES DESIGNATES THE AREA CLOSEST TO ONE OF THE ABOVE AND SPECIFIES WHICH LONGITUDE OF IMPACT SHOULD BE USED. (SEE FLOWCHART.)

INPUT

A) TABLES

L(INPUT) THE ADDRESS OF A FOUR-LOCATION TABLE CONTAINING LATITUDE OF MINIMUM IMPACT, LONGITUDE OF MINIMUM IMPACT, LATITUDE OF MAXIMUM IMPACT, AND LONGITUDE OF MAXIMUM IMPACT IN THAT ORDER IN FLOATING POINT RADIANS.

TCRCAL A NINE-LOATION TABLE CONTAINING THE LONGITUDE OF THE RECOVERY AREAS AND ARRANGED AS FOLLOWS -

TCRCAL	DEC 0.0
+1	LONGITUDE OF RECOVERY AREA A
+2	LONGITUDE OF RECOVERY AREA B
+3	LONGITUDE OF RECOVERY AREA C
+4	LONGITUDE OF RECOVERY AREA D
+5	LONGITUDE OF RECOVERY AREA E
+6	LONGITUDE OF RECOVERY AREA F
+7	LONGITUDE OF RECOVERY AREA 1A
+8	DEC 50.0

TCRCAD A SEVEN-LOCATION TABLE CONTAINING

THE OCTAL CODES OF THE RECOVERY AREAS AND ARRANGED AS FOLLOWS -

TCRCAD	CODE OF RECOVERY AREA A
+1	CODE OF RECOVERY AREA B
+2	CODE OF RECOVERY AREA C
+3	CODE OF RECOVERY AREA D
+4	CODE OF RECOVERY AREA E
+5	CODE OF RECOVERY AREA F
+6	CODE OF RECOVERY AREA 1A

B) CONSTANTS -

KM0000	DEC -0	FIXED POINT MINUS ZERO
KM0001	DEC -1	FIXED POINT MINUS ONE
KO3HPI		FLOATING POINT $3/2\pi$ RADIANS
KKK2PI		FLOATING POINT 2π RADIANS

OUTPUT

A) TABLE -

L(OUTPUT) A THREE-LOCATION TABLE ARRANGED AS FOLLOWS -

L(OUTPUT)	CODE OF SELECTED RECOVERY AREA IN OCTAL
+1	LONGITUDE OF SELECTED RECOVERY AREA IN FLOATING POINT RADIANS
+2	RECOVERY AREA INDICATOR IN FIXED POINT WHICH MAY HAVE ONE OF THE FOLLOWING VALUES -

+0 SELECTED AREA LIES BETWEEN LONGITUDES OF MAXIMUM AND MINIMUM DELAY IMPACT

-0 SELECTED AREA DOES NOT LIE BETWEEN LONGITUDES OF MINI-

MUM AND MAXIMUM DELAY IMPACT, BUT IS CLOSEST TO LONGITUDE OF MINIMUM DELAY IMPACT

-1 SELECTED AREA DOES NOT LIE BETWEEN LONGITUDES OF MINIMUM AND MAXIMUM DELAY IMPACT, BUT IS CLOSEST TO LONGITUDE OF MAXIMUM DELAY IMPACT

METHOD

CRADES DETERMINES WHICH RECOVERY AREAS LIE BETWEEN THE LONGITUDE ASSOCIATED WITH RETROFIRE AT MINIMUM DELAY AND THE LONGITUDE ASSOCIATED WITH RETROFIRE AT MAXIMUM DELAY. THESE TWO LONGITUDES FORM A BRACKET FROM WHICH THE REQUIRED INFORMATION IS DETERMINED. NORMALLY, THE SELECTED RECOVERY AREA WILL BE THAT AREA INCLUDED IN THE BRACKET AND COSEST TO THE MAXIMUM DELAY RETROFIRE IMPACT POINTS. ITS LONGITUDE AND CODE ARE READILY AVAILABLE FROM THE TCRCAL AND TCRCAD TABLES, RESPECTIVELY, AND THE RECOVERY AREA INDICATOR WILL BE SET TO +0.

IF THE EASTERN-MOST ATLANTIC OCEAN RECOVERY AREA (I.E.,AREA D) IS INCLUDED IN THE BRACKET, IT IS SELCTED EVEN THOUGH IT MAY NOT BE THE INCLUDED AREA CLOSEST TO MAXIMUM DELAY IMPACT. IN THIS WAY, LANDING IN CONTINENTAL AFRICA (AREAS E AND F) IS AVOIDED WHENEVER POSSIBLE.

IF THERE ARE NO AREAS INCLUDED IN THE BRACKET, THE SELCTION OF THE RECOVERY AREA IS BASED ON ITS DISTANCE FROM THE CLOSEST END OF THE BRACKET. THAT IS, EITHER THE RECOVERY AREA IMMEDIATELY WEST OF THE MINIMUM DELAY IMPACT POINT OR THE RECOVERY AREA IMMEDIATELY EAST OF THE

MAXIMUM DELAY IMPACT POINT WILL BE SELECTED DEPENDING UPON WHICH OF THESE AREAS IS CLOSEST TO ITS RESPECTIVE END OF THE BRACKET. THE RECOVERY AREA INDICATOR WILL BE SET ACCORDINGLY.

USAGE

A) CALLING SEQUENCE -

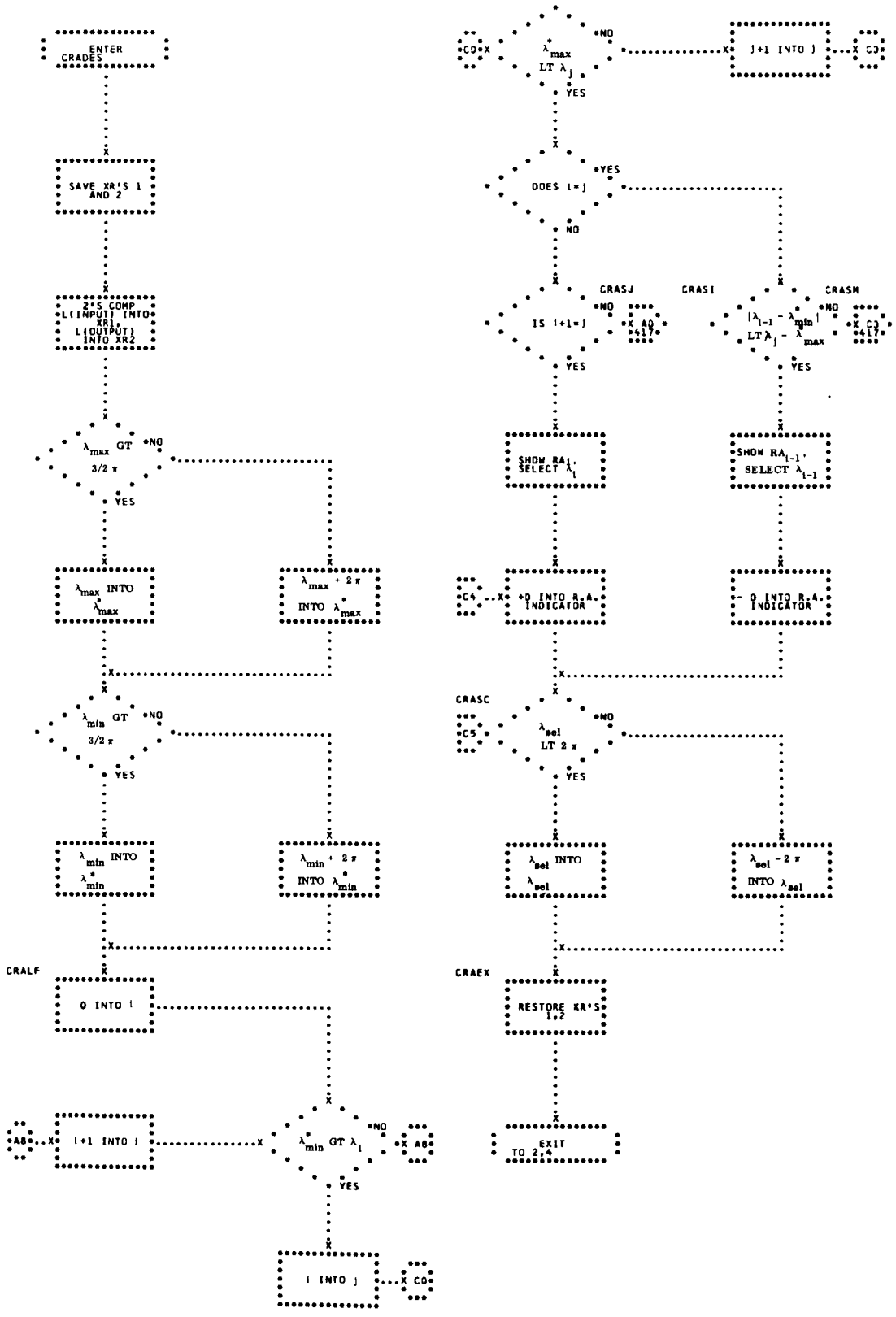
```
TSX     CRADES,4
PZE     L(INPUT),,L(OUTPUT)
RETURN
```

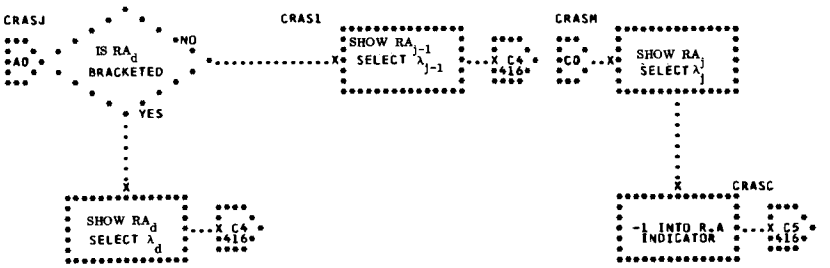
B) STORAGE REQUIRED -

85 LOCATIONS (EXCLUDING TABLES)

C) TIMING (7094) -

.254 MILLISECONDS, MAXIMUM


ENTER GRADES
SAVE XR'S 1 AND 2
2'S COMP L(INPUT) INTO XR1, L(OUTPUT) INTO XR2
λ_{max} GT 3/2 π
YES
NO
λ_{max} INTO λ_{max}
λ_{max} + 2 π INTO λ_{max}
λ_{min} GT 3/2 π
λ_{min} INTO λ_{min}
λ_{min} + 2 π INTO λ_{min}
CRALF
0 INTO I
I+1 INTO I
λ_{min} GT λ_I
I INTO J
CO
A8
λ_{max} LT λ_J
J+1 INTO J
DOES I=J
CRASJ
IS I+1=J
A0
+17
CRASI
CRASM
$|\lambda_{I-1} - \lambda_{min}|$ LT $\lambda_J - \lambda_{max}$
SHOW RA$_I$, SELECT λ_I
SHOW RA$_{I-1}$, SELECT λ_{I-1}
C4
+0 INTO R.A. INDICATOR
-0 INTO R.A. INDICATOR
CRASC
C5
λ_{sel} LT 2 π
λ_{sel} INTO λ_{sel}
λ_{sel} - 2 π INTO λ_{sel}
CRAEX
RESTORE XR'S 1,2
EXIT TO 2,4


CRASJ
AO
IS RA$_d$ BRACKETED
NO
YES
CRASI
SHOW RA$_{j-1}$ SELECT λ_{j-1}
X C4 416
CRASM
CO
SHOW RA$_j$ SELECT λ_j
SHOW RA$_d$ SELECT λ_d
X C4 416
CRASC
-1 INTO R.A INDICATOR
X C5 416

47. C9TDRF - RETROFIRE TIME CORRECTION

THIS PROGRAM COMPUTES THE CORRECTION TO THE ESTIMATED TIME FOR FIRING RETROROCKETS TO LAND IN A DESIGNATED RECOVERY AREA. (SEE FLOWCHART.)

INPUT

A) CALLING SEQUENCE REQUIREMENTS -

L(INPUT) - ADDRESS OF A SIX-LOCATION TABLE CONTAINING THE COMPONENTS OF $\overline{r}$ AND $\overline{v}$ IN FLOATING POINT GEMINI UNITS (USED ON FIRST ITERATION ONLY)

L (λ_c) - ADDRESS OF A CELL CONTAINING THE COMPUTED LONGITUDE OF IMPACT IN FLOATING POINT RADIANS ($0 \leq \lambda_c \leq 2\pi$)

L (λ_{SEL}) - ADDRESS OF A CELL CONTAINING THE LONGITUDE OF THE SELECTED RECOVERY AREA IN FLOATING POINT RADIANS

B) CONSTANTS -

KOO.PI	FLOATING POINT VALUE OF π (Pi)
KKK2PI	FLOATING POINT VALUE OF 2π (2Pi)
K.OMEG	FLOATING POINT ROTATION OF THE EARTH IN RADIANS/GEMINI TIME UNIT
KOO.UT	GEMINI TIME UNIT IN FLOATING POINT SECONDS
K.5SGU	ONE-HALF SECOND IN FLOATING POINT GEMINI TIME UNITS

C) TABLES (NOT USED ON FIRST ITERATION) -

TCDTCP	TIME CORRECTION OF THE LAST PREVIOUS ITERATION IN FLOATING POINT GEMINI TIME UNITS

TCLACP COMPUTED LONGITUDE OF IMPACT OF THE LAST PREVIOUS ITERATION IN FLOATING POINT RADIANS

OUTPUT

A ONE-LOCATION TABLE CONTAINING THE TIME CORRECTION IN FLOATING POINT SECONDS IS OUTPUT FROM THE PROGRAM. (THIS CORRESPONDS TO THE TIME WHICH IS STORED IN TCDTCP FOR USE IN THE NEXT ITERATION.)

METHOD

C9TDRF FIRST COMPUTES THE DIFFERENCE ($\Delta\lambda$) BETWEEN THE LONGITUDE OF THE SELECTED RECOVERY AREA λ_{SEL} AND THE COMPUTED LONGITUDE OF IMPACT (λ_c). IF THIS $|\Delta\lambda| > \pi$ IT IS NECESSARY THAT IT BE SCALED BY 2π IN ORDER TO KEEP THE FINAL TIME CORRECTION UNDER THE TIME TO COMPLETE ONE-HALF ORBIT. IN PRACTICE, THIS MOST OFTEN OCCURS WHEN λ_{SEL} AND λ_c BRACKET THE DISCONTINUITY IN LONGITUDE AT 360 DEGREES.

IF THE ENTRY IN C9TDRF IS A FIRST ITERATION, AN APPROXIMATION OF THE EASTWARD VELOCITY OF THE SPACECRAFT IS COMPUTED USING

$$\omega_Z = \frac{X\dot{Y} - Y\dot{X}}{X^2 + Y^2}$$

(A DERIVATION OF W IS GIVEN LATER IN THIS SECTION.) IT IS NECESSARY TO COMPUTE THE EASTWARD COMPONENT BECAUSE IN THE INTERPOLATION PROGRAM ONLY LONGITUDE IS CONSIDERED. A CHANGE IN LONGITUDE IS AFFECTED BY THE EASTWARD COMPONENT OF THE VELOCITY. SUBRTACTING THE VELOCITY DUE OF THE ROTATION OF THE EARTH ω_e AND USING THE FORMULA ''DISTANCE EQUALS THE RATE TIMES THE TIME,'' THE CORRECTION Δt EQUALS $\Delta\lambda/(\omega_Z - \omega_e)$.

IF THIS ENTRY IS NOT A FIRST ITERATION, THE DIFFERENCE BETWEEN λ_c AND THE PREVIOUSLY COMPUTED $\lambda_c(\lambda_{cp})$ IS COMPUTED. THIS DIFFERENCE IS $\Delta\lambda p$ THEN IF

(1) $\Delta t_p \geq 0$, and if

a) $\Delta\lambda_p \geq 0$, $\Delta\lambda_p$ is not scaled

b) $\Delta\lambda_p < 0$, $\Delta\lambda_p$ is scaled by 2π

(2) $\Delta t_p < 0$, and if

a) $\Delta\lambda_p \geq 0$, $\Delta\lambda_p$ is scaled by -2π

b) $\Delta\lambda_p < 0$, $\Delta\lambda_p$ is not scaled

THIS SCALING IS NECESSARY SO THAT THE INTERPOLATION FORMULA FOR THE COMPUTING A TIME CORRECTION CAN PRODUCE A CONVERGENT Δt. IF λ_c AND λ_{cp} BRACKET THE DISCONTINUITY IN LONGITUDE AT 360°, Δt WILL NOT CONVERGE WHEN Δtp AND $\Delta\lambda p$ HAVE OPPOSITE SIGNS.

FINALLY, THE NEW TIME CORRECTION CAN BE COMPUTED USING THE LINEAR INTERPOLATION FORMULA -

$$\Delta t = \left(\frac{\Delta\lambda}{\Delta\lambda_p} \right) \Delta t_p$$

WHETHER A FIRST ITERATION OR NOT, IF Δt IS LESS THAN .5 SECOND, IT IS SET TO ZERO AND AN EXIT IS MADE FROM THE PROGRAM. IF Δt IS EQUAL TO OR GREATER THAN .5 SECOND, Δt BECOMES Δt_p (STORED IN TCDTCP) AND λ_c BECOMES λ_{cp} (STORED IN RCLACP). THE TIME CORRECTION IN SECONDS IS STORED AND AN EXIT IS MADE FROM THE PROGRAM.

USAGE

A) CALLING SEQUENCE -

```
TSX     C9TDRF,4
G       L(INPUT),,L(OUTPUT)
PZE     L(λc),,L(λSEL)
NORMAL  RETURN
```

G = PZE INDICATES AN INITIAL REFINEMENT.

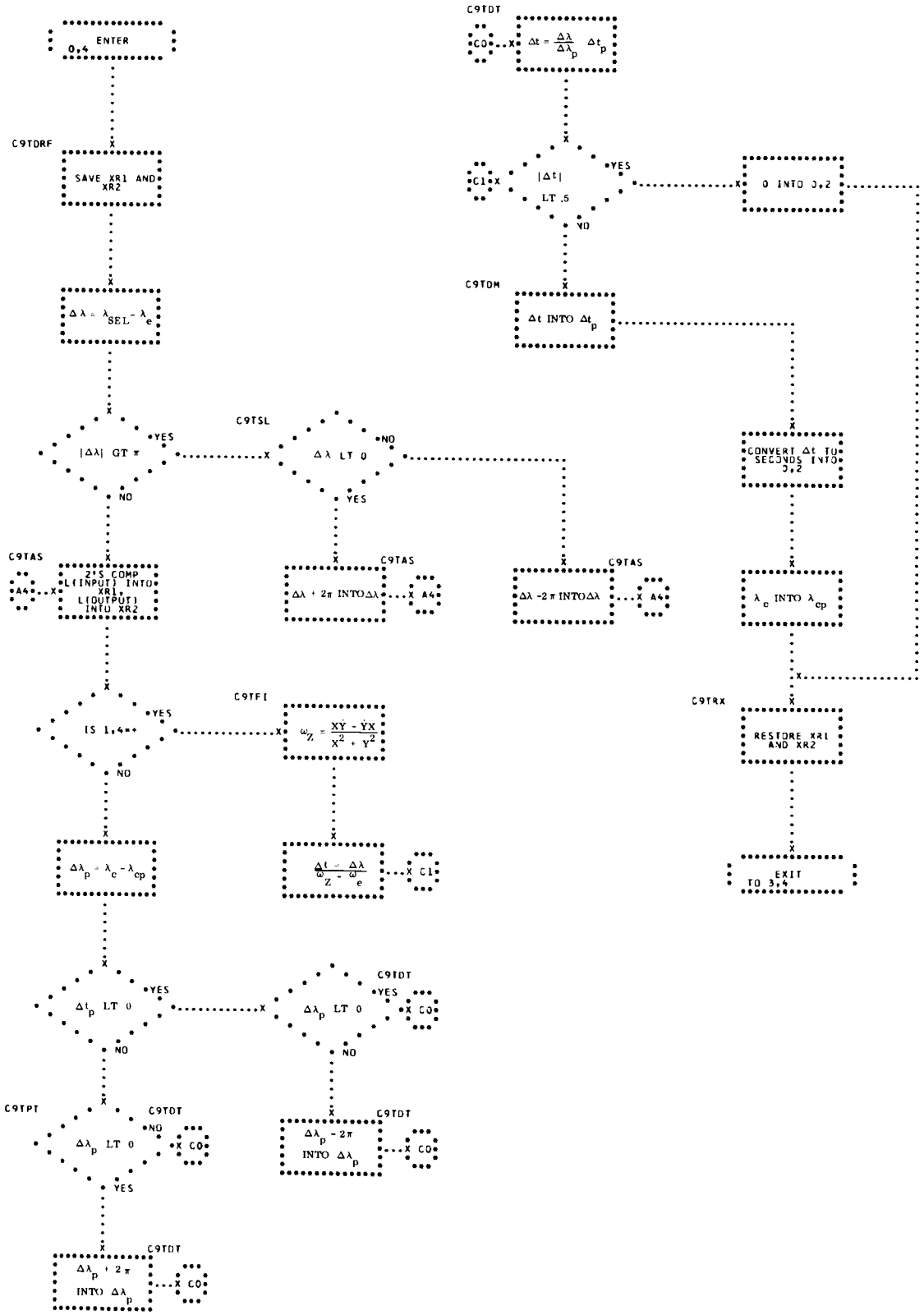
G = MZE INDICATES A SUCCESSIVE REFIENMENT.

B) STORAGE REQUIRED - 70 DECIMAL LOCATIONS EXLUDING THAT REQUIRED BY EXTERNAL CONSTANTS AND TABLES.

C) TIME REQUIRED (7094 MAXIMUM) -

0.288 MILLISECOND FOR AN INITIAL REFINEMENT

0.212 MILLISECOND FOR ANY OTHER REFINEMENT


ENTER 0,4
C9TDRF
SAVE XR1 AND XR2
$\Delta\lambda = \lambda_{SEL} - \lambda_e$
$|\Delta\lambda|$ GT π
YES
NO
C9TSL
$\Delta\lambda$ LT 0
NO
YES
C9TAS
2'S COMP L(INPUT) INTO XR1, L(OUTPUT) INTO XR2
$\Delta\lambda + 2\pi$ INTO $\Delta\lambda$
C9TAS
A4
$\Delta\lambda - 2\pi$ INTO $\Delta\lambda$
C9TAS
A4
C9TFI
IS 1,4=+
YES
NO
$\omega_Z = \frac{X\dot{Y} - \dot{Y}X}{X^2 + Y^2}$
$\Delta\lambda_p = \lambda_c - \lambda_{cp}$
$\frac{\Delta t = \Delta\lambda}{\omega_Z - \omega_e}$
C1
Δt_p LT 0
YES
NO
C9TDT
$\Delta\lambda_p$ LT 0
YES
NO
CO
C9TPT
C9TDT
$\Delta\lambda_p$ LT 0
NO
YES
C9TDT
$\Delta\lambda_p - 2\pi$ INTO $\Delta\lambda_p$
CO
C9TDT
$\Delta\lambda_p + 2\pi$ INTO $\Delta\lambda_p$
CO
C9TDT
CO
$\Delta t = \frac{\Delta\lambda}{\Delta\lambda_p} \Delta t_p$
C1
$|\Delta t|$ LT .5
YES
NO
0 INTO 3,2
C9TDM
Δt INTO Δt_p
CONVERT Δt TO SECONDS INTO 3,2
λ_c INTO λ_{cp}
C9TRX
RESTORE XR1 AND XR2
EXIT TO 3,4

48. CGFTCI - COMPUTATION OF IMPACT, GIVEN THE TIME TO FIRE RETROROCKETS

GIVEN THE TIME, t_4 AT WHICH TO FIRE THE RETROROCKETS, THIS PROGRAM COMPUTES THE LATITUDE AND LONGITUDE OF IMPACT. IT IS A SUBROUTINE OF THE AVERAGE VECTORS PROGRAM CFCFAV. (SEE FLOWCHART.)

INPUT

A) A ONE-LOCATION TABLE CONTAINING THE TIME AT WHICH RETROFIRE WILL OCCURE IN FLOATING POINT SECONDS AND A ONE-LOCATION TABLE CONTAINING THE BANK ANGLE IN FLOATING POINT RADIANS ARE INPUT TO THE PROGRAM.

B) SUBROUTINES C6PTLI, C6RETC, AND CORKNI ARE USED.

C) CONSTANTS -

K00000	DEC	0	ZERO
K00001	DEC	1	FIXED POINT ONE
K00050	DEC	50	FIXED POINT FIFTY

D) TABLES -

TCRKOT AN ORBIT TABLE OF $\overline{r}$ AND $\overline{v}$ VECTORS WHICH IS USED BY C6PTLI TO EXTRACT THE $\overline{r}$ AND $\overline{v}$ VECTOR AT RETROFIRE TIME. THIS TABLE MUST BE IN THE FORMAT $t_i, t_n, \overline{r}_1, \overline{v}_1, \ldots, \overline{r}_i, \overline{v}_i, \ldots, \overline{r}_n, \overline{v}_n$ WHERE t IS IN FLOATING POINT SECONDS AND THE COMPONENTS OF $\overline{r}$ AND $\overline{v}$ ARE IN FLOATING POINT GEMINI UNITS.

TRK01H A ONE-LOCATION TABLE CONTAINING THE TIME INTERVAL BETWEEN ENTRIES IN THE TABLE TCRKOT IN FLOATING POINT SEC-

ONDS.

TC6PLI	A SEVEN-LOCATION TABLE TO BE USED AS THE OUTPUT BLOCK BY C6PTLI AND WHICH CONTAINS THE $t,\overline{r},\overline{v}$ AT RETROFIRE TIME.
TCCGV5	A SEVEN-LOCATION TABLE TO BE USED AS THE OUTPUT BLOCK BY C6RETC AND WHICH CONTAINS THE $t,\overline{r},\overline{v}$ AT END OF RETROFIRE.
TCRFTQ	A ONE-LOCATION TABLE WHICH WHEN EQUAL TO ZERO INDICATES THAT NO TORQUE BE APPLIED TO THE INERTIAL PLATFORM DURING RETROFIRE AND WHICH WHEN NON-ZERO INDICATES THAT TORQUE IS APPLIED DURING RETROFIRE.
TCWF.R	A ONE-LOCATION TABLE CONTAINING THE REENTRY WEIGHT OF THE SPACECRAFT IN FLOATING POINT POUNDS.
TCAHC2	A ONE-LOCATION TABLE CONTAINING THE LAUNCH/ABORT NOMINAL IMPACT HEIGHT IN FLOATING POINT GEMINI UNITS.

OUTPUT

A TWO-LOCATION TABLE CONTAINING THE LATITUDE AND LONGITUDE OF IMPACT IN FLOATING POINT RADIANS IS OUTPUT FROM THE PROGRAM.

METHOD

USING THE GIVEN TIME OF RETROFIRE, t_4 CGFTCI ENTERS C6PTLI TO OBTAIN THE $\overline{r}$ AND $\overline{v}$ VECTORS ASSOCIATED WITH t_4 C6PTLI USED THE LAGRANGIAN INTERPOLATION TO EXTRACT THESE VECTORS FROM THE LAUNCH NUMERICAL INTEGRATION ORBIT TABLE, TCRKOT.

THESE INTERPOLATED VECTORS AT t_4 THEN SERVE AS INPUT

TO C6RETC, WHICH ASSUMES THAT ALL FOUR RETROROCKETS FIRE IN THEIR NORMAL SEQUENCE. THE OUTPUT FROM C6RETC IS THE $t, \bar{r}, \bar{v}$ AT THE END OF RETROFIRE (CALLED BURNOUT VECTORS).

THESE BURNOUT VECTORS ARE THEN INTEGRATED TO IMPACT BY THE LAUNCH NUMERICAL INTEGRATION PROGRAM, CORKNI (TYPE 1). THE LIFT CONFIGURATION IS DETERMINED BY THE BANK ANGLE SPECIFIED IN THE CALLING SEQUENCE OF CGFTCI. THE OUTPUT FROM CORKNI IS THE TIME, LATITUDE, AND LONGITUDE OF IMPACT.

USAGE

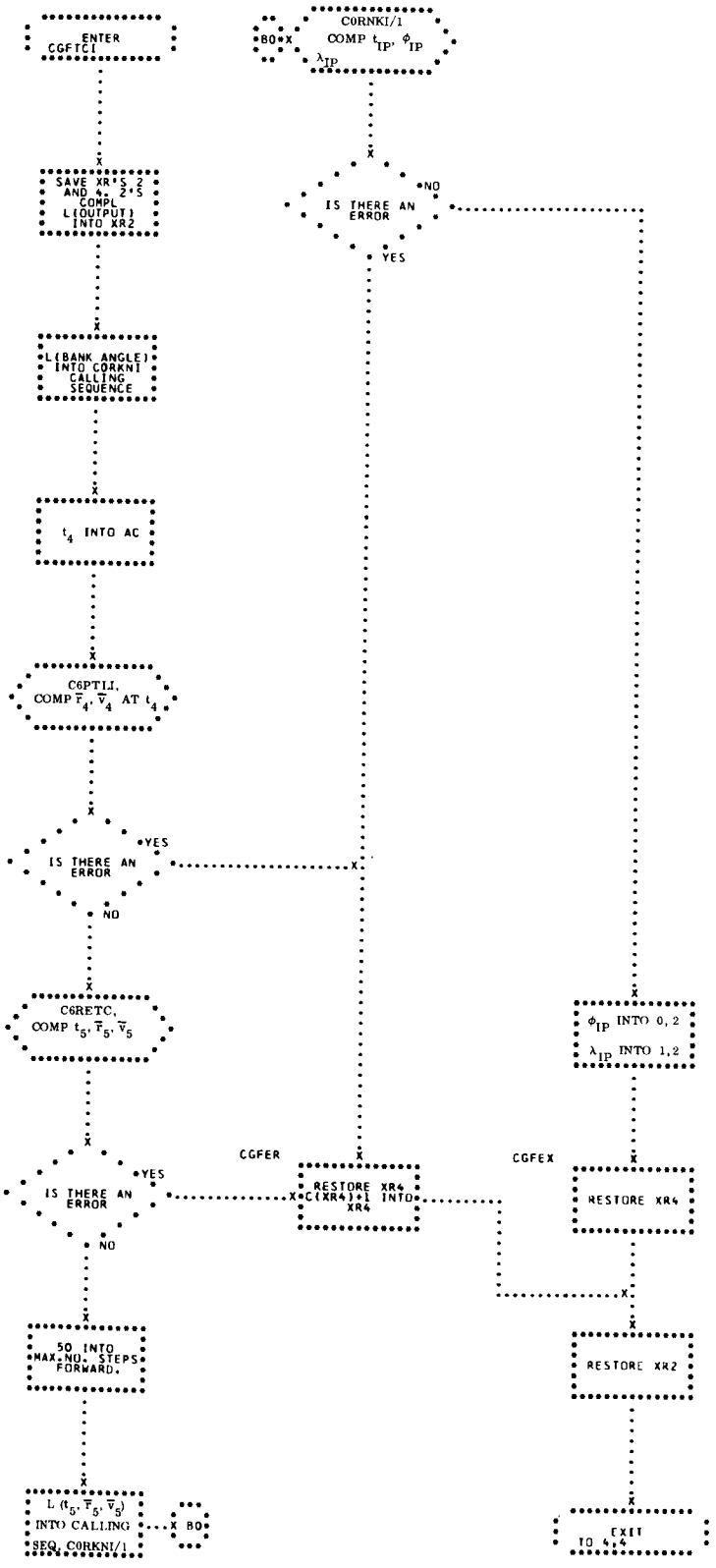
A) CALLING SEQUENCE -

```
TSX     CGFTCI,4
PZE     L(t4),,L(OUTPUT)
PZE     L(BANK ANGLE),,**
ERROR   RETURN
NORMAL  RETURN
```

B) STORAGE REQUIRED - 35 LOCATIONS EXCLUDING THAT NEEDED BY SUBROUTINES AND EXTERNAL CONSTANTS.

C) ERROR CONDITIONS - ERROR RETURN FROM ANY OF THE SUBROUTINES WILL CAUSE CGFTCI TO RETURN CONTROL TO THE MAIN PROGRAM VIA THE ERROR EXIT.

D) TIME REQUIRED (7094) - 0.045 MILLISECONDS (MAXIMUM) EXCLUDING ALL SUBROUTINE TIME. CONSIDERING SUBROUTINE TIME, CGFTCI COULD POSSIBLY TAKE 750 MILLISECONDS EXECUTION TIME IF CORKNI EXHAUSTED THE SAFETY COUNT OF 50 STEPS. HOWEVER, THE AVERAGE EXECUTION TIME IS 350 MILLISECONDS. THIS ACTUAL MACHINE TIME WAS COMPUTED USING A VECTOR AFTER NOMINAL SSCO AND A NOMINAL RETROFIRING SEQUENCE, AND THEN INTEGRATING THE BURNOUT VECTOR TO 60,000 FEET WITH A 60° BANK ANGLE.


ENTER CGFTC1
SAVE XR'S 2 AND 4, 2'S COMPL (OUTPUT) INTO XR2
L(BANK ANGLE) INTO CORKNI CALLING SEQUENCE
t_4 INTO AC
C6PTLI, COMP $\bar{r}_4$, $\bar{v}_4$ AT t_4
IS THERE AN ERROR
YES
NO
C6RETC, COMP t_5, $\bar{r}_5$, $\bar{v}_5$
IS THERE AN ERROR
YES
NO
50 INTO MAX. NO. STEPS FORWARD.
L (t_5, $\bar{r}_5$, $\bar{v}_5$) INTO CALLING SEQ. CORKNI/1
BO
BO
CORNKI/1 COMP t_{IP}, ϕ_{IP}, λ_{IP}
IS THERE AN ERROR
NO
YES
CGFER
RESTORE XR4 C(XR4)+1 INTO XR4
CGFEX
ϕ_{IP} INTO 0, 2
λ_{IP} INTO 1,2
RESTORE XR4
RESTORE XR2
EXIT TO 4,4

49. CVAFPA - COMPUTATION OF RELATIVE VELOCITY AND RELATIVE FLIGHT PATH ANGLE AT A SPECIFIED HEIGHT

THIS PROGRAM COMPUTES THE RELATIVE VELOCITY AND THE RELATIVE FLIGHT PATH ANGLE OF THE SPACECRAFT AT A GIVEN HEIGHT. IT IS A SUBROUTINE OF THE AVERAGE VECTOR PROGRAM CFCFAV. (SEE FLOWCHART)

INPUT

A) A SEVEN-LOCATION TABLE CONTAINING t IN FLOATING POINT SECONDS AND THE COMPONENTS OF $\overline{r}$ AND $\overline{v}$ IN FLOATING POINT GEMINI UNITS AT THE COMPLETION OF RETROROCKET BURNING IS INPUT TO THE PROGRAM.

B) LIBRARY SUBROUTINES U3VMAG AND U1ASCO ARE USED.

C) SUBROUTINES BOSCER AND CORKNI (TYPE 2) ARE ALSO REQUIRED.

D) CONSTANTS AND TABLES -

K.MOMG	DEC	-.05883378	NEGATIVE EARTH ROTATION IN RADIANS-GEMINI TIME UNIT
KM0000	DEC	-0	MINUS ZERO
K.OMEG	DEC	.05883378	EARTH ROTATION IN RADIANS/ GEMINI TIME UNIT
K00000	DEC	0	ZERO
K000.5	DEC	.5	FLOATING POINT FIVE-TENTHS
K00001	DEC	1	FIXED POINT ONE
K00002	DEC	2	FIXED POINT TWO
K060.0	DEC	60	FLOATING POINT

KFTSCG	DEC	25936.272	SIXTY FLOATING POINT GEMINI VELOCITY UNIT IN FEET/ SECOND
KMD1A7	PZE	7,0,1	
TCHCFV	A ONE-LOCATION TABLE CONTAINING THE HEIGHT AT WHICH VELOCITY AND GAMMA ARE TO BE COMPUTED IN FLOATING POINT GEMINI LENGTH UNITS		
TCWF.R	A ONE-LOCATION TABLE CONTAINING THE REENTRY WEIGHT OF THE SPACECRAFT IN FLOATING POINT POUNDS		

OUTPUT

A TWO-LOCATION TABLE CONTAINING THE RELATIVE VELOCITY IN FLOATING POINT FEET/SECOND AND THE RELATIVE FLIGHT PATH ANGLE IN FLOATING POINT RADIANS IS OUTPUT FROM THE PROGRAM.

METHOD

UPON ENTERING CVAFPA, A CHECK IS MADE ON THE HEIGHT OF THE INPUT VECTOR. IF THIS HEIGHT IS LESS THAN OR EQUAL TO THE CRITERION HEIGHT, THE RELATIVE VELOCITY AND FLIGHT PATH ANGLE ARE SET TO ZERO AND AN EXIT IS MADE FROM THE PROGRAM.

THE INPUT VECTOR IS INTEGRATED AHEAD AT 60-SECOND STEP SIZES UNTIL A VECTOR BELOW THE CRITERION HEIGHT IS REACHED. AT THIS POINT, THE VECTOR BELOW THE CRITERION HEIGHT IS REJECTED AND THE LAST PREVIOUS VECTOR IS INTEGRATED AGAIN WITH A 30-SECOND STEP SIZE, AND THE CONVERGENCE PROCESS TO COMPUTE A VECTOR AT THE CRITERION

HEIGHT BEGINS.

WITH THE VECTOR THAT RESULTS FROM THIS 30-SECOND INTEGRATION STEP (AND WITH SUCCEEDING VECTORS), THE PROCEDURE FOLLOWED IS -

A) IF THE VECTOR RESULTING FROM AN INTEGRATION STEP IS ABOVE THE CRITERION HEIGHT, IT IS INTEGRATED AHEAD AT ONE-HALF THE PRESENT STEP SIZE.

B) IF THE VECTOR RESULTING FROM AN INTEGRATION STEP IS BELOW THE CRITERION HEIGHT, THE VECTOR AT THE BEGINNING OF THIS INTEGRATION STEP IS INTEGRATED AHEAD AT ONE-HALF THE PRESENT STEP SIZE.

IN EITHER CASE THE NUMBER OF ITERATIONS IS REDUCED BY ONE AND THE PROCESS TERMINATES WITH THE TENTH ITERATION. (THE FIRST VECTOR BELOW THE CRITERION HEIGHT IS THE FIRST ITERATION.) THE STEP SIZE OF THIS FINAL INTEGRATION IS 0.117 SECOND.

THE RELATIVE VELOCITY VECTOR IS COMPUTED FROM THE VECTORS RESULTING FROM THE TENTH ITERATION USING -

$$\bar{V}_R = \begin{pmatrix} \dot{x} + y\omega \\ \dot{y} - x\omega \\ \dot{z} \end{pmatrix}$$

WHERE ω IS THE ROTATIONAL VELOCITY OF THE GEMINI SPHEROID IN RADIANS/GEMINI TIME UNITS. (SEE SECTION 55 FOR A DERIVATION OF THIS FORMULA.)

THE MAGNITUDE OF THE RELATIVE VELOCITY VECTOR AND THE RELATIVE FLIGHT PATH ANGLE ARE THEN COMPUTED.

USAGE

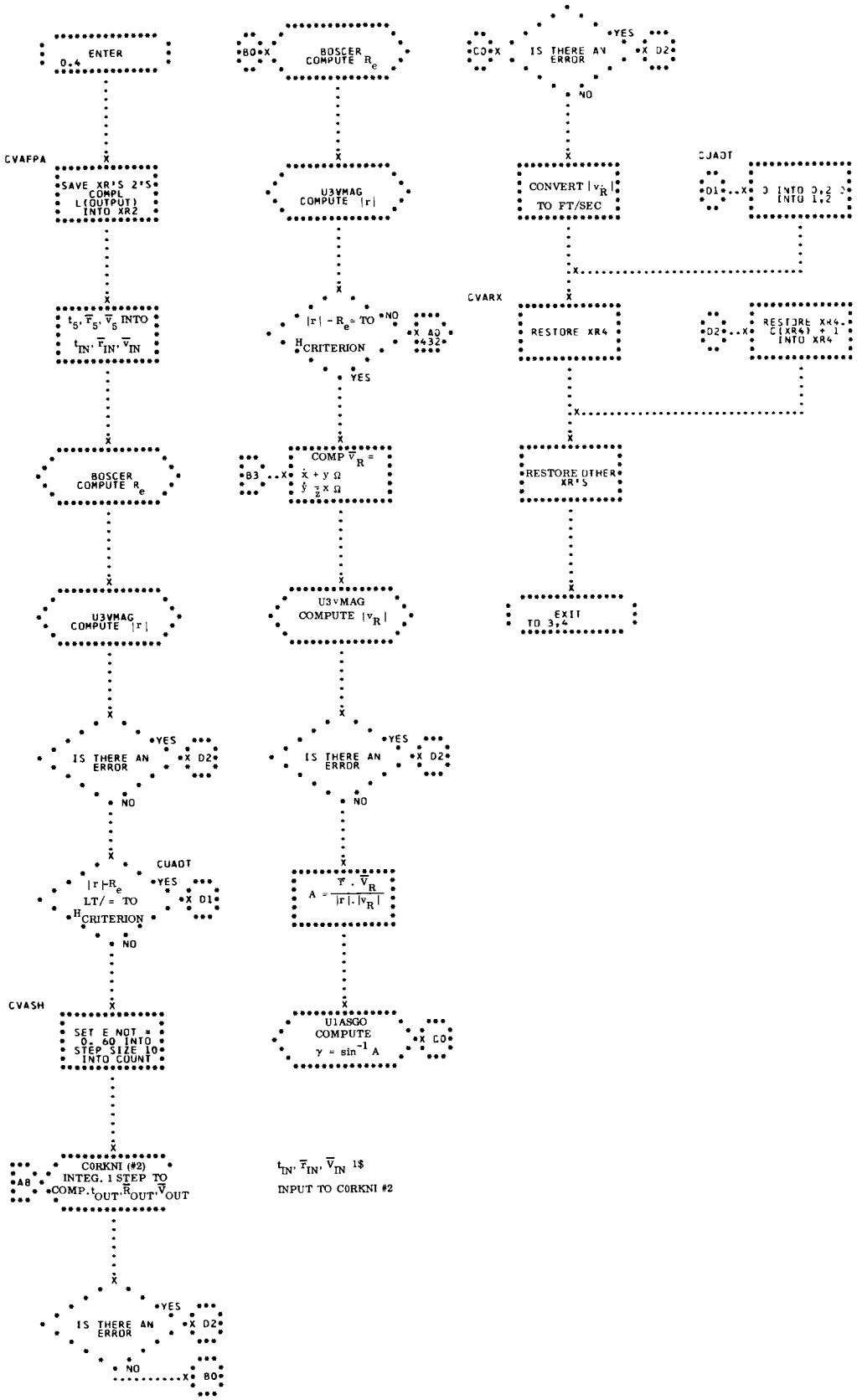
A) CALLING SEQUENCE -

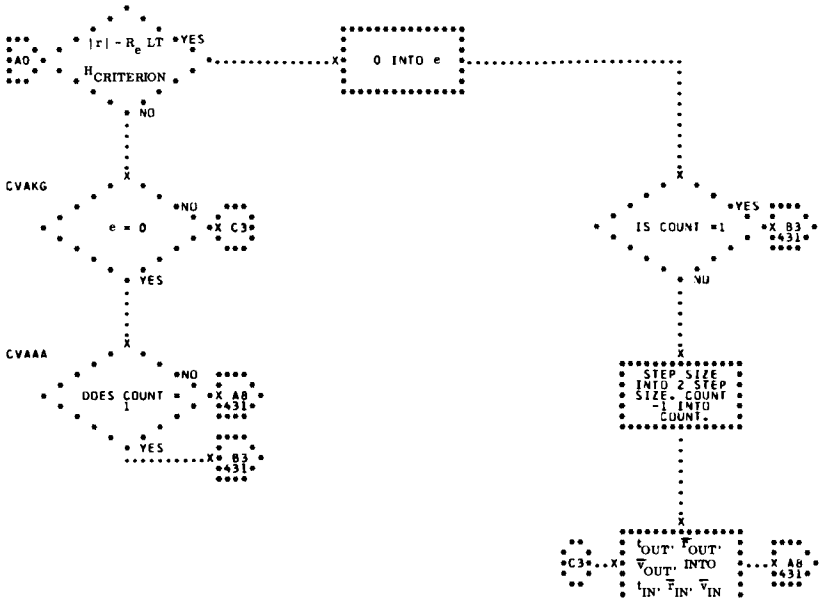
```
TSX       CVAFPA,4
PZE       L(INPUT),,L(OUTPUT)
ERROR     RETURN
NORMAL    RETURN
```

B) STORAGE REQUIRED - 127 DECIMAL LOCATIONS EXCLUDING STORAGE REQUIRED BY EXTERNAL CONSTANTS AND SUBROUTINES.

C) TIME REQUIRED (7094 MAXIMUM) - (108 + 10A) MILLISECONDS WHERE A = NUMBER OF INTEGRATION STEPS TAKEN AT 60-SECOND STEP SIZE. THIS TIME INCLUDES ALL SUBROUTINE TIME EXCEPT THAT OF THE U1ASCO SUBROUTINE.

D) ERROR CONDITIONS - AN ERROR RETURN IS A RESULT OF AN ERROR RETURN FROM A SUBROUTINE.


ENTER
0.4
CVAFPA
SAVE XR'S 2'S COMPL L(OUTPUT) INTO XR2
$t_5, \bar{r}_5, \bar{v}_5$ INTO $t_{IN}, \bar{r}_{IN}, \bar{v}_{IN}$
BOSCER COMPUTE R_e
U3VMAG COMPUTE $|r|$
IS THERE AN ERROR
YES
D2
NO
$|r| \vdash R_e$ LT/= TO $^HCRITERION$
CUADT
D1
CVASH
SET E NOT = 0.60 INTO STEP SIZE IO INTO COUNT
CORKNI (#2) INTEG. 1 STEP TO COMP. $t_{OUT}, \bar{r}_{OUT}, \bar{v}_{OUT}$
A8
B0
BOSCER COMPUTE R_e
U3VMAG COMPUTE $|r|$
$|r| - R_e \geq$ TO $^HCRITERION$
NO
A0
4432
YES
COMP $\bar{v}_R$ = $\dot{x} + y\,\Omega$ $\dot{y} \frac{}{z} x\,\Omega$
B3
U3VMAG COMPUTE $|v_R|$
$A = \frac{\bar{r} \cdot \bar{v}_R}{|r| \cdot |v_R|}$
U1ASGO COMPUTE $\gamma = \sin^{-1} A$
C0
$t_{IN}, \bar{r}_{IN}, \bar{v}_{IN}$ 1$
INPUT TO CORKNI #2
IS THERE AN ERROR
YES
D2
NO
CONVERT $|v_R|$ TO FT/SEC
CUADT
D1
CVARX
RESTORE XR4
D2
RESTORE OTHER XR'S
EXIT TO 3,4


AO
|r| - Re LT HCRITERION
YES
NO
0 INTO e
CVAKG
e = 0
NO
YES
C3
CVAAA
DOES COUNT = 1
NO
YES
A8
B3
431
IS COUNT =1
YES
NO
B3
431
STEP SIZE INTO 2 STEP SIZE. COUNT -1 INTO COUNT.
C3
tOUT, rOUT, vOUT, INTO tIN, rIN, vIN
A8
431

50. A3GSCP - COMPUTATION OF LATITUDE, LONGITUDE AND ALTITUDE

GIVEN AN INERTIAL POSITION VECTOR, $\bar{r}$, OF THE SPACECRAFT AT A PARTICULAR TIME, T, THIS PROGRAM COMPUTES THE GEODETIC LATITUDE (ϕ) AND THE LONGITUDE (λ) OF THE SUB-SPACECRAFT POSITION. IT ALSO COMPUTES THE GEODETIC ALTITUDE (H) OF THE SPACECRAFT. (SEE FLOWCHART.)

INPUT

A) A FOUR-LOCATION TABLE CONTAINING, RESPECTIVELY, t AND THE COMPONENTS OF $\bar{r}$ IS INPUT TO THE PROGRAM. THE COMPONENTS OF ARE IN FLOATING POINT GEMINI UNITS, WHILE t MAY BE EITHER FLOATING POINT SECONDS OR FIXED POINT WITH SECONDS IN THE DECREMENT AND MINUTES IN THE ADDRESS.

B) LIBRARY SUBROUTINES U1SQRT, U1ATAB, AND U1ATNA ARE USED.

C) CONSTANTS -

K00000	FIXED POINT ZERO
K00060	FIXED POINT SIXTY
K001.0	FLOATING POINT ONE
KCH233	OCTAL MASK USED IN TIME FORMAT CONVERSION
KKK2PI	FLOATING POINT VALUE OF 2π
K.INB2	INVERSE OF THE SQUARE OF THE SEMI-MINOR AXIS OF THE GEMINI SPHEROID IN FLOATING POINT GEMINI UNITS
K.B2GU	SQUARE OF THE SEMI-MINOR AXIS OF THE GEMINI SPHEROID IN FLOATING POINT GEMINI UNITS

K.ECC2 SQUARE OF THE ECCENTRICITY OF THE GEMINI SPHEROID IN FLOATING POINT FORMAT

K.OMES ANGULAR ROTATIONAL VELOCITY OF THE EARTH IN FLOATING POINT RADIANS/SECOND

KLAMDO INERTIAL LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH IN FLOATING POINT RADIANS. (NOTE - KLAMDO IS NOT A TRUE CONSTANT. IT IS A PARAMETER WHOSE VALUE CHANGES WITH EACH DAY OF THE YEAR. FOR SIMPLICITY, IT IS CURRENTLY ASSIGNED THE VALUE ZERO IN THE GEMINI SYSTEM.)

OUTPUT

A THREE-LOCATION TABLE CONTAINING, RESPECTIVELY, λ AND ϕ IN FLOATING POINT RADIANS AND H IN FLOATING POINT GEMINI UNITS IS OUTPUT FROM THE PROGRAM. λ RANGES FROM 0 TO 2π RADIANS, ϕ RANGES FROM $-\pi/2$ TO $+\pi/2$ RADIANS. (THE COMPUTATION OF H MAY BE OMITTED IF DESIRED. SEE USAGE.)

METHOD

IF A3GSCP IS ENTERED WITH TIME IN THE FIXED POINT FORMAT, IT IS IMMEDIATELY CONVERTED TO FLOATING POINT SECONDS.

THE ANGLE ϕ IS THE GEODETIC LATITUDE OF THE INTERSECTION OF THE GIVEN RADIUS VECTOR AND THE SURFACE OF THE EARTH. IT IS USED AS A FIRST ESTIMATE OF ϕ AND IS DESIGNATED BY ϕ_i.

USING A STANDARD GEODESY TEXTBOOK FORMULA,

$$\tan \phi_i = \frac{a^2}{b^2} \tan \phi'$$

SINCE A = 1 GEMINI UNIT,

$$\tan \phi_i = \frac{1}{b^2} \tan \phi' = \frac{1}{b^2} \frac{Z}{\sqrt{X^2 + Y^2}}$$

USING THIS TAN THE FOLLOWING TWO EQUATIONS CAN BE COMPUTED -

$$Q = \sqrt{b^2 \tan^2 \phi_i + 1}$$

$$\tan \phi_{i+1} = \frac{\frac{e^2 \tan \phi_i}{Q} + Z}{\sqrt{X^2 + Y^2}}$$

THIS COMPUTED VALUE OF TAN φ_{i+1} IS USED AS THE NEW ESTIMATE OF TAN ϕ. A SECOND ITERATION OF THE ABOVE TWO EQUATIONS IS MADE WITH TAN ϕ_i=TAN ϕ_{i+1}. THE SECOND COMPUTED VALUE OF TAN ϕ_{i+1} IS THE TRUE VALUE OF TAN ϕ CORRECT TO EIGHT SIGNIFICANT PLACES.

USING THE ARCTANGENT ROUTINE, ϕ IS DETERMINED AND IS SUCH THAT $-\pi/2 \leq \phi \leq \pi/2$.

NEXT, IS COMPUTED USING THE STANDARD EXPRESSION FOR IN THE GEMINI SYSTEM -

$$\lambda = \tan^{-1} \frac{Y}{X} - \omega t - \lambda_g$$

AND IS SUCH THAT $0 \leq \lambda \leq 2\pi$.

IF THE SIGN OF 1,4 IS POSITIVE, THE HEIGHT, H, IS TO BE COMPUTED. THE EQUATION USED IS

$$H = A \sec \phi - \frac{\sec \phi}{Q}$$

WHERE

$$A = \sqrt{X^2 + Y^2}$$

$$\sec \phi = \sqrt{\tan^2 \phi + 1}$$

$$Q = \sqrt{b^2 \tan^2 \phi + 1}$$

USAGE

A) CALLING SEQUENCE -

IF THE INPUT TIME IS IN FIXED POINT (SECONDS IN THE DECREMENT AND MINUTES IN THE ADDRESS), THE FOLLOWING CALLING SEQUENCE IS USED -

```
TSX     A3GSCP,4
G       L(INPUT),,L(OUTPUT)
ERROR   RETURN
NORMAL  RETURN
```

HOWEVER, IF THE INPUT TIME IS IN FLOATING POINT SECONDS, THE FOLLOWING CALLING SEQUENCE IS USED-

```
TSX     A3GP12,4 (OR TSX A3GSCP+12,4)
G       L(INPUT),,L(OUTPUT)
ERROR   RETURN
NORMAL  RETUNR
```

G=PZE INDICATES THAT H IS TO BE COMPUTED.

G=MZE INDICATES THAT H IS NOT TO BE COMPUTED.

B) SPACE REQUIRED - 100 LOCATIONS EXCLUDING THAT REQUIRED BY EXTERNAL CONSTANTS AND SUBROUTINES.

C) ERROR CONDITIONS - AN ERROR RETURN IS A RESULT OF AN ERROR RETURN FROM EITHER U1SQRT OR U1ATAB. (IT SHOULD BE NOTED THAT U1ATAB GIVES AN ERROR

RETURN WHEN THE INPUT VECTOR TO A3GSCP IS ALL ZEROS.)

C) TIME REQUIRED (7094 MAXIMUM) -

ENTERING AT A3GSCP (FIXED POINT TIME)

1) COMPUTING H - 2.07 MILLISECONDS

2) NOT COMPUTING H - 1.63 MILLISECONDS

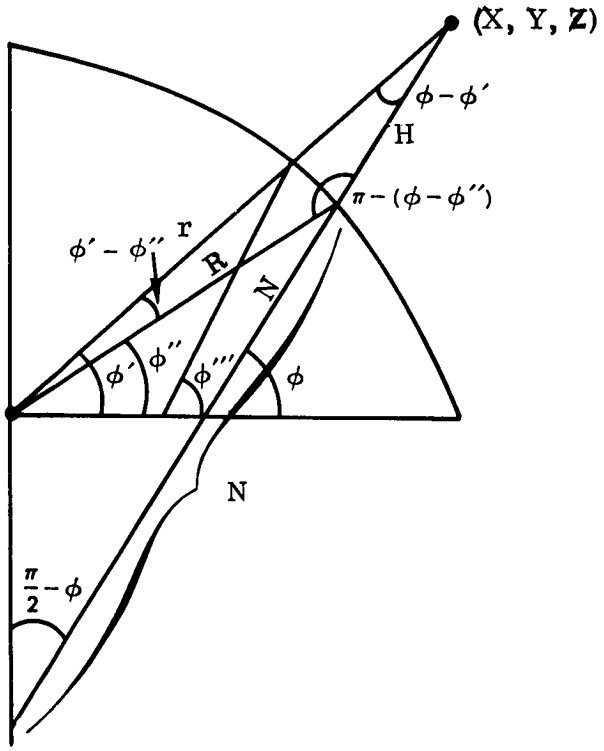
ENTERING AT A3GP12 (FLOATING POINT TIME)

1) COMPUTING H - 2.01 MILLISECONDS

2) NOT COMPUTING H -1.56 MILLISECONDS

E) CHECKOUT - THE PROGRAM WAS CHECKED OUT USING $\bar{r}$ VECTORS WHOSE GEODETIC LATITUDES VARIED AT 15° INTERVALS FROM 90°S TO 90°N. PROGRAM CALCULATIONS OF ϕ,λ AND H WERE CHECKED AGAINST HAND CALCULATIONS.

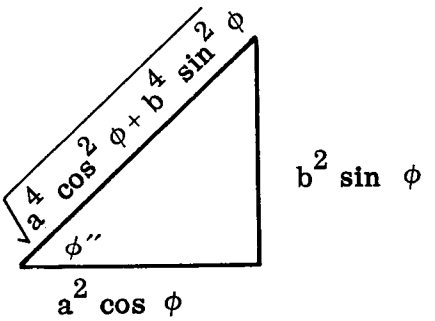
DERIVATIONS OF EQUATIONS USED IN A3GSCP



(1) $R = \frac{r \sin(\phi - \phi')}{\sin(\phi - \phi'')} = \frac{N \cos \phi}{\cos \phi''}$ (LAW OF SINES)

(2) $\tan \phi'' = \frac{b^2}{a^2} \tan \phi$ (GEODESY FORMULA)

THE FOLLOWING ILLUSTRATION SHOWS THE TRIGONOMETRIC RELATIONSHIPS FOR THE ANGLE ϕ '' USING EQUATION (2).



(3) $\sin(\phi-\phi'') = \sin\phi\cos\phi'' - \cos\phi\sin\phi''$

$$= \sin\phi\frac{a^2\cos\phi}{\sqrt{a^4\cos^2\phi + b^4\sin^2\phi}} - \cos\phi\frac{b^2\sin\phi}{\sqrt{a^4\cos^2\phi + b^4\sin^2\phi}}$$

$$= \frac{(a^2-b^2)\sin\phi\cos\phi}{\sqrt{a^4\cos^2\phi + b^4\sin^2\phi}}$$

(4) $r\sin(\phi-\phi') = r\sin\phi\cos\phi' - r\cos\phi\sin\phi'$

$$= \sqrt{X^2+Y^2}\sin\phi - Z\cos\phi$$

SUBSTITUTING (3) AND (4) INTO (1):

$$\frac{\sqrt{X^2+Y^2}\sin\phi - Z\cos\phi}{(a^2-b^2)\sin\phi\cos\phi} = \frac{N(\cos\phi)}{\cos\phi''\sqrt{a^4\cos^2\phi + b^4\sin^2\phi}} = \frac{N}{a^2}$$

OR

(5) $$\sqrt{X^2+Y^2}\tan\phi - Z = \frac{a^2-b^2}{a^2}N\sin\phi = e^2 N\sin\phi$$

(6) $$N = \frac{a}{\sqrt{1-e^2\sin^2\phi}} = \frac{1}{\sqrt{1-e^2\sin^2\phi}}$$ (GEODESY FORMULA)

$$= \frac{1}{\sqrt{1-(1-b^2)\sin^2\phi}} = \frac{1}{\sqrt{\cos^2\phi + b^2\sin^2\phi}}$$

$$= \frac{\sec\phi}{\sqrt{1+b^2\tan^2\phi}}$$

SUBSTITUTING (6) INTO (5):

$$\sqrt{X^2 + Y^2}\tan\phi - Z = e^2 \frac{\sin\phi \sec\phi}{\sqrt{1 + b^2 \tan^2\phi}}$$

OR

$$(7) \qquad \tan\phi = \frac{e^2 \tan\phi}{\frac{\sqrt{1 + b^2 \tan^2\phi}}{\sqrt{X^2 + Y^2}}} + Z$$

$$\frac{N + H}{r} = \frac{\cos\phi'}{\cos\phi} \text{ (law of sines)}$$

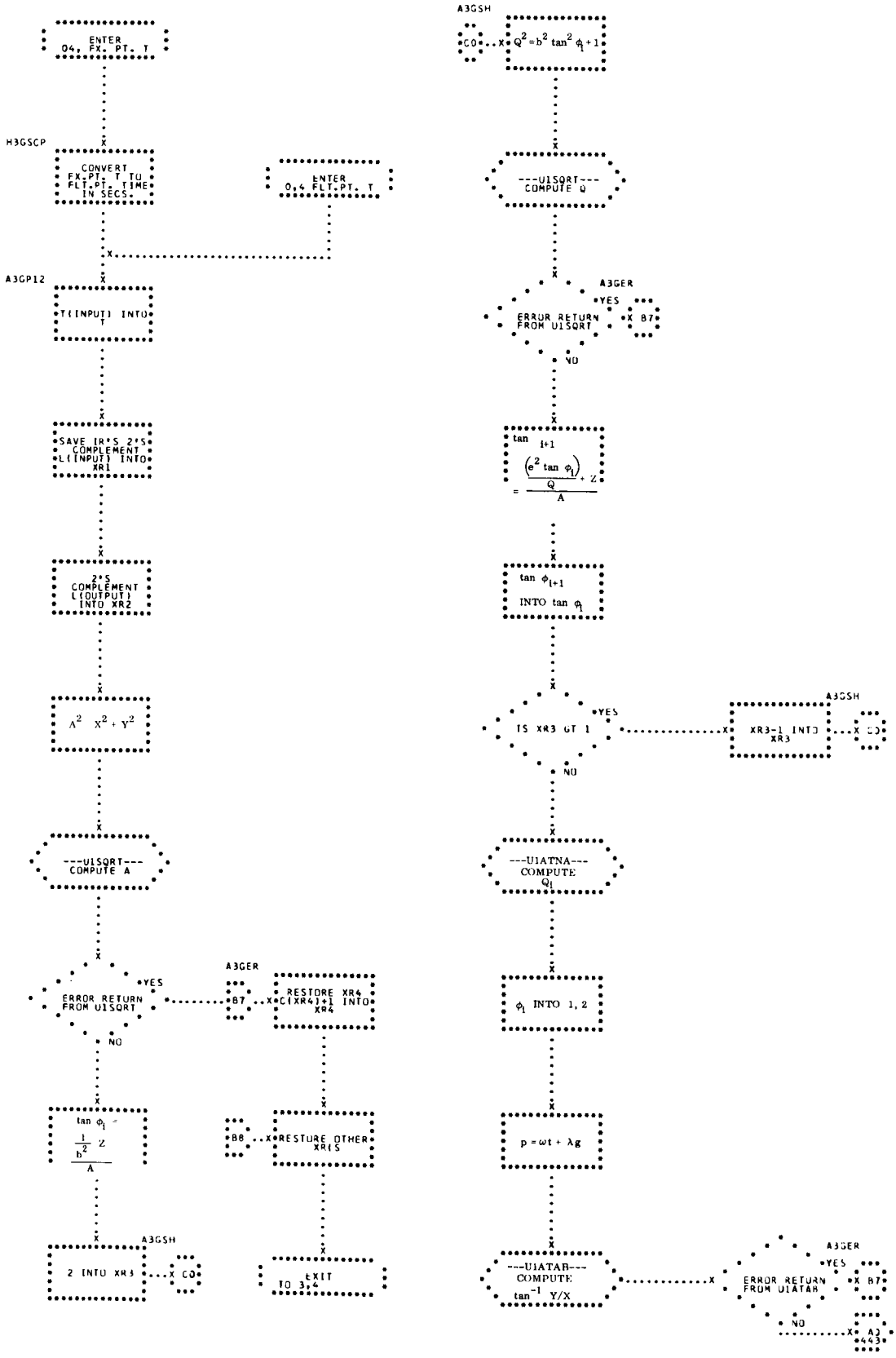
$$N + H = \frac{\sqrt{X^2 + Y^2}}{\cos\phi}$$

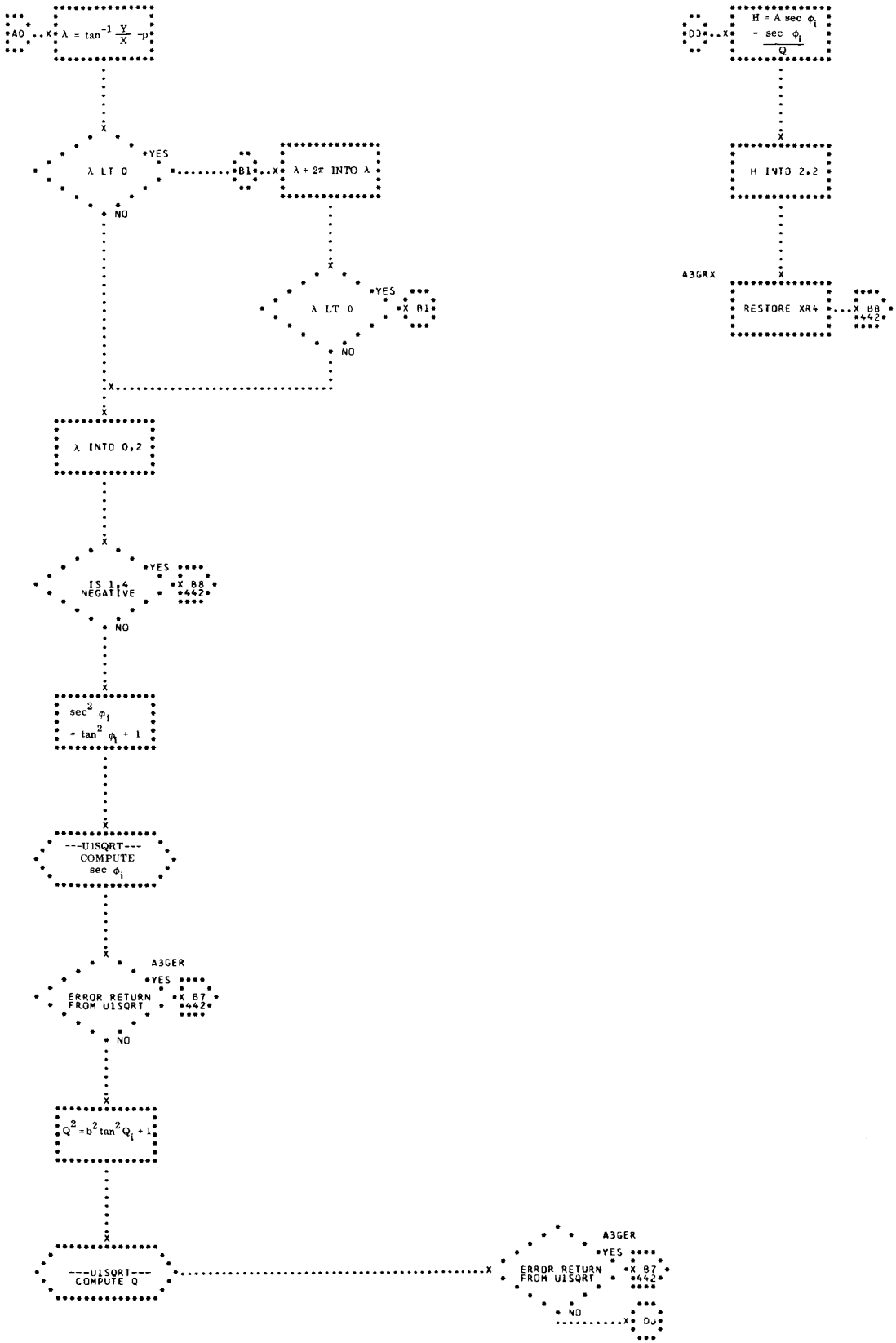
$$(8) \quad H = \sec\phi\sqrt{X^2 + Y^2} - N$$

SUBSTITUTING (6) INTO (8):

$$(9) \quad H = \sec\phi\sqrt{X^2 + Y^2} - \frac{\sec\phi}{\sqrt{1 + b^2 \tan^2\phi}}$$

EQUATIONS (6), (7), AND (9) ARE USED IN A3GSCP AS STATED BECAUSE OF THE RELATIVE SIMPLICITY OF COMPUTING THEM IN THE LEAST AMOUNT OF MACHINE TIME.


ENTER 04, FX. PT. T
A3GSCP
CONVERT FX.PT. T TO FLT.PT. TIME IN SECS.
ENTER 0.4 FLT.PT. T
A3GP12
T(INPUT) INTO T
SAVE IR'S 2'S COMPLEMENT L(INPUT) INTO XR1
2'S COMPLEMENT (OUTPUT) INTO XR2
A² X² + Y²
---U1SQRT--- COMPUTE A
ERROR RETURN FROM U1SQRT
YES
NO
A3GER
B7
RESTORE XR4 C(XR4)+1 INTO XR4
B8
RESTORE OTHER XR'S
tan φ1 = 1/b² Z / A
2 INTO XR3
A3GSH
CO
EXIT TO 3,4
A3GSH
C0
Q² = b² tan² φ1 + 1
---U1SQRT--- COMPUTE Q
ERROR RETURN FROM U1SQRT
A3GER
YES
B7
NO
tan i+1 (e² tan φ1)/Q + Z / A
tan φi+1 INTO tan φ1
IS XR3 GT 1
YES
XR3-1 INTO XR3
A3GSH
CO
NO
---U1ATNA--- COMPUTE Φ1
φ1 INTO 1, 2
p = ωt + λg
---U1ATAB--- COMPUTE tan⁻¹ Y/X
ERROR RETURN FROM U1ATAB
A3GER
YES
B7
NO
A3 *443*


A0
$\lambda = \tan^{-1}\frac{Y}{X} - p$
λ LT 0
YES
B1
$\lambda + 2\pi$ INTO λ
NO
λ LT 0
YES
B1
NO
λ INTO 0,2
IS 1,4 NEGATIVE
YES
X BB *442*
NO
$\sec^2 \phi_i = \tan^2 \phi_i + 1$
---U1SQRT---
COMPUTE
$\sec \phi_i$
A3GER
ERROR RETURN FROM U1SQRT
YES
X B7 *442*
NO
$Q^2 = b^2 \tan^2 Q_i + 1$
---U1SQRT---
COMPUTE Q
A3GER
ERROR RETURN FROM U1SQRT
YES
X B7 *442*
NO
DJ
D0
$H = A \sec \phi_i - \frac{\sec \phi_i}{Q}$
H INTO 2,2
A3GRX
RESTORE XR4
X BB *442*

51. BOSCER - EARTH RADIUS DETERMINATION

GIVEN A VECTOR $\bar{r}$ (X, Y, Z) OF THE SPACECRAFT, THE BOSCER PROGRAM COMPUTES THE RADIUS OF THE EARTH ALONG THE RADIUS VECTOR, $\bar{r}$. (SEE FLOWCHART.)

INPUT

BOSCER USES THE SQUARE ROOT ROUTINE, U1SQRT.

THE INPUT TO THE BOSCER PROGRAM IS A TABLE CONTAINING AT LEAST FOUR LOCATIONS WITH THE X, Y, AND Z COMPONENTS OF THE GIVEN $\bar{r}$ IN THE SECOND, THIRD, AND FOURTH LOCATIONS, RESPECTIVELY. THESE ARE IN FLOATING POINT GEMINI UNITS.

OUTPUT

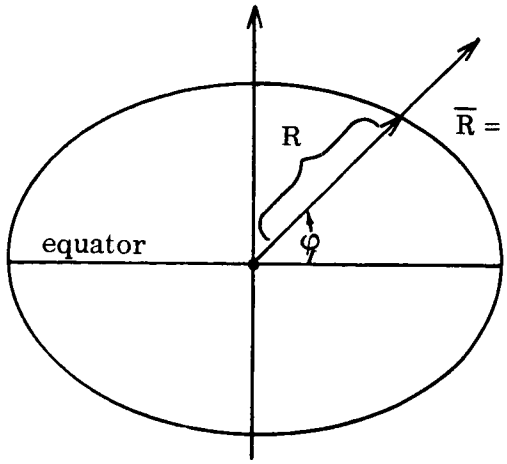
THE OUTPUT OF BOSCER IS THE RADIUS OF THE EARTH ALONG A GIVEN RADIUS VECTOR LEFT IN THE ACCUMULATOR UPON EXITING FROM THE PROGRAM. OUTPUT IS IN FLOATING POINT GEMINI UNITS.

METHOD

A) EQUATION USED -

$$R = \sqrt{\frac{(X^2 + Y^2 + Z^2)}{\frac{Z^2}{b^2} + (X^2 + Y^2)}}$$

B) THIS EQUATION MAY BE DERIVED AS FOLLOWS -



In this diagram the X and Y axes lie in the equatorial plane. φ is the geocentric latitude of $\overline{r}$

THE EQUATION OF THE GEMINI SPHEROID IS

$$\frac{X_1^2}{a^2} + \frac{Y_1^2}{a^2} + \frac{Z_1^2}{b^2} = 1$$

THUS
$$\frac{X_1^2 + Y_1^2}{a^2} + \frac{Z_1^2}{b^2} = 1 \qquad (1)$$

NOW
$$Z_1 = R \sin\varphi = R \frac{Z}{\sqrt{X^2 + Y^2 + Z^2}} \qquad (2)$$

AND
$$\sqrt{X_1^2 + Y_1^2} = R \cos\varphi = R \frac{\sqrt{X^2 + Y^2}}{\sqrt{X^2 + Y^2 + Z^2}} \qquad (3)$$

ALSO, A=1 WHEN ALL LENGTHS ARE EXPRESSED IN GEMINI UNITS (4)

SUBSTITUTING (2), (3), AND (4) INTO (1) -

$$\frac{R^2(X^2 + Y^2)}{(X^2 + Y^2 + Z^2)} + \frac{R^2 Z^2}{b^2(X^2 + Y^2 + Z^2)} = 1$$

$$R^2(X^2 + Y^2) + \frac{R^2 Z^2}{b^2} = X^2 + Y^2 + Z^2$$

$$R^2 \left(X^2 + Y^2 + \frac{Z^2}{b^2}\right) = X^2 + Y^2 + Z^2$$

$$R = \sqrt{\frac{X^2 + Y^2 + Z^2}{\frac{Z^2}{b^2} + (X^2 + Y^2)}}$$

USAGE

A) CALLING SEQUENCE -

```
TSX     BOSCER,4
PZE     L(INPUT TABLE)
RETURN  EARTH RADIUS IN AC
```

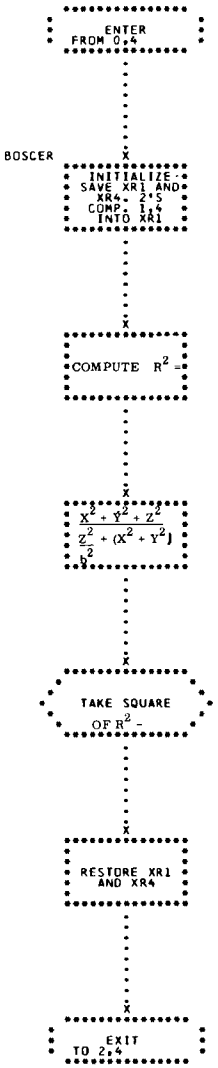
B) SPACE REQUIRED - 30 LOCATIONS EXLCUDING SUBROUTINE, CONSTANT, AND INPUT TABLE.

C) CONSTANT -

```
K.INB2  DEC    1.0067383   INVERSE OF SQUARE OF
                           SEMIMINOR AXIS OF
                           GEMINI SPHEROID
```

D) TIME REQUIRED - THE TIME REQUIRED FOR THE BOSCER PROGRAM IS .315 MILLISECONDS (IBM 7094 COMPUTER TIME).

E) CHECKOUT - THE BOSCER PROGRAM WAS CHECKED OUT USING $\overline{r}$ COMPONENTS WITH A HEIGHT KNOWN TO BE 450,000 FEET. PROGRAM CALCULATIONS WERE CHECKED AGAINST HAND CALCULATIONS.


ENTER FROM 0,4
BOSCER
INITIALIZE - SAVE XR1 AND XR4, 2**15 COMP. 1,4 INTO XR1
COMPUTE R^2 =
$\frac{X^2 + Y^2 + Z^2}{\frac{Z^2}{b^2} + (X^2 + Y^2)}$
TAKE SQUARE OF R^2 -
RESTORE XR1 AND XR4
EXIT TO 2,4

52. MFCCMN - SUFFIX TO CCMAIN

THE MAIN FUNCTION OF THIS PROGRAM IS TO PROCESS THE OUTPUT QUANTITIES OF THE MAIN LAUNCH CALCULATIONS PROGRAM, CCMAIN. IT ALSO DETERMINES WHETHER A PHASE CHANGE HAS BEEN REQUESTED AND, ACCORDINGLY, PERFORMS THE INTERFACE WITH THE LOW ABORT, HIGH ABORT, OR ORBIT PROGRAMS. IN ADDITION, MFCCMN CONTROLS THE ON-LINE PRINTING OF COMPUTER RECOMMENDATIONS AND THE ASSOCIATED LAUNCH COMPUTATIONS. (SEE FLOWCHART.)

INPUT

A) THIS PROGRAM USES SUBROUTINES CUPDAT AND CABDAT.

B) MACROS QENBZ, QENBA, QUEUE, TRNON, TRNOF, QPSLF, AND QMOVE ARE USED.

C) CONSTANTS -

KM0000	K00060	KRTDEG
K00000	K00080	K8.3M3
K00001	K005.0	K000P1
K00002	K006.0	KTFORD
K00003	K060.0	KTBACK
K00006	KCH232	KSETNG
K00008	KCH233	KLSACM

D) COMMUNICATION CELLS - THE CONTENTS OF THESE CELLS ARE EITHER USED BY MFCCMN OR ARE COMPUTED BY MFCCMN.

MC1RVT	MCGTLO	MCPERS
MC2RVT	MCHFS1	MCPFLD

MCACBD	MCHFSC	MCPHSE
MCACTV	MCHOMS	MCPSEL
MCAFIN	MCIMPA	MCPSPS
MCBGNG	MCIMPD	MCREST
MCBJMN	MCIMPL	MCSBDA
MCBDTF	MCIONG	MCSGNG
MCCCMD	MCIPBS	MCSINP
MCCFHP	MCLAST	MCSOPT
MCCURT	MCLFTM	MCSPD3
MCDISP	MCLGT1	MCSRST
MCEAAA	MCLGT2	MCSSOR
MCEABM	MCLIDT	MCSTTG
MCEABS	MCLIMP	MCTBAR
MCELFT	MCCLOT	MCTLST
MCEORP	MCLNOR	MCTOFS
MCESSC	MCLOPR	MCTRV2
MCFLSS	MCMAN2	MCTRVT
MCFTRC	MCMAN5	MCVRLM
MCGMLM	MCMAOS	MCWICH
MCGTIN	MCOPT1	MCXXXX

E) PARAMETERS -

A	MNCCIP	MNMESS
B	MNCCMN	MNMPRK
C	MNDIFC	MNNSRS
D	MNDIFK	MNORMC
E	MNDIFM	MNOUR1
G	MNEDBD	MNRSYS
MNABRT	MNLANA	MNSENS
MNBBNI	MNLED1	MNSRST
MNCCGB	MNLED2	MNWRRS

F) TABLES -

TCCOUT	TMGEBP	TMOUT3
TDCDEF	TMGLVC	TMOUT4
TDNIPM	TMILVB	TMOUT5
TDRANV	TMILVC	TMOUT6
TMABDV	TMIPB2	TMOUT7
TMBDOL	TMIPBP	TMREFR
TMBD1L	TMIPC2	TMREST
TMBD2L	TMIPCP	TMRV3I
TMBDA2	TMIPS2	TMSLB2
TMBDAP	TMIPSL	TMSLBD
TMBLVC	TMLANA	TMSMOL
TMCYCL	TMLIPC	TMSM1L
TMCYNO	TMMANZ	TMSM2L
TMERTS	TMMPRK	TMSML2
TMGEB2	TMOUT1	TMSOLP

METHOD

MFCCMN IS COMPRISED OF SIX MAIN BLOCKS OF LOGIC. THESE ARE SINGLE VECTOR PROCESSING, BOOSTER GO/NO-GO PROCESSING, AVERAGE VECTOR PROCESSING, LOW ABORT INTERFACE, HIGH ABORT INTERFACE, AND ORBIT INTERFACE. THESE WILL BE TREATED SEPARATELY IN THE FOLLOWING DISCUSSION.

SINGLE VECTOR PROCESSING

A) THE QUANTITIES IN THE APPROPRIATE CCMAIN OUTPUT TABLE ARE STORED IN THE TABLE TMLANA FOR USE BY THE OUTPUT DISPLAY PROGRAM OOLANA.

B) THE INPUT IS SET UP FOR THE IMPACT PREDICTOR, CIMPRK, EVERY SECOND PASS.

C) MFCCMN DETERMINES WHETHER OR NOT THE LOW ABORT PHASE IS TO BE ENTERED. IF SO, CONTROL IS GIVEN TO THE LOW ABORT INTERFACE.

BOOSTER GO/NO-GO PROCESSING

A) AS IN SINGLE VECTOR PROCESSING, CCMAIN OUTPUT IS STORED IN TMLANA FOR OUTPUT DISPLAY.

B) IN EVERY PASS FOR THE GO/NO-GO COMPUTATIONS WITH THE EXCEPTION OF THE LAST PASS, A RETURN IS MADE TO MPCCMN FOR THE COMPUTATION OF THE REMAINING RECOMMENDATIONS.

C) ON THE LAST PASS OF THE GO/NO-GO COMPUTATIONS, THE COMPUTER'S RECOMMENDATIONS AND PERTINENT LAUNCH COMPUTATIONAL QUANTITIES ARE PRINTED ON-LINE.

D) IF THE HIGH ABORT PHASE OR THE ORBIT PHASE IS TO BE ENTERED, CONTROL IS GIVEN TO THE APPROPRIATE INTERFACE PROGRAMS.

AVERAGE VECTOR PROCESSING

THIS PROCESSING IS BEGUN AFTER THE COMPLETION OF THE GO/NO-GO COMPUTATIONS.

A) ON ALL DISPLAY PASSES, THE SELECTED SOURCE OUTPUT FROM CCMAIN IS STORED IN TMLANA FOR OUTPUT

DISPLAY. A RETURN IS THEN MADE TO MPCCMN FOR BERMUDA DATA COMPUTATIONS.

B) ON THOSE PASSES WHICH ARE NOT DISPLAY PASSES, IF BERMUDA DATA IS AVAILABLE, THE COMPUTER'S RECOMMENDATIONS AND THE ASSOCIATED QUANTITIES FROM THIS DATA ARE PRINTED ON-LINE. IF BERMUDA IS THE SOURCE IN PROCESS, THIS ON-LINE PRINTING OCCURS EVERY 60 SECONDS. THIS SAME CYCLIC ON-LINE PRINTING WILL OCCUR WITH IP SPACECRAFT SOURCES PROVIDING THAT BERMUDA HAS ALREADY BEEN SELECTED AND IP IS THE SELECTED SOURCE.

C) IF THE HIGH ABORT OR ORBIT PHASE IS TO BE ENTERED, CONTROL IS GIVEN TO THE APPROPRIATE INTERFACE PROGRAMS.

LOW ABORT INTERFACE

IF AM3 (ABORT MODE THREE) HAS NOT OCCURRED, AND AAA (ASTRONAUT ACTUATED ABORT) HAS OCCURRED, THE LOW ABORT INTERFACE IS INITIATED.

A) ONCE AAA HAS BEEN RECEIVED, TIMES ARE SET UP FOR COLLECTING VECTORS ON THE BOOSTER SOURCES, B/ GE, IP MISTRAM, AND IP AZUSA, FROM t_{AAA} +4 TO +9 SECONDS. AT t_{AAA} +11 SECS. THE AVERAGE VECTORS ARE OBTAINED AND THE ABORT SWITCH IS INTERROGATED.

B) WHEN ABS (ABORT PHASE STARTED) HAS OCCURRED, THE SELECTED SOURCE IS EXAMINED TO DETERMINE WHICH VECTOR TO GIVE TO THE ABORT PHASE. IF THE

SOURCE IS A BOOSTER SOURCE, THE AVERAGE VECTOR OBTAINED AT t_{AAA} +11 IS USED. IF THE SOURCE IS A SPACECRAFT SOURCE, AN AVERAGE VECTOR IS COMPUTED ON THAT SOURCE. IF ERS HAS OCCURRED AND THE VECTOR IS FROM A BOOSTER SOURCE THE DELTA VELOCITY DUE TO THE ERS IS ADDED TO THE VECTOR.

C) IF ERS (EMERGENCY RETRO SALVO) HAS NOT YET OCCURRED, THE BERMUDA SUBCHANNELS ARE ACTIVATED IMMEDIATELY. IF ERS HAS OCCURRED THE TIME TO ACTIVATE BERMUDA FOR THE ABORT PROGRAM IS SET EQUAL TO THE TIME OF ERS + 6 SECONDS.

D) THE ABORT PROGRAMS ARE MOVED INTO A CORE FOR EXECUTION.

E) THE CONTENTS OF TMCYCL AND TMCYNO ARE CHANGED TO INDICATE THAT IN PLACE OF PROCESSING CCMAIN EVERY HALF SECOND, MABORT WILL BE ENTERED EVERY 6 SECONDS. THE ABORT VECTOR IS UPDATED TO CURRENT TIME AND PLACED IN THE INITIAL BUFFER FOR USE BY CLABRT.

F) CCMAIN IS THEN SUPPRESSED AND THE ABORT PROGRAM MABORT IS REQUESTED.

HIGH ABORT INTERFACE

IF AM3 HAS OCCURRED AND THE BOOSTER GO/NO-GO SEQUENCE HAS BEEN COMPLETED, THE HIGH ABORT INTERFACE LOGIC MAY BE ENTERED.

A) THE AVERAGE VECTOR FROM THE SELECTED SOURCE IS MOVED TO THE ABORT DEFINING VECTOR BUFFER.

B) THE COORDINATE CONVERSION PROGRAMS (COCOOI AND CSCOO2) ARE SUPPRESSED SO THAT ONLY TELEMETRY WILL BE PROCESSED.

C) THE SUBCHANNELS FOR LOW-SPEED INPUTS ARE ACTIVATED.

D) THE ABORT DEFINING VECTOR IS PRINTED ON-LINE.

E) ALL INPUT AND INDICATORS ARE SET FOR THE HIGH ABORT PROGRAMS. THE CONTENTS OF TMCYCL AND TMCYNO ARE CHANGED AS IN LOW ABORT INTERFACE.

F) CCMAIN IS SUPPRESSED AND THE ABORT PROGRAM MABORT IS REQUESTED.

ORBIT INTERFACE

ONCE THE GO/NO-GO COMPUTATIONS HAVE BEEN COMPLETED, MFCCMN PROCESSES DATA UNTIL RECOGNITION OF THEE ORBIT SWITCH, AT WHICH TIME CONTROL IS TRANSFERRED TO THE ORBIT INTERFACE LOGIC.

A) THE AVERAGE VECTOR FROM THE SELECTED SOURCE IS MOVED TC TDRANV AS THE ORBIT DEFINING VECTOR.

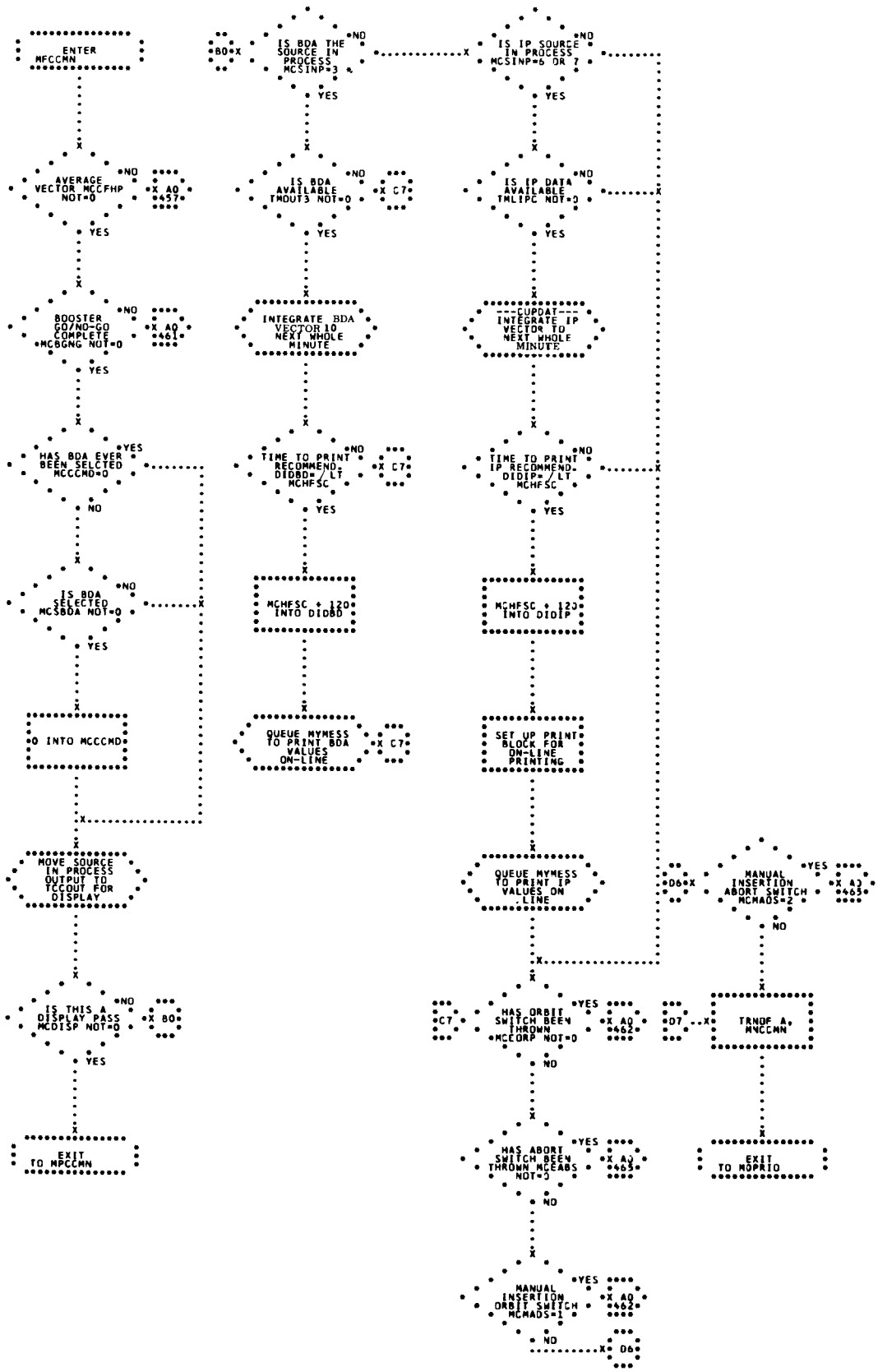
B) THE SUBCHANNELS FOR HIGH-SPEED INPUTS ARE DEACTIVATED WHILE THE SUBCHANNELS FOR LOW SPEED INPUTS ARE ACTIVATED.

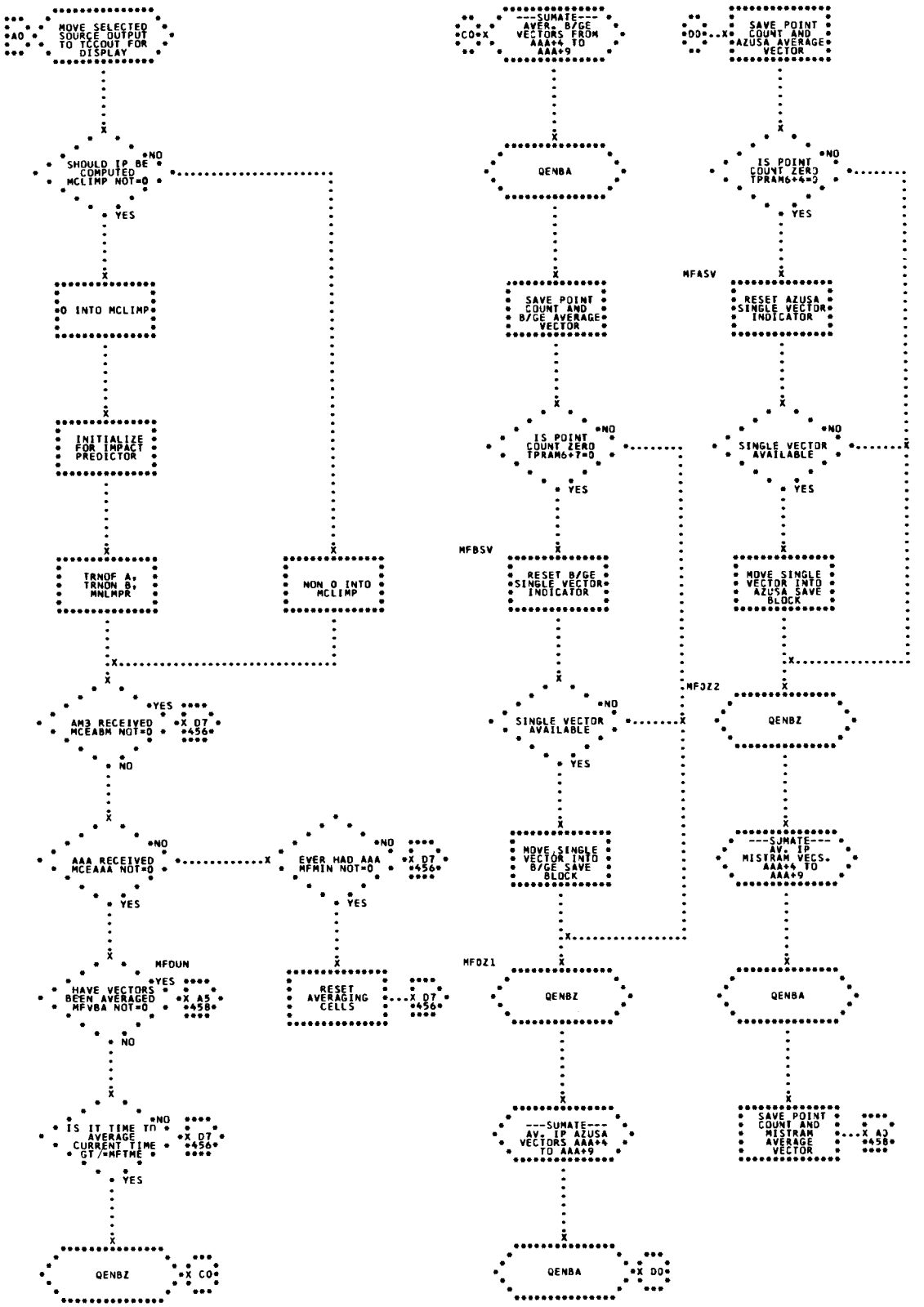
C) THE LAUNCH ROUTINES ARE ALL SUPPRESSED AND ALL NECESSARY INPUT AND INDICATORS ARE SET FOR THE ORBIT PROGRAMS.

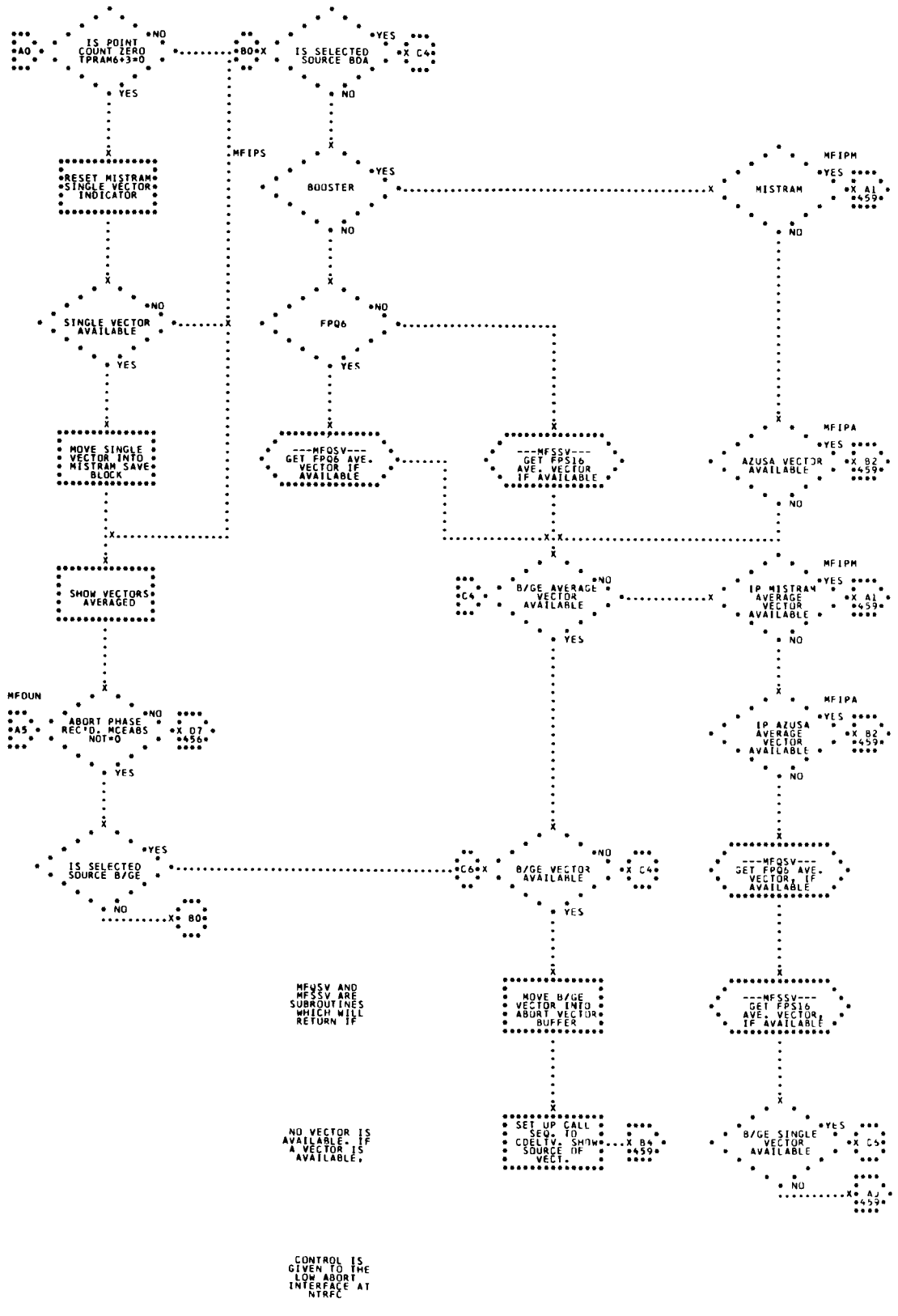
D) THE ORBIT DEFINING VECTOR IS PRINTED ON-LINE AND ORBIT PHASE IS INDICATED EXTERNALLY.

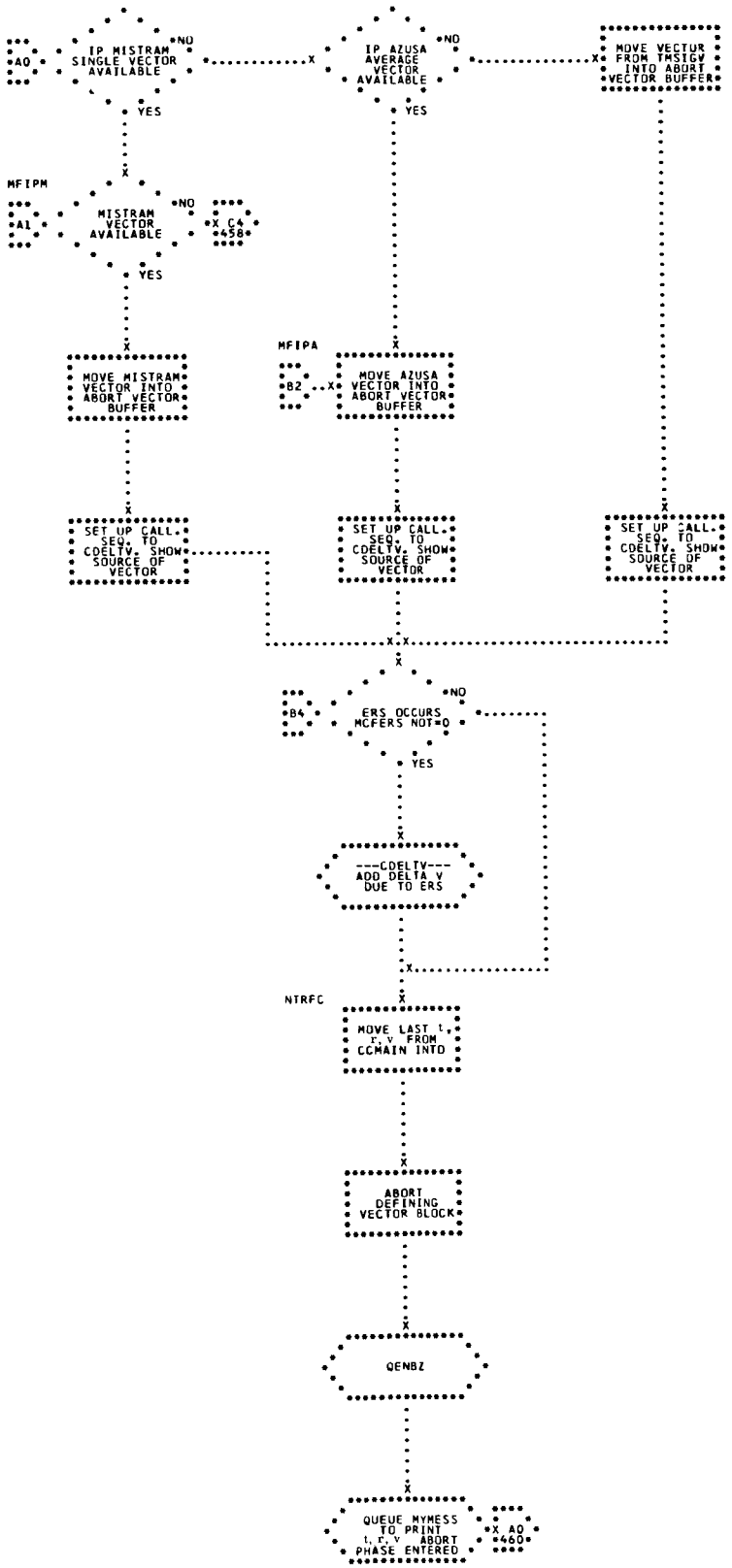
E) THE INFORMATION NECESSARY FOR WRITING ON THE RESTART TAPE IS SET UP.

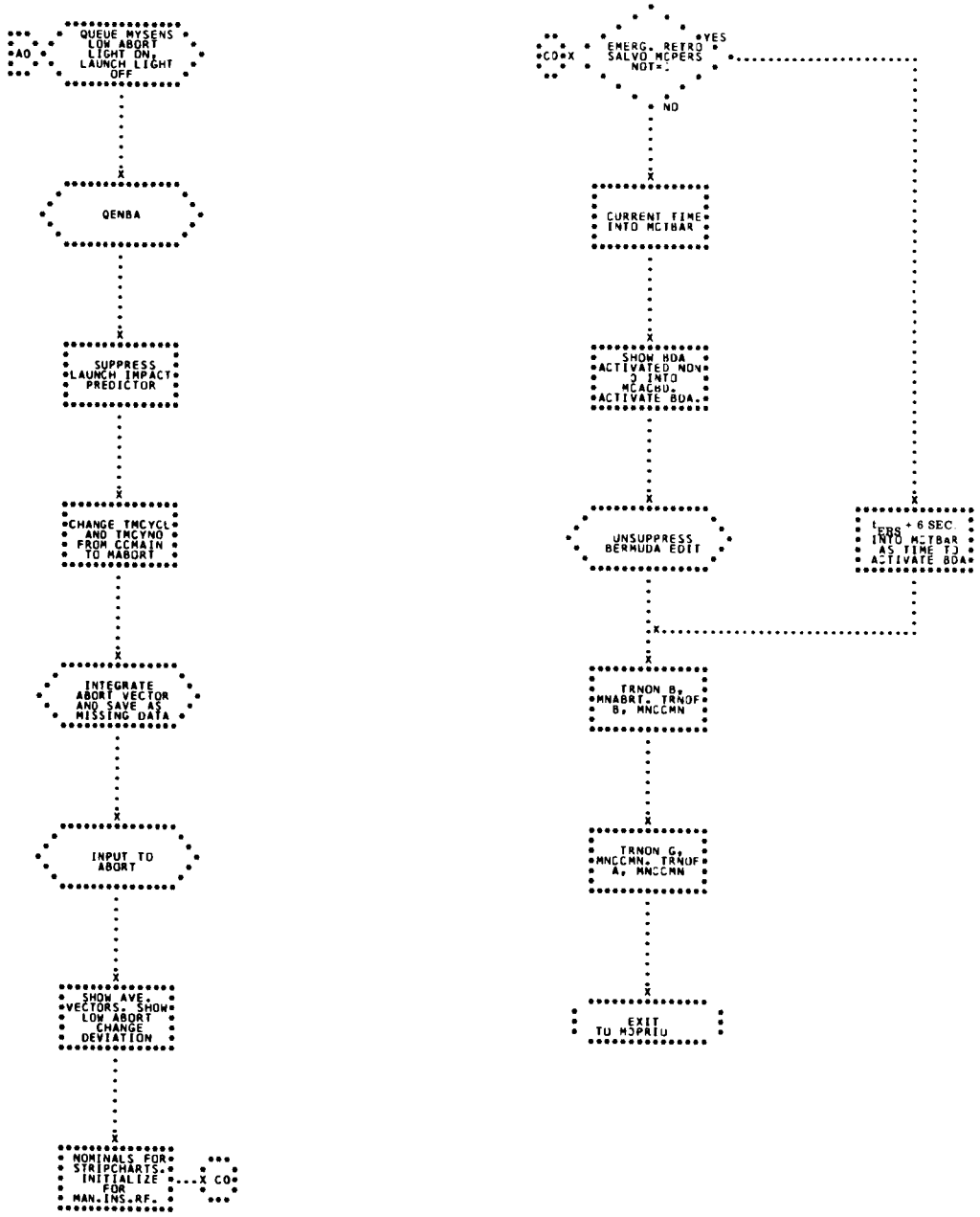
F) IN THE EVENT OF A RESTART, THE ORBIT INTERFACE PROGRAM IS ENTERED DIRECTLY FROM MPCCMN. THE PROGRAM THEN UNSUPPRESSES MYSRST SO THAT A VECTOR MAY BE READ FROM THE RESTART TAPE. IT THEN WAITS UNTIL EITHER THE VECTOR HAS BEEN READ IN OR UNTIL IT IS TOLD THAT NO VECTOR IS FORTHCOMING. IF NO VECTOR IS AVAILABLE ON THE RESTART TAPE, THE INTERFACE PROGRAM WILL WAIT FOR THE MANUAL INSERTION OF AN ORBIT DEFINING VECTOR. IN EITHER CASE, ONCE A VECTOR HAS BEEN FOUND, THE PROCESSING CONTINUES AS DURING A NORMAL LAUNCH-ORBIT INTERFACE.

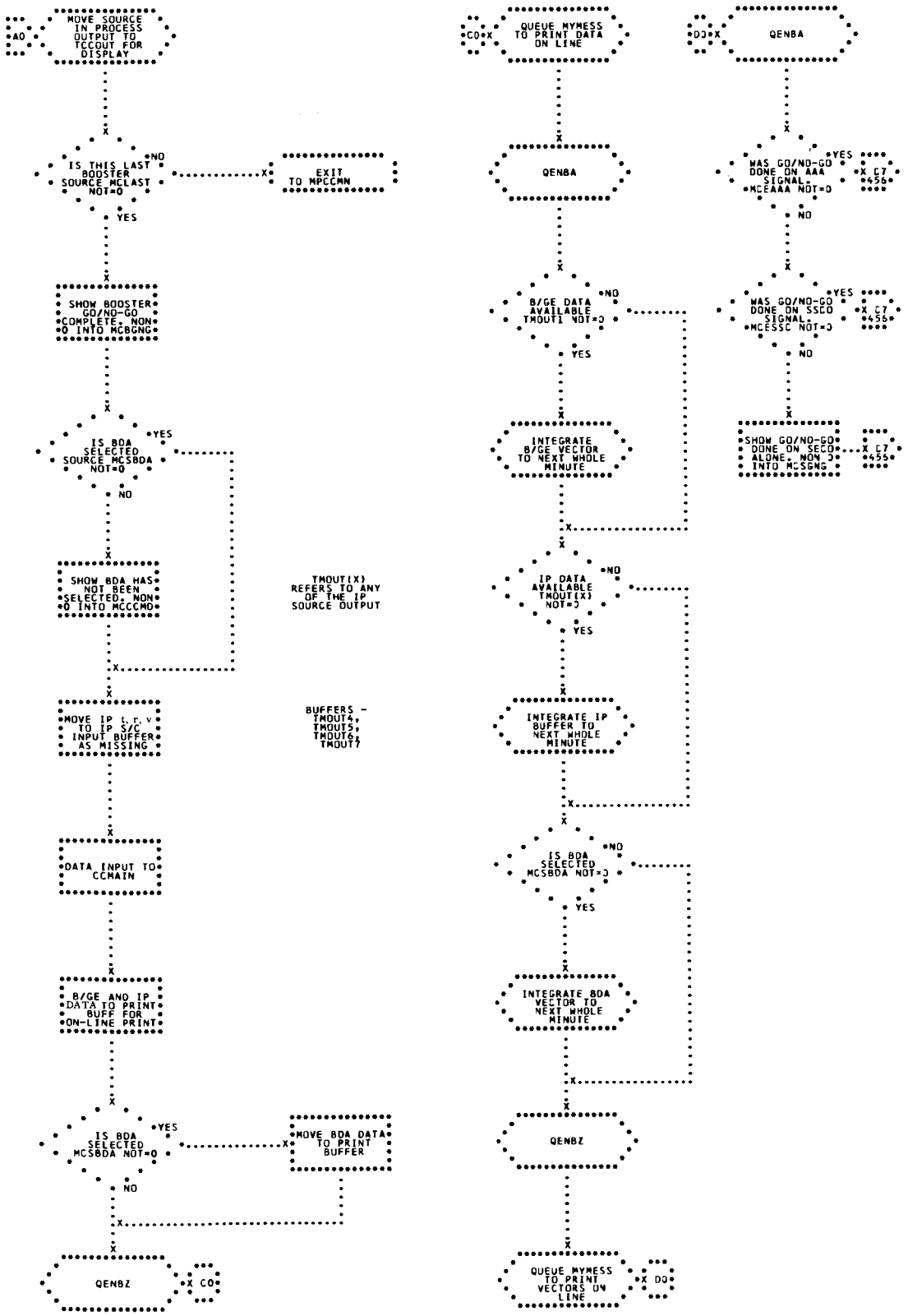

ENTER MFCCMN
AVERAGE VECTOR MCCFHP NOT=0
*NO
*X A0 *457*
YES
BOOSTER GO/NO-GO COMPLETE *MCBGNG NOT=0
*NO
*X A0 *461*
YES
HAS BDA EVER BEEN SELCTED MCCCMD=0
*YES
NO
IS BDA SELECTED MCSBDA NOT=0
*NO
YES
0 INTO MCCCMD
MOVE SOURCE IN PROCESS OUTPUT TO TCCOUT FOR DISPLAY
IS THIS A DISPLAY PASS MCDISP NOT=0
*NO
X B0
YES
EXIT TO MPCCMN
*B0*X
IS BDA THE SOURCE IN PROCESS MCSINP=3 *
*NO
YES
IS BDA AVAILABLE TMOUT3 NOT=0
*NO
X C7
YES
INTEGRATE BDA VECTOR TO NEXT WHOLE MINUTE
TIME TO PRINT RECOMMEND. DIDBD=/LT MCHFSC
*NO
X C7
YES
MCHFSC + 120 INTO DIDBD
QUEUE MYMESS TO PRINT BDA VALUES ON-LINE
X C7
IS IP SOURCE IN PROCESS MCSINP=6 OR 7
*NO
YES
IS IP DATA AVAILABLE TMLIPC NOT=0
*NO
YES
---CUPDAT--- INTEGRATE IP VECTOR TO NEXT WHOLE MINUTE
TIME TO PRINT IP RECOMMEND. DIDIP=/LT MCHFSC
*NO
YES
MCHFSC + 120 INTO DIDIP
SET UP PRINT BLOCK FOR ON-LINE PRINTING
QUEUE MYMESS TO PRINT IP VALUES ON LINE
*D6*X
MANUAL INSERTION ABORT SWITCH MCMADS=2
*YES
*X A1 *463*
NO
TRNOF A, MMCCMN
EXIT TO MOPRIO
C7
HAS ORBIT SWITCH BEEN THROWN *MCCORP NOT=0
*YES
*X A0 *462*
NO
*D7*X
HAS ABORT SWITCH BEEN THROWN MCEABS NOT=0
*YES
*X A0 *463*
NO
MANUAL INSERTION ORBIT SWITCH MCMADS=1
*YES
*X A0 *462*
NO
D6

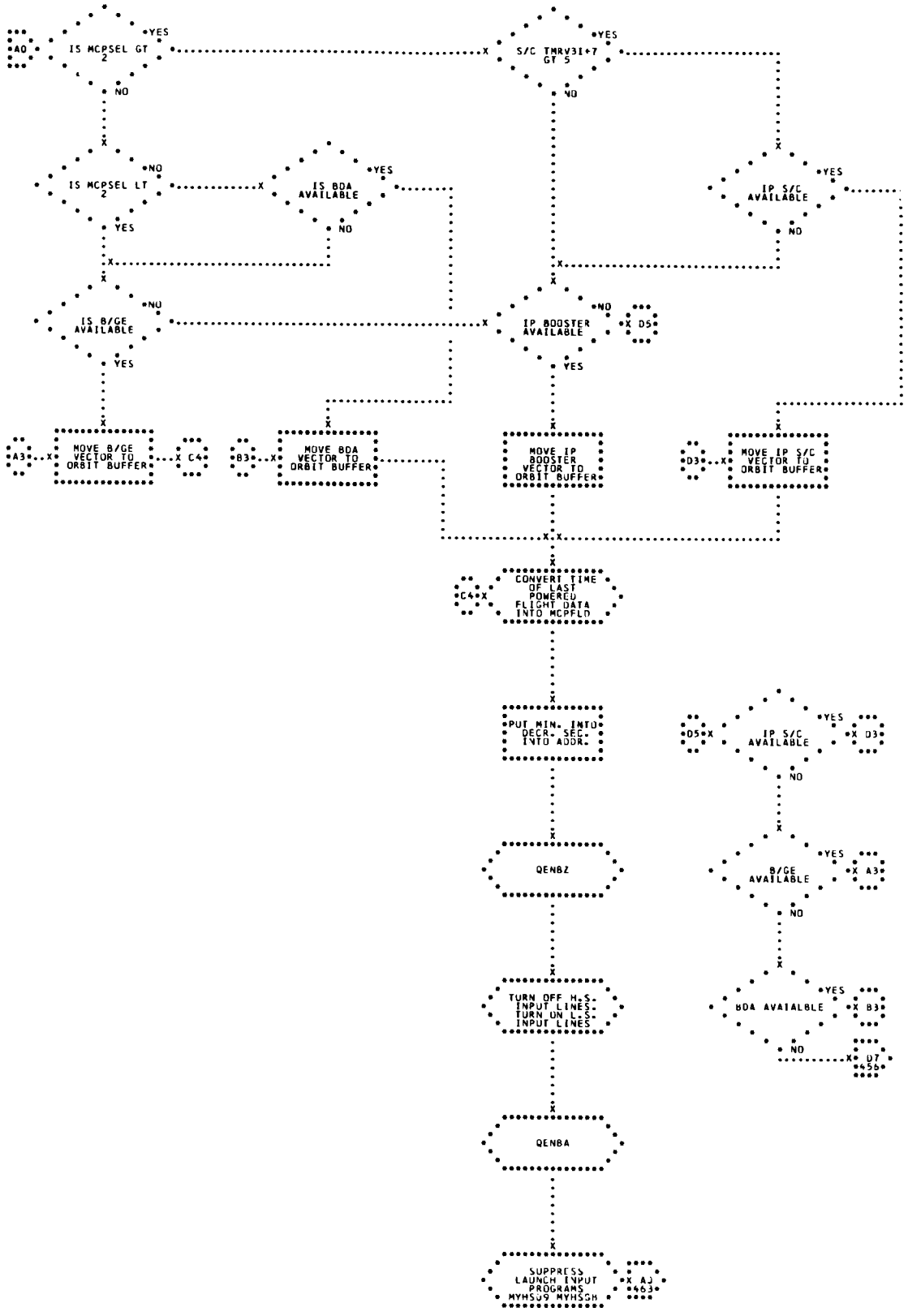

A0
MOVE SELECTED SOURCE OUTPUT TO TCOUT FOR DISPLAY
SHOULD IP BE COMPUTED MCLIMP NOT=0
NO
YES
0 INTO MCLIMP
INITIALIZE FOR IMPACT PREDICTOR
TRNOF A, TRNON B, MNLMPR
NON 0 INTO MCLIMP
AM3 RECEIVED MCEABM NOT=0
YES
X D7 456
NO
AAA RECEIVED MCEAAA NOT=0
NO
EVER HAD AAA MFMIN NOT=0
NO
X D7 456
YES
YES
MFDUN
HAVE VECTORS BEEN AVERAGED MFVBA NOT=0
YES
X A5 458
RESET AVERAGING CELLS
X D7 456
NO
IS IT TIME TO AVERAGE CURRENT TIME GT/=MFTME
NO
X D7 456
YES
QENBZ
X C0
C0
---SUMATE--- AVER. B/GE VECTORS FROM AAA+4 TO AAA+9
QENBA
SAVE POINT COUNT AND B/GE AVERAGE VECTOR
IS POINT COUNT ZERO TPRAM6+7=0
NO
YES
MFBSV
RESET B/GE SINGLE VECTOR INDICATOR
SINGLE VECTOR AVAILABLE
NO
YES
MOVE SINGLE VECTOR INTO B/GE SAVE BLOCK
MFDZ1
QENBZ
---SUMATE--- AV. IP AZUSA VECTORS AAA+4 TO AAA+9
QENBA
X D0
D0
SAVE POINT COUNT AND AZUSA AVERAGE VECTOR
IS POINT COUNT ZERO TPRAM6+6=0
NO
YES
MFASV
RESET AZUSA SINGLE VECTOR INDICATOR
SINGLE VECTOR AVAILABLE
NO
YES
MOVE SINGLE VECTOR INTO AZUSA SAVE BLOCK
MFDZ2
QENBZ
---SUMATE--- AV. IP MISTRAM VECS. AAA+4 TO AAA+9
QENBA
SAVE POINT COUNT AND MISTRAM AVERAGE VECTOR
X A2 458

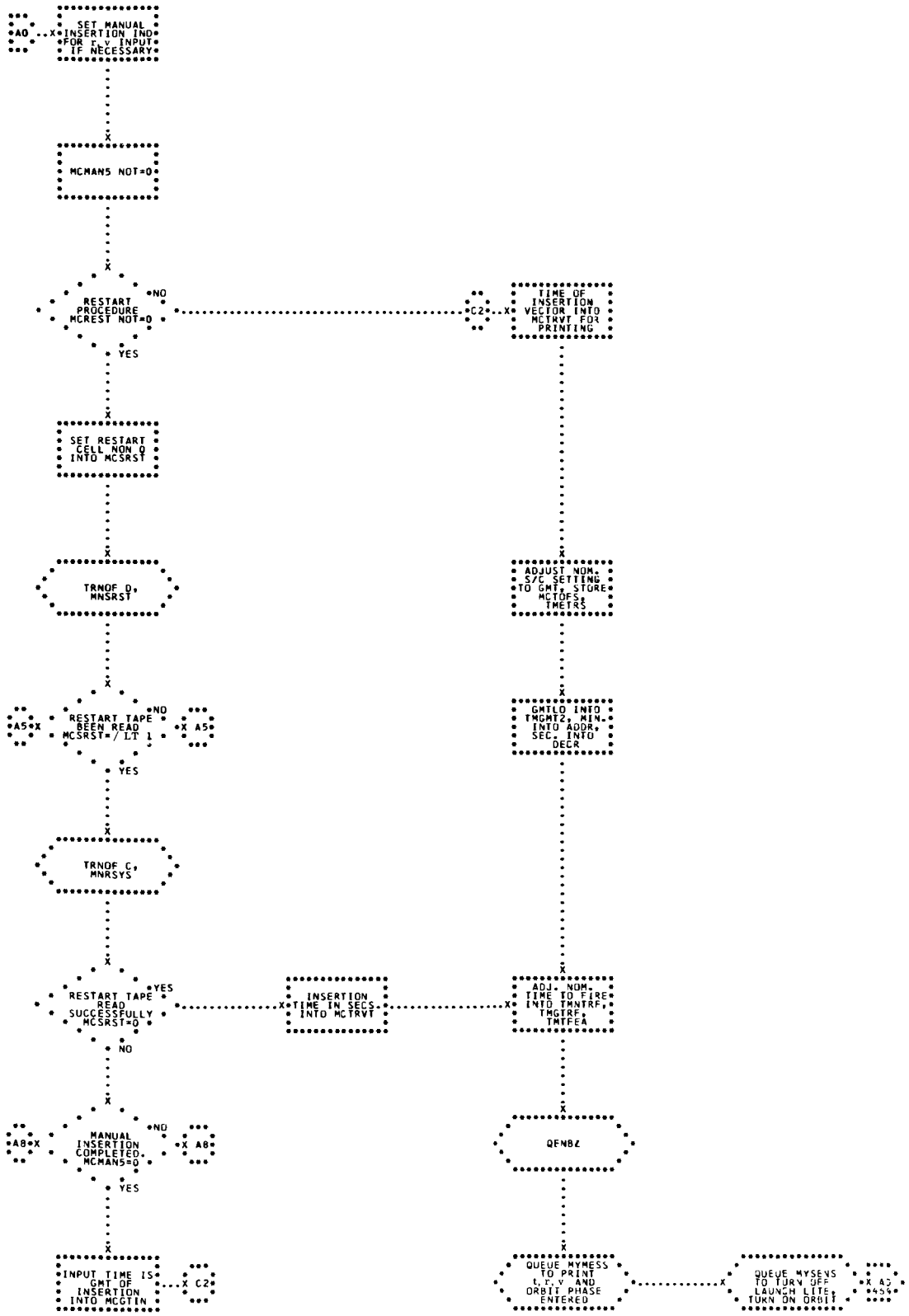

A0
IS POINT COUNT ZERO TPRAM6+3=0
NO
YES
B0
IS SELECTED SOURCE B0A
YES
C4
NO
MFIPS
RESET MISTRAM SINGLE VECTOR INDICATOR
BOOSTER
YES
NO
MISTRAM
MFIPM
YES
A1
4459
NO
SINGLE VECTOR AVAILABLE
NO
YES
FPQ6
NO
YES
MOVE SINGLE VECTOR INTO MISTRAM SAVE BLOCK
---MFQSV--- GET FPQ6 AVE. VECTOR IF AVAILABLE
---MFSSV--- GET FPS16 AVE. VECTOR IF AVAILABLE
AZUSA VECTOR AVAILABLE
MFIPA
YES
B2
4459
NO
SHOW VECTORS AVERAGED
C4
B/GE AVERAGE VECTOR AVAILABLE
NO
YES
IP MISTRAM AVERAGE VECTOR AVAILABLE
MFIPM
YES
A1
4459
NO
MFDUN
A5
ABORT PHASE REC'D. MCEABS NOT=0
NO
YES
D7
4456
IP AZUSA AVERAGE VECTOR AVAILABLE
MFIPA
YES
B2
4459
NO
IS SELECTED SOURCE B/GE
YES
NO
B0
C6
B/GE VECTOR AVAILABLE
NO
YES
C4
---MFQSV--- GET FPQ6 AVE. VECTOR, IF AVAILABLE
MFQSV AND MFSSV ARE SUBROUTINES WHICH WILL RETURN IF
MOVE B/GE VECTOR INTO ABORT VECTOR BUFFER
---MFSSV--- GET FPS16 AVE. VECTOR, IF AVAILABLE
NO VECTOR IS AVAILABLE. IF A VECTOR IS AVAILABLE,
SET UP CALL SEQ. TO CDELTV. SHOW SOURCE OF VECT.
B4
4459
B/GE SINGLE VECTOR AVAILABLE
YES
C5
NO
A1
4459
CONTROL IS GIVEN TO THE LOW ABORT INTERFACE AT NTRFC

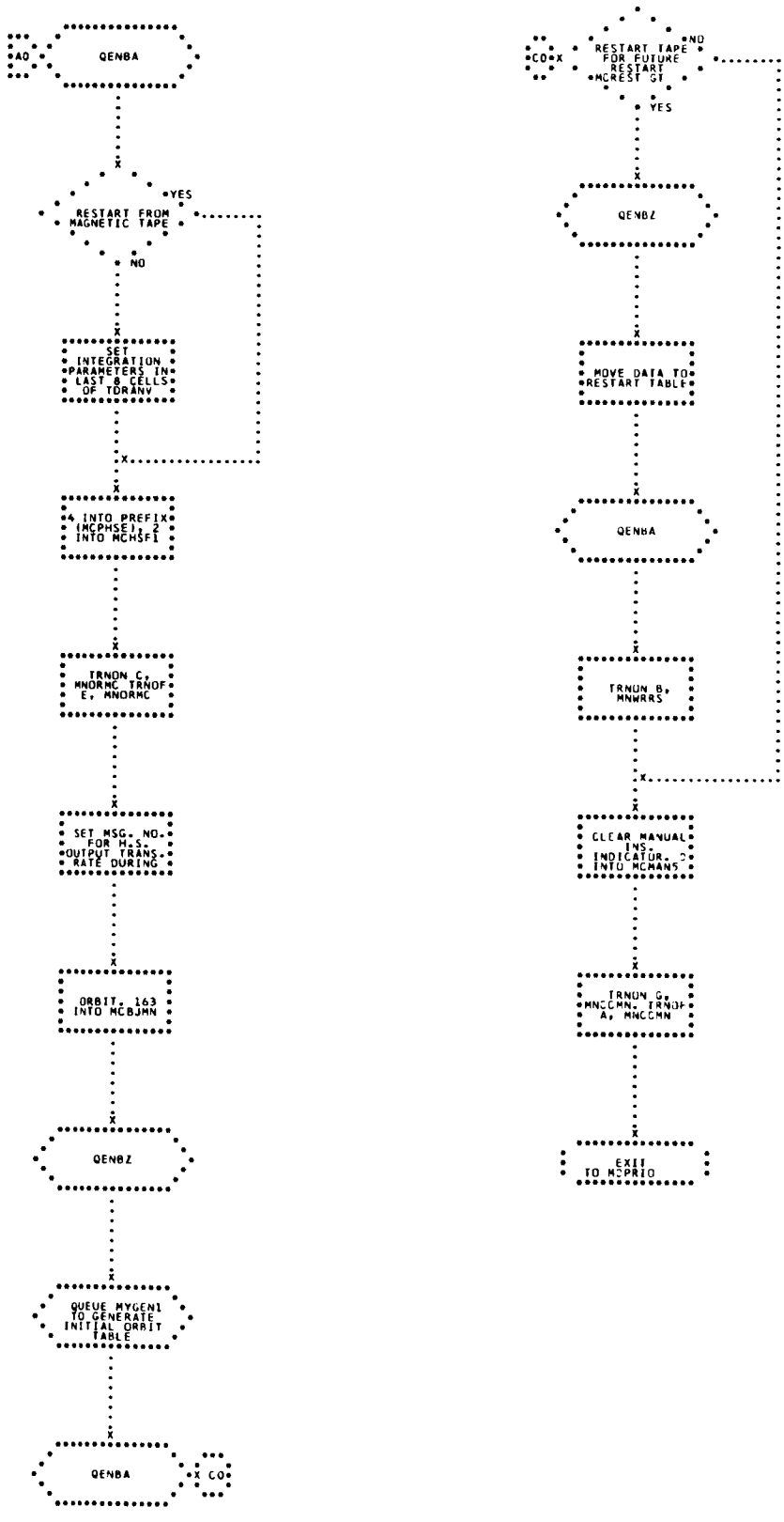

IP MISTRAM SINGLE VECTOR AVAILABLE
NO
YES
A0
IP AZUSA AVERAGE VECTOR AVAILABLE
NO
YES
MOVE VECTOR FROM TMSIGV INTO ABORT VECTOR BUFFER
MFIPM
MISTRAM VECTOR AVAILABLE
NO
YES
A1
X C4 458
MFIPA
B2
MOVE MISTRAM VECTOR INTO ABORT VECTOR BUFFER
MOVE AZUSA VECTOR INTO ABORT VECTOR BUFFER
SET UP CALL. SEQ. TO CDELTV, SHOW SOURCE OF VECTOR
SET UP CALL. SEQ. TO CDELTV, SHOW SOURCE OF VECTOR
SET UP CALL. SEQ. TO CDELTV, SHOW SOURCE OF VECTOR
B4
ERS OCCURS MCFERS NOT=0
NO
YES
---CDELTV--- ADD DELTA V DUE TO ERS
NTRFC
MOVE LAST t, r, v FROM CCMAIN INTO
ABORT DEFINING VECTOR BLOCK
QENBZ
QUEUE MYMESS TO PRINT t, r, v ABORT PHASE ENTERED
X A0 460

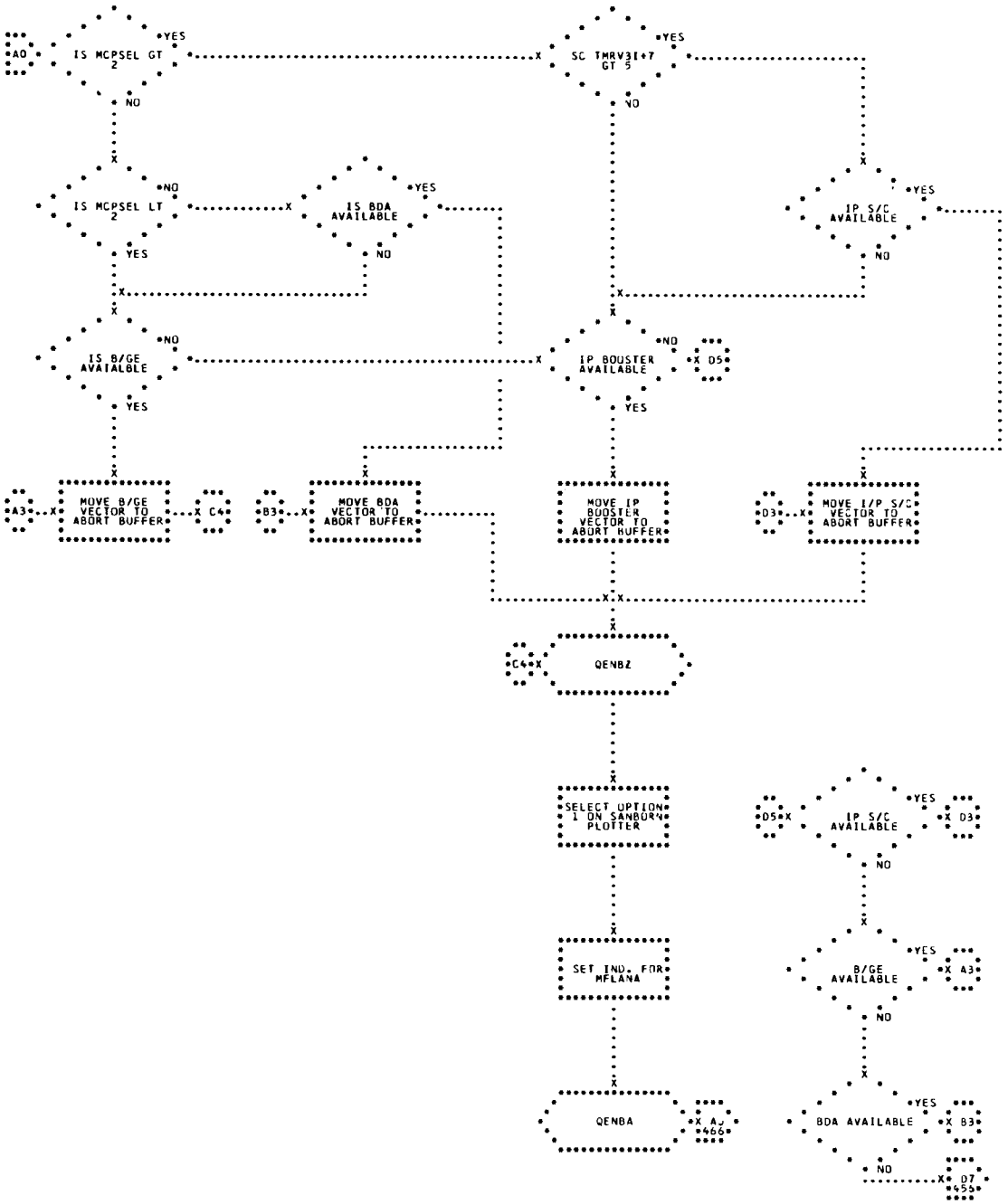

QUEUE MYSENS LOW ABORT LIGHT ON, LAUNCH LIGHT OFF
AO
QENBA
SUPPRESS LAUNCH IMPACT PREDICTOR
CHANGE TMCYCL AND TMCYNO FROM CCMAIN TO MABORT
INTEGRATE ABORT VECTOR AND SAVE AS MISSING DATA
INPUT TO ABORT
SHOW AVE. VECTORS. SHOW LOW ABORT CHANGE DEVIATION
NOMINALS FOR STRIPCHARTS. INITIALIZE FOR MAN.INS.RF.
CO
EMERG. RETRO SALVO MCPERS NOT=
YES
NO
CURRENT TIME INTO MCTBAR
SHOW HDA ACTIVATED NOW 2 INTO MCACHD. ACTIVATE BDA.
UNSUPPRESS BERMUDA EDIT
TEBS + 6 SEC. INTO MCTBAR AS TIME TO ACTIVATE BDA
TRNON B, MNABRT, TRNOF B, MNCCMN
TRNON G, MNCCMN, TRNOF A, MNCCMN
EXIT TO MJPRIO

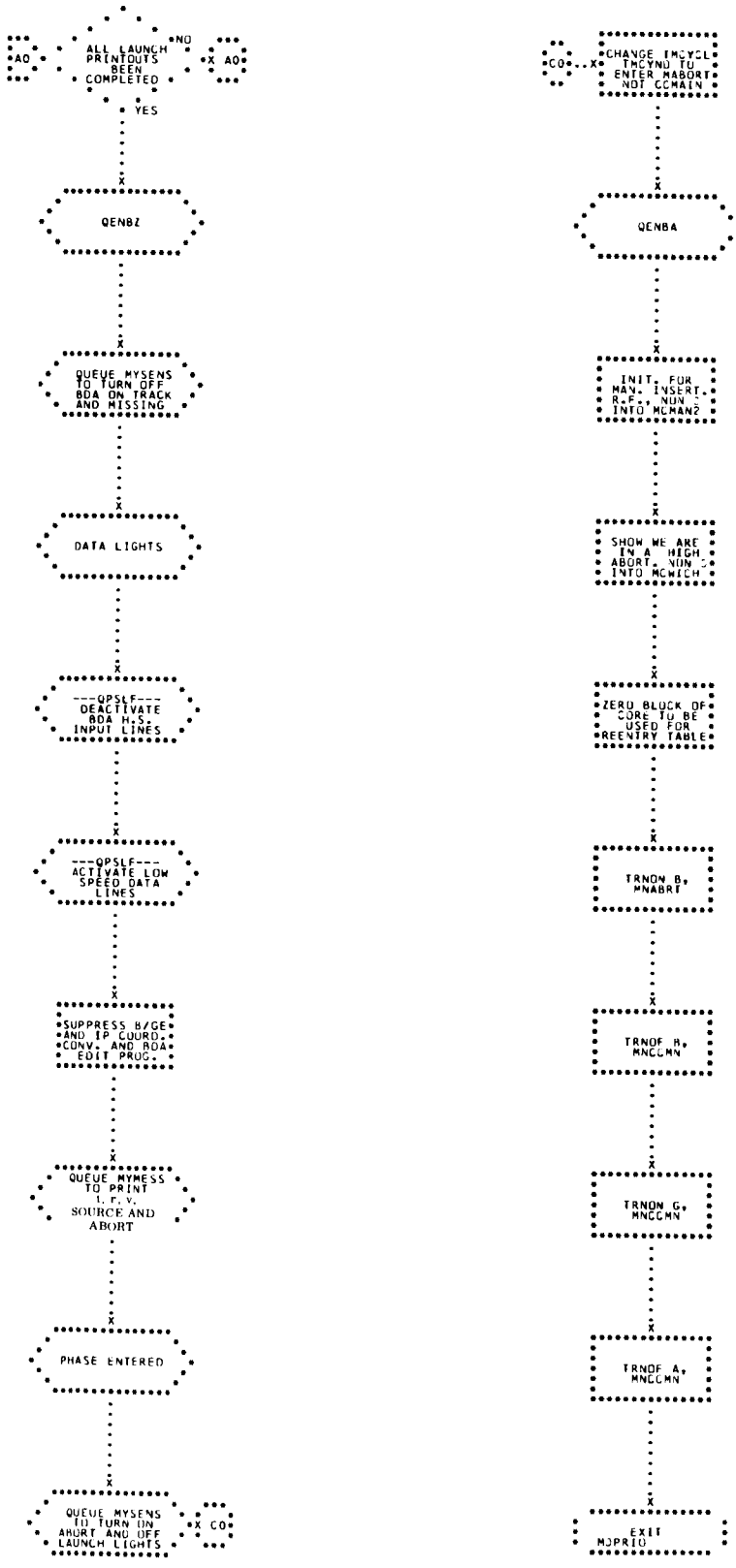

AO
MOVE SOURCE IN PROCESS OUTPUT TO TCCOUT FOR DISPLAY
IS THIS LAST BOOSTER SOURCE MCLAST NOT=0
NO
YES
EXIT TO MPCCMN
SHOW BOOSTER GO/NO-GO COMPLETE, NON-0 INTO MCBGNG
IS BDA SELECTED SOURCE MCSBDA NOT=0
YES
NO
SHOW BDA HAS NOT BEEN SELECTED, NON-0 INTO MCCCMD
TMOUT(X) REFERS TO ANY OF THE IP SOURCE OUTPUT
MOVE IP t, r, v TO IP S/C INPUT BUFFER AS MISSING
BUFFERS - TMOUT4, TMOUT5, TMOUT6, TMOUT7
DATA INPUT TO CCMAIN
B/GE AND IP DATA TO PRINT BUFF FOR ON-LINE PRINT
IS BDA SELECTED MCSBDA NOT=0
YES
NO
MOVE BDA DATA TO PRINT BUFFER
QENBZ
X CO
CO X
QUEUE MYMESS TO PRINT DATA ON LINE
QENBA
B/GE DATA AVAILABLE TMOUT1 NOT=0
NO
YES
INTEGRATE B/GE VECTOR TO NEXT WHOLE MINUTE
IP DATA AVAILABLE TMOUT(X) NOT=0
NO
YES
INTEGRATE IP BUFFER TO NEXT WHOLE MINUTE
IS BDA SELECTED MCSBDA NOT=0
NO
YES
INTEGRATE BDA VECTOR TO NEXT WHOLE MINUTE
QENBZ
QUEUE MYMESS TO PRINT VECTORS ON LINE
X DQ
DQ X
QENBA
WAS GO/NO-GO DONE ON AAA SIGNAL, MCEAAA NOT=0
YES
X C7 456
NO
WAS GO/NO-GO DONE ON SSCO SIGNAL, MCESSC NOT=0
YES
X C7 456
NO
SHOW GO/NO-GO DONE ON SSCO ALONE, NON-0 INTO MCSGNG
X C7 456


A0
IS MCPSEL GT 2
YES
NO
S/C TMRV3I+7 GT 5
YES
NO
IS MCPSEL LT 2
NO
YES
IS BDA AVAILABLE
YES
NO
IP S/C AVAILABLE
YES
NO
IS B/GE AVAILABLE
NO
YES
IP BOOSTER AVAILABLE
NO
D5
YES
A3
MOVE B/GE VECTOR TO ORBIT BUFFER
C4
B3
MOVE BDA VECTOR TO ORBIT BUFFER
MOVE IP BOOSTER VECTOR TO ORBIT BUFFER
D3
MOVE IP S/C VECTOR TO ORBIT BUFFER
C4
CONVERT TIME OF LAST POWERED FLIGHT DATA INTO MCPFLD
PUT MIN. INTO DECR. SEC. INTO ADDR.
QENBZ
TURN OFF H.S. INPUT LINES. TURN ON L.S. INPUT LINES
QENBA
SUPPRESS LAUNCH INPUT PROGRAMS MYHSU9 MYHSGH
AJ
463
D5
IP S/C AVAILABLE
YES
D3
NO
B/GE AVAILABLE
YES
A3
NO
BDA AVAILABLE
YES
B3
NO
D7
456


A0
SET MANUAL INSERTION IND FOR F.C. INPUT IF NECESSARY
MCMAN5 NOT=0
RESTART PROCEDURE MCREST NOT=0
NO
YES
C2
TIME OF INSERTION VECTOR INTO MCTRVT FOR PRINTING
SET RESTART CELL NON 0 INTO MCSRST
TRNOF D, MNSRST
ADJUST NOM. S/C SETTING TO GMT, STORE MCTOFS, TMEFS
A5
RESTART TAPE BEEN READ MCSRST=/LT 1
A5
GMTLO INTO TMGMT2, MIN. INTO ADDR, SEC. INTO DECR
TRNOF C, MNRSYS
RESTART TAPE READ SUCCESSFULLY MCSRST=0
INSERTION TIME IN SECS. INTO MCTRVT
ADJ. NOM. TIME TO FIRE INTO TMNTRF, TMGTRF, TMTFEA
A8
MANUAL INSERTION COMPLETED. MCMAN5=0
A8
QFNBZ
INPUT TIME IS GMT OF INSERTION INTO MCGTIN
C2
QUEUE MYMESS TO PRINT L.P.V AND ORBIT PHASE ENTERED
QUEUE MYSENS TO TURN OFF LAUNCH LITE, TURN ON ORBIT
A3


AO
QENBA
RESTART FROM MAGNETIC TAPE
YES
NO
SET INTEGRATION PARAMETERS IN LAST 8 CELLS OF TDRANV
4 INTO PREFIX (MCPHSE), 2 INTO MCHSFI
TRNON C, MNORMC TRNOF E, MNORMC
SET MSG. NO. FOR M.S. OUTPUT TRANS. RATE DURING
ORBIT, 163 INTO MCBJMN
QENBZ
QUEUE MYGEN1 TO GENERATE INITIAL ORBIT TABLE
QENBA
CO
CO
RESTART TAPE FOR FUTURE RESTART MCREST GT
NO
YES
QENBZ
MOVE DATA TO RESTART TABLE
QENBA
TRNON B, MNWRRS
CLEAR MANUAL INS. INDICATOR, 2 INTO MCMANS
TRNON G, MNCOMN, TRNOF A, MNCOMN
EXIT TO MCPRIO


A0
IS MCPSEL GT 2
YES
NO
SC TMRV3I+7 GT 5
YES
NO
IS MCPSEL LT 2
NO
YES
IS BDA AVAILABLE
YES
NO
IP S/C AVAILABLE
YES
NO
IS B/GE AVAILABLE
NO
YES
IP BOOSTER AVAILABLE
NO
YES
D5
A3
MOVE B/GE VECTOR TO ABORT BUFFER
C4
B3
MOVE BDA VECTOR TO ABORT BUFFER
MOVE IP BOOSTER VECTOR TO ABORT BUFFER
D3
MOVE I/P S/C VECTOR TO ABORT BUFFER
C4
QENBZ
SELECT OPTION 1 ON SANBORN PLOTTER
SET IND. FOR MFLANA
QENBA
A/466
D5
IP S/C AVAILABLE
YES
NO
D3
B/GE AVAILABLE
YES
NO
A3
BDA AVAILABLE
YES
NO
B3
D7 455


ALL LAUNCH PRINTOUTS BEEN COMPLETED
NO
AO
AO
YES
QENBZ
QUEUE MYSENS TO TURN OFF BOA ON TRACK AND MISSING
DATA LIGHTS
---QPSLE--- DEACTIVATE BOA H.S. INPUT LINES
---QPSLE--- ACTIVATE LOW SPEED DATA LINES
SUPPRESS B/GE AND IP COORD., CONV. AND BOA EDIT PROG.
QUEUE MYMESS TO PRINT L. F. V. SOURCE AND ABORT
PHASE ENTERED
QUEUE MYSENS TO TURN ON ABORT AND OFF LAUNCH LIGHTS
CO
CO
CHANGE TMCYCL TMCYND TO ENTER MABORT NOT CCMAIN
QENBA
INIT. FOR MAN. INSERT. R.F., NUN 3 INTO MCMAN2
SHOW WE ARE IN A HIGH ABORT, NUN 3 INTO MCMICH
ZERO BLOCK OF CORE TO BE USED FOR REENTRY TABLE
TRNON B, MNABRT
TRNOF B, MNCOMN
TRNON G, MNCOMN
TRNOF A, MNCOMN
EXIT MJPRIO

53. CORKNI - RUNGE-KUTTA NUMERICAL INTEGRATION

A FOURTH ORDER RUNGE-KUTTA NUMERICAL INTEGRATION PROGRAM IS USED DURING LAUNCH AND ABORT TO PREDICT IMPACT POINTS, UPDATE VECTORS, AND GENERATE ORBIT AND REENTRY TABLES. (SEE FLOWCHART)

RUNGE-KUTTA INTEGRATION IS A SINGLE-STEP INTEGRATION METHOD. IS WAS CHOSEN INSTEAD OF A MULTI-STEP METHOD BECAUSE OF THE GREAT NEED FOR SPEED OF EXECUTION. A SINGLE-STEP METHOD OFFERS TWO BASIC ADVANTAGES -

A) THE FUNCTIONS USED TO DESCRIBE THE ATMOSPHERE ARE NOT CONTINUOUS. WITHIN A GIVEN INTEGRATION STEP, A SINGLE-STEP METHOD CONSIDERS A MUCH SMALLER PORTION OF THE ATMOSPHERE THAN A MULTI-STEP METHOD USING THE SAME STEP SIZE. THEREFORE, A SINGLE STEP METHOD CAN USE MUCH LARGER STEP SIZES BEFORE DISCREPANCIES DUE TO THE DISCONTINUITIES BECOME SIGNIFICANT.

B) A MULTI-STEP METHOD REQUIRES A LENGTHY RESTART PROCEDURE WHEN CHANGING STEP SIZE. A SINGLE-STEP METHOD REQUIRES NONE.

IN AN ATTEMPT TO DEVELOP A FAST, SINGLE-STEP INTEGRATOR FOR IMPACT PREDICTION, RUNGE-KUTTA METHODS OF SECOND AND MODIFIED SECOND ORDER WERE TRIED. THEY WERE FOUND TO BE FAIRLY ACCURATE AT STEP SIZES ON THE ORDER OF EIGHT SECONDS, BUT WERE TOO WEAK AT LARGER STEP SIZES. THE FOURTH ORDER METHOD WAS THEREFORE USED. AFTER MUCH ANALYSIS, OPTIMUM STEP SIZES FOR VARIOUS SECTIONS OF REENTRY WERE FOUND. THESE STEP SIZES ARE OPTIMUM FROM THE STANDPOINT OF ACCURACY IN IMPACT PREDICTION AND FROM THE

STANDPOINT OF MACHINE TIME.

THERE ARE FOUR TYPES OF INPUT/OUTPUT PACKAGES FOR THIS INTEGRATION PROGRAM AND EACH HAS ITS OWN CONTROL LOGIC. ALL FOUR PROGRAMS, HOWEVER, USE THE SAME LOGIC FOR THE BASIC MATHEMATICS OF NUMERICAL INTEGRATION.

TYPE 1 INTEGRATION IS USED DURING REENTRY TO PREDICT THE TIME, LONGITUDE, AND LATITUDE OF IMPACT. GENERALLY, TYPE 2 INTEGRATION IS USED TO UPDATE VECTORS. TYPE 3 INTEGRATION GENERATES EITHER A REENTRY OR AN ORBIT TABLE OF $\bar{r}$ AND $\bar{v}$ VECTORS AND A TABLE OF CORRESPONDING HEIGHTS. TYPE 4 INTEGRATION GENERATES ONLY A TABLE OF $\bar{r}$ AND $\bar{v}$ VECTORS. EACH OF FOUR TYPES OF INTEGRATION WILL BE TREATED SEPARATELY IN THE FOLLOWING DISCUSSION.

INPUT

A) SUBROUTINES C6HPER AND C4DRFT ARE USED. IN ADDITION, TYPE 1 INTEGRATION USES C3GSCP.

B) LIBRARY SUBROUTINE U1SICO IS USED.

C) CONSTANTS -

KMD1A7	PZE 7,0,1
KM00.5	FLOATING POINT MINUS FIVE TENTHS
K0C0.5	FLOATING POINT FIVE TENTHS
K00.25	FLOATING POINT TWENTY-FIVE HUNDREDTHS
K0.166	FLOATING POINT ONE SIXTH
K002.0	FLOATING POINT TWO
K00000	ZERO
K00001	FIXED POINT ONE
K0C007	FIXED POINT SEVEN
K00.UT	FLOATING POINT GEMINI TIME UNIT IN SECONDS

D) TABLES -

TC.WCF	CONVERSION CONSTANT IN FLOATING POINT

WHICH WHEN DIVIDED BY THE MASS OF THE SPACECRAFT IN POUNDS EQUALS THE AREA TO MASS RATIO OF THE GEMINI SPACECRAFT IN GEMINI UNITS.

INPUT/OUTPUT REQUIREMENTS FOR TYPE 1 INTEGRATION

TYPE 1 INTEGRATION IS USED DURING REENTRY TO PREDICT THE TIME, LONGITUDE, AND LATITUDE OF IMPACT. THE IMPACT HEIGHT IS SPECIFIED IN THE CALLING SEQUENCE. IN PREDICTING IMPACT, TYPE 1 INTEGRATION MAKES USE OF TWO SETS OF STEP SIZES. THE SET OF SMALLER STEP SIZES IS USED WHEN TYPE 1 IS ENTERED WITH A VECTOR BELOW 350,000 FEET. THE SET OF LARGER STEP SIZES IS USED FOR THOSE VECTORS ABOVE 350,000 FEET. THE ECCENTRICITY OF THE TRAJECTORY NECESSITATES HAVING THESE TWO BLOCKS OF STEP SIZES.

VECTORS ARE INTEGRATED TO IMPACT USING THE FOLLOWING STEP SIZES(E_i) BETWEEN THE VARIOUS GIVEN HEIGHTS (C_i) ALONG THE REENTRY TRAJECTORY.

ENTRY BELOW 350,000 FEET -

H > 350,000 FEET (C_4) -45 SECONDS (F_4)
H > 2000,000 FEET(C_3) -30 SECONDS (F_3)
H > 100,000 FEET (C_2) -15 SECONDS (F_2)
H $\geq$ IMPACT HEIGHT (C_1) -5 SECONDS (F_1)

ENTRY ABOVE 350,000 FEET -

H > 350,000 FEET (C_4) -120 SECONDS (F_4)
H > 200,000 FEET (C_3) -60 SECONDS (F_3)
H > 100,000 FEET (C_2) -28 SECONDS (F_2)
H $\geq$ IMPACT HEIGHT (C_1) -15 SECONDS (F_1)

THE STEP SIZES ARE USED AND CHANGED BASICALLY IN THE TWO FOLLOWING WAYS -

A) FOR A TRAJECTORY WHERE H IS INCREASING, THE STEP SIZE (F_1) IS CHANGED (INCREASED) IF THE RESULTING VECTOR OF AN INTEGRATION STEP HAS A HEIGHT

GREATER THAN C_{i+1} THIS VECTOR ABOVE C_{i+1} IS THEN INTEGRATED USING STEP SIZE F_{i+1}

B) FOR A TRAJECTORY WHERE H IS DECREASING, THE STEP SIZE F_i IS CHANGED (DECREASED) IF THE RESULTING VECTOR OF AN INTEGRATION STEP HAS A HEIGHT LESS THAN OR EQUAL TO C_{i-1}. HOWEVER, IN THIS CASE THE VECTOR USED AT THE BEGINNING OF THIS STEP IS INTEGRATED AGAIN USING STEP SIZE F_{i-1}. (A COUNTER M=4 IS SET AT THIS POINT TO SHOW THAT STEP SIZE HAS BEEN CHANGED, AND PROVIDES THE POSSIBILITY THAT STEP SIZE F_{i-1} MAY BE USED FOUR TIMES ABOVE C_{i-1}. THIS ALSO PREVENTS AN IMMEDIATE CHANGE BACK TO STEP SIZE F_i ON THE NEXT STEP.) THUS, THE RESULT OF AN INTEGRATION STEP IS REJECTED IF THE RESULTING HEIGHT IS BELOW THE NEXT CRITERION. ALTHOUGH THIS TAKES A LITTLE MORE MACHINE TIME, IT IS NECESSARY TO INTEGRATE THE PREVIOUS VECTOR AGAIN TO MAINTAIN ACCURACY. (A CHECK ON THE HEIGHT AND THE STEP SIZE BEING USED IS MADE THREE TIMES IN EACH INTEGRATION STEP. THE HEIGHT IS COMPUTED IN C6HPEP AND THE CHECK IS MADE UPON EXIT FROM THIS SUBROUTINE.)

LINERAR INTERPOLATION IS USED WITH THE LAST VECTOR ABOVE AND THE FIRST VECTOR BELOW IMPACT TO OBTAIN A VECTOR AT IMPACT. USING THE VECTOR, C3GSCP COMPUTES THE LONGITUDE AND LATITUDE OF IMPACT. THE LONGITUDE AND LATITUDE OF IMPACT.

CALLING SEQUENCE -

```
TSX         CORKNI,4
PZE         L(INPUT),,L(OUTPUT)
PZE         K00001.,**
PZE         L(NUMBER OF INTEGRATION STEPS),,**
```

```
PZE         L(BANK ANGLE),,**
PZE         L(CAPSULE WEIGHT),,L(IMPACT HEIGHT)
ERROR       RETURN
NORMAL      RETURN
```

INPUT IS A SEVEN-LOCATION TABLE CONTAINING THE $t, \bar{r}, \bar{v}$ TO BE INTEGRATED, WHERE t IS IN FLOATING POINT SECONDS AND $\bar{r}$ and $\bar{v}$ ARE IN FLOATING POINT GEMINI UNITS.

OUTPUT IS A THREE-LOCATION TABLE CONTAINING t, λ, ϕ OF IMPACT, WHERE t IS IN FLOATING POINT SECONDS AND λ AND ϕ ARE IN FLOATING POINT RADIANS.

K00001 IS A FIXED ONE (1) AND INDICATES INTEGRATION TYPE.

NUMBER OF INTEGRATION STEPS IS A FIXED POINT NUMBER WHICH IS USED AS A SAFETY FACTOR. WITH A NOMINAL TRAJECTORY INTEGRATION TO IMPACT SHOULD ALWAYS TAKE LESS STEPS THAN THE SAFETY FACTOR. AN ERROR IS MADE FROM THE PROGRAM IF IMPACT IS NOT REACHED WITHIN THE SAFETY COUNT.

BANK ANGLE IS THE BANK ANGLE TO BE USED IN FLOATING POINT RADIANS AND WHOSE RANGE IS SUCH THAT $+0 \leq$ BANK ANGLE $\leq 2\pi$. A NEGATIVE BANK ANGLE REPRESENTS CONSTANT ROLL RATE AND HENCE, ZERO LIFT.

CAPSULE WEIGHT IS THE WEIGHT OF THE SPACECRAFT IN FLOATING POINT POUNDS. IT IS GENERALLY THE RE-ENTRY WEIGHT.

IMPACT HEIGHT IS THE HEIGHT IN FLOATING POINT GEMINI LENGTH UNITS AT WHICH OUTPUT IS DESIRED. HOWEVER, IF THE HEIGHT OF THE ANCHOR VECTOR IS LESS THAN THIS SPECIFIC HEIGHT, THE HEIGHT OF THE ANCHOR VECTOR IS USED AS THE IMPACT HEIGHT.

INPUT/OUTPUT REQUIREMENTS FOR TYPE 2 INTEGRATION

TYPE 2 INTEGRATION IS USED TO UPDATE VECTORS OR TO PREDICT A VECTOR AT A CERTAIN FUTURE TIME. IT MAY BE USED AT ANY TIME DURING LAUNCH OR ABORT BY SIMPLY SPECIFYING THE CAPSULE WEIGHT AND STEP SIZE.

TYPE 2 INTEGRATION WILL INTEGRATE AT ANY STEP SIZE (WITHIN REASONABLE LIMITS) FOR ANY NUMBER OF STEPS AND WILL OUTPUT THE RESULTING VECTOR. ALL OF THIS IS SPECIFIED IN THE CALLING SEQUENCE.

CALLING SEQUENCE -

```
TSX         CORKNI,4
PZE         L(INPUT),,L(OUTPUT)
PZE         K00002,,L(STEP SIZE)
PZE         L(NUMBER OF INTEGRATION STEPS),,**
PZE         L(BANK ANGLE),,L(THRUST INDICATOR)
PZE         L(CAPSULE WEIGHT),,**
ERROR       RETURN
NORMAL      RETURN
```

INPUT IS A SEVEN-LOCATION TABLE CONTAINING THE $t, \bar{r}, \bar{v}$ TO BE INTEGRATED, WHERE t IS IN FLOATING POINT SECONDS AND $\bar{r}$ and $\bar{v}$ ARE IN FLOATING POINT GEMINI UNITS.

OUTPUT IS A SEVEN-LOCATION TABLE CONTAINING THE INTEGRATED $t, \bar{r}, \bar{v}$. THE OUTPUT FORMAT IS THE SAME AS THE INPUT FORMAT.

K00002 IS A FIXED POINT TWO (2) AND INDICATES INTEGRATION TYPE.

STEP SIZE IS GIVEN IN FLOATING POINT SECONDS. (BY SPECIFYING A NEGATIVE STEP SIZE, ONE IS, IN EFFECT, SPECIFYING A BACKWARD INTEGRATION.)

NUMBER OF INTEGRATION STEPS IS A FIXED POINT NUMBER AND INDICATES THE NUMBER OF STEPS TO BE TAKEN.

BANK ANGLE IS THE BANK ANGLE TO BE USED IN FLOATING POINT RADIANS, AND WHOSE RANGE IS SUCH THAT $+0 \leq$ BANK ANGLE $\leq 2\pi$. A NEGATIVE BANK ANGLE REPRESENTS CONSTANT ROLL RATE AND HENCE, ZERO LIFT.

THRUST INDICATOR INDICATES WHETHER OR NOT THE ACCELERATION DUE TO THE NOMINAL ON-BOARD PROPULSION IS TO BE INCLUDED IN THE INTEGRATION. IF THIS THRUST INDICATOR EQUALS 0, THIS ACCELERATION WILL NOT BE TAKEN IN ACCOUNT. IF THE THRUST INDICATOR IS NONZERO, THIS ACCELERATION WILL BE COMPUTED.

CAPSULE WEIGHT IS THE WEIGHT OF THE SPACECRAFT IN FLOATING POINT POUNDS.

INPUT/OUTPUT REQUIREMENTS FOR TYPE 3 INTEGRATION

TYPE 3 INTEGRATION GENERATES AN ORBIT OR REENTRY TABLE OF $\bar{r}$ AND $\bar{v}$ VECTORS AND A CORRESPONDING TABLE OF HEIGHTS (H). THESE TABLES ARE USUALLY USED TO COMPUTE THE GMTRC AND BECAUSE THESE TABLES ARE USED BY INTERPOLATION PROGRAMS, IT IS NECESSARY TO HAVE THREE HEIGHTS IN THE TABLE BELOW H_{CRIT}, A CRITERION HEIGHT SPECIFIED IN THE CALLING SEQUENCE. THE MAXIMUM NUMBER OF ENTRIES IN THE TABLES AND THE INTERVAL BETWEEN EACH ENTRY IS ALSO GIVEN IN THE CALLING SEQUENCE.

CALLING SEQUENCE -

```
TSX         CORKNI,4
PZE         L(INPUT),,L r̄, v̄ OUTPUT TABLE)
PZE         K00003,,L(STEP SIZE)
PZE         L(NUMBER OF STEPS FORWARD),,
            L(NUMBER OF STEPS BACKWARD)
PZE         L(BANK ANGLE),,L(OUTPUT H TABLE)
PZE         L(CAPSULE WEIGHT),,L(CRITERION
            HEIGHT)
ERROR       RETURN
NORMAL      RETURN
```

INPUT IS A SEVEN-LOCATION TABLE CONTAINING THE $t, \bar{r}, \bar{v}$ TO BE INTEGRATED, WHERE t IS IN FLOATING POINT SECONDS AND $\bar{r}$ AND $\bar{v}$ ARE IN FLOATING POINT GEMINI UNITS.

$\bar{r}, \bar{v}$ OUTPUT TABLE IS A TABLE IN THE FORMAT $t_1, t_n, \bar{r}_1, \bar{v}_1, \ldots, \bar{r}_{n-1}, \bar{v}_{n-1}, \bar{r}_n, \bar{v}_n$ WHERE t_1 AND t_n ARE IN FLOATING POINT SECONDS AND $\bar{r}_i, \bar{v}_i$ ARE IN FLOATING POINT GEMINI UNITS.

K00003 IS FIXED POINT THREE (3) AND INDICATES INTEGRATION TYPE.

STEP SIZE IS GIVEN IN FLOATING POINT SECONDS. THIS STEP SIZE WILL BE USED TO INTEGRATE BACKWARD AND FORWARD. (THE INTEGRATOR CHANGES THE SIGN OF THE STEP SIZE TO INTEGRATE BACKWARD.)

NUMBER OF STEPS FORWARD INDICATES THE MAXIMUM NUMBER OF STEPS TO BE TAKEN WITH A POSITIVE STEP SIZE. IF THREE VECTORS ARE GENERATED BELOW H_{CRIT} BEFORE THE NUMBER OF STEPS FORWARD IS DEPLETED, A NORMAL EXIT WILL BE MADE. THAT IS, THE NUMBER OF STEPS FORWARD IS FOREMOST A SAFETY FACTOR.

NUMBER OF STEPS BACKWARD INDICATES THE NUMBER OF STEPS TO BE TAKEN WITH A NEGATIVE STEP SIZE. TYPE 3 WILL ALWAYS INTEGRATE THE SPECIFIED NUMBER OF STEPS BACKWARD.

BANK ANGLE IS THE BANK ANGLE TO BE USED IN FLOATING POINT RADIANS, AND WHOSE RANGE IS SUCH THAT $+\phi \leq$ BANK ANGLE $\leq 2\pi$. A NEGATIVE BANK ANGLE REPRESENTS A CONSTANT ROLL RATE AND HENCE, ZERO LIFT.

CUTPUT H TABLE IS A TABLE IN THE FORMAT $t_1, t_n, H_1, H_2, \ldots, H_{n-1}, H_n$, WHERE t_1 AND t_n ARE IN FLOATING POINT SECONDS AND H_i IN FLOATING POINT GEMINI UNITS.

CAPSULE WEIGHT IS THE WEIGHT OF THE SPACE CRAFT IN FLOATING POINT POUNDS.

CRITERION HEIGHT (H_{CRIT} IS THE HEIGHT IN FLOATING POINT GEMINI UNITS BELOW WHICH THREE VECTORS MUST BE GENERATED.

(IN BOTH OUTPUT TABLES THE TIMES OF THE ENTRIES ARE ALWAYS SUCH THAT $t_i < t_{i+1}$.)

UPON EXIT FROM TYPE 3 INTEGRATION THE VALUE OF THE ACCUMULATOR INDICATES THE STATUS OF THE GENERATED TABLE -

A) IF THE SIGN OF THE AC IS POSITIVE, THERE IS CONTAINED IN THE TABLES AT LEAST ONE ENTRY WITH A HEIGHT ABOVE H_{CRIT} AND THREE ENTRIES WITH HEIGHTS BELOW H_{CRIT}.

B) IF THE SIGN OF THE AC IS MINUS AND THE AC IS NONZERO, THERE IS CONTAINED IN THE TABLES AT LEAST ONE ENTRY WITH A HEIGHT GREATER THAN H_{CRIT}, BUT THERE ARE NOT THREE ENTRIES WITH HEIGHTS BELOW H_{CRIT}.

C) IF THE SIGN OF THE AC IS MINUS AND THE AC IS ZERO, THERE IS NO ENTRY IN THE TABLE WITH A HEIGHT GREATER THAN H_{CRIT}.

INPUT/OUTPUT REQUIREMENTS FOR TYPE 4 INTEGRATION

TYPE 4 INTEGRATION IS USED TO GENERATE AN ORBIT OR REENTRY TABLE OF $\bar{r}$ AND $\bar{v}$ VECTORS. TYPE 4 INTEGRATES UNTIL A VECTOR WITH A HEIGHT BELOW H_{STOP} (A HEIGHT SPECIFIED IN THE CALLING SEQUENCE) IS GENERATED, OR THE NUMBER OF STEPS FORWARD (AS SPECIFIED IN THE CALLING SEQUENCE) IS DEPLETED.

CALLING SEQUENCE -

```
        TSX         CORKNI,4
        PZE         L(INPUT),,L( r̄, v̄ OUTPUT TABLE)
        PZE         K00004,,6(STEP SIZE)
        PZE         L(NUMBER OF STEPS FORWARD),,
                    L(NUMBER OF STEPS BACKWARD)
        PZE         L(BANK ANGLE),,**
        PZE         L(CAPSULE WEIGHT),,
                    L(STOPPING HEIGHT)
        ERROR       RETURN
        NORMAL      RETURN
```

INPUT IS A SEVEN-LOCATION TABLE CONTAINING THE t, $\bar{r}$, $\bar{v}$ TO BE INTEGRATED WHERE t IS IN FLOATING POINT SECONDS AND $\bar{r}$ AND $\bar{v}$ ARE IN FLOATING POINT GEMINI UNITS.

$\bar{r}$, $\bar{v}$ OUTPUT TABLE IS A TABLE IN THE FORMAT $t_1, t_n, \bar{r}_1, \bar{v}_1, \ldots, \bar{r}_{n-1}, \bar{v}_{n-1}, \bar{r}_n, \bar{v}_n$ WHERE t_1 AND t_n ARE IN FLOATING POINT SECONDS AND $\bar{r}_i$ AND $\bar{v}_i$ ARE IN FLOATING POINT GEMINI UNITS.

K00004 IS FIXED POINT FOUR (4) AND INDICATES INTEGRATION TYPE.

STEP SIZE IS IN FLOATING POINT SECONDS.

NUMBER OF STEPS FORWARD IS GIVEN IN FIXED POINT.

A NORMAL EXIT IS MADE FROM THE PROGRAM IF A VECTOR BELOW H_{STOP} IS REACHED EVEN IF THE NUMBER OF STEPS FORWARD IS NOT DEPLETED.

NUMBER OF STEPS BACKWARD IS A FIXED POINT NUMBER AND TYPE 4 INTEGRATES THE SPECIFIED NUMBER OF STEPS WITH A NEGATIVE STEP SIZE.

BANK ANGLE IS THE BANK ANGLE TO BE USED IN FLOATING POINT RADIANS, AND WHOSE RANGE IS SUCH THAT $+0 \leq$ BANK ANGLE $\leq 2\pi$. A NEGATIVE BANK ANGLE REPRESENTS A CONSTANT ROLL RATE AND HENCE, ZERO LIFT.

CAPSULE WEIGHT IS THE WEIGHT OF THE SPACECRAFT IN FLOATING POINT POUNDS.

STOPPING HEIGHT H_{STOP} IS THE HEIGHT IN FLOATING POINT GEMINI UNITS BELOW WHICH INTEGRATION TYPE 4 STOPS INTEGRATING FORWARD.

UPON EXIT FROM TYPE 4, THE ACCUMULATOR GIVES THE STATUS OF THE GENERATED TABLE.

A) IF THE SIGN OF THE AC IS POSITIVE, THE TABLE HAS ONE ENTRY WITH A HEIGHT EQUAL TO OR LESS THAN H_{STOP}.

B) IF THE SIGN OF THE AC IS MINUS, THE TABLE HAS NO ENTRY WITH A HEIGHT EQUAL TO OR LESS THAN H_{STOP}.

METHOD

A. INITIALIZATION -

THE PROGRAM FIRST COMPUTES THOSE QUANTITIES AND SETS INDICATORS WHICH ARE USED BY ALL FOUR TYPES OF INTEGRATION. THESE INCLUDE THE SAVING OF INDEX REGISTERS, SETTING UP OF INPUT AND THE OUTPUT ADDRESS, SETTING THE FIRST PASS AND THRUST INDICATORS, FIXING THE NUMBER OF STEPS FORWARD AND SETTING THE SPECIFIED INTEGRATION TYPE.

IT IS ALSO NECESSARY TO COMPUTE THE RATIO (N) OF THE AREA OF THE SPACECRAFT TO THE MASS OF THE SPACECRAFT IN GEMINI UNITS WHICH IS USED BY C4DRFT.

$$N = \frac{\text{area of S/C (meters}^2\text{)}}{\text{mass of S/C (lbs)}} \cdot \frac{\text{mass of Gemini earth (kg)}}{\text{Gemini length unit (meters}^2\text{)}} \cdot \frac{1}{\text{\# of kg/lb}}$$

ALL THE FACTORS INVOLVED IN COMPUTING N, WITH THE EXCEPTION OF THE MASS OF THE S/C, ARE CONTAINED IN TC.WCF. THUS, IN REAL TIME THE ONLY COMPUTATION NECESSARY IS

$$N = \frac{c(TC.WCF)}{\text{mass of S/C (lbs)}}$$

$-\frac{N}{2}$ IS STORED IN T.HLFN FOR USE BY C4DRFT.

NEXT, THE SINE AND COSINE OF THE BANK ANGLE ARE COMPUTED AND STORED IN TCSNBA AND TCCSBA

CONTROL IS THEN TRANSFERRED TO ONE END OF THE FOUR TYPES OF INTEGRATION, TO FINISH IN A SPECIFIC WAY, THE INITIALIZATION PARTICULAR TO THAT TYPE OF INTEGRATION.

A) TYPE 1 DETERMINES THE HEIGHT OF THE ANCHOR VECTOR, SETS THE PROPER STEP SIZE BLOCK, AND FIXES THE NECESSARY STEP SIZE INDICATORS. IT ALSO SETS THE PROPER IMPACT HEIGHT.

B) TYPE 2 INTERROGATES THE THRUST INDICATOR AND SETS THE STEP SIZE TO BE USED.

C) TYPE 3 FIXES THE NUMBER OF BACKWARD STEPS, SETS UP THE STEP SIZE TO BE USED, AND ADJUSTS INDEX REGISTERS FOR STORING IN THE TWO OUTPUT TABLES.

D) TYPE 4 INITIALIZES THE SAME WAY AS TYPE 3 WITH THE EXECPTION OF STORING IN ONLY ONE OUTPUT TABLE.

B. MATHEMATICS OF RUNGE-KUTTA INTEGRATION -

THE RUNGE-KUTTA NUMERICAL INTEGRATION PROGRAM IS OF THE FOURTH ORDER AND INTEGRATES SECOND-ORDER EQUATIONS OF THE TYPE.

$y'' = f(x, y, y')$. -

THIS EQUATION IS INTEGRATED USING THE FOLLOWING FORMULAS (NUMERICAL MATHEMATICS ANALYSIS, J.B.SCARBOROUGH, THE JOHNS HOPKINS PRESS, 1950, CHAPTER XI, ART. 97 (B) (2).) IN THE GIVEN ORDER -

$$k_1 = h\, f(x_n, y_n, y'_n)$$

$$k_2 = h\, f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} y'_n + \frac{h}{8} k_1, y'_n + \frac{k_1}{2}\right)$$

$$k_3 = h\, f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} y'_n + \frac{h}{8} k_1, y'_n + \frac{k_2}{2}\right)$$

$$k_4 = h\, f\left(x_n + h, y_n + h\, y'_n + \frac{h}{2} k_3, y'_n + k_3\right)$$

$$\Delta y = h\left[y'_n + \frac{1}{6}(k_1 + k_2 + k_3)\right]$$

$$\Delta y' = \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \text{ where } n = 0, 1, 2 \ldots .$$

FOR THE GEMINI SYSTEM THE ABOVE GENERAL SECOND-ORDER SCALAR EQUATION BECOMES THE VECTOR EQUATION $\ddot{\bar{r}} = f(t, \bar{r}, \dot{\bar{r}})$, THAT IS A FUNCTION OF TIME, POSITION, AND VELOCITY. THE ACCELERATION $\ddot{\bar{r}}$ IS EQUAL TO $\ddot{\bar{r}}_p + \ddot{\bar{r}}_d, + \ddot{\bar{r}}_1 + \ddot{\bar{r}}_t$ WHERE

$\ddot{\bar{r}}_p$ IS THE ACCELERATION OF THE SPACECRAFT DUE TO THE POTENTIAL OF THE EARTH,

$\ddot{\bar{r}}_d$ IS THE ACCELERATION OF THE SPACECRAFT DUE TO ATMOSPHERIC DRAG,

$\ddot{\bar{r}}_1$ IS THE ACCELERATION OF THE SPACECRAFT DUE TO

LIFT, AND

$\ddot{\bar{r}}_t$ IS THE ACCELERATION OF THE SPACECRAFT DUE TO THE THRUST OF THE ON-BOARD PROPULSION SYSTEM.

IN THE GEMINI SYSTEM, h IS THE TIME INTERVAL OVER WHICH ONE INTEGRATION STEP TAKES PLACE. IT IS USUALLY REFERRED TO AS THE STEP SIZE.

THE BASIC WORK IS DONE BY THE SUBROUTINES IN COMPUTING THE ACCELERATION OVER THE INTERVAL. SINCE THE INPUT IS ALWAYS THE COMPONENTS OF THE POSITION AND VELOCITY VECTORS, THE ACCELERATIONS ARE COMPUTED ALONG THE COMPONENTS OF THE GEMINI COORDINATE SYSTEM. THUS, FOR EXAMPLE, THE NOTATION $\ddot{\bar{r}}_p$ IS USED TO DENOTE THE ACCELERATION DUE TO THE POTENTIAL OF THE EARTH ALONG THE AXES OF THE GEMINI COORDINATE SYSTEM.

THE k_i FORMULAS MUST THEN BE WRITTEN -

$$\overline{k_1} = h\, f\,(t_n, \bar{r}_n, \dot{\bar{r}}_n)$$

$$\overline{k_2} = h\, f\,(t_n + \frac{h}{2}, \bar{r}_n + \frac{h}{2}\,\dot{\bar{r}}_n + \frac{h}{8}\,\overline{k_1}, \dot{\bar{r}}_n + \frac{\overline{k_1}}{2})$$

$$\overline{k_3} = h\, f\,(t_n + \frac{h}{2}, \bar{r}_n + \frac{h}{2}\,\dot{\bar{r}}_n + \frac{h}{8}\,\overline{k_1}, \dot{\bar{r}}_n + \frac{\overline{k_2}}{2})$$

$$\overline{k_4} = h\, f\,(t_n + h, \bar{r}_n + h\,\dot{\bar{r}}_n + \frac{h}{2}\,\overline{k_3}, \dot{\bar{r}}_n + \overline{k_3})$$

$$\Delta\bar{r} = h\left[\dot{\bar{r}}_n + \frac{1}{6}\,(\bar{k}_1 + \bar{k}_2 + \bar{k}_3)\right]$$

$$\Delta\dot{\bar{r}} = \frac{1}{6}\,(\bar{k}_1 + 2\bar{k}_2 + 2\bar{k}_3 + \bar{k}_4)$$

THE INPUT $\bar{r}_n$ AND $\dot{\bar{r}}_n$ AT THE BEGINNING OF AN INTEGRATION STEP, I.E., AT t_n IS GIVEN TO C6HPEP AND C4DRFT, WHICH COMPUTE THE ACCELERATIONS AT THE BEGINNING OF THE INTERVAL. THUS WE HAVE COMPUTED $f\,(t_n, \bar{r}_n, \dot{\bar{r}}_n) = \ddot{\bar{r}}_p + \ddot{\bar{r}}_d + \ddot{\bar{r}}_l + \ddot{\bar{r}}_t$ (or $\ddot{\bar{r}}$),

AND $\overline{k_1} = h\ddot{\bar{r}}$.

IN ORDER TO COMPUTE $\overline{k_2}$ (at $t_n + \frac{h}{2}$), THE INPUT TO C6HPEP AND C4DRFT BECOMES $\bar{r}_n + \frac{h}{2}\dot{\bar{r}}_n + \frac{h}{8}\overline{k_1}$ and $\dot{\bar{r}}_n + \frac{k_1}{2}$. THEIR OUTPUT IS AN APPROXIMATION OF THE TOTAL ACCELERATION AT THE MIDDLE OF THE INTERVAL. AGAIN $\overline{k_2} = h\ddot{\bar{r}}$.

IN COMPUTING $\bar{k}_3$ THE INPUT FOR EVALUATING THE ''RE-ESTIMATED'' ACCELERATIONS AT THE MIDDLE OF THE INTERVAL (at $t_n + \frac{h}{2}$) is $\bar{r}_n + \frac{h}{2}\dot{\bar{r}}_n + \frac{h}{8}\bar{k}_1$ and $\dot{\bar{r}}_n + \frac{k_2}{2}$. THE FIRST INDEPENDENT VARIABLE (POSITION) IS IDENTICAL TO THE SAME VARIABLE USED IN COMPUTING $\bar{k}_2$. THEREFORE, THOSE FUNCTIONS USED IN COMPUTING ACCELERATIONS WHICH ARE FUNCTIONS OF POSITION ONLY, NEED NOT BE RECOMPUTED FOR THE EVALUATION OF k_3. THIS SYMMETRY SAVES A CONSIDERABLE AMOUNT OF MACHINE TIME AND IS ONE OF THE REASONS FOR CHOOSING THIS FORM OF NUMERICAL INTEGRATION EQUATIONS. IT IS SIMPLE THEN TO COMPUTE $\bar{k}_3 - h\ddot{\bar{r}}$.

FINALLY, IN APPROXIMATING THE ACCELERATION AT THE END OF THE INTERVAL(at $t_n + h$)C6HPEP AND C4DRFT USE AS INPUT $\bar{r}_n + h\dot{\bar{r}}_n + \frac{h}{2}\bar{k}_3$ AND $\dot{\bar{r}}_n + \overline{k_3}$. AGAIN $\overline{k_4} = h\ddot{\bar{r}}$.

THE POSITION AND VELOCITY VECTORS AT THE END OF THIS INTEGRATION STEP, WHICH CAN BE USED AS INPUT FOR ANOTHER INTEGRATION STEP ARE

$$\bar{r}_{n+1} = \bar{r}_n + h\left[\dot{\bar{r}}_n + \frac{1}{6}(\bar{k}_1 + \bar{k}_2 + \bar{k}_3)\right] \text{ and}$$

$$\dot{\bar{r}}_{n+1} = \dot{\bar{r}}_n + \frac{1}{6}(\bar{k}_1 + 2\bar{k}_2 + 2\bar{k}_3 + \bar{k}_4).$$

USAGE

A) ERROR CONDITIONS -

IN ALL FOUR TYPES OF INTEGRATION AN ERROR RETURN MAY BE THE RESULT OF AN ERROR RETURN FROM THE SUBROUTINES C6HPEP AND C4DRFT OR FROM THE LIBRARY SUBROUTINE U1SICO. IN ADDITION, TYPE 1 INTEGRATION CAN GIVE AN EROR RETURN FROM A DEPLETION OF THE SAFETY COUNT BEFORE H_{IMPACT} IS REACHED OR FROM AN ERROR RETURN FROM C3GSCP. THE PROGRAM ITSELF GIVES AN ERROR RETURN IF THE INTEGRATION TYPE SPECIFIED IS NOT A FIXED POINT 1, 2, 3 OR 4.

B) TIME REQUIRED -

TYPE 1 - THE TIME REQUIRED TO PREDICT AN IMPACT POINT BASICALLY DEPENDS UPON THE POSITION AND VELOCITY OF THE INPUT VECTOR. SMALL VARIATIONS IN TIME ARE DUE TO THE NUMBER OF STEP SIZE CHANGES NEEDED AND ALSO TO WHERE IN THE TRAJECTORY THIS CHANGE OCCURS. SOME AVERAGE TIMES ARE SHOWN BELOW. THESE VECTORS WERE TAKEN FROM A NOMINAL GT-1 TRAJECTORY AND INTEGRATED TO 60,000 FEET.

$\Delta t_{L.O.}$ (SECONDS)	ANCHOR VECTOR HEIGHT (FEET)	LIFT CONFIGURATION	TIMING (MILLI-SECONDS)
		NONE	101
94	12,000	MAXIMUM	156
		60° BANK ANGLE	166
		NONE	128

(SECONDS)	ANCHOR VECTOR HEIGHT (FEET)	LIFT CONFIGURATION	TIMING (MILLI-SECONDS)
104	84,000	MAXIMUM	197
		60° BANK ANGLE	211
		NONE	147
160	227,000	MAXIMUM	217
		60° BANK ANGLE	234
		NONE	70
275	489,000	MAXIMUM	114
		60° BANK ANGLE	117
		NONE	90
315	533,000	MAXIMUM	141
		60° BANK ANGLE	141
		NONE	134
335	531,000	MAXIMUM	234
		60° BANK ANGLE	233

TYPE 2 - TIMING FOR TYPE 2 INTEGRATION IS

NC LIFT - 10 MILLISECONDS/STEP*

MAXIMUM LIFT - 13 MILLISECONDS/STEP,

(IF THRUST IS TO BE COMPUTED, ADD 4.5 MILLI-SECONDS/STEP.)

60° BA - 15 MILLISECONDS/STEP*

TYPE 3 - ADD 0.2 MILLISECOND/STEP TO TYPE 2 TIMINGS AND CONSIDER 0.4 MILLISECONDS FOR INITIALIZATION.

TYPE 4 - ADD 0.1 MILLISECOND/STEP TO TYPE 2 TIMINGS AND CONSIDER 0.3 MILLISECOND FOR INITIALIZATION.

C) STORAGE REQUIRED - 780 LOCATIONS EXCLUDING THAT REQUIRED BY SUBROUTINES, EXTERNAL CONSTANTS, AND EXTERNAL TABLES.

D) CHECKOUT - PORTIONS OF THE INTEGRATOR WITH THE EXCEPTION OF THE LIFT AND THRUST FUNCTIONS WERE CHECKED AGAINST THE MA-9 INTEGRATOR. IN TYPE 1 INTEGRATION, EFFORTS WERE MADE TO LIMIT THE DISTANCE BETWEEN IMPACTS GENERATED BY CORKNI AND THE MA-9 INTEGRATOR TO THREE MILES. THE OTHER TYPES OF INTEGRATION WERE CHECKED WITH BALLISTIC TRAJECTORIES AND WERE FOUND TO COMPARE EXTREMELY WELL WITH THE MA-9 SYSTEM.

LIFTING REENTRY TRAJECTORIES COMPARE CLOSELY TO LIFTING REENTRY TRAJECTORIES OF MSC IN HOUSTON. TRAJECTORIES WITH THRUST WERE SATISFACTORILY COMPARED WITH THE GEMINI SYSTEM ORBIT INTEGRATOR AT GODDARD.

CORKNI FLOW CHART SYMBOLS

N_p (OR G) - NUMBER OF FORWARD INTEGRATION STEPS
N_b - NUMBER OF BACKWARD INTEGRATION STEPS
e - FIRST INTEGRATION STEP INDICATOR
T_p - THRUST COMPUTATION INDICATOR
A - AREA OF SPACECRAFT
W - WEIGHT OF SPACECRAFT
k_w - CONVERTS A/W TO GEMINI UNITS
C_i - HEIGHT CRITERION FOR CHANGING STEP SIZE
F_i - STEP SIZE INDICATOR
$\bar{F}_i$ - STEP SIZE VECTOR
$\bar{T}_F$ - STEP SIZE VECTOR CURRENTLY IN USE

CORKNI FLOW CHART SYMBOLS (CONTINUED)

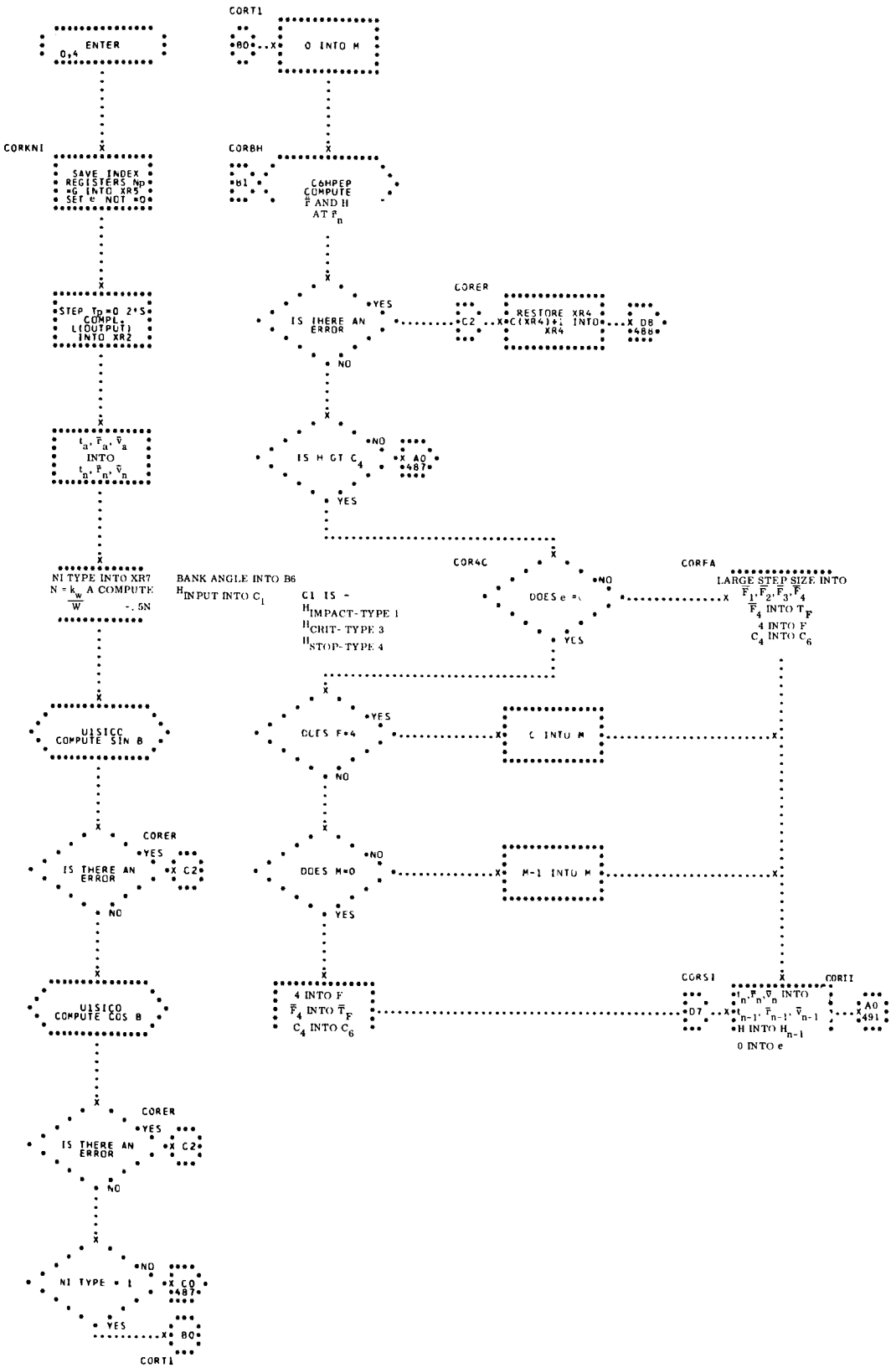
H - HEIGHT OF SPACECRAFT

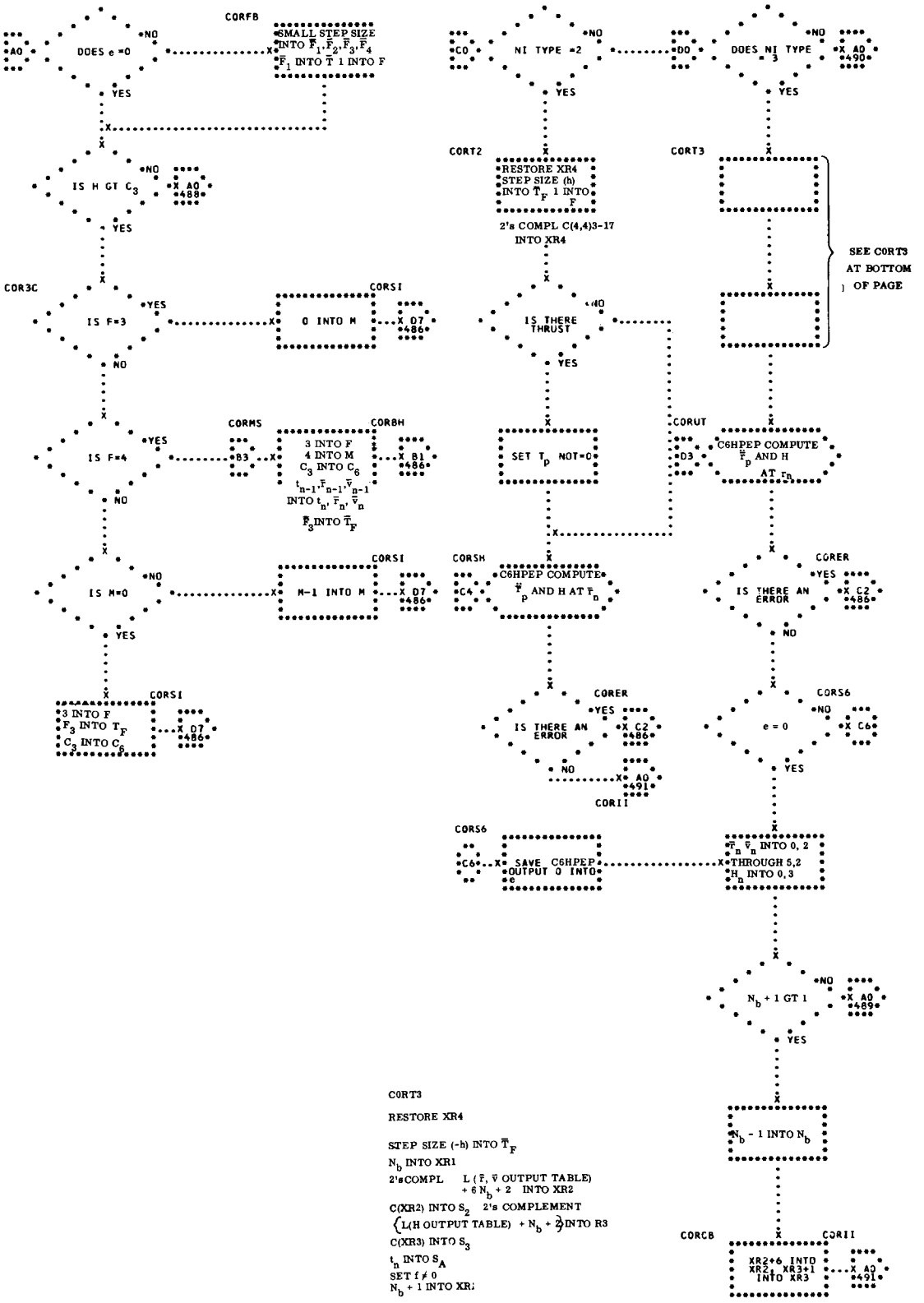
B* - BANK ANGLE

M - INDICATES NUMBER OF TIMES F_{i-1} IS USED ABOVE C_i

Q - KEY FOR INTERPOLATION FOR IMPACT (TYPE 1)

h - STEP SIZE (TYPE 2, TYPE 3, + TYPE 4)


CORT1
ENTER
0,4
B0
0 INTO M
CORKNI
SAVE INDEX REGISTERS Np =G INTO XR5 SET e NOT =0
CORBH
B1
C6HPEP COMPUTE r̄ AND H AT r̄n
STEP Tp=0 2'S COMPL. L(OUTPUT) INTO XR2
IS THERE AN ERROR
YES
NO
CORER
C2
RESTORE XR4 C(XR4)+1 INTO XR4
D8 488
ta, r̄a, v̄a INTO tn, r̄n, v̄n
IS H GT C4
A0 487
NI TYPE INTO XR7
N = kw A COMPUTE W -.5N
BANK ANGLE INTO B6
HINPUT INTO C1
C1 IS -
HIMPACT-TYPE 1
HCRIT-TYPE 3
HSTOP-TYPE 4
COR4C
DOES e =
CORFA
LARGE STEP SIZE INTO
F̄1, F̄2, F̄3, F̄4
F̄4 INTO TF
4 INTO F
C4 INTO C6
U1SICC COMPUTE SIN B
DOES F=4
C INTO M
IS THERE AN ERROR
C2
DOES M=0
M-1 INTO M
U1SICO COMPUTE COS B
4 INTO F
F̄4 INTO T̄F
C4 INTO C6
CORSI
D7
CORII
tn, r̄n, v̄n INTO tn-1, r̄n-1, v̄n-1
H INTO Hn-1
0 INTO e
A0 491
NI TYPE = 1
C0 487
B0
CORT1



CORT3

RESTORE XR4

STEP SIZE (-h) INTO $\bar{T}_F$

N_b INTO XR1

2's COMPL L ($\bar{r}$, $\bar{v}$ OUTPUT TABLE) + 6 N_b + 2 INTO XR2

C(XR2) INTO S_2 2's COMPLEMENT

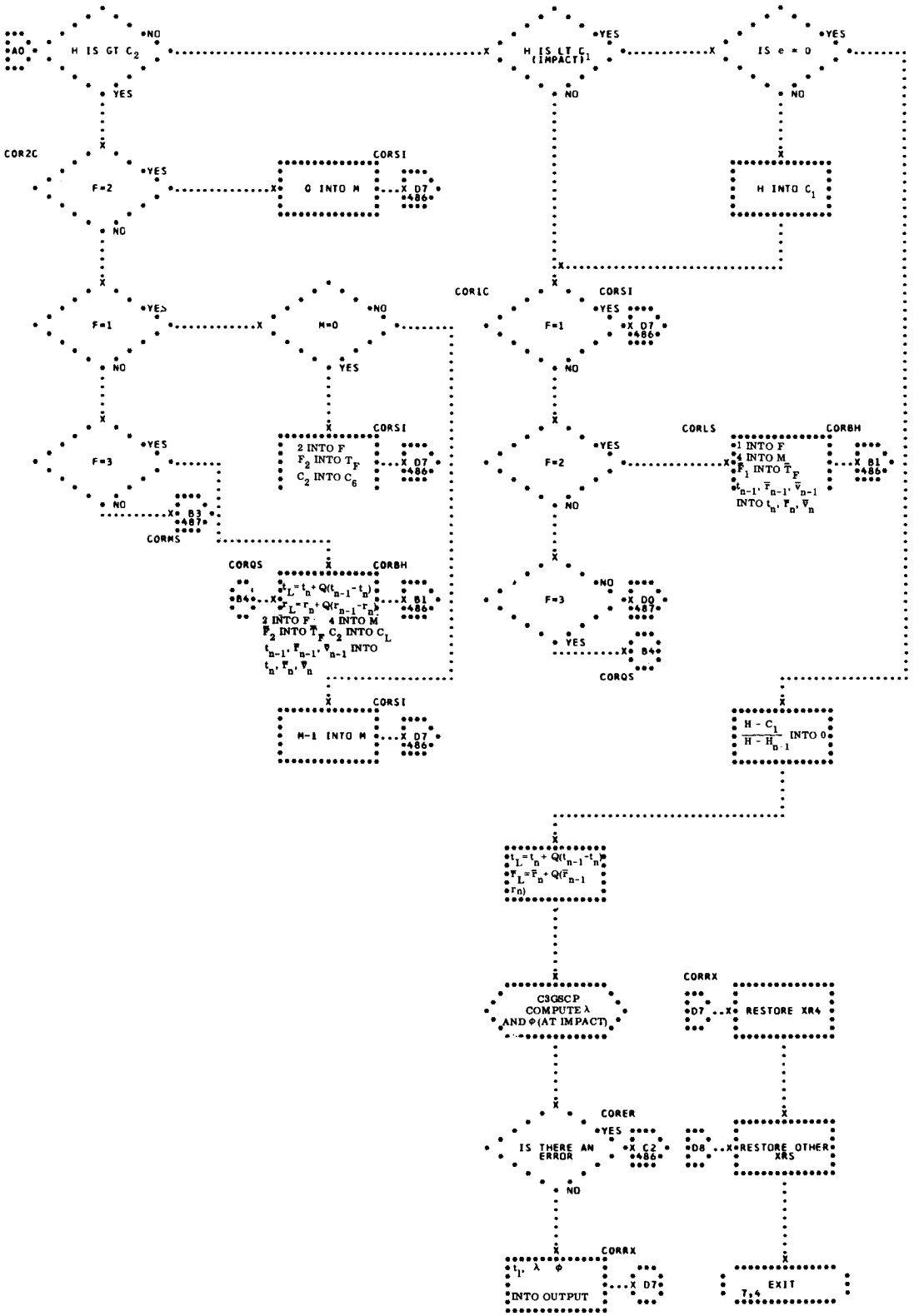
$\left\{L(H \text{ OUTPUT TABLE}) + N_b + 2\right\}$ INTO R3

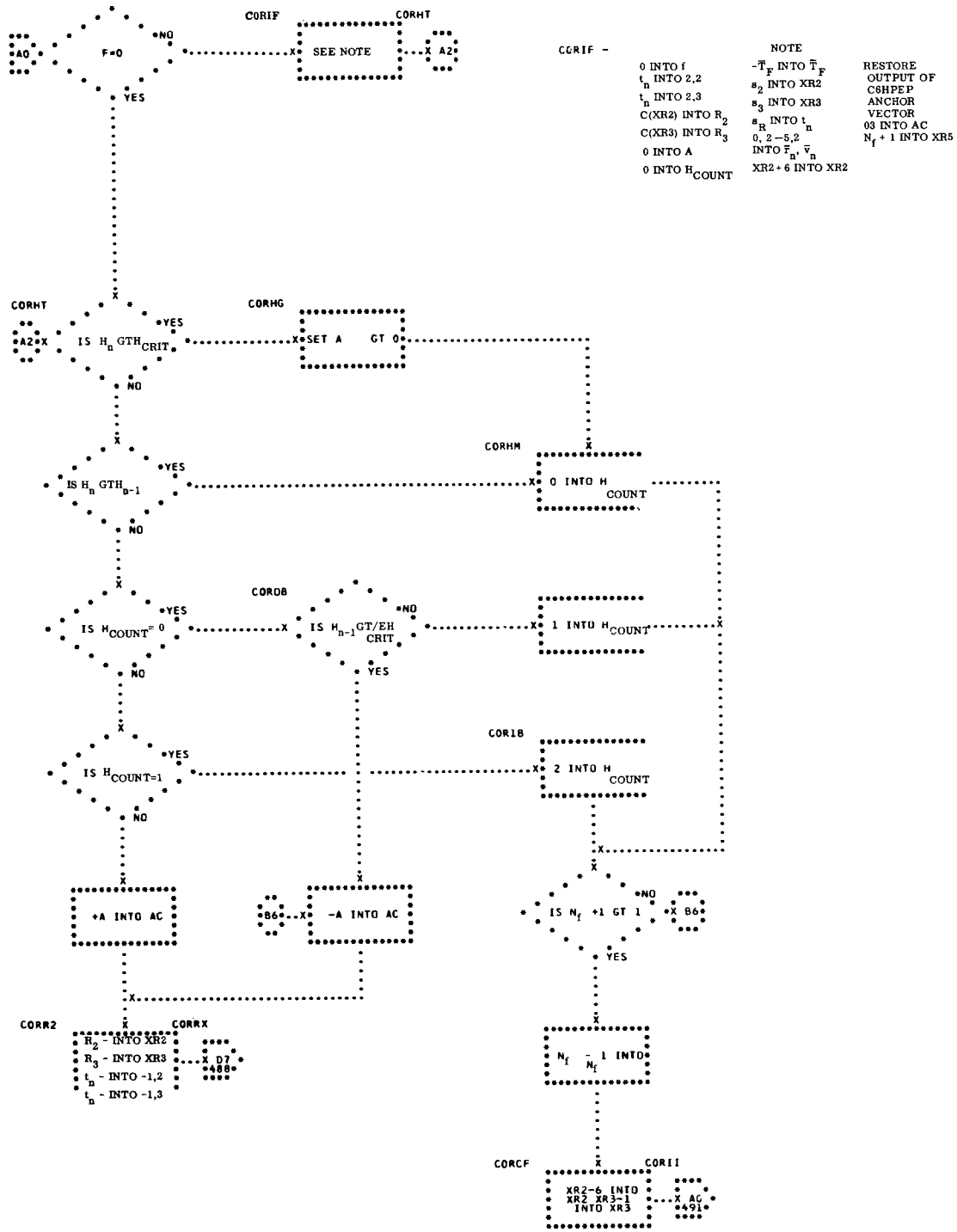
C(XR3) INTO S_3

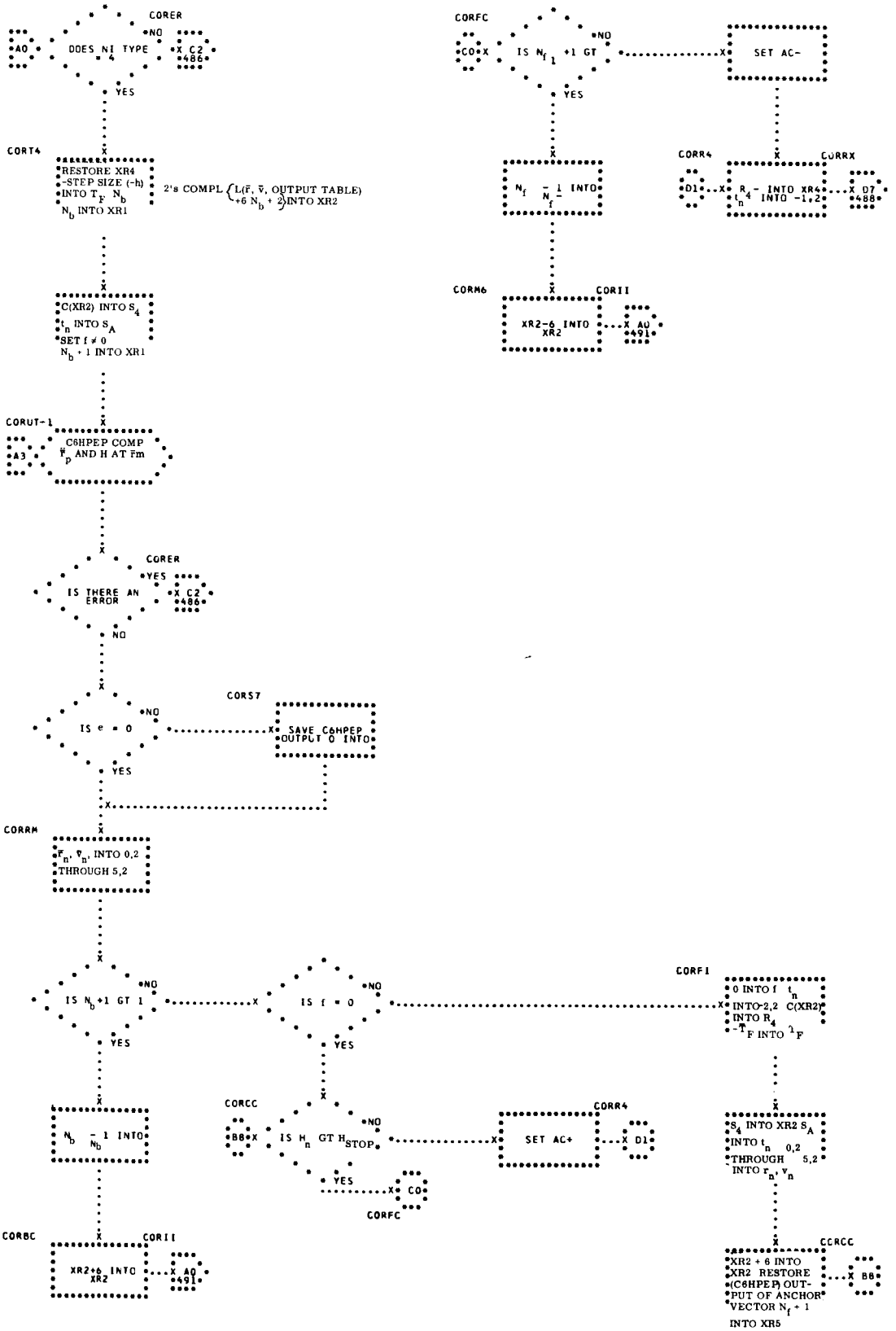
t_n INTO S_A

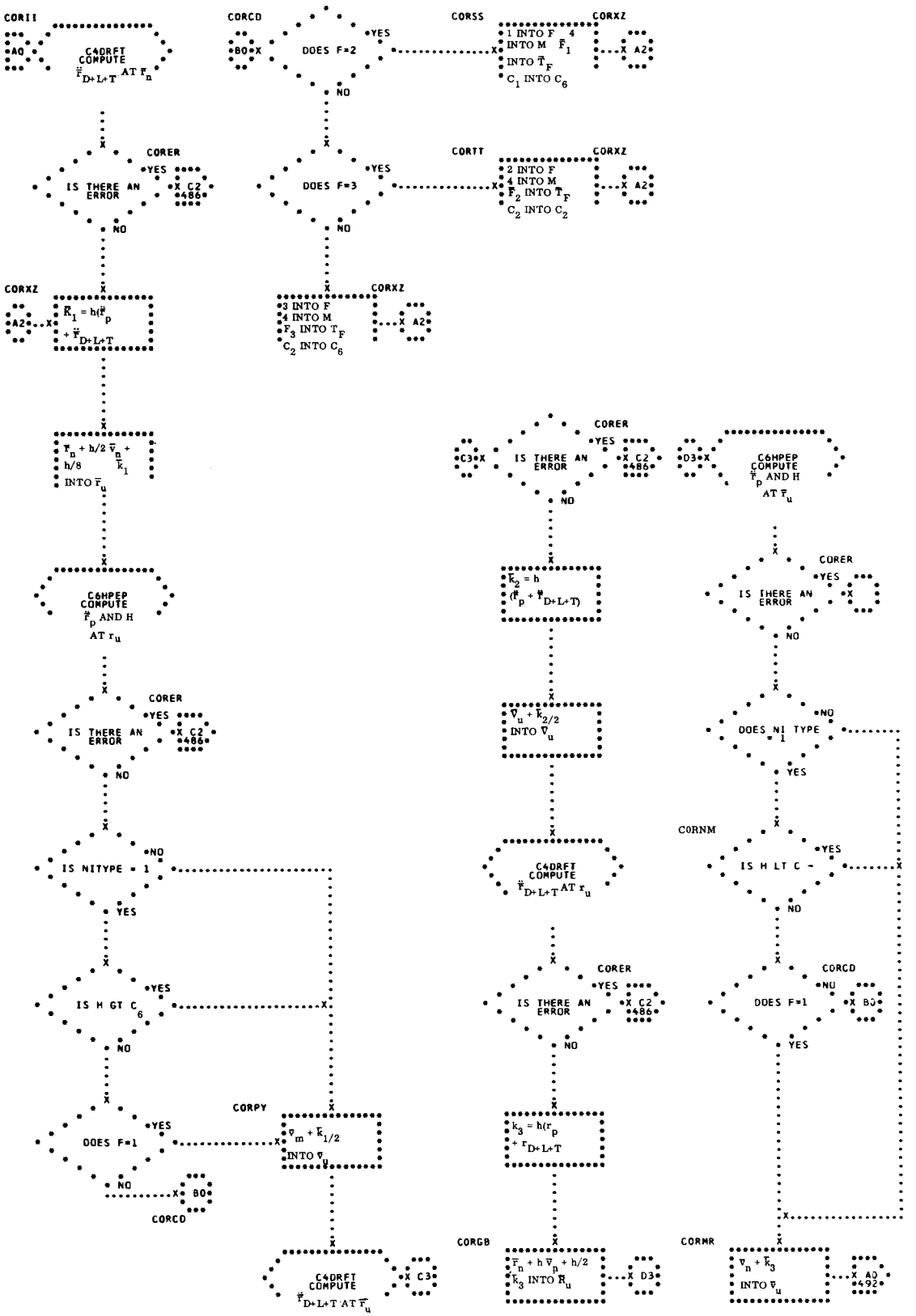
SET f ≠ 0

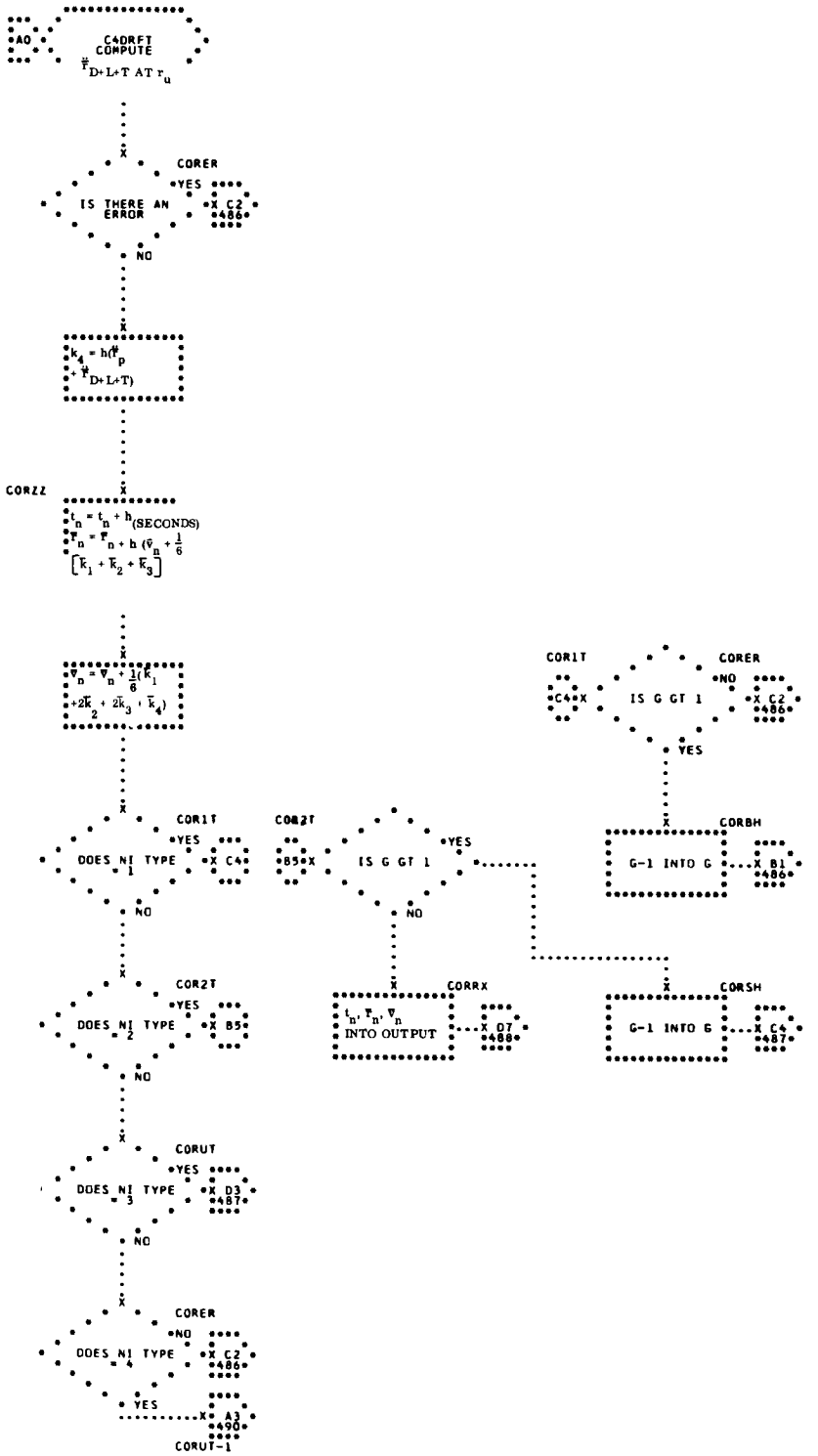
N_b + 1 INTO XR1


A0
H IS GT C_2
NO
YES
COR2C
F=2
YES
NO
CORSI
0 INTO H
X D7
486
F=1
YES
NO
H=0
NO
YES
CORSI
2 INTO F
F_2 INTO T_F
C_2 INTO C_6
X D7
486
F=3
YES
NO
B3
487
CORMS
CORQS
CORBH
$t_L = t_n + Q(t_{n-1} - t_n)$
$r_L = r_n + Q(r_{n-1} - r_n)$
2 INTO F
4 INTO M
$\bar{F}_2$ INTO $\bar{T}_F$
C_2 INTO C_L
$t_{n-1}, \bar{F}_{n-1}, \bar{v}_{n-1}$ INTO $t_n, \bar{F}_n, \bar{v}_n$
B4
X B1
486
CORSI
M-1 INTO M
X D7
486
H IS LT C_1 (IMPACT)
YES
NO
IS e = 0
YES
NO
H INTO C_1
CORIC
CORSI
F=1
YES
X D7
486
NO
F=2
YES
NO
CORLS
CORBH
1 INTO F
4 INTO M
$\bar{F}_1$ INTO $\bar{T}_F$
$t_{n-1}, \bar{F}_{n-1}, \bar{v}_{n-1}$ INTO $t_n, \bar{F}_n, \bar{v}_n$
X B1
486
F=3
NO
YES
X D0
487
X B4
CORQS
$\frac{H - C_1}{H - H_{n-1}}$ INTO 0
$t_L = t_n + Q(t_{n-1} - t_n)$
$\bar{r}_L = \bar{r}_n + Q(\bar{r}_{n-1}$
$r_n)$
C3GSCP
COMPUTE λ
AND φ (AT IMPACT)
CORRX
D7
X
RESTORE XR4
COMER
IS THERE AN ERROR
YES
X C2
486
NO
D8
X
RESTORE OTHER XRS
CORRX
t_1, λ φ
INTO OUTPUT
X D7
EXIT
$T_{5,4}$


CORIF
CORHT
F=0
NO
YES
A0
SEE NOTE
A2
CORIF -
0 INTO f
t_n INTO 2,2
t_n INTO 2,3
C(XR2) INTO R_2
C(XR3) INTO R_3
0 INTO A
0 INTO H_{COUNT}
NOTE
$-\bar{T}_F$ INTO $\bar{T}_F$
s_2 INTO XR2
s_3 INTO XR3
s_R INTO t_n
0, 2 —5,2
INTO $\bar{r}_n$, $\bar{v}_n$
XR2+6 INTO XR2
RESTORE OUTPUT OF C6HPEP ANCHOR VECTOR
03 INTO AC
N_f + 1 INTO XR5
CORHT
CORHG
IS H_n GTH$_{CRIT}$
SET A GT 0
CORHM
IS H_n GTH$_{n-1}$
0 INTO H COUNT
CORDB
IS H_{COUNT}= 0
IS H_{n-1}GT/EH CRIT
1 INTO H_{COUNT}
CORIB
IS H_{COUNT}=1
2 INTO H COUNT
+A INTO AC
B6
-A INTO AC
IS N_f +1 GT 1
B6
N_f - 1 INTO N_f
CORR2
CORRX
R_2 - INTO XR2
R_3 - INTO XR3
t_n - INTO -1,2
t_n - INTO -1,3
D7
488
CORCF
CORII
XR2-6 INTO XR2 XR3-1 INTO XR3
AC
491


CORER
A0
DOES NI TYPE = 4
NO
X C2 486
YES
CORT4
RESTORE XR4 -STEP SIZE (-h) INTO T_F N_b
N_b INTO XR1
2's COMPL {L($\bar{r}$, $\bar{v}$, OUTPUT TABLE) +6 N_b + 2} INTO XR2
C(XR2) INTO S_4
t_n INTO S_A
SET f ≠ 0
N_b + 1 INTO XR1
CORUT-1
A3
C6HPEP COMP $\bar{r}_p$ AND H AT $\bar{r}m$
CORER
IS THERE AN ERROR
YES
X C2 486
NO
IS e = 0
NO
CORS7
SAVE C6HPEP OUTPUT 0 INTO
YES
CORRM
$\bar{r}_n$, $\bar{v}_n$, INTO 0,2 THROUGH 5,2
IS N_b+1 GT 1
NO
YES
IS f = 0
NO
YES
CORF1
0 INTO f t_n
INTO-2,2 C(XR2)
INTO R_4
$-T_F$ INTO T_F
N_b - 1 INTO N_b
CORCC
B8
IS H_n GT H_{STOP}
NO
YES
CO
CORFC
SET AC+
CORR4
D1
S_4 INTO XR2 S_A
INTO t_n 0,2
THROUGH 5,2
INTO r_n, v_n
CORBC
CORII
XR2+6 INTO XR2
X A0 491
CORCC
XR2 + 6 INTO XR2 RESTORE (C6HPEP) OUTPUT OF ANCHOR VECTOR N_f + 1 INTO XR5
X B8
CORFC
CO
IS N_{f_1} +1 GT
NO
YES
SET AC-
N_f - 1 INTO N_f
CORR4
D1
R_4 - INTO XR4 t_n INTO -1,2
CORRX
X D7 488
CORM6
CORII
XR2-6 INTO XR2
X A0 491


CORII
A0
C4DRFT
COMPUTE
$\ddot{\bar{r}}_{D+L+T}$ AT $\bar{r}_n$
CORCD
B0
DOES F=2
YES
NO
CORSS
1 INTO F
4 INTO M
$\bar{F}_1$ INTO $\bar{T}_F$
C_1 INTO C_6
CORXZ
A2
CORER
IS THERE AN ERROR
YES
C2 486
NO
CORTT
2 INTO F
4 INTO M
$\bar{F}_2$ INTO $\bar{T}_F$
C_2 INTO C_2
DOES F=3
$\bar{K}_1 = h(\ddot{\bar{r}}_p + \ddot{\bar{r}}_{D+L+T})$
$\bar{r}_n + h/2\ \bar{v}_n + h/8\ \bar{K}_1$ INTO $\bar{r}_u$
3 INTO F
4 INTO M
F_3 INTO T_F
C_2 INTO C_6
C6HPEP
COMPUTE
$\ddot{\bar{r}}_p$ AND H
AT $\bar{r}_u$
IS NITYPE = 1
IS H GT C_6
DOES F=1
CORPY
$\bar{v}_m + \bar{K}_{1/2}$ INTO $\bar{v}_u$
C4DRFT
COMPUTE
$\ddot{\bar{r}}_{D+L+T}$ AT $\bar{r}_u$
C3
$\bar{K}_2 = h(\ddot{\bar{r}}_p + \ddot{\bar{r}}_{D+L+T})$
$\bar{v}_u + \bar{K}_{2/2}$ INTO $\bar{v}_u$
$\bar{K}_3 = h(\ddot{\bar{r}}_p + \ddot{\bar{r}}_{D+L+T})$
CORGB
$\bar{r}_n + h\,\bar{v}_n + h/2\ \bar{K}_3$ INTO $\bar{R}_u$
D3
CORNM
IS H LT C -
CORMR
$\bar{v}_n + \bar{K}_3$ INTO $\bar{v}_u$
A0 492


A0
C4DRFT
COMPUTE
$\ddot{\bar{r}}_{D+L+T}$ AT r_u
COREER
IS THERE AN ERROR
YES
X C2 486
NO
$k_4 = h(\ddot{\bar{r}}_p + \ddot{\bar{r}}_{D+L+T})$
CORZZ
$t_n = t_n + h$ (SECONDS)
$\bar{r}_n = \bar{r}_n + h(\bar{v}_n + \frac{1}{6}[\bar{k}_1 + \bar{k}_2 + \bar{k}_3])$
$\bar{v}_n = \bar{v}_n + \frac{1}{6}(\bar{k}_1 + 2\bar{k}_2 + 2\bar{k}_3 + \bar{k}_4)$
COR1T
DOES NI TYPE = 1
YES
X C4
NO
COR2T
DOES NI TYPE = 2
YES
X B5
NO
CORUT
DOES NI TYPE = 3
YES
X D3 487
NO
COREER
DOES NI TYPE = 4
NO
X C2 486
YES
X A3 490
CORUT-1
COR2T
B5
IS G GT 1
YES
NO
CORRX
$t_n, \bar{r}_n, \bar{v}_n$ INTO OUTPUT
X D7 488
COR1T
C4
IS G GT 1
COREER
NO
X C2 486
YES
CORBH
G-1 INTO G
X B1 486
CORSH
G-1 INTO G
X C4 487

54. C6HPEP - COMPUTATION OF GEOCENTRIC HEIGHT AND ACCELERATION DUE TO THE POTENTIAL OF THE EARTH

THIS PROGRAM COMPUTES THE PARTIAL DERIVATIVES OF THE POTENTIAL FUNCTION OF THE EARTH WITH RESPECT TO X, Y, Z OF THE GEMINI COORDINATE SYSTEM. THESE PARTIALS ARE THE COMPONENTS OF THE ACCELERATION OF THE SPACECRAFT DUE TO THE POTENTIAL OF THE EARTH. THE GEOCENTRIC HEIGHT OF THE SPACECRAFT IS ALSO COMPUTED. THIS PROGRAM IS A SUBROUTINE OF THE LAUNCH NUMERICAL INTEGRATION PROGRAM CORKNI. (SEE FLOWCHART.)

INPUT

A) A THREE-LOCATION TABLE CONTAINING THE COMPONENTS OF $\overline{r}$ IN FLOATING POINT GEMINI UNITS IS INPUT TO THE PROGRAM.

B) THE LIBRARY SUBROUTINE U1SQRT IS USED.

C) CONSTANTS -

K.INB2 DEC 1.0067383		FLOATING POINT INVERSE OF THE GEMINI SPHEROID SEMI-MINOR AXIS SQUARED IN GEMINI UNITS
KJ.2HC DEC .00162345		FLOATING POINT SECOND HARMONIC COEFFICIENT IN THE EARTH'S POTENTIAL FUNCTION
K001.0 DEC 1.0		FLOATING POINT ONE
K002.0 DEC 2.0		FLOATING POINT TWO
K005.0 DEC 5.0		FLOATING POINT FIVE

OUTPUT

A FOUR-LOCATION TABLE CONTAINING, RESPECTIVELY, THE PARTIAL DERIVATIVES ($\ddot{X}_P$, $\ddot{Y}_P$, $\ddot{Z}_P$) OF THE EARTH'S POTENTIAL FUNCTION AND THE GEOCENTRIC HEIGHT OF THE SPACECRAFT IS OUTPUT FROM THE PROGRAM. ALL OUTPUT IS IN FLOATING POINT GEMINI UNITS.

METHOD

THIS PROGRAM IS AN ADAPTATION OF THE MERCURY ORBIT PHASE PROGRAM N6CPEP. THE MATHEMATICAL METHODS USED ARE THE SAME, BUT THE LOGIC NOT NECESSARY FOR LAUNCH HAS BEEN ELIMINATED IN ORDER TO DECREASE MACHINE TIME. HOWEVER, C6HPEP ALSO COMPUTES THE HEIGHT OF THE SPACECRAFT.

THE FUNCTION OF THE EARTH'S POTENTIAL AT A DISTANCE, r, FROM THE EARTH'S MASS CENTER AND AT A GEOCENTRIC LATITUDE ϕ IS GIVEN BY -

$$V = \frac{\mu}{r}\left[1 + \frac{J}{3}\left(\frac{a}{r}\right)^2 P_2 + \frac{H}{5}\left(\frac{a}{r}\right)^3 P_3 + \frac{K}{35}\left(\frac{a}{r}\right)^4 P_4 + \ldots\right]$$

WHERE

$\mu = 1$ (THE GEMINI UNIT OF TIME WAS CHOSEN SO THAT THIS GEOCENTRIC GRAVITATIONAL CONSTANT = 1.)

$a = 1$ (SEMI-MAJOR AXIS OF THE GEMINI SPHEROID)

$\sin \phi^{\bullet} = \frac{z}{r}$ (GEOCENTRIC LATITUDE)

$P_2 =$ 1-3 SIN (A LEGENDRE POLYNOMIAL)

$t =$ 1623.45 X 10 (SECOND HARMONIC COEFFICIENT)

IN THE GEMINI SYSTEM THE EXPRESSION FOR THE POTENTIAL FUNCTION IS APPROXIMATED BY TRUNCATION AFTER THE SECOND HARMONIC TERM. THE PARTIAL DERIVATIVES OF THE POTENTIAL FUNCTION ARE -

$$\ddot{X}p = \frac{\partial V}{\partial X} = \frac{X}{r^3}\left[\frac{J}{r^2}\left(\frac{5Z^2}{r^2} - 1\right) - 1\right]$$

$$\ddot{Y}p = \frac{\partial V}{\partial Y} = \frac{Y}{r^3}\left[\frac{J}{r^2}\left(\frac{5Z^2}{r^2} - 1\right) - 1\right]$$

$$\ddot{Z}p = \frac{\partial V}{\partial Z} = \frac{Z}{r^3}\left[\frac{J}{r^2}\left(\frac{5Z^2}{r^2} - 3\right) - 1\right]$$

THE GEOCENTRIC HEIGHT OF THE SPACECRAFT IS COMPUTED USING THE EQUATION $h = r - R_e$ WHERE R_e IS THE RADIUS OF THE EARTH AND

$$R_e = \sqrt{\frac{X^2 + Y^2 + Z^2}{X^2 + Y^2 + Z^2/b^2}}$$

(FOR A DERIVATION OF THIS EXPRESSION, REFER TO THE EARTH RADIUS DETERMINATION PROGRAM, BOSCER).

THE GEMINI ORBIT SYSTEM NUMERICAL INTEGRATION PROGRAM, NOCPNI, MAKES USE OF A3GSCP TO CALCULATE THE HEIGHT OF THE SPACECRAFT. HOWEVER, COMPARISON OF A3GSCP HEIGHTS WITH THOSE OF THE ABOVE METHOD SHOW THAT THE DIFFERENCE IN HEIGHT NEVER EXCEEDS SIX FEET FOR AN ORBIT WITH AN APOGEE OF 174 NAUTICAL MILES. THIS MAXIMUM DIFFERENCE OCCURS AT APOGEE WHEN APOGEE IS AT 40° LATITUDE. FOR AN ORBIT WITH AN APOGEE OF 435 NAUTICAL MILES OCCURRING AT 45° , THE HEIGHTS DIFFER BY NOT MORE THAN 15 FEET.

THE METHOD OUTLINED ABOVE RESULTS IN A CONSIDERABLE SAVING OF MACHINE TIME. THEREFORE, IT HAS BEEN ADAPTED IN PLACE OF A3GSCP TO COMPUTE THE HEIGHT OF THE SPACECRAFT.

USAGE

A) CALLING SEQUENCE -

```
TSX     C6HPEP,4
PZE     L(INPUT),,L(OUTPUT)
ERROR   RETURN
NORMAL  RETURN
```

B) STORAGE REQUIRED -

76 LOCATIONS EXCLUDING THOSE REQUIRED BY U1SQRT.

C) ERROR CONDITIONS -

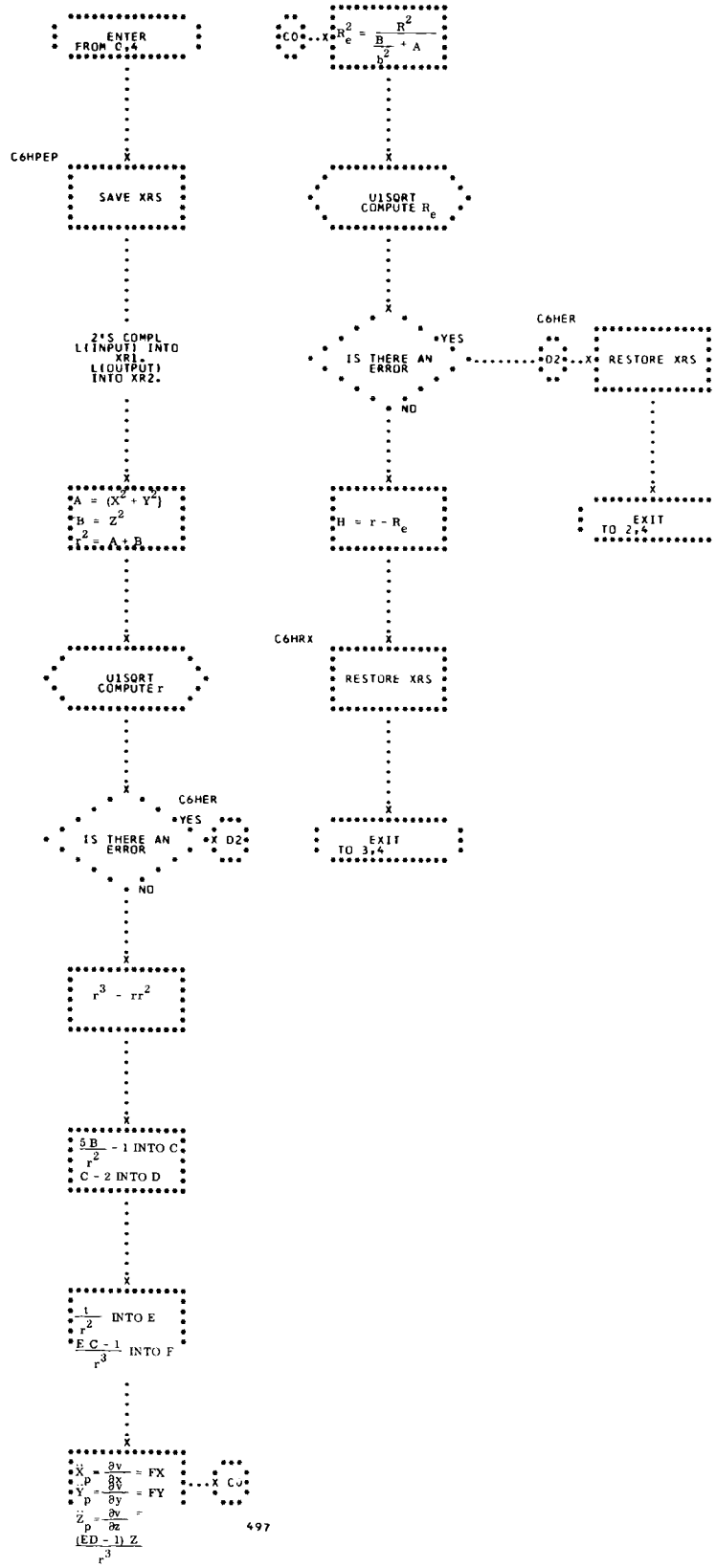
AN ERROR RETURN IS A RESULT OF AN ERROR RETURN FROM U1SQRT.

D) TIME REQUIRED (7094) -

0.778 MILLISECONDS (MAXIMUM) INCLUDING TIME REQUIRED BY U1SQRT.

E) CHECKOUT -

RESULTS OF C6HPEP WERE COMPARED AGAINST THOSE OF N6OPEP AND THESE WERE SATISFACTORY.


ENTER FROM C.4
C6HPEP
SAVE XRS
2'S COMPL L(INPUT) INTO XR1. L(OUTPUT) INTO XR2.
A = (X^2 + Y^2)
B = Z^2
r^2 = A + B
U1SORT COMPUTE r
C6HER
IS THERE AN ERROR
YES
X D2
NO
r^3 - rr^2
5B/r^2 - 1 INTO C
C - 2 INTO D
1/r^2 INTO E
E C - 1/r^3 INTO F
X = ρ ∂v/∂x = FX
Y = ρ ∂v/∂y = FY
Z = ρ ∂v/∂z = (ED - 1) Z/r^3
X CO
CO
R_e^2 = R^2/(B/2 + A)
U1SORT COMPUTE R_e
IS THERE AN ERROR
YES
D2
C6HER
RESTORE XRS
EXIT TO 2,4
NO
H = r - R_e
C6HRX
RESTORE XRS
EXIT TO 3,4

55. C4DRFT - COMPUTATION OF ACCELERATION DUE TO DRAG, LIFT, AND THRUST

THIS PROGRAM COMPUTES THE COMBINED ACCELERATION OF THE SPACECRAFT DUE TO DRAG, LIFT, AND THRUST. THIS ACCELERATION IS COMPUTED ALONG THE AXES OF THE GEMINI COORDINATE SYSTEM. IT IS A SUBROUTINE OF THE LAUNCH NUMERICAL INTEGRATION PROGRAM, CORKNI. (SEE FLOWCHART.)

INPUT

A) CALLING SEQUENCE REQUIREMENTS -

L(INPUT) - ADDRESS OF A SIX-LOCATION TABLE CONTAINING THE COMPONENTS OF THE POSITION AND VELOCITY IN FLOATING POINT GEMINI UNITS

L(H) - ADDRESS OF A CELL CONTAINING THE HEIGHT OF THE SPACECRAFT IN FLOATING POINT GEMINI UNITS

L(B) - ADDRESS OF A CELL CONTAINING THE BANK ANGLE IN FLOATING POINT RADIANS

$L(T_P)$ - ADDRESS OF A CELL CONTAINING THE THRUST COMPUTATION INDICATOR IN FIXED POINT

$T_P = 0$ DO NOT COMPUTE THRUST

$T_P \neq 0$ COMPUTE THRUST

$L(RHO_I)$ - ADDRESS OF A CELL CONTAINING THE DENSITY COMPUTATION INDICATOR IN FIXED POINT

$RHO_I = 0$ DO NOT COMPUTE DENSITY

$RHO_I \neq 0$ COMPUTE DENSITY

B) SUBROUTINES C4PDEN AND C4COEF AND LIBRARY ROUTINES U3VMAG AND U1SQRT ARE USED.

CONSTANTS -

```
KMOOOO DEC -0          MINUS ZERO
KOOOOO DEC  0          ZERO
```

K0C1.0	DEC	1.0	FLOATING POINT ONE
K00003	DEC	3	FIXED POINT THREE
K.OMEG	DEC	.058833780	FLOATING POINT ROTATIONAL VELOCITY OF GEMINI SPHEROID IN RADIANS/GEMINI TIME UNIT
K.MOMG	DEC	-.058833780	FLOATING POINT MINUS ROTATIONAL VELOCITY OF GEMINI SPHEROID IN RADIANS/GEMINI TIME UNIT
KO.RAD	DEC	6378165.0	FLOATING POINT EQUATORIAL RADIUS OF GEMINI SPHEROID IN METERS

D) TABLES -

T.HLFN	A ONE-LOCATION TABLE CONTAINING MINUS ONE-HALF THE RATIO OF THE AREA OF THE SPACECRAFT TO THE MASS OF THE SPACECRAFT IN FLOATING POINT GEMINI UNITS
TC.CNA	A ONE-LOCATION TABLE CONTAINING THE ACCELERATION DUE TO THE NOMINAL ONBOARD PROPULSION IN FLOATING POINT GEMINI ACCELERATION UNITS
TCCSBA	A ONE-LOCATION TABLE CONTAINING THE FLOATING POINT COSINE OF THE BANK ANGLE
TCSNBA	A ONE-LOCATION TABLE CONTAINING THE FLOATING POINT SINE OF THE BANK ANGLE
TCPDEN	A ONE-LOCATION TABLE TO BE USED AS THE OUTPUT TABLE BY C4PDEN WHICH CONTAINS THE ATMOSPHERIC DENSITY IN FLOATING POINT GEMINI UNITS

OUTPUT

A THREE-LOCATION TABLE CONTAINING THE COMPONENTS OF THE ACCELERATION VECTOR DUE TO DRAG, LIFT, AND THRUST IN FLOATING POINT GEMINI UNITS IS OUTPUT FROM THE PROGRAM.

METHOD

THIS PROGRAM IS IN PART AN ADAPTATION OF THE ORBIT PHASE PROGRAM, N4DRAG. LOGIC NOT NECESSARY FOR LAUNCH IS ELIMINATED TO DECREASE MACHINE TIME. HOWEVER, THE MATHEMATICAL METHODS USED ARE THE SAME AS IN N4DRAG. IN ADDITION, COMPUTATIONS REQUIRED FOR LIFT AND THRUST ARE INCLUDED IN THE PROGRAM.

THE COMPONENTS OF THE ACCELERATION VECTOR DUE TO DRAG AND LIFT ARE COMPUTED USING THE FOLLOWING EQUATIONS

$$\ddot{\bar{r}}_D = -\frac{1}{2}\left(C_D \rho N v_R \bar{v}_R\right)$$

$$\ddot{\bar{r}}_L = \frac{1}{2}\left[C_L \rho N v_R^2 \left(\bar{p}^* \cos B - \bar{h}^* \sin B\right)\right]$$

WHERE

$1/2 C_D \rho N v_R^2$ IS THE MAGNITUDE OF THE ACCELERATION DUE TO THE DRAG FORCE[1]

$\frac{-\bar{v}_R}{v_R}$ IS THE DIRECTION OF THE DRAG ACCELERATION

$1/2\left(C_L \rho N_v^2 R\right)$ IS THE MAGNITUDE OF THE ACCELERATION DUE TO THE LIFTING FORCE[2]

$\left(\bar{p}^* \cos B - \bar{h}^* \sin B\right)$ IS THE DIRECTION OF THE LIFTING ACCELERATION[3]

1 AN INTRODUCTION TO ASTRODYNAMICS, BAKER AND MAKEMSON, ACADEMIC PRESS, INC.,NEW YORK,N.Y., 1960, PP.207-208.

2 DYNAMICS OF ATMOSPHERIC ENTRY, R.L.DUNCAN, MCGRAW-HILL BOOK COMPANY,INC., NEW YORK,N.Y., 1962, P.84.

3 BAKER AND MAKEMSON, P.167.

C_D AND C_L ARE THE COEFFICIENTS OF DRAG AND LIFT RESPECTIVELY (COMPUTED IN C4COEF)

ρ IS THE ATMOSPHERIC DENSITY (COMPUTED IN C4PDEN)

v_R IS THE MAGNITUDE OF THE RELATIVE VELOCITY VECTOR

B IS THE BANK ANGLE OF THE SPACECRAFT

$$\bar{p}^* = \frac{\bar{p}}{p}, \text{ where } \bar{p} = \bar{v}_R \times (\bar{r} \times \bar{v}_R)$$

$$\bar{h}^* = \frac{\bar{h}}{h}, \text{ where } \bar{h} = \bar{r} \times \bar{v}_R$$

N IS THE RATIO OF THE AREA OF THE SPACECRAFT TO THE MASS OF THE SPACECRAFT.

ALL FACTORS ARE IN FLOATING POINT GEMINI UNITS WITH THE EXCEPTION OF C_D AND C_L WHICH ARE DIMENSIONLESS, AND B WHICH IS IN RADIANS.

THE HEIGHT OF THE SPACECRAFT IN GEOPOTENTIAL METERS, H*, IS COMPUTED USING THE FOLLOWING EQUATION -

$$H^* = \frac{k_0 H}{1 + H}$$

WHERE H IS THE HEIGHT IN GEMINI UNITS AND K_0 IS THE RADIUS OF THE GEMINI SPHEROID IN METERS.[4]

GIVEN THE COMPONENTS OF THE POSITION AND VELOCITY VECTORS, THE COMPONENTS OF THE RELATIVE VELOCITY VECTOR $\bar{v}_R$ ARE -

$$\bar{v}_R = \begin{pmatrix} \dot{X} + Y\omega \\ \dot{Y} - X\omega \\ \dot{Z} \end{pmatrix}$$

4 BAKER AND MAKEMSON, P.104.

WHERE ω IS THE ROTATIONAL VELOCITY OF THE GEMINI SPHEROID IN RADIANS/GEMINI TIME UNIT.

ACCELERATION DUE TO THRUST IS COMPUTED USING

$$\ddot{\bar{r}}_T = \frac{A}{\cos\gamma}\ \frac{\bar{v}}{|v|} - \frac{A\sin\gamma}{\cos\gamma}\ \frac{\bar{r}}{|r|}$$

WHERE

A IS THE NOMINAL THRUSTING ACCELERATION IN GEMINI ACCELERATION UNITS

γ IS THE FLIGHT PATH ANGLE IN RADIANS

$\bar{v}$ IS THE INERTIAL VELOCITY VECTOR

$|v|$ IS THE MAGNITUDE OF THE INERTIAL VELOCITY VECTOR

$\bar{r}$ IS THE INERTIAL POSITION VECTOR

$|r|$ IS THE MAGNITUDE OF THE INERTIAL POSITION VECTOR

COMPUTATION OF THRUST IN THIS WAY ASSUMES THAT THE PITCH, ROLL, AND YAW ANGLES ARE ZERO.

USAGE

A) CALLING SEQUENCE -

```
TSX     C4DRFT,4
PZE     L(INPUT,,L(OUTPUT)
PZE     L(H),,L(B)
PZE     L(T_p),,L(RHO_I)
ERROR   RETURN
NORMAL  RETURN
```

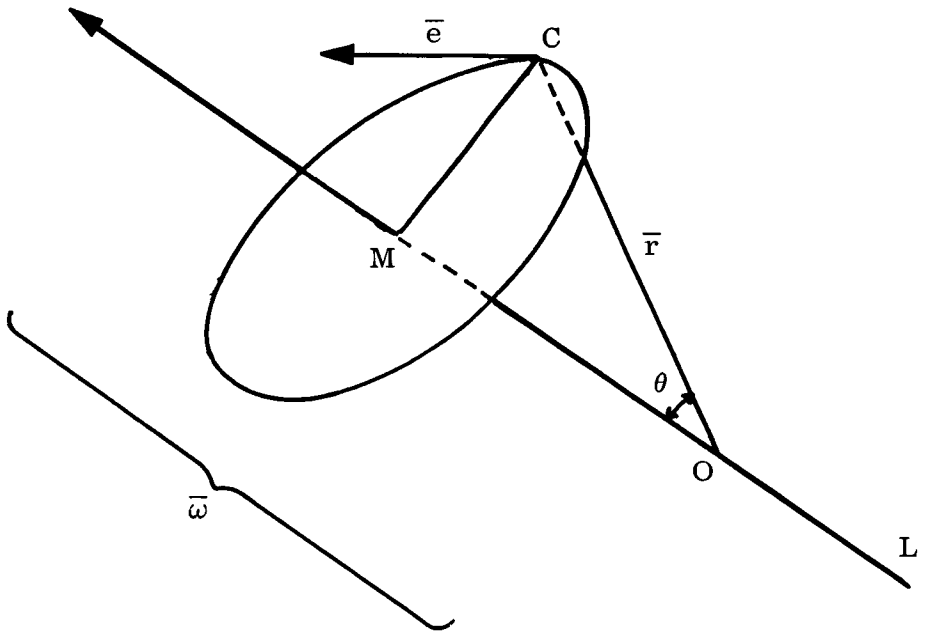
(THE RANGE OF ANY FIXED BANK ANGLE, B, IS SUCH THAT $+0 \leq B \leq 2\pi$, WHERE B IS EXPRESSED IN RADIANS. A NEGATIVE BANK ANGLE REPRESENTS CONSTANT ROLL RATE AND HENCE ZERO LIFT.)

B) STORAGE REQUIRED - 281 LOCATIONS EXCLUDING THAT REQUIRED BY EXTERNAL CONSTANTS AND SUBROUTINES.

C) ERROR CONDITIONS - AN ERROR RETURN RESULTS FROM AN ERROR RETURN FROM A SUBROUTINE.

D) TIME REQUIRED (7094) - 4.294 MILLISECONDS (MAXIMUM) INCLUDING THE MAXIMUM TIME FOR ALL SUBROUTINES.

DERIVATION OF VELOCITY VECTOR DUE TO THE EARTH'S ROTATION



LET L BE A LINE COINCIDENT WITH THE EARTH'S ROTATIONAL AXIS. LET $\bar{\omega}$ BE A VECTOR REPRESENTING THE ANGULAR VELOCITY IMPARTED TO THE SPACECRAFT DUE TO THE ROTATION OF THE EARTH. THUS, $\bar{\omega}$ LIES ALONG L AND HAS LENGTH $|\bar{\omega}|$, THE CONSTANT VELOCITY OF THE ROTATION OF THE EARTH IN RADIANS/GEMINI TIME UNIT. IT HAS AS ITS DIRECTION THAT IN WHICH A RIGHT-HANDED SCREW WOULD ADVANCE UNDER THE GIVEN ROTATION, I.E., IT PASSES THROUGH THE NORTH POLE.

O IS THE ORIGIN OF THE GEMINI COORDINATE SYSTEM AND C IS THE POSITION IN SPACE OCCUPIED BY THE SPACECRAFT. LET $\bar{r}$ REPRESENT THE POSITION VECTOR OF THE SPACECRAFT IN THE GEMINI COORDINATE SYSTEM.

THE ROTATION ABOUT L IMPARTS TO C A LINEAR VELOCITY WHICH IS DENOTED BY THE VECTOR $\bar{e}$. SINCE THAT PART OF THE VELOCITY OF C WHICH IS DUE TO THE ROATION OF THE EARTH DESCRIBES A CIRCLE WITH CENTER M (A POINT ON L), e IS PERPENDICULAR TO $\overline{MC}$ AND

(1) $|\bar{e}| = |\bar{\omega}| \, \overline{MC}$

LET θ BE THE ANGLE BETWEEN $\bar{\omega}$ AND $\bar{r}$. THEN

(2) $\overline{MC} = |\bar{r}| \sin \theta$

SUBSTITUTING (2) INTO (1)

(3) $|\bar{e}| = |\bar{\omega}| \, |\bar{r}| \sin \theta$

HOWEVER, e IS PERPENDICULAR TO $\bar{\omega}$ AND $\bar{r}$, AND HAS AS ITS POSITIVE DIRECTION THAT IN WHICH A RIGHT HANDED SCREW WOULD ADVANCE UNDER A ROTATION FROM $\bar{\omega}$ TO $\bar{r}$.

THEREFORE,

(4) $\bar{e} = \bar{\omega} \times \bar{r}$

HENCE,

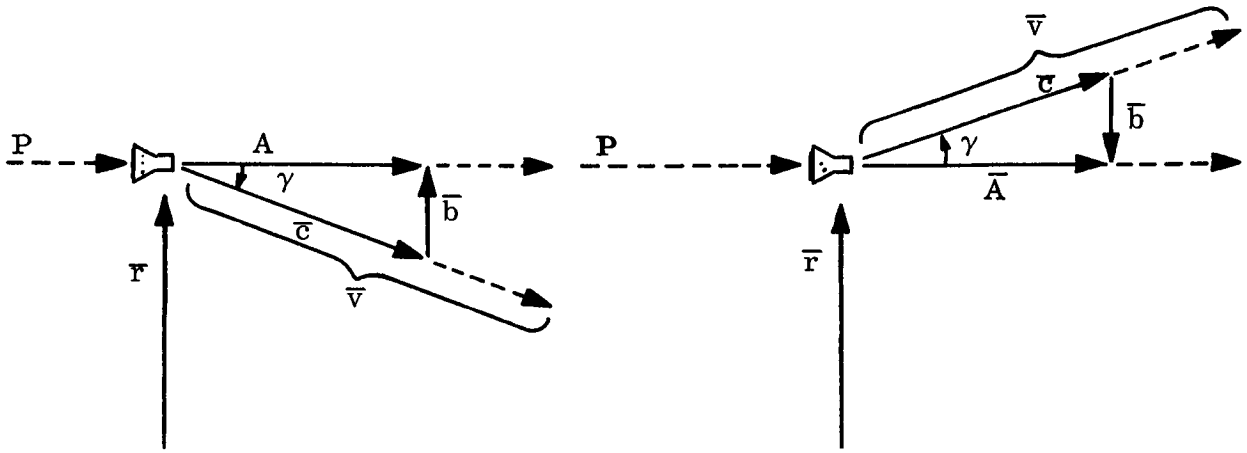
$$\bar{e} = \begin{pmatrix} \dot{X}_e \\ \dot{Y}_e \\ \dot{Z}_e \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ |\bar{\omega}| \end{pmatrix} \times \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{vmatrix} \bar{i} & \bar{j} & \bar{k} \\ 0 & 0 & |\bar{\omega}| \\ X & Y & Z \end{vmatrix} = \begin{pmatrix} -|\bar{\omega}| Y \\ |\bar{\omega}| X \\ 0 \end{pmatrix}$$

THUS, TO REMOVE THE PORTION OF A GEMINI SYSTEM INERTIAL VELOCITY VECTOR WHICH IS DUE TO THE ROTATION OF THE

EARTH, WE SUBTRACT FROM IT THE LINEAR VELOCITY $\bar{e}$ AND

$$\bar{v}_R = \begin{pmatrix} \dot{X} + Y\omega \\ \dot{Y} - X\omega \\ \dot{Z} \end{pmatrix}$$

DERIVATION OF EQUATION FOR ACCELERATION DUE TO SPACECRAFT THRUST



Negative Flight Path Angle

Positive Flight Path Angle

LET P BE THE LONGITUDINAL AXIS, $\bar{r}$ THE POSITION VECTOR, $\bar{v}$ THE VELOCITY VECTOR, AND γ THE FLIGHT PATH ANGLE OF THE SPACECRAFT. LET A BE THE MAGNITUDE OF THE TOTAL THRUST OF THE SPACECRAFT DUE TO ON-BOARD PROPULSION. THIS THRUST LIES ALONG P AND IS DIRECTED AS SHOWN. (FOR THESE COMPUTAIONS IT IS ASSUMED THAT THE PITCH, ROLL, AND YAW ANGLES OF THE SPACECRAFT ARE ZERO.)

LET $\ddot{\bar{r}}_T$ DENOTE THE COMPONENTS OF THIS THRUST ACCELERATION (A) ALONG THE AXIS OF THE GEMINI COORDINATE SYSTEM. BY VECTOR ADDITION

(1) $\ddot{\bar{r}}_T = \bar{c} + \bar{b}$

HOWEVER,

(2) $c = \dfrac{A}{\cos\gamma}$ and

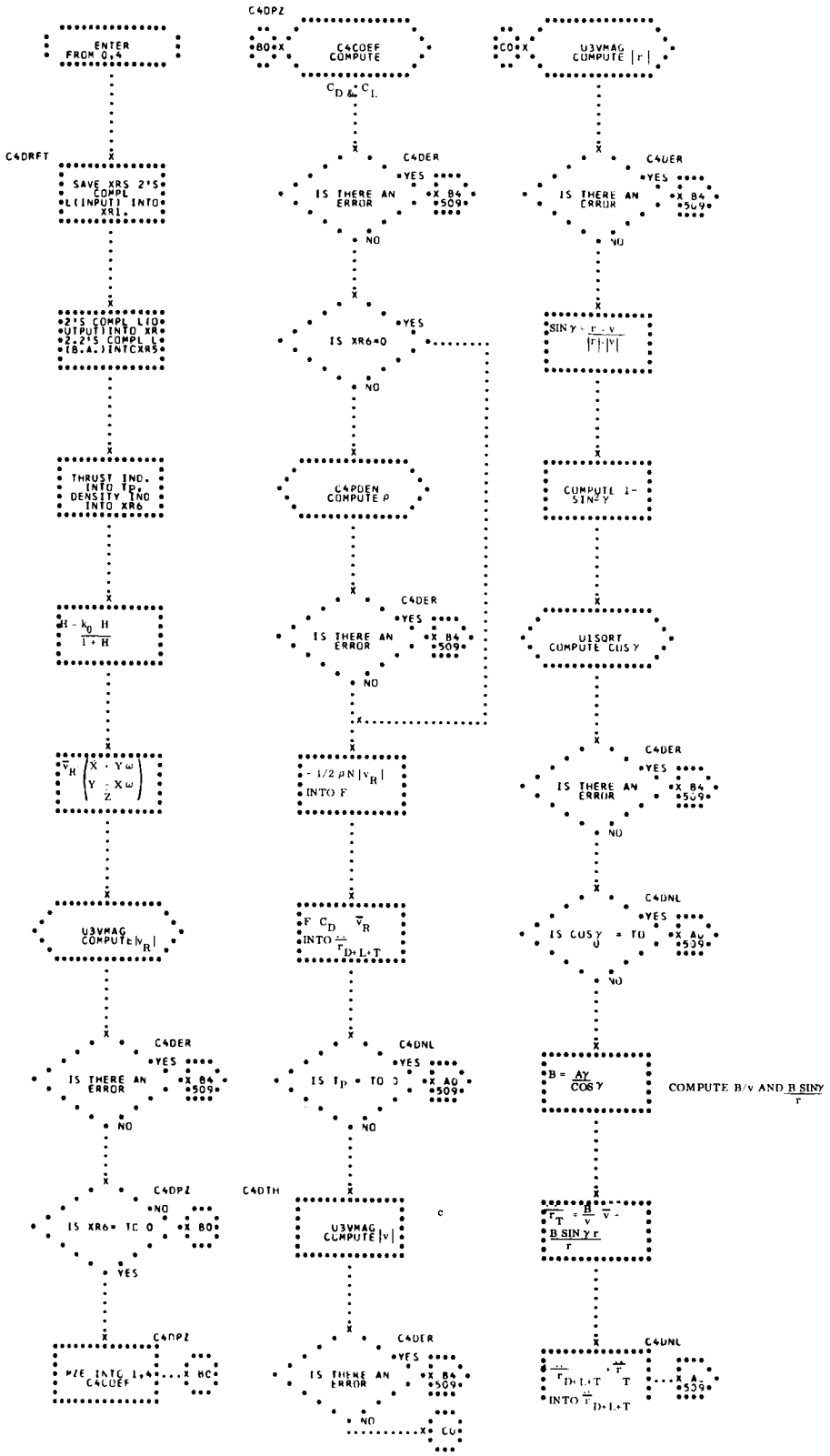
$b = c \sin\gamma = \dfrac{A \sin\gamma}{\cos\gamma}$

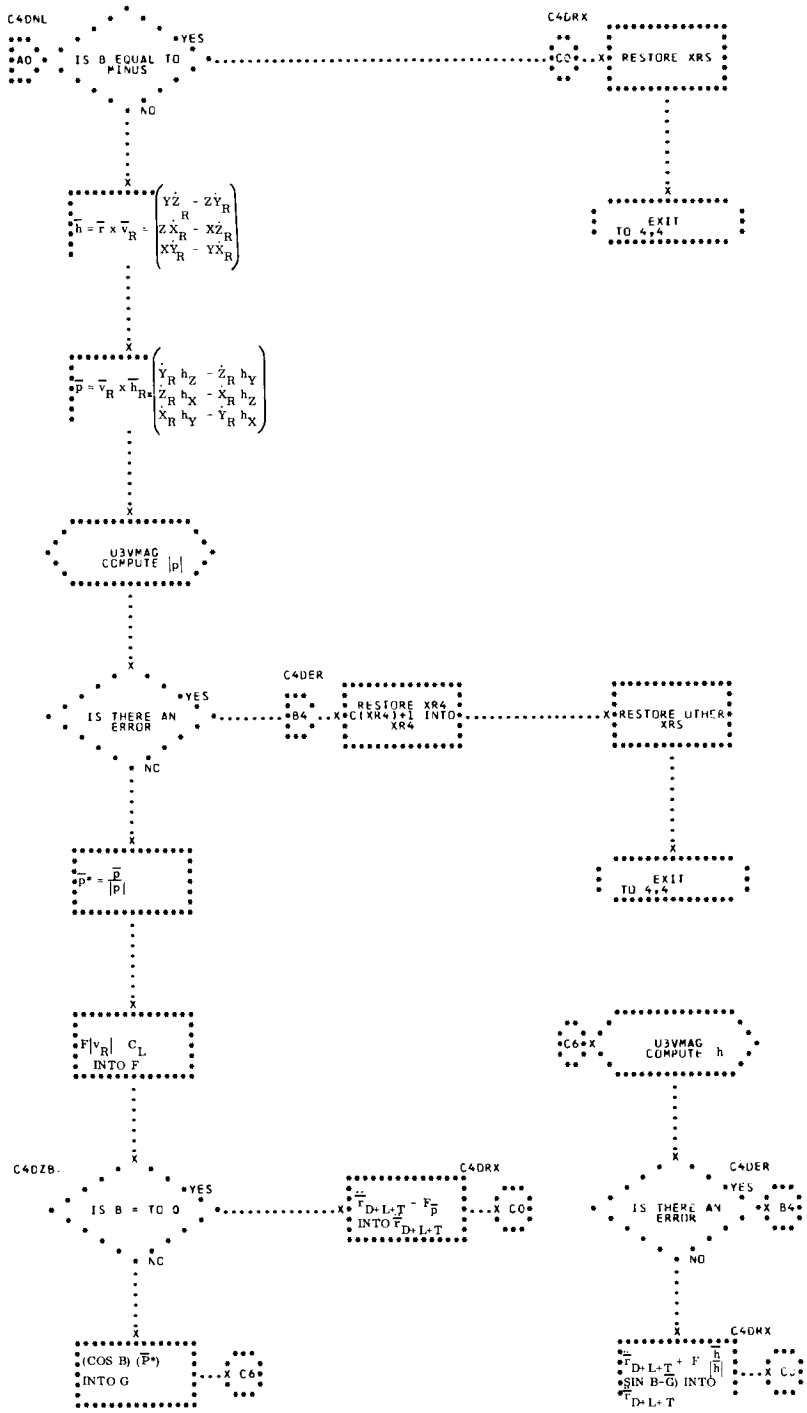
THESE TWC MAGNITUDES ARE DIRECTED AS FOLLOWS -

(3) C IS DIRECTED POSITIVELY ALONG $\overline{v}$ OR HAS DIRECTION $\frac{\overline{v}}{|v|}$ b IS DIRECTED NEGATIVELY ALONG $\overline{r}$ OR HAS DIRECTION $-\frac{\overline{r}}{|r|}$

SUBSTITUTING (2) AND (3) INTO (1),

$$\ddot{\overline{r}}_T = \frac{A}{\cos\gamma} \left|\frac{\overline{v}}{v}\right| - \frac{A \sin\gamma}{\cos\gamma} \left|\frac{\overline{r}}{r}\right|$$


ENTER FROM 0,4
C4DRFT
SAVE XRS 2'S COMPL (INPUT) INTO XR1.
2'S COMPL L(O UTPUT) INTO XR. 2.2'S COMPL L (B.A.) INTO XR5
THRUST IND. INTO TP. DENSITY IND INTO XR6
H - k0 H / 1 + H
V_R (X + Yω, Y - Xω, Z)
U3VMAG COMPUTE |V_R|
C4DER
IS THERE AN ERROR
YES
X 84 509
NO
IS XR6= TO 0
X 80
C4DPZ
M/E INTO L,4 C4COEF
X BC
C4DPZ
80 X
C4COEF COMPUTE
C_D & C_L
IS XR6=0
C4PDEN COMPUTE ρ
- 1/2 ρ N |V_R| INTO F
F C_D V_R INTO a/r_D+L+T
C4DNL
IS T_P = TO 0
X A0
C4DTH
U3VMAG COMPUTE |V|
C
C4DER
IS THERE AN ERROR
X CC
C0 X
U3VMAG COMPUTE |r|
SIN γ = r · v / |r||v|
COMPUTE 1 - SIN²γ
U1SQRT COMPUTE COS γ
IS COS γ = TO 0
B = Aγ / COS γ
COMPUTE B/v AND B SINγ / r
a/r_T = B/v - B SIN γ r / r
a/r_D+L+T + a/r_T T INTO a/T_D+L+T
X A6


C4DNL
A0
IS B EQUAL TO MINUS
YES
NO
C4DRX
C0
RESTORE XRS
EXIT TO 4,4
$\overline{h} = \overline{r} \times \overline{v}_R = \begin{pmatrix} Y\dot{Z}_R - Z\dot{Y}_R \\ Z\dot{X}_R - X\dot{Z}_R \\ X\dot{Y}_R - Y\dot{X}_R \end{pmatrix}$
$\overline{p} = \overline{v}_R \times \overline{h}_R = \begin{pmatrix} \dot{Y}_R h_Z - \dot{Z}_R h_Y \\ \dot{Z}_R h_X - \dot{X}_R h_Z \\ \dot{X}_R h_Y - \dot{Y}_R h_X \end{pmatrix}$
U3VMAG COMPUTE |p|
IS THERE AN ERROR
YES
NO
C4DER
B4
RESTORE XR4 (C(XR4)+1 INTO XR4
RESTORE OTHER XRS
EXIT TO 4,4
$\overline{p}^* = \frac{\overline{p}}{|p|}$
F|v_R| C_L INTO F
C6
U3VMAG COMPUTE h
C4DZB
IS B = TO 0
YES
NO
C4DRX
C0
$\ddot{\overline{r}}_{D+L+T} - F\overline{p}$ INTO $\overline{r}_{D+L+T}$
C4DER
IS THERE AN ERROR
YES
B4
NO
C4DRX
C0
(COS B) ($\overline{P}^*$) INTO G
C6
$\ddot{\overline{r}}_{D+L+T} + F\frac{\overline{h}}{|h|}$ (SIN B-G) INTO $\overline{r}_{D+L+T}$

56. C4PDEN - COMPUTATION OF ATMOSPHERIC DENSITY

THIS PROGRAM COMPUTES THE DENSITY OF THE ATMOSPHERE CORRESPONDING TO A GIVEN ALTITUDE. IT IS A SUBROUTINE OF C4DRFT, THE PROGRAM USED TO COMPUTE ACCELERATION DUE TO DRAG, LIFT, AND THRUST, WHICH IS IN TURN A SUBROUTINE OF THE LAUNCH NUMERICAL INTEGRATION PROGRAM, CORKNI. (SEE FLOWCHART.)

INPUT

A) A ONE-LOCATION TABLE CONTAINING THE HEIGHT OF THE SPACECRAFT IN FLOATING POINT GEOPOTENTIAL METERS IS INPUT TO THE PROGRAM.

B) THE LIBRARY SUBROUTINES U1EXPE AND U1LOGE ARE USED.

C) CONSTANTS -

K001.0 DEC	1.0	FLOATING POINT ONE
K700K. DEC	700000.	FLOATING POINT SEVEN HUNDRED THOUSAND
K04.M5 DEC	.000043425951	CONVERSION FACTOR KG/M^3 TO GEMINI DENSITY UNITS (GDU)

OUTPUT

A ONE-LOCATION TABLE CONTAINING THE DENSITY OF THE ATMOSPHERE IN FLOATING POINT GEMINI UNITS IS OUTPUT FROM THE PROGRAM.

METHOD

GIVEN AN H* (IN GEOPOTENTIAL METERS) THE FOLLOWING COMPUTATIONS ARE MADE, WHERE $-\rho$ IS THE ATMOSPHERIC DENSITY IN KG/M^3. (CF. AN INTRODUCTION TO ASTRODYNAMICS, BAKER AND MAKEMSON, ACADEMIC PRESS, INC., NEW YORK, 1960, PP. 103-107.)

A) IF H* $\geq$ 700,000 GEOPOTENTIAL METERS, $\rho = 0$.

B) IF H* < 700,000 GEOPOTENTIAL METERS, THEN

$$\ln \rho = \ln \rho_b - \left[1 + \frac{m_o g_o^*}{R^* L_m}\right] \cdot \ln\left[1 + \frac{L_m}{(T_m)_b}(H^* - H_b^*)\right]$$

WHERE

=28.966, THE MOLECULAR WEIGHT OF AIR AT SEA LEVEL

= 9.80665 $M^2/(SEC^2 M^*)$ IS THE ACCELERATION DUE TO GRAVITY AT HEIGHT H*

R^* = 8.31439 . 10^3 JOULES/^oK KG

H_b IS A BASE HEIGHT - ONE OF AN ELEVEN-LOCATION TABLE - EXPRESSED IN GEOPOTENTIAL METERS.

$(T_m)_b$ IS A TEMPERATURE - ONE OF AN ELEVEN-LOCATION TABLE - EXPRESSED IN ^oK AT THE VARIOUS BASE HEIGHTS.

In ρ_b IS A NATURAL LOGARITHM - ONE OF AN ELEVEN-LOCATION TABLE - OF THE DENSITY AT THE VARIOUS BASE HEIGHTS.

$(L_m)_b = \frac{dT_m}{dH^*}$ IS A TEMPERATURE GRADIENT - ONE OF AN ELEVEN-LOCATION TABLE - FOR REGIONS BETWEEN THE VARIOUS BASE HEIGHTS.

IN SOME ISOTHERMAL REGIONS, $L_m = 0$. IN THESE REGIONS

$$\ln \rho = \ln \rho_b - \frac{m_o g^*_o}{R^*} \left[\frac{H^* - H^*_b}{(T_m)_b} \right]$$

$$\rho \text{ (in kg/m}^3) = e^{\ln \rho}$$

$$\rho \text{ (in GDU)} = 4.3425951 \times 10^{-5} \rho \text{ (in kg/m}^3)$$

THE UNIT OF DENSITY FOR THE GEMINI SYSTEM IS DEFINED TO BE (THE MASS OF THE EARTH)/(THE EQUATORIAL RADIUS OF THE EARTH). THE MASS OF THE EARTH IS TAKEN AS 5.975 X 10^{24} KILOGRAMS AND THE EQUATORIAL DISTANCE AS 6,378,165 METERS.

HENCE, ρ (GDU) = 4.3425951 X 10^{-5} ρ (KG/M^3).

THE TABLES USED WERE OBTAINED FROM DATA TAKEN FROM ARDC MODEL 1959 ATMOSPHERE.

USAGE

A) CALLING SEQUENCE -

```
TSX       C4PDEN,4
PZE       L(INPUT),,L(OUTPUT)
ERROR     RETURN
NORMAL    RETURN
```

B) STORAGE REQUIRED -

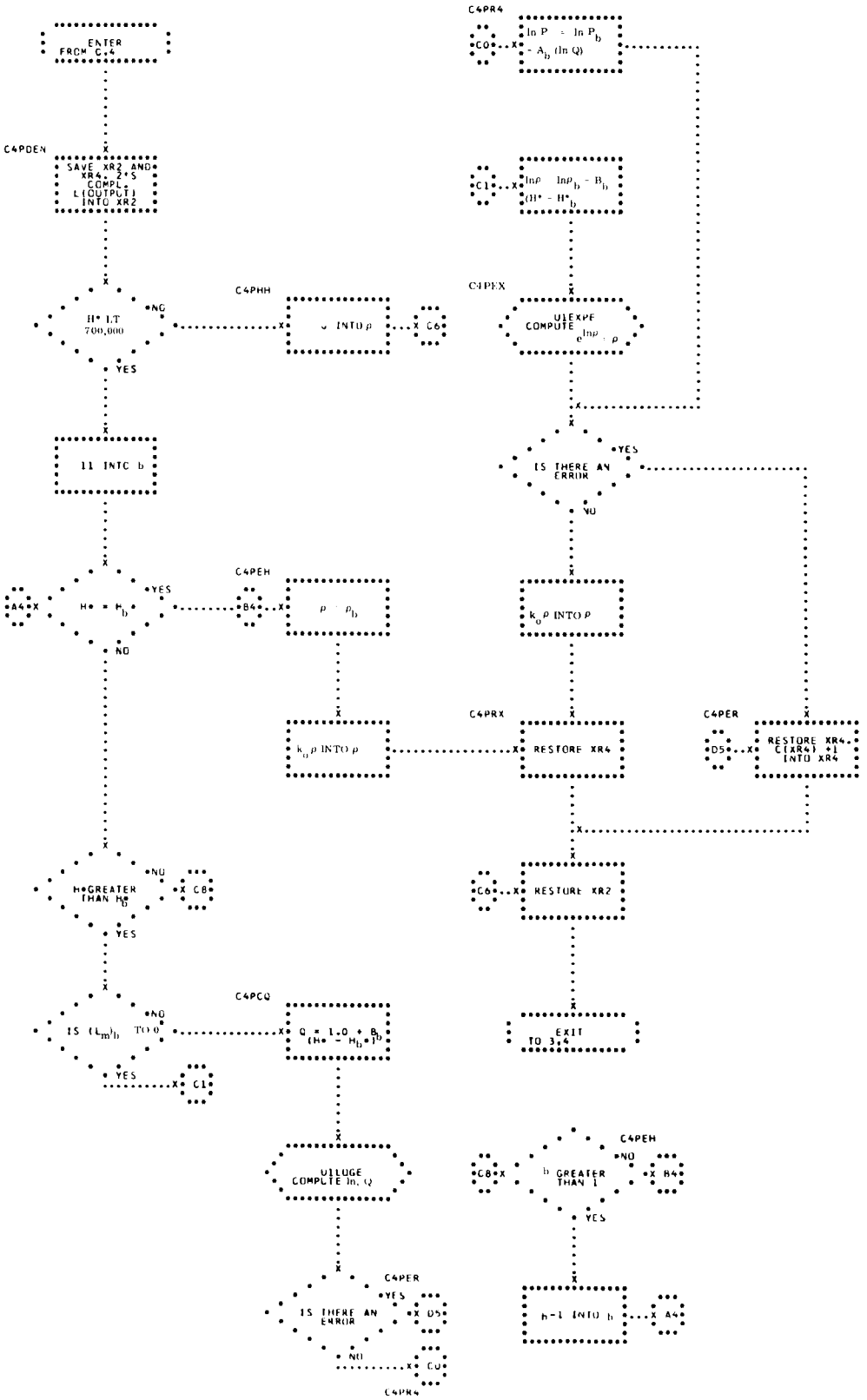
115 DECIMAL LOCATIONS EXCLUDING STORAGE REQUIRED BY SUBROUTINES AND EXTERNAL CONSTANTS.

C) TIME REQUIRED (7094) -

0.520 MILLISECONDS (MAXIMUM) INCLUDING 0.128 MILLISECONDS FOR U1LOGE AND 0.200 MILLISECONDS FOR U1EXPE.

D) ERROR CONDITIONS -
AN ERROR RETURN FROM C4PDEN IMPLIES AN ERROR RETURN FROM A LIBRARY SUBROUTINE.

E) CHECKOUT -
RESULTS FROM C4PDEN WERE COMPARED WITH THOSE OF A DESK CALCULATOR. THESE WERE SATISFACTORY.


ENTER FROM C.4
C4PDEN
SAVE XR2 AND XR4. 2'S COMPL. L(OUTPUT) INTO XR2
H* LT 700,000
NO
YES
C4PHH
INTO ρ
C6
11 INTO b
A4
H* = H_b
YES
NO
C4PEH
B4
ρ = ρ_b
k_o ρ INTO ρ
H* GREATER THAN H_b
NO
YES
C8
IS (L_m)_b TO 0
NO
YES
C1
C4PCQ
Q = 1.0 + B_b (H* - H_b*)^b
U1LOGE COMPUTE ln Q
C4PER
IS THERE AN ERROR
YES
NO
D5
C0
C4PR4
C4PR4
C0
ln P = ln P_b - A_b (ln Q)
C1
lnρ = lnρ_b - B_b (H* - H*_b)
C4PEX
U1EXPF COMPUTE e^lnρ = ρ
IS THERE AN ERROR
YES
NO
k_o ρ INTO ρ
C4PRX
RESTORE XR4
C4PER
D5
RESTORE XR4. C(XR4) +1 INTO XR4
C6
RESTORE XR2
EXIT TO 3,4
C4PEH
C8
b GREATER THAN 1
NO
YES
B4
b-1 INTO b
A4

57. C4COEF - COMPUTATION OF COEFFICIENTS OF DRAG AND LIFT

THIS PROGRAM COMPUTES THE COEFFICIENTS OF DRAG AND OF LIFT CORRESPONDING TO A GIVEN ALTITUDE AND VELOCITY OF THE SPACECRAFT. IT IS A SUBROUTINE OF C4DRFT, THE PROGRAM USED TC COMPUTE THE TOTAL ACCELERATION DUE TO DRAG, LIFT, AND THRUST, AND WHICH IS IN TURN A SUBROUTINE OF THE LAUNCH NUMERICAL INTEGRATION PROGRAM, CORKNI. (SEE FLOWCHART)

INPUT

A) A TWO-LOCATION TABLE CONTAINING THE HEIGHT OF THE SPACECRAFT AND THE MAGNITUDE OF THE RELATIVE VELOCITY VECTOR OF THE SPACECRAFT IS INPUT TO THE PROGRAM. THE HEIGHT IS IN FLOATING POINT GEOPOTENTIAL METERS AND THE VELOCITY IS IN FLOATING POINT GEMINI VELOCITY UNITS.

B) THE LIBRARY SUBROUTINE U1SQRT IS USED.

C) CONSTANTS AND TABLES -

K.CTMN	DEC	394.355200	CONVERTS RELATIVE VELOCITY - (GEMINI VELOCITY UNITS)/(SQUARE ROOT OF THE MOLECULAR SCALE TEMPERATURE) - TO MACH NUMBER
KSS.GU	DEC	30.513261	1/(850 FEET/ SECOND IN GEMINI VELOCITY UNITS)

KGM3.5	DEC	90147.607	300,000 FEET IN GEOPOTENTIAL METERS
TCSRTM	A ONE-LOCATION TABLE CONTAINING THE SQUARE ROOT OF T_M (THE MOLECULAR SCALE TEMPERATURE) IN FLOATING POINT ^oK		

OUTPUT

A TWO-LOCATION TABLE CONTAINING THE COEFFICIENTS OF DRAG (C_D) AND OF LIFT (C_L) IS OUTPUT FROM THE PROGRAM. THE COEFFICIENTS ARE IN DIMENSIONLESS FLOATING POINT FORMAT.

METHOD

GIVEN AN ALTITUDE H_1^* IN GEOPOTENTIAL METERS AND THE MAGNITUDE OF THE RELATIVE VELOCITY v_R, MACH NUMBER IS COMPUTED IN THE FOLLOWING WAY -

A) IF H_1^* 90147.607 GEOPOTENTIAL METERS (300,000 FEET),
MACH NUMBER = $\frac{v_R}{S}$ WHERE S IS THE SPEED OF SOUND ABOVE 300,000 FEET.
S IS EQUAL TO 850 FEET/SECOND OR 0.032772633 GEMINI VELOCITY UNITS.

B) IF $H_1^* \leq$ 90147.607 GEOPOTENTIAL METERS, THE FOLLOWING COMPUTATIONS ARE MADE -
$T_m = (T_m)_b + (L_m)_b \left(H_1^* - H_b^*\right)$

WHERE

T_m IS THE MOLECULAR-SCALE TEMPERATURE IN ^oK AT ALTITUDE H_1^*.
$(L_m)_b = \frac{dT_m}{dH^*}$, THE GRADIENT OF A MOLECULAR SCALE TEMPERATURE IN ^oK/GEOPOTENTIAL METER. (THIS IS

CONSTANT BETWEEN BASE ALTITUDES.)

H_b* IS ONE OF EIGHT BASE ALTITUDES IN GEOPOTENTIAL METERS.

$(T_m)_b$ IS THE VALUE OF T_m AT ALTITUDE H_b.

THIS PIECEWISE CONTINUOUS FUNCTION WAS EMPIRICALLY SELECTED AND THUS, IT IS ASSUMED THAT THE ATMOSPHERE IS MADE UP OF CONCENTRIC LAYERS (AN INTRODUCTION TO ASTRODYNAMICS, BAKER AND MAKEMSON, ACADEMIC PRESS, INC., NEW YORK, 1960, P. 104.))

$$\text{Mach number} = \frac{qv_R}{p\sqrt{T_m}} = \frac{394.35520v_R}{\sqrt{T_m}}$$

WHERE

P = 20.0463333 AND IS SUCH THAT $p\sqrt{T_m}$ IS THE SPEED OF SOUND IN METERS/SECOND, AND q = 7905.375769, WHICH CONVERTS GEMINI VELOCITY UNITS TO METERS/SECOND.

THE COEFFICIENTS OF DRAG AND OF LIFT ARE THEN SELECTED FROM NINETEEN LOCATION TABLES WHICH CORRESPOND TO A NINETEEN-LOCATION TABLE OF MACH NUMBERS. LINEAR INTERPOLATION IS USED IN THIS SELECTION.

USAGE

A) CALLING SEQUENCE -

```
TSX         C4COEF,4
G           L(INPUT),,L(OUTPUT)
ERROR       RETURN
NORMAL      RETURN
```

(G = PZE INDICATES THAT $\sqrt{T_m}$ MUST BE COMPUTED.

G = MZE INDICATES THAT THE LAST COMPUTED VALUE

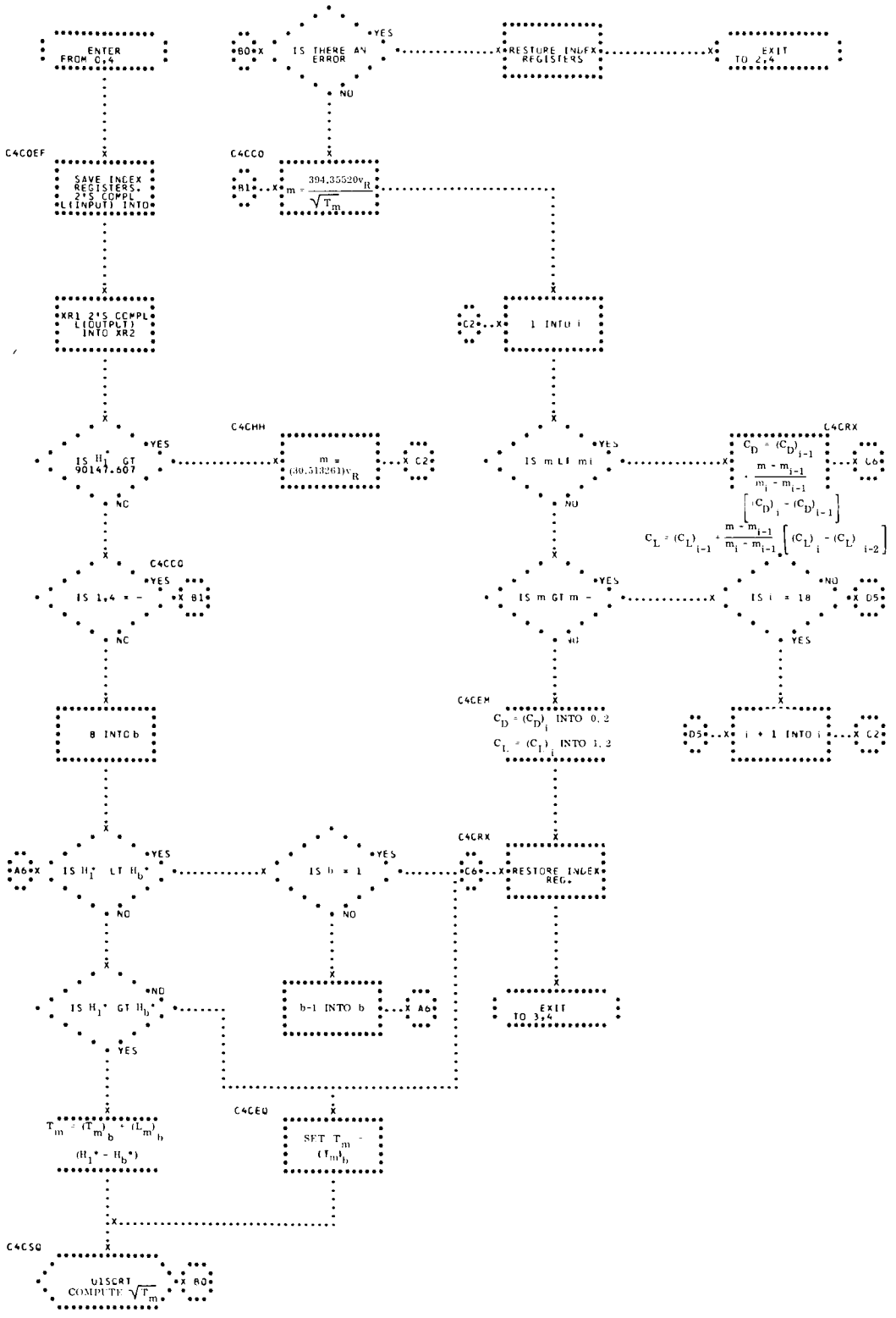
OF $\sqrt{T_m}$ IS TO USED.)

B) STORAGE REQUIRED - 148 LOCATIONS EXCLUDING STORAGE REQUIRED BY SUBROUTINES AND EXTERNAL CONSTANTS.

C) ERROR CONDITIONS - AN ERROR RETURN IS A RESULT OF AN ERROR RETURN FROM U1SQRT.

D) TIME REQUIRED (7094) - 0.602 MILLISECONDS (MAXIMUM) INCLUDING 0.150 MILLISECONDS FOR U1SQRT.

E) CHECKOUT - RESULTS WERE CHECKED AGAINST COMPUTATIONS MADE ON THE DESK CALCULATOR AND WERE SATISFACTORY.


ENTER FROM 0,4
C4C0EF
SAVE INDEX REGISTERS, 2'S COMPL. L(INPUT) INTO
XR1 2'S COMPL L(OUTPUT) INTO XR2
IS H_1 GT 90147.607
YES
NO
C4CHH
$m = \frac{394.35520v_R}{\sqrt{T_m}}$
$m = (30.51326)v_R$
X C2
IS 1,4 = -
C4CCQ
X B1
8 INTO b
IS H_1^* LT H_b^*
IS H_1^* GT H_b^*
IS b = 1
b-1 INTO b
X A6
$T_m = (T_m)_b + (L_m)_b (H_1^* - H_b^*)$
C4CEQ
SET $T_m = (T_m)_b$
C4CSQ
U1SCRT COMPUTE $\sqrt{T_m}$
X B0
IS THERE AN ERROR
X B0
RESTORE INDEX REGISTERS
EXIT TO 2,4
C4C0O
$m = \frac{394.35520v_R}{\sqrt{T_m}}$
B1
1 INTO i
IS m LT mi
IS m GT m -
C4CEM
$C_D = (C_D)_i$ INTO 0,2
$C_L = (C_L)_i$ INTO 1,2
L4CRX
$C_D = (C_D)_{i-1} + \frac{m - m_{i-1}}{m_i - m_{i-1}} \left[(C_D)_i - (C_D)_{i-1} \right]$
$C_L = (C_L)_{i-1} + \frac{m - m_{i-1}}{m_i - m_{i-1}} \left[(C_L)_i - (C_L)_{i-2} \right]$
X C6
IS i = 18
X D5
i + 1 INTO i
X C2
C4CRX
C6
RESTORE INDEX REG.
EXIT TO 3,4

58. C3GSCP - COMPUTATION OF LATITUDE AND LONGITUDE FOR NUMERICAL INTEGRATION

THIS PROGRAM IS AN ADAPTATION OF PROGRAM A3GSCP WHICH COMPUTES THE LATITUDE (ϕ), LONGITUDE (λ), AND HEIGHT (H) OF THE SPACECRAFT. A3GSCP IS USED BY ANOTHER PROGRAM DURING LAUNCH AND ABORT, AND, TO AVOID TRAPPING PROBLEMS, IT IS NECESSARY TO HAVE C3GSCP FOR USE BY THE NUMERICAL INTEGRATION PROGRAM, CORKNI. C3GSCP DOES NOT COMPUTE THE HEIGHT OF THE SPACECRAFT. (SEE FLOWCHART.)

INPUT

A) A FOUR-LOCATION TABLE CONTAINING t IN FLOATING POINT SECONDS AND THE COMPONENTS OF $\overline{r}$ IN FLOATING POINT GEMINI UNITS IS INPUT TO THE PROGRAM.

B) LIBRARY SUBROUTINES U1SQRT, U1ATAB, AND U1ATNA ARE USED.

C) CONSTANTS -

K001.0	FLOATING POINT ONE
K.INB2	INVERSE OF THE SQUARE OF THE SEMI-MINOR AXIS OF THE GEMINI SPHEROID IN FLOATING POINT GEMINI UNITS
K.ECC2	SQUARE OF THE ECCENTRICITY OF THE GEMINI SPHEROID IN FLOATING POINT DIMENSIONLESS FORMAT
K.OMES	ANGULAR ROTATIONAL VELOCITY OF THE EARTH IN FLOATING POINT RADIANS PER SECOND
KKK2PI	FLOATING POINT VALUE OF 2π
KLAMDO	INERTIAL LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH

K.B2GU SEMI-MINOR AXIS OF THE GEMINI SPHEROID SQUARED IN GEMINI UNITS

OUTPUT

A TWO-LOCATION TABLE CONTAINING λ AND ϕ IN THIS ORDER IS OUTPUT FROM THE PROGRAM. THE OUTPUT IS IN FLOATING POINT RADIANS WITH RANGES SUCH THAT

$0 \leq \lambda \leq 2\pi$ AND $\pi/2 \leq \phi \leq \pi/2$.

METHOD

REFER TO PROGRAM A3GSCP FOR THE DERIVATION OF EQUATIONS USED. THE TWO PROGRAMS ARE IDENTICAL WITH THE EXCEPTION OF COMPUTING THE HEIGHT OF THE SPACECRAFT.

USAGE

A) CALLING SEQUENCE -

```
TSX       C3GSCP,4
PZE       L(INPUT),,L(OUTPUT)
ERROR     RETURN
NORMAL    RETURN
```

B) STORAGE REQUIRED -

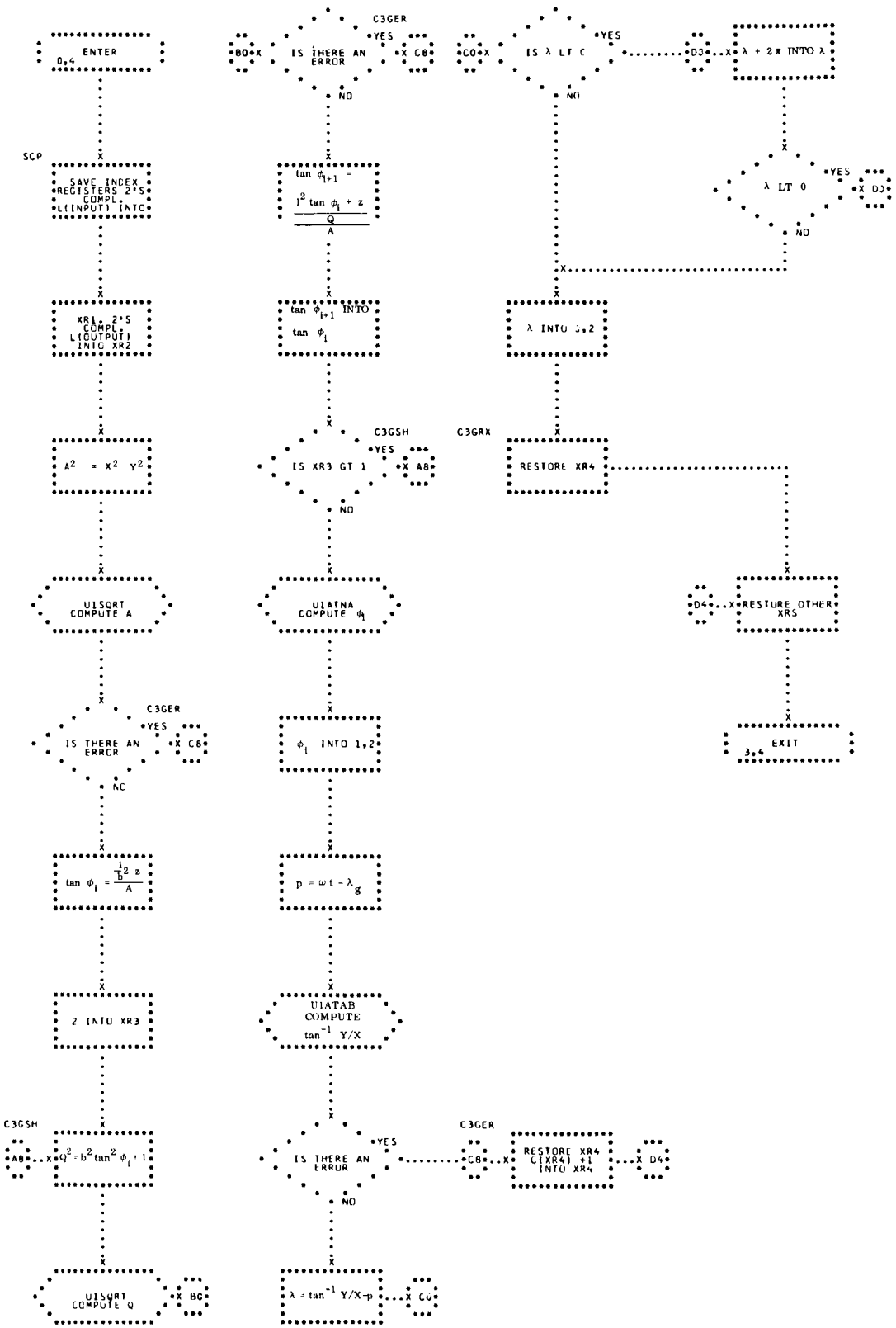
$(61)_{10}$ LOCATIONS EXCLUDING THAT REQUIRED BY SUBROUTINES AND EXTERNAL CONSTANTS.

C) TIME REQUIRED (7094) MAXIMUM -

1.42 MILLISECONDS

D) ERROR CONDITIONS -

AN ERROR RETURN IS A RESULT OF AN ERROR RETURN FROM ONE OF THE LIBRARY SUBROUTINES.


ENTER
0,4
SCP
SAVE INDEX REGISTERS 2'S COMPL. L(INPUT) INTO
XR1, 2'S COMPL. L(OUTPUT) INTO XR2
A2 = X2 Y2
U1SQRT COMPUTE A
C3GER
IS THERE AN ERROR
YES
NO
C8
tan φi = (1/b2 z)/A
2 INTO XR3
C3GSH
A8
Q2 = b2 tan2 φi + 1
U1SQRT COMPUTE Q
BC
C3GER
IS THERE AN ERROR
YES
NO
C8
tan φi+1 = (l2 tan φi + z)/(Q/A)
tan φi+1 INTO tan φi
C3GSH
IS XR3 GT 1
YES
NO
A8
U1ATNA COMPUTE φi
φi INTO 1,2
p = ωt - λg
U1ATAB COMPUTE tan-1 Y/X
IS THERE AN ERROR
YES
NO
C8
λ = tan-1 Y/X-p
CO
C3GER
IS λ LT C
YES
NO
D3
λ + 2π INTO λ
λ LT 0
YES
NO
D3
λ INTO 0,2
C3GRX
RESTORE XR4
D4
RESTORE OTHER XRS
EXIT
3,4
C3GER
RESTORE XR4 C(XR4) +1 INTO XR4
D4

59. MPMPRK - MONITOR PREFIX TO LIMPRK

MPMPRK SETS INDICATORS FOR LIMPRK. (SEE FLOWCHART)

INPUT

(NOT APPLICABLE)

OUTPUT

(NOT APPLICABLE)

METHOD

MPMPRK TURNS ON THE A INDICATOR AND TURNS OFF THE B INDICATOR FOR LIMPRK. CONTROL IS THEN TRANSFERRED TO LIMPRK.

USAGE

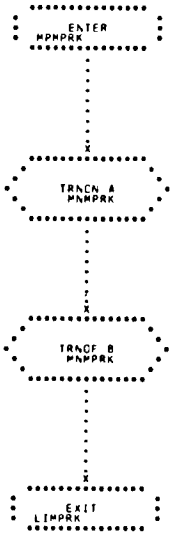
MPMPRK IS ENTERED FROM MOPRIO WHEN CONDITIONS PERMIT, IT TRANSFERS CONTROL TO LIMPRK.

A) PARAMETERS -
   A
   B
   MNMPRK

B) MACROS -
   TRNON
   TRNOF

C) STORAGE REQUIRED -
   5 LOCATIONS

D) TIMING (7094) -
   .014 MILLISECONDS


ENTER
MPMPRK
TRNCN A
MNMPRK
TRNCF B
MNMPRK
EXIT
LIMPRK

60. LIMPRK - CONTROL PROGRAM FOR CIMPRK

LIMPRK CONTROLS THE COMPUTATION OF IMPACT POINTS DURING LAUNCH PHASE. (SEE FLOWCHART)

INPUT

A) TABLE -

TMMPRK	A 7-LOCATION TABLE CONTAINING TIME IN FLOATING POINT GMT SECONDS, $\overline{r}$ AND $\overline{v}$ IN FLOATING POINT GEMINI UNITS.

B) COMMUNICATION CELLS -

MCIMPA	CONTAINS THE ABORT MODE INDICATOR
MCIMPD	CONTAINS THE MINIMUM DELAY TIME TO RETROFIRE IN FLOATING POINT SECONDS.
MCIMPL	CONTAINS THE CURRENT LIFT SWITCH SETTING.

OUTPUT

TABLE

MFIKA	A 4-LOCATION TABLE CONTAINING THE LATITUDES AND LONGITUDES OF THE MINIMUM AND MAXIMUM DELAY IMPACT POINTS.

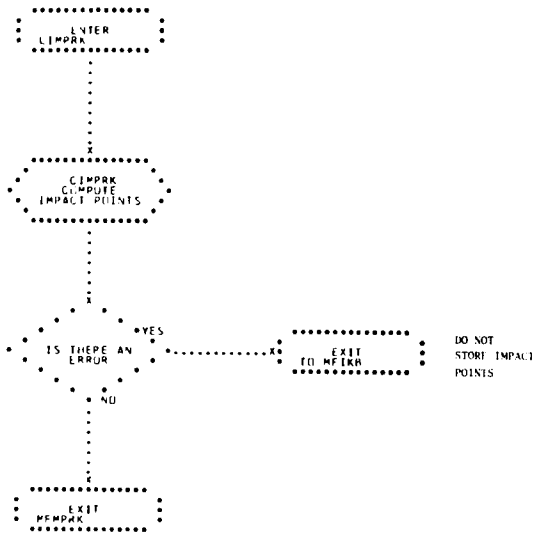
METHOD

LIMPRK USES SUBROUTINE CIMPRK TO COMPUTE MINIMUM AND MAXIMUM DELAY IMPACT POINTS.

USAGE

LIMPRK IS ENTERED FROM MPMPRK AND EXITS TO MFMPRK.

STORAGE REQUIRED - 6 LOCATIONS


ENTER
CIMPRK
CIMPRK
COMPUTE
IMPACT POINTS
IS THERE AN
ERROR
YES
NO
EXIT
TO MFIKR
DO NOT
STORE IMPACT
POINTS
EXIT
MFMPRK

61. CIMPRK - IMPACT PREDICTOR FOR NORMAL LAUNCH

THIS PROGRAM COMPUTES THE LATITUDE AND LONGITUDE OF BOTH MINIMUM AND MAXIMUM DELAY IMPACT DURING NORMAL LAUNCH. (SEE FLOWCHART.)

INPUT

A) CALLING SEQUENCE REQUIREMENTS -

L(INPUT) - THE ADDRESS OF A SEVEN-LOCATION TABLE CONTAINING T IN FLOATING POINT SECONDS AND THE COMPONENTS OF $\overline{r}$ AND $\overline{v}$ IN FLOATING POINT GEMINI UNITS

L(AMI) - THE ADDRESS OF A CELL CONTAINING THE ABORT MODE INDICATOR IN FIXED POINT

L $\Delta t_{r\text{-}min}$ - THE ADDRESS OF A CELL CONTAINING THE MINIMUM DELAY TIME TO RETROFIRE IN FLOATING POINT SECONDS

L(LIFT SETTING) - THE ADDRESS OF A CELL CONTAINING THE LIFT SWITCH SETTING IN FIXED POINT

B) SUBROUTINES CORKNI, C6RETC, CTSFT3, C6PTLI, AND C6PTLU ARE USED.

C) CONSTANTS -

K00001	FIXED POINT ONE
K00002	FIXED POINT TWO
K00003	FIXED POINT THREE
K00006	FIXED POINT SIX
K00018	FIXED POINT EIGHTEEN
K00050	FIXED POINT FIFTY
K00000	ZERO
KM0000	MINUS ZERO
K0C6.0	FLOATING POINT SIX

K045.0 FLOATING POINT FORTY-FIVE

K00.PI FLOATING POINT π (PI)

D) TABLES -

TCBAM2 BANK ANGLE IN FLOATING POINT RADIANS FOR ABORT MODE 2 PROGRAMMED LIFT

TCBAM3 BANK ANGLE IN FLOATING POINT RADIANS FOR ABORT MODE 3 PROGRAMMED LIFT

TCMXCH CRITERION HEIGHT FOR MAXIMUM DELAY IMPACT IN FLOATING POINT GEMINI UNITS

TCNLIP NOMINAL NORMAL LAUNCH IMPACT HEIGHT IN FLOATING POINT GEMINI UNITS

TCRFTQ TORQUE SETTING WHICH WHEN NONZERO (ZERO) INDICATES THAT THE INERTIAL PLATFORM OF THE SPACECRAFT IS (NOT) TORQUED DURING RETROFIRE

TCWF.R REENTRY WEIGHT OF THE SPACECRAFT IN FLOATING POINT POUNDS

TCWF.B ORBIT WEIGHT OF THE SPACECRAFT IN FLOATING POINT POUNDS

OUTPUT

L(OUTPUT) - THE ADDRESS OF A FOUR-LOCATION TABLE CONTAINING RESPECTIVELY THE COMPUTED MINIMUM DELAY LATITUDE AND LONGITUDE, AND THE COMPUTED MAXIMUM DELAY LATITUDE AND LONGITUDE, THE LATTER BEING DEPENDENT UPON ABORT MODE. THIS OUTPUT IS FLOATING POINT RADIANS.

METHOD

AFTER INITIALIZATION, CIMPRK SETS UP THE PROPER PARAMETERS FOR THE MAIN LOGIC OF THE PROGRAM BY CHECKING BOTH THE ABORT MODE INDICATOR (AMI) AND THE LIFT SWITCH SETTING (LI). IF AMI=1(=2,3), ABORT MODE 1(2,3) LOGIC

IS IN EFFECT. LIKEWISE, IF LI=0(=1,2), ZERO LIFT (PROGRAMMED LIFT, MAXIMUM LIFT) IS TO BE USED IN COMPUTING THE REENTRY TRAJECTORIES FOR BOTH MINIMUM AND MAXIMUM DELAY IMPACTS.

A) IF AMI=1, ONLY ONE IMPACT IS COMPUTED, AND ZERO LIFT (A NEGATIVE BANK ANGLE) IS USED BY CORKNI TYPE 1 INTEGRATION. ABORT MODE 1 IS IN EFFECT UNTIL THE TRAJECTORY REACHES A HEIGHT OF 70,000 FEET. BELOW THIS HEIGHT THE RETROROCKETS CANNOT BE USED AS POSIGRADES TO SEPARATE THE SPACECRAFT FROM THE BOOSTER. A MAXIMUM DELAY IMPACT IS UNDEFINED, ITS LATITUDE AND LONGITUDE ARE SET TO 0° AND 180°, RESPECTIVELY, TO SET IT OFF-SCALE TO THE RIGHT.

B) IF AMI=2, ONLY MINIMUM DELAY IMPACT IS COMPUTED USING THE BANK ANGLE SPECIFIED BY THE LIFT SWITCH SETTING. THE VALUES FOR MAXIMUM DELAY IMPACT ARE SET OFF-SCALE AS IN ABORT MODE 1. IN THIS ABORT MODE THE RETROROCKETS ARE USED AS POSIGRADES TO SEPARATE THE CAPSULE FROM THE BOOSTER IN THE EVENT OF AN ABORT. HENCE, NO MAXIMUM DELAY IMPACT IS POSSIBLE.

C) IF AMI=3, ABORT MODE CHANGEOVER (AMC) HAS OCCURRED, AND BOTH MINIMUM AND MAXIMUM DELAY IMPACTS ARE COMPUTED. IN THIS ABORT MODE, ONBOARD PROPULSION IS USED TO SEPARATE THE SPACECRAFT FROM THE BOOSTER, THUS THE RETROROCKETS MAY BE USED TO ACHIEVE EITHER A MINIMUM OR MAXIMUM DELAY IMPACT. THE LOGIC FOR THIS ABORT MODE IS AS FOLLOWS -

CORKNI TYPE 2 INTEGRATES THE INPUT VECTOR THROUGH Δt_r MIN TO GET $t_4, \overline{r}_4, \overline{v}_4$. USING VECTOR

t_4, $\overline{r}_4$, $\overline{v}_4$, C6RETC COMPUTES t_5, $\overline{r}_5$, $\overline{v}_5$, THE BURNOUT VECTOR FOR MINIMUM DELAY IMPACT. CORKNI TYPE 1 THEN INTEGRATES THIS BURNOUT VECTOR TO IMPACT USING THE BANK ANGLE SPECIFIED BY THE LIFT SWITCH SETTING.

USING t_4, $\overline{r}_4$, $\overline{v}_4$, CORKNI TYPE 3 INTEGRATION NEXT GENERATES AN ORBIT TABLE CONTAINING THE COMPONENETS OF $\overline{r}$ AND $\overline{v}$ AND AN ORBIT TABLE CONTAINING THE HEIGHTS OF THESE VECTORS. THESE TABLES ARE USED BY THE INTERPOLATION PROGRAMS C6PTLU AND C6PTLI TO COMPUTE A NEW t_4, $\overline{r}_4$, $\overline{v}_4$ FOR MAXIMUM DELAY IMPACT. THIS t_4 IS EQUAL TO THE TIME THE TRAJECTORY CROSSES THE HEIGHT CONTAINED IN TCMXCH MINUS 45 SECONDS. AGAIN C6RETC USES t_4, $\overline{r}_4$, $\overline{v}_4$ TO COMPUTE THE BURNOUT VECTOR t_5, $\overline{r}_5$, $\overline{v}_5$. CORKNI TYPE 1 THEN USES t_5, $\overline{r}_5$, $\overline{v}_5$ TO INTEGRATE TO IMPACT.

THERE ARE SEVERAL EXCEPTIONS TO THE ABOVE LOGIC.

1) IF, IN GENERATING THIS ORBIT TABLE OF VECTORS AND HEIGHTS, CORKNI (TYPE 3) INTEGRATION CANNOT OUTPUT ANY VECTORS FROM THE DESCENDING PORTION OF THE TRAJECTORY BELOW THE HEIGHT IN TCMXCH, MAXIMUM DELAY IMPACT IS UNDEFINED, AND ITS LATITUDE AND LONGITUDE ARE SET OFF-SCALE.
2) IF, IN GENERATING THIS ORBIT TABLE, CORKNI (TYPE 3) INDICATES THAT THE TRAJECTORY NEVER REACHES THE CRITERION HEIGHT IN TCMXCH, MAXIMUM DELAY IMPACT IS SET TO MINIMUM DELAY IMPACT.
3) IF $t_{4\,\max} \leq t_{4\,\min}$ MAXIMUM DELAY IMPACT IS SET EQUAL TO MINIMUM DELAY IMPACT.

USAGE

A) CALLING SEQUENCE -

```
TSX       CIMPRK,4
PZE       L(INPUT),,L(OUTPUT)
PZE       L(AMI),,  L (Δt r - min)
PZE       L(LI),,**
ERROR     RETURN
NORMAL    RETURN
```

B) STORAGE REQUIRED - 141 LOCATIONS EXCLUDING THAT REQUIRED BY SUBROUTINES AND EXTERNAL CONSTANTS

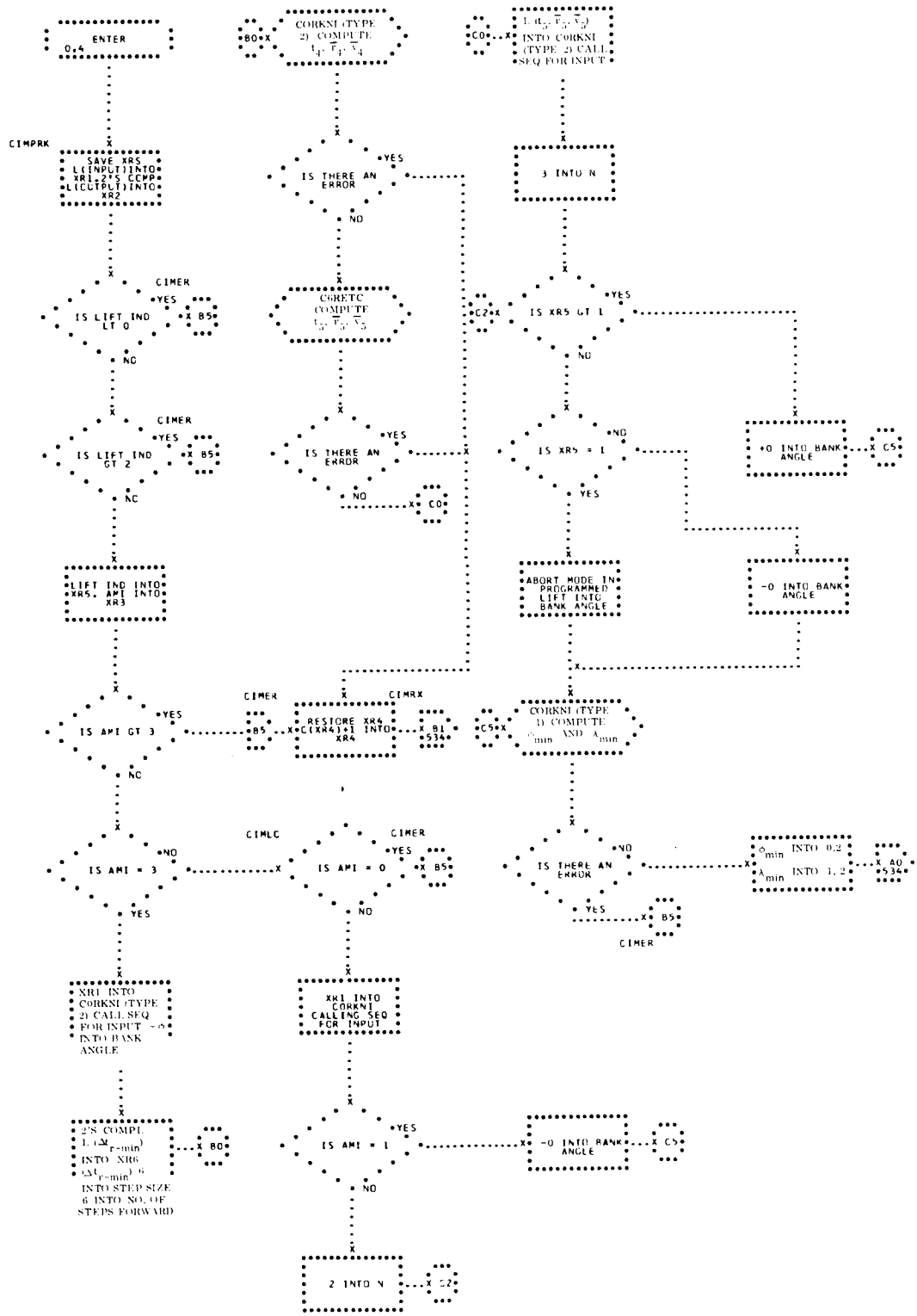
C) TIME REQUIRED (7094) - 1475 MILLISECONDS IS THE MAXIMUM TIME REQUIRED. THIS WOULD OCCUR IF CORKNI WOULD INTEGRATE THE MAXIMUM NUMBER OF STEPS IN ABORT MODE 3. A COMPUTED MAXIMUM MACHINE TIME FOR THE EACH ABORT MODE IS -

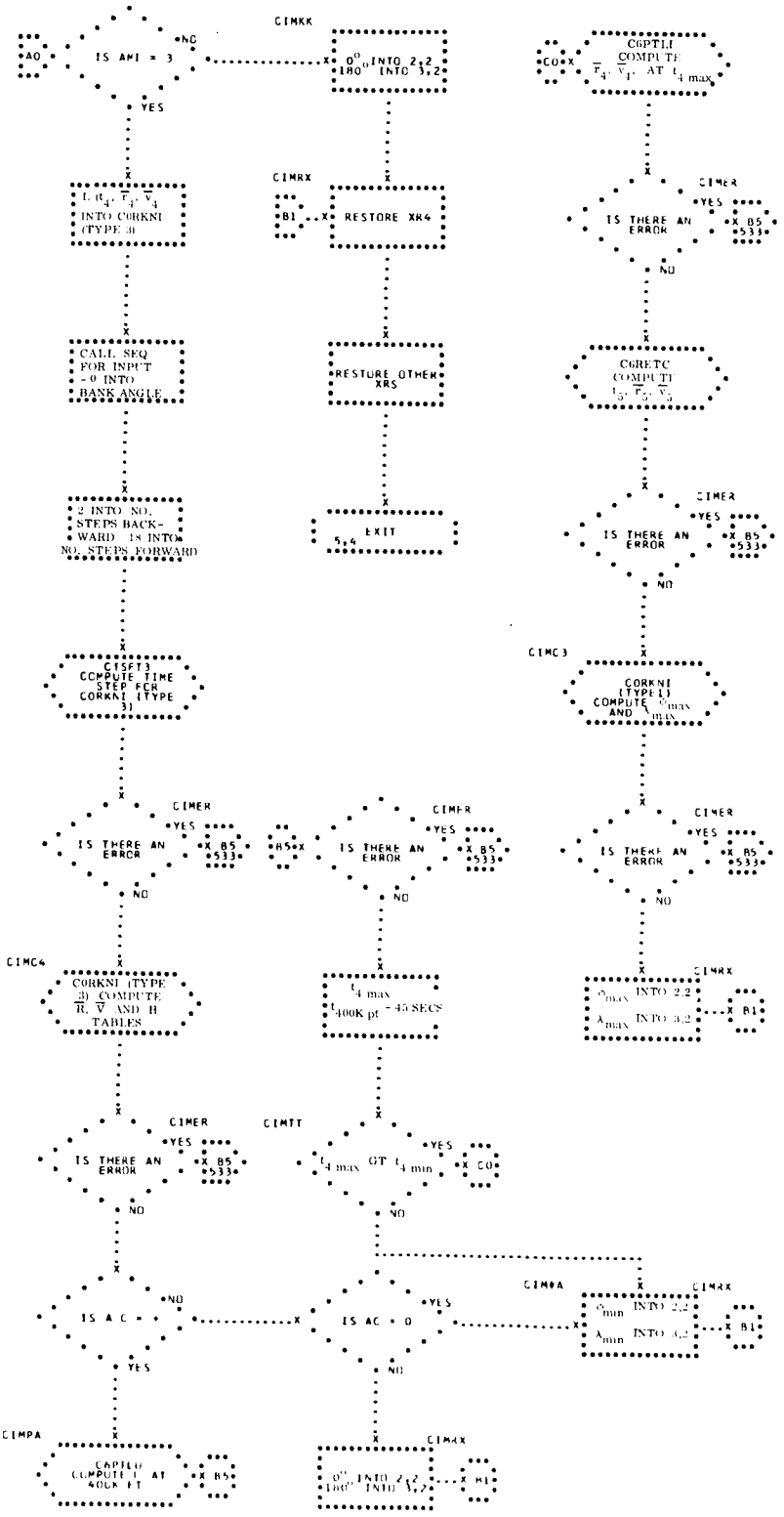
ABORT MODE 1 - 110 MILLISECONDS (USING A VECTOR WHOSE HEIGHT IS SLIGHTLY LESS THAN 70,000 FEET)

ABORT MODE 2 - 250 MILLISECONDS (USING A VECTOR 160 SECONDS AFTER LIFT-OFF, A 60° BANK ANGLE, AND THE SET OF SMALL STEP SIZES FOR CORKNI)

ABORT MODE 3 - 750 MILLISECONDS (USING A VECTOR JUST PRIOR TO SSCO, A 60° BANK ANGLE, AND THE SET OF LARGE STEP SIZES FOR CORKNI)

D) ERROR CONDITIONS - AN ERROR CONDITION IS A RESULT OF AN ERROR RETURN FROM A SUBROUTINE.


ENTER 0,4
CIMPRK
SAVE XR5 L(INPUT)INTO XR1,2'S COMP L(CUTPUT)INTO XR2
CIMER
IS LIFT IND LT 0
YES
X B5
NO
CIMER
IS LIFT IND GT 2
YES
X B5
NO
LIFT IND INTO XR5, AMI INTO XR3
IS AMI GT 3
YES
NO
CIMER
B5
RESTORE XR4 C(XR4)+1 INTO XR4
CIMRX
X B1 534
IS AMI = 3
NO
YES
CIMLC
CIMER
IS AMI = 0
YES
X B5
NO
XR1 INTO CORKNI (TYPE 2) CALL SEQ FOR INPUT +0 INTO BANK ANGLE
2'S COMPL L(Δt F-min) INTO XR6 (Δt F-min) 6 INTO STEP SIZE 6 INTO NO. OF STEPS FORWARD
X B0
XR1 INTO CORKNI CALLING SEQ FOR INPUT
IS AMI = 1
YES
-0 INTO BANK ANGLE
X C5
NO
2 INTO N
X C2
B0
CORKNI (TYPE 2) COMPUTE t5, r5, v5
IS THERE AN ERROR
YES
NO
CORETC COMPUTE t5, r5, v5
IS THERE AN ERROR
YES
NO
X C0
C0
L (t2, r2, v2) INTO CORKNI (TYPE 2) CALL SEQ FOR INPUT
3 INTO N
C2
IS XR5 GT 1
YES
NO
IS XR5 = 1
NO
YES
+0 INTO BANK ANGLE
X C5
ABORT MODE IN PROGRAMMED LIFT INTO BANK ANGLE
-0 INTO BANK ANGLE
C5
CORKNI (TYPE 1) COMPUTE θmin AND Amin
IS THERE AN ERROR
NO
YES
X B5
CIMER
θmin INTO 0,2
Amin INTO 1,2
X A0 534


CIMKK
AO
IS AMI = 3
NO
YES
0° INTO 2,2
180° INTO 3,2
C6PTLI
COMPUTE
AT
CO
CIMRX
B1
RESTORE XR4
CIMER
IS THERE AN ERROR
YES
NO
B5
533
INTO CORKNI (TYPE 3)
CALL SEQ FOR INPUT - 0 INTO BANK ANGLE
RESTORE OTHER XRS
C6RETC
COMPUTE
2 INTO NO. STEPS BACKWARD 18 INTO NO. STEPS FORWARD
EXIT
5,4
CIMC3
CORKNI (TYPE 1) COMPUTE AND
CIMC4
CORKNI (TYPE 3) COMPUTE R, V AND H TABLES
- 45 SECS
INTO 2,2
INTO 3,2
CIMTT
GT
CIMPA
C6PTLU
COMPUTE AT
400K FT

62. MFMPRK - MONITOR SUFFIX TO LIMPRK

MFMPRK PROCESSES THE OUTPUT FROM LIMPRK. (SEE FLOW-CHART)

INPUT

A) TABLE -

MFIKA	A 4-LOCATION TABLE, OUTPUT FROM LIMPRK, CONTAINING THE LATITUDES AND LONGITUDES OF THE MINIMUM AND MAXIMUM DELAY IMPACT POINTS.

OUTPUT

A) TABLE -

TCCOUT	LOCATIONS 8,9,12, AND 13 OF THESE TABLES WILL CONTAIN THE IMPACT POINTS OUTPUT BY LIMPRK.
TMLANA	

METHOD

MFMPRK MOVES THE DATA FROM THE OUTPUT BUFFER OF LIMPRK INTO SPECIFIED LOCATIONS OF THE CCMAIN OUTPUT BUFFER AND CORRESPONDING LOCATIONS OF THE INPUT BUFFER FOR OOLANA.

THE A INDICATOR IS THEN TURNED OFF AND CONTROL IS RETURNED TO MOPRIO.

USAGE

A) PARAMETERS -

A

MNMPRK

B) MACROS -

QENBZ

QENBA

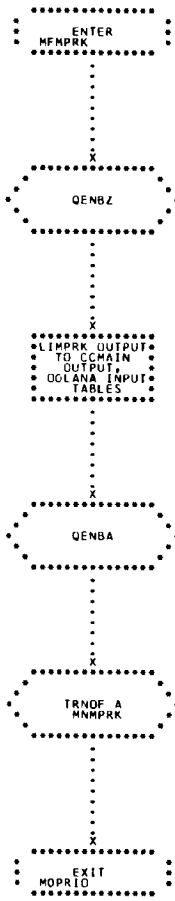
TRNOF

C) STORAGE REQUIRED -

22 LOCATIONS

D) TIMING (7094) -

.024 MILLISECONDS


ENTER
MFMPRK
QENBZ
LIMPRK OUTPUT
TO CCMAIN
OUTPUT,
OOLANA INPUT
TABLES
QENBA
TRNOF A
MNMPRK
EXIT
MOPRIO

63. CUPDAT - LAUNCH PROGRAM FOR UPDATING A GIVEN $t, \bar{r}, \bar{v}$

GIVEN A $t, \bar{r}, \bar{v}$ DEFINING THE ORBIT OF A SPACECRAFT AND THE DESIRED TIME FOR AN EXTRAPOLATED t, r, v THIS PROGRAM COMPUTES THE r AND v AT THE DESIRED TIME BY USE OF NUMERICAL INTEGRATION. (CUPDAT IS USED BY THE PREFIX MPLCCM OF THE MAIN LAUNCH CALCULATION PROGRAM, CCMAIN.) (SEE FLOWCHART)

INPUT

A) SUBROUTINE REQUIRED -
CORKNI - LAUNCH NUMERICAL INTEGRATION PROGRAM.

B) CONSTANTS -

K0CCC2	FIXED POINT 2
K00001	FIXED POINT 1
KM0000	MINUS ZERO

C) PARAMETERS -

TCWF.B	SPACECRAFT ORBIT WEIGHT, FLOATING POINT POUNDS

D) THE INPUT $t, \bar{r}, \bar{v}$ IS IN FLOATING POINT IN A SEVEN-LOCATION TABLE HAVING THE FORMAT -

L(INPUT VECTORS)t TIME IN GMT SECONDS

+1	x	$\bar{r}$ POSITION VECTOR IN GEMINI LENGTH UNITS
+2	y	
+3	z	
+4	$\dot{x}$	$\bar{v}$ VELOCITY VECTOR IN GEMINI VELOCITY UNITS
+5	$\dot{y}$	
+6	z	

E) L(TIME FOR WHICH UPDATED VECTOR IS DESIRED) TIME IN GMT SECONDS

F) L(THRUST CELL) - THRUST CELL CONTAINS -

1) 0 IF THRUST IS NOT TO BE INCLUDED DURING THE UPDATING

2) A NON-ZERO QUANTITY IF THRUST IS TO BE INCLUDED

OUTPUT

THE OUTPUT IS IN FLOATING POINT IN SEVEN-LOCATION TABLE IN THE FOLLOWING FORMAT -

L(UPDATED VECTORS)t		TIME IN GMT SECONDS
+1	x	$\bar{r}$ IN GEMINI LENGTH UNITS
+2	y	
+3	z	
+4	$\dot{x}$	$\bar{v}$ IN GEMINI VELOCITY UNITS
+5	$\dot{y}$	
+6	$\dot{z}$	

METHOD

CUPDAT SUBTRACTS THE TIME OF A GIVEN INPUT VECTOR FROM THE TIME AT WHICH THE UPDATED VECTOR IS DESIRED, THEREBY CALCULATING THE TIME INCREMENT OVER WHICH NUMERICAL INTEGRATION IS TO BE DONE. THE NUMERICAL INTEGRATION IS ACCOMPLISHED IN A SINGLE STEP USING PROGRAM CORKNI WITH THRUST OR WITHOUT THRUST. IN THE EVENT OF AN ERROR IN INTEGRATION, CUPDAT PLACES THE INPUT t, $\bar{r}$, $\bar{v}$, IN THE OUTPUT TABLE.

USAGE

A) CALLING SEQUENCE -

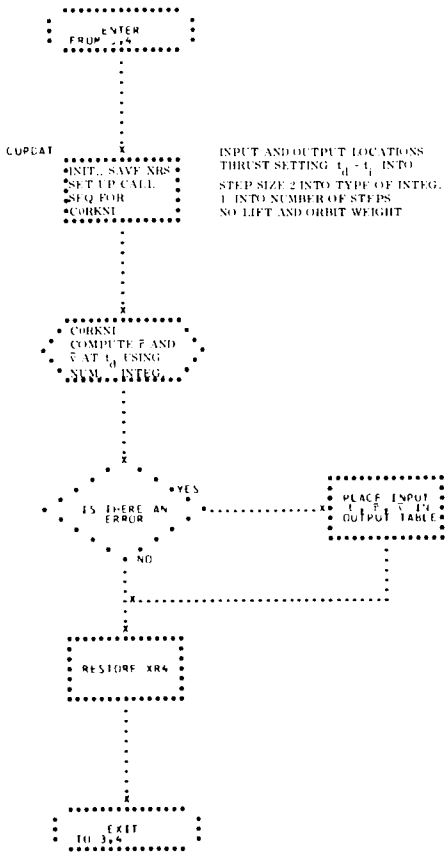
```
TSX     CUPDAT,4
PZE     L(INPUT VECTORS t, r̄, v̄ ),,L(UPDATED
        INPUT VECTORS t, r̄, v̄ )
PZE     L(TIME AT WHICH UPDATED t, r̄, v̄ DESIRED),,
        L(THRUST CELL)
RETURN
```

B) STORAGE REQUIRED - 31 LOCATIONS EXCLUDING SUB-ROUTINES, CONSTANTS, AND INPUT AND OUTPUT TABLES

C) TIMING - .0652 MILLISECONDS EXCLUDING CORKNI (NORMAL CASE)

.172 MILLISECONDS EXCLUDING CORKNI (ERROR CASE)

VOL III REAL TIME PROCESSORS PART I - LAUNCH/ABORT



64. MABORT - MONITOR CONTROL FOR ABORT PHASE

MABORT DETERMINES WHICH ABORT SITUATION THE MISSION HAS ENTERED AND DIRECTS PROCESSING OF DATA ACCORDINGLY. (SEE FLOWCHART)

INPUT

A) TABLES

USED IN BOTH ABORT PHASES -

TMABDV	AN EIGHT LOCATION TABLE CONTAINING THE ABORT DEFINING VECTOR IN THE SECOND THROUGH SEVENTH CELLS. THE FIRST AND EIGHTH CELLS CONTAIN THE VECTOR TIME IN FLOATING AND IN FIXED POINT GMT SECONDS, RESPECTIVELY.

USED IN HIGH ABORT PHASE -

TMRETO	AN ELEVEN LOCATION TABLE CONTAINING RETROFIRE INFORMATION OF WHICH ONLY THE FIRST FIVE CELLS ARE USED BY MABORT. THE FIRST FOUR CELLS CONTAIN THE TIMES OF FIRING OF RETROCKETS 1,2,3, AND FOUR, IN THAT ORDER. THE DECREMENT OF THE FIFTH CELL CONTAINS THE NOMINAL OR NON-NOMINAL

	RETROFIRE SEQUENCE INDICATOR.
USED IN LOW ABORT PHASE -	
TMBDAV	A SEVEN LOCATION TABLE CONTAINING THE BERMUDA VECTOR AND ITS ASSOCIATED TIME TAG. THIS TABLE IS SET BY THE BERMUDA DATA PROCESSORS.
TMSLMF	A TWO LOCATION TABLE USED AS INPUT TO THE SENSE LIGHT PROGRAM TO TURN OFF THE MISSING DATA LIGHT. THE FIRST LOCATION INDICATES THE LIGHTS TO BE TURNED ON, THE SECOND INDICATES THE LIGHTS TO BE TURNED OFF.
TMSLMN	A TWO LOCATION TABLE USED AS INPUT TO THE SENSE LIGHT PROGRAM TO TURN ON THE MISSING DATA LIGHT. THE FIRST LOCATION INDICATES THE LIGHTS TO BE TURNED ON, THE SECOND INDICATES THE LIGHTS TO BE TURNED OFF.
TMBDLO	A SEVEN LOCATION TABLE CONTAINING THE INPUT BERMUDA VECTOR REFLECTING THE MINIMUM LIFT CONFIGURATION AND ITS

	ASSOCIATED TIME TAG.
TMBDL1	A SEVEN LOCATION TABLE CONTAINING THE INPUT BERMUDA VECTOR REFLECTING THE PROGRAMMED LIFT CONFIGURATION AND ITS ASSOCIATED TIME TAG.
TMBDL2	A SEVEN LOCATION TABLE CONTAINING THE INPUT BERMUDA VECTOR REFLECTING THE MAXIMUM LIFT CONFIGURATION AND ITS ASSOCIATED TIME TAG.
TMSML0	A SEVEN LOCATION TABLE CONTAINING THE INPUT IP-7094 VECTOR REFLECTING THE MINIMUM LIFT CONFIGURATION AND ITS ASSOCIATED TIME TAG.
TMSML1	A SEVEN LOCATION TABLE CONTAINING THE INPUT IP-7094 VECTOR REFLECTING THE PROGRAMMED LIFT CONFIGURATION AND ITS ASSOCIATED TIME TAG.
TMSML2	A SEVEN LOCATION TABLE CONTAINING THE INPUT IP-7094 VECTOR REFLECTING THE MAXIMUM LIFT CONFIGURATION AND ITS ASSOCIATED TIME TAG.

USED IN BOTH ABORT PHASES -

MCAFIN	NON-ZERO IF ABORT PHASE IS FINISHED.
MCHFSC	CONTAINS CURRENT GMT IN FIXED POINT HALF-SECONDS, B35.
MCLFTM	CONTAINS TIME OF LIFTOFF.
MCMAN2	ZERO IF MANUALLY INSERTED RETORFIRE MESSAGE HAS BEEN ACCEPTED.
MCNRRF	CONTAINS NUMBER OF RETRO-ROCKETS FIRED IN FIXED POINT, B35.
MCWICH	TYPE OF ABORT INDICATOR- 0 = LOW ABORT 0 = HIGH ABORT
MCMONE	FVE 0

USED IN HIGH ABORT PHASE -

MC1RRF	CONTAINS TIME OF FIRING OF RETROROCKET =1
MC2RRF	CONTAINS TIME OF FIRING OF RETROROCKET =2
MC3RRF	CONTAINS TIME OF FIRING OF RETROROCKET =3
MC4RRF	CONTAINS TIME OF FIRING OF RETROROCKET =4
MCFLSS	FINAL LIFT SWITCH SETTING IN FIXED POINT, B 35, FROM LAUNCH PHASE

	MCFTRC	FINAL GMTRC ASSOCIATED WITH ABORT DEFINING VECTOR FROM LAUNCH PHASE
	MCPCNE	PON 0
	MCXXXX	OCT 777777777777
	USED IN LOW ABORT PHASE -	
	MCACBD	NON-ZERO IF BERMUDA HIGH-SPEED SUBCHANNELS OF THE DCC HAVE BEEN ACTIVATED.
	MCLIFT	CONTAINS SETTING OF LIFT SWITCH IN FIXED POINT, B35.
	MCMBDA	NON-ZERO IF BERMUDA DATA IS IN MISSING DATA PERIOD (SET BY BERMUDA DATA PROCESSORS).
	MCPBDA	NON-ZERO IF BERMUDA DATA IS IN MISSING DATA PERIOD (NOT RESET ONCE BERMUDA DATA HAS BEEN GIVEN TO ABORT PROGRAM).
	MCPSEL	REFLECTS SOURCE SELECTED FOR THIS ENTRY - 1 = B-GE, 2 = BERMUDA, 4 = IP-7094, IN FIXED POINT, B35.
	MCPTWO	PTW 0
B)	CONSTANTS -	
	K00000	DEC 0
	K00001	DEC 1
	K00002	DEC 2

```
K00003          DEC     3
K00004          DEC     4
K00007          DEC     7
K00008          DEC     8
K00060          DEC     60
K8.3M3          DEC     8.3E-3
K005.0          DEC     5.0
K030.0          DEC     30.0
KCH232          OCT     232000000000
KCH233          OCT     233000000000
KRTDEG          CONVERSION OF RADIANS TO
                DEGREES IN FLOATING
                POINT
```

C) PARAMETERS

```
A EQU 1         INDICATOR BIT A
B EQU 2         INDICATOR BIT B
C EQU 3         INDICATOR BIT C
D EQU 4         INDICATOR BIT D
E EQU 5         INDICATOR BIT E
MNABRT          PRIORITY NUMBER FOR
                MABORT
MNACQD          PRIORITY NUMBER FOR
                MYACQD
MNDIFC          PRIORITY NUMBER OF
                MPDIFC
MNDIFK          PRIORITY NUMBER OF
                MPDIFK
MNDIFM          PRIORITY NUMBER FOR
                MPDIFM
MNEDBD          PRIORITY NUMBER FOR
                MPEDBD
```

MNHS09	PRIORITY NUMBER FOR MPHS09
MNHSGB	PRIORITY NUMBER FOR MPHSGB
MNLANA	PRIORITY NUMBER FOR MPLANA
MNLED1	PRIORITY NUMBER FOR MPLED1
MNLED2	PRIORITY NUMBER FOR MPLED2
MNMESS	PRIORITY NUMRER FOR MYMESS
MNORMC	PRIORITY NUMBER FOR MPORMC
MNREN2	PRIORITY NUMBER FOR MYREN2
MNSENS	PRIORITY NUMBER FOR MYSENS

OUTPUT

A) TABLES

USED IN BOTH ABORT PHASES -

TCCOUT	A TWENTY-NINE LOCATION TABLE CONTAINING THE COMPUTED QUANTITIES FROM THE ABORT PORCESSOR IN USE AT THE CURRENT TIME.
TMABDV	A SEVEN LOCATION TABLE CONTAINING THE ABORT DEFINING VECTOR AND ITS ASSOCIATED TIME TAG.

TMCYCL	A FIVE LOCATION TABLE USED BY THE HALF-SECOND PROCESSOR TO REQUEST PROGRAMS ON PRESET CYCLES. THE LOCATION CONTAINING THE CYCLE FOR THE OUTPUT PROCESSOR IS CHANGED TO PERMIT ENTRY AT THREE SECOND INTER-VALS.
TMCYNO	A FIVE LOCATION TABLE CONTAINING THE PRESET CYCLE TIMES FOR THE HALF-SECOND TRAP PRO-CESSOR) THE LOCATION CONTAINING THE CYCLE FOR THE OUTPUT PROCESSOR IS CHANGED TO THREE SECONDS.
TMLANA	A TWENTY-NINE LOCATION TABLE CONTAINING THE IN-PUT QUANTITIES FOR THE OUTPUT PROCESSOR, OOLANA (SET FROM THE TABLE TCCOUT).
TMORMC+1	THE SECOND LOCATION OF THE TMORMC TABLE CON-TAINS THE LONGITUDE OF PRESENT POSITION IN FLOATING POINT RADIANS.

TMRARF	CONTAINS THE REENTRY DEFINING VECTOR AND ITS ASSOCIATED TIME TAG.
TMSSEC	CONTAINS THE TIME OF RETROFIRE IN FIXED POINT GMT SECONDS.

USED IN HIGH ABORT PHASE -

TMABPR	A TWELVE LOCATION TABLE CONTAINING ABORT INFORMATION FOR ON-LINE PRINTING.

USED IN LOW ABORT PHASE -

TMBDOL	A SEVEN LOCATION TABLE CONTAINING THE INPUT BERMUDA VECTORS INTEGRATED TO REFLECT A MINIMUM LIFT CONFIGURATION AND THEIR ASSOCIATED TIME.
TMBD1L	A SEVEN LOCATION TABLE CONTAINING THE INPUT BERMUDA VECTORS INTEGRATED TO REFLECT A PROGRAMMED LIFT CONFIGURATION WITH THEIR ASSOCIATED TIME.

TMBD2L	A SEVEN LOCATION TABLE CONTAINING THE INPUT BERMUDA VECTORS INTEGRATED TO REFLECT MAXIMUM LIFT AND THEIR ASSOCIATED TIME.
TMSM0L	A SEVEN LOCATION TABLE CONTAINING THE INPUT IP-7094 VECTORS INTEGRATED TO REFLECT MINIMUM LIFT AND THEIR ASSOCIATED TIME.
TMSM1L	A SEVEN LOCATION TABLE CONTAINING THE INPUT IP-7094 VECTORS INTEGRATED TO REFLECT PROGRAMMED LIFT AND THEIR ASSOCIATED TIME.
TMSM2L	A SEVEN LOCATION TABLE CONTAINING THE INPUT IP-7094 VECTORS INTEGRATED TO REFLECT MAXIMUM LIFT AND THEIR ASSOCIATED TIME.
TMSUM6	A SEVEN LOCATION TABLE CONTAINING THE AVERAGE FPQ-6 VECTORS AND THEIR ASSOCIATED TIME.
TMSUM7	A SEVEN LOCATION TABLE CONTAINING THE AVERAGE FPS-16 VECTORS AND THEIR ASSOCIATED TIME.

B) COMMUNICATION CELLS -

USED IN BOTH ABORT PHASES -

MCACTV	THE ACTIVATE MASK FOR THE SUBCHANNELS OF THE OCC.
MCBJMN	CONTAINS THE NUMBER OF THE ON-LINE MESSAGE OF THE HIGH-SPEED OUTPUT TRANSMISSION RATE TO CAPE KENNEDY.
MCCURT	CONTAINS CURRENT MACHINE TIME.
MCHFS1	CONTAINS THE NUMBER OF ENTRIES IN TMCYCL TO BE EXAMINED BY THE HALF-SECOND TRAP PROCESSOR.
MCLGT1	CONTAINS THE LOG TIME ASSOCIATED WITH THE CURRENT OUTPUT QUANTITIES.
MCLIDT	CONTAINS THE LOG TIME ASSOCIATED WITH THE CURRENT OUTPUT QUANTITIES.
MCLOPR	CONTAINS THE TIME OF LIFT-OFF FOR LOGGING PURPOSES.
MCPHSE	CONTAINS THE PHASE IDENTIFICATION IN THE PREFIX - 1 = HIGH ABORT, 2 = LOW ABORT, 5 = RE-ENTRY.

MCTABT	NON-ZERO TO INDICATE MISSION HAS ENTERED THE ABORT PHASE (USED BY THE OUTPUT PROCESSOR).

USED BY HIGH ABORT PHASE -

MCESAB	CONTAINS GMT OF FIRST EVEN SECOND FOLLOWING BURNOUT.

USED BY LOW ABORT PHASE -

MCMAXT	CONTAINS THE ENDING TIME FOR AVERAGE VECTOR SUMMATION.
MCMINT	CONTAINS THE STARTING TIME FOR AVERAGE VECTOR SUMMATION.
MCSINP	CONTAINS IDENTIFICATION OF SOURCE CURRENTLY IN PROCESS - 1 = B-GE, 3 = BERMUDA, 6 = FPQ-6, 7 = FPS-16.
MCSSOR	CONTAINS IDENTIFICATION OF SOURCE SELECTED FOR CURRENT ENTRY - 1 = BE-GE, 3 = BERMUDA, 6 = FPQ-6, 7 = FPS-16.

MCSUMS	SOURCE IDENTIFICATION ON WHICH AVERAGE VECTOR PROCESSING IS TO TAKE PLACE - 6 = FPQ-6, 7 = FPS-16.

METHOD

MABORT CONSISTS OF TWO SECTIONS OF LOGIC WITH ONE ENTRY POINT AND TWO EXIT POINTS. EACH OF THE ABORT PHASES - LOW AND HIGH - IS PROCESSED BY A SEPARATE SECTION OF LOGIC. ONE EXIT IS USED DURING NORMAL ABORT PHASE PROCESSING, THE OTHER EXIT PROVIDES INTERFACE WITH THE REENTRY PHASE WHEN THE ABORT PHASE HAS BEEN COMPLETED.

IN BOTH LOW AND HIGH ABORT PHASE PROCESSING, COMPUTATIONS ARE MADE BY SUBROUTINE TYPE PROGRAMS. INFORMATION ENTERS THE PROGRAM VIA A CALLING SEQUENCE AND CONTROL RETURNS TO MONITOR IMMEDIATELY FOLLOWING THE END OF COMPUTATIONS.

A) LOW ABORT PHASE PROCESSING

1. AS SOON AS THE LOW ABORT PHASE LOGIC IS ENTERED, MABORT DETERMINES WHETHER A MANUALLY INSERTED MESSAGE CONTAINING RETROFIRE INFORMATION HAS BEEN RECEIVED. IF IT HAS, CONTROL IS PASSED TO C) TO PROVIDE INTERFACE WITH THE REENTRY PHASE. IF NO MANUAL INSERTION MESSAGE HAS BEEN RECEIVED, BUT THE HEIGHT OF THE SPACECRAFT ABOVE AN OBLATE EARTH IS BELOW THE NOMINAL LAUNCH/ABORT IMPACT HEIGHT, EXIT TO MOPRIO IS MADE IMMEDIATELY.

2. IF NEITHER OF THE ABOVE CONDITIONS HOLDS, THE LIFT SWITCH SETTING AND THE CURRENT TIME ARE SAVED. WHEN CURRENT TIME SO INDICATES, IF THE BERMUDA HIGH-SPEED SUBCHANNELS OF THE DCC HAVE NOT YET BEEN ACTIVATED, THEY ARE ACTIVATED TO PERMIT RECEIPT OF BERMUDA RADAR DATA AND THE BERMUDA DATA EDIT PROGRAM IS UNSUPPRESSED.

3. PROCESSING IN THE LOW ABORT PHASE IS DONE ON SPACECRAFT DATA SOURCES ONLY - IP-7094 AND BERMUDA. EACH OF THESE SOURCES HAS THREE BUFFERS WHICH SERVE AS INPUT TO THE LOW ABORT PROCESSOR, EACH BUFFER REFLECTS INTEGRATION OF THE VECTOR TO CURRENT TIME APPLYING EACH OF THE THREE LIFT SWITCH SETTING - MINIMUM LIFT, PROGRAMMED LIFT, AND MAXIMUM LIFT.

BEFORE ANY COMPUTATIONS ARE MADE WITH DATA RECEIVED VIA THE IP-7094 SUBCHANNEL, AN AVERAGE VECTOR IS OBTAINED FROM THE VECTORS RECEIVED WITHIN THE LAST FIVE SECONDS OVER THAT LINE. IF THE FIRST IP-7094 SPACECRAFT SOURCE DID NOT SEND DATA DURING INTERVAL, THE SECOND IP-7094 SPACECRAFT SOURCE IS USED. IF NEITHER SOURCE HAS SENT DATA OVER THE PAST FIVE SECONDS, THE VECTORS OUTPUTTED BY THE INTEGRATION PROGRAM, CABDAT, ON THE LAST PASS, ARE PLACED IN THEIR RESPECTIVE INPUT BUFFERS FOR THIS PASS.

IF EITHER OF THE IP-7094 SPACECRAFT DATA SOURCES HAS SENT DATA, THE AVERAGE VECTOR

OBTAINED IS PLACED IN THE THREE INPUT BUFFERS FOR LATER USE BY CABDAT.

SMILIAR CHECKS ARE MADE ON RADAR DATA RECEIVED FROM BERMUDA. HOWEVER, SINCE THE VECTORS OBTAINED FROM THE BERMUDA DATA SOURCES ARE BASED ON SEVERAL SECONDS OF RADAR DATA, THEY REFLECT THE ''AVERAGE'' OF THIS DATA, THEREFORE, NO FURTHER AVERAGING IS NECESSARY.

4. MABORT THEN DETERMINES THE SELECTED SOURCE ON THIS ENTRY. IF NEITHER IP-7094 NOR BERMUDA HAS BEEN SELECTED, MABORT SETS B-GE TO THE SELEGTED SOURCE AND IP-7094 TO THE SOURCE CURRENTLY IN PROCESS.

IF EITHER IP-7094 OR BERMUDA IS THE SELECTED SOURCE, THIS SOURCE ALSO BECOMES THE SOURCE CURRENTLY IN PROCESS.

5. IF THE MISSING DATA LIGHT IS ON DURING A PERIOD IN WHICH DATA IS BEING RECEIVED, THE LIGHT IS TURNED OFF. CONVERSELY, IF THE LIGHT IS OFF DURING A MISSING DATA PERIOD, IT IS TURNED ON.

6. THE SUBROUTINE, CABDAT, IS ENTERED TO INTEGRATE THE VECTORS FROM IP-7094 AND BERMUDA TO THE CURRENT TIME USING EACH OF THE THREE LIFT CONFIGURATIONS.

BASED ON BOTH THE SELECTED SOURCE AND THE LIFT SWITCH SETTING, ONE OF THE VECTORS FROM CABDAT SERVES AS INPUT TO THE LOW ABORT PROCESSOR, CLABRT. UPON EXIT FROM CLABRT, IF

DATA WAS NOT MISSING ON THIS ENTRY, THE VECTOR TIME IN ELAPSED TIME SINCE LIFT-OFF IS COMPUTED AND SAVED FOR LOGGING PURPOSES. MABORT THEN CONTINUES AS IN F).

B) HIGH ABORT PHASE PROCESSING

1. UPON ENTRY TO THE HIGH ABORT PHASE LOGIC, THE CURRENT TIME IS COMPUTED AND SAVED.

2. ON THE FIRST PASS THROUGH HIGH ABORT PROCESSING, THE PROCESSOR CHABRT IS ENTERED WITH A TYPE 1 ENTRY INDICATION TO BUILD THE ORBIT AND REENTRY VECTOR AND HEIGHT TABLES BASED ON THE VECTORS AND THE COMPUTED GREENWICH MEAN TIME OF RETORFIRE (GMTRC) FROM AVERAGE VECTORS PROCESSING. AFTER THE MESSAGE ROUTINE HAS BEEN QUEUED TO PRINT THE ABORT QUANTITIES ON-LINE, MABORT CONTINUES AS IN F).

3. ON THE SECOND AND SUBSEQUENT PASSES THROUGH HIGH ABORT PROCESSING, MABORT DETERMINES WHETHER A MESSAGE CONTAINING RETORFIRE INFORMATION HAS BEEN MANUALLY INSERTED AND ACCEPTED. IF IT HAS, CONTROL PASSES TO D). IF NO SUCH MESSAGE HAS BEEN RECEIVED, BUT THE HEIGHT OF THE SPACECRAFT ABOVE AN OBLATE EARTH HAS FALLEN BELOW THE NOMINAL LAUNCH/ABORT IMPACT HEIGHT, EXIT TO MOPRIO IS MADE IMMEDIATELY.

4. IF NEITHER OF THE ABOVE SITUATIONS IS TURE, MABORT CHECKS THE TIME OF RETORFIRE. IF THIS TIME IS NON-ZERO, CONTROL PASSES TO

7. IF THIS TIME IS ZERO AND RETROFIRE HAS NOT OCCURRED YET, CONTROL PASSES TO 5. OTHERWISE, MABORT CONTINUES AS IN 6.
5. A CHECK IS MADE TO DETERMINE WHETHER A TYPE 2 ENTRY HAS EVER BEEN MADE TO CHABRT. IF SUCH AN ENTRY HAS BEEN MADE, THE RETRO-FIRE INFORMATION (I.E., THE NUMBER OF RETRO-ROCKETS FIRED AND THE EARLIEST TIME OF FIRING) RECEIVED VIA TELEMETRY AND USED BY THE TYPE 2 ENTRY IS GIVEN AS INPUT TO A TYPE 3 ENTRY TO CHABRT. IF A TYPE 2 ENTRY HAS NOT BEEN MADE, NO RETROFIRE INFORMATION HAS BEEN RECEIVED FROM TELEMETRY. THEREFORE, THE TYPE 3 ENTRY IS MADE USING THE COMPUTED TIME OF RETROFIRE (CMTRC) FROM AVERAGE VECTORS PROCESSING AND ZERO RETROROCKETS FIRED. IN EITHER CASE, UPON EXIT FROM CHABRT, CONTROL IS GIVEN TO F).
6. THE TIMES OF FIRING OF THE RETRO-ROCKETS ARE SORTED TO FIND THE EARLIEST TIME. SINCE PROCESSING OF A TYPE 2 ENTRY TO CHABRT MAY NOT TAKE PLACE UNTIL 30 SECONDS AFTER THE FIRING OF THE FIRST RETROROCKET, MABORT UPDATES THE EARLIEST TIME OF FIRING AND SAVES THE NEW TIME FOR A LATER CHECK.
7. THE TIME AT WHICH A TYPE 2 ENTRY MAY BE MADE TO CHABRT IS COMPARED TO THE CURRENT TIME. IF CURRENT TIME INDICATES THAT THE EN-TRY MAY NOT BE MADE, CONTROL GOES TO 5. OTHERWISE, BEFORE ENTERING CHABRT, MABORT DETERMINES WHETHER THIS TYPE 2 ENTRY IS THE

FIRST SUCH ENTRY. IF IT IS, MABORT CONTINUES AS IN 9. IF THE CURRENT TYPE 2 ENTRY IS NOT THE FIRST, BUT ALL TELEMETRY INDICATIONS OF THE NUMBER OF RETROROCKETS FIRED HAVE BEEN LOST, MABORT DOES NOT PROCESS TYE TYPE 2 ENTRY, BUT REINITIALIZES ALL ABORT PROGRAM CELLS AND GIVES CONTROL TO 2 TO START THE ABORT PHASE AGAIN.

8. IF THE TELEMETRY RECEIVED INDICATES THAT THE NUMBER OF RETROROCKETS FIRED HAS CHANGED FROM THAT OF THE PREVIOUS ENTRY, MABORT TRANSFERS CONTROL TO 6 WITH THE NEW INFORMATION. OTHERWISE, THE EARLIEST TIME OF FIRING IS FOUND AND IS COMPARED WITH THE EARLIEST TIME OF FIRING OF THE PREVIOUS ENTRY. IF NO DIFFERENCE IN THE TIMES IS FOUND, CONTROL PASSES TO 5. A DIFFERENCE IN THE EARLIEST TIMES OF FIRING CAUSES MABORT TO RECOMPUTE THE TIME AT WHICH A TYPE 2 ENTRY TO CHABRT MAY BE MADE AND TO CONTINUE AS IN 7.

9. ON THE FIRST TYPE 2 ENTRY TO CHABRT, MABORT CHECKS TO INSURE THAT AT LEAST ONE RETROROCKET HAS FIRED. IF IT FINDS THAT NONE HAVE FIRED, THE ABORT PROGRAM CELLS ARE REINITIALIZED AND CONTROL IS PASSES TO 2 TO BEGIN THE HIGH ABORT PHASE ANEW.

OTHERWISE, MABORT FINDS THE EARLIEST TIME OF FIRING. IF THIS TIME IS DIFFERENT FROM THE CORRESPONDING TIME OF THE PREVIOUS ENTRY, A NEW TIME AT WHICH A TYPE 2 ENTRY TO

CHABRT MAY BE MADE IS COMPUTED AND IS CHECKED AGAINST THE CURRENT TIME. IF CURRENT TIME WILL PERMIT A TYPE 2 ENTRY ON THIS PASS, CONTROL IS GIVEN TO 5. IF, HOWEVER, EITHER THE TIME OF FIRING HAS NOT CHANGED BETWEEN THIS ENTRY AND THE PREVIOUS ONE OR THIS TIME HAS CHANGED, BUT A TYPE 2 ENTRY TO CHABRT STILL MAY BE MADE, MABORT CONTINUES PROCESSING.

10. THE FIRING TIME OF EACH RETROROCKET IS PRINTED ON-LINE AND THE TYPE 2 ENTRY TO CHABRT MADE TO BUILD THE ORBIT AND REENTRY VECTOR AND HEIGHT TABLES BASED ON THE EARLIEST TIME OF FIRING, THE NUMBER OF RETROROCKETS FIRED, AND THE INPUT VECTORS FROM AVERAGE VECTORS PROCESSING.

AFTER THE ABORT QUANTITIES COMPUTED BY CHABRT HAVE BEEN PRINTED ON-LINE, MABORT DETERMINES, FROM THE SETTING OF THE CELL, MCESAB, WHETHER OR NOT TO ENTER THE ACQUISITION DATA PROGRAM. THE CELL MCESAB INITIALLY CONTAINS ALL ONE'S. HOWEVER, CHABRT SETS THE CELL TO ZERO UNTIL IT HAS FINISHED INTEGRATING THE INPUT VECTORS.

IF INTEGRATION IN CHABRT IS SUCCESSFUL, THE REENTRY VECTOR TABLE NEEDED BY THE ACQUISITION DATA PROGRAM IS AVAILABLE AND MCESAB CONTAINS THE GMT OF THE FIRST EVEN SECOND AFTER BURNOUT. THUS, MABORT TURNS ON THE ''READY'' (B) BIT FOR THE ACQUISITION DATA PROGRAM AND TRANSFERS TO F).

IF NO REENTRY TABLE HAS BEEN PRODUCED BY CHABRT, MCESAB REMAINED ZERO. SINCE THE ACQUISITION DATA PROGRAM CANNOT BE ENTERED, MABORT RESETS MCESAB TO ITS INITIAL CONTENTS (ALL ONE'S) BEFORE EXITING TO F).

C) UPON RECEIPT OF A MANUAL INSERTION MESSAGE CONTAINING RETROFIRE INFORMATION DURING THE LOW ABORT PHASE, THE ABORT DEFINING VECTOR SUPPLIED BY THE LAUNCH TO LOW ABORT INTERFACE IS MOVED INTO THE REENTRY TABLE, TMRARF. THE BERMUDA HIGH-SPEED DATA EDIT PROGRAM IS SUPPRESSED AND THE BERMUDA ON-TRACK LIGHT AND THE MISSING DATA LIGHT ARE TURNED OFF. MABORT CONTINUES PROCESSING AS DESCRIBED IN E).

D) UPON RECEIPT OF A MANUAL INSERTION MESSAGE CONTAINING RETROFIRE INFORMATION DURING THE HIGH ABORT PHASE, MABORT DETERMINES WHETHER THE RETROFIRE SEQUENCE WAS NOMINAL OR NON-NOMINAL. IF A NON-NOMINAL SEQUENCE OCCURRED, THE TIMES OF RETROFIRE FOR THOSE RETROROCKETS WHICH DID FIRE ARE SORTED TO FIND THE EARLIEST TIME. IF THE RETROFIRE SEQUENCE WAS NOMINAL, THE EARLIEST TIME OF FIRING BECOMES THE TIME OF FIRING OF THE FIRST RETROROCKET.

USING THIS TIME AS THE GREENWICH MEAN TIME OF RETROFIRE, CHABRT IS ENTERED WITH A TYPE 4 ENTRY INDICATION TO INTEGRATE THE ABORT DEFINING VECTOR SUPPLIED BY THE LAUNCH TO HIGH ABORT INTERFACE TO THE TIME RETROFIRE. THE RESULTING VECTOR IS THEN PLACED IN THE REENTRY TABLE, TMRARF, AND CONTROL IS GIVEN TO E).

E) ALL HIGH-SPEED INPUT SUBCHANNELS OF THE DCC ARE DEACTIVATED (IP-7094, B-GE, AND BERMUDA). THE INPUT PROGRAMS WHICH PROCESS IP-7094 AND B-GE DATA ARE SUPPRESSED AND THE INITIAL PROGRAM INPUT FOR THE RE-ENTRY PHASE (I.E., THE VECTORS IN TMRARF) IS PRINTED ON-LINE. AFTER TURNING OFF ALL CONSOLE LIGHTS EXCEPT THE REENTRY PHASE LIGHT, MABORT SETS THE PHASE INDICATOR TO SHOW THAT THE MISSION HAS ENTERED THE RE-ENTRY PHASE. IT THEN CHANGES THE CYCLE FOR OUTPUT COMPUTATIONS AND DISPLAYS TO THREE SECONDS.

MABORT THEN EXAMINES THE CELL CONTAINING THE NUMBER OF RETROROCKETS FIRED, MCNRRF, TO INSURE THAT IT CONTAINS A NON-ZERO NUMBER. IF THIS CELL IS ZERO, MABORT PLACES A FIXED POINT FOUR INTO MCNRRF TO DISTINGUISH BETWEEN THE REENTRY AND ORBIT PHASES. THE FIRST PROGRAM TO GAIN CONTROL IN THE REENTRY PHASE, MYREN2, IS QUEUED WITH ITS INITIAL INPUT VECTORS AND MABORT EXITS TO MOPRIO AFTER SUPPRESSING ITSELF.

F) DURING THE LOW ABORT PHASE, THE IDENTIFICATION OF THE SOURCE FROM WHICH CURRENT DISPLAY QUANTITIES WERE COMPUTED IS MOVED INTO THE DISPLAY TABLE. IN THE HIGH ABORT PHASE, THIS CELL IS SET TO AND REMAINS ZERO.

THE OUTPUT PROCESSOR IS SUPPRESSED AND THE NEW DISPLAY QUANTITIES ARE MOVED INTO ITS INPUT BUFFER. THE TIMES OF LIFT-OFF AND OF THE CURRENT DATA ARE SAVED FOR THE LOGGING PROGRAM AND THE PHASE INDICATOR USED BY THE OUTPUT PROCESSOR IS SET NON-ZERO TO INDICATE THE ABORT PHASE.

IF THIS IS NOT THE FIRST PASS THROUGH MABORT, IMMEDIATE EXIT IS MADE TO MOPRIO. ON THE FIRST ENTRY TO MABORT, THE PREFIX OF THE CELL MCPHSE IS SET TO INDICATE WHICH TYPE OF ABORT SITUATION THE MISSION HAS ENTERED (1, IF HIGH ABORT, 2= IF LOW ABORT). DURING THE LOW ABORT PHASE, THIS CELL IS THE ONLY ADDITIONAL INDICATOR SET ON THE FIRST ENTRY, THEREFORE, MABORT UNSUPPRESSES THE OUTPUT PROGRAM AND EXITS TO MOPRIO.

ON THE FIRST ENTRY TO MABORT IN THE HIGH ABORT PHASE, THE PROGRAM WHICH COMPUTES ACQUISITION DATA TO BE SENT TO THE REMOTE TRACKING SITES IS UNSUPPRESSED AND MABORT CHECKS THE SETTING OF THE CELL MCESAB TO DETERMINE WHETHER THE PROGRAM CAN BE ENTERED. THE CELL INITIALLY CONTAINS ALL ONE'S. HOWEVER, THE HIGH ABORT PROCESSOR, CHABRT, SET THE CELL TO ZERO UNTIL IT HAS FINISHED INTEGRATING THE INPUT VECTORS.

IF INTEGRATION IN CHABRT WAS SUCCESSFUL, THE REENTRY VECTOR TABLE WHICH THE ACQUISITION DATA PROGRAM MUST HAVE HAS BEEN PRODUCED. AT THAT TIME, THE CELL MCESAB WAS SET TO THE GMT OF THE FIRST EVEN SECOND FOLLOWING BURNOUT. THUS, MABORT TURNS ON THE ''READY'' (B) BIT FOR THE ACQUISITION DATA PROGRAM, UNSUPPRESSES THE OUTPUT PROCESSOR AND EXITS TO MOPRIO.

IF NO REENTRY VECTOR TABLE WAS PRODUCED BY CHABRT, MCESAB REMAINED ZERO. SINCE THE ACQUISITION DATA PROGRAM CANNOT BE ENTERED AT THIS POINT, MABORT RESETS MCESAB TO ITS INITIAL CONTENTS (ALL ONE'S)

SO THAT THE REENTRY PROGRAMS WILL KNOW THAT ACQUISITION DATA HAS NOT BEEN COMPUTED AND SENT. CONTROL IS RETURNED TO MOPRIO AFTER THE OUTPUT PROCESSOR HAS BEEN UNSUPPRESSED.

USAGE

MABORT IS ENTERED FROM AND EXITS TO MOPRIO.

1. STORAGE REQUIRED - 749 LOCATIONS
2. TIMING (7094) -

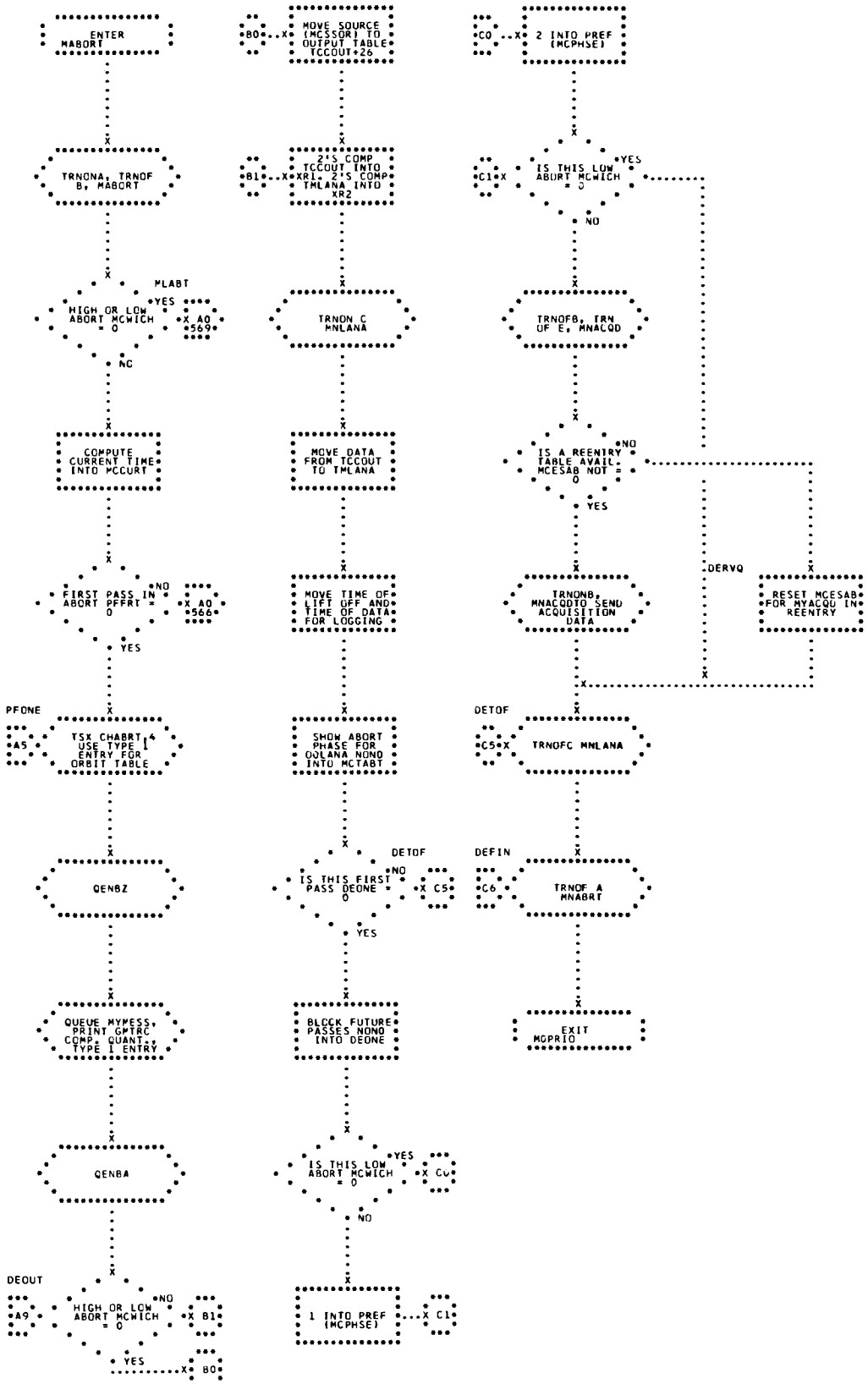
-

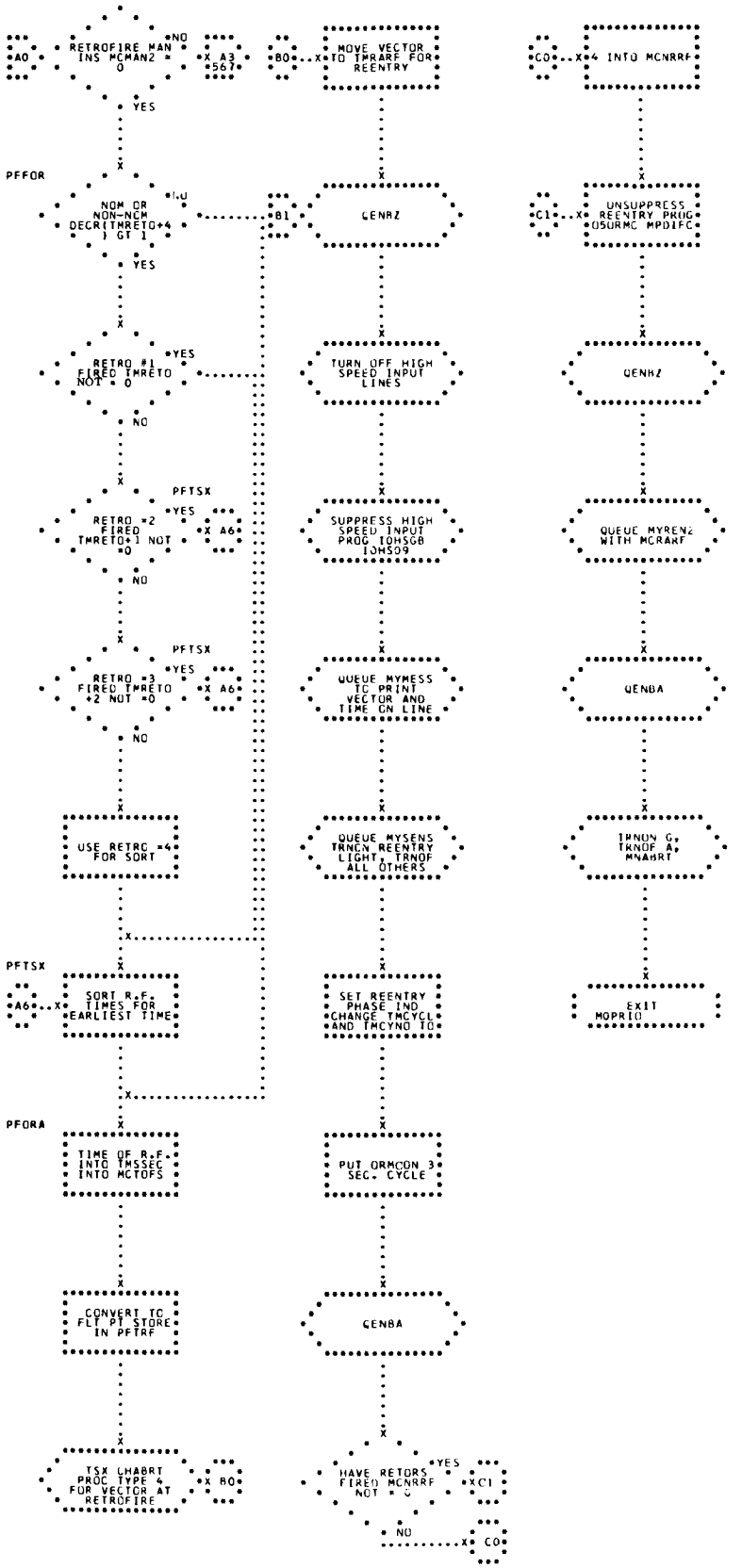
-

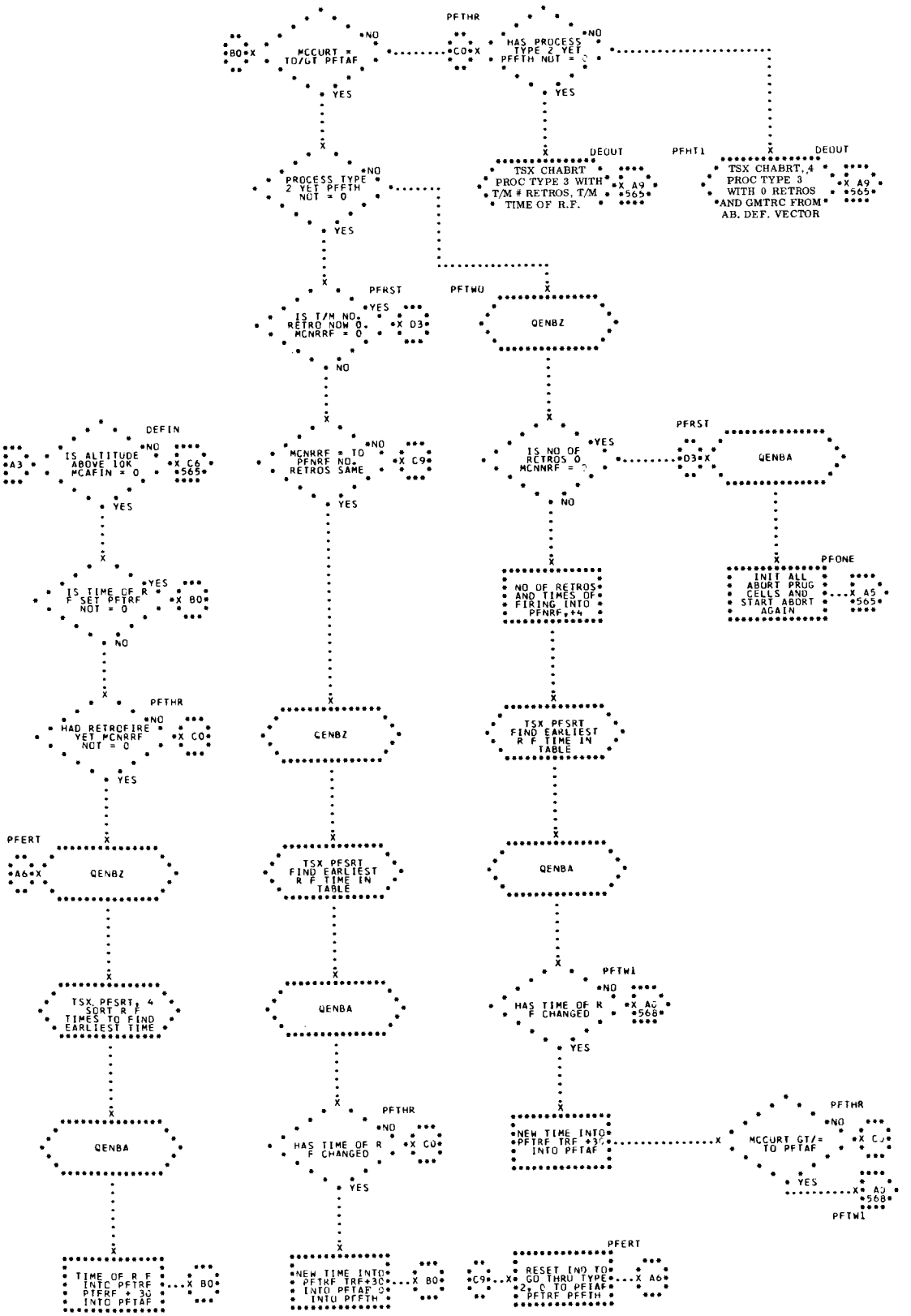
NOTE-ALL VECTOR TABLES ARE IN THE FOLLOWING FORMAT -

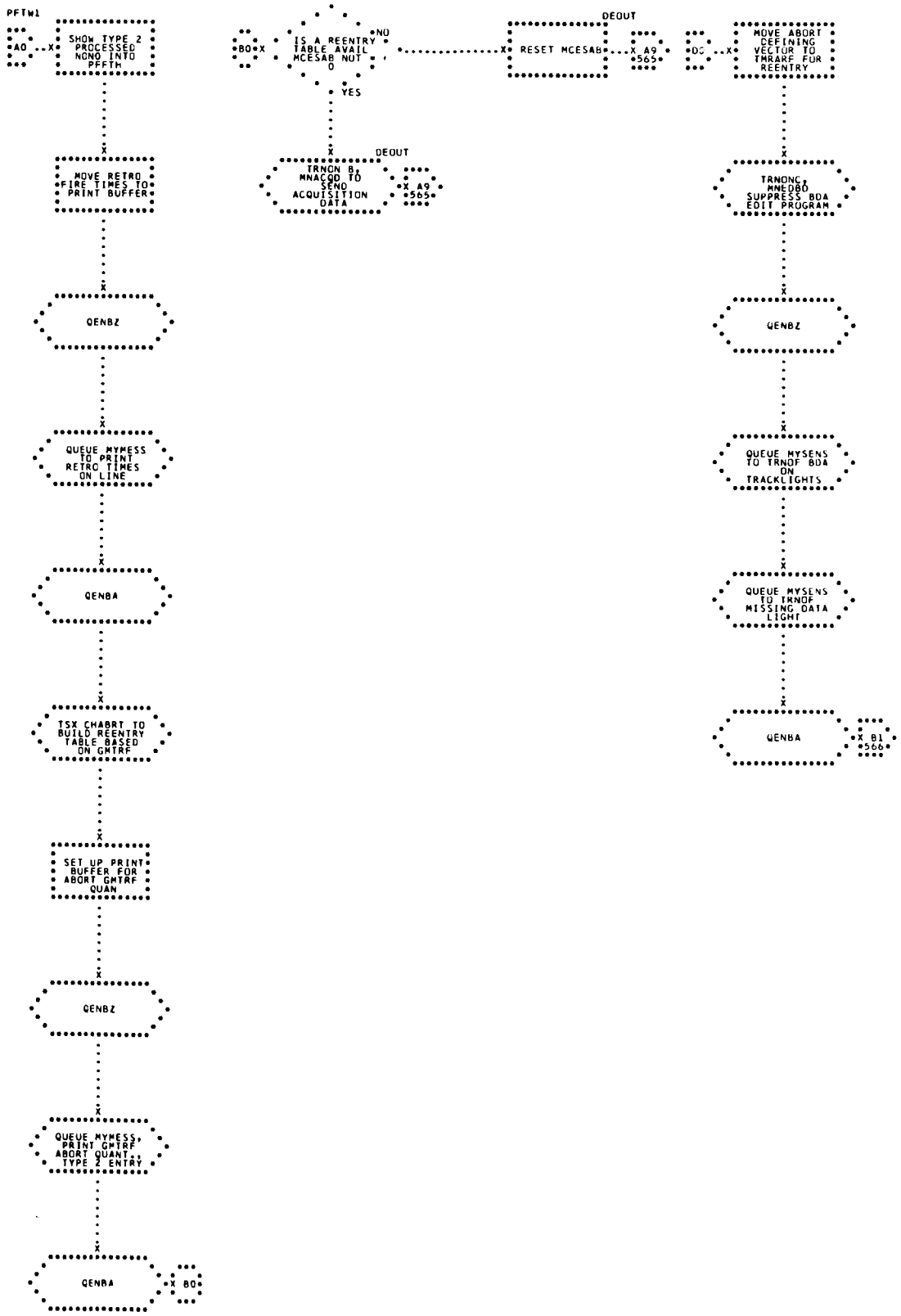
L(TABLE)	t	VECTOR TIME IN FLOATING POINT GMT SECONDS
+1	x	POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+2	y	
+3	z	
+4	$\dot{x}$	VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+5	$\dot{y}$	
+6	$\dot{z}$	

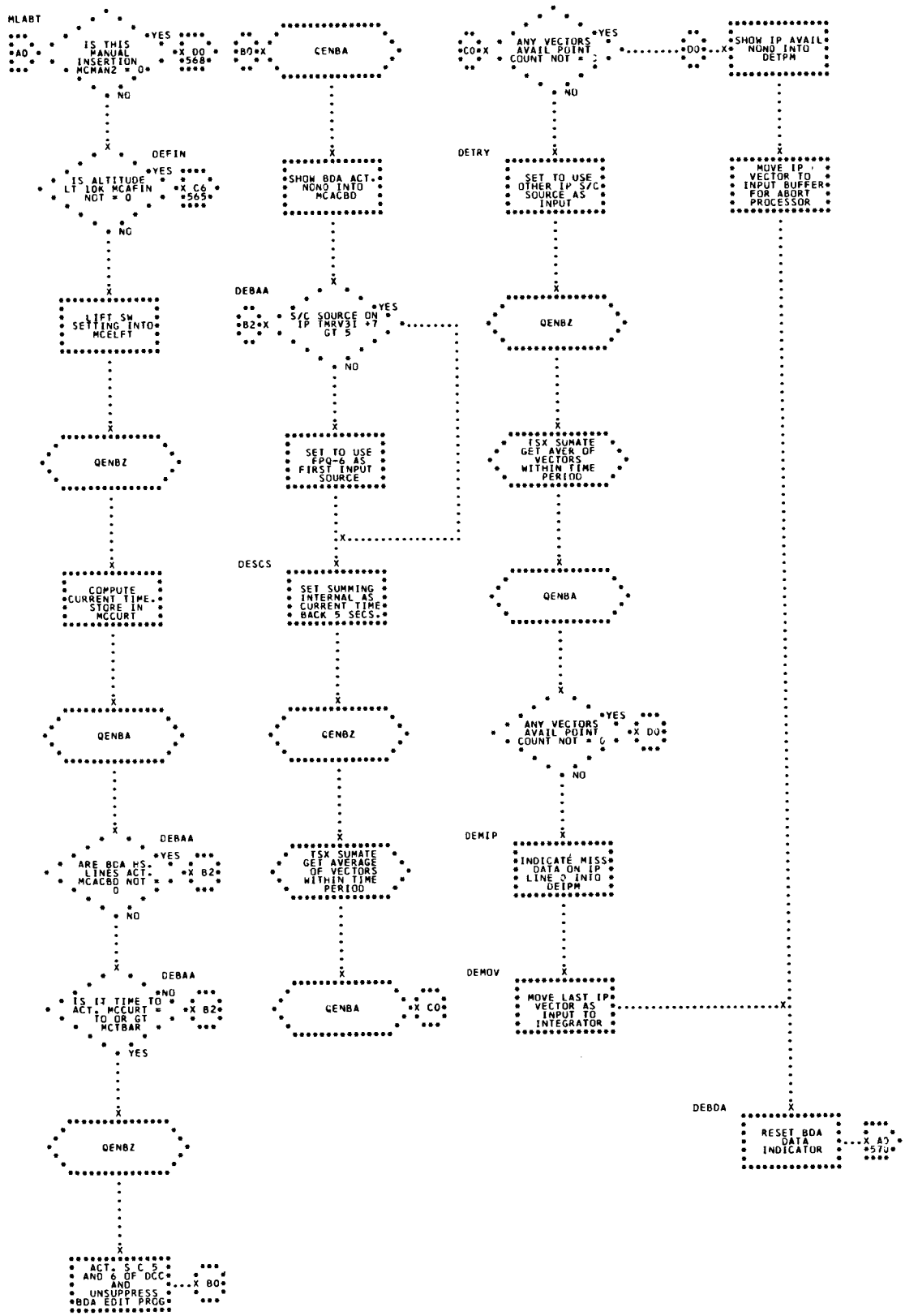
IN ADDITION, ALL TIMES, UNLESS OTHERWISE SPECIFIED, ARE IN FLOATING POINT GMT SECONDS.

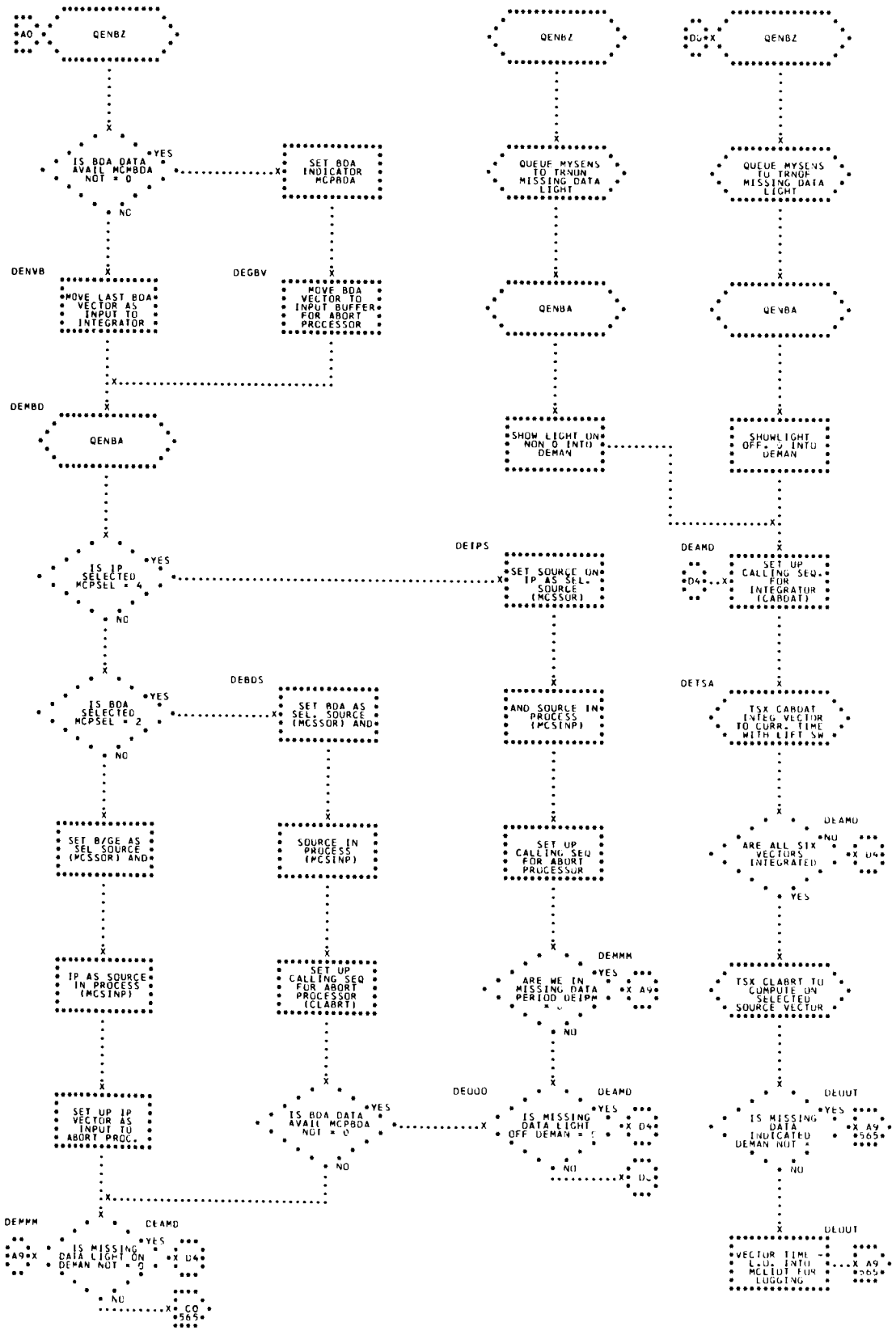

ENTER MABORT
TRNONA, TRNOF B, MABORT
HIGH OR LOW ABORT MCWICH = 0
YES
NO
MLABT
X AO 569
COMPUTE CURRENT TIME INTO MCCURT
FIRST PASS IN ABORT PFFRT = 0
NO
YES
X AO 566
PFONE
TSX CHABRT,4 USE TYPE 1 ENTRY FOR ORBIT TABLE
A5
QENBZ
QUEUE MYMESS, PRINT GMTRC COMP. QUANT., TYPE 1 ENTRY
QENBA
DEOUT
HIGH OR LOW ABORT MCWICH = 0
NO
YES
X B1
X B0
B0
MOVE SOURCE (MCSSOR) TO OUTPUT TABLE TCCOUT+26
B1
2'S COMP TCCOUT INTO XR1, 2'S COMP TMLANA INTO XR2
TRNON C MNLANA
MOVE DATA FROM TCCOUT TO TMLANA
MOVE TIME OF LIFT OFF AND TIME OF DATA FOR LOGGING
SHOW ABORT PHASE FOR OOLANA NONO INTO MCTABT
DETOF
IS THIS FIRST PASS DEONE = 0
NO
YES
X C5
BLOCK FUTURE PASSES NONO INTO DEONE
IS THIS LOW ABORT MCWICH = 0
YES
NO
X C4
1 INTO PREF (MCPHSE)
X C1
C0
2 INTO PREF (MCPHSE)
C1
IS THIS LOW ABORT MCWICH = 0
YES
NO
TRNOFB, TRN OF E, MNACQD
IS A REENTRY TABLE AVAIL- MCESAB NOT = 0
NO
YES
DERVQ
TRNONB, MNACQDTO SEND ACQUISITION DATA
RESET MCESAB FOR MYACQQ IN REENTRY
DETOF
C5
TRNOFC MNLANA
DEFIN
C6
TRNOF A MNABRT
EXIT MOPRIO


RETROFIRE MAN INS MCMAN2 = 0
NO
YES
A0
X A3 567
B0
MOVE VECTOR TO TMRARF FOR REENTRY
C0
4 INTO MCNRRF
PFFOR
NOM OR NON-NOM DECR(TMRETO+4) GT 1
4.0
B1
GENRZ
C1
UNSUPPRESS REENTRY PROG 050RMC MPDIFC
RETRO #1 FIRED TMRETO NOT = 0
TURN OFF HIGH SPEED INPUT LINES
QENRZ
PFTSX
RETRO =2 FIRED TMRETO+1 NOT =0
A6
SUPPRESS HIGH SPEED INPUT PROG IOHSGB IOHS09
QUEUE MYRENZ WITH MCRARF
RETRO =3 FIRED TMRETO +2 NOT =0
QUEUE MYMESS TO PRINT VECTOR AND TIME ON LINE
QENBA
USE RETRO =4 FOR SORT
QUEUE MYSENS TRNON REENTRY LIGHT, TRNOF ALL OTHERS
TRNON G, TRNOF A, MNABRT
SORT R.F. TIMES FOR EARLIEST TIME
SET REENTRY PHASE IND CHANGE TMCYCL AND TMCYNO TO
EXIT MOPRIO
PFORA
TIME OF R.F. INTO TMSSEC INTO MCTOFS
PUT ORMCON 3 SEC. CYCLE
CONVERT TO FLT PT STORE IN PFTRF
TSX LHABRT PROC TYPE 4 FOR VECTOR AT RETROFIRE
B0
HAVE RETORS FIRED MCNRRF NOT = 0
C1
NO
C0


MCCURT = TD/GT PFTAF
NO
YES
B0
X
PFTHR
C0
HAS PROCESS TYPE 2 YET PFFTH NOT = 0
PROCESS TYPE 2 YET PFFTH NOT = 0
DEOUT
TSX CHABRT PROC TYPE 3 WITH T/M # RETROS, T/M TIME OF R.F.
A9
565
PFHT1
TSX CHABRT,4 PROC TYPE 3 WITH 0 RETROS AND GMTRC FROM AB. DEF. VECTOR
PFRST
IS T/M NO. RETRO NOW 0. MCNRRF = 0
D3
PFTWO
QENBZ
DEFIN
IS ALTITUDE ABOVE 10K MCAFIN = 0
A3
C6
MCNRRF = TO PFNRF NO. RETROS SAME
C9
IS NO OF RETROS 0 MCNNRF = 0
QENBA
PFONE
INIT ALL ABORT PROG CELLS AND START ABORT AGAIN
A5
IS TIME OF R F SET PFTRF NOT = 0
NO OF RETROS AND TIMES OF FIRING INTO PFNRF,++
HAD RETROFIRE YET MCNRRF NOT = 0
TSX PFSRT FIND EARLIEST R F TIME IN TABLE
PFERT
A6
TSX PFSRT, 4 SORT R F TIMES TO FIND EARLIEST TIME
HAS TIME OF R F CHANGED
PFTW1
A0
568
NEW TIME INTO PFTRF TRF +30 INTO PFTAF
MCCURT GT/= TO PFTAF
TIME OF R F INTO PFTRF PTRF + 30 INTO PFTAF
NEW TIME INTO PFTRF TRF+30 INTO PFTAF + 30 INTO PFFTH
RESET IND TO GO THRU TYPE 2, 0 TO PFTAF PFTRF PFFTH


PFTM1
AO
SHOW TYPE 2 PROCESSED NONO INTO PFFTM
MOVE RETRO FIRE TIMES TO PRINT BUFFER
QENBZ
QUEUE MYMESS TO PRINT RETRO TIMES ON LINE
QENBA
TSX CHABRT TO BUILD REENTRY TABLE BASED ON GMTRF
SET UP PRINT BUFFER FOR ABORT GMTRF QUAN
QENBZ
QUEUE MYMESS, PRINT GMTRF ABORT QUANT., TYPE 2 ENTRY
QENBA
80
BO
IS A REENTRY TABLE AVAIL MCESAB NOT 0
NO
YES
DEQUT
TRNON B, MNACQD TO SEND ACQUISITION DATA
A9
565
DEQUT
RESET MCESAB
A9
565
DC
MOVE ABORT DEFINING VECTOR TO TMRARF FOR REENTRY
TRNONC, MNEDBD SUPPRESS BDA EDIT PROGRAM
QENBZ
QUEUE MYSENS TO TRNOF BDA ON TRACKLIGHTS
QUEUE MYSENS TO TRNOF MISSING DATA LIGHT
QENBA
B1
566


MLABT
IS THIS MANUAL INSERTION MCMAN2 = 0
YES
X DO 568
NO
DEFIN
IS ALTITUDE LT 10K MCAFIN NOT = 0
YES
X C6 565
NO
LIFT SW SETTING INTO MCELFT
QENBZ
COMPUTE CURRENT TIME. STORE IN MCCURT
QENBA
DEBAA
ARE BDA HS. LINES ACT. MCACBD NOT = 0
YES
X B2
NO
DEBAA
IS IT TIME TO ACT. MCCURT = TO OR GT MCTBAR
NO
X B2
YES
QENBZ
ACT. S C 5 AND 6 OF DCC AND UNSUPPRESS BDA EDIT PROG
X BO
BO X
QENBA
SHOW BDA ACT. NONO INTO MCACBD
DEBAA
S/C SOURCE ON IP TMRV3I +7 GT 5
YES
B2 X
NO
SET TO USE FPQ-6 AS FIRST INPUT SOURCE
DESCS
SET SUMMING INTERNAL AS CURRENT TIME BACK 5 SECS.
QENBZ
ISX SUMATE GET AVERAGE OF VECTORS WITHIN TIME PERIOD
QENBA
X CO
CO X
ANY VECTORS AVAIL POINT COUNT NOT = 0
YES
X DO
NO
DETRY
SET TO USE OTHER IP S/C SOURCE AS INPUT
QENBZ
ISX SUMATE GET AVER OF VECTORS WITHIN TIME PERIOD
QENBA
ANY VECTORS AVAIL POINT COUNT NOT = 0
YES
X DO
NO
DEMIP
INDICATE MISS DATA ON IP LINE 2 INTO DEIPM
DEMOV
MOVE LAST IP VECTOR AS INPUT TO INTEGRATOR
DO X
SHOW IP AVAIL NONO INTO DETPM
MOVE IP VECTOR TO INPUT BUFFER FOR ABORT PROCESSOR
DEBDA
RESET BDA DATA INDICATOR
X AO
570


AO
QENBZ
IS BDA DATA AVAIL MCMBDA NOT = 0
YES
NO
SET BDA INDICATOR MCPBDA
DENVB
MOVE LAST BDA VECTOR AS INPUT TO INTEGRATOR
DEGBV
MOVE BDA VECTOR TO INPUT BUFFER FOR ABORT PROCESSOR
DEMBD
QENBA
IS IP SELECTED MCPSEL = 4
IS BDA SELECTED MCPSEL = 2
DEBDS
SET BDA AS SEL. SOURCE (MCSSOR) AND
SET B/GE AS SEL SOURCE (MCSSOR) AND
SOURCE IN PROCESS (MCSINP)
IP AS SOURCE IN PROCESS (MCSINP)
SET UP CALLING SEQ FOR ABORT PROCESSOR (CLABRT)
SET UP IP VECTOR AS INPUT TO ABORT PROC.
IS BDA DATA AVAIL MCPBDA NOT = 0
DEMMM
DEAMD
IS MISSING DATA LIGHT DEMAN NOT = 0
D4
A9
CO
565
QENBZ
QUEUE MYSENS TO TRNON MISSING DATA LIGHT
QENBA
SHOW LIGHT ON NON 0 INTO DEMAN
DEIPS
SET SOURCE ON IP AS SEL. SOURCE (MCSSOR)
AND SOURCE IN PROCESS (MCSINP)
SET UP CALLING SEQ FOR ABORT PROCESSOR
DEQOO
IS MISSING DATA LIGHT OFF DEMAN = 0
QENBZ
QUEUE MYSENS TO TRNOF MISSING DATA LIGHT
QENBA
SHOWLIGHT OFF, 0 INTO DEMAN
DEAMD
SET UP CALLING SEQ. FOR INTEGRATOR (CABDAT)
DETSA
TSX CABDAT INTEG VECTOR TO CURR. TIME WITH LIFT SW.
ARE ALL SIX VECTORS INTEGRATED
NO
YES
TSX CLABRT TO COMPUTE ON SELECTED SOURCE VECTOR
DEOUT
IS MISSING DATA INDICATED DEMAN NOT =
VECTOR TIME - L.O. INTO MCLIDT FOR LOGGING

QUANTITIES OUTPUT BY ABORT PROGRAM ARE MOVED FROM TCCOUT TO TMLANA IN THE FOLLOWING FORMAT -

TCCOUT	QUANTITY	TMLANA
+0	LATITUDE-PP	+5
+1	LONGITUDE-PP	+6
+2	DOWNRANGE DIST.	+24
+3	CROSSRANGE DIST.	+23
+4	ALTITUDE	+25
+5	HEIGHT	+19
+6	GAMMA	+20
+7	INCLINATION ANG	+21
+8	VELOCITY	+22
+9	GMTRC	+13
+10	ESTRC	+15
+11	GTRS/EGT	+9
+18	GMTLC	+10
+19	LATITUDE-IP	+7
+20	LONGITUDE-IP	+8
+21	LAT. - MIN. LIFT	+11
+22	LONG. - MIN. LIFT	+12
+23	LAT. - MAX LIFT	+0
+24	LONG. - MAX LIFT	+1
+25	RECOVERY AREA	+17
+26	DATA SOURCE	+4

65. CLABRT - LOW ABORT PROCESSOR

THIS PROGRAM IS USED DURING THE ABORT PHASE FOR ABORTS PRIOR TO AMC (ABORT MODE CHANGEOVER) TO PRODUCE THE REQUIRED OUTPUT DISPLAY QUANTITIES. (SEE FLOWCHART).

INPUT

A) TABLES -

L(INPUT)			A SEVEN-LOCATION TABLE CONTAINING THE INPUT VECTORS AND THEIR ASSOCIATED TIME.
L(INPUT)		t	TIME OF VECTORS IN GMT FLOATING POINT SECONDS
+1		x	POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+2	y	y	
+3		z	
+4		$\dot{x}$	VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+5		$\dot{y}$	
+6		$\dot{z}$	
L(MISSING DATA INDICATOR)			A ONE-CELL TABLE CONTAINING THE INDICATION OF WHETHER DATA IS MISSING ON THE CURRENT ENTRY
L(LIFT SWITCH SETTING)			A ONE-CELL TABLE CONTAINING THE SETTING OF THE LIFT SWITCH
TCAHC2			A ONE-CELL TABLE CON-

	TAINING THE NOMINAL LAUNCH/ABORT IMPACT HEIGHT IN FLOATING POINT GEMINI LENGTH UNITS
TCRADF	A ONE-CELL TABLE CONTAINING THE OCTAL CODE OF RECOVERY AREA F
TCRADG	A ONE-CELL TABLE CONTAINING THE OCTAL CODE OF RECOVERY AREA G
TCRALF	A ONE-CELL TABLE CONTAINING THE LONGITUDE OF RECOVERY AREA F IN FLOATING POINT RADIANS
TCRALG	A ONE-CELL TABLE CONTAINING THE LONGITUDE OF RECOVERY AREA G IN FLOATING POINT RADIANS
TCRCAD	A SEVEN-LOCATION TABLE CONTAINING THE OCTAL CODES OF THE RECOVERY AREAS
TCRCAL	A NINE-LOCATION TABLE CONTAINING THE LONGITUDES OF THE RECOVERY AREAS IN FLOATING POINT RADIANS
TCLAFE	A ONE-CELL TABLE CONTAINING THE LOW ABORT FIRST ENTRY INDICATION. SET NON-ZERO

			AFTER FIRST ENTRY TO LOW ABORT MADE.
	TCICRT		A ONE-CELL TABLE CONTAINING THE INCLINATION ANGLE CRITERION IN FLOATING POINT RADIANS
	TCBAM2		A ONE-CELL TABLE CONTAINING THE PROGRAMMED BANK ANGLE FOR ABORT MODE 2 IN FLOATING POINT RADIANS
	TCFFCT		A ONE-CELL TABLE CONTAINING THE DELAY TIME UNTIL LANDING IN FLOATING POINT SECONDS
B)	CONSTANTS -		
	KM0000	DEC -0	FIXED POINT MINUS ZERO
	K00000	DEC 0	FIXED POINT ZERO
	K00001	DEC 1	FIXED POINT ONE
	K00007	DEC 7	FIXED POINT SEVEN
	K00100	DEC 100	FIXED POINT ONE HUNDRED
	K000.5	DEC 0.5	FLOATING POINT ONE-HALF
	K00.PI		PI IN FLOATING POINT RADIANS
	K03HPI		3/2 PI IN FLOATING POINT RADIANS
	KKK2PI		2 PI IN FLOATING POINT RADIANS
	KCGUNM		CONVERSION OF GEMINI

	LENGTH UNITS TO NAUTICAL MILES IN FLOATING POINT

OUTPUT

A) TABLES -

L(OUTPUT)	A TWENTY-SIX LOCATION OUTPUT TABLE CONTAINING THE REQUIRED OUTPUT DISPLAY QUANTITIES PRODUCED FROM THE INPUT VECTORS AND THEIR ASSOCIATED TIME (SEE TABLE)
TCLAFE	SET NON-ZERO WHEN FIRST ENTRY TO LOW ABORT MADE
TCRCAD+5	THE CODE OF EITHER RECOVERY AREA F OR G REPLACES THE CONTENTS OF THE SIXTH LOCATION OF THE TCRCAD TABLE
TCRCAL+6	THE LONGITUDE OF EITHER RECOVERY AREA F OR G REPLACES THE CONTENTS OF THE SEVENTH LOCATION OF THE TCRCAL TABLE

B) COMMUNICATION CELL -

MCAFIN	SET NON ZERO WHEN PRESENT POSITION FALLS BELOW THE NOMINAL

LAUNCH/ABORT IMPACT
HEIGHT SPECIFIED IN
TCAHC2

METHOD

AFTER INITIALIZING, CLABRT ENTERS THE SUBROUTINE CPRVCM TO COMPUTE ALL PRESENT VECTORS QUANTITIES FROM THE INPUT VECTORS AND THEIR ASSOCIATED TIME. THE RESULSTS ARE THEN PLACED IN THE SPECIFIED OUTPUT TABLE.

THE FOLLOWING QUANTITITES ARE SET TO ZERO - (1) THE TIME OF RETROFIRE IN GMT AND IN ELAPSED TIME SINCE LIFT-OFF (CALLED GMTRC AND ESTRC, RESPECTIVELY), (2) THE ELAPSED TIME SINCE RETROFIRE (CALLED EGT), (3) THE TIME IN GMT, THE RELATIVE VELOCITY MAGNITUDE, AND THE RELATIVE FLIGHT PATH ANGLE AT THE HEIGHT DEFINING THE BEGINNING OF THE ATMOSPHERE, (4) THE TIMES IN GMT AT THE BEGINNING AND AT THE END OF THE BLACKOUT PERIOD, AND (5) THE TIME IN GMT OF ARRIVAL AT THE NUMINAL LAUNCH/ABORT IMPACT HEIGHT.

THE LAUNCH NUMERICAL INTEGRATICN PROGRAM, CORKNI, IS ENTERED TWICE TO COMPUTE THE LATITUDE AND LONGITUDE OF IMPACT USING, FIRST, A MINIMUM LIFT CONFIGURATION, AND SECOND, A MAXIMUM LIFT CONFIGURATION. IF BOTH ENTRIES TO THE INTEGRATOR ARE SUCCESSFUL, THE COMPUTED IMPACT POINTS ARE PLACED IN THE OUTPUT TABLE. IF EITHER ENTRY FAILS, BOTH LATITUDES ARE SET TO ZERO AND BOTH LONGITUDES ARE SET TC PI RADIANS.

BOTH ENTRIES TO THE INTEGRATOR ARE SUCCESFUL, THE COMPUTED IMPACT POINTS ARE PLACED IN THE OUTPUT TABLE. IF EITHER ENTRY FAILS, BOTH LATITUDES ARE SET TO ZERO AND BOTH LONGITUDES ARE SET TO PI RADIANS.

SUBSEQUENT TO THE FIRST ENTRY, A MISSING DATA CHECK IS MADE, AND IN THE EVENT THAT DATA IS MISSING, THE ABOVE IMPACT PREDICTION IS BYPASSED. ON THE FIRST ENTRY, THE FIRST ENTRY INDICATOR IS SET NON-ZERO AND THE ABOVE IMPACT PREDICTION IS MADE REGARDLESS OF WHETHER DATA IS MISSING OR NOT.

THE SETTING OF THE LIFT SWITCH IS EXAMINED TO SELECT THE PROPER BANK ANGLE. ONCE THE BANK ANGLE HAS BEEN CHOSEN, IT DICTATES THE LIFT CONFIGURATION TO BE USED BY CORKNI IN COMPUTING THE LATITUDE AND THE LONGITUDE OF ACTUAL IMPACT.

ON A NORMAL RETURN FROM THE INTEGRATOR, THE TIME OF LANDING IN GMT (CALLED GMTLC) IS COMPUTED BY ADDING A DELAY FACTOR TO THE TIME ASSOCIATED WITH THE ACTUAL IMPACT POINT. THE TIME OF THE IMPACT POINT, GMTLC AND THE IMPACT POINT ARE PLACED IN THE OUTPUT TABLE.

BASED ON THE VALUE OF THE INCLINATION ANGLE COMPUTED BY CPRVCM, THE CODE AND LONGITUDE OF EITHER RECOVERY AREA F OR G ARE PLACED IN THE TCRCAD AND TCRCAL TABLES, RESPECTIVELY. THEN, THE CODE AND LONGITUDE OF THE RECOVERY AREA CLOSEST TO THE LONGITUDE OF ACTUAL IMPACT ARE STORED AS THE CODE AND LONGITUDE OF THE SELECTED RECOVERY AREA.

ON AN ERROR RETURN FROM THE INTEGRATOR IN ATTEMPTING TO COMPUTE THE ACTUAL IMPACT POINT, THE LATITUDE OF IMPACT AND THE GMTLC ARE SET TO ZERO, THE LONGITUDE OF IMPACT IS SET TO PI RADIANS, AND THEN THE SELECTED RECOVERY AREA BECOMES THE FICTITIOUS AREA, H.

FINALLY, CLABRT DETERMINES WHETHER THE ABORT PHASE IS FINISHED BY COMPARING THE COMPUTED HEIGHT OF THE SPACECRAFT ABOVE AN OBLATE EARTH TO THE NOMINAL LAUNCH/ABORT IMPACT HEIGHT. THE CELL MCAFIN IS THEN SET ACCORDINGLY BEFORE CLABRT RETURNS TO THE MAIN PROGRAM.

USAGE

A) CALLING SEQUENCE -

```
TSX          CLABRT,4
PZE          L(INPUT),,L(OUTPUT)
PZE          L(MISSING DATA INDICATOR),,
             L(LIFT SWITCH SETTING)
RETURN
```

B) STORAGE REQUIRED -

171 LOCATIONS

C) TIMING (7094) -

6.278 MILLISECONDS, MAXIMUM (EXCLUDING TIME REQUIRED FOR CORKNI)

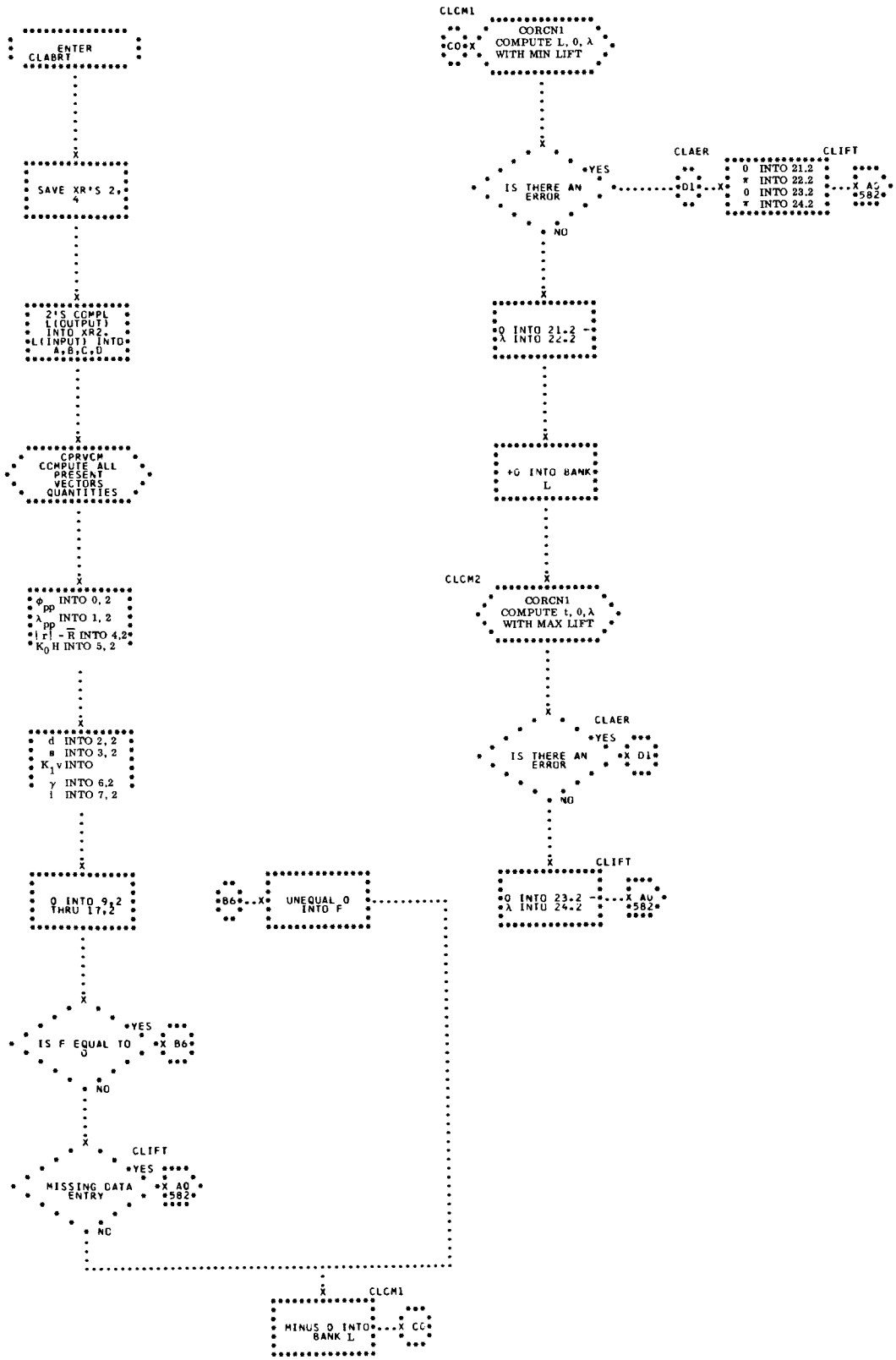
TABLE 65-1. OUTPUT TABLE FOR CLABRT

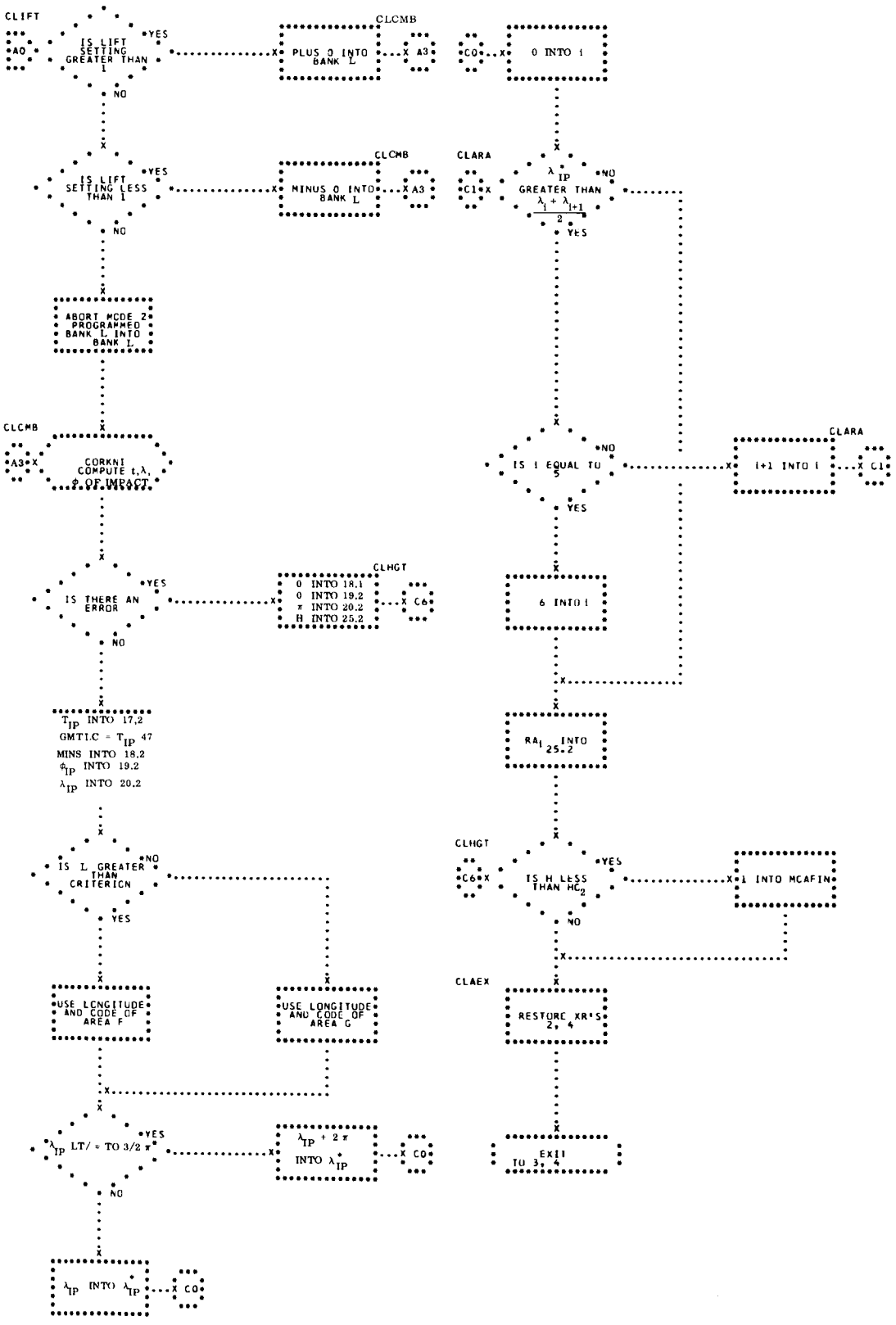
LOCATION		DESCRIPTION	UNITS
L(OUTPUT)		LATITUDE OF PRESENT POSITION	RADIANS
	+1	LONGITUDE OF PRESENT POSITION	RADIANS
	+2	DOWNRANGE DISTANCE	NAUTICAL MILES
	+3	CROSSRANGE DISTANCE	NAUTICAL MILES
	+4	HEIGHT OF SPACECRAFT ABOVE A SPHERICAL EARTH	NAUTICAL MILES
	+5	HEIGHT OF SPACECRAFT ABOVE AN OBLATE EARTH	NAUTICAL MILES
	+6	FLIGHT PATH ANGLE	RADIANS
	+7	INCLINATION ANGLE	RADIANS
	+8	VELOCITY MAGNITUDE	FEET/SECOND
	+9	0	
	+10	0	
	+11	0	
	+12	0	
		0	
	+14	0	
	+15	0	
	+16	0	
	+17	GMT OF ARRIVAL AT NOMINAL ABORT IMPACT HEIGHT	SECONDS
	+18	GMT OF LANDING COMPUTED	SECONDS
	+19	LATITUDE OF IMPACT	RADIANS
	+20	LONGITUDE OF IMPACT	RADIANS

TABLE 65-1. OUTPUT TABLE FOR CLABRT

LOCATION	DESCRIPTION	UNITS
+21	LATITUDE OF IMPACT USING A MINIMUM LIFT CONFIGURATION	RADIANS
+22	LONGITUDE OF IMPACT USING A MINIMUM LIFT CONFIGURATION	RADIANS
+23	LATITUDE OF IMPACT USING A MAXIMUM LIFT CONFIGURATION	RADIANS
+24	LONGITUDE OF IMPACT USING A MAXIMUM LIFT CONFIGURATION	RADIANS
+25	RECOVERY AREA	SPECIAL OCTAL CODE

(NOTE - ALL QUANTITIES ARE IN FLOATING POINT UNLESS OTHERWISE NOTED.)


ENTER CLABRT
SAVE XR*S 2, 4
2'S COMPL L(OUTPUT) INTO xR2, L(INPUT) INTO A,B,C,D
CPRVCM COMPUTE ALL PRESENT VECTORS QUANTITIES
ϕ_{pp} INTO 0, 2
λ_{pp} INTO 1, 2
|r| - R INTO 4,2
K_0H INTO 5, 2
d INTO 2, 2
s INTO 3, 2
K_1v INTO
γ INTO 6,2
i INTO 7, 2
0 INTO 9,2 THRU 17,2
IS F EQUAL TO 0
YES
NO
B6
MISSING DATA ENTRY
CLIFT
A0 582
UNEQUAL 0 INTO F
MINUS 0 INTO BANK L
CC
CLCM1
CORCN1 COMPUTE L, 0, λ WITH MIN LIFT
IS THERE AN ERROR
CLAER
D1
0 INTO 21.2
π INTO 22.2
0 INTO 23.2
π INTO 24.2
A0 582
+0 INTO 21.2
+λ INTO 22.2
+0 INTO BANK L
CLCM2
CORCN1 COMPUTE t, 0, λ WITH MAX LIFT
0 INTO 23.2
λ INTO 24.2
A0 582


CLIFT
A0
IS LIFT SETTING GREATER THAN 1
YES
NO
CLCMB
PLUS 0 INTO BANK L
A3
C0
0 INTO i
IS LIFT SETTING LESS THAN 1
YES
NO
CLCMB
MINUS 0 INTO BANK L
A3
CLARA
C1
λ_{IP} GREATER THAN $\frac{\lambda_i + \lambda_{i+1}}{2}$
NO
YES
ABORT MODE 2 PROGRAMMED BANK L INTO BANK L
CLCMB
A3
CORKNI COMPUTE t,λ, Φ OF IMPACT
IS THERE AN ERROR
YES
NO
CLHGT
0 INTO 18.1
0 INTO 19.2
π INTO 20.2
H INTO 25.2
C6
T_{IP} INTO 17,2
GMTLC = T_{IP} 47
MINS INTO 18.2
Φ_{IP} INTO 19.2
λ_{IP} INTO 20.2
IS L GREATER THAN CRITERION
NO
YES
USE LONGITUDE AND CODE OF AREA F
USE LONGITUDE AND CODE OF AREA G
λ_{IP} LT/ = TO 3/2 π
YES
NO
λ_{IP} + 2 π INTO λ_{IP}
C0
λ_{IP} INTO λ_{IP}
C0
IS i EQUAL TO 5
NO
YES
CLARA
i+1 INTO i
C1
6 INTO i
$RA_{i\ 25.2}$ INTO
CLHGT
C6
IS H LESS THAN HC_2
YES
NO
1 INTO MCAFIN
CLAEX
RESTORE XR'S 2, 4
EXIT TO 3, 4

66. CHABRT - LAUNCH PROCESSOR FOR HIGH ABORT

THIS PROGRAM IS USED TO GENERATE THE REQUIRED OUTPUT DISPLAY QUANTITIES WHEN A DECISION IS MADE TO ENTER THE ABORT PHASE AFTER THE HOLD PHASE DISPLAYS BASED ON THE AVERAGE VECTORS COMPUTATIONS HAVE BEEN EVALUATED. (SEE FLOWCHART.)

INPUT

A) TABLES

L(INPUT) -	A SEVEN-LOCATION TABLE CONTAINING THE INPUT AVERAGE VECTORS FROM THE SOURCE SELECTED WHEN THE ABORT SWITCH WAS THROWN
L(TYPE ENTRY)	CONTAINS THE TYPE OF ENTRY BEING MADE TO THE PROGRAM IN FIXED POINT, B35
L(T/M INDICATED RETROFIRE INFORMATION)	A FIVE-LOCATION TABLE CONTAINING IN THE FIRST CELL THE NUMBER OF RETROROCKETS FIRED IN FIXED POINT, B35, IN THE SECOND THROUGH FIFTH CELLS THE TIMES OF FIRING OF RETROROCKETS 1,3,2, AND 4, IN THAT ORDER, IN FLOATING POINT GMT SECONDS. THE TIME OF ANY RETRO-ROCKETS THAT DOES NOT FIRE IS SET TO ZERO.
L(CURRENT TIME)	CONTAINS CURRENT TIME IN FLOATING POINT GMT SECONDS
L(GMTRC OR GMTRF)	CONTAINS THE TIME OF RETROFIRE IN GMT FLOATING POINT SECONDS,

	EITHER COMPUTED OR FROM TELEMETRY
L(LIFT SWITCH SETTING)	CONTAINS THE SETTING OF THE LIFT SWITCH, WHICH MAY BE 0,1, OR 2 IMPLYING THE USE OF MINIMUM, PROGRAMMED, OR MAXIMUM LIFT, RESPECTIVELY, IN FIXED POINT, B35
TCAHC2	CONTAINS THE NOMINAL ABORT IMPACT HEIGHT IN FLOATING POINT GEMINI LENGTH UNITS.
TCAHC1	CONTAINS THE HEIGHT DEFINING THE BEGINNING OF THE ATMOSPHERE IN FLOATING POINT GEMINI LENGTH UNITS
TCBAM3	CONTAINS THE ABORT MODE 3 PROGRAMMED BANK ANGLE IN FLOATING POINT RADIANS
TCERR3	ZERO IF THE REENTRY HEIGHT AND VECTOR TABLES HAVE BEEN GENERATED, OTHERWISE, NON-ZERO
TCERR4	ZERO IS THE ABORT ORBIT VECTOR TABLE HAS BEEN GENERATED, OTHERWISE, NON-ZERO
TCFFCT	CONTAINS THE DELAY TIME FROM THE TIME OF ARRIVAL AT THE NOMINAL ABORT IMPACT HEIGHT TO THE TIME OF LANDING IN FLOATING POINT SECONDS
TCHBO1	CONTAINS THE HEIGHT AT THE BEGINNING OF THE BLACKOUT PERIOD IN FLOATING POINT GEMINI LENGTH UNITS
TCHBO2	CONTAINS THE HEIGHT AT THE END

	OF THE BLACKOUT PERIOD IN FLOATING POINT GEMINI LENGTH UNITS
TCRCAD	A SEVEN-LOCATION TABLE CONTAINING THE CODES OF THE RECOVERY AREAS IN OCTAL
TCRCAL	A NINE-LOCATION TABLE CONTAINING THE LONGITUDES OF THE RECOVERY AREAS IN FLOATING POINT RADIANS
TCRFTQ	ZERO IF THE INERTIAL PLATFORM WILL NOT BE TORQUED DURING RETROFIRE
TCRKOH	THE ABORT/REENTRY HEIGHT TABLE ARRANGED AS FOLLOWS - t_{start} t_{stop} , H_1, H_2, H_3, ETC., WHERE t_{start} IS THE TIME ASSOCIATED WITH THE FIRST HEIGHT ENTRY IN THE TABLE, t_{stop} IS THE TIME ASSOCIATED WITH THE LAST HEIGHT ENTRY IN THE TABLE, AND H_i ARE THE HEIGHT ENTRIES IN THE TABLE. ALL TIMES ARE IN GMT FLOATING POINT SECONDS. HEIGHT ENTRIES ARE IN FLOATING POINT GEMINI LENGTH UNITS.
TCRKOT	THE ABORT ORBIT VECTOR TABLE ARRANGED AS FOLLOWS - t_{start} , t_{stop} , $\bar{r}_1, \bar{v}_1, \bar{r}_2, \bar{v}_2$, ETC., WHERE t_{start} IS THE TIME ASSOCIATED WITH THE FIRST POSITION, VELOCITY VECTOR ENTRIES IN THE TABLE, t_{stop} IS THE TIME ASSOCIATED WITH THE LAST POSITION, VELOCITY VECTOR

		ENTRIES IN THE TABLE, AND $\overline{r}_i, \overline{v}_i$ ARE THE POSITION AND VELOCITY VECTOR ENTRIES. ALL TIMES ARE IN FLOATING POINT GMT SECONDS, $\overline{r}_i$ ARE IN FLOATING POINT GEMINI LENGTH UNITS, $\overline{v}_i$ ARE IN FLOATING POINT GEMINI VELOCITY UNITS
	TCRKRT	THE ABORT REENTRY VECTOR TABLE ARRANGED AS TCRKOT.
	TCRRPY	A THREE-LOCATION TABLE CONTAINING THE ROLL, PITCH AND YAW ANGLES FOR USE IN NON-NOMINAL RETROFIRE SEQUENCES IN FLOATING POINT RADIANS
	TCWF.B	THE ORBIT SPACECRAFT WEIGHT IN FLOATING POINT POUNDS
	TCWF.R	THE REENTRY SPACECRAFT WEIGHT IN FLOATING POINT POUNDS
	TC6RCW	THE WEIGHT OF THE SPACECRAFT AT THE BEGINNING OF RETROFIRE IN FLOATING POINT POUNDS.
	TRKO1H	THE INTERVAL BETWEEN POSITION, VELOCITY VECTOR ENTRIES IN TCRKOT IN FLOATING POINT SECONDS
	TRKR1H	THE INTERVAL USED IN TCRKRT AND TCRKOH IN FLOATING POINT SECONDS.
	TC200K	200,000 FEET IN GEMINI LENGTH UNITS, FLOATING POINT
B)	CONSTANTS -	
	K00000	DEC 0 FIXED POINT ZERO
	K00001	DEC 1 FIXED POINT ONE
	K00002	DEC 2 FIXED POINT TWO

K00003	DEC 3	FIXED POINT THREE
K00004	DEC 4	FIXED POINT FOUR
K00007	DEC 7	FIXED POINT SEVEN
K00030	DEC 30	FIXED POINT THIRTY
K00100	DEC 100	FIXED POINT HUNDRED
KK0360	DEC 360	FIXED POINT THREE HUNDRED SIXTY
K03600	DEC 3600	FIXED POINT THIRTY-SIX HUNDRED
KM0000	DEC -0	FIXED POINT MINUS ZERO
K000.5	DEC 0.5	FLOATING POINT ONE-HALF
K005.5	DEC 5.5	FLOATING POINT FIVE AND ONE-HALF
K015.0	DEC 15.0	FLOATING POINT FIFTEEN
K00.PI		PI IN FLOATING POINT RADIANS
K03HPI		THREE HALVES PI IN FLOATING POINT RADIANS
KKK2PI		TWICE PI IN FLOATING POINT RADIANS
KCGUNM		CONVERSION OF GEMINI LENGTH UNITS TO NAUTICAL MILES, FLOATING POINT
KFTSCG		CONVERSION OF GEMINI VELOCITY UNITS TO FEET/ SECOND IN FLOATING POINT
K.OMEG		ROTATION OF EARTH IN RADIANS PER GEMINI TIME UNIT, FLOATING POINT
K.MOMG		NEGATIVE ROTATION OF EARTH IN RADIANS PER GEMINI TIME UNIT, FLOAT-

ING POINT

KCH233 OCT 233000000000
CHARACTERISTIC OF 233 OCTAL

C) COMMUNICTAION CELL -

MCLFTM TIME OF LIFT-OFF IN FLOATING POINT SECONDS

D) SUBROUTINES -

CHABRT USES THE SUBROUTINES CTSFT3, CORKNI, C6PTLI, CABDAT, C6RETC, C2TAFH, CPRVCM, A3GP12 AND THE LIBRARY SUBROUTINES U1VMAG AND U1ASCO IN ITS COMPUTATIONS.

OUTPUT

THE OUTPUT PRODUCED BY CHABRT DEPENDS UPON THE TYPE OF ENTRY MADE TO THE PROCESSOR -

TYPE 1

A) TABLES -

TCRKOH THE ABORT REENTRY HEIGHT TABLE

TCRKOT THE ABORT ORBIT VECTOR TABLE

TCRKRT THE ABORT REENTRY VECTOR TABLE

L(OUTPUT) THE COMPLETE TWENTY-SIX LOCATION OUTPUT TABLE CONTAINING THE COMPUTED DISPLAY QUANTITIES

B) COMMUNICATION CELLS -

MCAFIN SET TO FIXED POINT ONE, B35 IF ABORT PHASE IS FINISHED, OTHERWISE, ZERO.

MCESAB FIRST INTEGRAL SECOND AFTER BURNOUT OF RETROROCKETS IN FIXED POINT B35, SET TO ZERO WHEN EITHER AN ORBIT OR REENTRY INTEGRATION TABLE IS NOT AVAILABLE.

TYPE 2

A) TABLES -

TCRKOH THE ABORT REENTRY HEIGHT TABLE

TCRKRT THE REENTRY VECTOR TABLE

L(OUTPUT) THE COMPLETE TWENTY-SIX LOCATION OUTPUT TABLE CONTAINING THE COMPUTED OUTPUT DISPLAY QUANTITIES

B) COMMUNICATION CELLS -

MCAFIN SET TO FIXED POINT ONE, B35, IF ABORT PHASE IS FINISHED, OTHERWISE, ZERO

MCESAB FIRST INTEGRAL SECOND AFTER BURNOUT OF RETROROCKETS IN FIXED POINT B35, SET TO ZERO WHEN EITHER AN ORBIT OR REENTRY INTEGRATION TABLE IS NOT AVAILABLE.

TYPE 3

A) TABLE -

L(OUTPUT) THE FIRST TWELVE CELLS OF THE TWENTY-SIX LOCATION OUTPUT TABLE ARE FILLED WITH THE COMPUTED PRESENT VECTORS DISPLAY QUANTITIES

B) COMMUNICATION CELL —

MCAFIN SET TO FIXED POINT ONE, B35, IF ABORT PHASE IS FINISHED, OTHERWISE, ZERO.

TYPE 4

A) TABLE —

L(OUTPUT) A SEVEN-LOCATION OUTPUT TABLE CONTAINING THE POSITION AND VELOCITY VECTORS ASSOCIATED WITH THE TIME OF RETROFIRE

B) COMMUNICATION CELL —

MCNRV4 SET TO ZERO IF THE POSITION AND VELOCITY VECTORS ASSOCIATED WITH THE

TIME OF RETROFIRE ARE AVAILABLE IN THE OUTPUT TABLE, OTHERWISE, SET TO FIXED POINT ONE, B35

METHOD

AFTER SAVING INDEX REGISTERS AND INITIALIZING CHABRT SAVES THE GIVEN TIME OF RETROFIRE (EITHER THE TIME COMPUTED IN THE HOLD PHASE OR THE TIME RECEIVED FROM TELEMETRY) FOR LATER COMPUTATIONS OF THE POSITION AND VELOCITY VECTORS ASSOCIATED WITH THIS TIME. IT THEN DETERMINES THE TYPE OF ENTRY BEING MADE ND PROCESSES THE DATA ACCORDINGLY.

TYPE 1 ENTRY -

THE TYPE 1 ENTRY IS THE FIRST ENTRY MADE TO CHABRT AND IS USED TO GENERATE THE INITIAL OUTPUT DISPLAY QUANTITIES.

BEFORE ATTEMPTING TO GENERATE THE ORBIT VECTOR TABLE USING THE INPUT AVERAGE VECTORS AS THE ANCHOR VECTORS, CHABRT SETS AN INDICATOR NON-ZERO TO SHOW THAT THE ABOVE TABLE HAS NOT BEEN GENERATED YET.

THE SUBROUTINE, CTSFT3, IS ENTERED TO COMPUTE THE TIME STEP TO BE USED BY INTEGRATION IN SETTING UP THE TABLE. ON AN ERROR RETURN, ENTRY TO THE INTEGRATOR IS BYPASSED AND THE INDICATOR REMAINS NON-ZERO.

ON A NORMAL RETURN FROM THE SUBROUTINE, THE INTEGRATOR, CORKNI, IS ENTERED TO SET UP THE ORBIT VECTOR TABLE. IF IT IS SUCCESSFUL, CHABRT RESETS THE INDICATOR TO ZERO SO THAT THE NEW TABLE MAY BE USED FOR COMPUTATIONS. OTHERWISE, THE INDICATOR REMAINS NON-ZERO SO THAT THE TABLE WILL NOT BE USED.

IF THE GIVEN TIME OF RETROFIRE IS ZERO, AS IN AN OVERSPEED CASE, CONTROL IS GIVEN TO C). IF THIS TIME IS

NOT ZERO, AND THE ORBIT VECTOR TABLE IS AVAILABLE, THE SUBROUTINE C6PTLI IS ENTERED TO INTERPOLATE FOR THE POSITION AND VELOCITY VECTORS AT THE BEGINNING OF RETROFIRE. IF EITHER THE TABLE IS NOT AVAILABLE OR C6PTLI FAILS TO PRODUCE THE REQUIRED VECTORS, CHABRT USES THE SUBROUTINE CABDAT TO INTEGRATE THE INPUT AVERAGE VECTORS TO THE TIME OF RETROFIRE USING A MINIMUM LIFT CONFIGUATION AND CAPSULE ORBIT WEIGHT. IF CABDAT ALSO FAILS, CONTROL IS PASSED TO C).

OTHERWISE, THE VECTORS AT THE BEGINNING OF RETROFIRE ARE PROCESSED BY THE SUBROUTINE C6RETC TO REFLECT SPACECRAFT CONDITIONS AT THE END OF RETROFIRE IN A NORMAL SITUATION. AN ERROR RETURN FROM THIS SUBROUTINE CAUSES A TRANSFER OF CONTROL TO C).

A) THE NEXT INTEGRAL SECOND AFTER BURNOUT IS COMPUTED AND PLACED IN CELL MCESAB FOR USE BY THE ACQUISITION DATA GENERATION PROGRAM. THE INTEGRATOR THEN USES THE POSITION AND VELOCITY VECOTRS AT THE END OF RETORFIRE AS THE ANCHOR VECTORS IN GENERATING THE ABORT REENTRY VECTOR AND HEIGHT TABLES. THE LIFT CONFIGURATION APPLIED IS THAT SPECIFIED BY THE GIVEN SETTING OF THE LIFT SWITCH. IF THE INTEGRATOR IS UNSUCCESSFUL IN OBTAINING THE DESIRED TABLES, CONTROL IS PASSED TO C).

B) IF THE TABLES ARE SUCCESSFULLY GENERATED, CHABRT SETS A SECOND INDICATOR ZERO TO SHOW THIS AND COMPUTES THE FOLLOWING QUANTITIES FOR OUTPUT -

1) THE TIME OF ARRIVAL AT THE HEIGHT DEFINING THE BEGINNING OF THE ATMOSPHERE.
2) THE RELATIVE FLIGHT PATH ANGLE AND THE MAGNITUDE OF THE RELATIVE VELOCITY VECTOR FROM THE VECTORS ASSOCIATED WITH THE TIME OF ARRIVAL AT 350,000 FEET ABOVE THE EARTH.
3) THE TIMES OF ARRIVAL AT THE HEIGHTS DEFINING THE BEGINNING AND THE END OF THE BLACKOUT PERIOD.
4) THE TIME OF ARRIVAL AT THE NOMINAL ABORT IMPACT HEIGHT AND, AFTER ADDING A PREDETERMINED DELAY FACTOR, THE TIME OF LANDING.
5) THE LATITUDE AND THE LONGITUDE OF IMPACT FROM THE VECTORS ASSOCIATED WITH THE TIME OF ARRIVAL AT THE NOMINAL ABORT IMPACT HEIGHT AND THE RECOVERY AREA ASSOCIATED WITH THE IMPACT POINT.
6) THE LONGITUDE AND LATITUDE OF IMPACT USING BOTH THE MINIMUM AND THE MAXIMUM LIFT CONFIGURATIONS.

IF AN ERROR RETURN IS MADE FROM ANY OF THE ABOVE COMPUTATIONS, THE QUANTITIES IN ERROR ARE SET TO ZERO EXCEPT LONGITUDE OF IMPACT (SET OFF THE PLOTBOARD SCALE) AND THE RECOVERY AREA (SET TO A FICTITIOUS AREA, H). CHABRT THEN CONTINUES AS DESCRIBED UNDER A TYEP 3 ENTRY.

C) THE SECOND INDICATOR IS SET NON-ZERO TO SHOW THAT THE ABORT REENTRY VECTOR AND HEIGHT TABLES HAVE NOT BEEN GENERATED. ALL OF THE QUANTITIES COMPUTED FROM THESE TABLES ARE THEN SET TO ZERO EXCEPT LONGITUDES OF IMPACT (SET OFF THE PLOT-

BOARD SCALE) AND THE RECOVERY AREA (SET TO A FICTITICUS AREA, H). THE PROGRAM THEN CONTINUES AS DESCRIBED UNDER A TYPE 3 ENTRY.

TYPE 2 ENTRY

THIS ENTRY IS MADE THIRTY SECONDS AFTER THE TELEMETRY INDICATED TIME OF FIRING OF THE FIRST RETROROCKET TO GENERATE NEW OUTPUT DISPLAYS BASED ON THE CURRENT RETROFIRE INFORMATION. EACH TIME THE TELEMETRY INDICATED NUMBER OF RETRORCCKETS FIRED OR THE TIME OF FIRING OF ANY RETROROCKET CHANGES, THIS ENTRY IS AGAIN MADE TO UPDATE THE DISPLAYS. HOWEVER, IF A MANUAL INSERTION MESSAGE CONTAINING RETROFIRE INFORMATION IS RECEIVED PRIOR TO ANY TELEMETRY INDICATIONS OF RETORFIRE, THIS TYPE OF ENTRY IS OMITTED.

IF AN ABORT ORBIT VECTOR TABLE HAS BEEN GENERATED USING THE INPUT AVERAGE VECTORS AS THE ANCHOR VECTORS, THE SUBROUTINE C6PTLI IS ENTERED TO INTERPOLATE FOR THE POSITION AND VELOCITY VECTORS AT THE GIVEN TIME OF RETROFIRE. IF THE SUBROUTINE IS SUCCESSFUL, CONTROL PASSES TO A). IF EITHER THE SUBROUTINE CANNOT SUCCESSFULLY INTERPOLATE FOR THE DESIRED VECTORS OR THE VECTOR TABLE IS NOT AVAILABLE, CABDAT IS ENTERED TO INTEGRATE THE INPUT AVERAGE VECTORS TO THE TIME OF RETROFIRE USING A MINIMUM LIFT CONFIGURATION AND THE CAPSULE ORBIT WEIGHT. IF CABDAT PRODUCES THE DESIRED VECTORS, CONTROL IS GIVEN TO A). IF CABDAT ALSO FAILS, CONTROL PASSES TO C) UNDER TYPE 1 ENTRY.

A) THE VECTORS AT THE BEGINNING OF RETROFIRE ARE THEN PROCESSED TO REFLECT SPACECRAFT CONDITIONS AT THE END OF RETROFIRE BY C6RETC. ENTRY TO THE SUBROUTINE C6RETC IS MADE VIA THE NON-NOMINAL

BRANCH WITH THE NON-NOMINAL RETROFIRE TABLE SET UP AS FOLLOWS -

1) THE FIRST FIVE CELLS CONTAIN THE GIVEN NUMBER OF RETROROCKETS FIRED AND THE TIMES OF FIRING OF RETROROCKETS NUMBERED 1, 3, 2 AND 4 IN THAT ORDER. IF ANY RETROROCKET HAS NOT FIRED, ITS TIME OF FIRING MUST BE SET TO ZERO.
2) THE LAST THREE CELLS CONTAIN THE ROLL, PITCH AND YAW ANGLES, RESPECTIVELY, OF THE SPACECRAFT AT THE BEGINNING OF RETROFIRE.

FAILURE TO COMPUTE THE DESIRED VECTORS BY C6RETC PRODUCES A TRANSFER TO C) UNDER TYPE 1 ENTRY. ON A NORMAL RETURN FROM C6RETC, CONTROL IS GIVEN TO A) UNDER TYPE 1 ENTRY.

TYPE 3 ENTRY

THIS ENTRY IS MADE EVERY COMPUTING CYCLE PRIOR TO THE RECEIPT OF A MANUALLY INSERTED RETROFIRE MESSAGE TO PRODUCE THE PRESENT VECTORS QUANTITIES EXCEPT WHEN A TYPE 1 OR A TYPE 2 ENTRY IS MADE.

THE POSITION AND VELOCITY VECTORS AT THE CURRENT TIME, FROM WHICH THE PRESENT VECTORS QUANTITIES ARE COMPUTED ARE OBTAINED AS FOLLOWS - THE SUBROUTINE C6PTLI INTERPOLATES FOR THE DESIRED VECTORS FROM EITHER THE ORBIT VECTOR TABLE, IF THE NUMBER OF RETROROCKETS FIRED IS ZERO, OR FROM THE REENTRY VECTOR TABLE, IF THE NUMBER OF RETROROCKETS FIRED IS NON-ZERO. IF THE ORBIT VECTOR TABLE IS TO BE USED AND HAS NOT BEEN GENERATED OR C6PTLI CANNOT SUCCESSFULLY INTERPOLATE FOR THE VECTORS, CHABRT ATTEMPTS TO GENERATE THE DESIRED VECTORS BY USE OF

CABDAT. IF CABDAT HAS AN ERROR RETURN OR IF THE REENTRY TABLE WAS IN USE, CHABRT SETS ALL OF THE PRESENT VECTORS QUANTITIES TO PREDETERMINED VALUES AND CONTINUES AS IN B).

A) USING THE CURRENT VECTORS OBTAINED, THE SUBROUTINE CPRVCM IS ENTERED TO COMPUTE THE PRESENT VECTORS QUANTITIES WHICH ARE THEN PLACED IN THE OUTPUT TABLE.

IF THE COMPUTED HEIGHT OF THE SPACECRAFT ABOVE AN OBLATE EARTH FALLS BELOW THE NOMINAL ABORT IMPACT HEIGHT, THE CELL MCAFIN IS SET TO ONE, INDICATING THAT THE ABORT PHASE IS FINISHED. THEREAFTER, ONLY TYPE 4 ENTRIES TO CHABRT SHOULD BE MADE. OTHERWISE, THE CELL REMAINS ZERO. CHABRT CONTINUES AS DESCRIBED IN B).

B) THE TIME OF RETROFIRE IS PLACED IN THE OUTPUT TABLE. IF THIS TIME IS ZERO, THE QUANTITIES COMPUTED FROM IT - THE ELAPSED TIME OF RETROFIRE AND THE TIME REMAINING UNTIL OR THE TIME SINCE RETROFIRE - ARE ALSO SET TO ZERO.

IF THE TIME OF RETROFIRE IS NOT ZERO, THE ELAPSED TIME OF RETROFIRE IS COMPUTED BY SUBTRACTING FROM THE GIVEN TIME OF RETROFIRE THE TIME OF LIFT-OFF AND IS PLACED IN THE OUTPUT TABLE.

THE FINAL QUANTITY - THE TIME REMAINING UNTIL OR THE TIME SINCE RETROFIRE - IS DEFINED TO BE THE DIFFERENCE BETWEEN THE GIVEN TIME OF RETROFIRE AND CURRENT TIME. THIS DIFFERENCE IS PLACED IN THE OUTPUT TABLE IF EITHER THE TELEMETRY INDICATES THAT AT LEAST ONE RETROROCKET

FIRED OR TELEMETRY SHOWS THAT NO RETROROCKETS FIRED, BUT THIS DIFFERENCE IS GREATER THAN ZERO. OTHERWISE, THIS QUANTITY IS SET TO ZERO.

CHABRT THEN RESTORES INDEX REGISTERS AND RETURNS CONTROL TO THE CALLING PROGRAM.

TYPE 4 ENTRY

THE TYPE 4 ENTRY IS MADE AFTER THE RECEIPT OF A MANUAL INSERTION MESSAGE CONTAINING RETROFIRE INFORMATION TO PRODUCE THE POSITION AND VELOCITY VECTORS ASSOCIATED WITH THE TIME OF RETROFIRE FOR SUBSEQUENT USE BY THE REENTRY PROCESSORS.

IF THE ORBIT VECTOR TABLE IS AVAILABLE, THE SUBROUTINE C6PTLI IS USED TO INTERPOLATE FOR THE DESIRED VECTORS. IF THE SUBROUTINE IS SUCCESSFUL IN OBTAINING THE VECTORS, CHABRT INDICATES THIS BY SETTING THE CELL MCNRV4 TO ZERO. RETURN IS THEN MADE TO THE CALLING PROGRAM.

IF EITHER THE VECTOR TABLE IS NOT AVAILABLE OR THE SUBROUTINE CANNOT INTERPOLATE FOR THE DESIRED VECTORS, CHABRT USES CABDAT TO INTEGRATE THE INPUT AVERAGE VECTORS TO THE TIME OF RETROFIRE USING A MINIMUM LIFT CONFIGURATION AND THE CAPSULE ORBIT WEIGHT. IF CABDAT PRODUCES THE VECTORS, MCNRV4 IS SET TO ZERO TO INDICATE THAT THE VECTORS ARE AVAILABLE AND RETURN IS MADE TO THE CONTROL PROGRAM.

IF CABDAT CANNOT OBTAIN THE VECTORS, MCNRV4 IS SET TO ONE TO SHOW THIS FACT BEFORE THE CALLING PROGRAM IS GIVEN CONTROL.

USAGE

A) CALLING SEQUENCE -

```
TSX     CHABRT,4
PZE     L(INPUT AVERAGE VECTORS),,L(OUTPUT)
PZE     L(TYPE ENTRY),,L(T/M INDICATED RETROFIRE
        INFORMATION)
PZE     L(CURRENT TIME),,L(GMTRC/F)
PZE     L(LIFT SWITCH SETTING)
RETURN
```

ON A TYPE 1 ENTRY, THE FIRST CELL IN THE T/M INDICATED RETROFIRE INFORMATION TABLE MUST BE ZERO INDICATING NO RETROROCKETS FIRED. ON A TYPE 2 ENTRY, THIS CELL MUST CONTAIN A NON-ZERO QUANTITY NOT TO EXCEED 4.

B) SORAGE REQUIRED - 442 LOCATIONS EXCLUDING STORAGE REQUIRED BY SUBROUTINES AND TABLES).

C) MISCELLANEOUS -

THE REENTRY TABLES ALLOW STOARGE FOR ONLY 60 MINUTES OF FORWARD INTEGRATION. THUS, IF ONLY ONE OR TWO RETROROCKETS FIRE, THE REENTRY HEIGHT CRITERIA WILL NOT BE REACHED. THEREFORE, NO REENTRY DISPLAYS WILL BE GENERATED.

IF PRESENT POSITION FALLS BELOW THE NOMINAL ABORT IMPACT HEIGHT CRITERION, THE CELL MCAFIN IS SET TO ONE INDICATING THAT ONLY TYPE 4 ENTRIES SHOULD BE MADE FROM HERE ON. THIS IS DONE BECAUSE 1) THE VECTORS PRODUCED BY TYPE 4 ENTRY WILL SHORTLY BE TURNED OVER TO THE REENTRY PHASE PROCESSORS AND 2) ANY OTHER TYPE OF ENTRY WILL PRODUCE NO SIGNIFICANT CHANGE IN DISPLAYS.

TABLE 66-1. OUTPUT TABLE FOR CHABRT
(SHEET 1 OF 3)

TYPE 1, 2, AND 3 ENTRIES

L(OUTPUT)		
	ϕ_{pp}	LATITUDE OF PRESENT POSITION IN FLOATING RADIANS
+1	λ_{pp}	LONGITUDE OF PRESENT POSITION IN FLOATING POINT RADIANS
+2	d	DOWNRANGE DISTANCE IN FLOATING POINT NAUTICAL MILES
+3	s	CROSSRANGE DISTANCE IN FLOATING POINT NAUTICAL MILES
+4	H	HEIGHT OF SPACECRAFT ABOVE A SPHERICAL EARTH IN FLOATING POINT NAUTICAL MILES
+5	H	HEIGHT OF SPACECRAFT ABOVE AN OBLATE EARTH IN FLOATING POINT NAUTICAL MILES
+6	γ	FLIGHT PATH ANGLE IN FLOATING POINT RADIANS
+7	i	INCLINATION ANGLE IN FLOATING POINT RADIANS
+8	v	VELOCITY MAGNITUDE IN FLOATING POINT FEET PER SECOND
+9	GMTRC/ GMTRF	GMT OF RETROFIRE EITHER COMPUTED OR FROM TELEMETRY IN FLOATING POINT SECONDS
+10	ESTRC/ ESTRF	ELAPSED TIME OF RETROFIRE IN FLOATING POINT SECONDS SINCE LIFT-OFF
+11	GTRS/ EGT SINCE RF	TIME REMAINING UNTIL RETROFIRE OR TIME SINCE RETROFIRE IN FLOATING POINT SECONDS

TABLE 66-1. OUTPUT TABLE FOR CHABRT
(SHEET 2 OF 3)

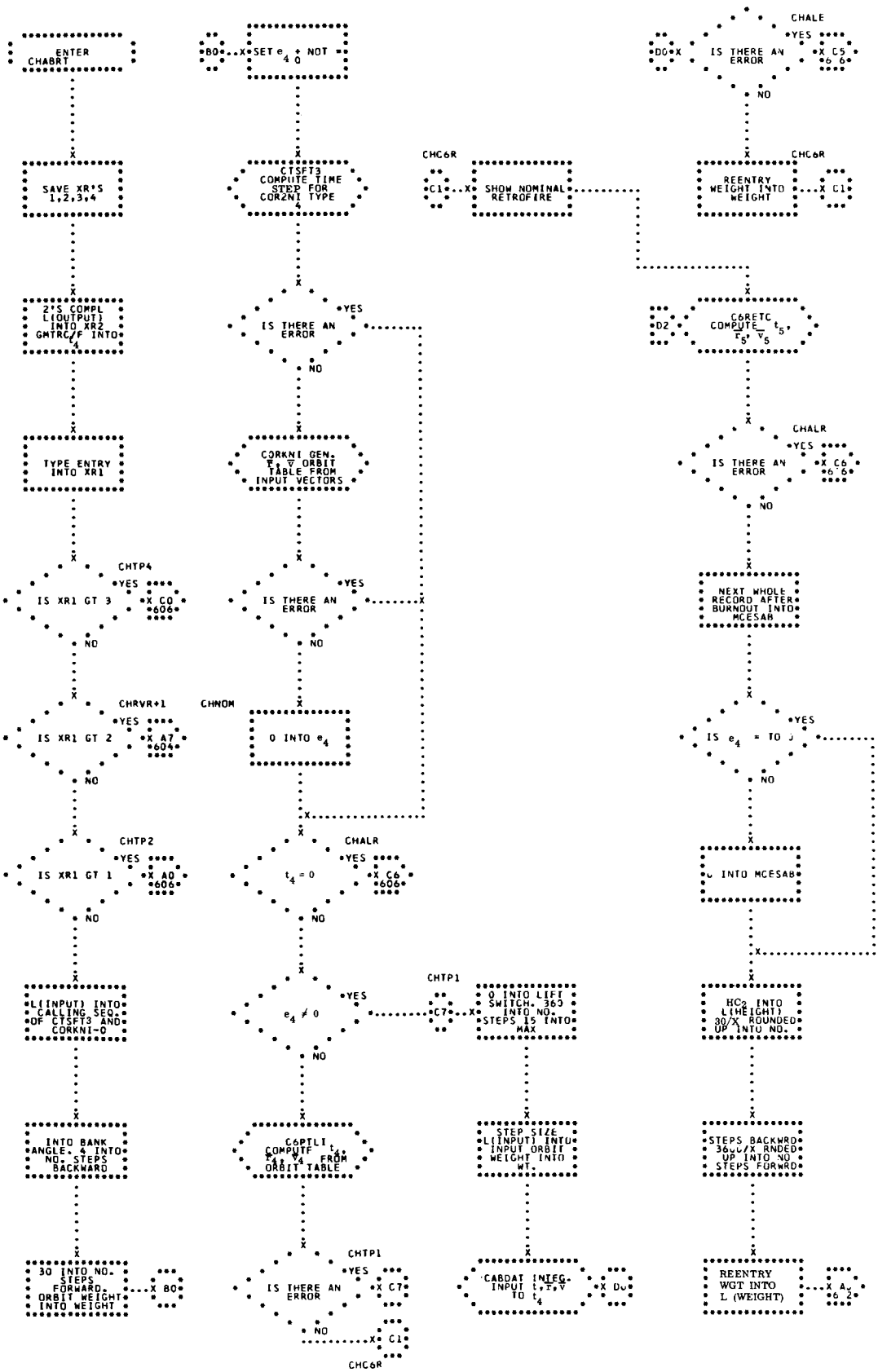
+12	GMT_{HC1}	GMT OF ARRIVAL AT HEIGHT DEFINING BEGINNING OF ATMOSPHERE IN FLOATING POINT SECONDS
+13	V_{350K}	RELATIVE VELOCITY MAGNITUDE AT 350 THOUSAND VEET IN FLOATING POINT FEET PER SECOND
+14	γ_{350K}	RELATIVE FLIGHT PATH ANGLE AT 350 THOUSAND FEET IN FLOATING POINT RADIANS
+15	GMT_{BO1}	GMT OF ARRIVAL AT HEIGHT AT BEGINNING OF BLACKOUT PERIOD IN FLOATING POINT SECONDS
+16	GMT_{BO2}	GMT OF ARRIVAL AT HEIGHT AT END OF BLACKOUT PERIOD IN FLOATING POINT SECONDS
+17	GMT_{HC2}	GMT OF ARRIVAL AT NOMINAL ABORT IMPACT HEIGHT IN FLOATING POINT SECONDS
+18	GMTLC	GMT OF LANDING IN FLOATING POINT SECONDS
+19	ϕ_{IP}	LATITUDE OF IMPACT IN FLOATING POINT RADIANS
+20	λ_{IP}	LONGITUDE OF IMPACT IN FLOATING POINT RADIANS
+21	ϕ_{ZL}	LATITUDE OF INPACT USING MINIMUM LIFT CONFIGURATION IN FLOATING POINT RADIANS
+22	λ_{ZL}	LONGITUDE OF IMPACT USING MINIMUM LIFT CONFIGURATION IN FLOATING POINT RADIANS

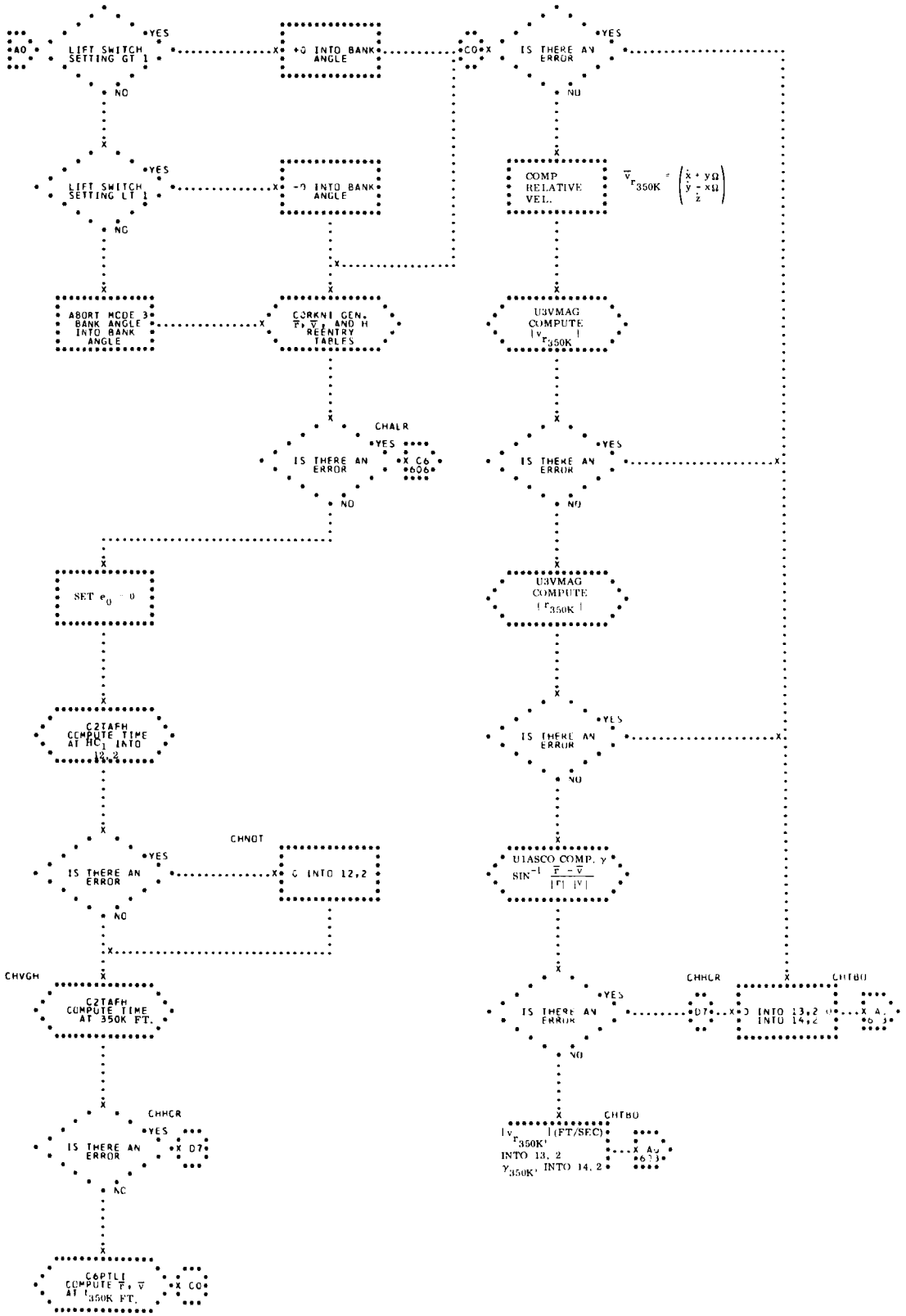
TABLE 66-1. OUTPUT TABLE FOR CHABRT
(SHEET 3 OF 3)

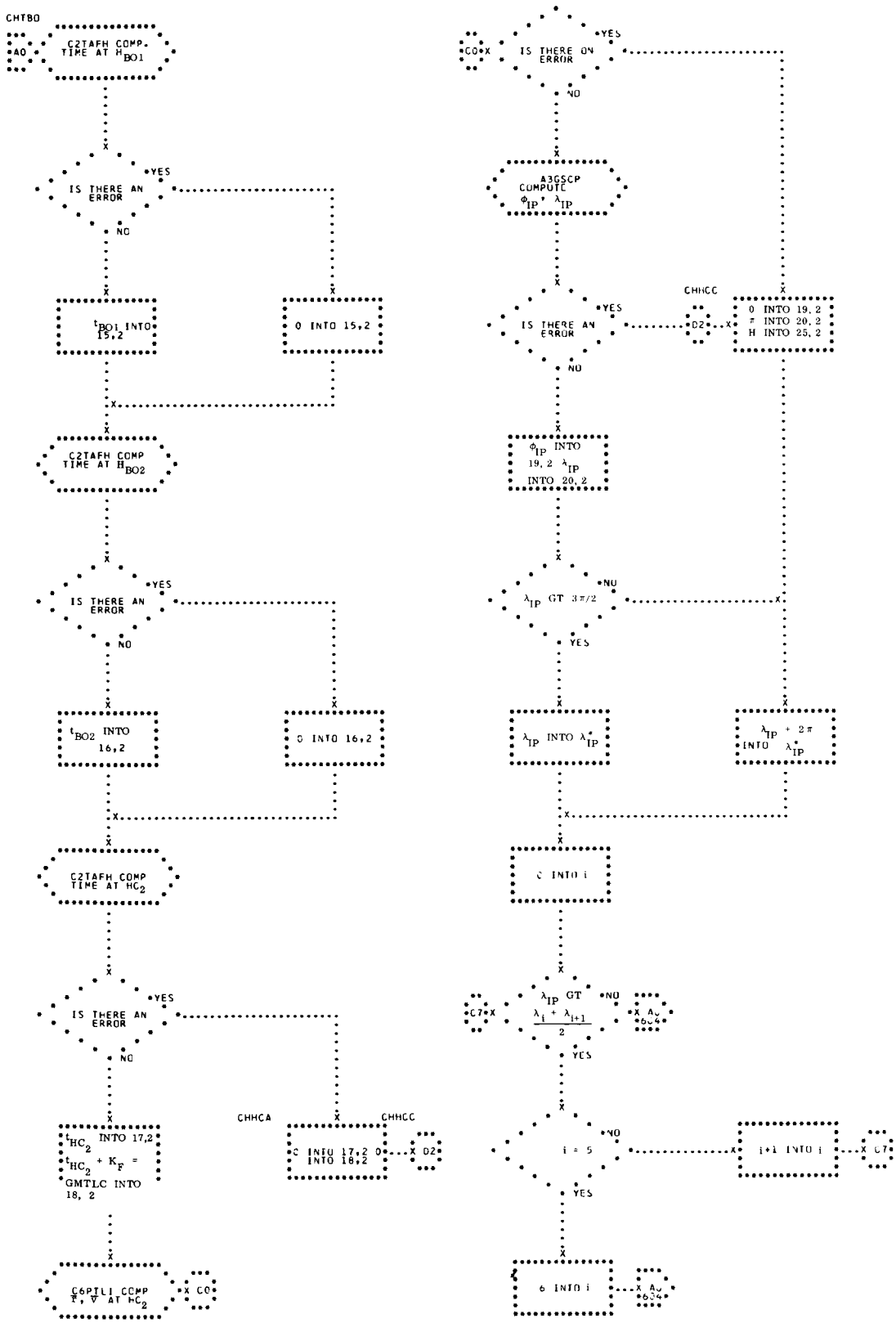
+23	ϕ_{ML}	LATITIUDE OF IMPACT USING MAXIMUM LIFT CONFIGURATION IN FLOATING POINT RADIANS
+24	λ_{ML}	LONGITUDE OF IMPACT USING MINIMUM LIFT CONFIGURATION IN FLOATING POINT RADIANS
+25	R.A.	RECOVERY AREA IN SPECIAL OCTAL CODE

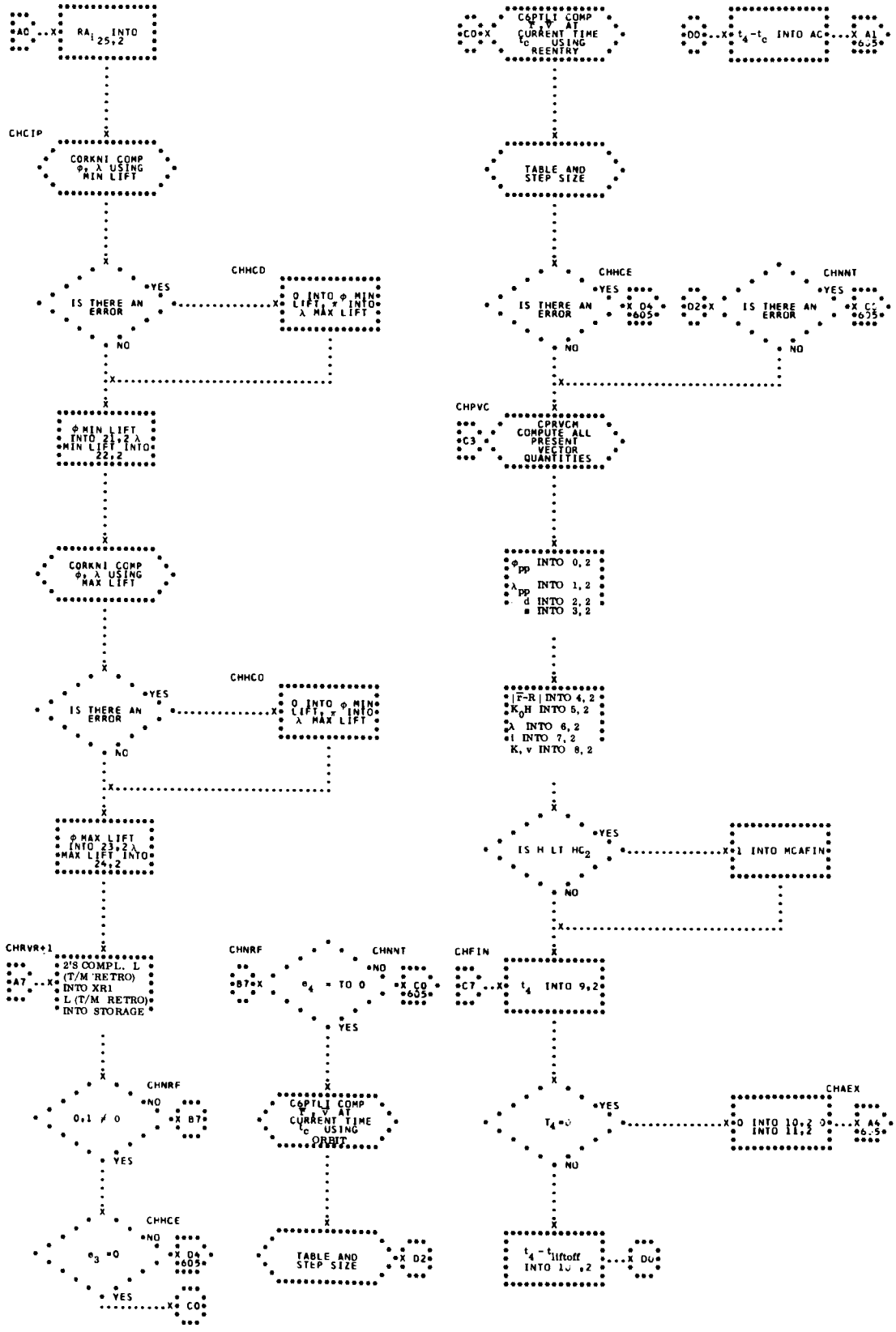
TYPE 4 ENTRY

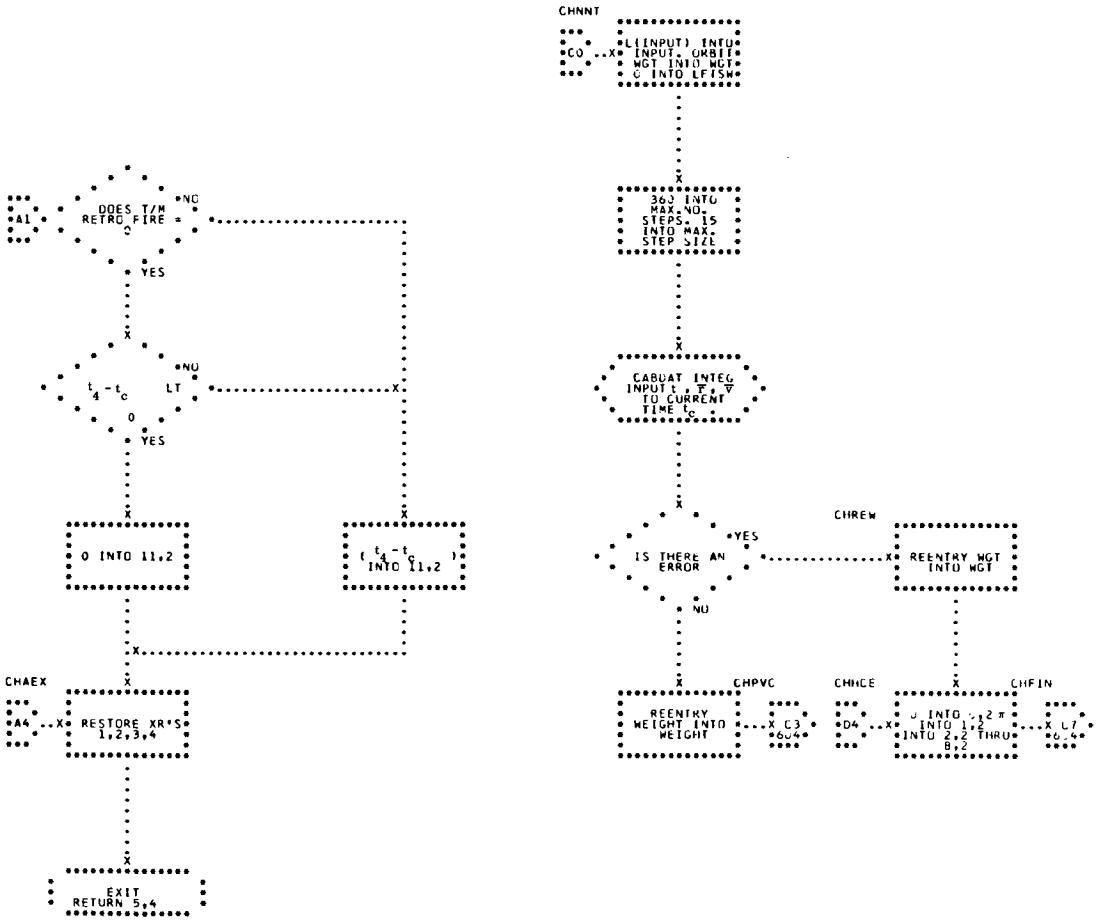
L(OUTPUT)	TIME OF RETROFIRE GMT FLOATING POINT SECONDS
+1	x-COMPONENT OF POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+2	y-COMPONENT OF POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+3	z-COMPONENT OF POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS
+4	x-COMPONENT OF VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+5	y-COMPONENT OF VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS
+6	z-COMPONENT OF VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS

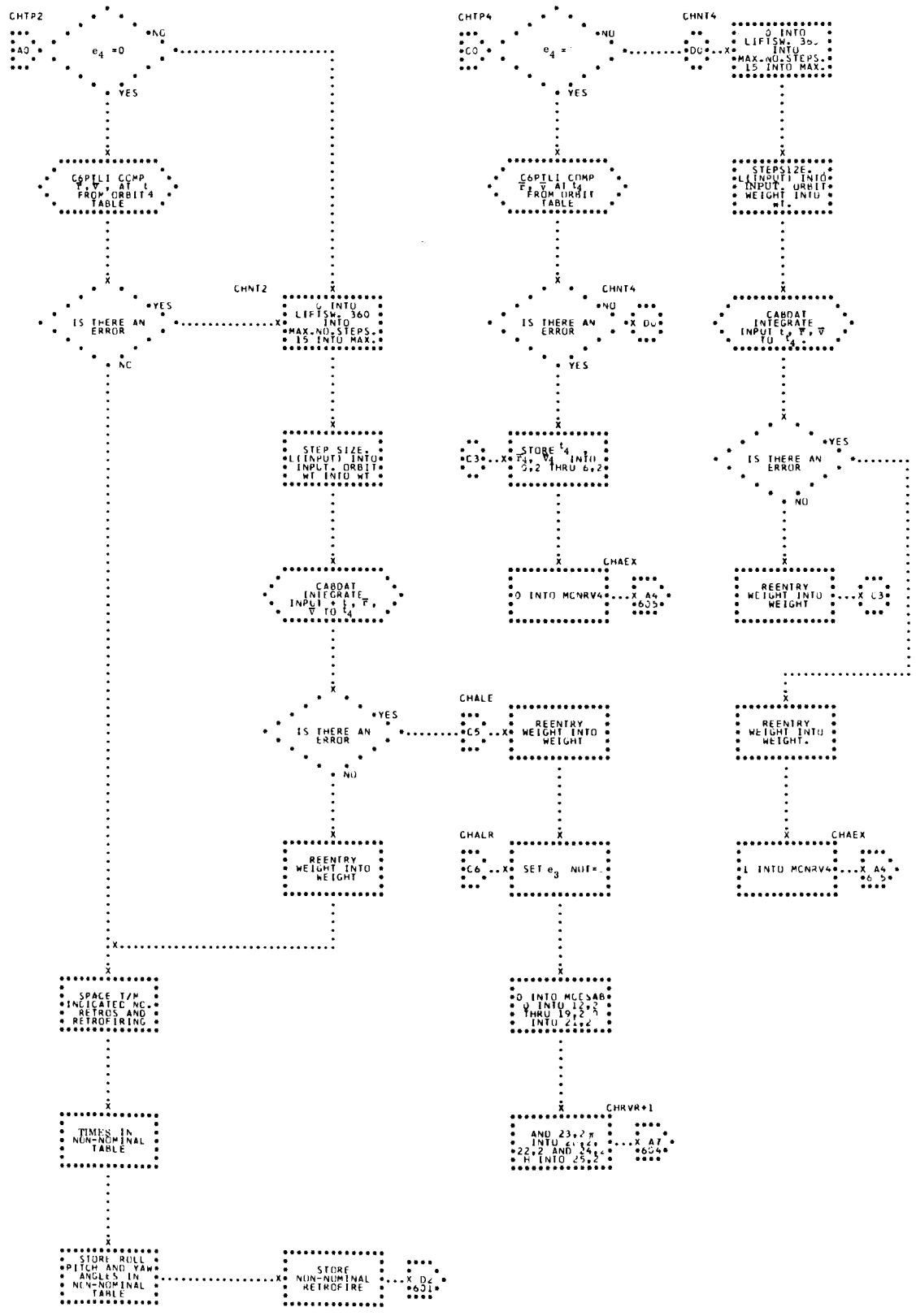

ENTER CHABRT
SAVE XR'S 1,2,3,4
2'S COMPL L(OUTPUT) INTO XR2 GMTRC/F INTO t_4
TYPE ENTRY INTO XR1
CHTP4
IS XR1 GT 3
YES
X C0 606
NO
CHRVR+1
IS XR1 GT 2
X A7 604
CHTP2
IS XR1 GT 1
X A0 606
L(INPUT) INTO CALLING SEQ. OF CTSFT3 AND CORKNI-0
INTO BANK ANGLE. 4 INTO NO. STEPS BACKWARD
30 INTO NO. STEPS FORWARD. ORBIT WEIGHT INTO WEIGHT
X 80
B0 X SET e_4 + NOT
CTSFT3 COMPUTE TIME STEP FOR CORZNI TYPE 4
IS THERE AN ERROR
CORKNI GEN. $\bar{r}_4$, $\bar{v}$ ORBIT TABLE FROM INPUT VECTORS
CHNOM
0 INTO e_4
CHALR
$t_4 = 0$
X C6 606
$e_4 \neq 0$
CHTP1
X C7
C6PTLI COMPUTE t_4, $\bar{r}_4$, $\bar{v}_4$ FROM ORBIT TABLE
IS THERE AN ERROR
X C7
X C1
CHC6R
CHC6R
C1 X SHOW NOMINAL RETROFIRE
0 INTO LIFT SWITCH. 360 INTO NO. STEPS IS INTO MAX
STEP SIZE L(INPUT) INTO INPUT ORBIT WEIGHT INTO WT
CABDAT INTEG. INPUT $t, \bar{r}, \bar{v}$ TO t_4
X D0
CHALE
D0 X IS THERE AN ERROR
YES X C5 606
NO
REENTRY WEIGHT INTO WEIGHT
CHC6R
X C1
D2 C6RETC COMPUTE t_5, $\bar{r}_5$, $\bar{v}_5$
CHALR
IS THERE AN ERROR
YES X C6 606
NEXT WHOLE RECORD AFTER BURNOUT INTO MCESAB
IS e_4 = TO 0
0 INTO MCESAB
HC_2 INTO L(HEIGHT) 30/X ROUNDED UP INTO NO.
STEPS BACKWRD 360/X RNDED UP INTO NO STEPS FORWRD
REENTRY WGT INTO L (WEIGHT)
X A2 602


A0
LIFT SWITCH SETTING GT 1
YES
NO
+Q INTO BANK ANGLE
LIFT SWITCH SETTING LT 1
YES
NO
-Q INTO BANK ANGLE
ABORT MODE 3 BANK ANGLE INTO BANK ANGLE
CORKNI GEN. F, V, AND H REENTRY TABLES
CHALR
IS THERE AN ERROR
YES
NO
X C6 606
SET e0 = 0
C2TAFH COMPUTE TIME AT HC1 INTO 12,2
IS THERE AN ERROR
YES
NO
CHNOT
C INTO 12,2
CHVGH
C2TAFH COMPUTE TIME AT 350K FT.
CHHCR
IS THERE AN ERROR
YES
NO
X D7
C6PTLI COMPUTE F, V AT h350K FT.
X C0
C0
IS THERE AN ERROR
YES
NO
COMP RELATIVE VEL.
U3VMAG COMPUTE |vr350K|
IS THERE AN ERROR
YES
NO
X
U3VMAG COMPUTE |Fr350K|
IS THERE AN ERROR
YES
NO
X
U1ASCO COMP. γ
IS THERE AN ERROR
YES
NO
CHHCR
X D7
CHTBO
D INTO 13,2
INTO 14,2
X A, 603
CHTBO
|vr350K| (FT/SEC) INTO 13, 2
γ350K INTO 14, 2
X Ay 603


CHTBO
AO
C2TAFH COMP. TIME AT H_{BO1}
IS THERE AN ERROR
YES
NO
t_{BO1} INTO 15,2
0 INTO 15,2
C2TAFH COMP TIME AT H_{BO2}
IS THERE AN ERROR
YES
NO
t_{BO2} INTO 16,2
0 INTO 16,2
C2TAFH COMP TIME AT HC_2
IS THERE AN ERROR
YES
NO
t_{HC_2} INTO 17,2
$t_{HC_2} + K_F$ = GMTLC INTO 18, 2
C6PILI COMP $\bar{r}, \bar{v}$ AT HC_2
CO
CHHCA
C INTO 17,2 0 INTO 18,2
CHHCC
D2
CO
IS THERE ON ERROR
YES
NO
A3GSCP COMPUTE Φ_{IP}, λ_{IP}
IS THERE AN ERROR
YES
CHHCC
D2
NO
0 INTO 19, 2
π INTO 20, 2
H INTO 25, 2
Φ_{IP} INTO 19, 2 λ_{IP} INTO 20, 2
λ_{IP} GT $3\pi/2$
NO
YES
λ_{IP} INTO λ^*_{IP}
$\lambda_{IP} + 2\pi$ INTO λ^*_{IP}
C INTO i
λ_{IP} GT $\frac{\lambda_i + \lambda_{i+1}}{2}$
NO
YES
C7
A4
i = 5
NO
YES
i+1 INTO i
C7
6 INTO i
A4


A0
RA$_i$ 25,2
CHCIP
CORKNI COMP φ, λ USING MIN LIFT
IS THERE AN ERROR
YES
NO
CHHCD
0 INTO φ MIN LIFT, λ INTO λ MAX LIFT
φ MIN LIFT INTO 21,2 λ MIN LIFT INTO 22,2
CORKNI COMP φ, λ USING MAX LIFT
IS THERE AN ERROR
YES
NO
CHHCO
0 INTO φ MIN LIFT, λ INTO λ MAX LIFT
φ MAX LIFT INTO 23,2 λ MAX LIFT INTO 24,2
CHRVR+1
A7
2'S COMPL. L (T/M RETRO) INTO XR1
L (T/M RETRO) INTO STORAGE
CHNRF
0,1 ≠ 0
NO
YES
B7
CHHCE
e_3 =0
NO
YES
D4
605
C0
CHNRF
e_4 = TO 0
NO
YES
CHNNT
C0
605
C&PTLI COMP r, v AT CURRENT TIME t_c USING ORBIT
TABLE AND STEP SIZE
D2
C&PTLI COMP r, v AT CURRENT TIME t_c USING REENTRY
TABLE AND STEP SIZE
CHHCE
IS THERE AN ERROR
YES
NO
D4
605
CHPVC
C3
CPRVCM COMPUTE ALL PRESENT VECTOR QUANTITIES
ϕ_{pp} INTO 0,2
λ_{pp} INTO 1,2
d INTO 2,2
■ INTO 3,2
|F-R| INTO 4,2
K_0H INTO 5,2
λ INTO 6,2
i INTO 7,2
K, v INTO 8,2
IS H LT HC_2
YES
NO
1 INTO MCAFIN
CHFIN
C7
t_4 INTO 9,2
T_4=0
YES
NO
CHAEX
0 INTO 10,2 0 INTO 11,2
A6
6.5
$t_4 - t_{liftoff}$ INTO 10,2
D0
D0
$t_4 - t_c$ INTO AC
A1
6.5
D2
CHNNT
IS THERE AN ERROR
YES
NO
C2
675


CHNNT
C0
L(INPUT) INTO INPUT, OKBIT WGT INTO WGT 0 INTO LFTSW
360 INTO MAX NO. STEPS, IS INTO MAX. STEP SIZE
CABOAT INTEG INPUT t, r, v TO CURRENT TIME t_c
IS THERE AN ERROR
YES
NO
CHREW
REENTRY WGT INTO WGT
A1
DOES T/M RETRO FIRE
NO
YES
t_4 - t_c LT 0
NO
YES
0 INTO 11,2
(t_4 - t_c) INTO 11,2
CHAEX
A4
RESTORE XR'S 1,2,3,4
EXIT RETURN 5,4
CHPVC
REENTRY WEIGHT INTO WEIGHT
C3
604
CHNCE
D4
0 INTO 1,2 π INTO 1,2 INTO 2,2 THRU 8,2
CHFIN
L7
604


CHTP2
AO
e_4 =0
NO
YES
C6PILI COMP $\bar{r}, \bar{v}$, AT t_4 FROM ORBIT TABLE
IS THERE AN ERROR
YES
NO
CHNT2
X
0 INTO LIFTSW, 360 INTO MAX.NO.STEPS. 15 INTO MAX.
STEP SIZE, L(INPUT) INTO INPUT. ORBIT WT INTO WT
CABDAT INTEGRATE INPUT $t, \bar{r}, \bar{v}$ TO t_4
IS THERE AN ERROR
YES
NO
CHALE
C5
REENTRY WEIGHT INTO WEIGHT
REENTRY WEIGHT INTO WEIGHT
CHALR
C6
SET e_3 NOT=
SPACE T/M INDICATED NO. RETROS AND RETROFIRING
TIMES IN NON-NOMINAL TABLE
STORE ROLL, PITCH AND YAW ANGLES IN NON-NOMINAL TABLE
STORE NON-NOMINAL RETROFIRE
D6
6031
CHTP4
CO
e_4 =
NO
YES
C6PILI COMP $\bar{r}, \bar{v}$ AT t_4 FROM ORBIT TABLE
CHNT4
IS THERE AN ERROR
YES
NO
CHAEX
C3
STORE t_4, $\bar{r}_4$, $\bar{v}_4$ INTO 0,2 THRU 6,2
CHAEX
0 INTO MCNRV4
A4
6035
0 INTO LIFTSW, 360 INTO MAX.NO.STEPS. 15 INTO MAX.
STEPSIZE, L(INPUT) INTO INPUT. ORBIT WEIGHT INTO WT.
CABDAT INTEGRATE INPUT $t, \bar{r}, \bar{v}$ TO t_4
IS THERE AN ERROR
YES
NO
REENTRY WEIGHT INTO WEIGHT
C3
REENTRY WEIGHT INTO WEIGHT.
CHAEX
1 INTO MCNRV4
A4
6 5
0 INTO MCESAB
0 INTO 12,2 THRU 19,2 INTO 21,2
CHRVR+1
AND 23,2 INTO 22,2 AND 24,2 H INTO 25,2
A7
6064

67. C2TAFH - COMPUTATION OF TIME AS A FUNCTION OF HEIGHT

THIS PROGRAM COMPUTES THE TIME ASSOCIATED WITH A GIVEN HEIGHT AS A FUNCTION OF THAT HEIGHT. (SEE FLOW-CHART).

INPUT

A) TABLES -

L(INPUT HEIGHT TABLE)	THE ADDRESS OF THE HIEGHT TABLE ARRANGED AS FOLLOWS - T_{start}, THE TIME ASSOCIATED WITH THE FIRST HEIGHT ENTRY IN THE TABLE. T_{stop}, THE TIME ASSOCIATED WITH THE LAST HEIGHT ENTRY IN THE TABLE, H_1, H_2, H_3, ETC., THE HEIGHT ENTRIES IN THE TABLE. ALL TIMES ARE IN GMT FLOATING POINT SECONDS. ALL HEIGHTS ARE IN FLAOTING POINT GEMINI LENGTH UNITS.
L(OUTPUT)	THE ADDRESS OF THE ONE CELL OUTPUT TABLE
L (H_d)	THE ADDRESS OF A ONE-CELL TABLE CONTAINING THE GIVEN HEIGHT FOR WHICH THE TIME IS DESIRED IN FLOATING POINT GEMINI LENGTH UNITS
L(h)	THE ADDRESS OF A ONE-LOCATION TABLE CONTAINING THE TIME INTER-VAL BETWEEN ENTRIES IN THE HEIGHT TABLE IN FLOATING POINT

SECONDS

B) CONSTANTS -

K00000	DEC 0	FIXED POINT ZERO
KCH233	OCT 233000000000	CHARACTERISTIC OF 233 OCTAL

OUTPUT

A) TABLE -

L(OUTPUT)	THE ADDRESS OF A ONE-LOCATION TABLE CONTAINING THE OUTPUT TIME ASSOCIATED WITH THE DESIRED HEIGHT IN FLOATING POINT GMT SECONDS

METHOD

AFTER INITIALIZING FOR THIS PASS, C2TAFH COMPUTES N, THE NUMBER OF ENTRIES IN THE HEIGHT TABLE, AS FOLLOWS -

$$N = ((t_{stop} - t_{start})/h) + 1$$

IF ONLY ONE HEIGHT APPEARS IN THE TABLE (I.E., $T_{start} = T_{stop}$), C2TAFH RETURNS TO THE CONTROL PROGRAM VIA THE ERROR EXIT.

THE HEIGHTS IN THE TABLE ARE EXPECTED TO BE IN DESCENDING ORDER, HOWEVER, C2TAFH SEARCHES THE TABLE TO DETERMINE WHETHER THIS CONDITION HAS BEEN SATISFIED. IF C2TAFH FINDS THAT THE HEIGHTS ARE NOT IN DESCENDING ORDER, CONTROL IS RETURNED TO THE MAIN PROGRAM VIA THE ERROR EXIT.

ONCE HEIGHTS SATISFYING THE ABOVE CRITERION ARE FOUND, C2TAFH COMPARES THE GIVEN HEIGHT TO SUCCEEDING

HEIGHTS IN THE TABLE IN AN EFFORT TO FIND TWO CONSECUTIVE HEIGHTS SUCH THAT

$$H_i \geq H_d \geq H_{i+1}$$

IF NO SUCH HEIGHTS ARE FOUND, C2TAFH RETURNS TO THE CALLING PROGRAM VIA THE ERROR EXIT. OTHERWISE, THE TIME ASSOCIATED WITH H_d, THE GIVEN HEIGHT, IS COMPUTED. IT MAY HAVE ONE OF THE FOLLOWING VALUES -

a) $t_d = t_{H_i}$, if $H_d = H_i$

b) $t_d = t_{H_{i+1}}$, if $H_d = H_{i+1}$

c) $t_d = t_{stop} + h((H_d - H_{i+1})/(H_{i+1} - H_i))$, if $H_i > H_d > H_{i+1}$

THE DESIRED TIME IS THEN STORED IN THE GIVEN OUTPUT TABLE AND C2TAFH GIVES CONTROL TO THE MAIN PROGRAM VIA THE NORMAL RETURN.

USAGE

A) CALLING SEQUENCE-

```
TSX     C2TAFH,4
PZE     L(INPUT HEIGHT TABLE),,L(OUTPUT)
PZE     L (Hd),,L(h)
ERROR RETURN
NORMAL RETURN
```

B) ERROR CONDITIONS - AN ERROR RETURN FROM C2TAFH IMPLIES ONE OF THE FOLLOWING-

1) ONLY ONE HEIGHT APPEARED IN THE HEIGHT TABLE
2) THE HEIGHTS IN THE GIVEN TABLE WERE NOT IN DESCENDING ORDER.
3) THE GIVEN HEIGHT DID NOT FALL BETWEEN TWO CONSECUTIVE DESCENDING HEIGHTS IN THE TABLE.

C) STORAGE REQUIRED -

79 LOCATIONS

D) TIMING (7094) -

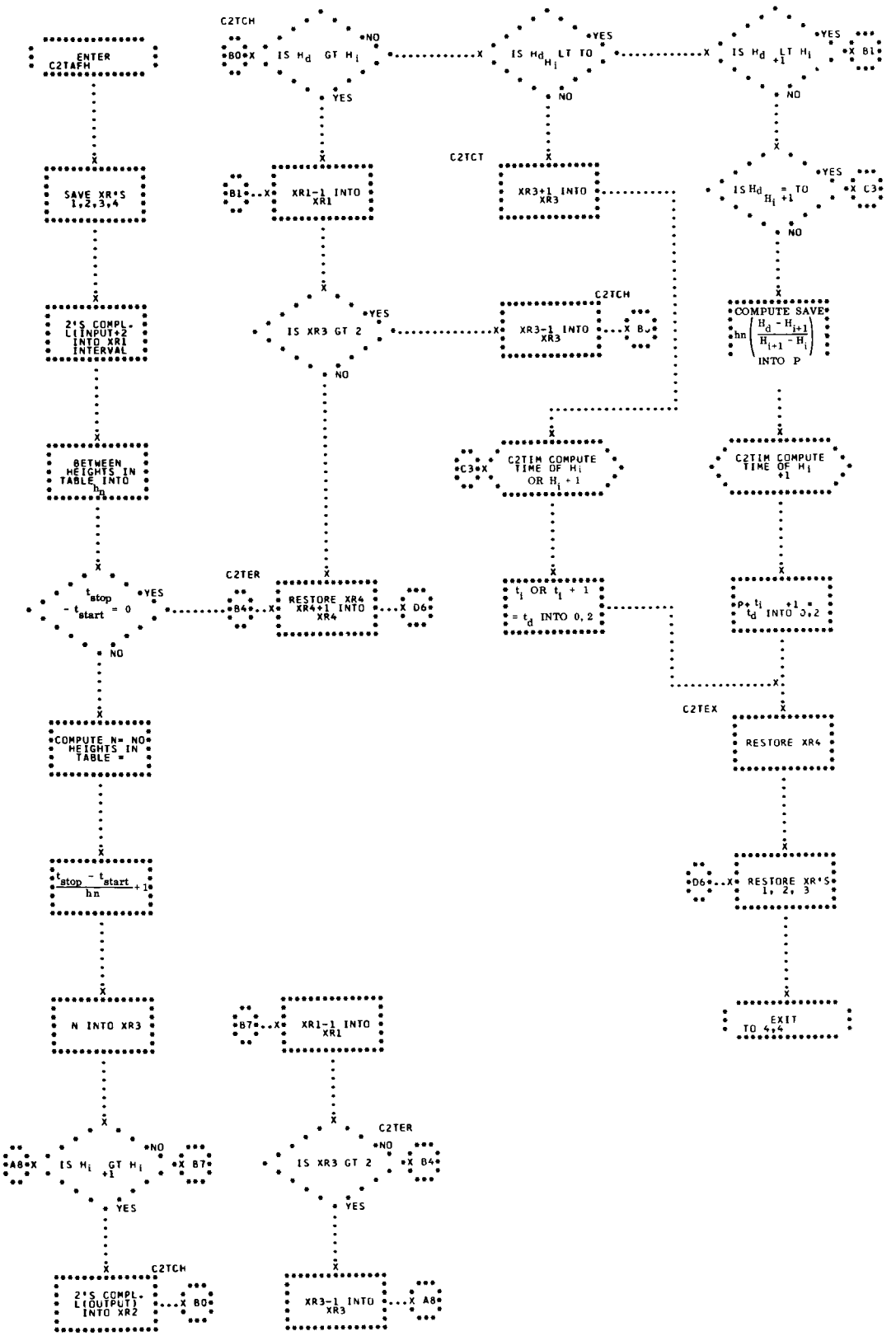
.112 + N(.012) MILLISECONDS, MINIMUM

.333 + (N-K)(.010) + J (.012) MILLISECONDS, MAXIMUM

WHERE N = NUMBER OF ENTRIES IN THE HEIGHT TABLE

K = NUMBER OF ENTRIES IN THE HEIGHT TABLE ALREADY EXAMINED

J = NUMBER OF ENTRIES IN THE HEIGHT TABLE WHICH DO NOT SATISFY $H_i \geq H_d \geq H_{i+1}$


ENTER C2TAFH
SAVE XR*S 1,2,3,4
2'S COMPL-L(INPUT+2 INTO XR1 INTERVAL
BETWEEN HEIGHTS IN TABLE INTO h_n
$t_{stop} - t_{start} = 0$
YES
NO
COMPUTE N= NO HEIGHTS IN TABLE =
$\frac{t_{stop} - t_{start}}{hn} + 1$
N INTO XR3
H_i GT H_{i+1}
YES
NO
2'S COMPL-L(OUTPUT) INTO XR2
C2TCH
B0
C2TER
B4
RESTORE XR4 XR4+1 INTO XR4
D6
B7
XR1-1 INTO XR1
IS XR3 GT 2
XR3-1 INTO XR3
C2TCT
XR3+1 INTO XR3
H_d GT H_i
H_d LT TO H_i
H_d LT H_{i+1}
H_d = TO H_{i+1}
B1
C3
B5
COMPUTE SAVE $hn\left(\frac{H_d - H_i}{H_{i+1} - H_i}\right)$ INTO P
C2TIM COMPUTE TIME OF H_i OR H_i + 1
C2TIM COMPUTE TIME OF H_i +1
t_i OR t_i + 1 = t_d INTO 0,2
P+t_i +1 = t_d INTO 0,2
C2TEX
RESTORE XR4
RESTORE XR*S 1, 2, 3
EXIT TO 4,4

68. CABDAT - LAUNCH ABORT PHASE PROGRAM FOR UPDATING A GIVEN

GIVEN A $t, \overline{r}, \overline{v}$ DEFINING THE ORBIT OF A SPACECRAFT AND THE DESIRED TIME FOR AN EXTRAPOLATED $t, \overline{r}, \overline{v}$ THIS PROGRAM COMPUTES THE $\overline{r}$ AND $\overline{v}$ AT THE DESIRED TIME BY USE OF NUMERICAL INTEGRATION. CABDAT IS USED BY THE SUFFIX MFCCMN OF THE MAIN LAUNCH CALCULATION PROGRAM, CCMAIN, BY THE MONITOR, CONTROL PROGRAM MABORT FOR THE LOW ABORT PROCESSOR CLABRT, AND BY THE HIGH ABORT PROCESSOR CHABRT. (SEE FLOWCHART).

INPUT

A) SUBROUTINE REQUIRED -
CORKNI - LAUNCH NUMERICAL INTEGRATION PROGRAM

B) PARAMETERS -
TCWF.R - SPACECRAFT REENTRY WEIGHT IN FLOATING POINT POUNDS
TCBAM2 - ABORT MODE 2 PROGRAMMED LIFT BANK ANGLE FLOATING POINT RADIANS

C) CONSTANTS -
KM0000 MINUS ZERO
K00000 ZERO
K00001 FIXED POINT ONE
K00002 FIXED POINT TWO
K000.5 FLOATING POINT ONE HALF
KCH233 MASK FOR FLOATING POINT CHARACTERISTIC 233C00000000

D) THE INPUT $t, \overline{r}, \overline{v}$ IS IN FLOATING POINT IN A SEVEN-LOCATION TABLE HAVING THE FORMAT -
L(INPUT VECTORS) t TIME IN GMT SECONDS

+1	x	$\bar{r}$	POSITION VECTOR IN GEMINI LENGTH UNITS
+2	y		
+3	z		
+4	$\dot{x}$	$\bar{v}$	VELOCITY VECTOR IN GEMINI VELOCITY UNITS
+5	$\dot{y}$		
+6	$\dot{z}$		

E) L(TIME OF DESIRED OUTPUT) - THE CONTENTS OF THIS LOCATION IS IN FLOATING POINT GMT SECONDS

F) L(LIFT SWITCH SETTING - LIFT SWITCH SETTING CONTAINS -

1) 0 FOR ZERO
2) 1 FOR PROGRAMMED LIFT
3) 2 FOR MAXIMUM LIFT

} FIXED POINT B35

G) L(MAX.NO.STEPS - THE CONTENTS OF THIS LOCATION IS IN FIXED POINT, B35

H) L(MAX.STEP SIZE) - THE CONTENTS OF THIS LOCATION IS IN FLOATING POINT SECONDS

OUTPUT

THE OUTPUT IS IN FLOATING POINT IN A SEVEN-LOCATION TABLE IN THE FOLLOWING FORMAT -

L(UPDATED VECTORS) t TIME IN GMT SECONDS

+1	x	$\bar{r}$ POSITION VECTOR IN GEMINI LENGTH UNITS
+2	y	
+3	z	
+4	$\dot{x}$	$\bar{v}$ VELOCITY VECTOR IN GEMINI VELOCITY UNITS
+5	$\dot{y}$	
+6	$\dot{z}$	

METHOD

CABDAT SUBTRACTS THE TIME OF THE GIVEN INPUT VECTORS FROM THE TIME AT WHICH THE UPDATED VECTORS ARE DESIRED, THEREBY CALCULATING THE TIME INCREMENT OVER WHICH NUMERICAL INTEGRATION IS TO BE DONE. IF THE TIME INCREMENT IS LESS THAN OR EQUAL TO THE MAXIMUM ALLOWABLE STEP SIZE, THE INTEGRATION IS ACCOMPLISHED IN A SINGLE STEP. IF THE TIME INTERVAL IS GREATER THAN THE MAXIMUM ALLOWABLE STEP SIZE, A SERIES OF ITERATIONS ARE PERFORMED, HALFING THE STEP SIZE AND DOUBLING THE NUMBER OF STEPS TO BE TAKEN UNTIL THE STEP SIZE IS LESS THAN OR EQUAL TO THE MAXIMUM ALLOWED AND WITH THE NUMBER OF STEPS BEING LESS THAN OR EQUAL TO THE MAXIMUM NUMBER OF STEPS SPECIFIED. IF IT IS NOT POSSIBLE TO USE A SUFFICIENTLY SMALL STEP SIZE, THE TIME INTERVAL FOR INTEGRATION IS DIVIDED BY THE MAXIMUM NUMBER OF STEPS TO BE TAKEN TO OBTAIN A STEP SIZE. NO THRUST IS APPLIED, AND ZERO LIFT, PROGRAMMED LIFT OR MAXIMUM LIFT IS USED ACCORDING TO THE LIFT SWITCH SETTING. IN THE EVENT OF AN ERROR RETURN FROM INTEGRATION, THE INPUT VECTORS ARE NOT UPDATED.

USAGE

A) CALLING SEQUENCE -

```
TSX     CABDAT,4
PZE     L(INPUT t, r̄, v̄),,L(OUTPUT t, r̄, v̄,)
PZE     L(TIME OF DESIRED OUTPUT),,
        L(LIFT SWITCH SETTING)
PZE     L(MAC.NO.STEPS),,L(MAX.STEP SIZE)
ERROR RETURN
NORMAL RETURN
```

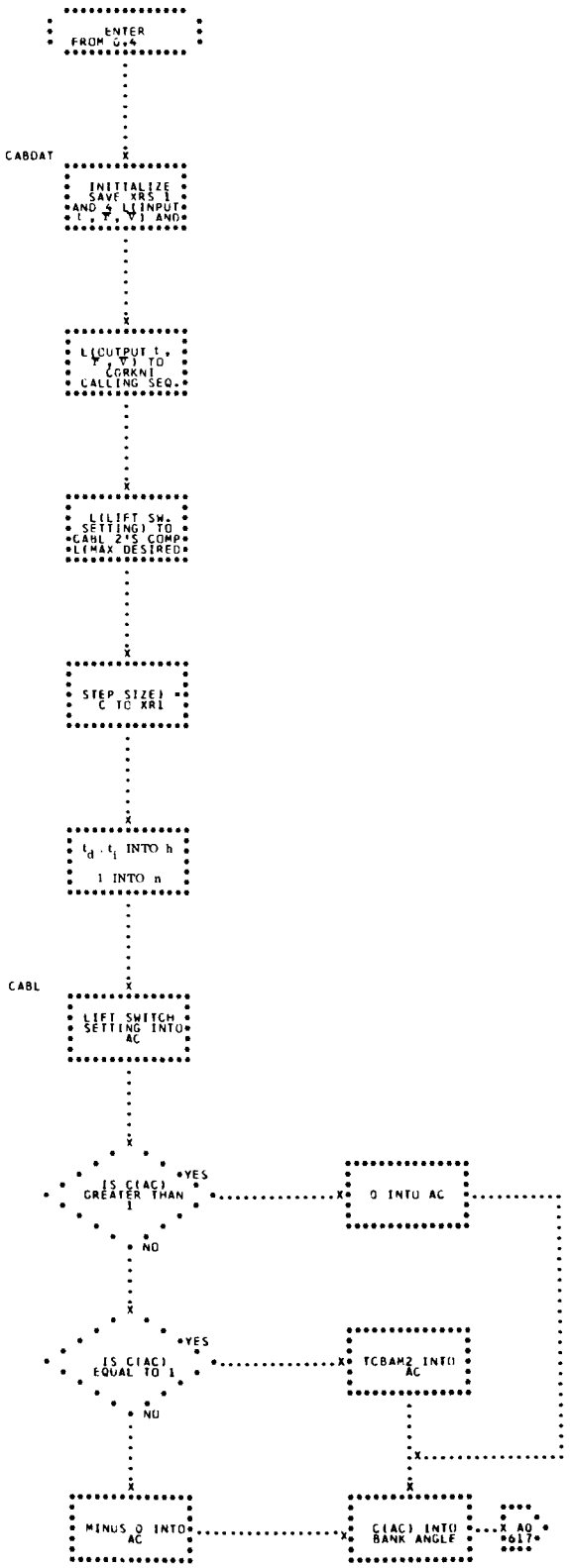
B) STORAGE REQUIRED -

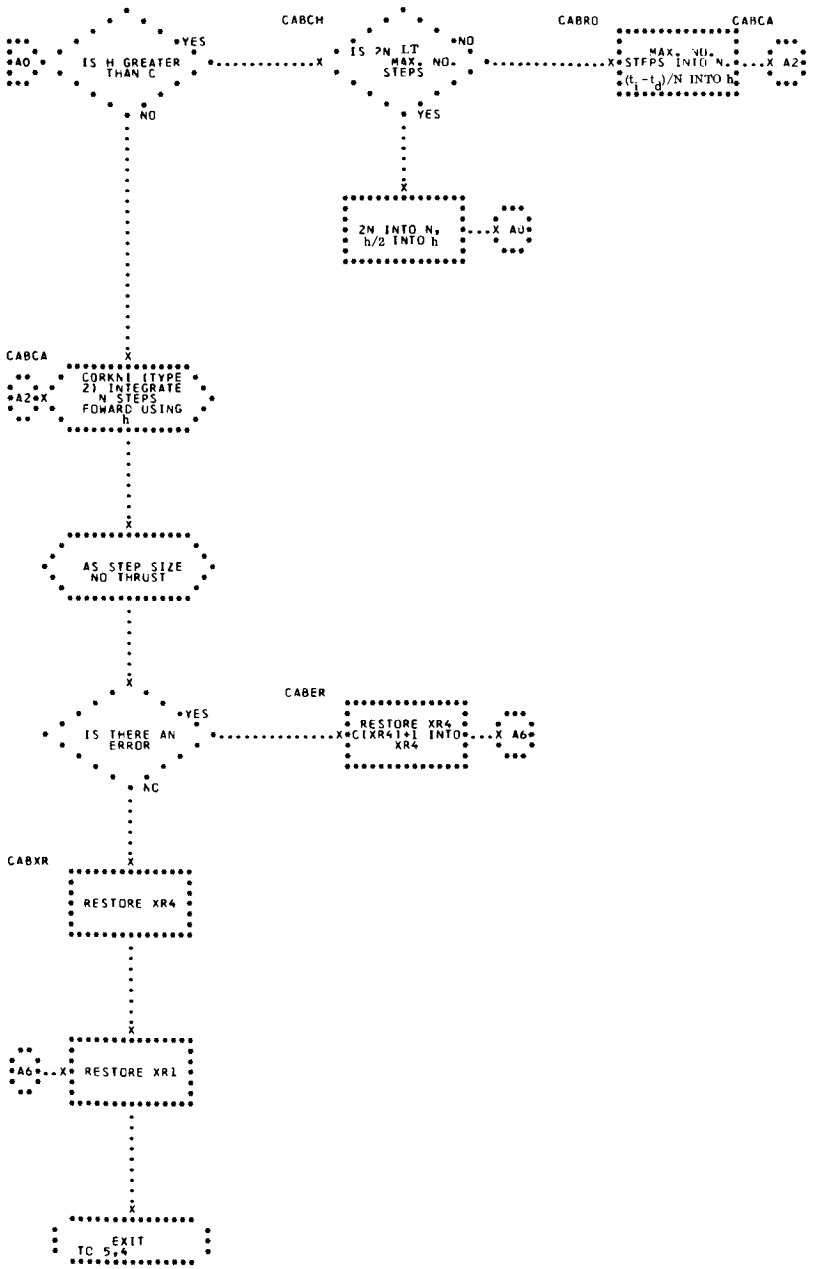
62 LOCATIONS EXLUDING CORKNI

C) TIMING-

1) .1044 MILLISECONDS EXCLUDING CORKNI (MINIMUM)

2) .1044 + .0512N MILLISECONDS EXCLUDING CORKNI
WHERE N IS THE NUMBER OF ITERATIONS PREFORMED
(AVERAGE)


ENTER FROM 0.4
CABDAT
INITIALIZE SAVE XRS 1 AND 4 L(INPUT t, T, V) AND
L(OUTPUT t, T, V) TO CGRKN1 CALLING SEQ.
L(LIFT SW. SETTING) TO CABL 2'S COMP L(MAX DESIRED
STEP SIZE) C TO XR1
t_d . t_i INTO h
1 INTO n
CABL
LIFT SWITCH SETTING INTO AC
IS C(AC) GREATER THAN 1
YES
NO
0 INTO AC
IS C(AC) EQUAL TO 1
YES
NO
TCBAM2 INTO AC
MINUS 0 INTO AC
C(AC) INTO BANK ANGLE
AO 617


A0
IS H GREATER THAN C
YES
NO
CABCH
IS 2N LT MAX. NO. STEPS
NO
YES
CABRO
MAX. NO. STEPS INTO N, (t_i - t_d)/N INTO h
CABCA
X A2
2N INTO N, h/2 INTO h
X A0
CABCA
A2
CORKN1 (TYPE 21) INTEGRATE N STEPS FOWARD USING h
AS STEP SIZE NO THRUST
CABER
IS THERE AN ERROR
YES
NO
RESTORE XR4 C(XR4)+1 INTO XR4
X A6
CABXR
RESTORE XR4
A6
RESTORE XR1
EXIT TO 5,4

PRECEDING PAGE BLANK NOT FILMED.

PART II

ORBIT/REENTRY

PRECEDING PAGE BLANK NOT FILMED.

1. BASIC COMPUTING CYCLE DURING ORBIT/REENTRY

ORBIT PHASE

DURING ORBIT AND REENTRY PHASES THE GODDARD REAL-TIME SYSTEM DRIVES DISPLAYS AT MISSION CONTROL CENTER, SENDS OUT DIGITAL COMMAND SYSTEM UPDATES, PROVIDES ACQUISITION DATA TO APPROPRIATE REMOTE SITE RADARS, AND BROADCASTS PCM SUMMARIES TO THE NETWORK.

IN ORDER TO PERFORM THESE FUNCTIONS (EXCEPT FOR PCM), THE ORBIT PHASE MUST HAVE A POSTIION AND VELOCITY VECTOR AT SOME TIME, PREFERABLY NEAR CURRENT GMT. BY USING THE COWEL OR RUNGE-KUTTA METHOD OF NUMERICALLY INTEGRATING THE THREE NEWTONIAN EUQATIONS OF MOTION, BOTH BACKWARD AND FORWARD FROM THIS INITIAL VECTOR, THE ABOVE REQUIREMENTS ARE MET. FOR PCM PERCENT FULL-SCALE VALUES OF ASTRONAUT AND CABIN CONDITIONS ARE INPUT INTO THE SYSTEM VIA TTY FROM UNIVAC 1218 COMPUTERS AT REMOTE SITES. THE GRTS CONVERTS THESE VALUES TO ENGINEERING QUANTITIES AND BROADCASTS THEM BACK TO THE WORLD.

DURING ORBIT PHASE THE POSITION AND VELOCITY VECTOR COMES FROM ONE OF TWELVE DIFFERENT SOURCES, EACH OF WHICH CAN TRIGGER THE BASIC COMPUTING CYCLE BY QUEUING MYOUR1 (THE MONITOR ORBIT UPDATE ROUTINE) WHICH INITIATES THE CYCLE. THESE SOURCES ARE LISTED BELOW ALONG WITH THE PROGRAM QUEUING MYOUR1 -

A) LAUNCH (MFCOMN) -

IF THE ABORT/ORBIT DECISION IS POSITIVE, THE LAUNCH PROGRAMS COMPUTE THE AVERAGE VECTOR OVER 10 SECONDS OF FREE FLIGHT AND QUEUE MYOUR1 WITH THIS VECTOR.

B) DIFFERENTIAL CORRECTION (MFDIFC) -

A COMPLETE RADAR MESSAGE WILL INITIATE A SERIES OF EVENTS WHICH CAN BE CONSIDERED THE HEART OF ORBIT COMPUTATIONS. THE LOW-SPEED INPUT PROCESSOR, IOTTIN, PLACES THE DATA IN AN ASSIGNED MESSAGE BLOCK AFTER CONVERTING THE INPUT DATA INTO THE FORMAT REQUIRED FOR PROCESSING. WHEN AN END-OF-TRANSMISSION IS DETECTED, AN INDICATION IS GIVEN TO THE MONITOR SUFFIX OF IOTTIN, MFTTIN. EACH RADAR AT A REPORTING STATION IS TREATED AS A SEPARATE SITE.
MFTTIN PASSES CONTROL TO THE EDIT PROGRAM, EOLED1, WHICH SUBJECTS THE DATA TO A TIME SEQUENCE CHECK AND CERTAIN SMOOTHNESS TESTS TO ELIMINATE, WHEN NECESSARY, OBSERVATIONS IN ERROR. THE REMAINING RADAR POINTS ARE THEN CORRECTED FOR REFRACTION AND LOCAL CONDITIONS. ALSO, THE TIME OF THE SPACECRAFTS' CLOSEST APPROACH TO THE RADAR STATION IS DETERMINED.
THE DIFFERENTIAL CORRECTION PROGRAM, DODIFC, USES THE NEW RADAR DATA TO CORRECT THE PREVIOUSLY DETERMINED ORBIT AND PRODUCES A CORRECTED RANGE AND VELOCITY VECTOR AT AN ASSOCIATED TIME - THESE ARE THE BASIC VALUES FOR THE NEW ORBIT PREDICTION.
DODIFC MAY, HOWEVER, REJECT THE CORRECTION BECAUSE THE DATA APPEARS ERRONEOUS OR BECAUSE THE CORRECTION WOULD BE NEGLIGIBLE. IF A CORRECTION IS ACCEPTABLE, DODIFC MAY REQUEST ANOTHER CORRECTION BE ATTEMPTED WITH THE SAME DATA TO FURTHER REFINE THE VECTORS. AFTER IT IS REFINED TO WITHIN SOME ACCEPTANCE/REJECTION CRITERION, THE BASIC CYCLE IS QUEUED WITH THE FINAL ORBIT.

C) RESTART (MFCCMN) -

EACH TIME NEW ORBIT PARAMETERS ARE COMPUTED, THE PARAMETERS, INCLUDING THE INTEGRATION PARAMETERS, ARE WRITTEN ON MAGNETIC TAPE TO MAINTAIN A HISTORY OF CRITICAL CHANGES SO AS TO PERMIT THE RESTORATION OF A PREVIOUS MISSION CONDITION AT ANY TIME. WITH THIS METHOD, THE COMPUTER IS CLEARED, THE GEMINI PROGRAMMING SYSTEM IS RELOADED, AND THE REQUESTED DATA IS READ FROM TAPE BY MYSRST. AFTER THE DATA HAS BEEN READ, MFCCMN IS ENTERED TO PREPARE THE GEMINI PROGRAMMING SYSTEM FOR ORBIT PROCESSING AND MYOUR1 IS THEN GIVEN CONTROL.

D) MANUAL INSERTION VECTOR (MFMN05) -

UNDER MANUAL CONTROL, THE POSITION AND VELOCITY VECTORS AND THEIR ASSOCIATED TIME ARE PUNCHED ON PAPER TAPE AND READ INTO CORE STORAGE OVER SUB-CHANNEL 30 OF THE DATA COMMUNICATION CHANNEL. EACH TIME A NEW ORBIT IS CALCULATED, ANCHOR POINT VECTORS ARE TO BE PRINTED ON-LINE. THEREFORE, THE GEMINI PROGRAMMING SYSTEM MAY REVERT TO A PREVIOUS ORBIT UNDER MANUAL INTERVENTION WITHOUT USING THE RESTART PROCEDURE.

E) NOMINAL ELEMENTS (MYNORB) -

A TABLE OF NOMINAL ORBITAL ELEMENTS AT INSERTION IS ORIGINED IN THE COMPUTER AND IF NO GOOD VECTOR IS AVAILABLE FROM THE LAUNCH PROGRAM, THESE ELEMENTS CAN BE CONVERTED TO AN INITIAL ORBIT DEFINING VECTOR, UNDER PCC CONTROL.

F) BACK-UP DIFFERENTIAL CORRECTION (MYKEYS) -

UNDER PCC CONTROL, ANY BLOCK OF RADAR DATA MAY BE BROUGHT INTO CORE AND CURVE-FITTET TO PRODUCE A POSITION AND VELOCITY VECTOR, WHICH MAY THEN BE QUEUED TO BEGIN THE BASIC CYCLE.

G) WEIGHT CHANGE (MFMN03) -

IF THE S/C ORBIT AND/OR REENTRY WEIGHT CHANGE DURING A MISSION, ALL ORBITAL COMPUTATIONS MUST BE RECALCULATED TO REFLECT THE NEW WEIGHT.

H) DENSITY FACTOR CHANGE (MFMN12) -

IF AN OFF-LINE PROGRAM SHOWS THAT OUR ATMOSPHERE MODEL FOR THE GIVEN S/C SURFACE AREA IS NOT ACCURATE FOR THE GIVEN DAY OF THE MISSION (IT CHANGES AS A FUNCTION OF SEASON, ETC.), THEN A NEW DENSITY MULTIPLYING FACTOR MAY BE MANUALLY INSERTED, AND ALL ORBITAL COMPUTATIONS THUS UPDATED.

I) AUTOMATIC CATCH-UP (MYMINS) -

IF THERE IS NO RADAR DATA TO UPDATE THE ORBIT (E.G., DURING THE NIGHTTIME IN THE U.S., THE S/C IS GENERALLY OFF THE RANGE OF THE MANNED FLIGHT NETWORK), THEN PERIODICALLY EVERY 30 MINUTES, THE VECTOR AT PRESENT TIME IS QUEUED TO MYOUR1. THIS INSURES SUFFICIENT ORBIT TABLE AHEAD OF CURRENT GMT FOR ACQUISITION AND OUTPUT CALCULATIONS.

J) MANEUVER MANUAL INSERTION (MYTABU) -

WHENEVER A PLANNED S/C MANEUVER IS ANTICIPATED, THE GMT OF THE BURN, DURATION, S/C PITCH AND YAW ATTITUDES MAY BE MANUALLY INSERTED INTO THE COMPUTER

AND SUBSEQUENT ORBIT CALCULATIONS MUST BE UPDATED TO TAKE THIS MANEUVER INTO ACCOUNT.

K) GOING THROUGH A MANEUVER (MYTABU) -
WHENEVER THE S/C COMPLETES A MANUAL MANEUVER (I.E., GMT $>$ GMTPC [GMT OF PROPULSION COMPLETION]), THE ORBIT IS UPDATED TO RELFECT THE NEW S/C ORBIT.

L) CONFIRMATION OF THRUST (MYCOTO) -
IF THE MANEUVER WAS NON-NOMINAL, THE ACTUAL S/C BURN PARAMETERS CAN BE INSERTED, AT WHICH TIME THE PRE-BURN ORBIT WILL BE TAKEN, ACTUAL MANEUVER CONDITIONS APPLIED, AND THE NEW ORBIT RESULTING WILL BE QUEUED TC MYOUR1.

NCW, ONCE MYOUR1 HAS A VECTOR, IT FIRST WILL TRANSFER TO ITS SUBROUTINE GVMNRF, WHICH WILL INTEGRATE AHEAD THRCUGH ALL THE PLANNED MANEUVERS FOR THE NEXT 22 ORBITS, STORING CNLY THCSE VECTORS AT THE FIRST EVEN MINUTE AFTER BURN-OUT CF EACH OF THE MANEUVERS. THESE MANEUVER-REFLECTING VECTORS CAN THEN BE USED FOR ALL COMPUTATIONS, THAT MUST TAKE INTC ACCOUNT PLANNED BURNS.

NEXT, MYOUR1, WILL GENERATE THE ORBIT TABLE TO BE USED FCR DRIVING PRESENT POSTION DISPLAYS AND FOR CALCULATING ACQUISITICN DATA. THIS TABLE, TNDDAC, IS REALLY A SET CF INTEGRATION TABLES COVERING THE PERIOD 5 MINUTES BACK AND 90 MINUTES AHEAD OF CURRENT GMT, EACH SUCCESSIVE CVERLAPPING TABLE CONTAINING EITHER FREE-FLIGHT CR BURN VECTORS AT ONE MINUTE OR 8 SECOND INTERVALS, RESPECTIVELY.

NEXT, IF CURRENT TIME IS WITHIN 50 MINUTES OF THE

RETRO-FIRE SETTING, THEN MYREN2, THE REENTRY PROCESSOR, IS QUEUED TO GENERATE GUIDANCE PARAMETERS, GMTRB (GMT TO REVERSE BANK ANGLE) AND BANK ANGLE, USING R5RARF AS A SUBROUTINE AND TO GENERATE A REENTRY PREDICTION TABLE BASED ON THESE PARAMETERS FOR ACQUISITION PURPOSES. THEN MYAPE3, THE APOGEE AND PERIGEE PROCESSOR, IS ENTERED TO CALCULATE THE POSITION AND HEIGHT OF APOGEE AND PERIGEE, AND ORBITAL ELEMENTS THERE FOR ON-LINE PRINT AND DISPLAYS. AFTER THIS, THE GMTRC (GMT OF RETRO-FIRE COMPUTED), GMTRB AND BANK ANGLE MUST BE CALCULATED IN ORDER FOR THE S/C TO LAND AT SOME PRE-SELECTED TARGET. THIS IS DONE BY MYTTF4, THE TIME-TO-FIRE PROCESSOR, USING R5RARF, REENTRY AND RETRO-FIRE PROCESSOR, AS A SUBROUTINE AND IS DISPLAYED FOR THE NEXT PRIMARY AND CONTINGENCY RECOVERY AREA.

FINALLY, IF THE S/C IS IN THE FIRST REVOLUTION, THE ORBIT CAPABILITY IS COMPUTED BY INTEGRATING AHEAD FOR 22 ORBITS OR UNTIL IMPACT. ALL OF THESE ORBIT COMPUTATIONS REQUIRE THE COWLE OR RUNGE-KUTTA INTEGRATOR PROCESSOR, NOCPNI.

OF ALL THE CALCULATIONS OF THE ORBIT PHASE THAT ARE COMPUTED AS A FUNCTION OF THE ORBIT, SOME ARE COMPUTED FROM THE CURRENT ORBIT-DEFINING POSITION AND VELOCITY VECTORS (1), SOME FROM THE ORBIT TABLE, TNDIAC, (2) (MINIMUM OF 60 MINUTES FORWARD AND REFLECTING MANEUVERS), AND SOME FROM THE TABLE OF MANEUVER-REFLECTING VECTORS, TMREFL, (3) (22 ORBITS AHEAD). THE FOLLOWING TABLE LISTS THEM IN ORDER -

(1) A.) DIFFERENTIAL CORRECTION
    B.) APOGEE AND PERIGEE COMPUTATIONS
    C.) ORBITAL CAPABILITY
(2) A.) EPHEMERIS RELATED PRESENT-POSITION DISPLAYS

B.) ACQUISITION DATA

(3) A.) TIME-TO-FIRE RETROS CALCULATIONS

B.) ORBIT NAVIGATION VECTORS FOR THE DCS

C.) GMT AS A FUNCTION OF LONGITUDE

D.) PROPULSION PANEL (MANEUVERS) COMPUTATIONS

THE LATTER THREE COMPUTATIONS ARE INITIATED BY MANUAL REQUEST, ONLY.

ORBIT-TO-REENTRY PHASE CHANGE

WHEN RETROFIRE OCCURS, THE MISSION CONTROL CENTER INFORMS GODDARD OF THE TIME-OF-FIRING OF EACH RETRO, THE TARGET COORDINATES, AND THE DESIRED GUIDANCE MODE. THIS TARGET COORDINATORS, AND THE DESIRED GUIDANCE MODE. THIS DATA IS PUNCHED ON PAPER TAPE AND READ INTO THE COMPUTER'S CORE STORAGE VIA SUBCHANNEL 30 OF THE DCC. THE PAPER TAPE PROCESSOR IOMANI, GIVES CONTROL TO MYREST, WHICH IS THE INTERPHASE PROCESSOR.

BEFORE RETROFIRE INFORMATION IS RECEIVED AT GODDARD, DATA FROM POST-RETROFIRE TRACKING MAY HAVE BEEN PROCESSED WITH THE SYSTEM IN THE ORBIT MODE. CONSEQUENTLY, THE INITIAL REENTRY TABLE COULD BE IN ERROR IF THE EXISTING ORBIT HAD BEEN USED TO DETERMINE THE RANGE AND VELOCITY VECTORS AT BURNOUT. TO PREVENT THIS, MYREST SEARCHES THE RESTART TAPE FOR THE LATEST POSITION AND VELOCITY VECTORS, BASED SOLELY ON RADAR DATA PRIOR TO RETROFIRE. MYREN2 TAKES THE ACTUAL TIME-OF-FIRING AND DERIVES THE BURNOUT RANGE, VELOCITY, AND TIME BY USING THE NUMBER OF ROCKETS FIRED, CALCULATES THE GUIDANCE PARAMETERS AND GENERATES THE FIRST ACTUAL REENTRY TABLE.

WHEN PHASE CHANGE TAKES PLACE, REENTRY OBSERVATIONS (THOSE TIME-TAGGED AFTER BURNOUT) MAY BE IN THE COMPUTER AND MAY BE INTERMINGLED WITH ORBITAL OBSERVATIONS.

MYPCIT DIRECTS CONTROL SO THAT ALL STORED RADAR OBSERVATIONS PLUS THOSE BEING RECEIVED WILL BE TIME-CHECKED, AND THOSE WITH PREBURNOUT TIME TAGS WILL BE DISCARDED. DATA WHICH MAY HAVE BEEN PREVIOUSLY CORRECTED IS AGAIN PRESENTED TC DODIFC - THIS TIME WITH ONLY REENTRY DATA AND WITH AN INDICATION OF REENTRY PHASE.

REENTRY PHASE

IN REENTRY, ONLY THE GUIDANCE PARAMETERS, GMTRB AND BANK ANGLE, AND THE REENTRY TABLE MUST BE CALCULATED. THUS, WHEN A COMPLETE RADAR MESSAGE IS AVAILABLE, IT IS EDITED (EOLED1) AND CORRECTED (DODIFC) TO PROVIDE A CORRECTED ANCHOR POINT FOR MYREN2. THE GENERATION OF A CORRECTED REENTRY TABLE TERMINATES THE CYCLE. AS IN ORBIT, DODIFC MAY REJECT EITHER THE CORRECTION OR THE REQUEST THAT A SUCCESSIVE CORRECTION BE ATTEMPTED TO DEVELOP GREATER PRECISION BEFORE PROCESSING NEW DATA.

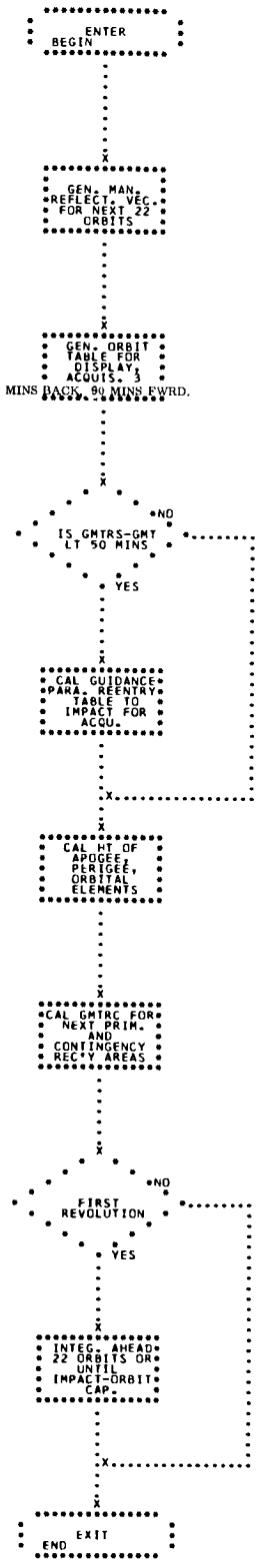
OUTPUT COMPUTATIONS

EVERY SIX SECONDS THROUGHOUT THE ORBIT PHASE THE OUTPUT CALCULATIONS PROGRAM O5ORMC IS ENTERED. THIS PROGRAM, USING THE LATEST ORBIT TABLE, DETERMINES CRITICAL ORBITAL ELEMENTS SUCH AS LATITUDE AND LONGITUDE OF PRESENT POSITION, INCLINATION ANGLE, HEIGHT, AND VELOCITY. THIS DATA IS GIVEN TO OOORRE WHICH CONVERTS, SCALES, AND PACKS THE DATA INTO THE PROPER FORMAT FOR TRANSMISSION OVER THE HIGH-SPEED LINES TO CAPE KENNEDY.

IN REENTRY THE PROCESS IS ESSENTIALLY THE SAME EXCEPT THE FREQUENCY IS EVERY THREE SECONDS AND ONLY THE REENTRY TABLE IS USED.

TC ASSIST THE RADAR SITES IN LOCATING THE SPACECRAFT, ACQUISITION DATA IS COMPUTED AND TRANSMITTED OVER

THE TELEYTPE NETWORK SO THAT EACH SITE RECEIVES TWO MESSAGES PER ORBIT, PROVIDING THE SPACECRAFT PASSES WITHIN RANGE, I.E., WITHIN -1 OF THE HORIZON. THE SITES RECEIVE THIS INFORMATION APPROXIMATELY 25 MINUTES AND FIVE MINUTES BEFORE THE SPACECRAFT APPEARS OVER THE HORI‡ON. ACQUISITION DATA CONSISTS OF RANGE, AZIMUTH, ELEVATION, AND TIME FOR ELEVATIONS OF 1 , 10 , AND 30 ABOVE THE HORIZON AS WELL AS FOR THE POINT OF CLOSEST APPROACH. THE ACQUISITION COMPUTATIONS PROGRAM, AQDATA, CHECKS THE PRESENT SPACECRAFT CLOCK SETTING. IF THE TIME OF THE ACQUISITION MESSAGE IS LATER THAN THE MOST RECENT TIME-TO-FIRE, OR TIME-OF-FIRING, THE REENTRY TABLE WILL BE USED TO GENERATE THE DATA, OTHERWISE, THE ORBIT TABLE WILL PROVIDE THE NECESSARY INPUT.


ENTER
BEGIN
GEN. MAN.
REFLECT. VEC.
FOR NEXT 22
ORBITS
GEN. ORBIT
TABLE FOR
DISPLAY,
ACQUIS. 3
MINS BACK, 90 MINS FWRD.
NO
IS GMTRS-GMT
LT 50 MINS
YES
CAL GUIDANCE
PARA. REENTRY
TABLE TO
IMPACT FOR
ACQU.
CAL HT OF
APOGEE,
PERIGEE,
ORBITAL
ELEMENTS
CAL GMTRC FOR
NEXT PRIM.
AND
CONTINGENCY
REC'Y AREAS
NO
FIRST
REVOLUTION
YES
INTEG. AHEAD
22 ORBITS OR
UNTIL
IMPACT-ORBIT
CAP.
EXIT
END

2. MSVTCA-GENERATION OF TMVTCA TABLE

MSVTCA GENERATES r AND v VERCTORS FOR USE BY DIFFERENTIAL CORRECTION (DC) DURING THE ORBIT PHASE. (SEE FLOWCHART.)

INPUT

A) MNLTMS — SYMBOL EQUATED WITH THE TOTAL POSSIBLE NUMBER OF ENTRIES CONTAINED IN TMSTMS.

B) TDCDEF — ADDRESS OF DEFINING VECTOR, STORED IN SEVEN CELLS -

TDCDEF	x	r
+1	y	
+2	z	
+3	$\dot{x}$	v
+4	$\dot{y}$	
+5	$\dot{z}$	
+6	t	

C) TMDATA — TABLE OF 3-WORD ENTRIES, EACH CONTAINING INFORMATION RELATING TO A PARTICULAR EDITED RADAR DATA BLOCK. THE ORDER OF 3-WORD ENTRIES CORRESPONDS TO THE NUMBER OF THEIR RESPECTIVE DATA BLOCKS LOCATED IN THE 32K BUFFERING MODULE.

D) TMSTMS — TABLE OF SINGLE-CELL WORDS, EACH ADDRESS CONTAINING THE ADDRESS OF THE STATION INFORMATION IN TMDATA. THE TABLE IS SEQUENCED ON THE BASIS OF THE TIME OF CLOSEST APPROACH (t_{ca})

INDICATED IN TMDATA, THE MOST RECENT ENTRIES APPEARING AT THE TOP OF THE TABLE.

OUTPUT

TMVTCA TABLE OF COMPUTED VECTORS, ONE OOR EACH ENTRY IN TMSTMS, AND STORED SUCCESSIVELY IN THE SAME ORDER AS TMSTMS.

METHOD

IF THE FIRST TMSTMS ENTRY IS NONZERO, THE PROGRAM EXAMINES EACH TMSTMS ENTRY, ONE AT A TIME, INTEGRATING AND STORING THE OUTPUT VECTOR BEFORE PROCEEDING TO THE NEXT TMSTMS WORD. RETURN TO MPDIFC IS ACCOMPLISHED AFTER PROCESSING THE LAST USABLE ENTRY (SIGNIFIED BY A MINUS SIGN BIT IN THE TMSTMS WORD) - THUS, WHEN A MINUS SIGN BIT IS ENCOUNTERED, THE PROGRAM IS SET FOR THE LAST PASS.

SUCCESSFUL COMPLETION OF ALL PASSES THROUGH TMSTMS IS INDICATED BY PLACING A POSITIVE ZERO IN THE AC BEFORE EXITING FROM THE PROGRAM. IN THE EVENT OF SOME ERROR CONDITION, A NONZERO IS PLACED IN THE AC BEFORE LEAVING THE PROGRAM.

THE VECTORS GENERATED ARE USED BY DC TO OBTAIN THE ORBITAL ARCS IMMEDIATELY AROUND EACH STATION SO AS TO PERFORM A DIFFERENTIAL CORRECTION. DIFFERENCES BETWEEN OBSERVED AND COMPUTED VALUES ARETHEN EXAMINED BY DC, AND THE DEFINING VECTOR (TDCDEF) IS CORRECTED, IF NECESSARY.

USAGE

A) CALLING SEQUENCE -

```
TSX     MSVTCA, 4
        NORMAL RETURN
```

B) STORAGE REQUIRED - 78 LOCATIONS.

C) CONSTANTS -

K00001 AND KMSTKA

D) MACROS -

QTSX

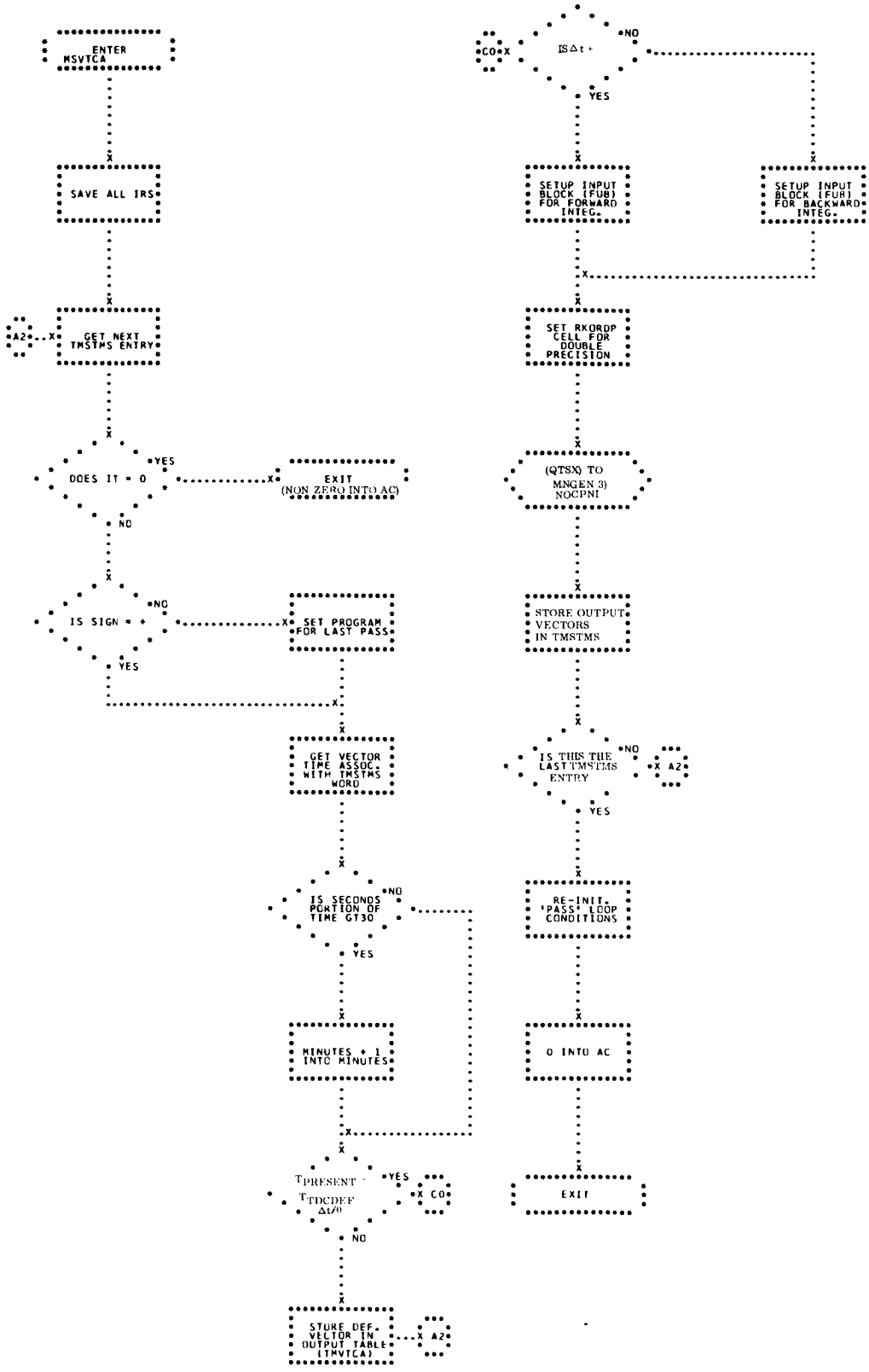
E) TABLES -

TMVTCA

F) TIME REQUIRED -

1) MAXIMUM (FOR FULL TMSTMS TABLE)-2023 CYCLES

2) MINUMUM (FOR ONE TMSTMS ENTRY)-101 CYCLES


ENTER MSVTCA
SAVE ALL IRS
A2
GET NEXT TMSTMS ENTRY
DOES IT = 0
YES
EXIT (NON ZERO INTO AC)
NO
IS SIGN = +
NO
SET PROGRAM FOR LAST PASS
YES
GET VECTOR TIME ASSOC. WITH TMSTMS WORD
IS SECONDS PORTION OF TIME GT30
NO
YES
MINUTES + 1 INTO MINUTES
TPRESENT - TTDCDEF Δt/0
YES
CO
NO
STORE DEF. VECTOR IN OUTPUT TABLE (TMVTCA)
A2
CO
IS Δt +
NO
YES
SETUP INPUT BLOCK (FUB) FOR FORWARD INTEG.
SETUP INPUT BLOCK (FUB) FOR BACKWARD INTEG.
SET RKORDP CELL FOR DOUBLE PRECISION
(QTSX) TO MNGEN 3) NOCPNI
STORE OUTPUT VECTORS IN TMSTMS
IS THIS THE LAST TMSTMS ENTRY
NO
A2
YES
RE-INIT. 'PASS' LOOP CONDITIONS
0 INTO AC
EXIT

3. MSLCUC - MONITOR SUBROUTINE TO MOVE A RADAR MESSAGE BLOCK TO THE 32K CORE BUFFER MODULE

MSLCUC RECEIVES CONTROL FROM MPDIFC AFTER A RADAR MESSAGE BLOCK FROM A REMOTE STATION HAS BEEN EDITED BY EOLED1 AND IS READY TO BE TRANSFERRED TO THE 32K BUFFER MODULE. (SEE FLOWCHART)

INPUT

A) AC — ADDRESS PORTION CONTAINS THE BASE ADDRESS OF THE EDITED DATA BLOCK TO BE TRANSFERRED FROM THE 32 K OPERATING MODULE TO THE 32K BUFFER MODULE.

B) MCBLKS — ADDRESS PORTION CONTAINS A NUMBER CORRESPONDING TO THE NUMBER OF UNUSED EDITED DATA BLOCKS AVAILABLE IN THE 32K BUFFER MODULE.

CT MCLCBM — ADDRESS PORTION CONTAINS A MONITOR NUMBER DEFINING THE TOTAL NUMBER OF EDITED DATA BLOCKS CONTAINED IN THE 32K OPERATING MODULE.

D) MCUCBL — ADDRESS PORTION CONTAINS A MONITOR NUMBER DEFINING THE LENGTH OF AN EDITED DATA BLOCK.

E) MCUCMB — ADDRESS PORTION CONTAINS A MONITOR NUMBER DEFINING THE TOTAL NUMBER OF BLOCKS RESERVED FOR EDITED DATA IN THE 32K BUFFER MODULE.

F) MNLCMB — MONITOR NUMBER EQUAL TO THE TOTAL NUMBER OF BLOCKS RESERVED FOR LOW-SPEED INPUT DATA IN THE 32K OPERATING MODULE.

G) MNUCRL MONITOR NUMBER EQUAL TO THE TOTAL NUMBER OF CELLS CONTAINED IN AN EDITED DATA BLOCK.

H) MNUCMB MONITOR NUMBER EQUAL TO THE TOTAL NUMBER OF BLOCKS RESERVED FOR EDITED DATA IN THE 32K BUFFER MODULE.

OUTPUT

A) MCBKNO CONTAINS IN THE ADDRESS FIELD A NUMBER CORRESPONDING TO ONE OF THE SEQUENTIALLY NUMBERED EDITED DATA BLOCKS IN THE 32K BUFFER MODULE INTO WHICH THE INCOMING EDITED DATA HAS BEEN TRANSFERRED.

B) MCINST CONTAINS IN THE ADDRESS PORTION THE ADDRESS OF THE 3-WORD STATION INFORMATION BLOCK IN TMDATA. THIS STATION INFORMATION CORRESPONDS TO THE EDITED DATA BLOCK MOST RECENTLY TRANSFERRED TO THE 32K BUFFER MODULE.

C) MCSTNO CONTAINS IN THE ADDRESS PORTION THE INTERNAL STATION NUMBER CORRESPONDING TO THE STATION DATA MOST RECENTLY TRANSFERRED TO THE 32K BUFFER MODULE.

D) TMDATA A TABLE, CONTAINING IN 3-WORD GROUPS, STATION INFORMATION DATA CORRESPONDING TO AN EDITED DATA BLOCK WHICH HAS BEEN TRANSFERRED TO THE 32K BUFFER MODULE. (FORMAT SHOWN IN TMDATA FORMAT TABLE.)

E) TMRADR CONTAINS AN ENTRY FOR EACH BLOCK IN THE 32K CORE BUFFER MODULE. EACH ENTRY INDICATES THE STATION NUMBER AND THE NUMBER OF RADAR OBSERVATIONS CONTAINED

IN THE DATA BLOCK FOR THAT STATION.

METHOD

ON ENTRY TO MSLCUC, THE ADDRESS OF THE ACCUMULATOR CONTAINS THE BASE ADDRESS OF THE EDITED DATA BLOCK TO BE TRANSFERRED TO THE 32K BUFFER MODULE. MCBLKS CONTAINS A NUMBER CORRESPONDING TO THE NUMBER OF UNUSED BLOCKS IN THE 32K BUFFER MODULE AND IF THE CONTENTS OF MCBLKS IS EQUAL TO ZERO, TESTS ARE PERFORMED TO DETERMINE WHICH BLOCK IN THE BUFFER AREA HAS THE OLDEST TIME OF MINIMUM RANGE. THE TIME OF MINIMUM RANGE OF THE INCOMING EDITED DATA BLOCK IS COMPARED WITH THE OLDEST TIME OF MINIMUM RANGE IN THE BUFFER AREA. IF THE TIME ASSOCIATED WITH THE INCOMING DATA BLOCK IS LESS THAN THE OLDEST TIME OF MINIMUM RANGE, AN ON-LINE MESSAGE IS QUEUED STATING THAT THE INCOMING DATA IS OLDER THAN THE OLDEST DATA BLOCK IN THE 32K BUFFER MODULE. MSLCUC THEN PLACES A ZERO IN THE ACCUMULATOR AND TRANSFERS CONTROL TO MPDIFC.

IF THE TIME OF MINIMUM RANGE ASSOCIATED WITH THE INCOMING DATA BLOCK IS GREATER THAN THE OLDEST TIME OF MINIMUM RANGE IN THE BUFFER AREA, THE INCOMING EDITED DATA BLOCK REPLACES THIS OLDEST BLOCK. MSLCUC THEN CONSTRUCTS THE 3-WORD STATION INFORMATION ENTRY AND INSERTS IT IN THE APPROPRIATE PLACE IN THE TMDATA TABLE. MSLCUC FURTHER SETS A MINUS SIGN IN THE TMFRBK ENTRY CORRESPONDING TO THE OPERATING ADDRESS OF THE EDITED DATA BLOCK. THE MINUS SIGN SIGNIFIES TO IOTTIN THAT THE DATA BLOCK CAN BE USED FOR NEW DATA. AN ON-LINE MESSAGE IS QUEUED STATING THAT THE EDITED DATA BLOCK HAS BEEN MOVED TO BLOCK NUMBER IN THE 32K BUFFER MODULE. MSLCUC THEN TRANSFERS CONTROL TO MPDIFC WITH THE CONTENTS OF THE ACCUMULATOR SET TO NONZERO.

USAGE

MSLCUC IS ENTERED FROM AND EXITS TO MPDIFC.

A) CALLING SEQUENCE -

```
TSX     MSLCUC,4
        RETURN
```

B) STORAGE REQUIRED - 105 LOCATIONS EXCLUSIVE OF SUBROUTINES.

C) MACROS -

QENBA
QENBZ
QMOVE
QUEUE

D) PARAMETERS -

MNLCMB
MNMESS
MNUCBL

E) SUBROUTINE -

MSWTMS

F) COMMUNICATION CELLS -

MCBKNO
MCBLKS
MCINST
MCLCMB
MCSTMO
MCUCBL
MCUCMB

G) CONSTANTS -

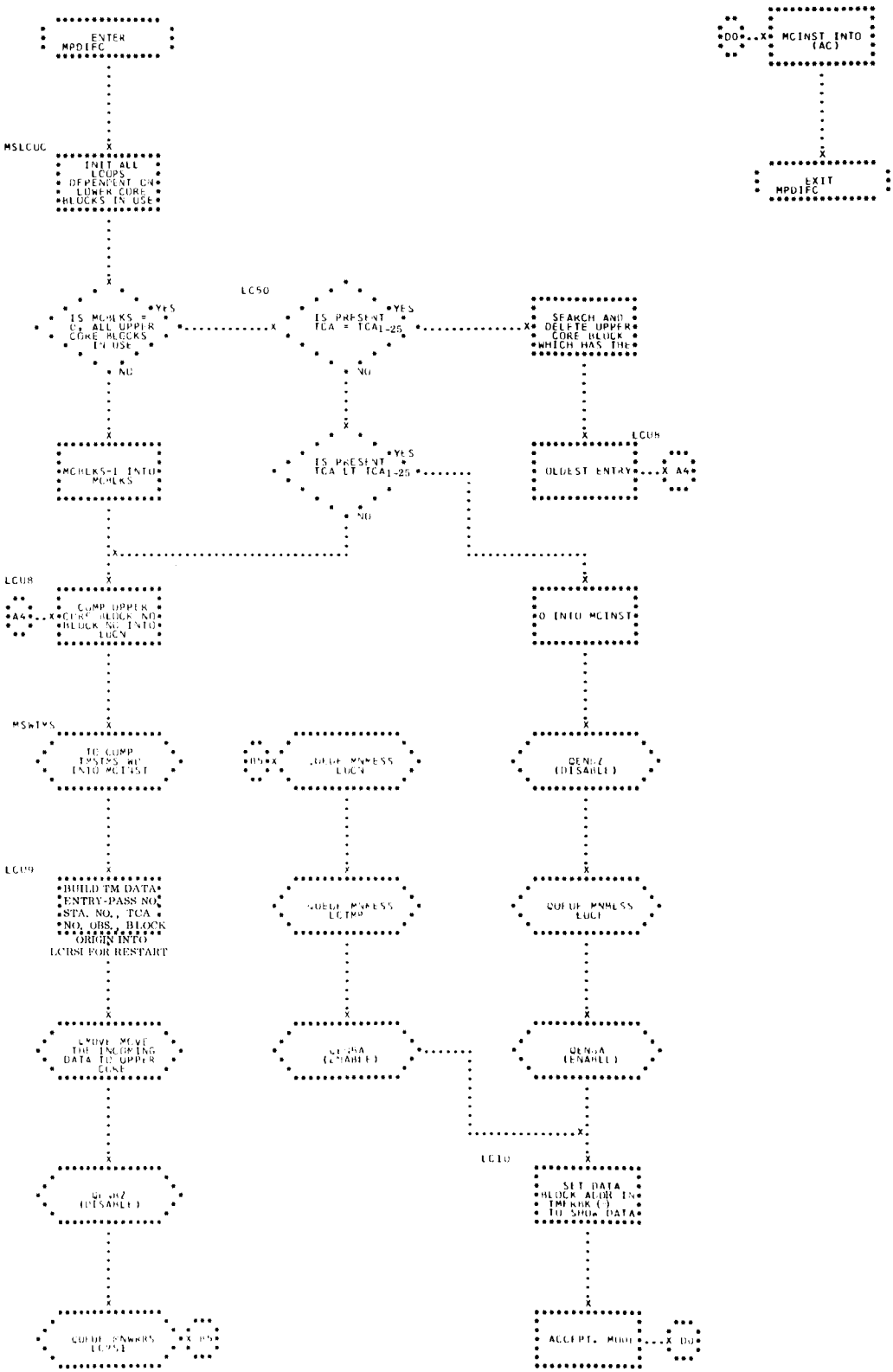
K00000
K00001
K00003
K00009

H) TIME REQUIRED EXCLUDING SUBROUTINE -

MINIMUM - 0.847 MILLISECOND
MAXIMUM - 1.355 MILLISECONDS

TMDATA TABLE FORMAT

TM DATA	STATION	S	DECREMENT	T	ADDRESS
		+	PASS NUMBER	0,	STA. IDENT. NO.
+1	1	+	LOC. OF STA. MESSAGE BLOCK IN UPPER CORE	0,	NO. RADAR OBS.
+2		+	FIXED-PT. SEC. t_{ca}	0,	FIXED-PT. MIN. t_{ca}
•					
•	2				
•					
•	•				
•	•				
•	•				
•	•				
+72		+	PASS NUMBER	0,	STA. IDENT. NO.
+73	MNUCMB*3	+	LOC. OF STA. MESSAGE BLOCK IN UPPER CORE	0,	NO. RADAR OBS.
+74		+	FIXED-PT. SEC. t_{ca}	0,	FIXED-PT. MIN. t_{ca}


ENTER MPDIFC
MSLCUC
INIT ALL LOOPS DEPENDENT ON LOWER CORE BLOCKS IN USE
IS MCBLKS = 0, ALL UPPER CORE BLOCKS IN USE
YES
NO
LC50
IS PRESENT TCA = TCA1-25
SEARCH AND DELETE UPPER CORE BLOCK WHICH HAS THE
MCBLKS-1 INTO MCBLKS
IS PRESENT TCA LT TCA1-25
LC08
OLDEST ENTRY
A4
LCU8
COMP UPPER CORE BLOCK NO BLOCK NO INTO LUCN
0 INTO MCINST
MSWTMS
TO COMP TMSTMS WD INTO MCINST
QUEUE MNMESS LUCN
QENBZ (DISABLE)
LCU9
BUILD TM DATA ENTRY-PASS NO STA. NO., TCA NO. OBS., BLOCK ORIGIN INTO LCRSI FOR RESTART
QUEUE MNMESS LCTMP
QUEUE MNMESS LUCF
CMOVE MOVE THE INCOMING DATA TO UPPER CORE
QENBA (ENABLE)
QENBZ (DISABLE)
LC10
SET DATA BLOCK ADDR IN TMFRBK (0) TO SHOW DATA
QUEUE MNWRRS LCRSI
ACCEPT. MODE
DO
MCINST INTO (AC)
EXIT MPDIFC

4. MSWTMS-MONITOR SUBROUTINE TO COMPUTE A TMDATA BLOCK ADDRESS

MSWTMS RECEIVES CONTROL WHEN THE ADDRESS OF A TMDATA BLOCK IS TO BE COMPUTED. (SEE FLOWCHART)

INPUT

A) AC — ADDRESS PORTION CONTAINS THE BLOCK NUMBE OF A LOW-SPEED EDITED DATA BLOCK LOCATED IN THE 32K BUFFERING MODULE WHOSE CORRES PONDING BLOCK ADDRESS IN TMDATA IS TO BE COMPUTED.

B) MCTMDA — ADDRESS PORTION CONTAINS LOCATION OF THE TMDATA BLOCK.

OUTPUT

A) MCINST — ADDRESS PORTION CONTAINS THE COMPUTED TMDATA ADDRESS.

B) AC — CONTAINS CONTENTS OF MCINST ON EXIT.

METHOD

MSWTMS DETERMINES THE ADDRESS OF A 3-WORD TMDATA BLOCK BY MULTIPLYING THE BLOCK NUMBER MINUS ONE BY THREE, AND THEN ADDS THE PRODUCT TO THE BASE ADDRESS OF TMDATA.

USAGE

MSWTMS IS USED BY MPDIFM AND MSLCUC.

A) CALLING SEQUENCE -

```
TSX     MSWTMS, 4
        RETURN
```

B) STORAGE REQUIRED -

10 LOCATIONS

C) CONSTANTS -

K00001

K00003

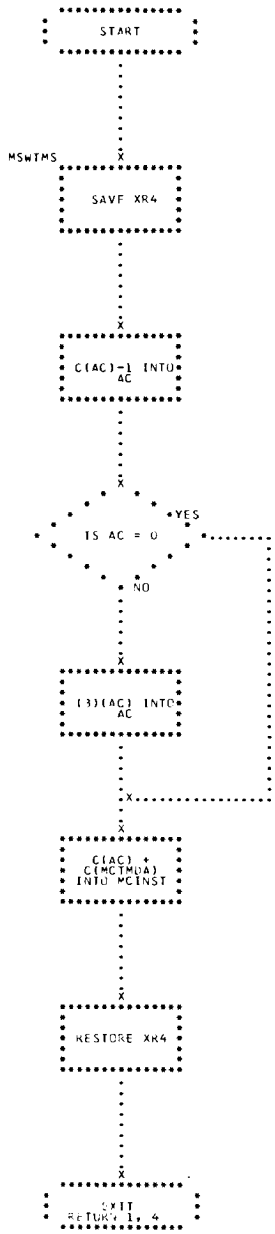
D) COMMUNICATION CELLS -

MCINST

MCTMDA

I) TIME REQUIRED -

26 TO 35 MICROSECONDS



A(MINST) =
TMDATA ADDR.

5. MSDELT - MONITOR SUBROUTINE TO DELETE AN ENTRY IN THE TMSTMS TABLE

MSDELT RECEIVES CONTROL FROM MPDIFM WHEN DATA FROM A STATION IS TO BE EXCLUDED FROM THE DIFFERENTIAL CORRECTION PROCESS. EXCLUDING STATION DATA IS ACCOMPLISHED BY DELETING ITS ENTRY IN THE TMSTMS TABLE. (SEE FLOWCHART.)

INPUT

A) MCBKNO CONTAINS IN ITS ADDRESS FIELD THE BLOCK NUMBER OF THE ENTRY TO BE DELETED FROM TMSTMS.

B) MCINST CONTAINS IN ITS ADDRESS FIELD THE ADDRESS OF THE 3-WORD TMDATA BLOCK CORRESPONDING TO THE ENTRY TO BE DELETED FROM TMSTMS.

C) TMSTMS A TABLE OF MNLTMS ENTRIES, CONTAINING IN THE ADDRESS FIELD OF EACH ENTRY THE LOCATION OF THE TMDATA BLOCK ASSIGNED TO EACH REPORTING STATION WHICH IS TO BE USED IN THE DIFFERENTIAL CORRECTION. THE ENTRIES ARE ORDERED ACCORDING TO THE TIME OF MINIMUM RANGE (TMR), WITH THE ENTRY FOR THE MOST RECENT TIME FIRST. THE LAST ENTRY IN THE TABLE IS SET TO NEGATIVE, ALL PRECEDING ENTRIES ARE POSITIVE, WHILE ALL REMAINING LOCATIONS ARE NEGATIVE ZERO.

OUTPUT

A) TMSTMS TABLE IS UPDATED TO REFLECT THE LATEST DELETION.

B) MESSAGES - QUEUED FOR ON-LINE PRINTING TO SIGNAL SUCCESSFUL OR UNSUCCESSFUL DELETION OF THE APPROPRIATE TMSTMS ENTRY.

METHOD

THE APPROPRIATE ENTRY IS DELETED FROM TMSTMS BY MATCHING THE TMDATA ADDRESS OF THE STATION TO BE DELETED WITH ALL ENTRIES IN THE TMSTMS BLOCK. WHEN A MATCH IS FOUND THE BLOCK IS SHIFTED UP BY ONE LOCATION, WITH THE LOCATION HAVING THE MATCH BEING THE FIRST TO BE OVERLAIN. THE LAST ENTRY IN THE TMSTMS BLOCK IS THEN SET TO MINUS #ERO AND A SUCCESSFUL DELETION MESSAGE IS QUEUED FOR PRINTING.

IF A ZERO ENTRY HAS BEEN ENCOUNTERED IN TMSTMS, OR THE LAST LOCATION IN THE TABLE HAS BEEN INTERROGATED WITHOUT A SUCCESSFUL COMPARISON, THE SEARCH IS TERMINATED AS AN UNSUCCESSFUL DELETION MESSAGE IS QUEUED FOR PRINTING.

RETURN IS THEN MADE TO ONE LOCATION AFTER THE POINT OF TRANSFER TO THE SUBROUTINE.

USAGE

MSDELT IS USED ONLY BY MPDIFM.

A) CALLING SEQUENCE -

```
TSX     MSDELT,4
        RETURN
```

B) STORAGE REQUIRED -

32 LOCATIONS EXCLUSIVE OF MACROS

C) MACROS -

QENBA

QENBZ

QUEUE

D) PARAMETERS -

MNLTMS

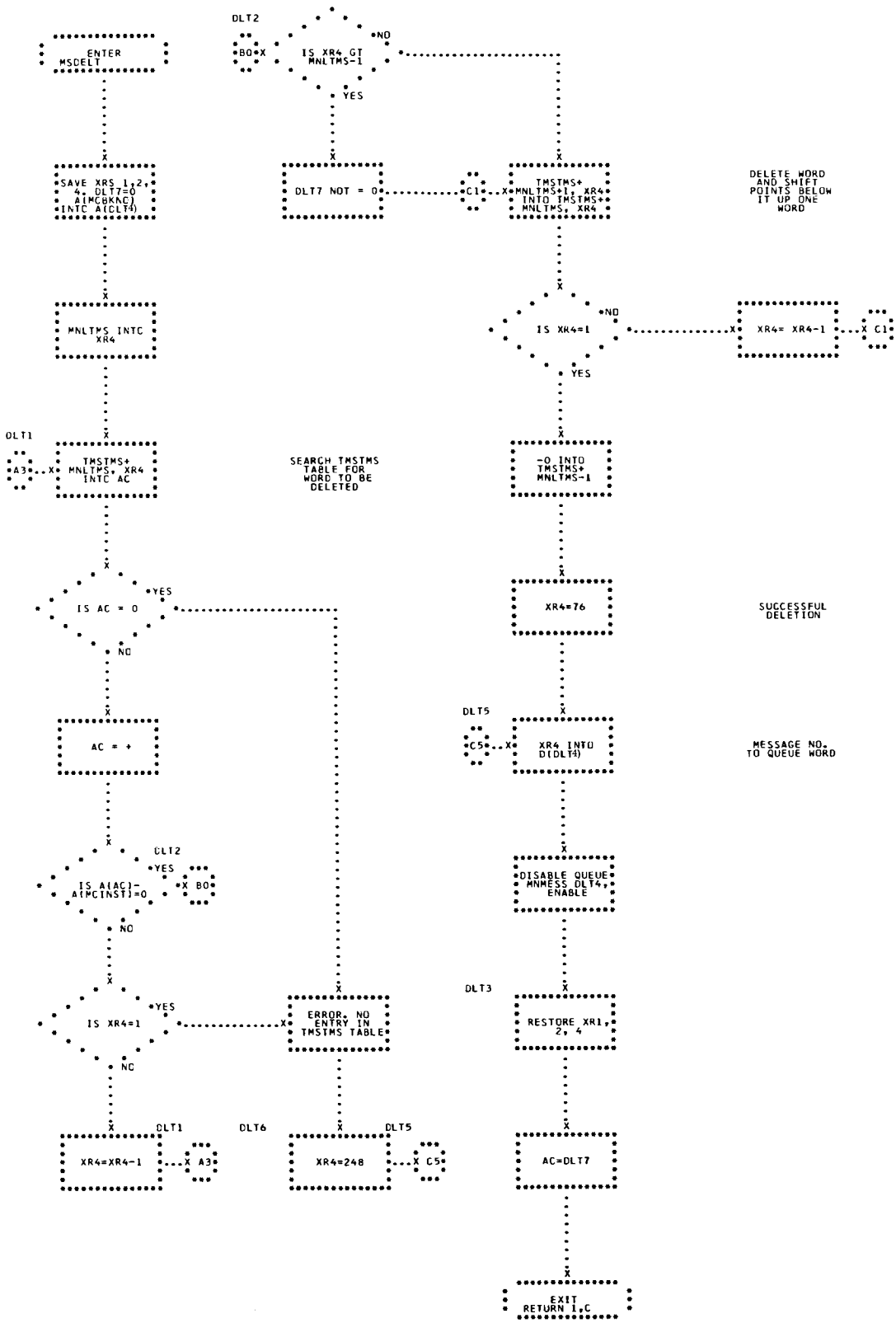
MNMESS

E) CONSTANT -

KM0000

F) TIME REQUIRED -

0.057 TO 0.763 MILLISECOND, EXCLUSIVE OF MACROS AND ASSUMING MNLTMS = 25.


ENTER MSDELT
SAVE XR$ 1,2, 4, DLT7=0 A(MCBKNC) INTC A(CLT4)
MNLTMS INTC XR4
DLT1
A3
TMSTMS+ MNLTMS, XR4 INTC AC
IS AC = 0
YES
NO
AC = +
DLT2
IS A(AC)- A(PCINST)=0
YES
B0
NO
IS XR4=1
YES
NO
XR4=XR4-1
DLT1
A3
DLT2
B0
IS XR4 GT MNLTMS-1
NO
YES
DLT7 NOT = 0
C1
SEARCH TMSTMS TABLE FOR WORD TO BE DELETED
ERROR, NO ENTRY IN TMSTMS TABLE
DLT6
DLT5
XR4=248
C5
TMSTMS+1, XR4 INTO TMSTMS+ MNLTMS, XR4
DELETE WORD AND SHIFT POINTS BELOW IT UP ONE WORD
IS XR4=1
NO
YES
XR4= XR4-1
C1
-0 INTO TMSTMS+ MNLTMS-1
XR4=76
SUCCESSFUL DELETION
DLT5
C5
XR4 INTO D(DLT4)
MESSAGE NO. TO QUEUE WORD
DISABLE QUEUE MNMESS DLT4, ENABLE
DLT3
RESTORE XR1, 2, 4
AC=DLT7
EXIT RETURN 1,C

6. MSTINT - SUBROUTINE TO REMOVE RADAR OBSERVATIONS FROM DIFFERENTIAL CORRECTION

MSTINT REMOVES RADAR OBSERVATIONS FROM DIFFERENTIAL CORRECTION (DC) THAT ARE TOO OLD TO ACCURATELY INFLUENCE PREDICTED TRAJECTORY BY UPDATING TMSTMS, A REFERENCE TABLE TO DATA BLOCKS CONTAINING RADAR OBSERVATIONS. (SEE FLOWCHART.)

INPUT

A) TMSTMS - A TABLE OF MNLTMS ENTRIES CONTAINING IN THE ADDRESS OF EACH ENTRY THE LOCATION OF THE TMDATA BLOCK ASSIGNED TO EACH RADAR STATION USED IN DC. ENTRIES ARE ORDERED BY THEIR RESPECTIVE TIMES OF CLOSEST APPROACH, FROM NEWEST TO OLDEST, AS THE TABLE PROCEEDS FROM TOP TO BOTTOM. THE OLDEST ENTRY IN THE DC INTERVAL IS NEGATIVE. ALL REMAINING LOCATIONS ARE NEGATIVE ZEROS.

B) MNLTMS - WORD EQUATED WITH THE GREATEST POSSIBLE NUMBER OF ENTRIES IN TMSTMS.

C) TMDATA - REFERENCED BY TMSTMS TO OBTAIN CORRESPONDING TIMES OF CLOSEST APPROACH.

D) MCTM80 - CONTAINS IN ITS ADDRESS THE CURRENT GMT IN FIXED-POINT MINUTES.

E) MCMXDC - CONTAINS IN ITS ADDRESS THE MAXIMUM ALLOWABLE DELAY IN MINUTES WHEN ACCEPTING RADAR OBSERVATIONS FOR THE DC. OBSERVATIONS TAGGED WITH A TIME OF CLOSEST APPROACH OLDER THAN MCMXDC

MINUTES FROM CURRENT (MCTM80) TIME ARE OMITTED FROM THE DC.

OUTPUT

A) TMSTMS IS UPDATED, I.E., ENTRIES OLDER THAN CURRENT GMT MINUS THE MAXIMUM DELAY ARE SET TO NEGATIVE ZEROS.

B) THE AC IS NORMALLY SET TO ZERO ON EXIT. IF ALL THE ENTRIES IN TMSTMS ARE OLDER THAN THE MAXIMUM ALLOWABLE DELAY, THE AC IS SET TO NONZERO.

METHOD

MSTINT SETS EACH ENTRY IN TMSTMS, WHICH HAS A CORRESPONDING TIME OF CLOSEST APPROACH (t_{ca}) OLDER THAN MCMXDC MINUTES FROM CURRENT GMT, EQUAL TO MINUS ZERO THEREBY REMOVING OBSERVATIONS FROM DC THAT ARE TOO OLD.

THE T_{ca} OF EACH ENTRY IN TMSTMS IS OBTAINED FROM THE SECOND WORD IN THE TMDATA BLOCK REFERENCED BY THAT ENTRY. THE TIME ASSOCIATED WITH EACH TMSTMS ENTRY, PROCEEDING FROM NEWEST TO OLDEST, IS COMPARED WITH THE TIME MCMXDC MINUTES AGO, UNTIL AN ENTRY OLDER THAN THIS TIME IS LOCATED OR UNTIL ALL ENTRIES IN THE TABLE ARE EXAMINED. ENTRIES IN TMSTMS WITH TIMES EARLIER THAN MCMXDC ARE SET TO MINUS ZEROS. THE LAST ACCEPTABLE ENTRY IN THE CORRECTION INTERVAL IS SET TO NEGATIVE.

USAGE

MSTINT IS ENTERED FROM MPDIFC OR MPDIFM AND RETURNS TO THE PROGRAM FROM WHICH IT WAS ENTERED.

A) STORAGE REQUIRED -
34 LOCATIONS

B) MACROS -

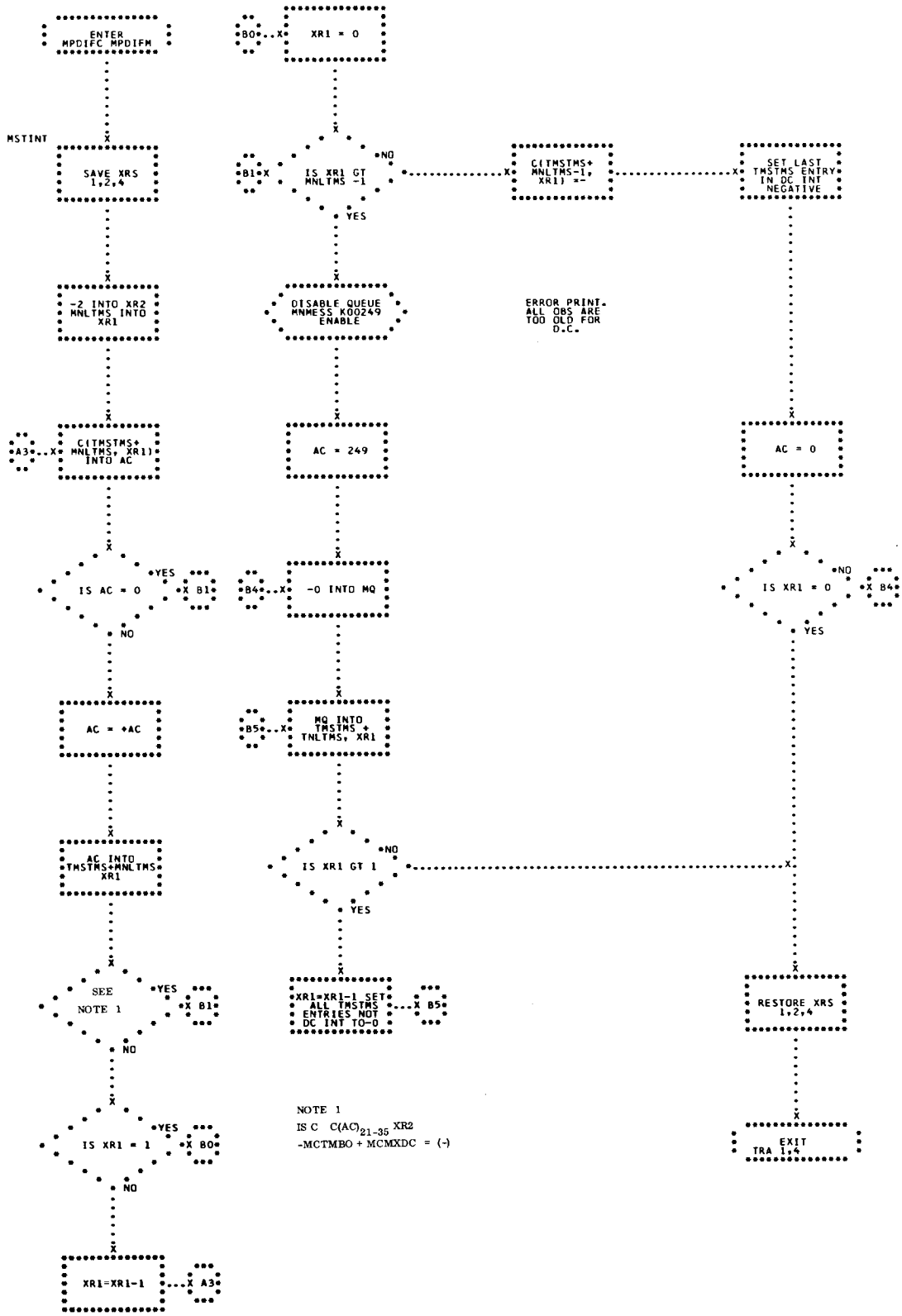
QENBZ
QENBA
QUEUE

C) PARAMETERS -
MNMESS
MNLTMS

D) COMMUNICATION CELLS -
MCTM80
MCMXDC

E) TABLES -
TMSTMS
TMDATA (VIA TMSTMS)

F) CONSTANTS -
KNSTKA
K00000
KM0000
K00249


ENTER MPDIFC MPDIFM
MSTINT
SAVE XRS 1,2,4
-2 INTO XR2 MNLTMS INTO XR1
C(TMSTMS+ MNLTMS, XR1) INTO AC
IS AC = 0
YES
NO
AC = +AC
AC INTO TMSTMS+MNLTMS XR1
SEE NOTE 1
YES
NO
IS XR1 = 1
YES
NO
XR1=XR1-1
A3
B1
XR1 = 0
IS XR1 GT MNLTMS -1
NO
YES
B0
B1
DISABLE QUEUE MNMESS K00249 ENABLE
AC = 249
-0 INTO MQ
MQ INTO TMSTMS + TNLTMS, XR1
IS XR1 GT 1
NO
YES
XR1=XR1-1 SET ALL TMSTMS ENTRIES NOT DC INT TO-0
B4
B5
B5
NOTE 1
IS C C(AC)21-35 XR2
-MCTMBO + MCMXDC = (-)
C(TMSTMS+ MNLTMS-1, XR1) =-
ERROR PRINT. ALL OBS ARE TOO OLD FOR D.C.
SET LAST TMSTMS ENTRY IN DC INT NEGATIVE
AC = 0
IS XR1 = 0
NO
YES
B4
RESTORE XRS 1,2,4
EXIT TRA 1,4

7. MSINSR - SUBROUTINE TO INSERT RADAR OBSERVATIONS INTO DIFFERENTIAL CORRECTION

MSINSR INSERTS DESIRED RADAR OBSERVATIONS INTO DIFFERENTIAL CORRECTION (DC) BY UPDATING TMSTMS WITH NEW ENTRIES - TMSTMS PROVIDES THE REFERENCE TO DATA BLOCK CONTAINING RADAR OBSERVATIONS. (SEE FLOWCHART)

INPUT

A) MCINST CONTAINS THE NEW ENTRY TO BE INSERTED IN TMSTMS. THE DECREMENT OF MCINST CONTAINS ZERO. THE ADDRESS CONTAINS THE LOCATION OF THE 3-WORD TMDATA BLOCK WHICH IS CORRELATED WITH EACH NEW SET OF EDITED RADAR OBSERVATIONS TO BE ADDED TO THE DC.

B) MCBKNO CONTAINS IN ITS ADDRESS THE BLOCK NUMBER OF THE NEW EDITED RADAR OBSERVATIONS. EACH SET (BLOCK) OF OBSERVATIONS IN THE 32K BUFFER MODULE HAS A BLOCK NUMBER AND A CORRESPONDING 3-WORD ENTRY IN TMDATA, NUMBERED SEQUENTIALLY STARTING FROM THE TOP OF TMDATA.

C) MCTM80 CONTAINS IN ITS ADDRESS THE CURRENT GMT IN MINUTES.

D) MCMXDC CONTAINS IN ITS ADDRESS THE MAXIMUM ALLOWABLE DELAY IN MINUTES WHEN ACCEPTING RADAR OBSERVATIONS FOR DC. OBSERVATIONS TAGGED WITH A TIME OF CLOSEST APPROACH OLDER THAN MCMXDC MINUTES FROM CURRENT (MCTM80) TIME ARE

OMITTED FROM DC.

E) TMSTMS A TABLE OF MNLTMS ENTRIES CONTAINING IN THE ADDRESS OF EACH ENTRY THE LOCATION OF THE TMDATA BLOCK ASSIGNED TO EACH RADAR STATION USED IN DC. ENTRIES ARE ORDERED BY THEIR RESPECTIVE TIMES OF CLOSEST APPROACH (CONTAINED IN TMDATA), FROM NEWEST TO OLDEST, AS THE TABLE PROCEEDS FROM TOP TO BOTTOM. THE OLDEST ENTRY IN THE DC INTERVAL IS NEGATIVE. ALL REMAINING LOCATIONS ARE NEGATIVE ZEROS.

F) MNLTMS WORD EQUATED WITH THE GREATEST POSSIBLE NUMBER OF ENTRIES IN TMSTMS.

G) TMDATA LOGICALLY DIVIDED INTO BLOCKS OF THREE WORDS EACH. EACH OF THESE 3-WORD ENTRIES CORRESPONDS TO A SET OF RADAR OBSERVATIONS IN THE 32K BUFFER MODULE. MSINSR USES THE ASSOCIATED TIMES OF CLOSEST APPROACH FOUND IN THE THIRD CELL OF EACH ENTRY. MINUTES ARE LOCATED IN THE ADDRESS FIELD AND SECONDS IN THE DECREMENT.

OUTPUT

A) TMSTMS AN UPDATED TABLE SEQUENCED ON THE BASIS OF THE TIMES OF CLOSEST APPROACH IN TMDATA, THE MOST RECENT ENTRY APPEARING AT THE TOP OF THE TABLE.

B) MCLENT CONTAINS THE LOCATION OF THE NEW ENTRY IN THE TMSTMS TABLE RELATIVE TO THE FIRST LOCATION IN THE TABLE. MCLENT

RANGES FROM ZERO TO (MNLTMS-1).

C) AC — ZERO ON EXITING FROM MSINSR IF THE NEW ENTRY WAS SUCCESSFULLY INSERTED IN TMSTMS. IF THE NEW ENTRY COULD NOT BE INSERTED, THE AC IS SET TO NONZERO.

D) MYMESS — QUEUED TO PRINT ON-LINE THE SUCCESS OR FAILURE OF ADDING NEW OBSERVATIONS.

METHOD

THE TIME OF CLOSEST APPROACH (t_{ca}) IN THE THIRD WORD OF EVERY TMDATA BLOCK REFERENCED BY EACH TMSTMS ENTRY IS OBTAINED AND COMPARED WITH THE t_{ca} OF THE NEW OBSERVATIONS. THE CONTENTS OF MCINST IS INSERTED IN THE FIRST TMSTMS CELL HAVING t_{ca} OLDER THAN OR EQUAL TO THAT ASSOCIATED WITH MCINST. AFTER EACH NEW OBSERVATION IS ADDED THE ENTRIES IN TMSTMS ARE SHIFTED DOWN ONE PLACE IN THE TABLE. IF THE TABLE WAS FULL, THE OLDEST ENTRY IN TMSTMS IS REMOVED.

MSINSR WILL NOT INCORPORATE THE NEW OBSERVATIONS INTO THE DC IF -

A) t_{ca} OF THE NEW OBSERVATION IS OLDER THAN MCMXDC MINUTES.

B) MPDIFM ATTEMPTS TO INSERT OBSERVATIONS INTO THE DC THAT ARE ALREADY PRESENT.

C) t_{ca} OF THE NEW OBSERVATION IS OLDER THAN THAT OF THE LAST ENTRY IN TMSTMS, PROVIDED THE TABLE IS FULL.

THE SUCCESS OR FAILURE OF INSERTING A NEW BLOCK OF OBSERVATIONS INTO DC IS PRINTED ON-LINE, ALONG WITH THE ASSOCIATED BLOCK NUMBER, FOR POSSIBLE USE WITH THE MANUAL INTERVENTION CAPABILITY OF THE REAL-TIME GEMINI SYSTEM. THIS CAPABILITY ENABLES AN OPERATOR TO INSERT OR DELETE

A BLOCK OF RADAR OBSERVATIONS VIA THE PCC.

USAGE

MSINSR IS ENTERED FROM MPDIFC OR MPDIFM AND RETURNS TO THE PROGRAM FROM WHICH IT WAS ENTERED.

A) STORAGE REQUIRED -

66 LOCATIONS

B) MACROS -

QENBZ

QENBA

QUEUE

C) PARAMETERS -

MNMESS

MNLTMS

D) COMMUNICATION CELLS -

MCBKNO

MCINST

MCMXDC

MCTM80

MCLENT

E) TABLES -

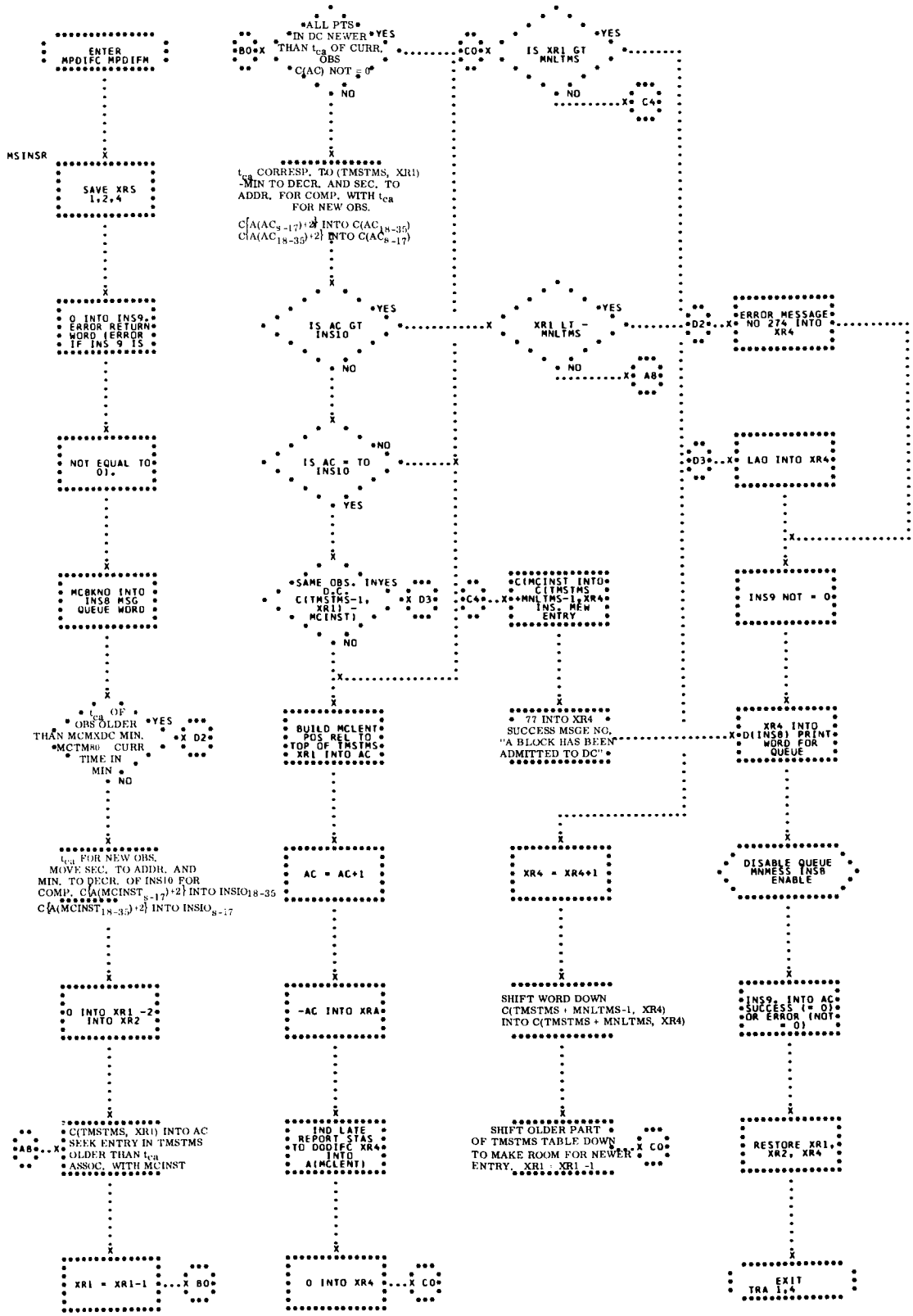
TMSTMS

TMDATA (VIA TMSTMS)

F) CONSTANTS -

K00001

KMSTKA


ENTER MPDIFC MPDIFM
MSINSR
SAVE XRS 1,2,4
0 INTO INS9, ERROR RETURN WORD (ERROR IF INS 9 IS NOT EQUAL TO 0).
MCBKNO INTO INS8 MSG QUEUE WORD
IS t_{ca} OF OBS OLDER THAN MCMXDC MIN. MCTM$_{80}$ CURR TIME IN MIN
YES
NO
D2
t_{ca} FOR NEW OBS, MOVE SEC. TO ADDR. AND MIN. TO DECR. OF INS10 FOR COMP. C{A(MCINST$_{8-17}$)+2} INTO INSIO$_{18-35}$ C{A(MCINST$_{18-35}$)+2} INTO INSIO$_{8-17}$
0 INTO XR1 -2 INTO XR2
C(TMSTMS, XR1) INTO AC SEEK ENTRY IN TMSTMS OLDER THAN t_{ca} ASSOC. WITH MCINST
AB
XR1 = XR1-1
B0
ALL PTS IN DC NEWER THAN t_{ca} OF CURR. OBS C(AC) NOT = 0
t_{ca} CORRESP. TO (TMSTMS, XR1) -MIN TO DECR. AND SEC. TO ADDR. FOR COMP. WITH t_{ca} FOR NEW OBS.
C{A(AC$_{8-17}$)+2} INTO C(AC$_{18-35}$) C{A(AC$_{18-35}$)+2} INTO C(AC$_{8-17}$)
IS AC GT INS10
IS AC = TO INS10
SAME OBS. INYES D.C. C(TMSTMS-1, XR1) - MC(INST)
C0
C4
BUILD MCLENT POS REL TO TOP OF TMSTMS XR1 INTO AC
AC = AC+1
-AC INTO XR4
IND LATE REPORT STAS TO DODIFC XR4 INTO A(MCLENT)
0 INTO XR4
IS XR1 GT MNLTMS
XR1 LT - MNLTMS
A8
C(MCINST INTO C(TMSTMS + MNLTMS-1, XR4 INS. MEN ENTRY
77 INTO XR4 SUCCESS MSGE NO. "A BLOCK HAS BEEN ADMITTED TO DC"
XR4 = XR4+1
SHIFT WORD DOWN C(TMSTMS + MNLTMS-1, XR4) INTO C(TMSTMS + MNLTMS, XR4)
SHIFT OLDER PART OF TMSTMS TABLE DOWN TO MAKE ROOM FOR NEWER ENTRY. XR1 = XR1 -1
ERROR MESSAGE NO 274 INTO XR4
LAQ INTO XR4
INS9 NOT = 0
XR4 INTO D(INS8) PRINT WORD FOR QUEUE
DISABLE QUEUE MNMESS INS8 ENABLE
INS9, INTO AC SUCCESS (= 0) OR ERROR (NOT = 0)
RESTORE XR1, XR2, XR4
EXIT TRA 1,4

8. MYOUR1 - ORBIT BASIC CYCLE CONTROL PROCESSOR

PROCESSING AND CONTROL FUNCTIONS OF THE BASIC ORBIT ARE COORDINATED BY MYOUR1. INCLUDED IN THIS CYCLE ARE THE GENERATION OF MANEUVER-REFLECTING VECTORS, AN ORBIT EPHEMERIS AND IF NEEDED, A REENTRY EPHEMERIS, COMPUTATION OF APOGEE AND PERIGEE AND DETERMINATION OF TIMES-TO-FIRE FOR THE NEXT CONTINGENCY AND PRIMARY RECOVERY AREAS. FOR EVERY NEW DEFINITION OR ADJUSTMENT OF THE ORBITAL TRAJECTORY, THE PROCEDURE TO UPDATE TIME-RELATED OUTPUT IS BEGUN BY MYOUR1. IT IS IN THIS CAPACITY THAT MYOUR1 PROVIDES DIRECTION FOR THE OTHER PROCESSORS OF THE BASIC ORBIT CYCLE.

PROGRAM FUNCTIONS INCLUDE THE FOLLOWING -

1. ORBIT CATCH-UP CAPABILITY IS PROVIDED. AN INPUT VECTOR OLDER THAN 30 MINUTES IS INTEGRATED UP TO PRESENT GMT REFLECTING INTO THE FINAL ORBIT ANY REQUIRED S/C MANEUVERS. DURING THIS CATCH-UP OPERATION, A CONTINUOUS MONITOR IS MAINTAINED ON REVOLUTION NUMBER.

2. INPUT D.C. VECTORS REFLECTING A PREMANEUVER ORBIT ARE UPDATED TO INCLUDE THE NOMINAL S/C MANEUVER.

3. FOR PURPOSES OF D. C., A VECTOR HISTORY OF VARIOUS ORBITS IS MAINTAINED.

4. THE KEPLERIAN HEIGHT OF PERIGEE IS CALCULATED TO PROVIDE A BASIS FOR DETERMINING INTEGRATOR TIME STEP.

5. VECTORS REFLECTING ANY PROPOSED S/C MANEUVERS ARE GENERATED BY GVMMRF.

6. A MANEUVER-REFLECTING ORBIT EPHEMERIS, TNDIAC, IS GENERATED FOR ACQUISITION AND DISPLAY REQUIREMENTS.

7. IF PRESENT GMT IS WITHIN 50 MINUTES OF THE RETROFIRE SETTING (GMTRS), A VECTOR IS GIVEN TO MYREN2 TO GENERATE THE REENTRY EPHEMERIS.

8. AS DICTATED BY MYOUR1'S QUEUE WORD, OTHER PROCESSORS OF THE BASIC CYCLE ARE READIED FOR EXECUTION.

METHOD

REFER TO FLOWCHART.

INPUT

A) MYOUR1 QUEUE WORD -

1. +NZ = ORBIT REDEFINITION REQUEST

ADDRESS		LOCATION OF 8-WORD INPUT TABLE CONTAINING R,V,TIME AND REVOLUTION NUMBER IN THIS PRESCRIBED ORDER.
TAG		NONZERO INDICATES A D.C. VECTOR AS INPUT
DECREMENT	=	1,ORBIT REGENERATION DUE TO AN EXECUTED MANEUVER.
	=	16384,ORBIT REGENERATION DUE TO A PLANNED MANEUVER.

2. -NZ = ORBIT REGENERATION DUE TO A CHANGE IN ONE OF THE PERTURBING FACTORS, SUCH AS S/C WEIGHT, ATMOSPHERIC DENSITY, ETC.
3. +Z = 30 MINUTE ORBIT UPDATE
4. -Z = REQUEST TO GENERATE REENTRY TABLE.

B) COMMUNICATION CELLS -

MCREST	NON≠ERO INDICATES SYSTEM RESTART
MCMINS	PRESENT GMT COUNT IN MINUTES
MCHFSC	PRESENT GMT COUNT IN HALF-SECONDS

MCPHSE	PREFIX INDICATES MISSION PHASE
MCMANC	CONTAINS MANEUVER COUNT
MCREVN	CONTAINS PRESENT REVOLUTION NUMBER
MCTLST	DATA TIME CELL, SET BY D.C.
MCIMAT	CONTAINS GMTPS OF LAST INSERTED MANEUVER

C) TABLES -

TMANUV	S/C MANEUVER TABLE
TMCODW	TABLE OF S/C WEIGHTS USED FOR EXECUTED MANEUVERS
TMCODP	TABLE OF PARAMETERS ASSOCIATED WITH EXECUTED S/C MANEUVERS
TMGTRS	TABLE OF RETRO-SETTING QUANTITIES
TN4LLH	CONTAINS LATITUDE AND LONGITUDE OF IMPACT
TOTUC1	TABLE OF S/C THRUST LEVELS
TOTUC2	TABLE OF S/C MASS EXPULSION RATES
TOTUC3	TABLE OF S/C CANT ANGLES
TOTUC5	TABLE OF S/C ORIENTATION ANGLES (1)
TOTUC6	TABLE OF S/C ORIENTATION ANGLES (2)

OUTPUT

A) COMMUNICATION CELLS -

MCVPER	VACCUM PERIGEE IN N.M.
MCVPMV	CONTROL WORD FOR PRINTING

		MANEUVER-REFLECTING VECTORS
	MCVTIS	CONTAINS TIME OF IMPACT
	MCNIAC	CONTAINS INTEGRATOR ERROR NUMBER
	MCOEAV	NZ INDICATES ORBIT EPHEMERIS AVAILABLE

B) TABLES -

TMREFL	TABLE CONTAINING PRESENT ORBIT-DEFINING VECTOR AND MANEUVER-REFLECTING VECTORS
TMCCDV	CONTAINS ORBIT VECTOR HISTORY-FOR MYCOTO
TMORBD	CONTAINS PRESENT ORBIT VECTOR-FOR MYLIF8
TMREST	CONTAINS PRESENT ORBIT VECTOR-FOR MYWRRS
TDCOEF	CONTAINS PRESENT ORBIT VECTOR-FOR D.C.
TMLAST	CONTAINS VECTOR DEFINING PREVIOUS ORBIT-FOR D.C.
TMMCUR	CONTAINS INPUT TO INTEGRATOR
TRUSTS	CONTAINS INPUT TO THRUSTER
TRCUR1	CONTAINS INPUT FOR MYREN2
TPDIAC	TABLE OF INTEGRATION TABLE ADDRESSES
TMPRIP	CONTAINS S/C IMPACT POSITION
TPCUR1	TABLE OF ERROR INFORMATION

C) QUEUES FOR MYMESS,MYQPRA,MYQNEA,MYAPE3 AND MYREN2

D) 8-BIT TURNED ON FOR MYAPE3 AND MYLIF8

USAGE

A) ENTRY AND EXIT TO MOPRIO

B) COMMUNICATION CELLS -

MCREST	MCIMAT
MCMINS	MCVPER
MCHFSC	MCVPMV
MCPHSE	MCVTIS
MCMANC	MCNIAC
MCREVN	MCOEAV
MCTLST	MCVPOC

C) TABLES -

TMANUV	TOTUC6	TMMOUR
TMCODV	TN4LLH	TRUSTS
TMCODP	TNDIAC	TROUR1
TMCODW	TPDIAC	TMPRIP
TMGTRS	TMREFL	TPOUR1
TOTUC1	TMORBD	TMNMSK
TOTUC2	TMREST	TMFMSK
TOTUC3	TDCDEF	TMPRIO
TOTUC5	TMLAST	

D) CONSTANTS -

KTGMSK	KCH600	K00004
K00030	K00008	KCH233
K00001	K000P1	K77777
K00060	K001.0	K06000
KLCAPE	K.ERGU	K00.PI
K00002	KCGUNM	KKK2PI
K00049	K00090	K.1RAD
KM0000	K00006	KM0001

E) PARAMETERS -

A	MNREN2	MNTTOX
B	MNAPE3	MNTTOY
C	MNTTF4	MNQPRA
E	MNNAV5	MNQNEA
F	MNGMT6	MNTABU
G	MNMUV7	MNMESS
H	MNLIF8	MNORMC
MNBBNI	MNDIC9	MNFPPR
MNCOTO	MNCPNI	MNWRRS
MNGMT6	MNACQD	

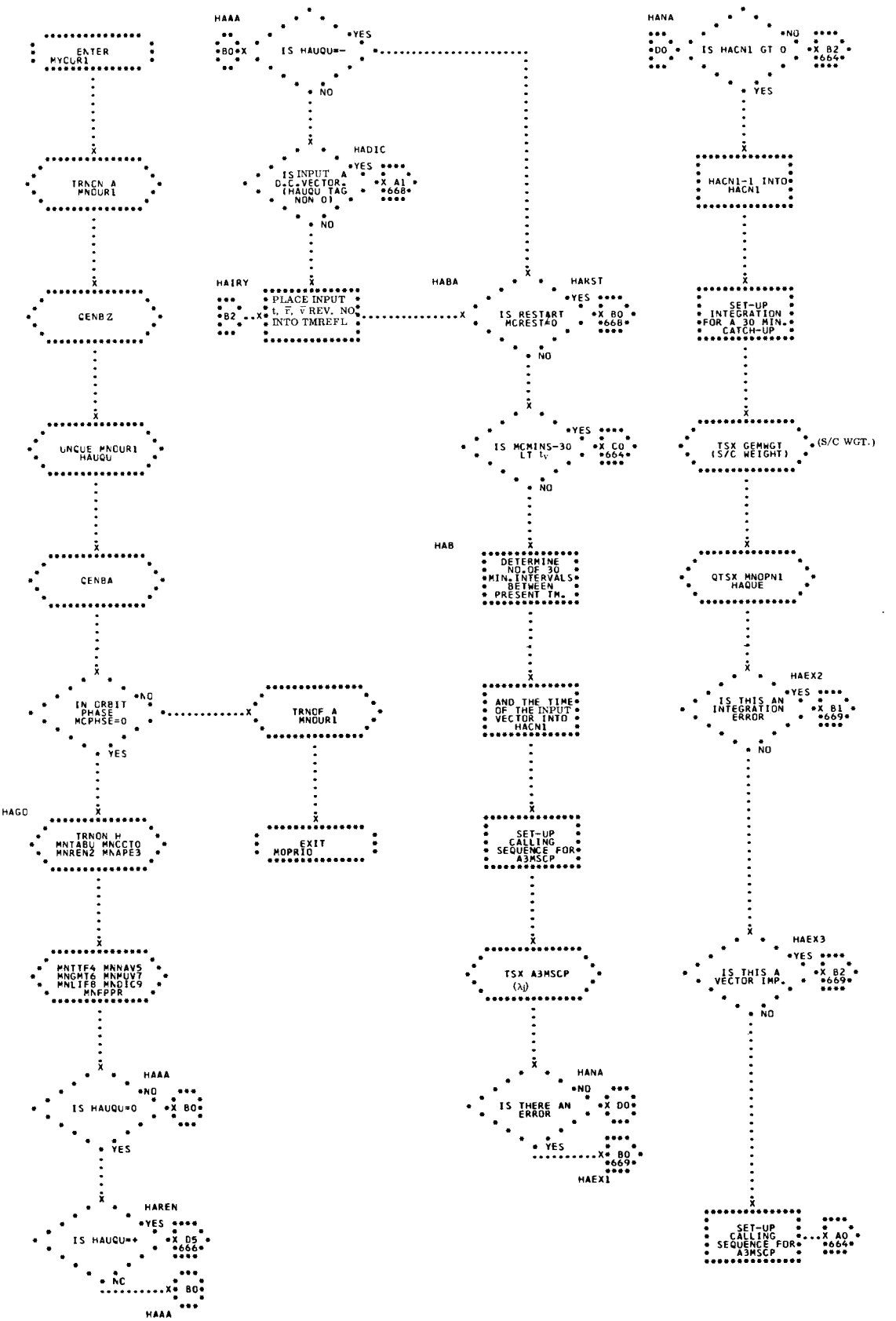
F) MACROS

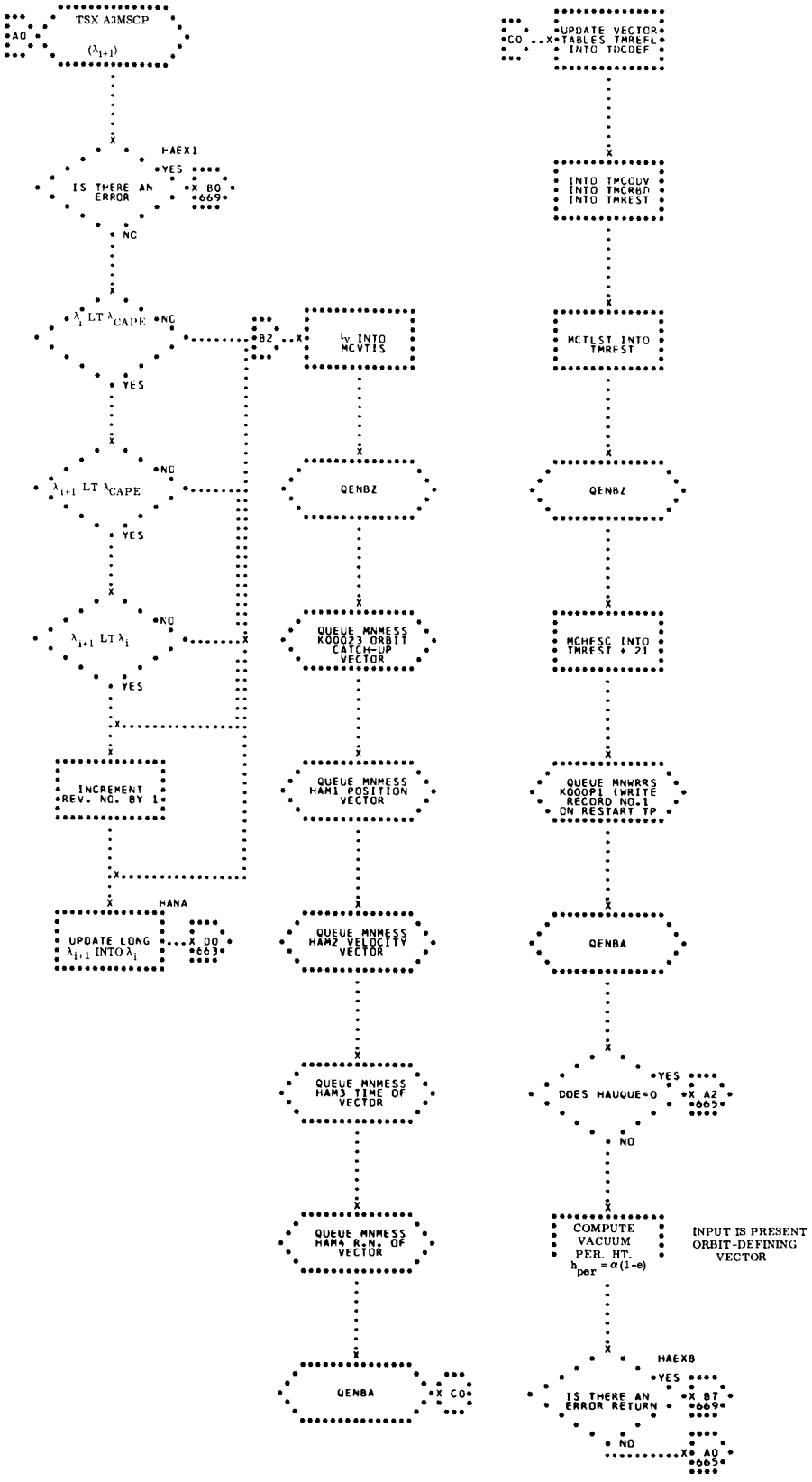
TRNON
TRNOF
QUEUE
UNQUE
QENBA
QENBZ
QTSX

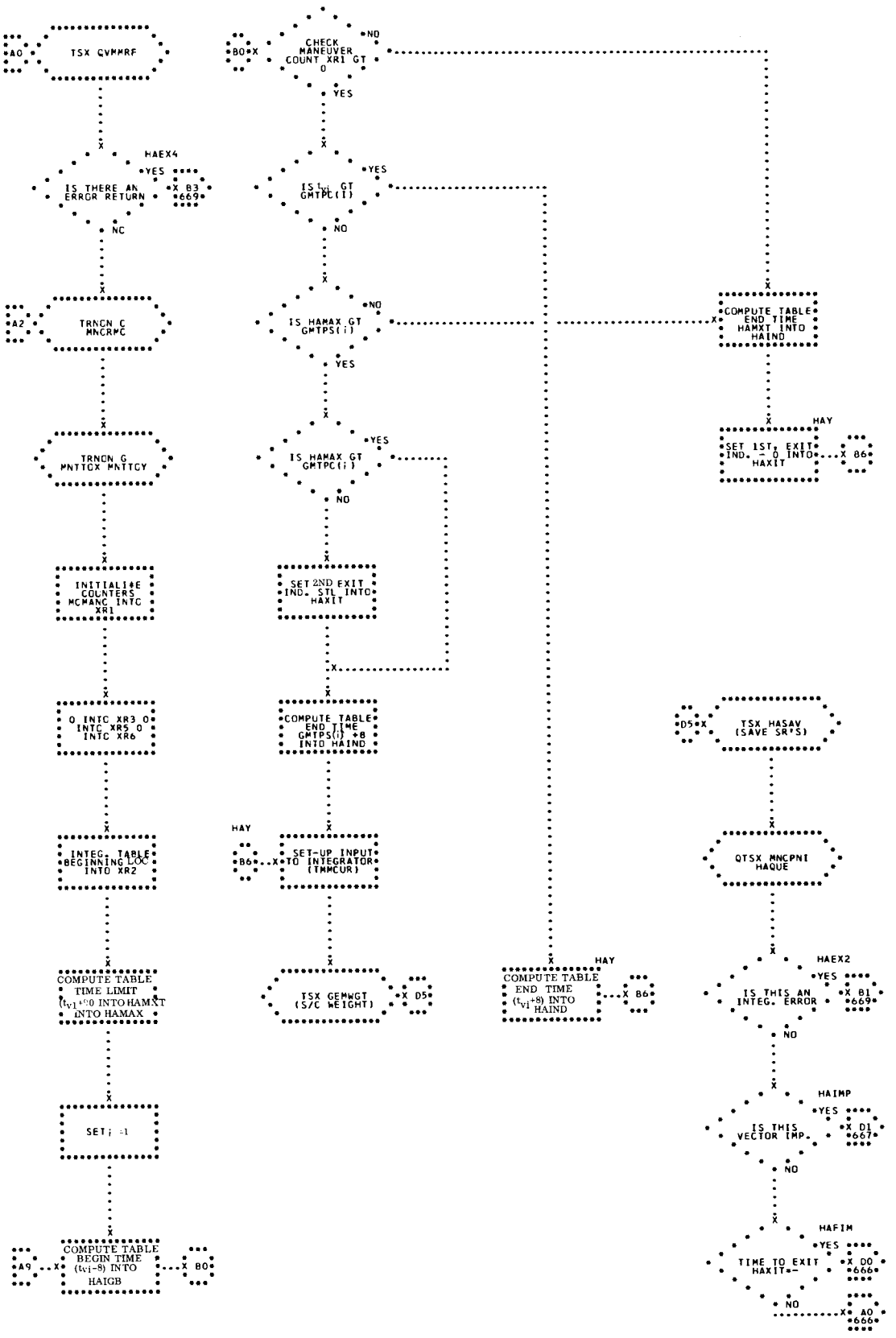
G) SUBROUTINES -

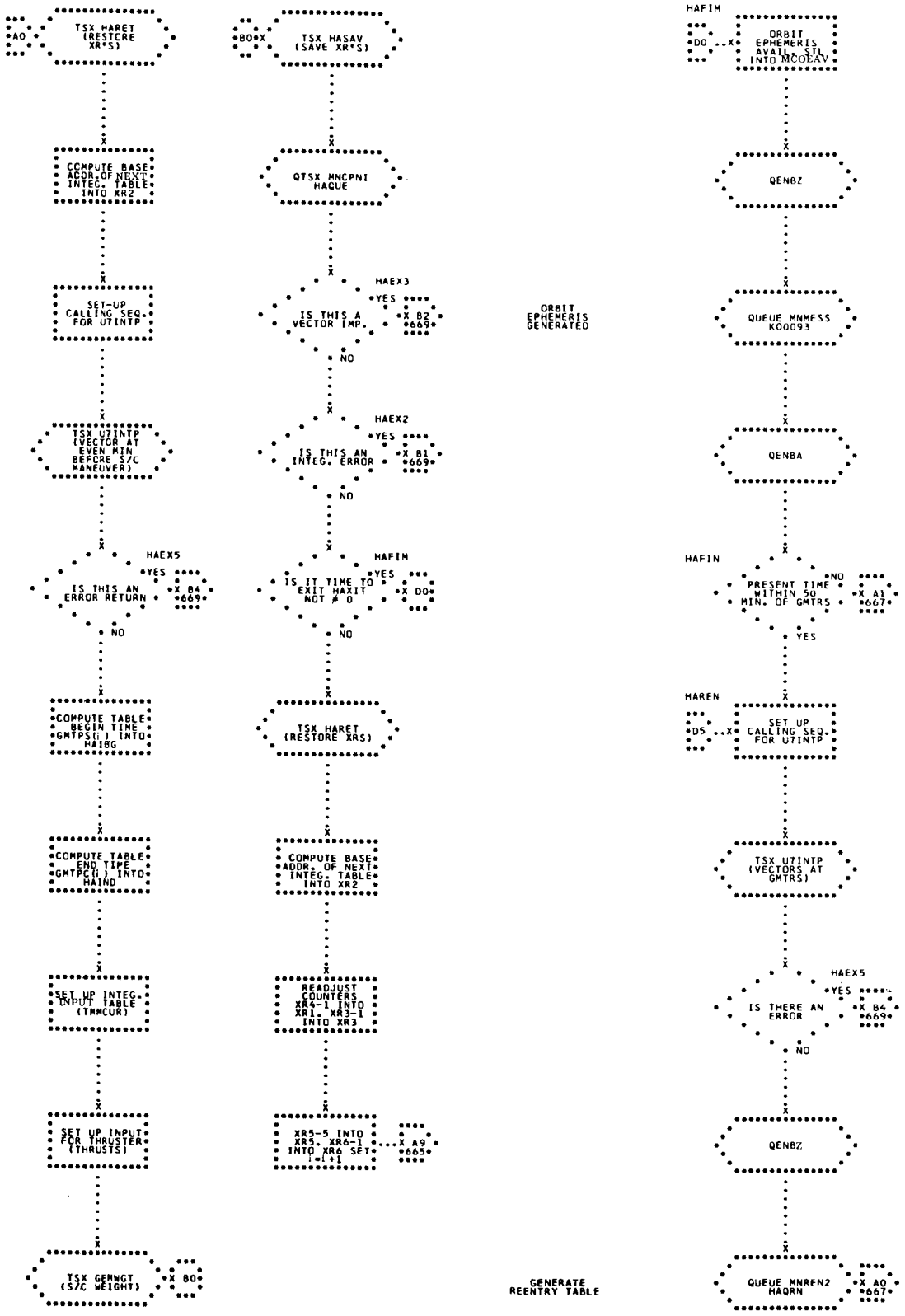
A3MSCP
GEMWGT
GVMMRF
VMAG
U3DOT
SQRT
UINTP
GINTAB

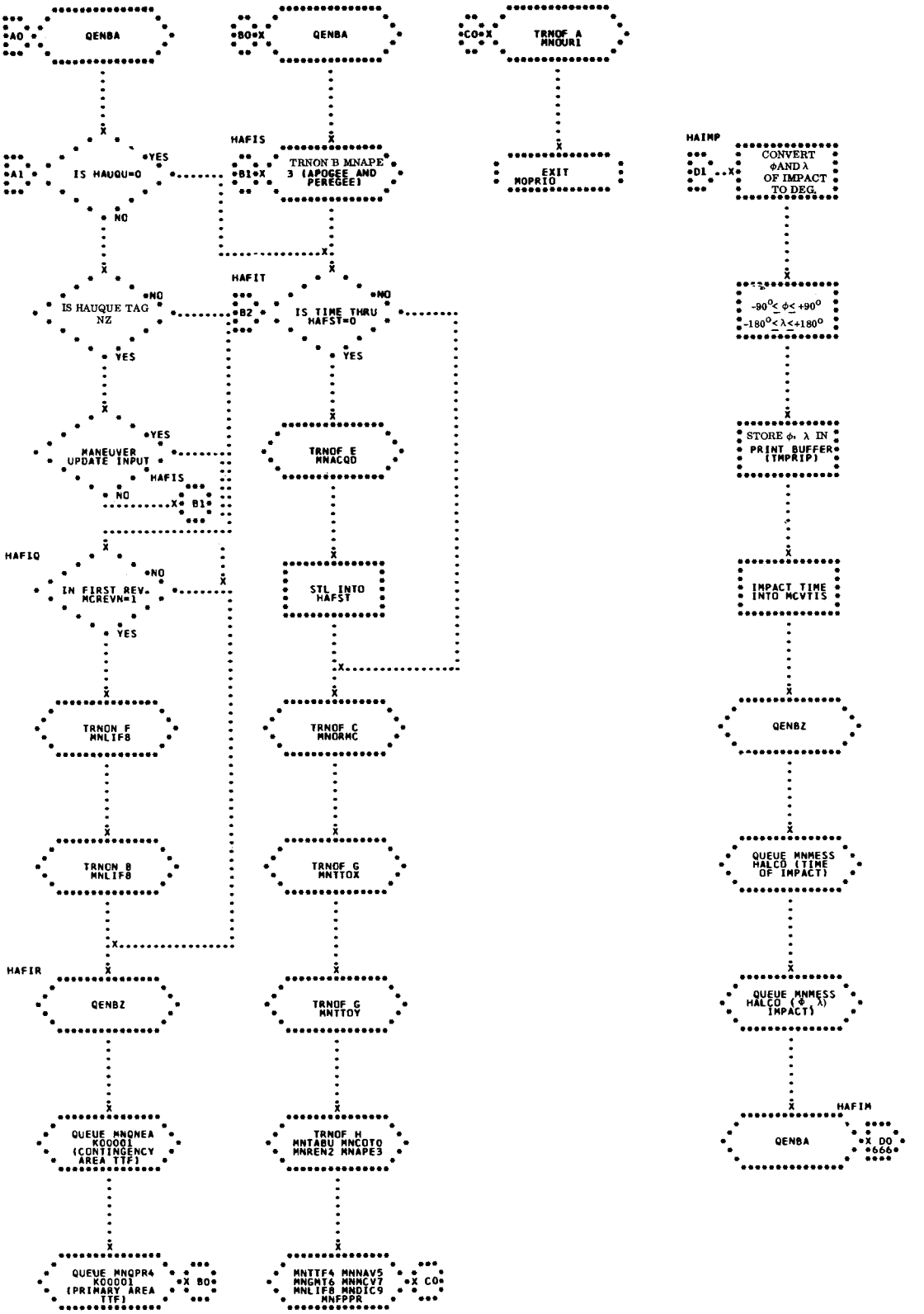
H) STORAGE REQUIRED - 720 LOCATIONS

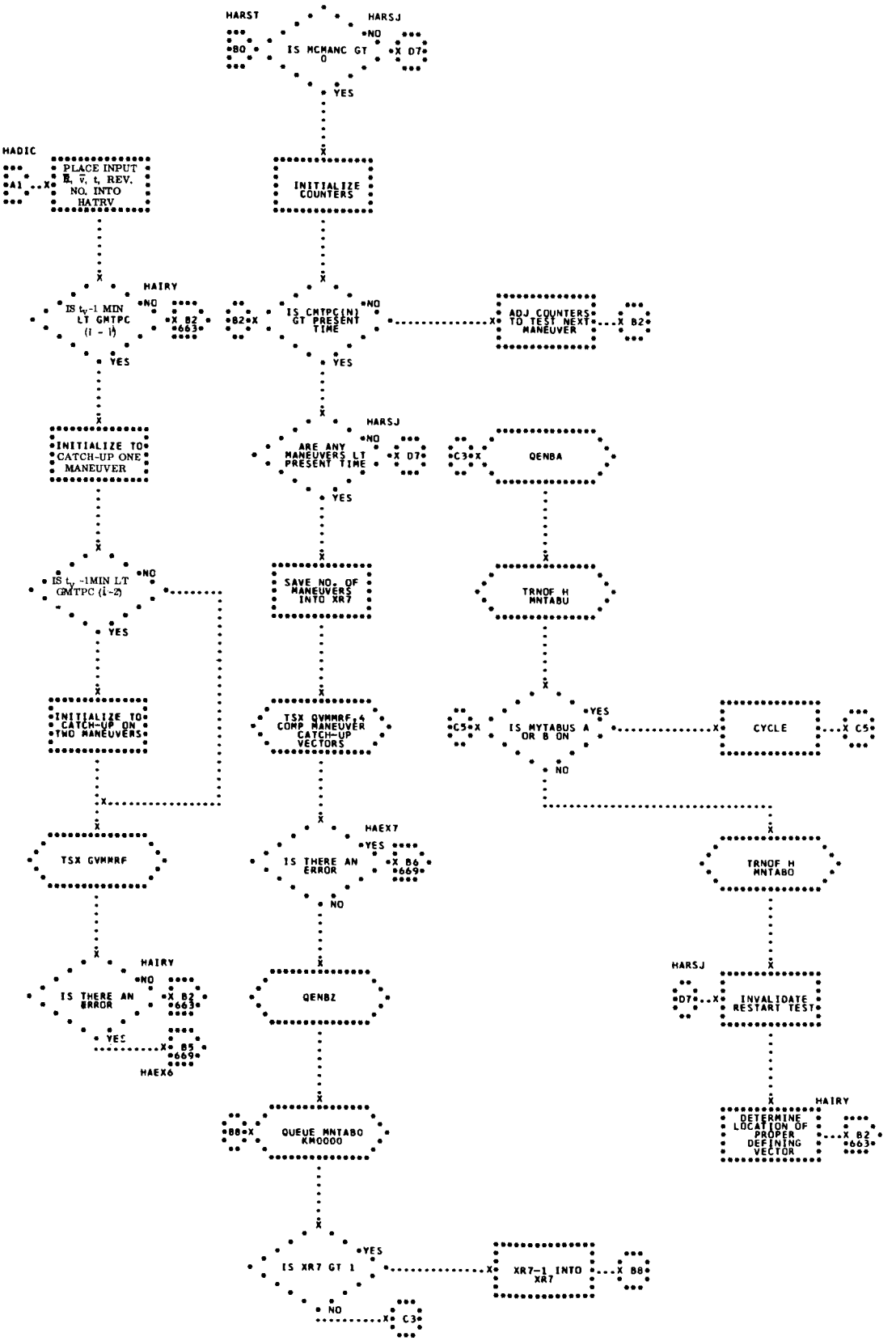

ENTER MYCUR1
TRNON A MNOUR1
GENBZ
UNCUE MNOUR1 HAUQU
GENBA
IN ORBIT PHASE MCPHSE=0
YES
NO
TRNOF A MNOUR1
EXIT MOPRIO
HAGD
TRNON H MNTABU MNCCTO MNREN2 MNAPE3
MNTTF4 MNNAV5 MNGMT6 MNMUV7 MNLIF8 MNDIC9 MNFPPR
HAAA
IS HAUQU=0
HAREN
IS HAUQU=+
NC
HAAA
HAAA
IS HAUQU=-
NO
HADIC
IS INPUT A D.C.VECTOR (HAUQU TAG NON 0)
HAIRY
PLACE INPUT t, r, v REV. NO. INTO TMREFL
HABA
HAKST
IS RESTART MCREST=0
HAB
IS MCMINS-30 LT tv
DETERMINE NO. OF 30 MIN. INTERVALS BETWEEN PRESENT TM.
AND THE TIME OF THE INPUT VECTOR INTO HACN1
SET-UP CALLING SEQUENCE FOR A3MSCP
TSX A3MSCP (λJ)
HANA
IS THERE AN ERROR
HAEX1
HANA
IS HACN1 GT 0
HACN1-1 INTO HACN1
SET-UP INTEGRATION FOR A 30 MIN. CATCH-UP
TSX GEMWGT (S/C WEIGHT)
(S/C WGT.)
QTSX MNOPN1 HAQUE
HAEX2
IS THIS AN INTEGRATION ERROR
HAEX3
IS THIS A VECTOR IMP.
SET-UP CALLING SEQUENCE FOR A3MSCP

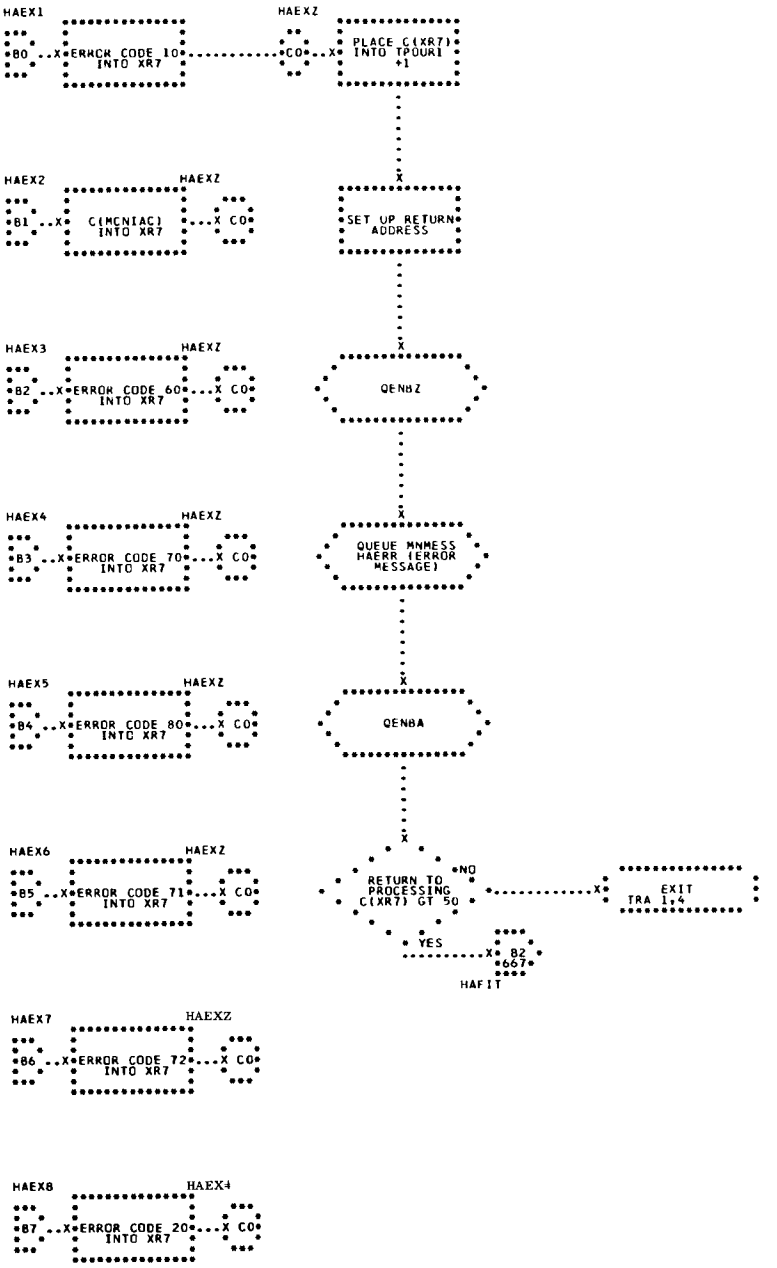

TSX A3MSCP
A0
(λ_{i+1})
HAEX1
IS THERE AN ERROR
YES
X B0
669
NO
λ_i LT λ_{CAPE}
NO
B2
YES
λ_{i+1} LT λ_{CAPE}
NO
YES
λ_{i+1} LT λ_i
NO
YES
INCREMENT REV. NO. BY 1
HANA
UPDATE LONG λ_{i+1} INTO λ_i
X D0
663
t_V INTO MCVTIS
QENBZ
QUEUE MNMESS K00023 ORBIT CATCH-UP VECTOR
QUEUE MNMESS HAM1 POSITION VECTOR
QUEUE MNMESS HAM2 VELOCITY VECTOR
QUEUE MNMESS HAM3 TIME OF VECTOR
QUEUE MNMESS HAM4 R.N. OF VECTOR
QENBA
X C0
C0
UPDATE VECTOR TABLES TMREFL INTO TOCDEF
INTO TMCOUV
INTO TMCRBD
INTO TMREST
MCTLST INTO TMREST
QENBZ
MCHESC INTO TMREST + 21
QUEUE MNWRRS K0001 (WRITE RECORD NO.1 ON RESTART TP
QENBA
DOES HAUQUE=0
YES
X A2
665
NO
COMPUTE VACUUM PER. HT. $h_{per} = a(1-e)$
INPUT IS PRESENT ORBIT-DEFINING VECTOR
HAEX8
IS THERE AN ERROR RETURN
YES
X B7
669
NO
X A0
665


A0
TSX GVMMRF
HAEX4
IS THERE AN ERROR RETURN
YES
X B3
669
NO
A2
TRNON C MNCRMC
TRNON G MNTTCX MNTTCY
INITIALIZE COUNTERS MCMANC INTC XR1
0 INTC XR3 0 INTC XR5 0 INTC XR6
INTEG. TABLE BEGINNING LOC INTO XR2
COMPUTE TABLE TIME LIMIT (t_{V1}+0 INTO HAMXT INTO HAMAX
SET j =1
A9
COMPUTE TABLE BEGIN TIME (t_{V1}-8) INTO HAIGB
X B0
B0
CHECK MANEUVER COUNT XR1 GT 0
NO
YES
IS t_{V1} GT GMTPC(I)
YES
NO
IS HAMAX GT GMTPS(I)
NO
YES
IS HAMAX GT GMTPC(I)
YES
NO
SET 2ND EXIT IND. = 1 INTO HAXIT
COMPUTE TABLE END TIME GMTPS(I) +8 INTO HAIND
HAY
B6
SET-UP INPUT TO INTEGRATOR (TMMCUR)
X D5
ISX GEMWGT (S/C WEIGHT)
HAY
COMPUTE TABLE END TIME (t_{V1}+8) INTO HAIND
X B6
COMPUTE TABLE END TIME HAMXT INTO HAIND
HAY
SET 1ST. EXIT IND. = 0 INTO HAXIT
X B6
D5
TSX HASAV (SAVE SR'S)
QTSX MNCPNI HAQUE
HAEX2
IS THIS AN INTEG. ERROR
YES
X B1
669
NO
HAIMP
IS THIS VECTOR IMP.
YES
X D1
667
NO
HAFIM
TIME TO EXIT HAXIT=-
YES
X D0
666
NO
X A0
666


TSX HARET (RESTORE XR*S)
AO
COMPUTE BASE ADDR.OF NEXT INTEG. TABLE INTO XR2
SET-UP CALLING SEQ. FOR U7INTP
TSX U7INTP (VECTOR AT EVEN MIN BEFORE S/C MANEUVER)
HAEX5
IS THIS AN ERROR RETURN
YES
X B4 *669*
NO
COMPUTE TABLE BEGIN TIME GMTPS(I) INTO HAIBG
COMPUTE TABLE END TIME GMTPC(I) INTO HAIND
SET UP INTEG. INPUT TABLE (THMCUR)
SET UP INPUT FOR THRUSTER (THRUSTS)
TSX GEMNGT (S/C WEIGHT)
X B0
B0
TSX HASAV (SAVE XR*S)
QTSX MNCPNI HAQUE
HAEX3
IS THIS A VECTOR IMP.
YES
X B2 *669*
NO
HAEX2
IS THIS AN INTEG. ERROR
YES
X B1 *669*
NO
HAEIM
IS IT TIME TO EXIT HAXIT NOT = 0
YES
X D0
NO
TSX HARET (RESTORE XRS)
COMPUTE BASE ADDR. OF NEXT INTEG. TABLE INTO XR2
READJUST COUNTERS XR4-1 INTO XR1, XR3-1 INTO XR3
XR5-5 INTO XR5, XR6-1 INTO XR6 SET I=I+1
X A9 *665*
ORBIT EPHEMERIS GENERATED
GENERATE REENTRY TABLE
HAEIM
D0
ORBIT EPHEMERIS AVAIL. STL INTO MCOEAV
QENBZ
QUEUE MNMESS K00093
QENBA
HAFIN
PRESENT TIME WITHIN 50 MIN. OF GMTRS
NO
X A1 *667*
YES
HAREN
D5
SET UP CALLING SEQ. FOR U7INTP
TSX U7INTP (VECTORS AT GMTRS)
HAEX5
IS THERE AN ERROR
YES
X B4 *669*
NO
QENBZ
QUEUE MNREN2 HAQRN
X A0 *667*


A0
QENBA
A1
IS HAUQU=0
YES
NO
IS HAUQUE TAG NZ
NO
YES
MANEUVER UPDATE INPUT
YES
HAFIS
NO
B1
HAFIQ
IN FIRST REV-MCREVN=1
NO
YES
TRNON F MNLIF8
TRNON B MNLIF8
HAFIR
QENBZ
QUEUE MNQNEA K00001 (CONTINGENCY AREA TTF)
QUEUE MNQPR4 K00001 (PRIMARY AREA TTF)
X B0
B0
QENBA
HAFIS
B1
TRNON B MNAPE 3 (APOGEE AND PEREGEE)
HAFIT
B2
IS TIME THRU HAFST=0
NO
YES
TRNOF E MNACQD
STL INTO HAFST
TRNOF C MNORMC
TRNOF G MNTTOX
TRNOF G MNTTOY
TRNOF H MNTARU MNCOTO MNREN2 MNAPE3
MNTTF4 MNNAV5 MNGMT6 MNNCV7 MNLIF8 MNDIC9 MNFPPR
X C0
C0
TRNOF A MNOURI
EXIT MOPRIO
HAIMP
D1
CONVERT ϕ AND λ OF IMPACT TO DEG.
$-90^{0} \leq \phi \leq +90^{0}$
$-180^{0} \leq \lambda \leq +180^{0}$
STORE ϕ, λ IN PRINT BUFFER (IMPRIP)
IMPACT TIME INTO MCVTIS
QENBZ
QUEUE MNMESS HALCO (TIME OF IMPACT)
QUEUE MNMESS HALCO (ϕ, λ) IMPACT)
HAFIM
QENBA
X D0
D0
666


HARST
HARSJ
B0
IS MCMANC GT 0
NO
D7
YES
HADIC
A1
PLACE INPUT R, V, t, REV. NO. INTO HATRV
INITIALIZE COUNTERS
HAIRY
IS t_v-1 MIN LT GMTPC (i - 1)
NO
B2 663
YES
B2
IS CMTPC(N) GT PRESENT TIME
NO
ADJ COUNTERS TO TEST NEXT MANEUVER
B2
YES
INITIALIZE TO CATCH-UP ONE MANEUVER
HARSJ
ARE ANY MANEUVERS LT PRESENT TIME
NO
D7
YES
C3
QENBA
IS t_v -1MIN LT GMTPC (i-2)
NO
YES
SAVE NO. OF MANEUVERS INTO XR7
TRNOF H MNTABU
INITIALIZE TO CATCH-UP ON TWO MANEUVERS
TSX GVMMRF,4 COMP MANEUVER CATCH-UP VECTORS
C5
IS MYTABUS A OR B ON
YES
NO
CYCLE
C5
TSX GVMMRF
HAEX7
IS THERE AN ERROR
YES
B6 669
NO
TRNOF H MNTABO
HAIRY
IS THERE AN ERROR
NO
B2 663
YES
B5 669
HAEX6
QENBZ
HARSJ
D7
INVALIDATE RESTART TEST
HAIRY
DETERMINE LOCATION OF PROPER DEFINING VECTOR
B2 663
B8
QUEUE MNTABO KM0000
IS XR7 GT 1
YES
NO
C3
XR7-1 INTO XR7
B8


HAEX1
B0
ERROR CODE 10 INTO XR7
HAEXZ
C0
PLACE C(XR7) INTO TPOUR1 +1
HAEX2
B1
C(MCNIAC) INTO XR7
HAEXZ
C0
SET UP RETURN ADDRESS
HAEX3
B2
ERROR CODE 60 INTO XR7
HAEXZ
C0
QENBZ
HAEX4
B3
ERROR CODE 70 INTO XR7
HAEXZ
C0
QUEUE MNMESS HAERR (ERROR MESSAGE)
HAEX5
B4
ERROR CODE 80 INTO XR7
HAEXZ
C0
QENBA
HAEX6
B5
ERROR CODE 71 INTO XR7
HAEXZ
C0
RETURN TO PROCESSING C(XR7) GT 50
NO
EXIT TRA 1,4
YES
B2 667
HAFIT
HAEX7
B6
ERROR CODE 72 INTO XR7
HAEXZ
C0
HAEX8
B7
ERROR CODE 20 INTO XR7
HAEX4
C0

9. GVMMRF - MANEUVER REFLECTING VECTORS

GVMMRF IS A SUBROUTINE OF MYOUR1. ITS FUNCTION IS TO GENERATE AND STORE VECTORS REFLECTING THE MANEUVERS CURRENTLY IN THE MANEUVER TABLE. (SEE FLOWCHART)

INPUT

A) CONDITIONS ON ENTRY -
AC CONTAINS THE NUMBER OF MANEUVERS TO BE PERFORMED.

B) SYSTEM MACROS
QTSX

C) CONSTANTS
K0C000 K00001 K00002
K00060 K00030 KCH233

D) TABLES -

TMREFL X
+1 Y
+2 Z
+3 $\dot{X}$
+4 $\dot{Y}$
+5 $\dot{Z}$
+6 T
+7 ANCHOR POINT REVOLUTION NUMBER

TMANUV GMTPS
+1 GMTPC
+2 A (ΔT), D(THRUSTER INDICATOR)
+3 PITCH
+4 YAW

TGMVNI A 17-LOCATION TABLE CORRESPONDING TO TMBBNI (INPUT TABLE FOR THE INTEGRATOR)

TRUSTS A 25-LOCATION TABLE USED AS INPUT TO THE THRUST PROGRAM

+0	ΔT_1	+13	ORIENT ANGLE$_2$
+1	ΔT_2	+14	ORIENT ANGLE$_3$
+2	ΔT_3	+15	ORIENT ANGLE$_4$
+3	ΔT_4	+16	CANT ANGLE$_1$
+4	GMTPS$_1$	+17	CANT ANGLE$_2$
+5	GMTPS$_2$	+18	CANT ANGLE$_3$
+6	GMTPS$_3$	+19	CANT ANGLE$_4$
+7	GMTPS$_4$	+20	PITCH
+8	THRUST LEVEL$_1$	+21	ROLL
+9	THRUST LEVEL$_2$	+22	YAW
+10	THRUST LEVEL$_3$	+23	THRUST IND.
+11	THRUST LEVEL$_4$	+24	MASS EXPLUSION RATE
+12	ORIENT ANGLE$_1$		

E) PROGRAMS

MYCUR1

MPCPNI

GEMWGT

F) COMMUNICATIONS CELLS -

MCVPER CONTAINS THE KEPLERIAN PERIGEE FOR THE PRESENT ORBIT DEFINING VECTOR

OUTPUT

A) TABLES -

TMREFL r_1, v_1, t_1 THE MANEUVER REFLECTING VECTORS

$APRN_1$ GENERATED ARE STORED IN TMREFL IMMEDIATELY FOLLOWING THE ORIGINAL ORBIT DEFINING VECTOR (r_1, v_1, t_1, $APRN_1$)

r_n, v_n, t_n

$APRN_n$

B) EXIT -
CONTROL IS RETURNED TO MYOUR1

METHOD

THE CURRENT ORBIT DEFINING VECTOR IS SELECTED FROM TMREFL AND IS USED TO INITIALIZE THE INTEGRATOR. INTEGRATION THEN PROCEEDS THROUGH EACH MANEUVER IN THE TABLE STORING IN TMREFL, FOR EACH MANEUVER, A MANEUVER REFLECTING VECTOR AT THE NEXT EVEN MINUTE AFTER THE BURN.

USAGE

A) ENTRY - FROM MYOUR1
B) EXIT - TO MYOUR1
C) CALLING SEQUENCE -

```
        TSX   GVMMRF
+1      PZE   (ADDRESS OF INPUT TABLE),,(ADDRESS OF OUT-
              PUT TABLE)
+2      PZE   (ADDRESS OF MANEUVER TABLE),,(WEIGHT IN-
              DICATOR)
+3      TRA   ERROR RETURN
+4            NORMAL RETURN
```

α+1 = M*E-STORE ONLY LAST VECTOR
= PZE-STORE ALL VECTORS

α+2 (WEIGHT INDICATOR = 0 - USE GEMWGT TO FIND CAPSULE WEIGHT OF THE WEIGHT TABLE, $\neq$ 0 -

ADDRESS OF WEIGHT TABLE

D) STORAGE REQUIRED -

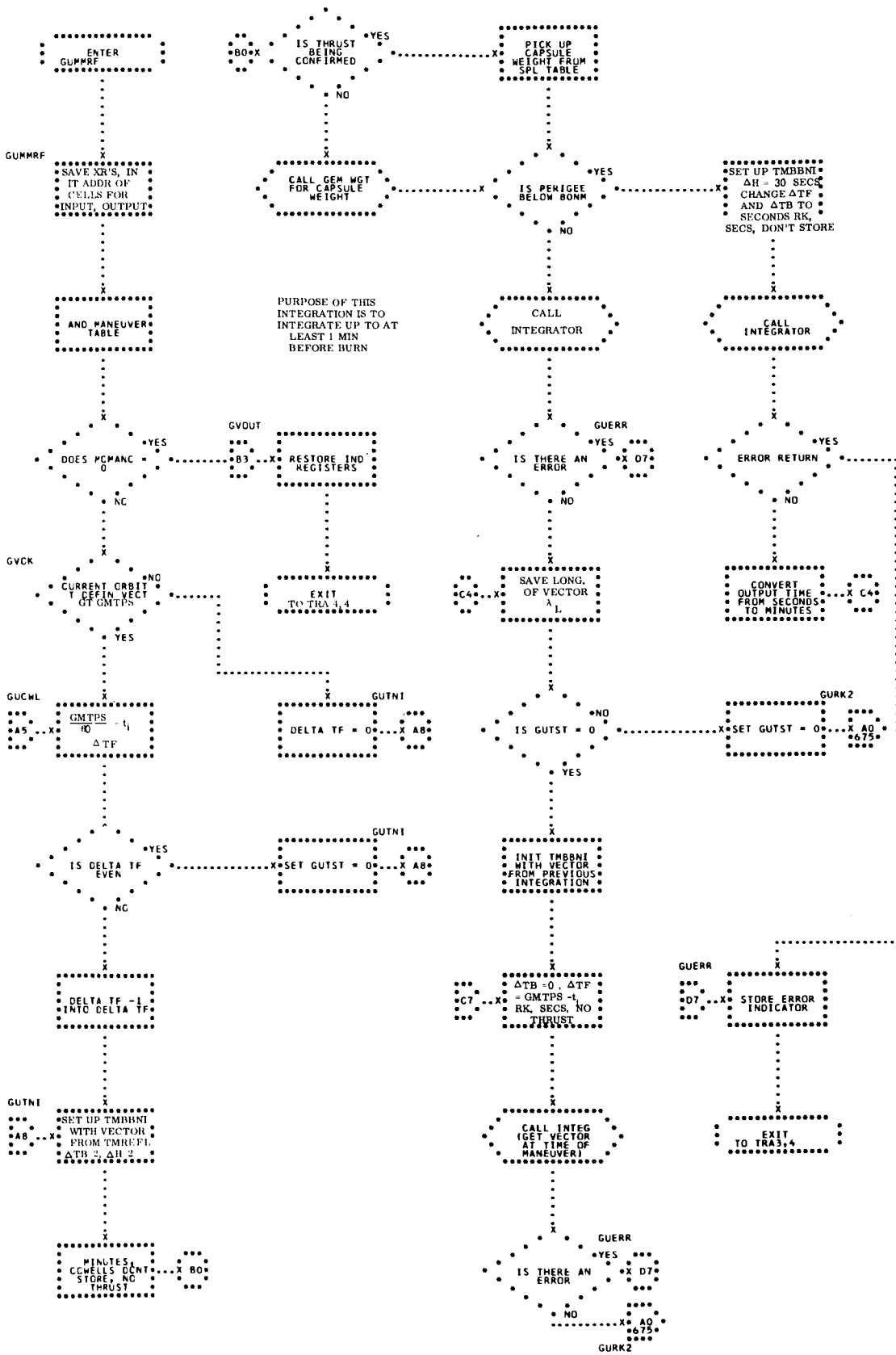
505 LOCATIONS USED FOR THE FOLLOWING

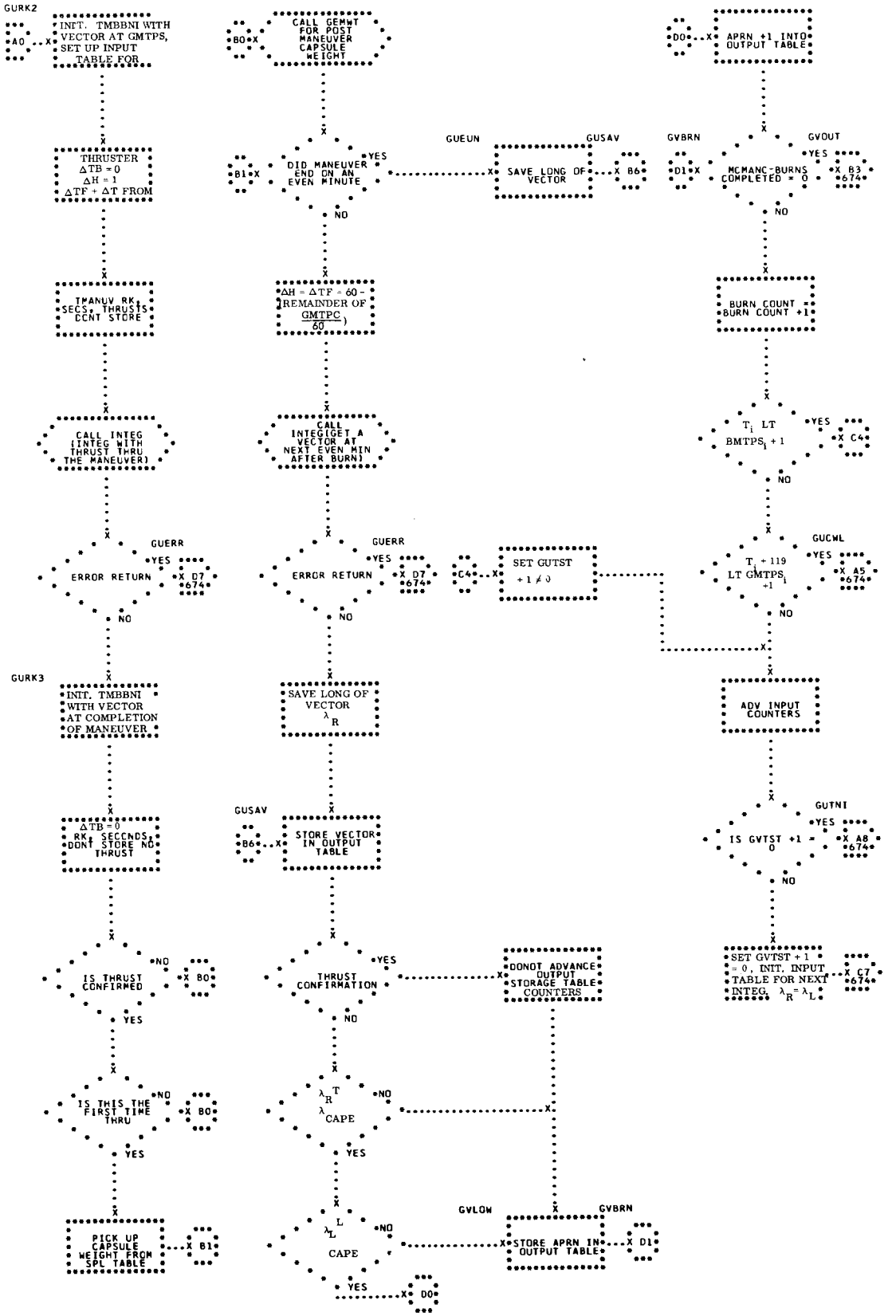
1) 440 INSTRUCTIONS

2) 9 INTERNAL CONSTANTS

3) 55 STORAGE CELLS

E) ERROR CODES - NONE


ENTER GUMMRF
GUMMRF
SAVE XR'S, IN IT ADDR OF CELLS FOR INPUT, OUTPUT
AND MANEUVER TABLE
DOES MCMANC = 0
YES
NO
GVOUT
B3
RESTORE IND REGISTERS
GVCK
CURRENT ORBIT T DEFIN VECT GT GMTPS
NO
YES
EXIT TO TRA 4,4
GUCWL
A5
GMTPS / MO - t_i
ΔTF
DELTA TF = 0
GUTNI
A8
IS DELTA TF EVEN
SET GUTST = 0
DELTA TF -1 INTO DELTA TF
SET UP TMBBNI WITH VECTOR FROM TMREF1, ΔTB 2, ΔH 2
MINUTES, CGWELLS DONT STORE, NO THRUST
B0
IS THRUST BEING CONFIRMED
CALL GEM WGT FOR CAPSULE WEIGHT
PICK UP CAPSULE WEIGHT FROM SPL TABLE
IS PERIGEE BELOW 80NM
SET UP TMBBNI ΔH = 30 SECS, CHANGE ΔTF AND ΔTB TO SECONDS RK, SECS, DON'T STORE
PURPOSE OF THIS INTEGRATION IS TO INTEGRATE UP TO AT LEAST 1 MIN BEFORE BURN
CALL INTEGRATOR
GUERR
IS THERE AN ERROR
D7
SAVE LONG. OF VECTOR λ_L
C4
IS GUTST = 0
GURK2
SET GUTST = 0
A0
.675
INIT TMBBNI WITH VECTOR FROM PREVIOUS INTEGRATION
ΔTB =0 , ΔTF = GMTPS -t_i RK, SECS, NO THRUST
C7
CALL INTEG (GET VECTOR AT TIME OF MANEUVER)
ERROR RETURN
CONVERT OUTPUT TIME FROM SECONDS TO MINUTES
STORE ERROR INDICATOR
EXIT TO TRA3,4


GURK2
INIT. TMBBNI WITH VECTOR AT GMTPS, SET UP INPUT TABLE FOR
A0
THRUSTER ΔTB = 0 ΔH = 1 ΔTF + ΔT FROM
TMANUV RK, SECS, THRUSTS CONT STORE
CALL INTEG (INTEG WITH THRUST THRU THE MANEUVER)
GUERR
ERROR RETURN
YES
X D7 *674*
NO
GURK3
INIT. TMBBNI WITH VECTOR AT COMPLETION OF MANEUVER
ΔTB = 0 RK, SECONDS, DONT STORE NO THRUST
IS THRUST CONFIRMED
NO
X B0
YES
IS THIS THE FIRST TIME THRU
NO
X B0
YES
PICK UP CAPSULE WEIGHT FROM SPL TABLE
X B1
CALL GEMWT FOR POST MANEUVER CAPSULE WEIGHT
B0
DID MANEUVER END ON AN EVEN MINUTE
YES
B1
NO
ΔH = ΔTF = 60 - (REMAINDER OF GMTPC / 60)
CALL INTEGIGET A VECTOR AT NEXT EVEN MIN AFTER BURN)
GUERR
ERROR RETURN
YES
X D7 *674*
NO
SAVE LONG OF VECTOR λR
GUSAV
B6
STORE VECTOR IN OUTPUT TABLE
THRUST CONFIRMATION
YES
NO
λR T / λ CAPE
NO
YES
λL L / λ CAPE
NO
YES
X D0
GUEUN
SAVE LONG OF VECTOR
GUSAV
X B6
DONOT ADVANCE OUTPUT STORAGE TABLE COUNTERS
C4
SET GUTST +1 ≠ 0
GVLOW
GVBRN
STORE APRN IN OUTPUT TABLE
X D1
D0
APRN +1 INTO OUTPUT TABLE
GVBRN
D1
GVOUT
MCMANC-BURNS COMPLETED = 0
YES
X B3 *674*
NO
BURN COUNT = BURN COUNT +1
Ti LT BMTPSi + 1
YES
X C4
NO
GUCWL
T + 119 LT GMTPSi+1
YES
X A5 *674*
NO
ADV INPUT COUNTERS
GUTNI
IS GVTST +1 = 0
YES
X A8 *674*
NO
SET GVTST + 1 = 0, INIT. INPUT TABLE FOR NEXT INTEG. λR = λL
X C7 *674*

10. MYREN2 - REENTRY CONTROL PROGRAM

MYREN2 CONTROLS THE GENERATION OF A REENTRY TRAJECTORY AND BACKUP GUIDANCE PARAMETERS AND SETS UP REENTRY DATA FOR PLOTBOARD AND DIGITAL DISPLAYS. (SEE FLOWCHART.)

INPUT

A) PROCESSOR -

NOCPNI	NUMERICAL INTEGRATOR

B) SUBROUTINES -

RUCIVI	ROTATES ACCUMULATED ACCELEROMETER OUTPUTS (IVI'S) FROM PLATFORM COORDINATE SYSTEM INTO INERTIAL COORDINATE SYSTEM.
RUPDAT	UPDATES POST-RETROFIRE VECTOR BASED ON IVI READINGS.
R5RARF	COMPUTES BACKUP GUIDANCE PARAMETERS AND MIN LIFT AND MAX LIFT IMPACT POINTS.
RVECHT	COMPUTES A VECTOR AT A SPECIFIED HEIGHT.
A3MSCP	COMPUTES SUB-CAPSULE POSITION λ, ϕ, H
RHBRVG	COMPUTES RELATIVE VELOCITY (V) AND FLIGHT PATH ANGLE (Υ).

C) MACROS -

TRNON	CAUSES SETTING OF MONITOR CONTROL BIT IN TMPRIO.
TRNOF	CAUSES RESETTING OF MONITOR CONTROL BIT IN TMPRIO.
QENBZ	DISABLES ALL INTERRUPTS
QENBA	ENABLES ALL INTERRUPTS

UNQUE	SETS UP CALLING SEQUENCE TO MOUNQU WHICH SUPPLIES A QUEUE WORD TO THE ROUTINE UNQUEUEING.
QUEUE	SETS UP THE CALLING SEQUENCE TO MOQUEU WHICH PLACES ENTRIES INTO QUEUE TABLES.
QTSX	SIMULATES A REALTIME TRAP. PROVIDES FOR INTERRUPTING A PROGRAM AT THE PLACE THE QTSX IS INSERTED - USED TO QUEUE THE INTEGRATOR.

D) TABLES -

TMDLVS	A TEN-LOCATION TABLE CONTAINING AN ORBIT VECTOR CORRESPONDING IN TIME TO THE BURNOUT VECTOR. THE VELOCITY VECTOR IS USED IN COMPUTING $\Delta\bar{V}$'s FOR THE POST-RETROFIRE UPDATE.
TMGTRS	A 5-LOCATION TABLE CONTAINING CONTROLS FOR THE ORBIT CASE REENTRY (SEE TMGTRS FORMAT TABLE IN THIS SECTION).
TMIMPP	GMTLC MINUTES IN ADR. + SECONDS IN DEC.
+1	λ_{IP} - LONGITUDE AT 10K FEET
+2	ϕ_{IP} - LATITUDE AT 10K FEET
TMLMPT	TABLE OF PRIMARY AREAS WITH MNNUPA NUMBER OF PAIRS λ, ϕ IN RADIANS.
TMOFNI	A 10-LOCATION TABLE USED TO OUTPUT SINGLE VECTORS FROM THE INTEGRATOR (SEE TMOFNI FORMAT TABLE IN THIS SECTION).
TMPLON	A MNNUPA-LOCATION TABLE CONTAINING CODES FOR PRIMARY AREAS.

TMRETO	AN 11-LOCATION TABLE CONTAINING RETROFIRE CONDITIONS (SEE TMRETO FORMAT TABLE IN THIS SECTION).
TMRNSB	A ONE CELL TABLE CONTAINING THE MAXIMUM NUMBER OF VECTORS ALLOWED FOR BACKWARD INTEGRATION DURING REENTRY.
TMRORF	OUTPUT TABLE FROM R5RARF CONTAINS GUIDANCE PARAMETERS AND FOOTPRINT IMPACT POINTS (SEE R5RARF UNDER OUTPUT).
TMRTCO	A 7-LOCATION TABLE CONTAINING THE CONTROLS FOR THE REENTRY CASE I.E., WHEN THE FUNCTION CODE IS THREE (SEE TMRTCO FORMAT TABLE IN THIS SECTION).
TNREEP	AN 8-LOCATION TABLE CONTAINING BLACKOUT TIMES (TNREEP+2, +3) AND TIME AT 325K (TNREEP+6) WHICH ARE GENERATED BY THE INTEGRATOR (NOCPNI).
TNVECT	A 28-LOCATION TABLE GENERATED BY THE INTEGRATOR AND CONTAINING FOUR VECTORS - A PAIR EACH WHICH BRACKET 400K AND 350K FEET RESPECTIVELY.
TOSCCH	SPACECRAFT CHARACTERISTICS TABLE MYREN2 USES VALUES TO INPUT TO THE THRUST PROGRAM (ROCKNR) VIA TRETRO (SEE OUTPUT TABLES)
TSETCT+9 (KPERST)	CONTAINS THE STORE INTERVAL USED BY DC IN GENERATING PERTURBED TABLES.
TSETCT+10 (KRDELT)	CONTAINS, IN FIXED POINT SECONDS, THE INTEGRATION STEP SIZE USED BY DC.
TRLIMT	A ONE-LOCATION TABLE CONTAINING THE LIMIT ON MISS DISTANCE GENERATED BY

R5RARF (COMPUTES GUIDANCE PARAMETERS).

E) CONSTANTS -

FIXED POINT

K00000	K00010	K77777
K00001	K00024	KTGMSK
K00002	K00060	KMNMK1
K00004	K00090	KCH233
K00006	K00180	

FLOATING POINT

K0.1DG	K022.0	K0360.
K000.5	KRTDEG	K325.3
K001.0	K060.0	K035.4
K010.0	K0180.	K040.4

F) COMMUNCIATION CELLS -

MCTOFS TIME OF FIRST RETRO ROCKET FIRED IN FIXED POINT SECONDS.

MCTABT ABORT PHASE INDICATOR. WHEN ZERO ABORT PHASE OTHERWISE ORBIT PHASE.

MCGRGP BANK ANGLE FROM ABORT PHASES (HIGH OR LOW)
IF 0, MAX LIFT TRAJECTORY REQUESTED
IF 60 , 60 BANK " "
IF 999, ZERO LIFT " "

MCGTLO GMT OF LIFTOFF IN FIXED POINT SECONDS

MCOWGT ORBIT SPACECRAFT WEIGHT IN FLOATING POINT POUNDS.

MCPWGT PRE-RETRO WEIGHT IN FLOATING POINT POUNDS.

MCRWGT — REENTRY WEIGHT IN FLOATING POINT POUNDS.

G) PARAMETERS -

1) MONITOR CONTROL BITS

A,C,D,E,H,K

2) ROUTINE NUMBERS

MNACQD, MNAPE3, MNCOTO, MNCPNI, MNDIC9
MNDIFC, MNDIFK, MNDIFM, MNFPPR, MNGMT6
MNLED1, MNLED2, MNLIF8, MNMESS, MNMUV7
MNNAV5, MNORMC, MNOUR1, MNREN2, MNTTF4

3) TABLE SIZES

MNBBNI — INPUT TO INTEGRATOR.

MNNUPA — NUMBER OF PRIMARY AREAS.

MNORTS — SIZE OF TNINT2

MNRETS — AMOUNT OF STORAGE FOR REENTRY TRAJECTORY FOR GT-4 WAS THE SUM OF FOUR TABLES (TNINT2, TMCAPE TNDIAC, TNINTH).

OUTPUT

A) TABLES -

TOBLK1 — BEGINNING OF BLACKOUT TIME IN FIXED POINT SECONDS USED FOR ACQUISITION.

TOBLK2 — END OF BLACKOUT TIME IN FIXED POINT SECONDS.

TDRANV — CONTAINS BURNOUT VECTOR AFTER RETROFIRE AND ONLY UNTIL SUCCESSFUL DC IN REENTRY. THE FORMAT IS $\bar{r}$, $\bar{v}$, t IN GEMINI UNITS AND FIXED POINT SECONDS.

TFESAB — TIME OF BURNOUT IN FIXED POINT SECONDS USED BY DC.

THBOUT	A 10-LOCATION TABLE USED TO BUFFER DATA TO THE ON-LINE MESSAGE ROUTINE (MYMESS).
TM400K	SAME AS TNVECT (SEE INPUT).
TMBNOT	A 7-LOCATION TABLE CONTAINS THE BURN-OUT VECTOR IN THE FORMAT $\bar{r}$, $\bar{v}$, t WHERE $\bar{r}$ AND $\bar{v}$ ARE IN GEMINI UNITS AND t IN FIXED POINT SECONDS.
TMITNI	A MNBBNI-LOCATION TABLE CONTAINS ALL INPUT TO THE INTEGRATOR NOCPNI (SEE FORMAT IN MPCPNI WRITEUP).
TMORMC	CONTAINS HIGH SPEED OUTPUT. (SEE FORMAT IN O5ORMC WRITEUP).
TMRCNP	A 5-LOCATION TABLE CONTAINS DATA WHICH REFLECTS THE TYPE OF REENTRY TRAJECTORY GENERATED (SEE TMRCNP TABLE FORMAT IN THIS WRITEUP).
TMSSGE	CONTAINS REENTRY DATA TO BE PRINTED ON-LINE (SEE TMSSGE TABLE FORMAT IN THIS SECTION.)
TNINT2	REENTRY REFERENCE TRAJECTORY IN THE FORMAT, TIME UNIT INDICATOR, STORE INTERVAL, t_1, t_n, $\bar{r}_1$, $\bar{v}_1$, $\bar{r}_2$, $\bar{v}_2$, ..., $\bar{r}_n$, $\bar{v}_n$ TIME IN FIXED SECONDS $\bar{r}$'S + $\bar{v}$'S IN GEMINI UNITS.
TNREEP	AN 8-LOCATION TABLE OF WHICH MYREN2 OUTPUTS v_{350K}, γ_{350K}, λ_{400K}, ϕ_{400K}, t_{400K} (TNREEP, +1, +4, +5, AND +7 RESPECTIVELY).

TRETRO CONTAINS INPUT TO ROCKNR PROGRAM (THRUST) VIA INPUT TO THE INTEGRATOR. MOST OF THE DATA IS FROM TOSCCH (SEE FORMAT IN ROCKNR WRITEUP).

B) COMMUNICATION CELLS -

MCESAB CONTAINS TIME OF FIRST WHOLE SECOND AFTER BURNOUT.

MCBNOT TIME OF BURNOUT WITH MINUTES IN DECREMENT AND SECONDS IN ADDRESS.

METHOD

WHEN MYREN2 RECEIVES CONTROL ITS A BIT IS TURNED ON, ALL USERS OF THE INTEGRATOR (NOCPNI) ARE H-SUPPRESSED - EXCEPT OF COURSE MYREN2, THE E BIT FOR MPLED1 IS TURNED OFF, AND MYREN2 IS UNQUEUED. THE E BIT IS ON ONLY WHEN A LOW ABORT REENTRY IS REQUESTED.

MYREN2 IS DIVIDED INTO FIVE SECTIONS, WHICH ARE DISCUSSED IN THE FOLLOWING ORDER -

1) DETERMINATION OF THE BURNOUT VECTOR
2) COMPUTATION OF BACKUP GUIDANCE PARAMETERS
3) GENERATION OF THE NOMINAL TRAJECTORY
4) GENERATION OF A MANUAL TRAJECTORY
5) OUTPUT.

1. DETERMINATION OF THE BURNOUT VECTOR -

IF THE TIME OF THE INPUT VECTOR IS PRIOR TO THE RETROFIRE TIME, THEN THE VECTOR IS INTEGRATED UP TO THAT TIME. THE RETROFIRE TIME IS TAKEN FROM TMGTRS FOR THE ORBIT CASE AND FROM MCTDFS FOR THE RETROFIRE CASE. WHEN NO RETROS FIRE, MCTOFS WILL NOT BE A RETROFIRE TIME AS SUCH, BUT AN INTEGRATION WILL BE PERFORMED IF THE VECTOR TIME IS PRIOR TO MCTOFS. REENTRY FROM LOW ABORT IS ALWAYS A NO-RETROFIRE CASE.

THE TIME-TO-FIRE VECTOR IS INTEGRATED AT A ONE SECOND STEP SIZE THROUGH THE RETRO BURN PERIOD WHICH IS DETERMINED FROM THE RETROFIRE CONDITIONS (TMRETO).

A) NOMINAL RETROFIRE - ALL FOUR RETROS FIRE SERIALLY IN THE ORDER 1,3,2,4 WITH NOMINAL BURN TIME, PITCH, ROLL, AND YAW (SEE TOSCCH FOR NOMINAL VALUES). NOMINAL RETROFIRE IS ALWAYS ASSUMED FOR THE ORBIT CASE. CONTROL IS THEN PASSED TO C).

B) NON-NOMINAL RETROFIRE - THE RETROFIRE BURN DURATION IS COMPUTED BY

$$\Delta T = t_{LAST} - t_{FIRST} + 6 \text{ SECONDS WHERE}$$

t_{LAST} IS THE LAST RETROFIRE TIME

t_{FIRST} IS THE EARLIEST RETROFIRE TIME

IF THE D(TMRETO+4)=2, THE BURNOUT VECTOR IS INTEGRATED WITH THE NON-NOMINAL PITCH, ROLL, YAW AND RETROFIRE TIMES. IF THE D(TMRETO+4)=3, THE BURNOUT VECTOR IS INTEGRATED WITH NON-NOMINAL RETROFIRE TIMES BUT WITH NOMINAL PITCH, ROLL, AND YAW. IN ADDITION THE BURNOUT VECTOR IS UPDATED BASED ON IVI READINGS (THEY APPEAR IN THE SAME CELLS AS THE NON-NOMINAL PITCH, ROLL,

AND YAW) AS FOLLOWS -

1) COMPUTE A $\Delta\overline{V}$ AT THE TIME OF BURNOUT
$\Delta\overline{V} = \overline{V}_{BO} - \overline{V}_{ORBIT}$

2) USING RUCIVI SUBROUTINE ROTATE IVI'S FROM PLATFORM TO INERTIAL COORDINATES.

3) USING RUPDAT UPDATE THE BURNOUT VECTOR BASED ON THE INERTIAL IVI'S AND $\Delta\overline{V}$.

C) THE BURNOUT VECTOR (WHICH INCLUDES THE VECTOR WHEN NO RETROS FIRE) IS STORED IN TMBNOT AND ALSO IN TDRANV FOR THE RETROFIRE CASE, AND ALSO PRINTED ON-LINE.

D) IF THE MANUAL OVERRIDE WAS SET, CONTROL IS PASSED TO THE TRAJECTORY GENERATION FOR MANUAL REQUEST SECTION, OTHERWISE TO THE GUIDANCE PARAMETER COMPUTATION SECTION.

2. COMPUTATION OF BACKUP GUIDANCE PARAMETERS (GMTRB, B1, GMTIR, B).

R5RARF IS USED TO COMPUTE THE BACKUP GUIDANCE PARAMETERS. R5RARF IS BYPASSED FOR THREE REASONS -

A) WHEN A CONTINGENCY TRAJECTORY IS REQUESTED WITH A ZERO TARGET

B) FOR A MANUAL OVERRIDE OF GUIDANCE PARAMETERS (MANUAL INSERTION NO. 20)

C) WHEN IN THE DC ITERATION LOOP

R5RARF IS GIVEN A BURNOUT VECTOR OR LATER (TMITNI) AND A TARGET. IF THE NORMAL EXIT IS TAKEN, THEN CONTROL IS TRANSFERRED TO THE TRAJECTORY GENERATION SECTION. IF THE ERROR EXIT IS TAKEN, R5RARF LEAVES A NUMBER IN THE AC WHICH INDICATES THE REASON FOR THE EXIT AND MYREN2

TAKES THE FOLLOWING STEPS -

A) IF ZERO LIFT IP COULD NOT BE GENERATED, (AC=1), A MESSAGE IS PRINTED AND AN EXIT TAKEN.

B) IF EITHER MAX LIFT OR +60 BANK IP COULD NOT BE INTEGRATED (AC=2 OR 27), A MESSAGE IS PRINTED AND A ZERO LEFT TRAJECTORY IS STORED.

C) IF THE ANCHOR VECTOR IS BELOW 325K FEET (AC=36), THEN THE TRAJECTORY STORED WILL BE BASED ON CONTROLS (TMRCNP) USED FOR THE PREVIOUS TRAJECTORY GENERATION SECTION.

D) IF THE PRIMARY AREAS ARE BEING SCANNED FOR A NEW TARGET PROCEED TO PARAGRAPH L).

E) PRINT ON-LINE THE R5RARF ERROR NUMBERS, λ_T, ϕ_T, $\Delta\phi, \Delta\lambda$, BANK ANGLE (B1), λ_{IP} AND ϕ_{IP}. THE IP IS THE CLOSEST POINT TO THE TARGET.

F) IF GMTRB = GMTIR = 0, SET THE CONTROL TO GENERATE ZERO LIFT TRAJECTORY AND PROCEED TO H).

G) IF THE R5RARF ERROR NUMBER = 23 (BEHIND THE FOOTPRINT), THEN SET CONTROLS TO GENERATE ZERO LIFT TRAJECTORY AND PROCEED TO H).

H) COMPUTE MISS DISTANCE.
$$\text{m. d.} = \sqrt{\Delta\phi^2 + \Delta\lambda^2}$$

I) IF THE ORIGINAL TARGET IS A MANUAL TARGET (MANUAL OVERRIDE CODE=1), A MESSAGE IS PRINTED ON-LINE AND CONTROLS ARE SET FOR A CONTINGENCY TRAJECTORY IF TRAJECTORY CODE=5, OTHERWISE THE CONTROLS ARE SET TO CAUSE IMPACT AT A POINT NEAREST THE TARGET. CONTROL IS TRANSFERRED TO TRAJECTORY GENERATION SECTION.

J) IF THE TARGET IS A GMTRS TARGET, A MESSAGE IS PRINTED ON-LINE AND CONTROLS ARE SET TO GENERATE CONTINGENCY TRAJECTORY IF THE CODE=5, OTHERWISE EXIT.

K) IF THE MISS DISTANCE IS EQUAL TO OR LESS THAN 50 NAUTICAL MILES (TRLIMT), TARGET SELECTION IS BYPASSED AND CONTROL TRANSFERRED TO THE TRAJECTORY GENERATION SECTION.
IF THE MISS DISTANCE IS GREATER THAN 50 NAUTICAL MILES PROCEED TO M).

L) IF THE PRIMARY AREA BEING TESTED FAILS TO CONVERGE IN THE PARTIALS OF R5RARF (ERROR=3,4, OR 5), THEN THE SOLUTION OBTAINED WILL BE ACCEPTED AND CONTROL TRANSFERRED TO THE TRAJECTORY GENERATION SECTION.

M) IF ALL PRIMARY AREAS HAVE BEEN SCANNED THEN A MESSAGE IS PRINTED ON-LINE AND CONTROL IS TRANSFERRED TO THE TRAJECTORY GENERATION SECTION.
IF THERE ARE MORE AREAS TO BE SCANNED THEN THE NEXT ONE IS SELECTED FROM TMLMPT AND ITS ASSOCIATED RECOVERY CODE FROM TMPLON. CONTROL IS TRANSFERRED TO R5RARF (BEGINNING OF THIS SECTION) WITH THE SELECTED PRIMARY AREA AS A TARGET.

3. NOMINAL TRAJECTORY GENERATION

A) IF THE PRIMARY AREAS WERE SCANNED FOR A TARGET RECOMMENDATION, THEN THE TARGET AND ITS ASSOCIATED BACKUP GUIDANCE PARAMETERS ARE PRINTED ON-LINE AND CONTROL IS PASSED TO C).

B) IF A NEW TARGET WAS NOT RECOMMENDED, THEN AN INTERNAL GUIDANCE PARAMETER TABLE IS SET UP FROM THE OUTPUT OF R5RARF (TMRORF) FOR THE COMPILED TARGET.

C) THE HIGH SPEED OUTPUT AND REQUISITION PROGRAMS ARE C AND E, SUPPRESSED RESPECTIVELY, WHILE THE REENTRY TRAJECTORY IS BEING GENERATED.

D) INPUT (TMITNI) IS SET UP TO INTEGRATOR (NOCPNI), WHICH GENERATES THE TRAJECTORY REQUESTED AND STORES IT IN TNINT2. IF AN ERROR OCCURS IN TRYING TO INTEGRATE THE TRAJECTORY, A MESSAGE IS PRINTED ON-LINE AND A ZERO LIFT TRAJECTORY IS STORED AS THE NOMINAL.

E) FOR ALL TRAJECTORIES SUCCESSFULLY GENERATED, TMRCNP IS SET TO REFLECT THE TYPE GENERATED AND A MESSAGE IS PRINTED ON-LINE STATING THE TYPE OF TRAJECTORY. CONTROL IS PASSED TO THE OUTPUT SECTION.

4. TRAJECTORY GENERATION FOR MANUAL REQUESTS.

A) FIRST, ZERO LIFT IMPACT POINT IS INTEGRATED. IF AN ERROR OCCURS IN THE INTEGRATION, AN ERROR MESSAGE IS PRINTED AND CONTROL IS TRANSFERRED TO MOPRIO LEAVING THE DIFFERENTIAL CORRECTION PROGRAM SUPPRESSED. IF ZERO LIFT IP IS SUCCESSFULLY GENERATED CONTROL PASSES TO B).

B) NEXT, THE MAX LIFT IMPACT POINT IS INTEGRATED AND IF A FAILURE OCCURS AN ON-LINE MESSAGE IS PRINTED AND A ZERO LIFT TRAJECTORY IS STORED AS NOMINAL. THE ABOVE TWO IMPACT POINTS ARE THE FOOTPRINT BOUNDARIES.

C) IF IN THE DC LOOP, CONTROL IS PASSED TO THE NOMINAL TRAJECTORY GENERATION SECTION, PARAGRAPH C), OTHERWISE PROCEED TO D).

D) THE HIGH SPEED OUTPUT AND ACQUISITION PROGRAMS ARE SUPPRESSED DURING THE GENERATION OF THE MANUAL TRAJECTORY. IF AN ERROR OCCURS DURING THE INTEGRATION, A MESSAGE IS PRINTED ON-LINE GIVING THE GUIDANCE PARAMETERS AND TRAJECTORY CODE USED.

E) FOR A SUCCESSFUL TRAJECTORY GENERATION AN ON-LINE MESSAGE IS PRINTED AND TMRCNP IS SET UP TO REFLECT THE TYPE OF TRAJECTORY. CONTROL IS PASSED TO THE OUTPUT SECTION.

5. OUTPUT

IN THIS SECTION TWO OUTPUT TABLES AER SET UP - THE DIGITAL DISPLAYS (TMORMC) AND THE ON-LINE MESSAGE TABLE (TMSSGE).

A) USING VECTORS FROM THE INTEGRATOR (TM400K) AND THE SUBROUTINE RVECHT, A VECTOR AT 400,000 IS COMPUTED. USING THIS VECTOR AND THE SUBROUTINE A3MSGP, THE TIME AT 400K IS COMPUTED.

B) A VECTOR IS COMPUTED AT 350,000 FEET USING THE SAME METHOD AS IN A). USING THIS VECTOR AND RHBRVG, THE RELATIVE VELOCITY AND FLIGHT PATH ANGLE ARE COMPUTED.

C) NEXT THE FUNCTION CODE IS TESTED FOR THE FOLLOWING -

1) IF F.C. IS GREATER THAN 3, BYPASS SETTING THE DIGITAL DISPLAY (TMORMC), THAT IS IF EXECUTING IN THE DC LOOP.

2) IF EXECUTING THE ORBIT OR RETROFIRE CASES, SET MCESAB WITH THE BURNOUT TIME (TMBNOT+6).

3) IF EXECUTING IN THE ORBIT PHASE, SET THE TIME TO REVERSE BANK AND ITS ASSOCIATED BANK ANGLE, YAW AND DELTA V TO ZERO IN THE DIGITAL OUTPUT TABLE (TMORMC). THE SIGN OF GMTRB IS SET NEGATIVE SO THE PROPULSION PANEL LIGHTS UP RED.

4) IF EXECUTING EITHER THE RETROFIRE CASE OR A MANUAL REQUEST THE ENTIRE DIGITAL DISPLAY TABLE IS SET. ALL INTERRUPTS ARE DISABLED DURING THE TRANSFER INTO TMORMC.

D) TMSSGE, THE ON-LINE MESSAGE TABLE, IS SET FROM TABLES GENERATED BY THE INTEGRATOR (TMIMPP AND TNREEP) AND FROM THE GUIDANCE PARAMETERS AND FOOTPRINT IPS TABLES, COMPUTED BY R5RARF OR MANUALLY INSERTED. IN ADDITION, MYREN2 STORES THE BLACKOUT TIMES IN TOBLK1 AND TOBLK2, RESPECTIVELY, MYREN2 IS K SUPPRESSED UNTIL THE MESSAGE BLOCK IS PRINTED. MYMESS TURNS OFF THE K-BIT.

E) THE EXIT CONSISTS OF UNSUPPRESSING THE PREFIX TO DC AND UNSUPPRESSING ALL USERS OF THE INTEGRATOR, THE HIGH SPEED OUTPUT (O5ORMC), THE ACQUISITION PROGRAM, AND THE A OF MYREN2. THE DC PROGRAMS REMAIN SUPPRESSED UNTIL A REENTRY TRAJECTORY IS GENERATED. CONTROL IS PASSED TO MOPRIO.

USAGE

MYREN2 IS A QUEUED PROGRAM. THE QUEUE WORD CONTAINS THE LOCATION OF A VECTOR ($\bar{R}$,$\bar{V}$,t) IN THE ADDRESS AND A FUNCTION CODE IN THE DECREMENT. IN ADDITION TO THE QUEUE WORD, A CONTROL TABLE MUST BE SET DEPENDING ON THE VALUE OF THE FUNCTION CODE.

A FUNCTION CODE OF 1 INDICATES A REENTRY TRAJECTORY IS REQUESTED DURING THE ORBIT PHASE AND TMGTRS MUST BE SET. A FUNCTION CODE OF 2 INDICATES A REENTRY TRAJECTORY IS REQUESTED BASED ON THE RETRO-FIRE MESSAGE AND TMRETO MUST BE SET. A FUNCTION CODE OF 3 INDICATES A MANUAL TRAJECTORY IS REQUESTED AND TMRTCO MUST BE SET (TMRTCO IS INITIALIZED DURING PROCESSING OF THE RETRO-FIRE MESSAGE). A FUNCTION CODE OF 4 INDICATES THAT THE DC ITERATION IS IN PROCESS AND THE PREVIOUS SET OF CONTROLS WILL BE USED.

STORAGE REQUIREMENTS -

A CORE - 118 LOCATIONS

B CORE - 1223 LOCATIONS

TIMING -

1) ORBIT CASE - 14.85 MILLISECONDS
2) RETROFIRE NOMINAL - 15.02 MILLISECONDS
3) RETROFIRE NON-NOMINAL WITH IVI'S - 18.41 MILLISECONDS
4) MANUAL TRAJECTORY USING R5RARF - 13.869 MILLISECONDS
5) MANUAL TRAJECTORY WITH R5RARF OVERRIDE - 15.223 MILLISECONDS

THE ABOVE ESTIMATES EXCLUDE R5RARF AND INTEGRATOR (NOCPNI) TIMES.

ERROR CONDITIONS -

ALL ERRORS PRODUCE ON-LINE MESSAGES. MYREN2 MESSAGE NUMBERS ARE 300-328 INCLUSIVE.

TABLE 10-1

TMGTRS - GMTRS - RETRO-FIRE SETTING IN FX. PT. SECONDS

+1 TRAJ. CODE = 1 ON BOARD GUIDANCE
= 3 GMTRB AND BANK ANGLE
= 4 GMTIR AND BANK ANGLE
= 5 CONTINGENCY (FIXED B.A.)

+2 TARGET LONG. IN FL. PT. RADIANS

+3 TARGET LAT. IN FL. PT. RADIANS

+4 RECOVERY CODE - REV. NO IN THE DEC.
AREA IN ADDRESS.

TABLE 10-2

TMOFNI - INDICATOR TO SPECIFY TIME UNIT

0 - TIME IN MINUTES

1 - TIME IN SECONDS

+1	h	- INTEG. STEP SIZE
+2	t_1	- TIME OF FIRST VECTOR
+3	t_n	- TIME OF LAST VECTOR (TIME OF THE VECTOR)
+4	X	- X COMPONENT OF POSITION IN G.L.U.
+5	Y	- Y COMPONENT OF POSITION IN G.L.U.
+6	Z	- Z COMPONENT OF POSITION IN G.L.U.
+7	$\dot{X}$	- X COMPONENT OF VELOCITY IN G.V.U.
+8	$\dot{Y}$	- Y COMPONENT OF VELOCITY IN G.V.U.
+9	$\dot{Z}$	- Z COMPONENT OF VELOCITY IN G.V.U.

THE OUTPUT TABLE USED FOR TIME-TO-FIRE, BURNOUT, ZERO LIFT IMPACT, AND MAX LIFT IMPACT VECTORS.

TABLE 10-3

TMRETO - $GMTRF_1$ - TIME RETRO 1 FIRES IN FX. PT. SECONDS

+1 $GMTRF_2$ - TIME RETRO 2 FIRES IN FX. PT. SECONDS

+2 $GMTRF_3$ - TIME RETRO 3 FIRES IN FX. PT. SECONDS

+3 $GMTRF_4$ - TIME RETRO 4 FIRES IN FX. PT. SECONDS

+4 MODE,, TYPE OF RETROFIRE INDICATOR

A) MODE (TRAJ. CODE, SEE TMGTRS) IN THE ADDRESS

B) TYPE OF RETRO FIRE INDICATOR IN THE DECREMENT

IF = 1, NOMINAL RETRO FIRE

IF = 2, NON-NOMINAL RETRO FIRE

IF = 3, $\Delta\overline{V}$(NOT USED)

+5 PITCH - IN FLOATING POINT RADIANS

+6 ROLL - IN FLOATING POINT RADIANS

+7 YAW - IN FLOATING POINT RADIANS

+8 λ_T - TARGET LONGITUDE IN FLOATING POINT RADIANS

+9 ϕ_T - TARGET LATITUDE IN FLOATING POINT RADIANS

+10 --- - USED AS TEMPORARY STORAGE

+11 B_C CONTINGENCY BANK ANGLE IN FLOATING POINT DEGREES. A BANK ANGLE GREATER THAN 360° WITH A TRAJECTORY CODE OF 5 INDICATES REENTRY FROM LOW ABORT AND A ZERO LIFT TRAJECTORY WILL BE GENERATED.

TABLE 10-4

TMRCNP CONTAINS DATA WHICH REFLECTS THE TYPE OF REENTRY TRAJECTORY GENERATED. THE TABLE IS USED BY THE DIFFERENTIAL CORRECTION PROGRAM (DODIFC).

TMRCNP	T. C. = 1	T. C. = 3	T. C. = 4	T. C. = 5
+1	λ_T	GMTRB	GMTIR	0
+2	ϕ_T	B_1	B	B_c
+3	t_{325K}			

λ_T - LONG. OF TARGET IN FLOATING PT. RADIANS
ϕ_T - LAT. OF TARGET IN FLOATING PT. RADIANS
GMTRB - TIME TO REVERSE IN FL. PT. SECONDS
B_1 - GMTRB BANK ANGLE IN FL. PT. DEGREES
GMTIR - TIME TO INITIATE ROLL IN FL. PT. SECONDS
B - GMTIR BANK ANGLE IN FL. PT. DEGREES
B_c - CONTINGENCY BANK IN FL. PT. DEGREES
t_{325K} - TIME AT 325,000 FEET IN FL. PT. SECONDS - TIME WHICH SINGLE STATION D.C. IS IN EFFECT.

TABLE 10-5

TMRTCO - MANUAL OVERRIDE CODE IN THE ADDRESS FIELD

IF = 0, NO MANUAL OVERRIDES

IF = 1, MANUAL TARGET

IF = 2, MANUAL GUIDANCE

+1 TRAJ. CODE IN THE ADDRESS (SEE TMGTRS)

+2 λ_T - LONGITUDE OF TARGET IN FL. PT. RADIANS

+3 ϕ_T - LATITUDE OF TARGET IN FL. PT. RADIANS

+4 B_c - CONTINGENCY BANK IN FL. PT. DEGREES

+5 GMTRB/GMTIR - TIME TO REVERSE BANK/TIME TO INITIATE ROLL IN FX. PT. SECONDS.

+6 B_1/B - BANK ANGLE ASSOCIATED WITH GMT SPECIFIED IN FL. PT. DEGREES.

TABLE 10-6

TMSSGE - RECOVERY AREA CODE (CODE,,LRN)

+1 V - VELOCITY AT 400,000 FT IN FL. PT. FEET/SEC.

+2 Υ - FLIGHT PATH ANGLE AT 400K FT. FL. PT. DEGREES

+3 GMTBO1 - BEGINNING OF BLACKOUT IN FX. PT. SECS.

+4 GMTBO2 - ENDING OF BLACKOUT IN FX. PT. SECS.

+5 GMTLC - GMT AT 10K FEET IN FX. PT. SECONDS

+6 ϕ_{IP} - LAT. OF IMPACT

+7 λ_{IP} - LONG. OF IMPACT

+8 ϕ_{IP} (ZL) - LAT. OF ZERO LIFT IP IN FL. PT. DEGREES

+9 λ_{IP} (ZL) - LONG OF ZERO LIFT IP IN FL. PT. DEGREES

+10 ϕ_{IP} (ML) - LAT. OF MAX. LIFT IP IN FL. PT. DEGREES

+11 λ_{IP} (ML) - LONG OF MAX. LIFT IP IN FL. PT. DEGREES

+12 GMTRF - TIME OF FIRST RETRO-FIRING FX. PT. SECS.

+13 GETRC - GROUND ELAPSED TIME SINCE LIFTOFF FX. SECS

+14 GMTRB - TIME TO REVERSE BANK IN FL. SECONDS

+15 BI - GMTRB BANK ANGLE IN FL. PT. DEGREES

+16 GMTIR - TIME TO INITIATE ROLL IN FX. PT. SECONDS

+17 B - GMTIR BANK ANGLE IN FL. PT. DEGREES

+18 ϕ_{400K} - LAT. AT 400K FEET IN FL. PT. DEGREES

+19 λ_{400K} - LONG. AT 400K FEET IN FL. PT. DEGREES

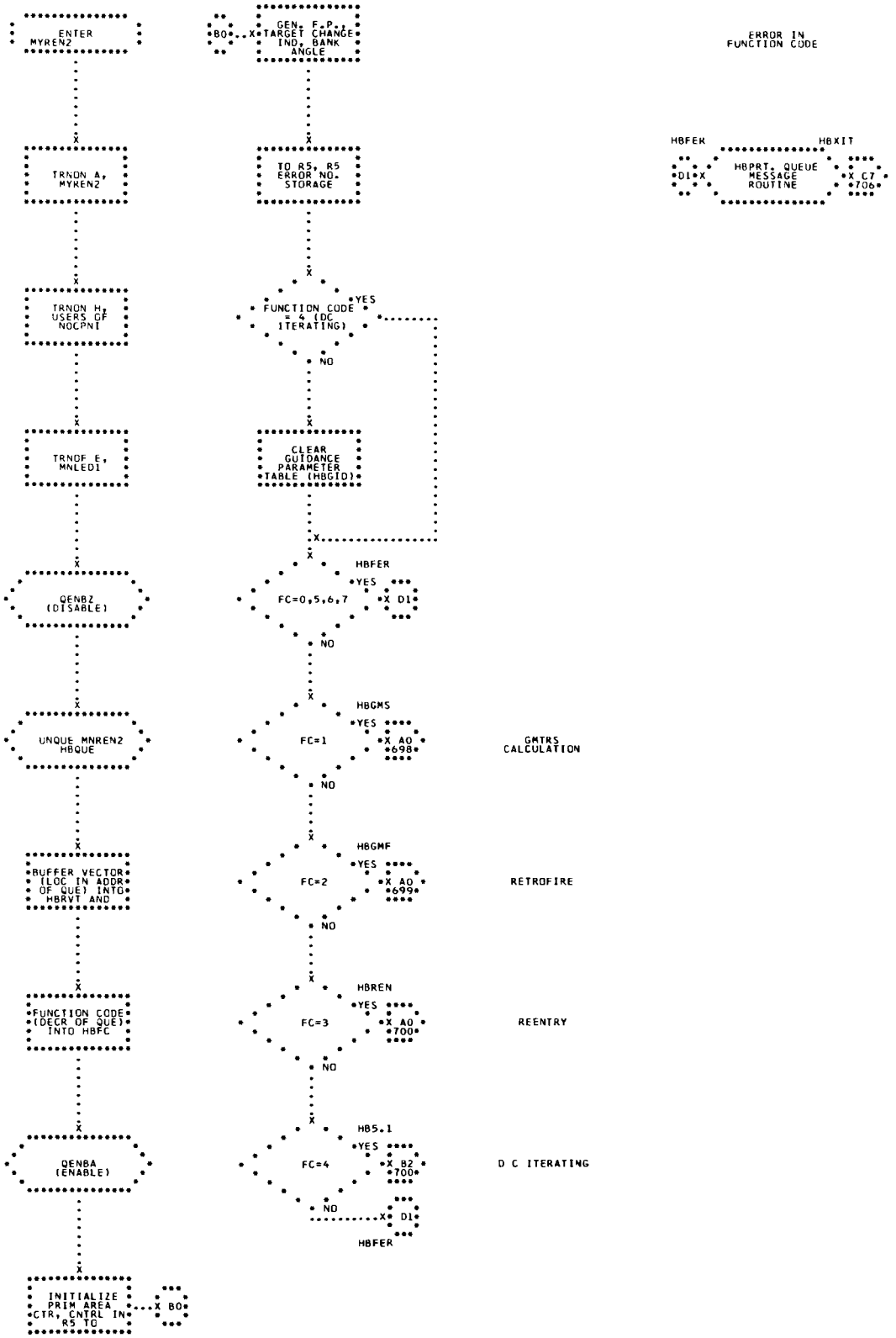
+20 $\Delta\phi$ - CROSSRANGE ERROR IN NAUTICAL MILES

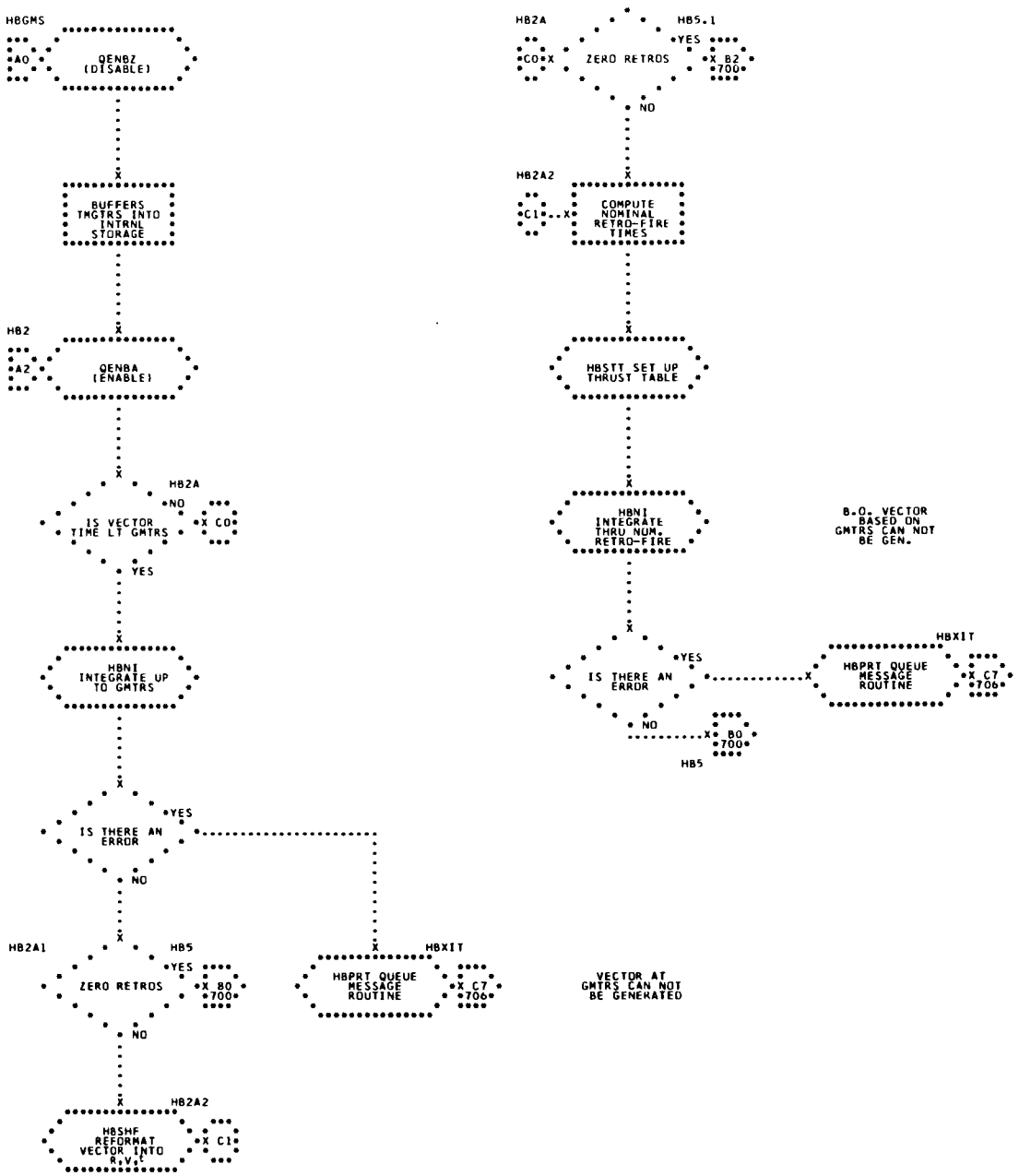
+21 $\Delta\lambda$ - DOWNRANGE ERROR IN NAUTICAL MILES

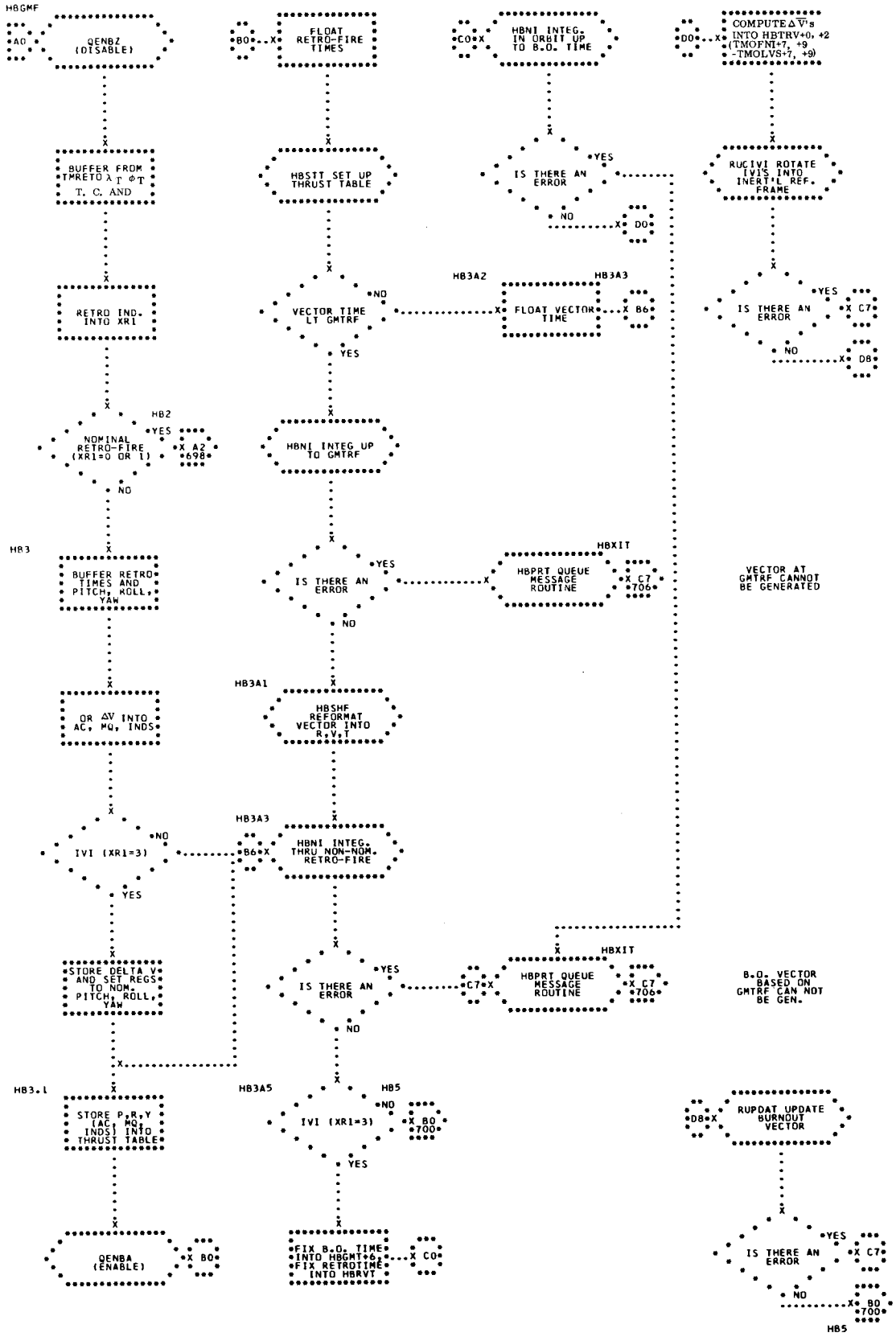
+22 ϕ_T - LAT. OF TARGET IN FL. PT. DEGREES

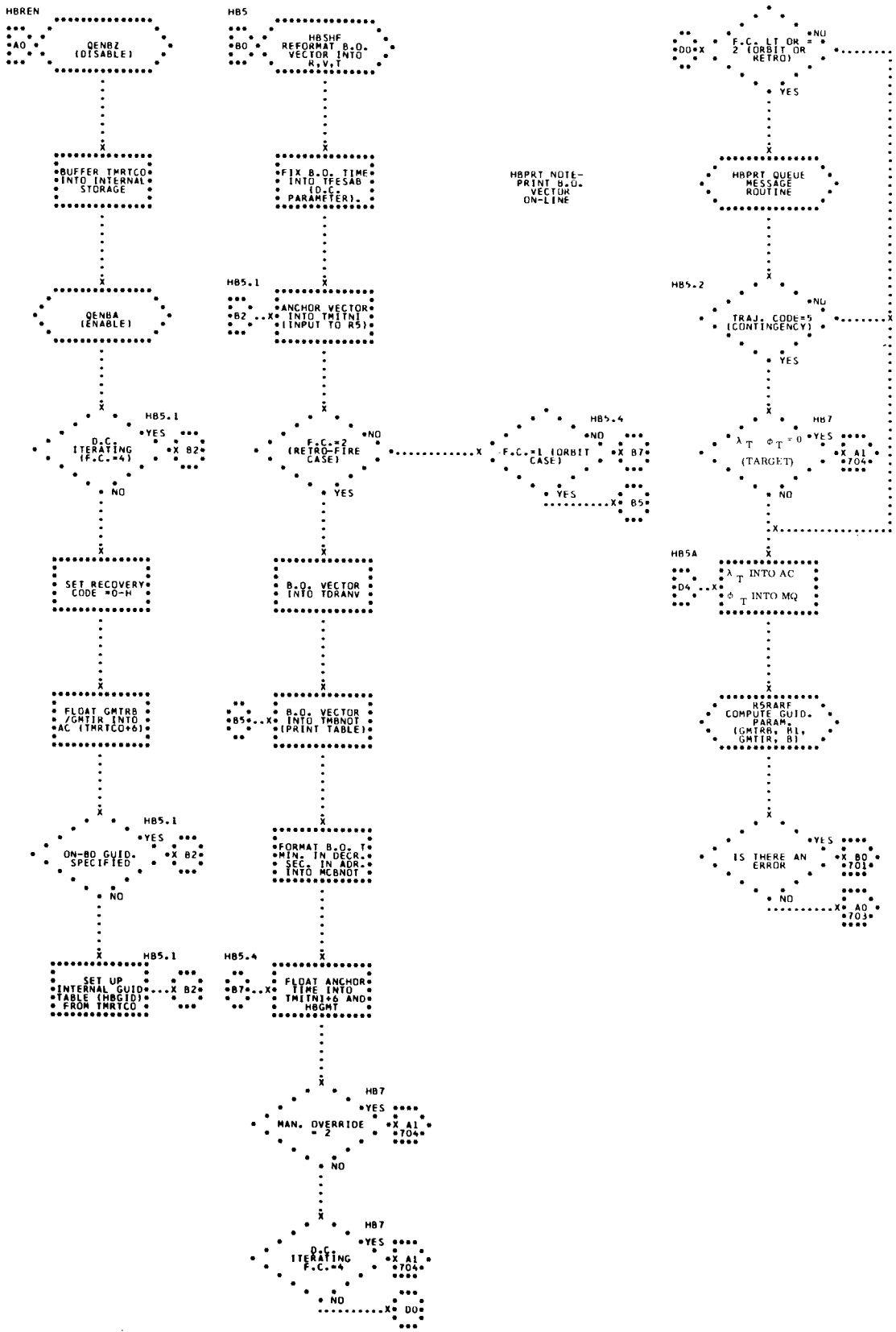
+23 λ_T - LONG. OF TARGET IN FL. PT. DEGREES

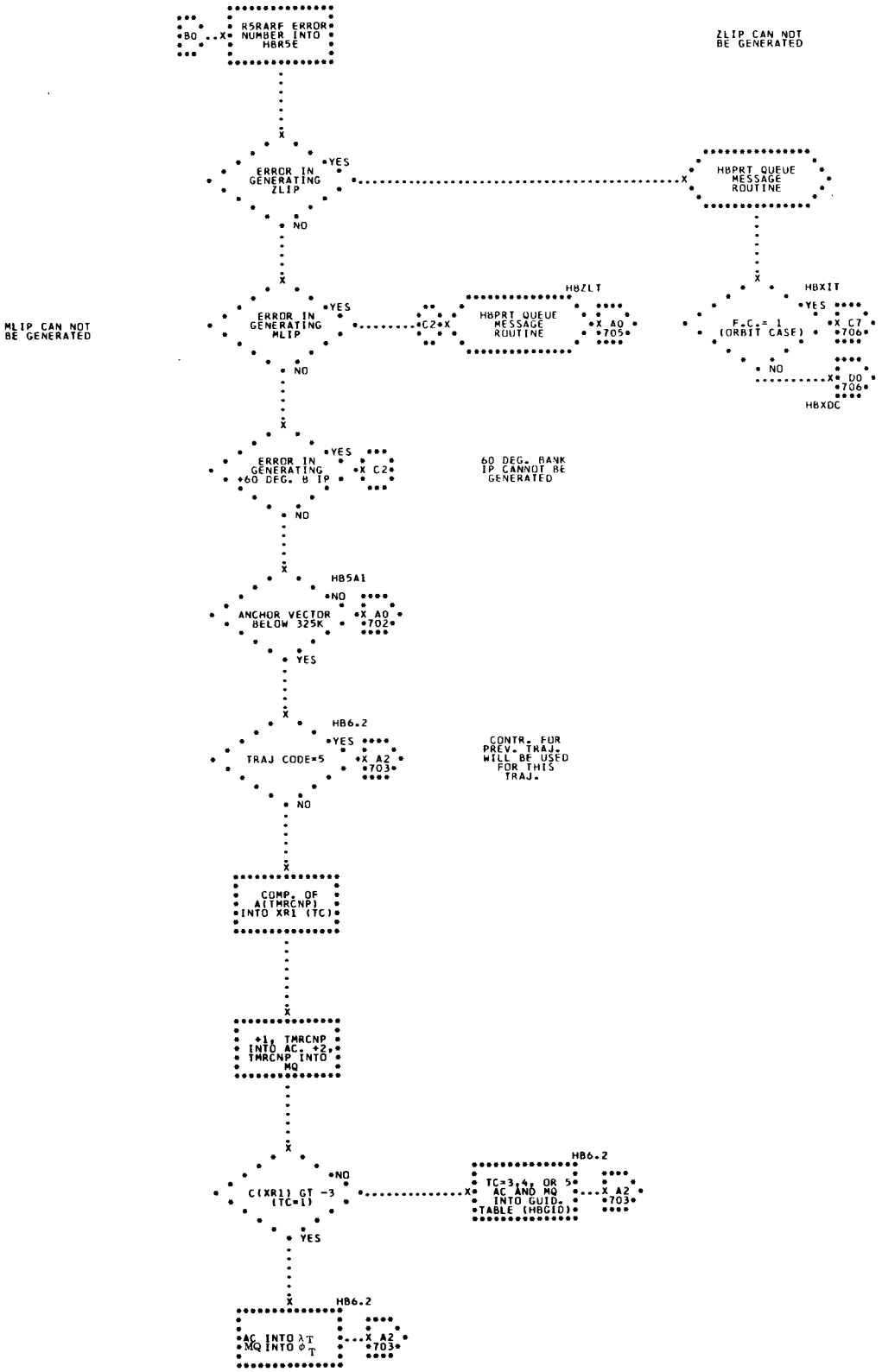
+24 CONTROL WORD TO MESSAGE ROUTINE USED TO SUPPRESS QUEUEING ROUTINE.
ADDRESS - ROUTINE NUMBER (MNREN2)
DECREMENT - SUPPRESSION BIT TO BE USED (K) FOR MYREN2.

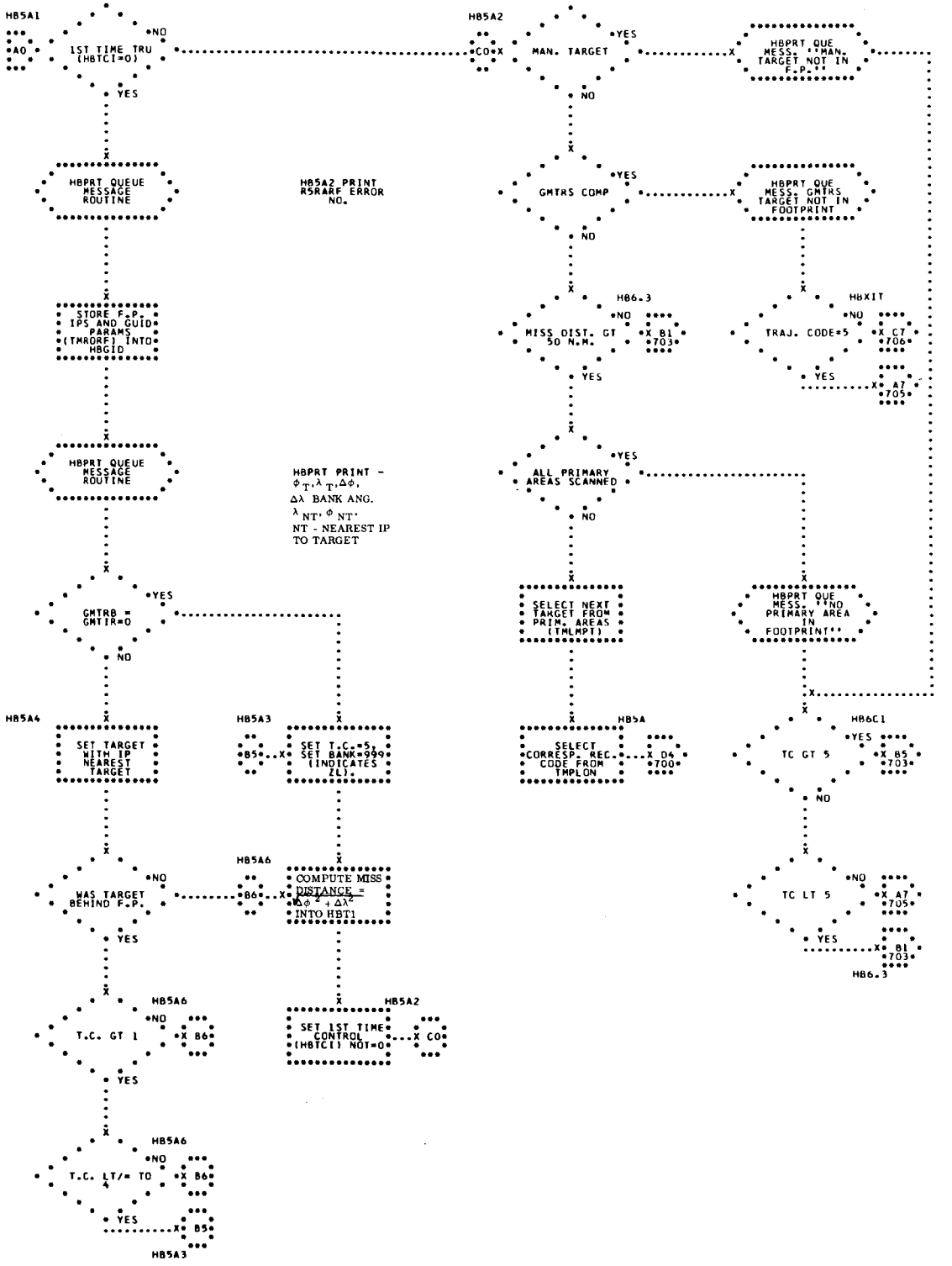

ENTER MYREN2
TRNON A, MYREN2
TRNON H, USERS OF NOCPNI
TRNOF E, MNLED1
QENBZ (DISABLE)
UNQUE MNREN2 HBQUE
BUFFER VECTOR (LOC IN ADDR OF QUE) INTO HBRVT AND
FUNCTION CODE (DECR OF QUE) INTO HBFC
QENBA (ENABLE)
INITIALIZE PRIM AREA CTR, CNTRL IN R5 TO
B0
GEN. F.P., TARGET CHANGE IND., BANK ANGLE
TO R5, R5 ERROR NO. STORAGE
FUNCTION CODE = 4 (DC ITERATING)
YES
NO
CLEAR GUIDANCE PARAMETER TABLE (HBGID)
HBFER
FC=0,5,6,7
YES
D1
NO
HBGMS
FC=1
YES
A0 698
NO
GMTRS CALCULATION
HBGMF
FC=2
YES
A0 699
NO
RETROFIRE
HBREN
FC=3
YES
A0 700
NO
REENTRY
HB5.1
FC=4
YES
B2 700
NO
D1
HBFER
D C ITERATING
ERROR IN FUNCTION CODE
HBFER
D1
HBPRT, QUEUE MESSAGE ROUTINE
HBXIT
C7 706

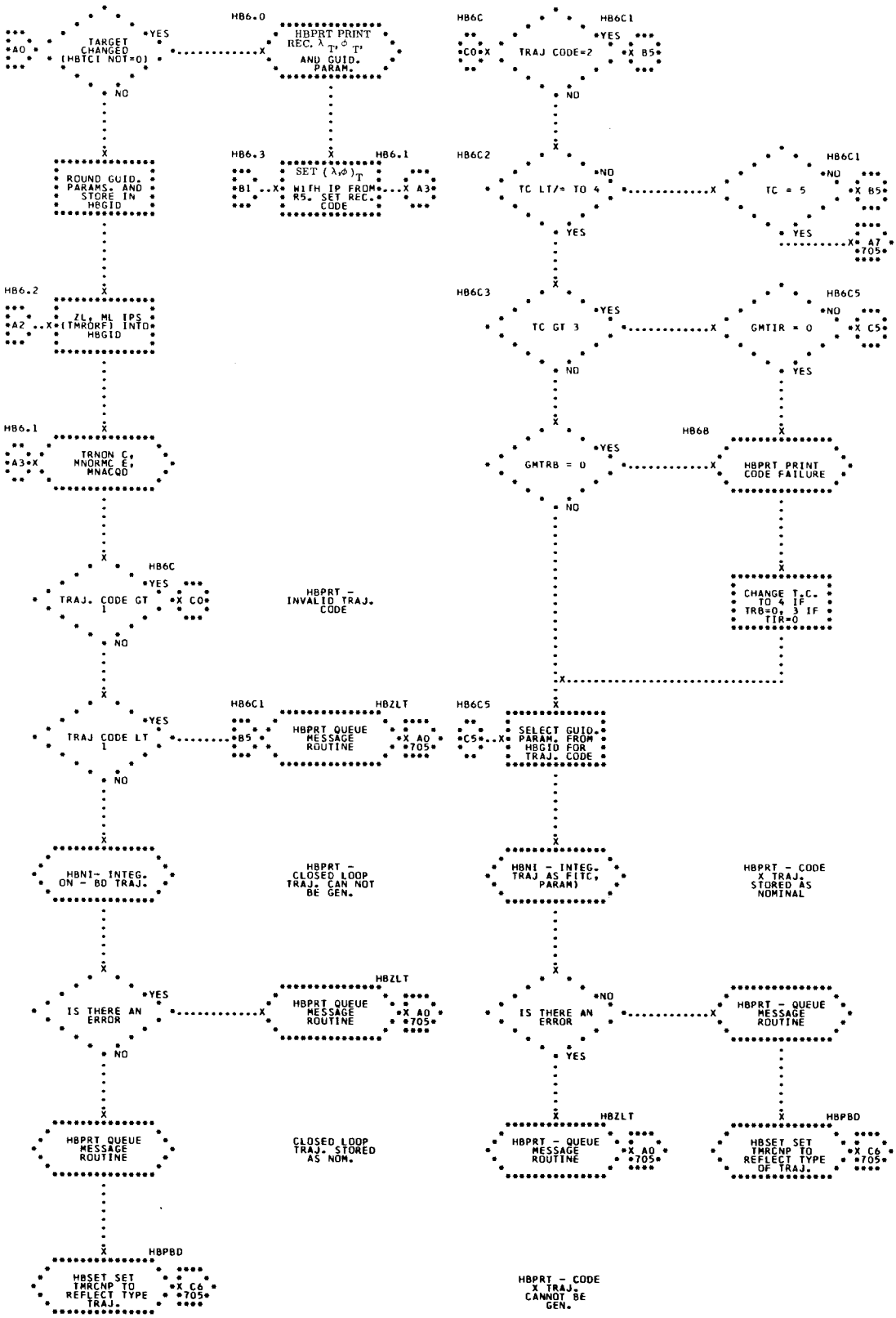

HBGMS
A0
QENBZ (DISABLE)
BUFFERS TMGTRS INTO INTRNL STORAGE
HB2
A2
QENBA (ENABLE)
HB2A
NO
IS VECTOR TIME LT GMTRS
X C0
YES
HBNI INTEGRATE UP TO GMTRS
YES
IS THERE AN ERROR
NO
HB2A1
HB5
YES
ZERO RETROS
X B0 *700*
NO
HB2A2
HBSHF REFORMAT VECTOR INTO R,V
X C1
HBXIT
HBPRT QUEUE MESSAGE ROUTINE
X C7 *706*
VECTOR AT GMTRS CAN NOT BE GENERATED
HB2A
HB5.1
C0
X
ZERO RETROS
YES
X B2 *700*
NO
HB2A2
C1
X
COMPUTE NOMINAL RETRO-FIRE TIMES
HBSTT SET UP THRUST TABLE
HBNI INTEGRATE THRU NUM. RETRO-FIRE
B.O. VECTOR BASED ON GMTRS CAN NOT BE GEN.
YES
IS THERE AN ERROR
HBXIT
HBPRT QUEUE MESSAGE ROUTINE
X C7 *706*
NO
X B0 *700*
HB5

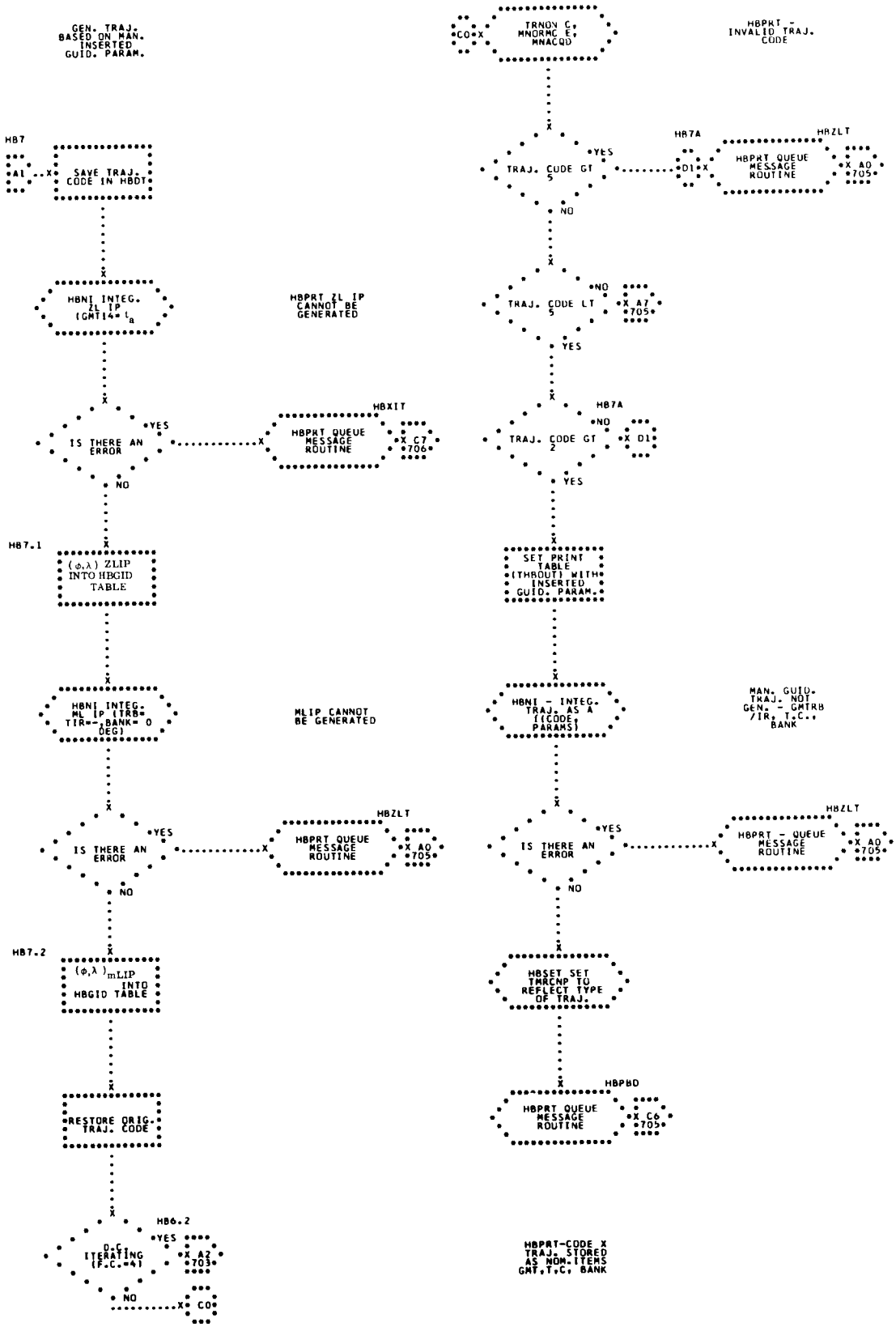

HBGMF
A0
QENBZ (DISABLE)
BUFFER FROM TMRETO λ Γ Φ T T. C. AND
RETRO IND. INTO XR1
HB2
NOMINAL RETRO-FIRE (XR1=0 OR 1)
YES
X A2 +698
NO
HB3
BUFFER RETRO TIMES AND PITCH, ROLL, YAW
OR ΔV INTO AC, MQ, INDS
IVI (XR1=3)
NO
YES
STORE DELTA V AND SET REGS TO NOM. PITCH, ROLL, YAW
HB3.1
STORE P,R,Y (AC) MQ, INDS) INTO THRUST TABLE
QENBA (ENABLE)
X B0
B0
FLOAT RETRO-FIRE TIMES
HB3TT SET UP THRUST TABLE
VECTOR TIME LT GMTRF
NO
YES
HB3A2
FLOAT VECTOR TIME
HB3A3
X B6
HBNI INTEG UP TO GMTRF
IS THERE AN ERROR
YES
NO
HBXIT
HBPRT QUEUE MESSAGE ROUTINE
X C7 +706
VECTOR AT GMTRF CANNOT BE GENERATED
HB3A1
HBSHF REFORMAT VECTOR INTO R,V,T
HB3A3
B6
HBNI INTEG. THRU NON-NOM. RETRO-FIRE
IS THERE AN ERROR
YES
NO
C7
HBXIT
HBPRT QUEUE MESSAGE ROUTINE
X C7 +706
B.O. VECTOR BASED ON GMTRF CAN NOT BE GEN.
HB3A5
HB5
IVI (XR1=3)
NO
X B0 +700
YES
FIX B.O. TIME INTO HBGMT FIX RETROTIME INTO HBRVT
X C0
C0
HBNI INTEG. IN ORBIT UP TO B.O. TIME
IS THERE AN ERROR
YES
NO
D0
COMPUTE ΔV̄'s INTO HBTRV+0, +2 (TMOFNI+7, +9 -TMOLVS+7, +9)
RUCIVI ROTATE IVI'S INTO INERT'L REF. FRAME
IS THERE AN ERROR
YES
X C7
NO
D8
D8
RUPDAT UPDATE BURNOUT VECTOR
IS THERE AN ERROR
YES
X C7
NO
X B0 +700
HB5

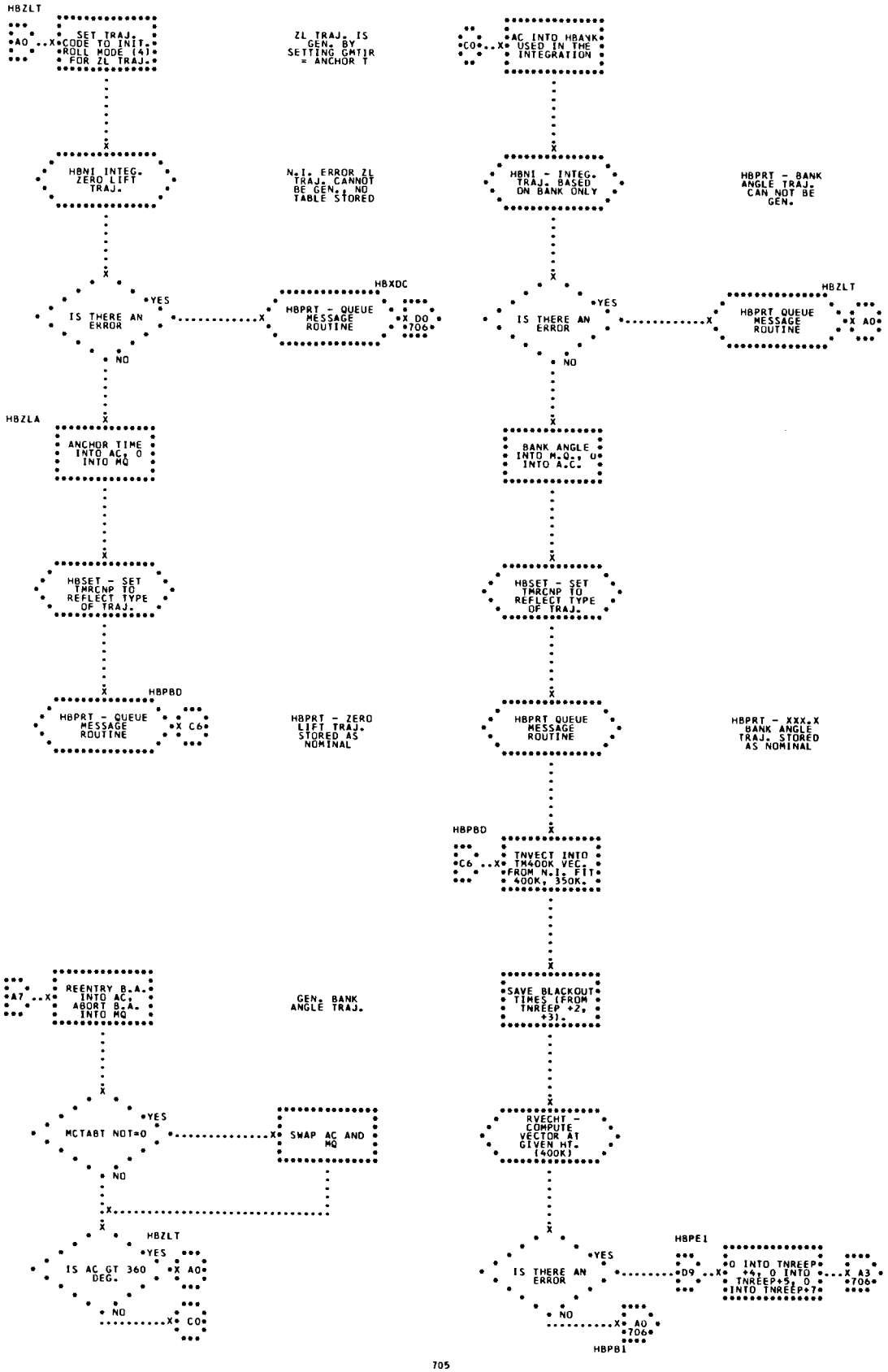

HBREN
QENBZ (DISABLE)
BUFFER TMRTCO INTO INTERNAL STORAGE
QENBA (ENABLE)
D.C. ITERATING (F.C.=4)
YES
X B2
NO
SET RECOVERY CODE =0-H
FLOAT GMTRB /GMTIR INTO AC (TMRTCO+6)
ON-80 GUID. SPECIFIED
SET UP INTERNAL GUID TABLE (HBGID) FROM TMRTCO
HB5
HB5HF REFORMAT B.O. VECTOR INTO R,V,T
FIX B.O. TIME INTO TFESAB (D.C. PARAMETER).
HB5.1
ANCHOR VECTOR INTO TMITNI (INPUT TO R5)
F.C.=2 (RETRO-FIRE CASE)
NO
YES
B.O. VECTOR INTO TDRANV
B.O. VECTOR INTO TMBNOT (PRINT TABLE)
FORMAT B.O. T MIN. IN DECR. SEC. IN ADR. INTO MCBNOT
HB5.4
FLOAT ANCHOR TIME INTO TMITNI+6 AND HBGMT
HB7
MAN. OVERRIDE = 2
X A1 704
D.C. ITERATING F.C.=4
X D0
HBPRT NOTE- PRINT B.O. VECTOR ON-LINE
F.C.=1 (ORBIT CASE)
X B7
X B5
F.C. LT OR = 2 (ORBIT OR RETRO)
NO
YES
X D0
HBPRT QUEUE MESSAGE ROUTINE
HB5.2
TRAJ. CODE=5 (CONTINGENCY)
HB7
λT φT = 0 (TARGET)
X A1 704
HB5A
λT INTO AC
φT INTO MQ
X D4
RSRARF COMPUTE GUID. PARAM. (GMTRB, R1, GMTIR, B1)
IS THERE AN ERROR
X B0 701
X A0 703

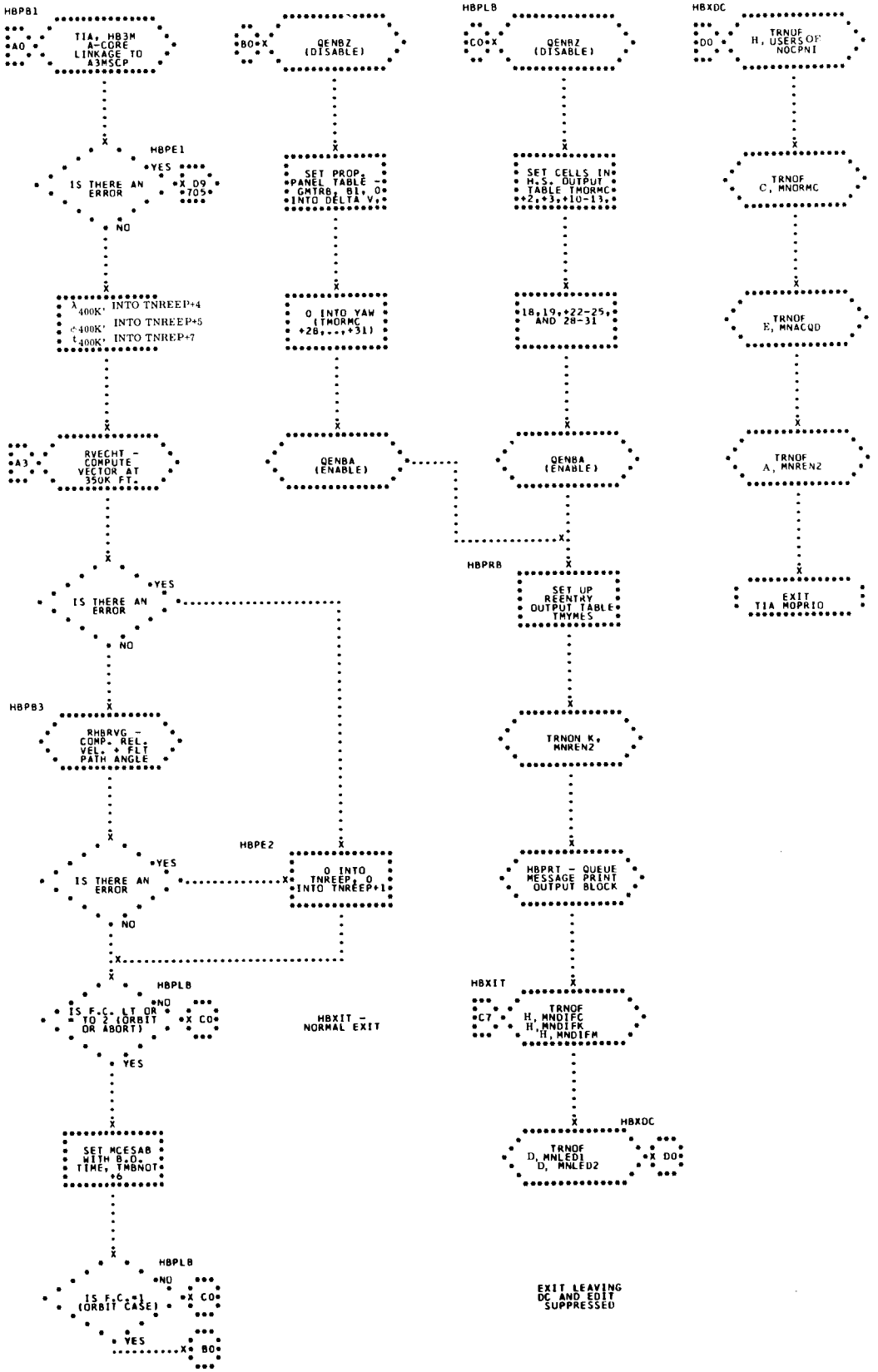

R5RARF ERROR NUMBER INTO HBR5E
B0
ERROR IN GENERATING ZLIP
YES
NO
ZLIP CAN NOT BE GENERATED
HBPRT QUEUE MESSAGE ROUTINE
HBXIT
F₂C₂= 1 (ORBIT CASE)
X C7 *706*
NO
X D0 *706*
HBXDC
ERROR IN GENERATING MLIP
MLIP CAN NOT BE GENERATED
C2
HBZLT
HBPRT QUEUE MESSAGE ROUTINE
X A0 *705*
ERROR IN GENERATING +60 DEG. B IP
X C2
60 DEG. BANK IP CANNOT BE GENERATED
HB5A1
ANCHOR VECTOR BELOW 325K
NO
X A0 *702*
YES
HB6.2
TRAJ CODE=5
YES
X A2 *703*
NO
CONTR. FOR PREV. TRAJ. WILL BE USED FOR THIS TRAJ.
COMP. OF A(TMRCNP) INTO XR1 (TC)
+1, TMRCNP INTO AC, +2, TMRCNP INTO MQ
C(XR1) GT -3 (TC=1)
NO
YES
HB6.2
TC=3,4, OR 5
AC AND MQ INTO GUID. TABLE (HBGID)
X A2 *703*
HB6.2
AC INTO λT
MQ INTO φT
X A2 *703*

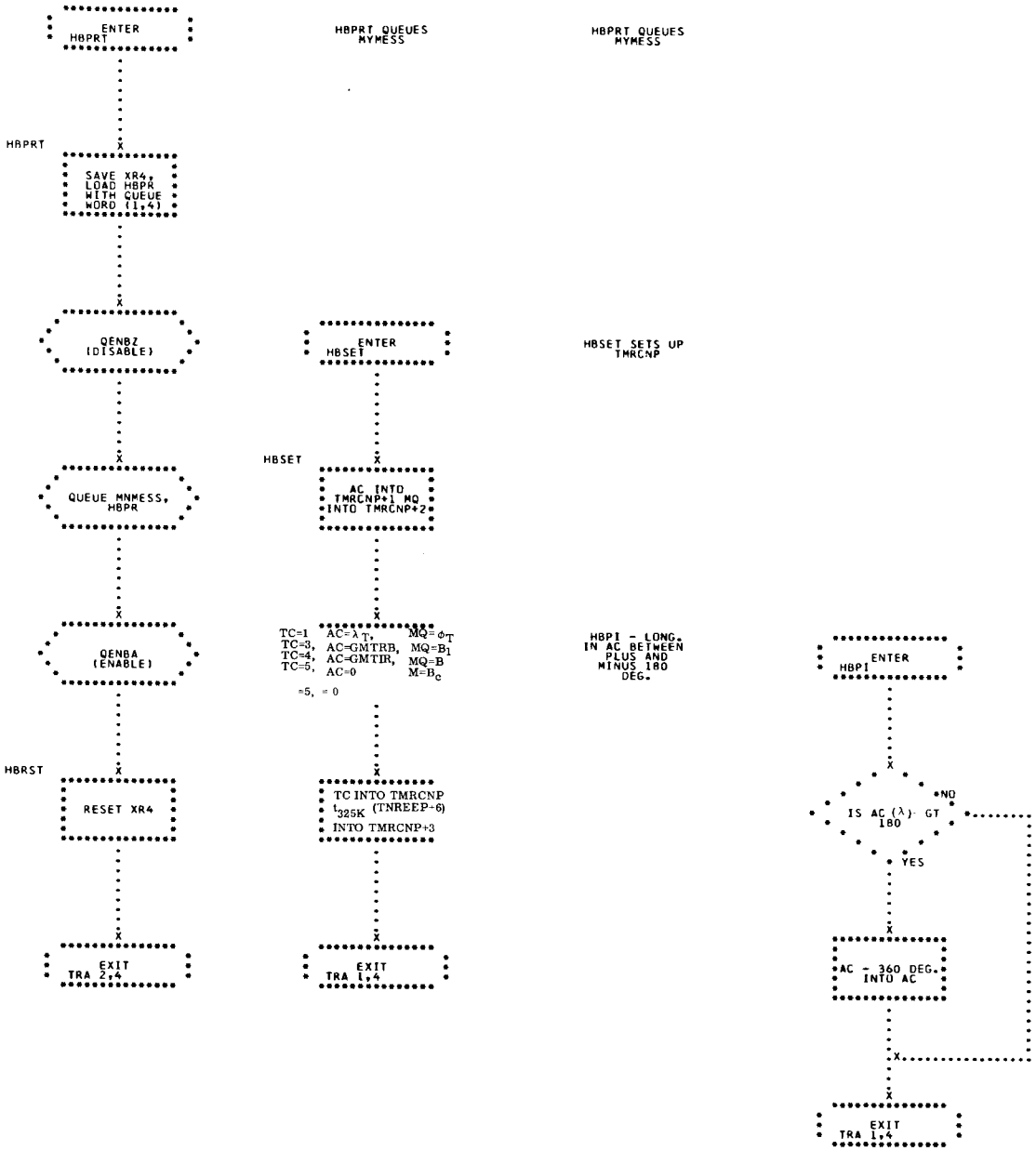

HB5A1
1ST TIME TRU (HBTCI=0)
NO
YES
HBPRT QUEUE MESSAGE ROUTINE
HB5A2 PRINT R5RARF ERROR NO.
STORE F.P. IPS AND GUID PARAMS (TMRORF) INTO HBGID
HBPRT QUEUE MESSAGE ROUTINE
HBPRT PRINT - $\phi_T, \lambda_T, \Delta\phi$, $\Delta\lambda$ BANK ANG. λ_{NT}, ϕ_{NT}, NT - NEAREST IP TO TARGET
GMTRB = GMTIR=0
YES
NO
HB5A4
SET TARGET WITH IP NEAREST TARGET
HB5A3
SET T.C.=5, SET BANK=999 (INDICATES ZL)
HB5A6
WAS TARGET BEHIND F.P.
COMPUTE MISS DISTANCE = $\sqrt{\Delta\phi^2 + \Delta\lambda^2}$ INTO HBT1
T.C. GT 1
T.C. LT/= TO 4
SET 1ST TIME CONTROL (HBTCI) NOT=0
HB5A2
HB5A3
HB5A2
MAN. TARGET
HBPRT QUE MESS. "MAN. TARGET NOT IN F.P."
GMTRS COMP
HBPRT QUE MESS. GMTRS TARGET NOT IN FOOTPRINT
HB6.3
MISS DIST. GT 50 N.M.
HBXIT
TRAJ. CODE=5
ALL PRIMARY AREAS SCANNED
SELECT NEXT TARGET FROM PRIM. AREAS (TMLMPT)
HBPRT QUE MESS. "NO PRIMARY AREA IN FOOTPRINT"
HB5A
SELECT CORRESP. REC. CODE FROM TMPLON
HB6C1
TC GT 5
TC LT 5
HB6.3

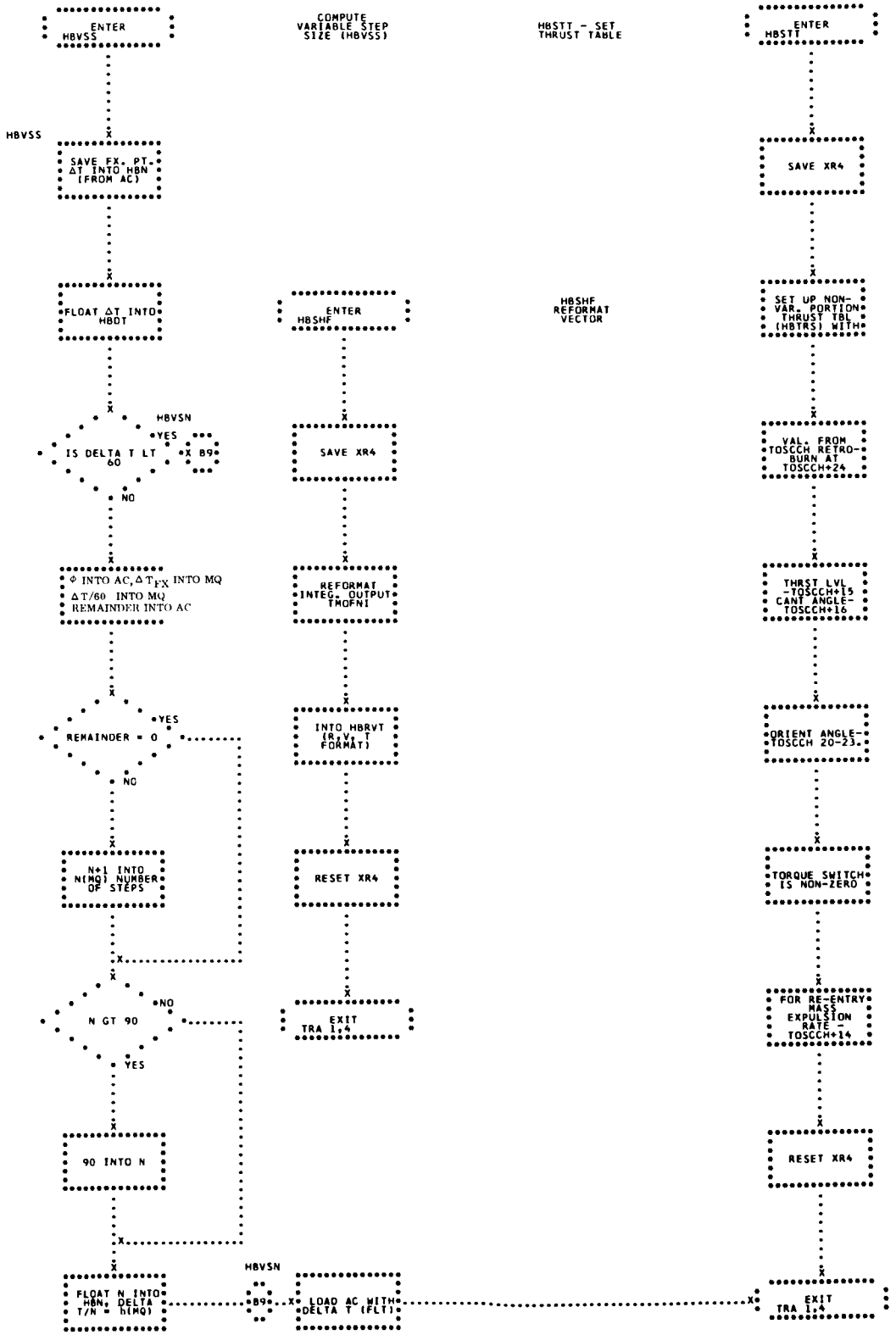

TARGET CHANGED (HBTCI NOT=0)
YES
NO
A0
HB6.0
HBPRT PRINT REC. λT, φT AND GUID. PARAM.
ROUND GUID. PARAMS. AND STORE IN HBGID
HB6.3
B1
HB6.1
SET (λ,φ)T WITH IP FROM R5. SET REC. CODE
A3
HB6.2
A2
ZL, ML IPS (TMRORF) INTO HBGID
HB6.1
A3
TRNON C, MNORMC E, MNACQD
HB6C
TRAJ. CODE GT 1
C0
HBPRT - INVALID TRAJ. CODE
TRAJ CODE LT 1
HB6C1
B5
HBZLT
HBPRT QUEUE MESSAGE ROUTINE
A0
705
HBNI- INTEG. ON - BD TRAJ.
HBPRT - CLOSED LOOP TRAJ. CAN NOT BE GEN.
IS THERE AN ERROR
HBPRT QUEUE MESSAGE ROUTINE
CLOSED LOOP TRAJ. STORED AS NOM.
HBPBD
HBSET SET TMRCNP TO REFLECT TYPE TRAJ.
C6
HB6C
TRAJ CODE=2
HB6C1
B5
HB6C2
TC LT/= TO 4
TC = 5
A7
HB6C3
TC GT 3
HB6C5
GMTIR = 0
C5
GMTRB = 0
HB68
HBPRT PRINT CODE FAILURE
CHANGE T.C. TO 4 IF TRB=0, 3 IF TIR=0
HB6C5
C5
SELECT GUID. PARAM. FROM HBGID FOR TRAJ. CODE
HBNI - INTEG. TRAJ AS FITC, PARAM)
HBPRT - CODE X TRAJ. STORED AS NOMINAL
IS THERE AN ERROR
HBPRT - QUEUE MESSAGE ROUTINE
HBPRT - CODE X TRAJ. CANNOT BE GEN.
HBSET SET TMRCNP TO REFLECT TYPE OF TRAJ.

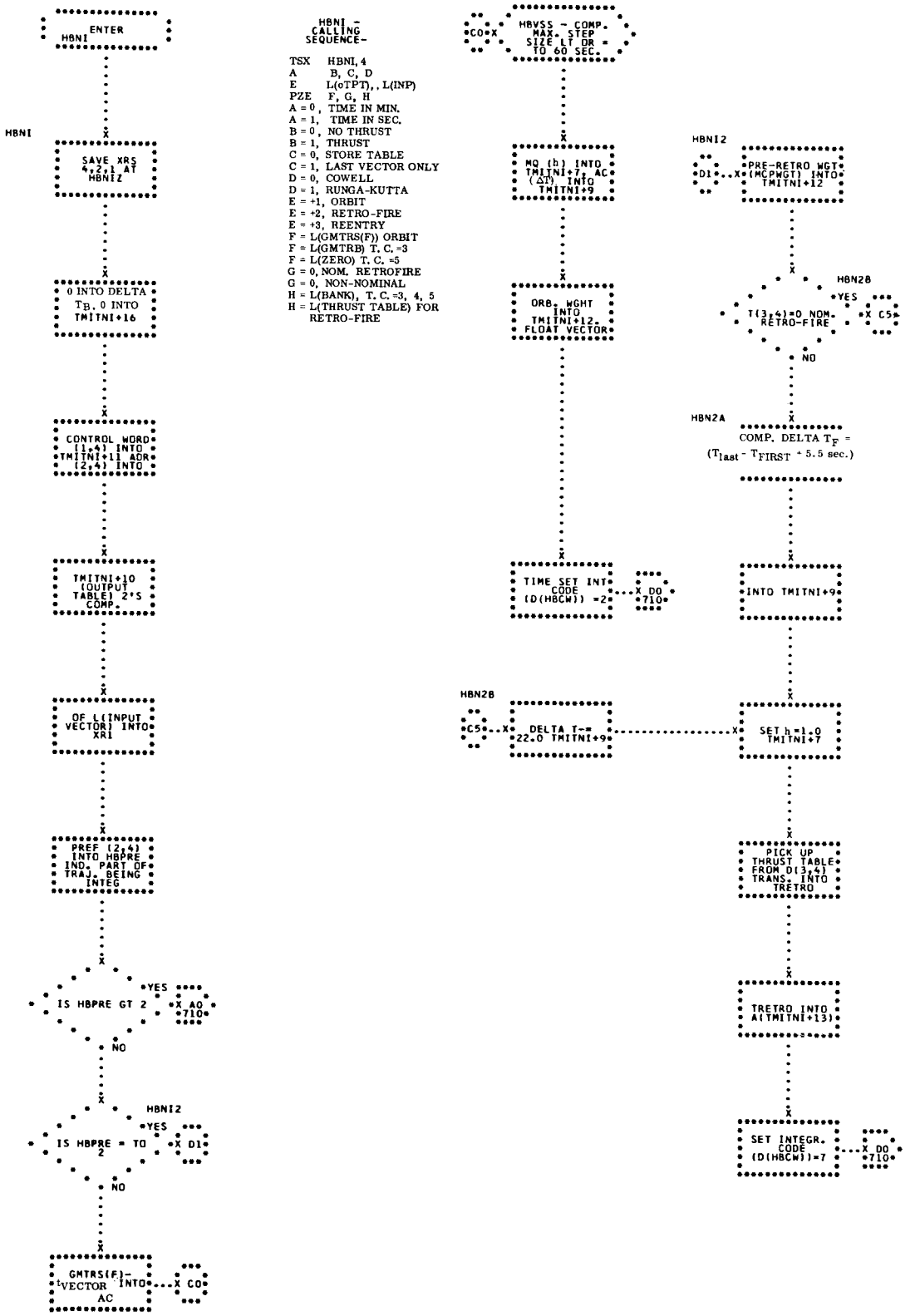

GEN. TRAJ. BASED ON MAN. INSERTED GUID. PARAM.
HB7
SAVE TRAJ. CODE IN HBDT
HBNI INTEG. ZL IP (GMT14= t_a
HBPRT ZL IP CANNOT BE GENERATED
IS THERE AN ERROR
YES
HBXIT
HBPRT QUEUE MESSAGE ROUTINE
X C7 *706*
NO
HB7.1
(ϕ,λ) ZLIP INTO HBGID TABLE
HBNI INTEG. ML IP (TRB= TIR=, BANK= 0 DEG)
MLIP CANNOT BE GENERATED
IS THERE AN ERROR
YES
HBZLT
HBPRT QUEUE MESSAGE ROUTINE
X A0 *705*
NO
HB7.2
(ϕ,λ)$_{mLIP}$ INTO HBGID TABLE
RESTORE ORIG. TRAJ. CODE
HB6.2
D.C ITERATING (F.C.=4)
YES
X A2 *703*
NO
X C0
TRNOV C, MNORMC E, MNACQD
HBPRT - INVALID TRAJ. CODE
TRAJ. CODE GT 5
YES
HB7A
D1
HBZLT
HBPRT QUEUE MESSAGE ROUTINE
X A0 *705*
NO
TRAJ. CODE LT 5
NO
X A7 *705*
YES
TRAJ. CODE GT 2
NO
HB7A
X D1
YES
SET PRINT TABLE (THROUT) WITH INSERTED GUID. PARAM.
HBNI - INTEG. TRAJ. AS A ((CODE, PARAMS)
MAN. GUID. TRAJ. NOT GEN. - GMTRB /IR, T.C., BANK
IS THERE AN ERROR
YES
HBZLT
HBPRT - QUEUE MESSAGE ROUTINE
X A0 *705*
NO
HBSET SET TMRCNP TO REFLECT TYPE OF TRAJ.
HBPBD
HBPRT QUEUE MESSAGE ROUTINE
X C6 *705*
HBPRT-CODE X TRAJ. STORED AS NOM. ITEMS GMT, T.C, BANK

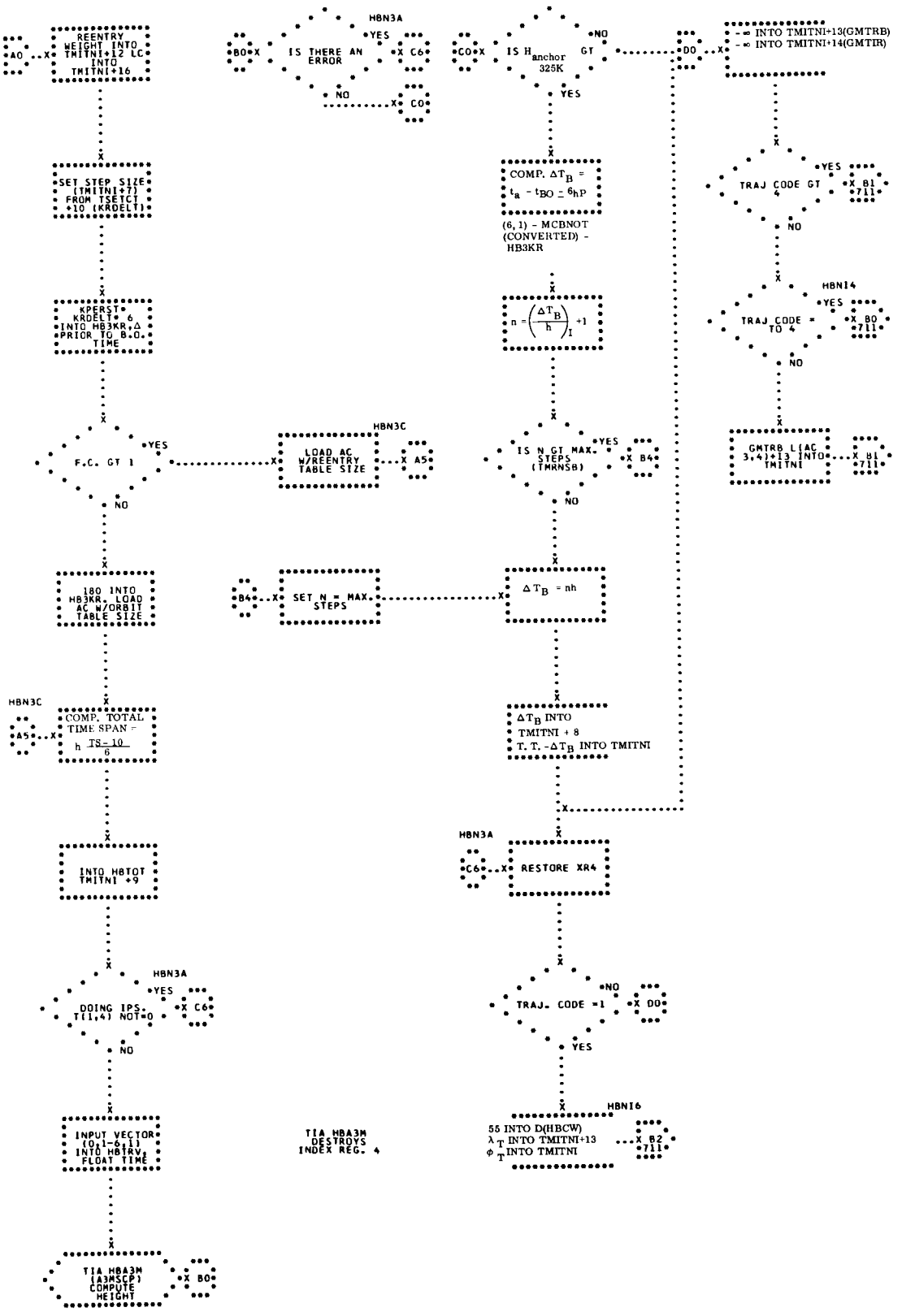

HBZLT
AO
SET TRAJ. CODE TO INIT. ROLL MODE (4) FOR ZL TRAJ.
ZL TRAJ. IS GEN. BY SETTING CATIR = ANCHOR T
HBNI INTEG. ZERO LIFT TRAJ.
N.I. ERROR ZL TRAJ. CANNOT BE GEN., NO TABLE STORED
IS THERE AN ERROR
YES
HBXDC
HBPRT - QUEUE MESSAGE ROUTINE
X DO *706*
NO
HBZLA
ANCHOR TIME INTO AC, 0 INTO MQ
HBSET - SET TMRCNP TO REFLECT TYPE OF TRAJ.
HBPBD
HBPRT - QUEUE MESSAGE ROUTINE
X C6
HBPRT - ZERO LIFT TRAJ. STORED AS NOMINAL
A7
REENTRY B.A. INTO AC, ABORT B.A. INTO MQ
GEN. BANK ANGLE TRAJ.
MCTABT NOT=0
SWAP AC AND MQ
HBZLT
IS AC GT 360 DEG.
X AO
CO
C0
AC INTO HBANK USED IN THE INTEGRATION
HBNI - INTEG. TRAJ. BASED ON BANK ONLY
HBPRT - BANK ANGLE TRAJ. CAN NOT BE GEN.
HBZLT
HBPRT QUEUE MESSAGE ROUTINE
X AO
BANK ANGLE INTO M.Q., 0 INTO A.C.
HBSET - SET TMRCNP TO REFLECT TYPE OF TRAJ.
HBPRT QUEUE MESSAGE ROUTINE
HBPRT - XXX.X BANK ANGLE TRAJ. STORED AS NOMINAL
HBPBD
C6
TNVECT INTO TM400K VEC. FROM N.I. FIT *400K, 350K*
SAVE BLACKOUT TIMES (FROM TNREEP +2, +3).
RVECHT - COMPUTE VECTOR AT GIVEN HT. (400K)
IS THERE AN ERROR
HBPE1
D9
0 INTO TNREEP +4, 0 INTO TNREEP+5, 0 INTO TNREEP+7
X A3 *706*
NO
X AO *706*
HBPB1

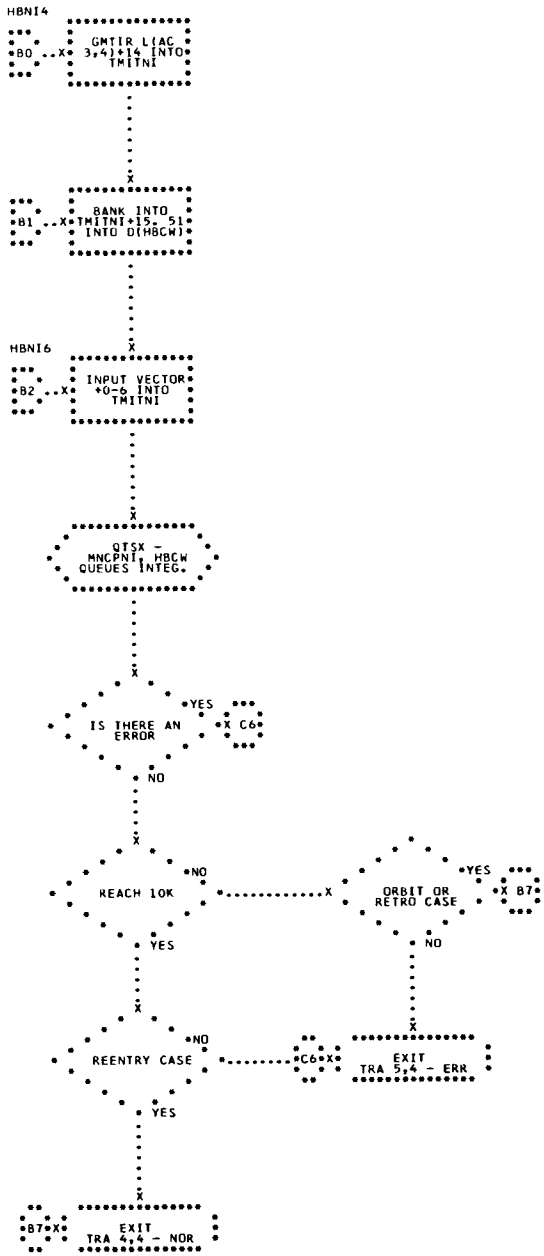

HBPB1
TIA, HB3M A-CORE LINKAGE TO A3MSCP
A0
HBPE1
IS THERE AN ERROR
YES
X D9 *705*
NO
λ_{400K} INTO TNREEP+4
ϕ_{400K} INTO TNREEP+5
t_{400K} INTO TNREEP+7
RVECHT - COMPUTE VECTOR AT 350K FT.
A3
IS THERE AN ERROR
YES
NO
HBPB3
RHBRVG - COMP. REL. VEL. & FLT PATH ANGLE
IS THERE AN ERROR
YES
NO
HBPE2
0 INTO TNREEP, 0 INTO TNREEP+1
HBPLB
IS F.C. LT OR = TO 2 (ORBIT OR ABORT)
NO
X CO
YES
SET MCESAB WITH 8.0, TIME, TMBNOT+6
HBPLB
IS F.C.=1 (ORBIT CASE)
NO
X CO
YES
X BO
BO
QENBZ (DISABLE)
SET PROP. PANEL TABLE - GMTRB, B1, 0 INTO DELTA V,
0 INTO YAW (TMORMC +28,...,+31)
QENBA (ENABLE)
HBXIT - NORMAL EXIT
HBPLB
CO
QENBZ (DISABLE)
SET CELLS IN H.S. OUTPUT TABLE TMORMC +2,+3,+10-13,
18,19,+22-25, AND 28-31
QENBA (ENABLE)
HBPRB
SET UP REENTRY OUTPUT TABLE TMYMES
TRNON K, MNREN2
HBPRT - QUEUE MESSAGE PRINT OUTPUT BLOCK
HBXIT
C7
TRNOF H, MNDIFC H, MNDIFK H, MNDIFM
HBXDC
TRNOF D, MNLED1 D, MNLED2
X DO
EXIT LEAVING DC AND EDIT SUPPRESSED
HBXDC
DO
TRNOF H, USERS OF NOCPNI
TRNOF C, MNORMC
TRNOF E, MNACQD
TRNOF A, MNREN2
EXIT TIA MOPRIO


ENTER HBPRT
HBPRT
SAVE XR4, LOAD HBPR WITH QUEUE WORD (1,4)
QENBZ (DISABLE)
QUEUE MNMESS, HBPR
QENBA (ENABLE)
HBRST
RESET XR4
EXIT TRA 2,4
HBPRT QUEUES MYMESS
ENTER HBSET
HBSET
AC INTO TMRCNP+1 MQ INTO TMRCNP+2
TC=1 AC=λT, MQ=φT
TC=3, AC=GMTRB, MQ=B1
TC=4, AC=-GMTIR, MQ=B
TC=5, AC=0 M=Bc
=5, = 0
TC INTO TMRCNP
t325K (TNREEP-6)
INTO TMRCNP+3
EXIT TRA 1,4
HBPRT QUEUES MYMESS
HBSET SETS UP TMRCNP
HBPI - LONG. IN AC BETWEEN PLUS AND MINUS 180 DEG.
ENTER HBPI
IS AC (λ) GT 180
NO
YES
AC - 360 DEG. INTO AC
EXIT TRA 1,4


ENTER HBVSS
COMPUTE VARIABLE STEP SIZE (HBVSS)
HBSTT - SET THRUST TABLE
ENTER HBSTT
HBVSS
SAVE FX. PT. ΔT INTO HBN (FROM AC)
SAVE XR4
FLOAT ΔT INTO HBDT
ENTER HBSHF
HBSHF REFORMAT VECTOR
SET UP NON-VAR. PORTION THRUST TBL (HBTRS) WITH
HBVSN
YES
IS DELTA T LT 60
X B9
NO
SAVE XR4
VAL. FROM TOSCCH RETRO-BURN AT TOSCCH+24
Φ INTO AC, ΔT_FX INTO MQ
ΔT/60 INTO MQ
REMAINDER INTO AC
REFORMAT INTEG. OUTPUT TMOFNI
THRST LVL -TOSCCH+15 CANT ANGLE-TOSCCH+16
YES
REMAINDER = 0
NO
INTO HBRVT (R,V,T FORMAT)
ORIENT ANGLE-TOSCCH 20-23.
N+1 INTO N(MQ) NUMBER OF STEPS
RESET XR4
TORQUE SWITCH IS NON-ZERO
NO
N GT 90
YES
EXIT TRA 1,4
FOR RE-ENTRY MASS EXPULSION RATE - TOSCCH+14
90 INTO N
RESET XR4
HBVSN
FLOAT N INTO HBN, DELTA T/N = h(MQ)
B9
LOAD AC WITH DELTA T (FLT)
EXIT TRA 1,4


ENTER HBNI
HBNI
SAVE XRS 4,2,1 AT HBNIZ
0 INTO DELTA TB, 0 INTO TMITNI+16
CONTROL WORD (1,4) INTO TMITNI+11 ADR (2,4) INTO
TMITNI+10 (OUTPUT TABLE) 2'S COMP.
OF L(INPUT VECTOR) INTO XR1
PREF (2,4) INTO HBPRE IND. PART OF TRAJ. BEING INTEG
IS HBPRE GT 2
YES
X A0 *710*
NO
HBNI2
IS HBPRE = TO 2
YES
X D1
NO
GMTRS(F)-VECTOR INTO AC
X C0
HBNI - CALLING SEQUENCE-
TSX HBNI, 4
A B, C, D
E L(oTPT), , L(INP)
PZE F, G, H
A = 0, TIME IN MIN.
A = 1, TIME IN SEC.
B = 0, NO THRUST
B = 1, THRUST
C = 0, STORE TABLE
C = 1, LAST VECTOR ONLY
D = 0, COWELL
D = 1, RUNGA-KUTTA
E = +1, ORBIT
E = +2, RETRO-FIRE
E = +3, REENTRY
F = L(GMTRS(F)) ORBIT
F = L(GMTRB) T. C. =3
F = L(ZERO) T. C. =5
G = 0, NOM. RETROFIRE
G = 0, NON-NOMINAL
H = L(BANK), T. C. =3, 4, 5
H = L(THRUST TABLE) FOR RETRO-FIRE
C0 HBVSS - COMP. MAX. STEP SIZE LT OR = TO 60 SEC.
MQ (h) INTO TMITNI+7, AC (ΔT) INTO TMITNI+9
ORB. WGHT INTO TMITNI+12. FLOAT VECTOR
TIME SET INT CODE (D(HBCW)) =2
X D0 *710*
HBN2B
C5
DELTA T-= 22.0 TMITNI+9
HBNI2
D1
PRE-RETRO WGT (MCPWGT) INTO TMITNI+12
HBN2B
YES
T(3,4)=0 NOM. RETRO-FIRE
X C5
NO
HBN2A
COMP. DELTA TF = (Tlast - TFIRST + 5.5 sec.)
INTO TMITNI+9
SET h=1.0 TMITNI+7
PICK UP THRUST TABLE FROM D(3,4) TRANS. INTO TRETRO
TRETRO INTO A(TMITNI+13)
SET INTEGR. CODE (D(HBCW))=7
X D0 *710*


REENTRY WEIGHT INTO TMITNI+12 LC INTO TMITNI+16
A0
SET STEP SIZE (TMITNI+7) FROM TSETCT +10 (KRDELT)
KPERST* KRDELT* 6 INTO HB3KR, Δ PRIOR TO B.O. TIME
F.C. GT 1
YES
NO
180 INTO HB3KR, LOAD AC W/ORBIT TABLE SIZE
HBN3C
A5
COMP. TOTAL TIME SPAN = h TS-10/6
INTO HBTOT TMITNI +9
HBN3A
DOING IPS. T(1,4) NOT=0
YES
C6
NO
INPUT VECTOR (0,1-6,1) INTO HBTRV, FLOAT TIME
TIA HBA3M (A3MSCP) COMPUTE HEIGHT
B0
HBN3A
IS THERE AN ERROR
YES
C6
NO
C0
LOAD AC W/REENTRY TABLE SIZE
HBN3C
A5
B4
SET N = MAX. STEPS
TIA HBA3M DESTROYS INDEX REG. 4
IS H anchor GT 325K
C0
NO
YES
COMP. ΔT_B = t_a - t_BO ≥ δ_hP
(6,1) - MCBNOT (CONVERTED) - HB3KR
n = (ΔT_B/h)_I + 1
IS N GT MAX. STEPS (TMRNSB)
YES
B4
NO
ΔT_B = nh
ΔT_B INTO TMITNI + 8
T. T. -ΔT_B INTO TMITNI
HBN3A
C6
RESTORE XR4
TRAJ. CODE =1
NO
D0
YES
HBN16
55 INTO D(HBCW)
λ_T INTO TMITNI+13
φ_T INTO TMITNI
B2
711
D0
-∞ INTO TMITNI+13(GMTRB)
-∞ INTO TMITNI+14(GMTIR)
TRAJ CODE GT 4
YES
B1
711
NO
HBN14
TRAJ CODE = TO 4
YES
B0
711
NO
GMTRB L(AC 3,4)+13 INTO TMITNI
B1
711


HBNI4
GMTIR L(AC 3,4)+14 INTO TMITNI
B0
BANK INTO TMITNI+15, 5 INTO D(HBCW)
B1
HBNI6
INPUT VECTOR +0-6 INTO TMITNI
B2
QTSX - MNCPNT, HBCW QUEUES INTEG.
IS THERE AN ERROR
YES
X C6
NO
REACH 10K
NO
YES
ORBIT OR RETRO CASE
YES
X B7
NO
REENTRY CASE
NO
C6
EXIT TRA 5,4 - ERR
YES
B7
EXIT TRA 4,4 - NOR

11 RUPDAT - FOR UPDATING THE POST RETROFIRE TRAJECTORY

THIS PROGRAM UPDATES THE POST RETROFIRE POSITION AND VELOCITY VECTORS IMMEDIATELY AFTER RETROFIRE, USING THE POST RETROFIRE ACCUMULATED ACCELEROMETER OUTPUT, A SIMULATED RETROFIRE OUTPUT, AND FOUR RETROFIRING INDICATORS. RUPDAT IS A SUBROUTINE OF MYREN2. (SEE FLOWCHART)

INPUT

A) TABLES -

A 3-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

$L(\Delta\bar{v}_{acc})$	$\Delta\dot{X}_{acc}$	= X COMPONENT OF THE VELOCITY INCREMENT, FLOATING POINT GEMINI VELOCITY UNITS
+1	$\Delta\dot{Y}_{acc}$	= Y COMPONENT OF THE VELOCITY INCREMENT, FLOATING POINT GEMINI VELOCITY UNITS
+2	$\Delta\dot{Z}_{acc}$	= Z COMPONENT OF THE VELOCITY INCREMENT, FLOATING POINT GEMINI VELOCITY UNITS

A 3-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

$L(\Delta\bar{v}_{int})$	$\Delta\dot{X}_{int}$	= X COMPONENT OF THE SIMULATED VELOCITY INCREMENT, FLOATING POINT GEMINI VELOCITY UNITS

+1	$\Delta\dot{Y}_{int}$	= Y COMPONENT OF THE SIMULATED VELOCITY INCREMENT, FLOATING POINT GEMINI VELOCITY UNITS
+2	$\Delta\dot{Z}_{int}$	= Z COMPONENT OF THE SIMULATED VELOCITY INCREMENT, FLOATING POINT GEMINI VELOCITY UNITS

A 4-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L(RETROFIRING IND.)	GMT OF FIRING OF RETORROCKET NUMBER 1, FIXED POINT SECONDS, OR 0 IF NOT FIRED
+1	GMT OF FIRING OF RETRO ROCKET NUMBER 2, FIXED POINT SECONDS, OR 0 IF NOT FIRED
+2	GMT OF FIRING OF RETRO ROCKET NUMBER 3, FIXED POINT SECONDS, OR 0 IF NOT FIRED
+3	GMT OF FIRING OF RETRO ROCKET NUMBER 4, FIXED POINT SECONDS, OR 0 IF NOT FIRED

A 6-LOCATION TABLE CONTAINING THE POSITION AND VELOCITY VECTORS AT THE TERMINATION OF RETRO-FIRE, ARRANGED IN THE FOLLOWING FORMAT -

L ($\bar{r}_{BO}$, $\bar{v}_{BO}$)	X BEFORE UPDATING, FLOATING POINT GEMINI LENGTH UNITS
+1	Y BEFORE UPDATING, FLOATING POINT GEMINI LENGTH UNITS
+2	Z BEFORE UPDATING, FLOATING POINT GEMINI LENGTH UNITS
+3	$\bar{x}$ BEFORE UPDATING, FLOATING POINT

		GEMINI VELOCITY UNITS
	+4	$\dot{Y}$ BEFORE UPDATING, FLOATING POINT GEMINI VELOCITY UNITS
	+5	$\dot{Z}$ BEFORE UPDATING, FLOATING POINT GEMINI VELOCITY UNITS

B) CONSTANTS -

K00000	ZERO
KCH233	OCTAL MASK FOR FLOATING POINT CHARACTERISTIC, 233 000 000 000

OUTPUT

A 6-LOCATION TABLE CONTAINING THE UPDATED BURN-OUT VECTORS, ARRANGED IN THE FOLLOWING FORMAT - (THIS IS THE SAME TABLE THAT WAS REFERRED TO UNDER INPUT.)

L($\bar{r}_{BO}$, $\bar{v}_{BO}$)	X AFTER UPDATING, FLOATING POINT GEMINI LENGTH UNITS
+1	Y AFTER UPDATING, FLOATING POINT GEMINI LENGTH UNITS
+2	Z AFTER UPDATING, FLOATING POINT GEMINI LENGTH UNITS
+3	$\dot{X}$ AFTER UPDATING, FLOATING POINT GEMINI VELOCITY UNITS
+4	$\dot{Y}$ AFTER UPDATING, FLOATING POINT GEMINI VELOCITY UNITS
+5	$\dot{Z}$ AFTER UPDATING, FLOATING POINT GEMINI VELOCITY UNITS

METHOD

A) THE MATHEMATICS OF THE PROGRAM IS AS FOLLOWS - (THIS FORMULATION IS BASED ON A MEMORANDUM FROM THE MISSION ANALYSIS BRANCH, MSC, HOUSTON, DATED JANUARY 19, 1965.)
THE VELOCITY VECTOR IS UPDATED USING THESE EQUATIONS

$$\Delta\bar{v}_{acc} - \Delta\bar{v}_{int} = \bar{D}$$

$$\bar{v}_{BO} + \bar{D} = \bar{v}_{BO}$$

TO ACCOUNT FOR RETROMISFIRES ON POSITION ERROR, IT IS NECESSARY TO CONSIDER EACH ROCKET SEPARATELY SINCE NO VELOCITY ERROR WILL BE ACCUMULATED DURING THE NO-THRUSTING PERIOD. THE FOLLOWING METHOD IS USED -

$$\bar{r}_{E_1} = \left(\frac{K_1}{2N}\right) T_{b_1} \bar{D}$$

$$\bar{r}_{E_3} = \left(\frac{2K_1 + K_3}{2N}\right) T_{b_3} \bar{D}$$

$$\bar{r}_{E_2} = \left(\frac{2K_1 + 2K_3 + K_2}{2N}\right) T_{b_2} \bar{D}$$

$$\bar{r}_{E_4} = \left(\frac{2K_1 + 2K_3 + 2K_2 + K_4}{2N}\right) T_{b_4} \bar{D}$$

WHERE

NUMERAL SUBSCRIPTS REFER TO THE RETROROCKETS WITH A 1,3,2,4 FIRING SEQUENCE

K_i = 1 IF i^{th} RETROROCKET FIRES

K_i = 0 IF i^{th} RETROROCKET MISFIRES

T_{bi} = BURNING TIME OF ith RETROROCKET

i = 1,2,3,4

$N = K_1 + K_3 + K_2 + K_4$

ASSUMING $T_{b1} = T_{b3} = T_{b2} = T_{b4}$ AND SUMMING THESE POSITION ERRORS GIVES

$$F\overline{D} = \frac{ST_b}{2N}\overline{D}$$

WHERE

$S = 7K_1 + 5K_3 + 3K_2 + K_4$

T_b = SIMULATED BURNING TIME OF EACH RETROROCKET, CURRENTLY 5.5 SECONDS

FINALLY, THE POSITION VECTOR IS UPDATED USING THIS EQUATION -

$$\overline{r}_{BO} + F\overline{D} = \overline{r}_{BO}$$

B) THE LOGIC OF THIS PROGRAM IS AS FOLLOWS -

WHEN CONTROL IS TRANSFERRED TO RUPDAT, THE CONTENTS OF THE INDEX REGISTERS USED ARE SAVED.

NEXT THE RETROFIRING INDICATORS ARE TESTED AND THE QUANTITIES 2N AND S ARE COMPUTED.

IF 2N = 0 (I.E., NONE OF THE FOUR RETROROCKETS FIRED), THEN THE POSITION AND VELOCITY VECTORS ARE NOT UPDATED AND CONTROL IS TRANSFERRED TO THE LAST PARAGRAPH OF THIS SECTION.

IF 2N $\neq$ 0, F AND $\bar{D}$ ARE COMPUTED, AND THE VELOCITY AND POSITION VECTORS ARE UPDATED. FINALLY, THE CONTENTS OF THE INDEX REGISTERS USED ARE RESTORED AND CONTROL IS TRANSFERRED BACK TO MYREN2.

USAGE

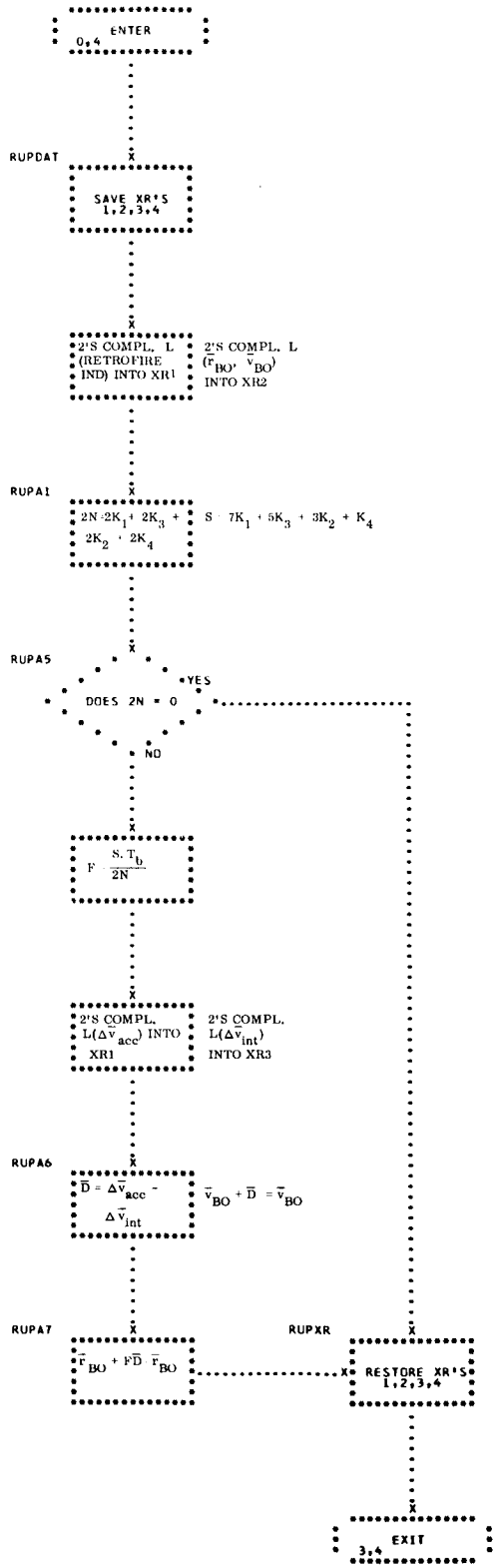
A) CALLING SEQUENCE -

```
TSX       RUPDAT,4
PZE       L(Δv̄acc), , L(Δv̄int)
PZE       L (r̄BO, v̄BO) ,,L(RETROFIRING IND.)
NORMAL    RETURN
```

THIS IS A B CORE PROGRAM WITH ALL STORAGE AND CONSTANTS IN A CORE, WHICH MEANS THE ECC IS NORMALLY SET TO A CORE AND THE ICC SET TO B CORE. THIS MINIMIZES THE NUMBER OF SEA AND SEB INSTRUCTIONS.

B) STORAGE REQUIRED - A CORE - 5 LOCATIONS
B CORE - 63 LOCATIONS

C) TIMING (7094) - .470 MS (MAXIMUM TIME REQUIRED, I.E., ALL FOUR RETROROCKETS FIRED.)


ENTER
0,4
RUPDAT
SAVE XR'S
1,2,3,4
2'S COMPL. L
(RETROFIRE
IND) INTO XR1
2'S COMPL. L
($\bar{r}_{BO}$, $\bar{v}_{BO}$)
INTO XR2
RUPA1
$2N=2K_1+2K_3+2K_2+2K_4$
$S=7K_1+5K_3+3K_2+K_4$
RUPA5
DOES 2N = 0
YES
NO
$F=\frac{S.T_b}{2N}$
2'S COMPL.
$L(\Delta\bar{v}_{acc})$ INTO
XR1
2'S COMPL.
$L(\Delta\bar{v}_{int})$
INTO XR3
RUPA6
$\bar{D}=\Delta\bar{v}_{acc}-\Delta\bar{v}_{int}$
$\bar{v}_{BO}+\bar{D}=\bar{v}_{BO}$
RUPA7
RUPXR
$\bar{r}_{BO}+F\bar{D}\ \bar{r}_{BO}$
RESTORE XR'S
1,2,3,4
EXIT
3,4

12. RUCIVI - PLATFORM TO INERTIAL COORDINATES CONVERSION

THIS PROGRAM CONVERTS VELOCITY VECTOR INCREMENTS FROM INERTIAL PLATFORM COORDINATES TO INERTIAL GEMINI COORDINATES. RUCIVI IS A SUBROUTINE OF MYREN2. (SEE FLOWCHART.)

INPUT

A) TABLES -

A 3-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L($\Delta\bar{v}_{IVI}$)	$\Delta\dot{X}_p = X_p$	COMPONENT OF THE VELOCITY INCREMENT, FLOATING POINT FT/SEC
+1	$\Delta\dot{Y}_p = Y_p$	COMPONENT OF THE VELOCITY INCREMENT, FLOATING POINT FT/SEC
+2	$\Delta\dot{Z}_p = Z_p$	COMPONENT OF THE VELOCITY INCREMENT, FLOATING POINT FT/SEC

A 6-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L($\bar{r}$, $\bar{v}$) AT

DEFINITION X, FLOATING POINT GEMINI LENGTH OF PLATFORM) UNITS

+1	Y, FLOATING POINT GEMINI LENGTH UNITS
+2	Z, FLOATING POINT GEMINI LENGTH UNITS
+3	$\dot{X}$, FLOATING POINT GEMINI LENGTH UNITS

+4	$\dot{Y}$, FLOATING POINT GEMINI VELOCITY UNITS
+5	$\dot{Z}$, FLOATING POINT GEMINI VELOCITY UNITS

B) CONSTANTS -

KFS.VU K_0 CONVERTS FT/SEC TO GEMINI VELOCITY UNITS, +.38556042E-4

C) SUBROUTINES -

LIBRARY SUBROUTINES U3VMAG AND U3XPRO ARE USED.

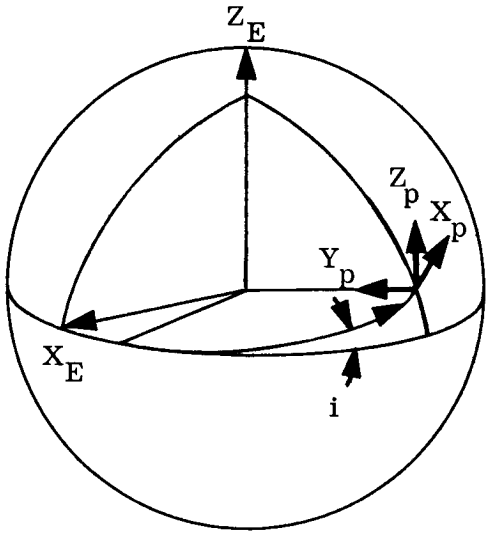
OUTPUT

A 3-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT

$L(\Delta\bar{v}_{acc})$	$\Delta\dot{X}_{acc}$	X COMPONENT OF THE VELOCITY INCREMENT, FLOATING POINT GEMINI VELOCITY UNITS
+1	$\Delta\dot{Y}_{acc}$	Y COMPONENT OF THE VELOCITY INCREMENT, FLOATING POINT GEMINI VELOCITY UNITS
+2	$\Delta\dot{Z}_{acc}$	Z COMPONENT OF THE VELOCITY INCREMENT, FLOATING POINT GEMINI VELOCITY UNITS

METHOD

THE INERTIAL PLATFORM COORDINATE SYSTEM IS DEFINED AS FOLLOWS -



$$\bar{X}_p = \bar{Y}_p \times \bar{Z}_p$$

$$\bar{Y}_p = -\bar{r}$$

$$\bar{Z}_p = \bar{r} \times \bar{v}$$

WHERE

$\bar{r}$ AND $\bar{v}$ DEFINE THE INERTIAL PLATFORM COORDINATE SYSTEM AT A CERTAIN INSTANT.

WHEN CONTROL IS TRANSFREERED TO RUCIVI, THE CONTENTS OF INDEX REGISTERS 1,2 AND 4 ARE SAVED.
NEXT, THE VECTORS $\bar{Y}_p$, $\bar{Z}_p$, and $\bar{X}_p$ ARE NORMALIZED, GIVING THE FOLLOWING UNIT VECTORS -

$$\bar{Y}_p^* = -\frac{\bar{r}}{|\bar{r}|} = \begin{pmatrix} Y_{pi} \\ Y_{pj} \\ Y_{pk} \end{pmatrix}$$

$$\bar{Z}_p^* = \frac{(\bar{r} \times \bar{v})}{|\bar{r} \times \bar{v}|} = \begin{pmatrix} Z_{pi} \\ Z_{pj} \\ Z_{pk} \end{pmatrix}$$

$$\bar{X}_p^* = \frac{(\bar{Y}_p^* \times \bar{Z}_p^*)}{|\bar{Y}_p^* \times \bar{Z}_p^*|} = \begin{pmatrix} X_{pi} \\ X_{pj} \\ X_{pk} \end{pmatrix}$$

THEN ΔV_{acc} IS COMPUTED, USING THE FOLLOWING EQUATIONS -

$$\Delta\dot{X}_{acc} = K_o [(\Delta\dot{X}_p)(X_{pi}) + (\Delta\dot{Y}_p)(Y_{pi}) + (\Delta\dot{Z}_p)(Z_{pi})]$$

$$\Delta\dot{Y}_{acc} = K_o [(\Delta\dot{X}_p)(X_{pj}) + (\Delta\dot{Y}_p)(Y_{pj}) + (\Delta\dot{Z}_p)(Z_{pj})]$$

$$\Delta\dot{Z}_{acc} = K_o [(\Delta\dot{X}_p)(X_{pk}) + (\Delta\dot{Y}_p)(Y_{pk}) + (\Delta\dot{Z}_p)(Z_{pk})]$$

FINALLY, THE CONTENTS OF INDEX REGISTERS 1,2, AND 4 ARE RESTORED AND CONTROL IS TRANSFERRED BACK TO MYREN2.

USAGE

A) CALLING SEQUENCE -

TSX	RUCIVI,4
PZE	$L(\Delta\bar{v}_{IVI})$, , $L(\Delta\bar{v}_{acc})$
PZE	L ($\bar{r}$, $\bar{v}$ AT DEFINITION OF PLATFORM)
ERROR	RETURN
NORMAL	RETURN

THIS IS A B-CORE PROGRAM WITH ALL STORAGE, CONSTANTS, AND SUBROUTINE LINKAGES IN A CORE, WHICH MEANS THE ECC IS NORMALLY SET TO A CORE AND ICC SET TO B CORE. THIS MINIMIZES THE NUMBER OF SEA AND SEB INSTRUCTIONS.

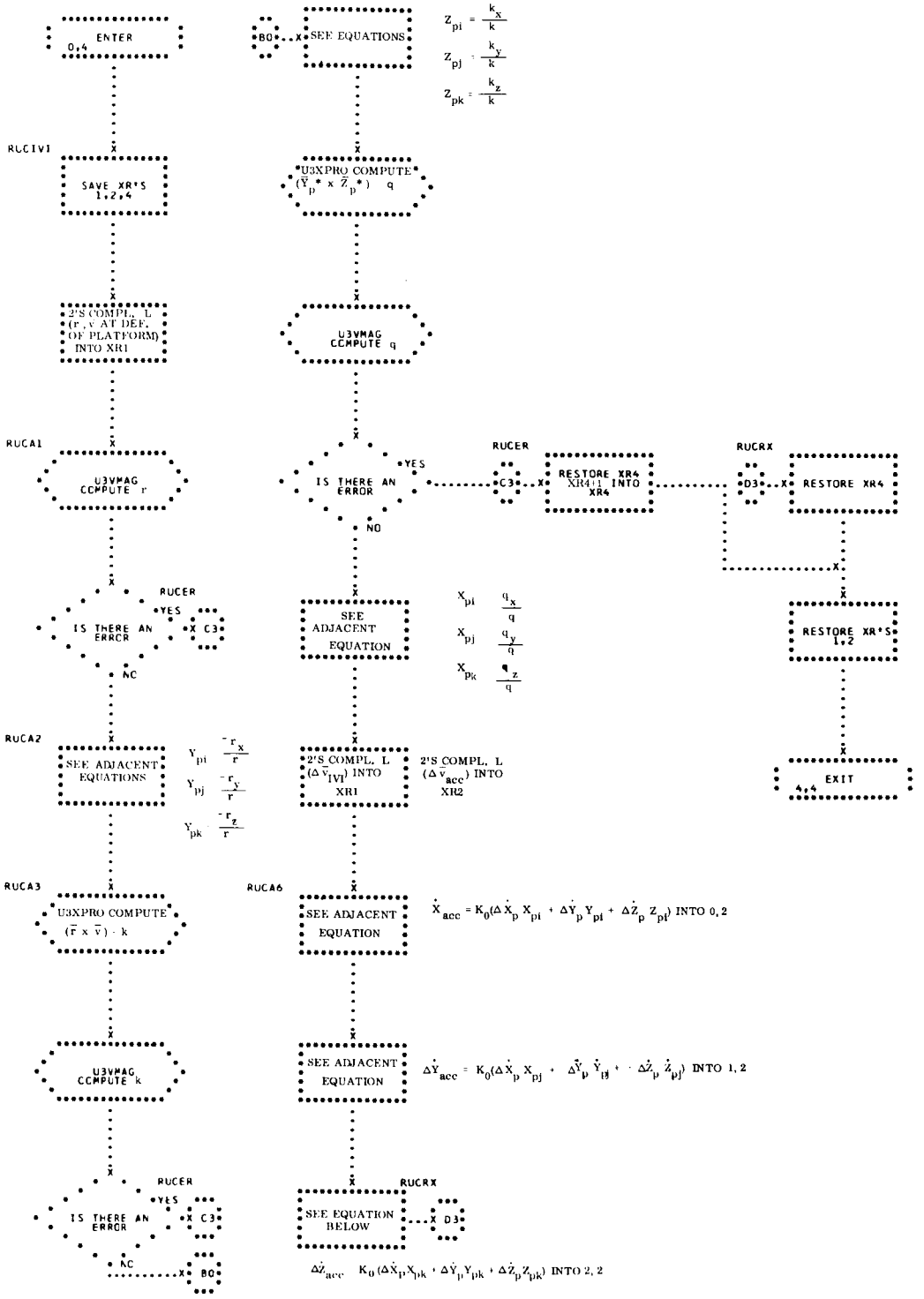
B) STORAGE REQUIRED (EXCLUDING SUBROUTINES U3VMAG AND U3XPRO) -

A CORE - 30 LOCATIONS

B CORE - 68 LOCATIONS

C) TIMING (7094) - 1.983 MS (INCLUDING TIME REQUIRED BY SUBROUTINES U3VMAG AND U3XPRO).

D) ERROR CONDITIONS - AN ERROR RETURN IS THE RESULT OF AN ERROR RETURN FROM LIBRARY SUBROUTINE U3VMAG.


ENTER 0,4
RUCIV1
SAVE XR*S 1,2,4
2'S COMPL. L (r, v AT DEE, OF PLATFORM) INTO XR1
RUCA1
U3VMAG COMPUTE r
RUCER
IS THERE AN ERROR
YES
C3
NO
RUCA2
SEE ADJACENT EQUATIONS
$Y_{pi} = \frac{-r_x}{r}$
$Y_{pj} = \frac{-r_y}{r}$
$Y_{pk} = \frac{-r_z}{r}$
RUCA3
U3XPRO COMPUTE $(\bar{r} \times \bar{v}) \cdot k$
U3VMAG COMPUTE k
RUCER
IS THERE AN ERROR
YES
C3
NO
B0
B0
SEE EQUATIONS
$Z_{pi} = \frac{k_x}{k}$
$Z_{pj} = \frac{k_y}{k}$
$Z_{pk} = \frac{k_z}{k}$
U3XPRO COMPUTE* $(\bar{Y}_p^* \times \bar{Z}_p^*)$ q
U3VMAG COMPUTE q
IS THERE AN ERROR
YES
C3
NO
RUCER
RESTORE XR4 XR4+1 INTO XR4
RUCRX
D3
RESTORE XR4
SEE ADJACENT EQUATION
$X_{pi} = \frac{q_x}{q}$
$X_{pj} = \frac{q_y}{q}$
$X_{pk} = \frac{q_z}{q}$
RESTORE XR*S 1,2
2'S COMPL. L $(\Delta \bar{v}_{IVI})$ INTO XR1
2'S COMPL. L $(\Delta \bar{v}_{acc})$ INTO XR2
EXIT 4,4
RUCA6
SEE ADJACENT EQUATION
$\dot{X}_{acc} = K_0(\Delta \dot{X}_p X_{pi} + \Delta \dot{Y}_p Y_{pi} + \Delta \dot{Z}_p Z_{pi})$ INTO 0, 2
SEE ADJACENT EQUATION
$\Delta \dot{Y}_{acc} = K_0(\Delta \dot{X}_p X_{pj} + \Delta \dot{Y}_p Y_{pj} + \Delta \dot{Z}_p Z_{pj})$ INTO 1, 2
RUCRX
SEE EQUATION BELOW
D3
$\Delta \dot{Z}_{acc} = K_0(\Delta \dot{X}_p X_{pk} + \Delta \dot{Y}_p Y_{pk} + \Delta \dot{Z}_p Z_{pk})$ INTO 2, 2

13. MYAPE3 - APOGEE-PERIGEE CONTROL PROCESSOR

INTEGRATED VALUES FOR APOGEE AND PERIGEE ARE COMPUTED UNDER THE PROGRAM CONTROL OF MYAPE3. AS USED THROUGHOUT THIS PAPER, APOGEE IS DEFINED AS THE MAXIMUM INERTIAL POSITION VECTOR OF THE ORBIT, AND PERIGEE AS THE MINIMUM INERTIAL POSITION VECTOR OF THE ORBIT. THE MAGNITUDES OF THE POSITION VECTORS ARE EXAMINED IN THE DERIVATION OF APOGEE-PERIGEE VALUES. THE FLOWCHART FOR MYAPE3 IS SHOWN IN FLOWCHART. PROGRAM LOGIC FLOW IS AS FOLLOWS -

USING THE INPUT $\vec{R}$, $\vec{V}$ AND TIME, THE INTEGRATOR COMPUTES AN INITIAL GUESS FOR BOTH APOGEE AND PERIGEE, THE GUESS CONTAINING AN UNCERTAINTY OF 2 MINUTES. SUBSEQUENT PASSES THROUGH THE INTEGRATOR YIELD FINAL VALUES ACCURATE TO THE NEAREST SECOND, THIS BEING DONE BY VARIANCE OF THE INTEGRATION TIME STEP. IN ALL PASSES, THE POSITION VECTOR MAGNITUDES GOVERN APOGEE-PERIGEE SELECTION.

PRINT BUFFERS ARE THEN CONSTRUCTED WHICH CONTAIN THE ORBITAL CHARACTERISTICS DERIVED FROM THE APOGEE AND PERIGEE VECTORS. MESSAGE QUEUES ARE GIVEN AND THE PROGRAM EXITS TO MOPRIO.

INPUT

B-BIT TURNED ON BY MYOUR1 AND MYMINS

QUEUE RECEIVED FROM MYOUR1

QUEUE WORD - A(LOC. OF INPUT VECTOR),,0

A) TABLES -

TMREFL	CONTAINS PRESENT ORBIT-DEFINING VECTOR
TMQKEY	CONTAINS QUEUE INFORMATION
TMWORK	CONTAINS INTEGRATOR INPUT

B) COMMUNICATION CELL -

MCMINS CONTAINS GMT IN FIXED MINUTES

OUTPUT

A) COMMUNICATION CELLS -

MCPER COMPUTED ORBITAL PERIOD IN FLOATING MINUTES

MCTAPG CONTAINS TIME OF APOGEE IN FIXED SECONDS

MCTPER CONTAINS TIME OF PERIGEE IN FIXED SECONDS

MCNIAC CONTAINS INTEGRATOR ERROR NUMBER

B) TABLES -

TMCLAS FIRST 7 LOCATIONS CONTAIN CLASSICAL ELEMENTS OF PRESENT ORBIT VECTOR

TMAPG3 CONTAINS ORBITAL CHARACTERISTICS BASED ON APOGEE VECTOR

TMPER3 CONTAINS ORBITAL CHARACTERISTICS BASED ON PERIGEE VECTOR

TMORMC CONTAINS ORBIT, REENTRY DISPLAY QUANTITIES

TPAPE3 CONTAINS MYAPE3 ERROR INFORMATION

C) QUEUES ARE GIVEN TO MYMESS FOR ON-LINE PRINTS.

METHOD

ENTRY TO MYAPE3 IS ACCOMPLISHED IN EITHER OF TWO WAYS, ONE IS TO HAVE THE B-BIT TURNED ON DIRECTLY BY ANOTHER PROGRAM AND THE OTHER IS TO HAVE THE B-BIT TURNED ON AS A RESULT OF AN ENTRY IN MYAPE3'S QUEUE TABLE. THE FIRST, OR NORMAL, ENTRY DIRECTS THE PROGRAM TO DETERMINE APOGEE AND PERIGEE BASED ON THE PRESENT ORBIT-DEFINING VECTOR, WHEREAS THE SECOND, OR QUEUED ENTRY, INDICATES AN

APOGEE-PERIGEE DEFINITION RESULTING FROM A PROPOSED S/C MANEUVER. VALUES CALCULATED FROM A QUEUED ENTRY ARE PRINTED ON-LINE FOR VISUAL INTERPRETATION, BUT ARE NOT FORMATTED FOR TRANSMISSION TO OTHER DISPLAY UNITS.

ONCE THE TYPE OF ENTRY HAS BEEN ESTABLISHED, AN INDICATOR IS SET (Z=NORMAL, NZ=QUEUED), AND THE ADDRESS OF THE INPUT VECTOR IS SAVED. OTHER INTEGRATOR USERS ARE H SUPPRESSED TO PERMIT NOCPNI TO ''PROCESS TO COMPLETION'' FOR EACH PARTICULAR ENTRY. A TIME TEST IS THEN MADE ON THE INPUT VECTOR, IF IT IS OLDER THAN 30 MINUTES FROM PRESENT TIME, AN ERROR EXIT IS TAKEN. IF THE VECTOR IS VALID AND A NORMAL ENTRY IS INDICATED, A TSX IS MADE TO PAODE (DESCRIBED LATER) TO CALCULATE THIS VECTOR'S ELEMENTS. THESE ELEMENTS ARE PLACED IN THE BUFFER, TMCLAS, FOR DISPLAY ON DOD MOD II UNIT 1 OF THE EMDS. THE FIRST 7 LOCATIONS OF THIS TABLE ARE FORMATTED AS FOLLOWS -

LOCATION	QUANTITY	UNITS
TMCLAS	TIME OF CLASSICAL ELEMENTS	FL. SECS.
+4	TRUE ANOMALY	FL. RAD.
+2	INCLINATION ANGLE	FL. RAD.
+3	SEMI-MAJOR AXIS	FL. N.M.
+4	ECCENTRICITY	
+5	ARGUMENT OF PERIGEE	FL. RAD.
+6	LONGITUDE OF ASCENDING NODE	FL. RAD.

•
•
•

PREPARATION OF THE INTEGRATOR INPUT TABLE TO OBTAIN THE FIRST-GUESS APOGEE-PERIGEE VECTORS IS NEXT FOR BOTH TYPES OF ENTRIES. THE TABLE (TMWORK) IS FORMATTED AS SUCH -

LOCATION	QUANTITY
TMWORK	X COMPONENT OF POSITION VECTOR
+1	Y COMPONENT OF POSITION VECTOR
+2	Z COMPONENT OF POSITION VECTOR
+3	X COMPONENT OF VELOCITY VECTOR
+4	Y COMPONENT OF VELOCITY VECTOR
+5	Z COMPONENT OF VELOCITY VECTOR
+6	TIME (FIXED SECS.)
+7	h = 2 MINUTES
+8	$\Delta t_B = 0$
+9	$\Delta t_f = 2*\tau$ WHERE - τ = ORBITAL PERIOD
+10	A (LOC. 15 WORD BUFFER),,0
+11	PZE 0,1,0
+12	S/C WEIGHT (FL. LBS.)
+13	VALIDITY TIME (FL. POINT)

THE ABOVE MENTIONED VALIDITY TIME (PRESENT GMT+7 MINUTES) IS A CONTROL FOR THE INTEGRATOR TO FACILITATE PROPER VECTOR SELECTION. A VECTOR COMPUTED PRIOR TO THIS TIME IS INVALID (IT DESCRIBES THE PRESENT APOGEE OR PERIGEE), THEREFORE THE ORBIT MUST BE INTEGRATED AHEAD FOR ANOTHER PERIOD.

WHEN THE INPUT TABLE IS COMPLETED, A QTSX IS MADE TO NOCPNI TO GET THE FIRST-GUESS VECTORS. OUTPUT IS LOCATED IN THE INTERNAL TABLE, PAFIG.

LOCATION	QUANITY	
PAFIG	X	
+1	Y	
+2	Z	
+3	$\dot{X}$	MAXIMUM $\vec{R}$
+4	$\dot{Y}$	
+5	$\dot{Z}$	
+6	TIME	

+7 X
+8 Y
+9 Z
+10 $\dot{X}$
+11 $\dot{Y}$
+12 $\dot{Z}$
+13 TIME

} MINIMUM $\vec{R}$

+14 NZ= ORBIT IS TOO CIRCULAR

A TYPE 9 INTEGRATION IS USED IN COMPUTING THE FIRST-GUESS VECTORS. THE INTEGRATOR SEARCHES AHEAD AT THE INPUT TIME INTERVAL (PRESENTLY 2 MINUTES) UNTIL IT ENCOUNTERS EITHER A MAXIMUM OR MINIMUM $/\vec{R}/$. THE FIRST VECTOR BEYOND THE STRADDLED $/\vec{R}/$ IS STORED IN THE OUTPUT BUFFER. IN SIMILAR FASHION, THE VECTOR FOR THE OTHER APSIS IS COMPUTED AND STORED. USING THIS DETERMINATION METHOD AND ASSUMING AN ORBIT SYMMETRIC TO THE LINES OF APSIDES, THE MAXIMUM UNCERTAINTY TIME SHOULD NOT EXCEED 3 MINUTES.

IF THE VECTOR FOR EITHER APSIS CANNOT BE CALCULATED (DUE TO VECTOR IMPACT), THE TIME IN THE OUTPUT BUFFER IS SET MINUS. THE FIFTEENTH LOCATION OF THE OUTPUT TABLE IS RESERVED AS AN ECCENTRICITY INDICATOR. IF, DURING THE APOGEE-PERIGEE SEARCH, THE MAGNITUDES OF THREE CONSECUTIVE VECTORS ARE EQUAL, THIS LOCATION WILL BE SET NON ZERO TO INDICATE THAT THE ORBIT IS TOO CIRCULAR TO DETERMINE THE APSIS POINTS.

IF A FIRST-GUESS APOGEE VECTOR IS AVAILABLE, THIS VECTOR IS TURNED OVER TO THE SUBROUTINE, PATTY, TO GET THE REFINED APOGEE VECTOR.

CALLING SEQUENCE -

```
     TSX PATTY,4
+1   PZE (INPUT),,0
+2   TRA ERROR RETURN
```

+3 NORMAL RETURN

IF THE SECOND WORD OF THE CALLING SEQUENCE IS PZE, THE INPUT IS THE APOGEE VECTOR, IF MZE, THE INPUT IS THE PERIGEE VECTOR. OUTPUT IS THE REQUESTED VECTOR LOCATED IN TMWORK.

TMWORK, THE INTEGRATOR INPUT TABLE, IS CONSTRUCTED FOR A TYPE 0 INTEGRATION. THE REFINEMENT PROCESS REQURES FIVE PASSES THROUGH THE INTEGRATOR TO OBTAIN AN APSIS VECTOR OF 1-SECOND ACCURACY. A 2-POWER METHOD OF SELECTING TIME STEP, STORE CYCLE AND TOTAL DELTA INTERVAL IS USED FOR THESE 5 PASSES.

	TOTAL DELTA-INTERVAL TDI	STORE CYCLE SC	TIME STEP TS
PASS 1	1024	16	16
PASS 2	245	8	8
PASS 3	64	4	4
PASS 4	16	2	2
PASS 5	4	1	1

NOTE - ALL TIMES ARE GIVEN IN SECONDS.

AFTER EACH INTEGRATOR PASS, THE OUTPUT TABLE OF 5 VECTORS IS INTERROGATED TO DETERMINE EITHER THE MINIMUM OR THE MAXIMUM POSITION VECTOR, DEPENDING ON ENTRY TYPE. THIS VECTOR IS THEN USED AS INPUT FOR THE NEXT PASS. ON THE FIFTH AND FINAL PASS, THE VECTOR FROM THE OUTPUT TABLE IS STORED INTO TMWORK AS THE SUBROUTINE OUTPUT.

WITH A NORMAL RETURN FROM PATTY, THE APOGEE VECTOR IS GIVEN TO PATSU. THIS SUBROUTINE GENERATES THE PRINT BUFFER OF ORBITAL CHARACTERISTICS DERIVED FROM THE APOGEE VECTOR (TMAPG3).

LOCATION	QUANTITY	UNITS
TMAPG3	TIME OF APOGEE	FX. SECS.
+1	HEIGHT OF APOGEE	FL. N.M.

+2	SEMI-MAJOR AXIS	FL. N.M.
+3	ORBITAL PERIOD	FL. MINS.
+4	ECCENTRICITY	
+5	LONGITUDE OF APOGEE	FL. DEG.
+6	LATITUDE OF APOGEE	FL. DEG.
+7	INCLINATION ANGLE	FL. DEG.
+8	LONGITUDE ASCENDING NODE	FL. DEG.
+9	ARGUMENT OF LATITUDE	FL. DEG.
+10	ARGUMENT OF PERIGEE	FL. DEG.
+11	MEAN ANOMALY	FL. DEG.
+12	ECCENTRIC ANOMALY	FL. DEG.
+13	TRUE ANOMALY	FL. DEG.

MYMESS IS THEN GIVEN A QUEUE TO PRINT THIS BLOCK OF DATA ON-LINE.

PROGRAM LOGIC IS NOW EXECUTED TO FIND THE REFINED PERIGEE VECTOR. WITH THE FIRST-GUESS VECTOR, IF AVAILABLE, A TSX IS MADE TO PATTY. USING THE METHOD PREVIOUSLY EXPLAINED, A REFINED PERIGEE VECTOR BECOMES THE SUBROUTINE OUTPUT.

PATSU TAKES THE VECTOR AND GENERATES THE PRINT BUFFER OF ORBITAL CHARACTERISTICS DERIVED FROM THE PERIGEE VECTOR (TMPER3).

LOCATION	QUANTITY	UNITS
TMPER3	TIME OF PERIGEE	FX. SECS.
+1	HEIGHT OF PERIGEE	FL. N. M.
+2	SEMI-MAJOR AXIS	FL. N. M.
+3	ORBITAL PERIOD	FL. MINS.
+4	ECCENTRICITY	
+5	LONGITUDE OF PERIGEE	FL. DEG.
+6	LATITUDE OF PERIGEE	FL. DEG.
+7	INCLINATION ANGLE	FL. DEG.
+8	LONGITUDE ASCENDING NODE	FL. DEG.

+9	ARGUMENT OF LATITUDE	FL. DEG.
+10	ARGUMENT OF PERIGEE	FL. DEG.
+11	MEAN ANOMALY	FL. DEG.
+12	ECCENTRIC ANOMALY	FL. DEG.
+13	TRUE ANOMALY	FL. DEG.

A QUEUE IS GIVEN MYMESS TO PRINT ON-LINE THIS DATA BUFFER. NOCPNI USERS HAVE THEIR H BITS TURNED OFF AND MYAPE3 EXITS TO MOPRIO.

PAODE IS A SUBROUTINE USED TO CALCULATE ORBITAL ELEMENTS FROM AN INPUT $\vec{R}$, $\vec{V}$ AND TIME.

CALLING SEQUENCE -

```
        TSX     PAODE,4
+1      PZE     (INPUT),,(OUTPUT)
+2      PZE     (R VECTOR),,(V VECTOR)
+3      TRA     ERROR RETURN
+4              NORMAL RETURN
```

TMCLAS, TMAPG3, AND TMPER3 ARE ALL BUILT USING THE QUANTITIES COMPUTED BY PAODE. THE OUTPUT TABLE FORMAT IS AS FOLLOWS -

LOCATION	QUANTITY	UNITS
OUTPUT	SEMI-MAJOR AXIS	FL. N. M.
+1	ORBITAL PERIOD	FL. MINS.
+2	ECCENTRICITY	
+3	INCLINATION ANGLE	FL. RAD.
+4	LONGITUDE ASCENDING NODE	FL. RAD.
+5	ARGUMENT OF LATITUDE	FL. RAD.
+6	ARGUMENT OF PERIGEE	FL. RAD.
+7	MEAN ANOMALY	FL. RAD.
+8	ECCENTRIC ANOMALY	FL. RAD.
+9	TRUE ANOMALY	FL. RAD.
+10	TIME SINCE PERIGEE	FIXED SECS.
+11	TIME OF NEXT PERIGEE	FIXED SECS.

+12 TIME OF NEXT APOGEE FIXED SECS.

EQUATIONS USED BY PAODE

$$|R|^2 = X^2 + Y^2 + Z^2$$

$$|V|^2 = \dot{X}^2 + \dot{Y}^2 + \dot{Z}^2$$

$$\vec{R} \cdot \vec{V} = X\dot{X} + Y\dot{Y} + Z\dot{Z}$$

1. SEMI-MAJOR AXIS (A)

$$a = \frac{|\vec{R}|}{1 - e \cos E}$$

WHERE

$$e \cos E = |\vec{R}|\,|\vec{V}|^2 - 1$$

2. ORBITAL PERIOD(τ)

$$\tau = k_1\, 2\pi a^{3/2}$$

WHERE

k_1 = CONVERSION FACTOR (GTU/MIN)

3. ECCENTRICITY (E)

$$e = \sqrt{(e \sin E)^2 + (e \cos E)^2}$$

WHERE

$$e \sin E = \frac{\vec{R} \cdot \vec{V}}{\sqrt{a}}$$

4. ECCENTRIC ANOMALY (E)

$$E = \tan^{-1}\left(\frac{e \sin E}{e \cos E}\right)$$

WHERE

5. MEAN ANOMALY (M)

$$M = E - e \sin E$$

6. TIME SINCE PERIGEE ($\tilde{T}$)

$$\tilde{T} = k_1 \, Ma^{3/2}$$

7. TIME OF PERIGEE (T_p)

$$T_p = t_0 + \tau - T$$

WHERE

t_0 = present time

8. TIME OF APOGEE (T_a)

$$T_a = T_p \pm \frac{\tau}{2}$$

9. INCLINATION ANGLE (i)

$$i = \sin^{-1}\left(\frac{\sqrt{h_x^2 + h_y^2}}{|\vec{h}|}\right)$$

WHERE-

$$\vec{h} = \vec{R} \times \vec{V}$$

10. LONGITUDE OF ASCENDING NODE (Ω)

$$\Omega = \tan^{-1}\left(\frac{h_x}{-h_y}\right)$$

11. TRUE ANOMALY (ν)

$$\nu = \tan^{-1}\left(\frac{e \sin E \sqrt{1 - e^2}}{e \cos E - e^2}\right)$$

12. ARGUMENT OF LATITUDE (μ)

$$u = \tan^{-1}\left(\frac{Z|\vec{h}|}{Yh_x - Xh_y}\right)$$

13. ARGUMENT OF PERIGEE (ω)

$$\omega = u - \nu$$

MYAPE3 ERROR CODES -

55 - INPUT VECTOR OLDER THAN 30 MINUTES
70 - LIBRARY SUBROUTINE ERROR
75 - ORBIT IS TOO CIRCULAR
85 - INPUT VECTOR IMPACTED
$\geq$ 100 - INTEGRATOR ERROR

USAGE -

A) ENTRY FROM AND EXIT TO MOPRIO

B) COMMUNICATIONS CELLS -

MCMINS MCTPER
MCPER. MCNIAC
MCTAPG

C) TABLES -

TMNMSK TMREFL TMPER3
TMFMSK TMWORK TMORMC
TMQKEY TMCLAS TPAPE3
TMPRIO TMAPG3

D) CONSTANTS -

K00030	K00002
KCH233	K001.0
K00007	K00.MH
K00006	K0060.
K00060	K002.0
KM7777	KRTDEG
KCGUNM	K00001
KKK2PI	K00085

E) MACROS -

TRNON	QUEUE
TRNOF	UNQUE
QENBZ	QTSX
QENBA	

F) PARAMETERS -

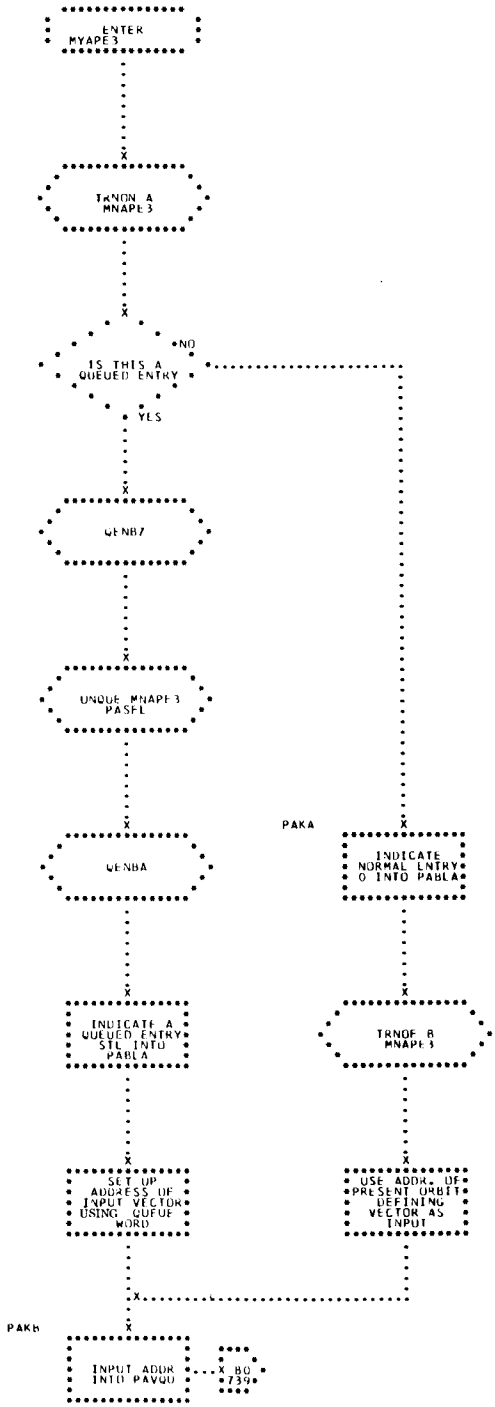
A,B,H	MNMUV7
MNCOTO	MNLIF8
MNCUR1	MNDIC9
MNREN2	MNFPPR
MNAPE3	MNCPNI
MNTTF4	MNMESS
MNNAV5	MNBBNI
MNGMT6	

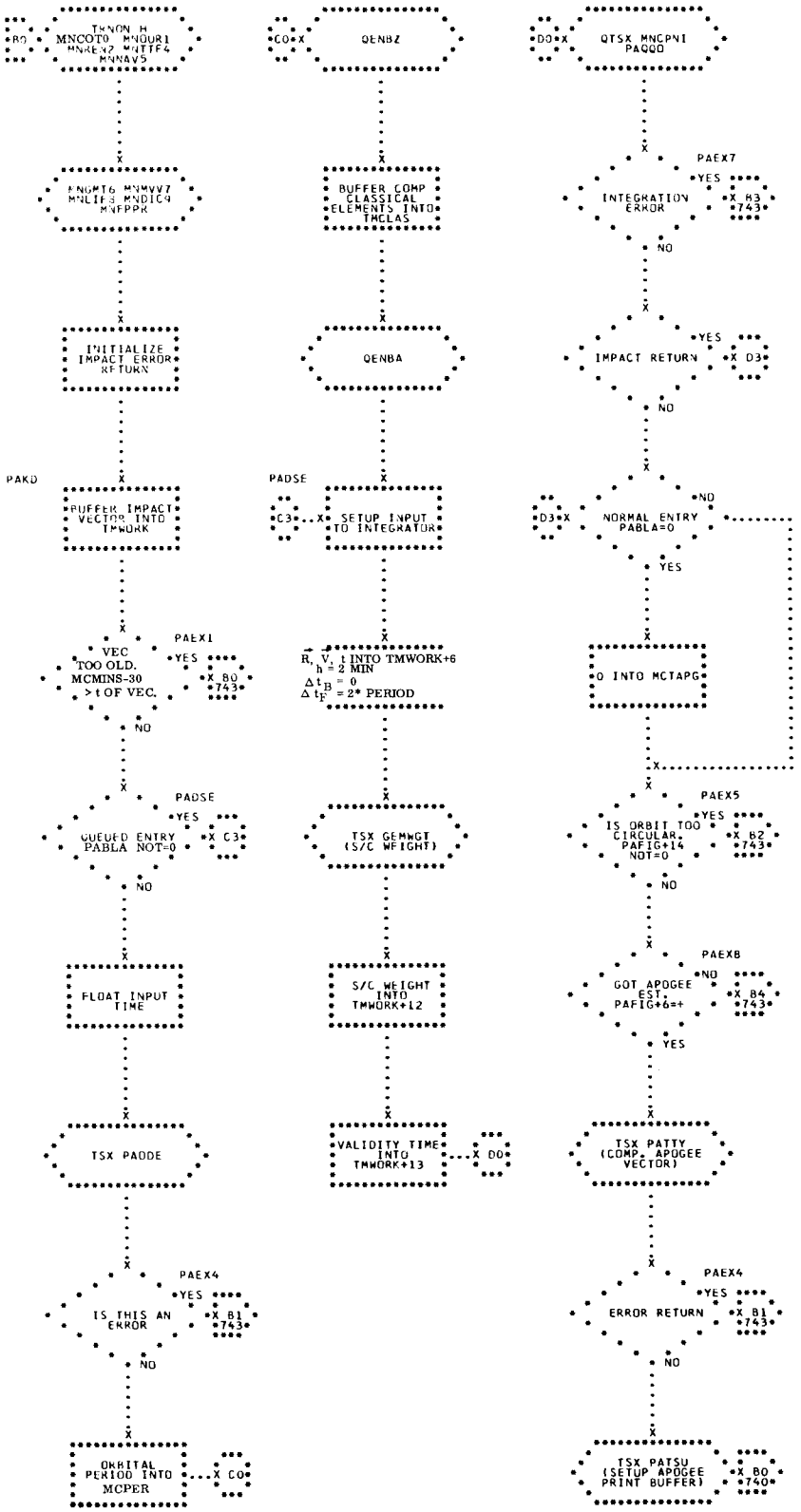
G) SUBROUTINE -

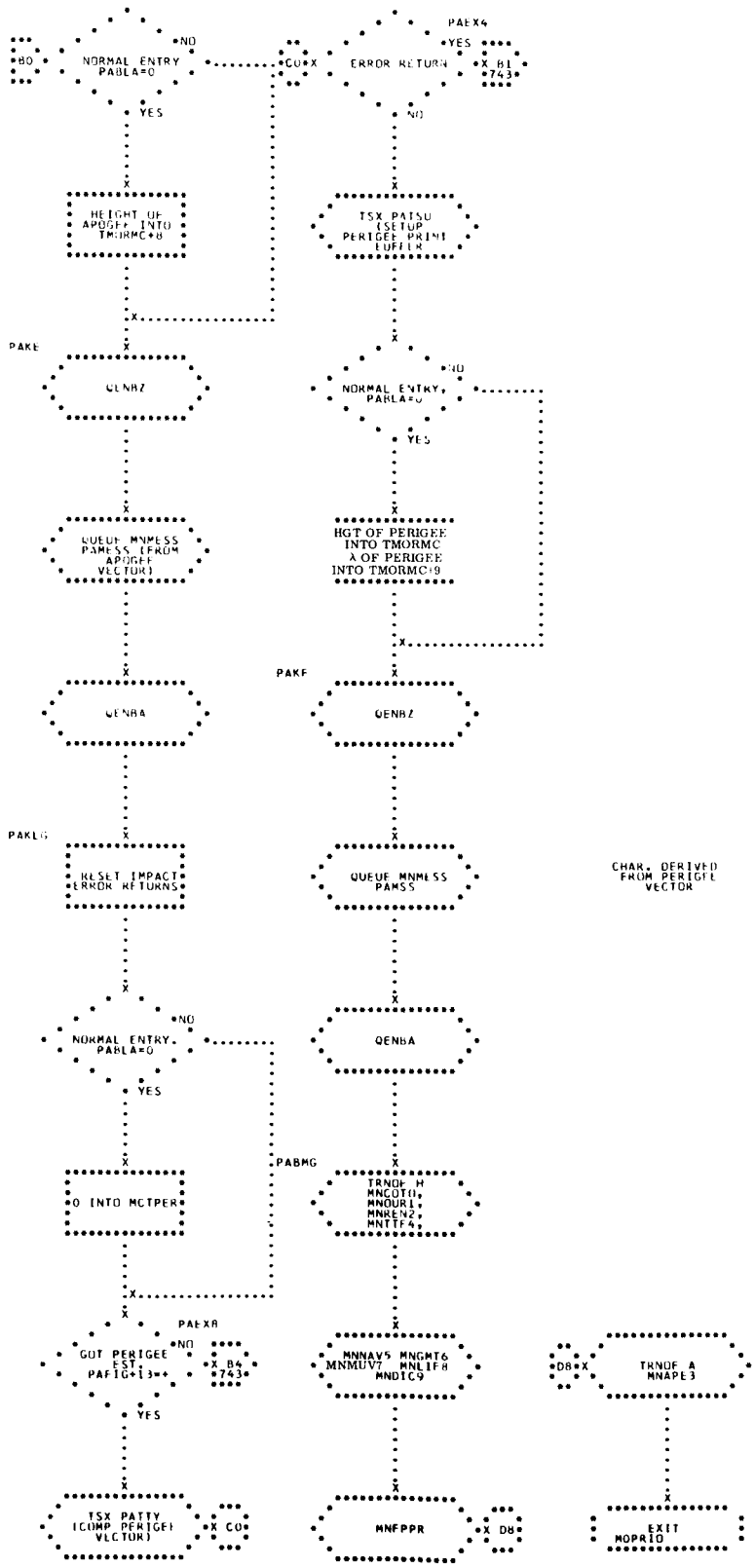
VMAG	U3XPR
SQRT	ARSIN
U3DOT	A3MSCP
ATNAB	GEMWGT

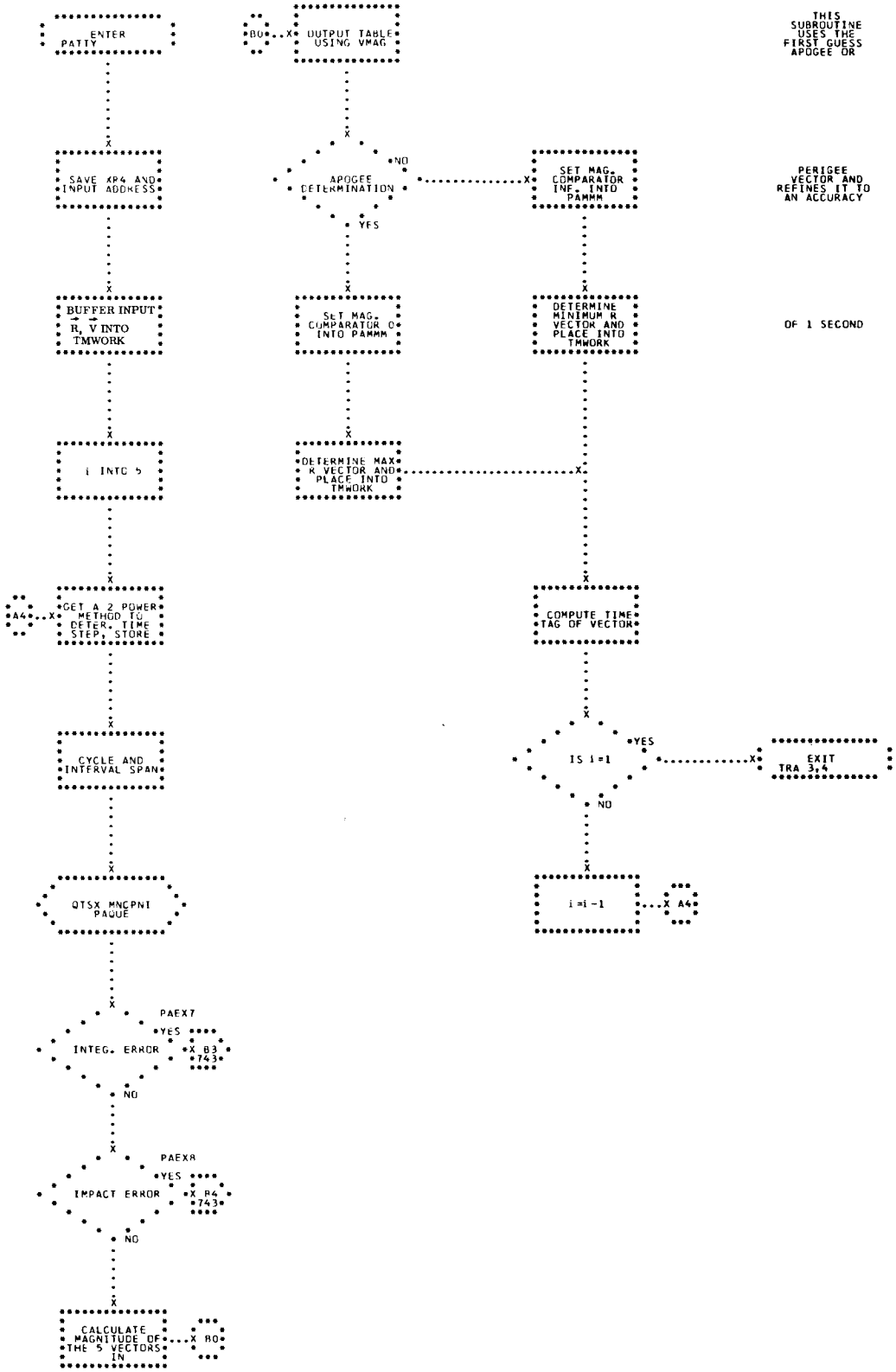
H) STORAGE REQUIRED -

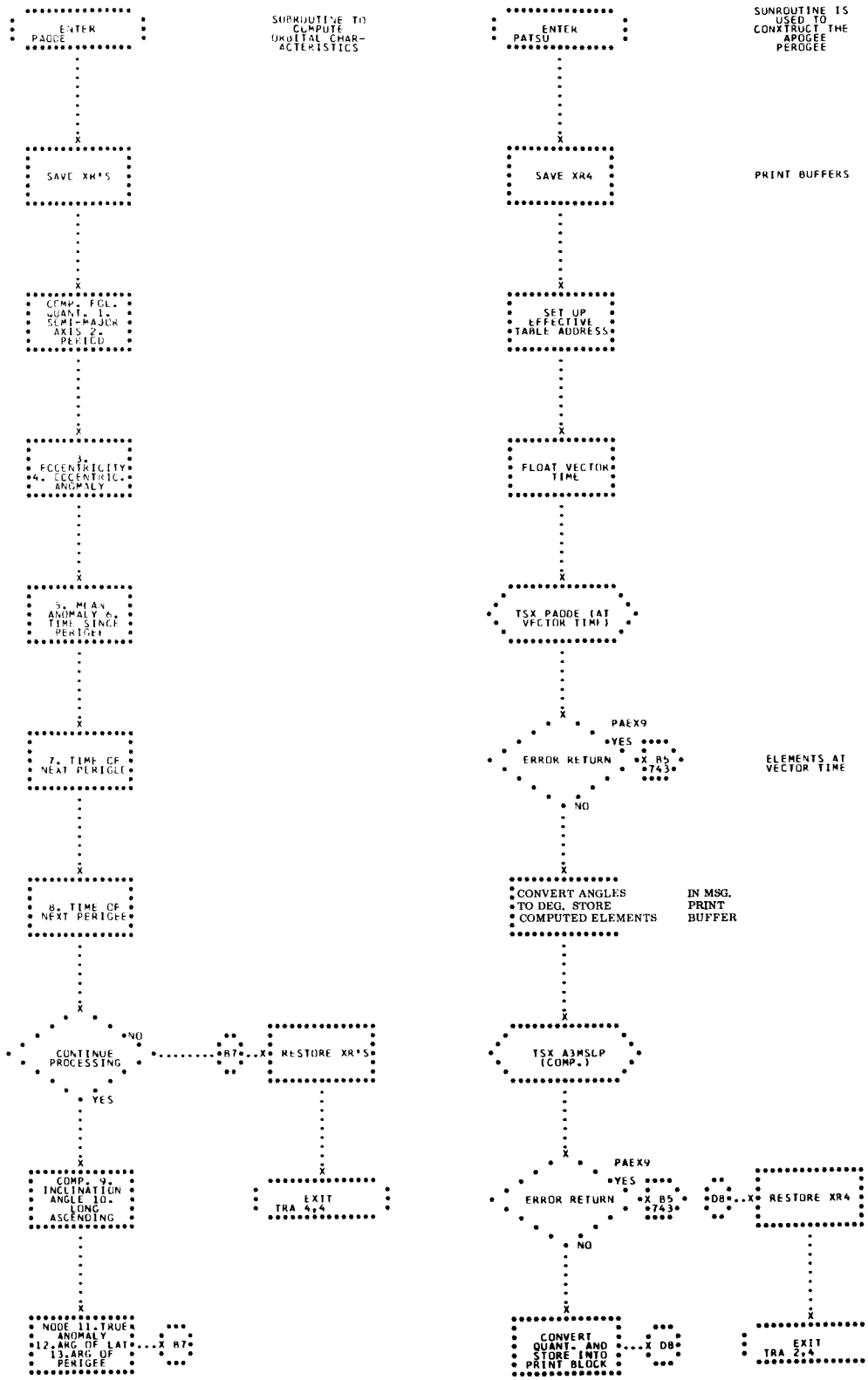
600 LOCATIONS

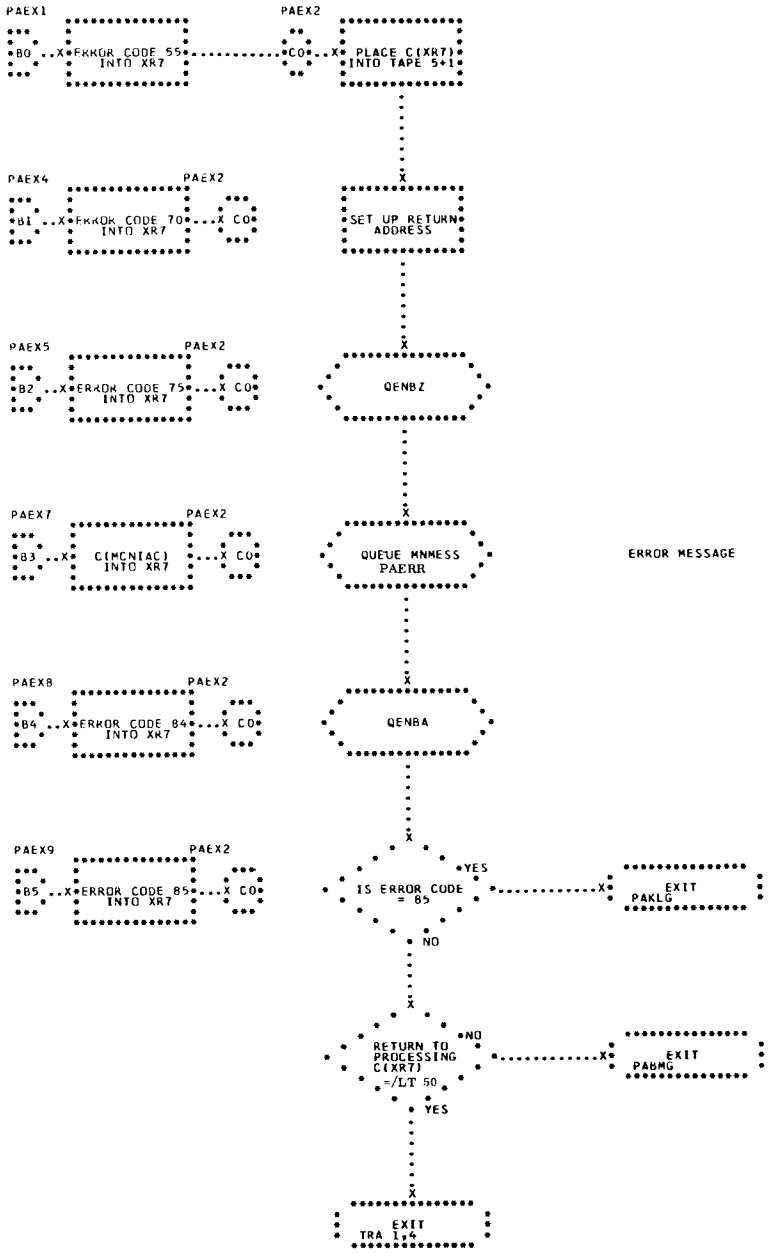

ENTER MYAPE3
TRNON A MNAPE3
IS THIS A QUEUED ENTRY
NO
YES
GENB7
UNQUE MNAPE3 PASEL
GENBA
INDICATE A QUEUED ENTRY STL INTO PABLA
SET UP ADDRESS OF INPUT VECTOR USING QUEUE WORD
PAKA
INDICATE NORMAL ENTRY 0 INTO PABLA
TRNOF B MNAPE3
USE ADDR. OF PRESENT ORBIT DEFINING VECTOR AS INPUT
PAKB
INPUT ADDR INTO PAVQU
X BO 739


TSXON H
MNCOTO MNOUR1
MNXEX2 MNTTF4
MNNAV5
B0
ANOMT6 MNMVV7
MNLIF3 MNDIC4
MVFPPR
INITIALIZE IMPACT ERROR RETURN
PAKD
BUFFER IMPACT VECTOR INTO TMWORK
PAEX1
VEC TOO OLD. MCMINS-30 >t OF VEC.
YES
X B0 *743*
NO
PADSE
QUEUED ENTRY PABLA NOT=0
YES
X C3
NO
FLOAT INPUT TIME
TSX PADDE
PAEX4
IS THIS AN ERROR
YES
X B1 *743*
NO
ORBITAL PERIOD INTO MCPER
X C0
C0
QENBZ
BUFFER COMP CLASSICAL ELEMENTS INTO TMCLAS
QENBA
PADSE
C3
SETUP INPUT TO INTEGRATOR
R, V, t INTO TMWORK+6
h = 2 MIN
Δt_B = 0
Δt_F = 2* PERIOD
TSX GEMWGT (S/C WEIGHT)
S/C WEIGHT INTO TMWORK+12
VALIDITY TIME INTO TMWORK+13
X D0
D0
QTSX MNCPNT PAQQQ
PAEX7
INTEGRATION ERROR
YES
X B3 *743*
NO
IMPACT RETURN
YES
X D3
NO
D3
NORMAL ENTRY PABLA=0
NO
YES
0 INTO MCTAPG
PAEX5
IS ORBIT TOO CIRCULAR. PAFIG+14 NOT=0
YES
X B2 *743*
NO
PAEXB
GOT APOGEE EST. PAFIG+6=+
NO
X B4 *743*
YES
TSX PATTY (COMP. APOGEE VECTOR)
PAEX4
ERROR RETURN
YES
X B1 *743*
NO
TSX PATSU (SETUP APOGEE PRINT BUFFER)
X B0 *740*


NORMAL ENTRY PABLA=0
NO
YES
BO
CU*X
ERROR RETURN
PAEX4
YES
NO
X B1 *743*
HEIGHT OF APOGEE INTO TMORMC*8
TSX PAISU (SETUP PERIGEE PRINT BUFFER
PAKE
QENBZ
NORMAL ENTRY, PABLA=0
NO
YES
QUEUE MNMESS PAMESS (FROM APOGEE VECTOR)
HGT OF PERIGEE INTO TMORMC A OF PERIGEE INTO TMORMC+9
PAKF
QENBA
QENBZ
PAKLG
RESET IMPACT ERROR RETURNS
QUEUE MNMESS PAMSS
CHAR. DERIVED FROM PERIGEE VECTOR
NORMAL ENTRY, PABLA=0
NO
YES
QENBA
PABMG
0 INTO MCTPER
TRNOF H MNCOT0, MNOUR1, MNREN2, MNTTF4
PAEXB
GOT PERIGEE EST. PAFIG+13=+
NO
X B4 *743*
YES
MNNAV5 MNGMT6 MNMUV7 MNL1F8 MNDIC9
D8*X
TRNOF A MNAPE3
TSX PATTY (COMP PERIGEE VECTOR)
X C0
MNFPPR
X D8
EXIT MOPRIO


ENTER PATTY
SAVE XP4 AND INPUT ADDRESS
BUFFER INPUT R, V INTO TMWORK
i INTO 5
GET A 2 POWER METHOD TO DETER. TIME STEP, STORE
CYCLE AND INTERVAL SPAN
QTSX MNCPNI PAQUE
INTEG. ERROR
YES
PAEX7
X B3 *743*
NO
IMPACT ERROR
YES
PAEXR
X B4 *743*
NO
CALCULATE MAGNITUDE OF THE 5 VECTORS IN
X B0
B0
OUTPUT TABLE USING VMAG
APOGEE DETERMINATION
NO
YES
SET MAG. COMPARATOR 0 INTO PAMMM
DETERMINE MAX R VECTOR AND PLACE INTO TMWORK
SET MAG. COMPARATOR INE. INTO PAMMM
DETERMINE MINIMUM R VECTOR AND PLACE INTO TMWORK
COMPUTE TIME TAG OF VECTOR
IS i=1
YES
EXIT TRA 3,4
NO
i=i-1
X A4
A4
THIS SUBROUTINE USES THE FIRST GUESS APOGEE OR
PERIGEE VECTOR AND REFINES IT TO AN ACCURACY
OF 1 SECOND


ENTER PADDE
SUBROUTINE TO COMPUTE ORBITAL CHARACTERISTICS
SAVE XR'S
COMP. FOL. QUANT. 1. SEMI-MAJOR AXIS 2. PERIOD
3. ECCENTRICITY 4. ECCENTRIC. ANOMALY
5. MEAN ANOMALY 6. TIME SINCE PERIGEE
7. TIME OF NEAT PERIGEE
8. TIME OF NEXT PERIGEE
CONTINUE PROCESSING
NO
YES
B7
RESTORE XR'S
EXIT TRA 4,4
COMP. 9. INCLINATION ANGLE 10. LONG ASCENDING
NODE 11. TRUE ANOMALY 12. ARG OF LAT 13. ARG OF PERIGEE
X B7
ENTER PATSU
SUNROUTINE IS USED TO CONXTRUCT THE APOGEE PERDGEE
SAVE XR4
PRINT BUFFERS
SET UP EFFECTIVE TABLE ADDRESS
FLOAT VECTOR TIME
TSX PADDE (AT VECTOR TIME)
PAEX9
YES
ERROR RETURN
X B5 +743
NO
ELEMENTS AT VECTOR TIME
CONVERT ANGLES TO DEG. STORE COMPUTED ELEMENTS
IN MSG. PRINT BUFFER
TSX A3MSLP (COMP.)
PAEX9
YES
ERROR RETURN
X B5 +743
NO
D8
RESTORE XR4
EXIT TRA 2,4
CONVERT QUANT. AND STORE INTO PRINT BLOCK
X D8


PAEX1
B0
ERROR CODE 55 INTO XR7
PAEX2
C0
PLACE C(XR7) INTO TAPE 5+1
PAEX4
B1
ERROR CODE 70 INTO XR7
PAEX2
C0
SET UP RETURN ADDRESS
PAEX5
B2
ERROR CODE 75 INTO XR7
PAEX2
C0
QENBZ
PAEX7
B3
C(MCNIAC) INTO XR7
PAEX2
C0
QUEUE MNMESS PAERR
ERROR MESSAGE
PAEX8
B4
ERROR CODE 84 INTO XR7
PAEX2
C0
QENBA
PAEX9
B5
ERROR CODE 85 INTO XR7
PAEX2
C0
IS ERROR CODE = 85
YES
EXIT PAKLG
NO
RETURN TO PROCESSING C(XR7) =/LT 50
NO
EXIT PABMG
YES
EXIT TRA 1,4

14. MYTTF4 - TIME-TO-FIRE CONTROL PROCESSOR

MYTTF4, USING SUBROUTINES R6ESTF AND R5RARF COMPUTES GMTRC (GREENWICH-MEAN-TIME OF RETROFIRE COMPUTED), GMTRB (GREENWICH-MEAN-TIME TO REVERSE BANK ANGLE) AND BANK ANGLE, PLUS OTHER ASSOCIATED QUANTITIES BASED ON VARIOUS MANEUVER REFLECTED VECTORS, AND STORES FOR DISPLAY AND ON-LINE PRINT THE APPROPRIATE VALID RESULTS. (SEE FLOWCHARTS)

INPUT

A) QUEUE WORD - PREFIX CONTAINS AN INDICATION OF THE TYPE OF RECOVERY AREA -

+0	STORED PRIMARY AREA, SPA
-0	STORED CONTIGENCY AREA, SCA
+1	MANUAL PRIMARY AREA, MPA
-1	MANUAL CONTINGENCY AREA, MCA

DECREMENT CONTAINS THE LANDING REVOLUTION NUMBER (FIXED POINT INTEGER) OF THE RECOVERY AREA (TARGET).

ADDRESS CONTAINS THE LOCATION OF A 4-WORD TABLE CONTAINING THE FOLLOWING FLOATING POINT QUANTITIES IN THIS ORDER -

λ_T	LONGITUDE OF THE TARGET (POSITION RADIANS)
ϕ_T	LATITUDE OF THE TARGET (RADIANS)
B	BANK ANGLE TO BE USED (DEGREE)
Δ_V	DELTA VELOCITY TO BE USED IN THE PRE-RETRO OAMS BURN. (FT/SEC)

B) TABLES -

TMREFL- MANEUVER-REFLECTED VECTOR TABLE.

TABLE OF VECTORS IN THE FOLLOWING FORMAT -

x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$	COMPONENTS OF POSITION AND VELOCITY VECTORS IN FLOATING POINT GEMINI UNITS
t	TIME TAG OF THE VECTOR IN FIXED POINT SECONDS
APRN	ANCHOR POINT REVOLUTION NUMBER OF THE VECTOR

THE FIRST VECTOR IN THIS TABLE DEFINES THE LATEST ORBIT IN THE COMPUTER. SUBSEQUENT VECTORS DEFINE THOSE ORBITS OBTAINED FROM INTERGRATING THROUGH ALL MANEUVERS IN THE MANEUVER TABLE.

TMANUV MANEUVER TABLE. CONTAINS PARAMETER THAT DEFINE ALL PLANNED MANEUVERS FROM CURRENT GMT TO 22 ORBITS AHEAD, IN THIS FORMAT. GMTPS (GMT OF PROPULSION SETTING), GMTPC (GMT OF PROPULSION COMPLETION), T (THRUSTER INDICATOR, 0 FORE, 1 AFT) IN DECREMENT AND Δt (DURATION OF BURN) IN ADDRESS, PITCH ANGLE, YAW ANGLE. ALL TIMES FIXED PT. SECS, AND ALL ANGLES FLOATING PT. RADIANS.

TMCAPE 162-LOCATION TABLE ALLOWING AT MOST 23 CAPE KENNEDY VECTORS. THE FIRST WORD CONTAINS THE ORBIT NUMBER OF THE FIRST VECTOR IN ITS ADDRESS AND THE ORBIT NUMBER OF THE LAST VECTOR IN ITS DECREMENT. VECTORS ARE STORED x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$, t.

TMETTF 1-WORD TABLE CONTAINING THE ESTIMATED TIME-TO-FIRE RETROROCKETS, OUTPUT OF R6ESTF, IN FIXED PT. SECS.

TMRORF OUTPUT TABLE OF R5RARF, CONTAINING GMTRC, GMTRB, B, GMTIR (GMT TO INITIATE ROLL), B1 (BANK ANGLE ASSOCIATED WITH GMTIR), ZERO LIFT AND FULL LIFT IMPACT POINTS, AND MISSED DISTANCE.

TNREEP TABLE CONTAINING TIMES AT BEGINNING AND END OF BLACKOUT, VELOCITY AND FLIGHT-PATH ANGLE AT 350,000 FT., GMT AND SUB-CAPSULE LATITUDE AND LONGITUDE AT 400,000 FT.

TMIMPP TABLE CONTAINING TIME, LATITUDE AND LONGITUDE AT 10,000 FT. SET UP BY THE INTEGRATOR.

TMBLST TABLE SET UP BY R5RARF AND CONTAINING IN ITS FIRST 6 LOCATIONS, THE x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$ AT GMTRC.

TOSCCH S/C CHARACTERISTIC TABLE. SECOND WORD CONTAINS THE THRUST LEVEL, FOURTH WORD MASS EXPULSION RATE, EIGHTH WORD THE CANT ANGLE OF ALL THE AFT THRUSTERS, AND THE THIRTEENTH WORD CONTAINS THE TIME FROM 10,000 FT. TO IMPACT.

C) COMMUNICATION CELLS -

MCMANC CONTAINS NUMBER OF MANEUVERS IN MANEUVER TABLE.

MCGTLO CONTAINS FIXED PT. TIME OF LIFT-OFF.

MCUNCR SUM OF THE UNCERTAINTY IN TIME TWO GMTRC IS POSSIBLE FOR SAME TARGET PLUS UNCERTAINTY IN ESTIMATED TIME-TO-FIRE.

MCUNC2 SUM OF UNCERTAINTY IN ESTIMATED TIME-TO-FIRE PLUS VALUE FOR INTERPOLATION.

MCUNC3 UNCERTAINTY IN TIME TWO GMTRC'S POSSIBLE.

MCVPER VACUUM PERIGEE OF FIRST VECTOR IN TABLE TMREFL, IN NAUTICAL MILES.

MCHFSC CURRENT GMT, IN FIXED PT. HALF-SECONDS.

MCMINS CURRENT GMT, IN FIXED PT. MINUTES.

KAVGDP AVERAGE TIME BETWEEN PRIMARY AREAS.

KAVGDT AVERAGE TIME BETWEEN CONTINGENCY AREAS.

D) CONSTANTS -

K00008	K0180.
K00001	K00360
K00000	K000P3
K00060	K00024
K080.0	K00016
K00007	K00002
KCH233	K86400
K00005	K.82GU
K00009	K00003
K00030	K00105
KRTDEG	K002.0

E) SYSTEM SUBROUTINES -

R5RARF COMPUTES GMTRC, GMTRB, B, GMTIR, B1

R6ESTF COMPUTES AN ESTIMATED GMTRC TO BIT THE DESIRED TARGET.

BEMWGT COMPUTES S/C WEIGHT

NOCPNI INTEGRATOR IS USED AS A SUBROUTINE BY USING QTSX.

F) LIBRARY SUBROUTINES -

BBBCOS

BBBSIN

BATNAB

G) MACROS -

QTSX

QENBZ

QENBA

QUEUE

UNQUE

TRNON

TRNOF

OUTPUT

A) TABLES -

TMORMC A 33-WORD TABLE CONTAINING ORBIT AND RE-ENTRY OUTPUT QUANTITIES FOR THE DIGITAL DISPLAYS AND PLOTBOARDS. DURING ORBIT, THE 19TH THROUGH 25TH WORDS ARE SET UP BY MYTTF4. (SEE O5ORMC OR OOORRE FOR THE FORMAT.)

TMYMES A 27-WORD TABLE OF INPUT TO THE ON-LINE MESSAGE ROUTINE (FORMAT IN MYMESS WRITE-UP.)

TMCPVC A 17-WORD INPUT TABLE TO THE INTEGRATOR FOR CAPE VECTORS. FORMAT SAME AS TMBBNI. (SEE MPCPNI.)

TMGTRC A 17-WORD INPUT TABLE TO THE INTEGRATOR FOR TTF ORBIT TABLE. FORMAT SAME AS TMBBNI.

TDCAPR DCS INPUT BLOCK FOR THE AUTOMATIC PRE-RETRO LOAD, IN THIS ORDER - X,Y,Z,$\dot{X}$,$\dot{Y}$,$\dot{Z}$. GMTRC, GMTRB, B1, ϕ_T, λ_T.

TDCMPR DCS INPUT BLOCK FOR THE MANUAL PRE-RETRO LOAD, SAME FORMAT AS TDCAPR.

TMRCWT ONE WORD CONTAINING ORBIT WEIGHT FOR R5RARF IN FLOATING PT. LBS.

TMETTY ONE WORD CONTAINING ESTIMATED TTF IN FLOATING PT. SECS FOR R5RARF.

TMDELT ONE WORD CONTAINING THE DURATION OF THE PRE-RETRO BURN.

TMDVHP TWO WORDS CONTAINING ΔV AND VACUUM PERIGEE HEIGHT FOR THE ON-LINE PRINT.

TMDCSV CONTAINS 6 COMPONENTS OF THE VECTOR AT GMTRC FOR THE DCS UPDATES.

B) COMMUNICATIONS CELLS -

MCGEOC GEOCENTRIC LATITUDE FOR ON-LINE PRINT

MCTYPE CONTAINS THE QUEUE WORD, AS EXPLAINED IN A) IN INPUT SECTION.

MCRECA CONTAINS INFORMATION FOR THE RECOVERY AREA DIGITAL DISPLAY FOR THE SCA.

MCRECP SAME AS MCRECA FOR THE SPA.

MCXXXX CELL SET BY MYTTF4 TO CURRENT GMT PLUS TWO MINUTES, SO THAT MYMINS WILL DELAY QUEUING MYQNEA TO CHECK FOR AUTOMATIC UPDATE OF SCA.

MCYYYYY SAME AS MCXXXX FOR MYQPRA AND SPA.

MCTNEA CONTAINS TIME-OF-FIRING THE PRE-RETRO OAMS FOR THE CURRENTLY DISPLAYED SCA.

MCTSPA SAME AS MCTNEA FOR THE SPA.

MCNOGT IF THERE IS AN ERROR IN COMPUTING GMTRC FOR THE SCA, THIS CELL CONTAINS CURRENT GMT PLUS AN AVERAGE TIME BETWEEN CONTIGENCY AREAS.

MCERRP SAME AS MCNOGT FOR THE SPA.

KBNKLE CONTAINS THE BANK ANGLE FOR R5RARF TO USE FOR EACH TARGET.

METHOD

MYTTF4 IS QUEUED BY MYQPRA, MYQNEA, AND MFMN13 TO CALCULATE AND DISPLAY GMTRC AND OTHER QUANTITIES FOR STORED PRIMARY AREAS, STORED CONTINGENCY AREAS AND MANUALLY INSERTED AREAS, RESPECTIVELY. THE FIRST MANEUVER-REFLECTING VECTOR IS SELECTED AND, UNLESS THE TARGET IS MORE THAN ONE REVOLUTION AWAY AND THERE IS AN INTERVENING BURN MORE THAN ONE REVOLUTION AWAY, THIS VECTOR IS INTEGRATED UP TO THE REVOLUTION THAT THE TARGET IS IN, AND CAPE VECTORS STORED.

THE CAPE VECTOR IN THE TARGET'S REVOLUTION IS USED TO COMPUTE AN ESTIMATED TIME-TO-FIRE TO HIT THE TARGET, AND IF THIS FIRST GUESS IS VALID (THAT IS, THE ORBIT THAT IT IS BASED ON HAS A GMTRC POSSIBLE SUCH THAT THE RETROFIRE TIME IS LATER THAN THE PREVIOUS PROPULSION COMPLETION AND PRIOR TO THE NEXT PROPULSION INITIATION) AN ORBIT TABLE OF STORED VECTORS IS GENERATED SURROUNDING THE ESTIMATED TIME-TO-FIRE. THEN CONTROL IS

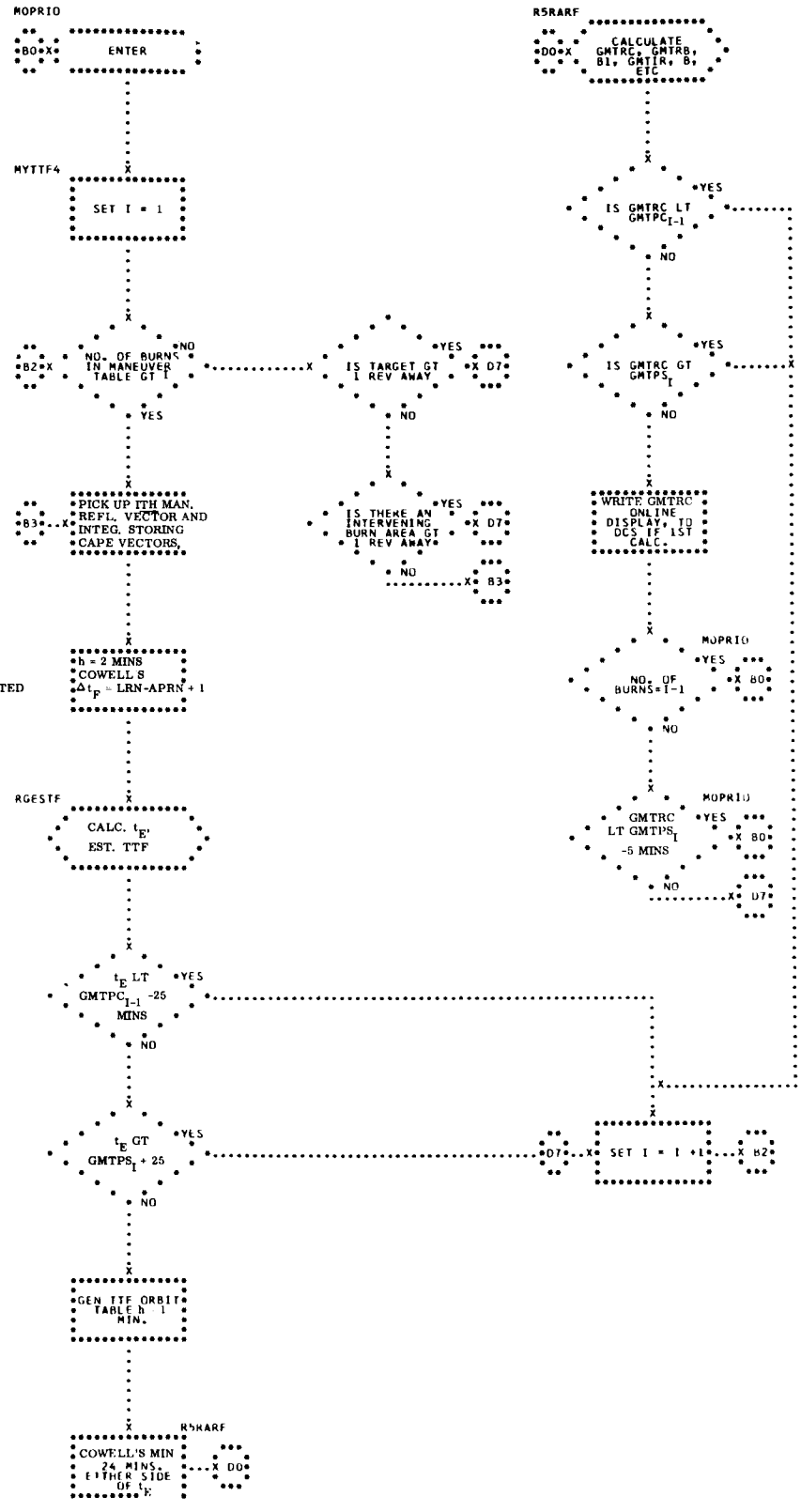
GIVEN TO R5RARF, WHICH PROCEEDS WITH THE MATHEMATICAL CALCULATION NECESSARY TO PRODUCE GMTRC, GMTRB AND BANK ANGLE. IF THE GMTRC IS VALID (AS IN THE ESTIMATED TIME-TO-FIRE) IT IS DISPLAYED AND THE ASSOCIATED OUTPUT WRITTEN ON-LINE. ALSO, THE DCS PRE-RETRO LOAD IS QUEUED TO BE FORMATTED.

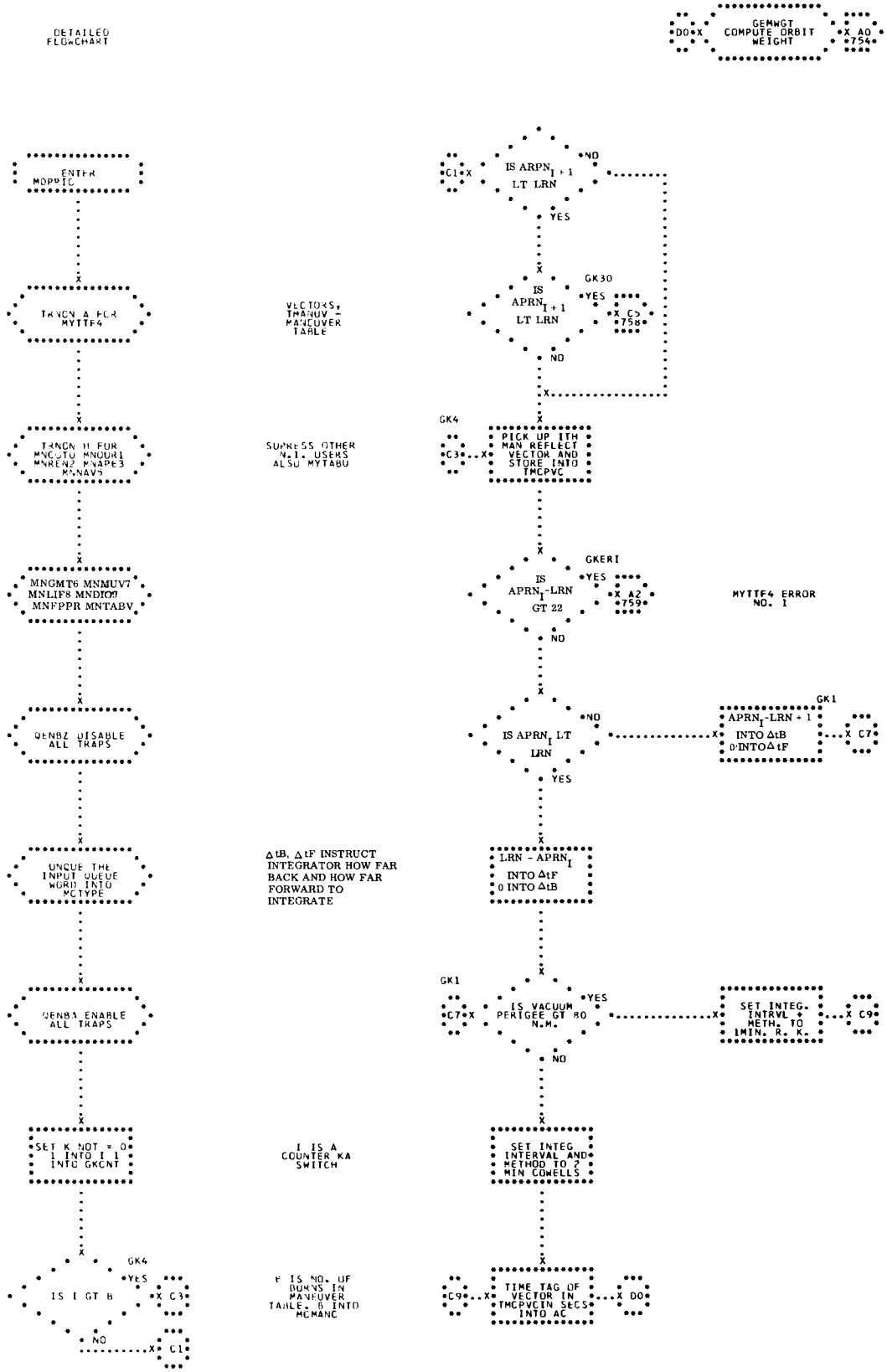
IF THE GMTRC IS INVALID, OR IF THE VALID GMTRC IS WITHIN FIVE MINUTES OF A PROPULSION SETTING, AND IF THERE ARE ADDITIONAL MANEUVER-REFLECTING VECTORS IN TMREFL, THEN THE NEXT SUCH VECTOR IS SELECTED AND THE CYCLE REPEATED.

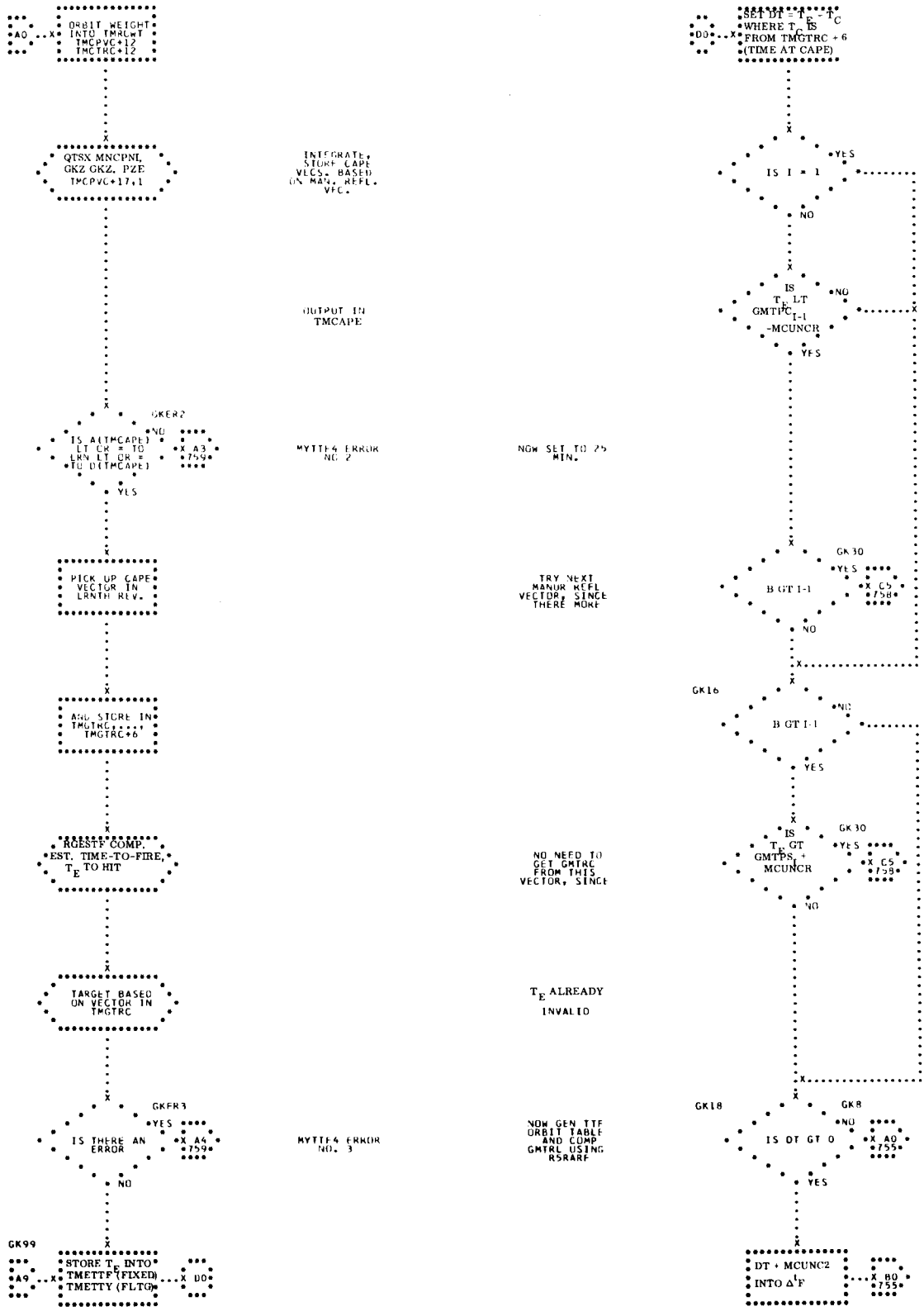
USAGE

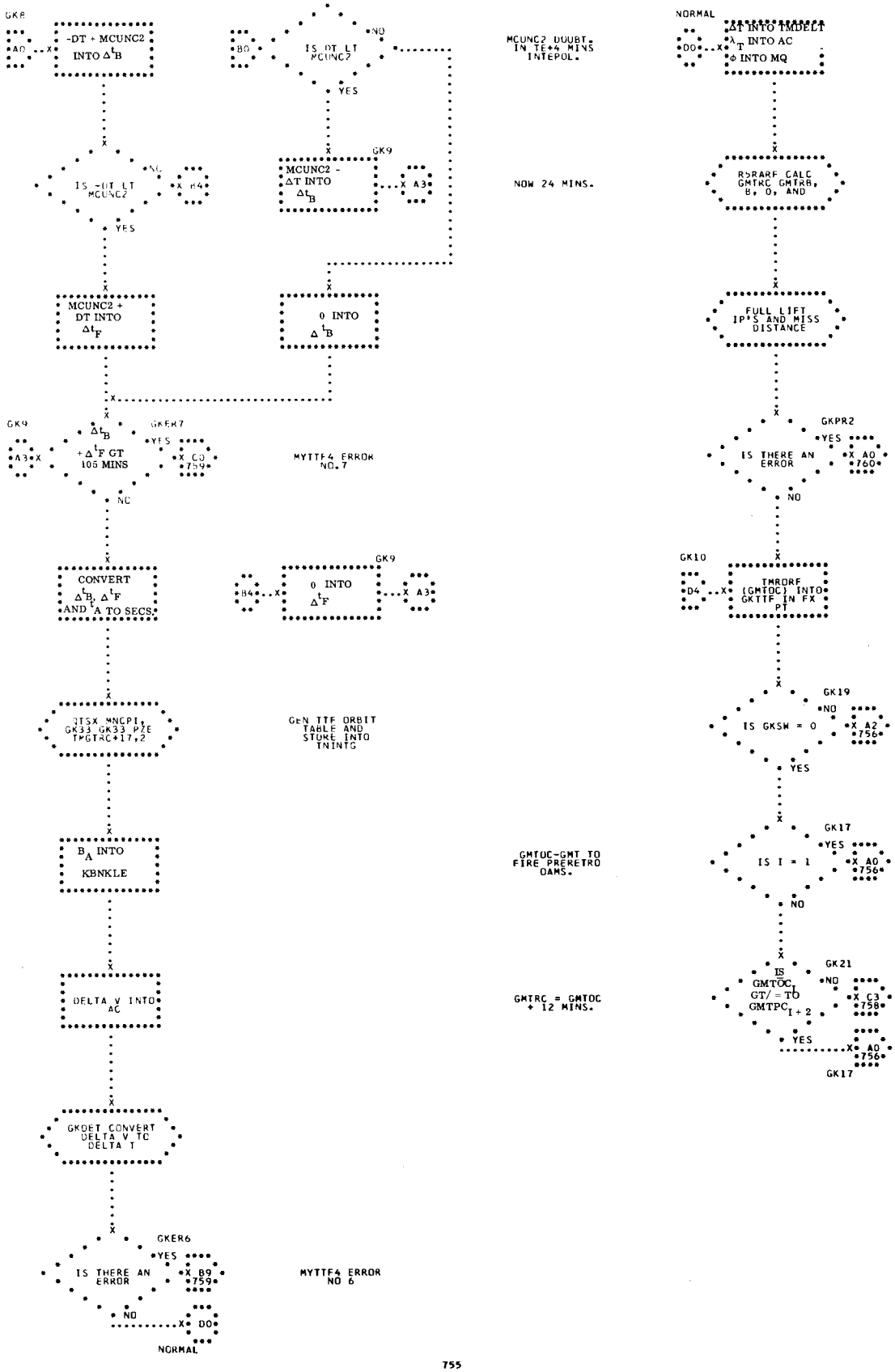
STORAGE REQUIRED -	697 LOCATIONS
TIMING -	MAXIMUM .228 MS + (2.850 MS + R6 + R5) (N +1) WHERE R6 IS THE TIME REQUIRED TO EXECUTE R6ESTF AND R5 IS THE TIME REQUIRED TO EXECUTE R5RARF. N IS THE NUMBER OF MANEUVERS IN THE MANEUVER TABLE.

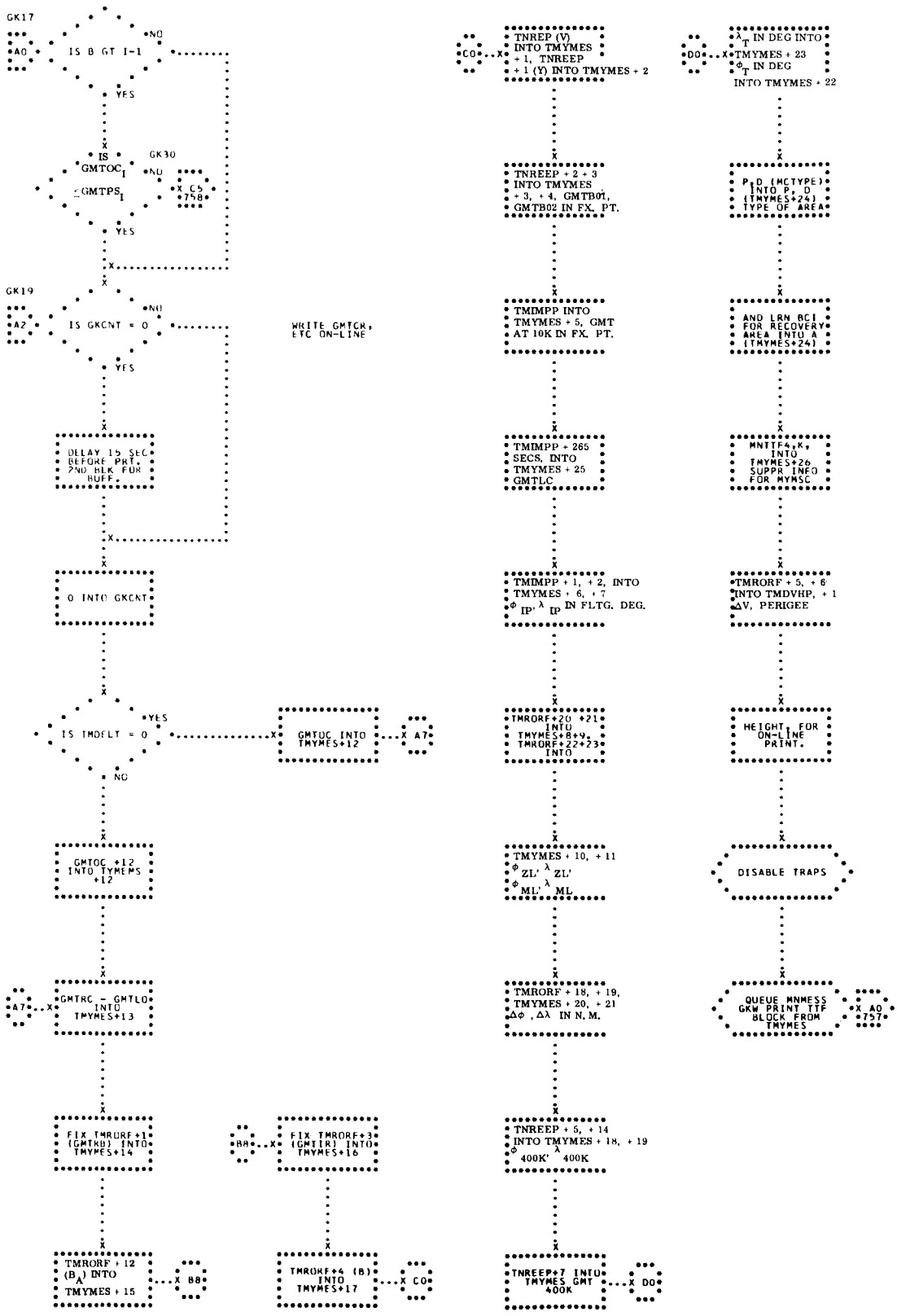
GENERAL FLOWCHART

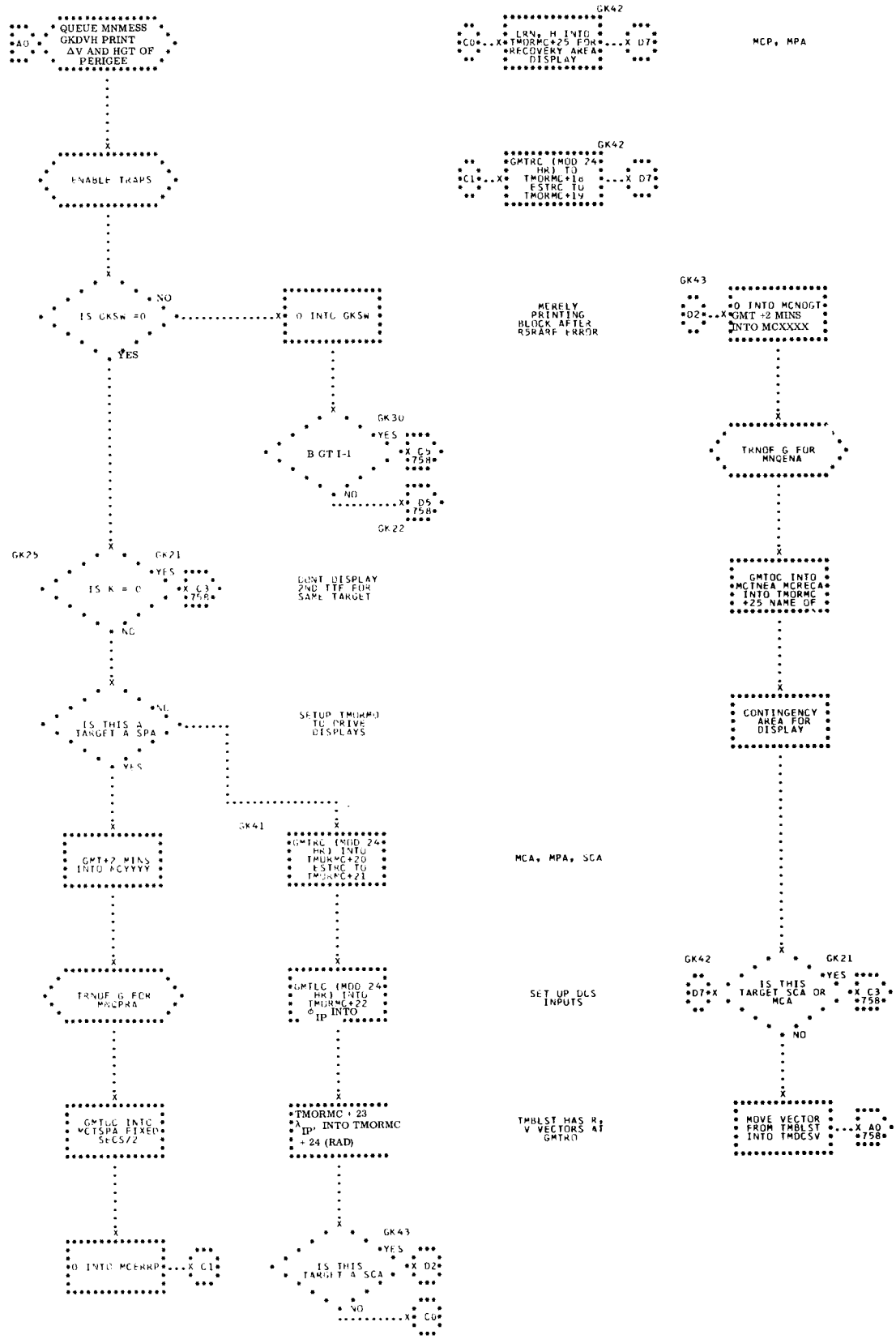


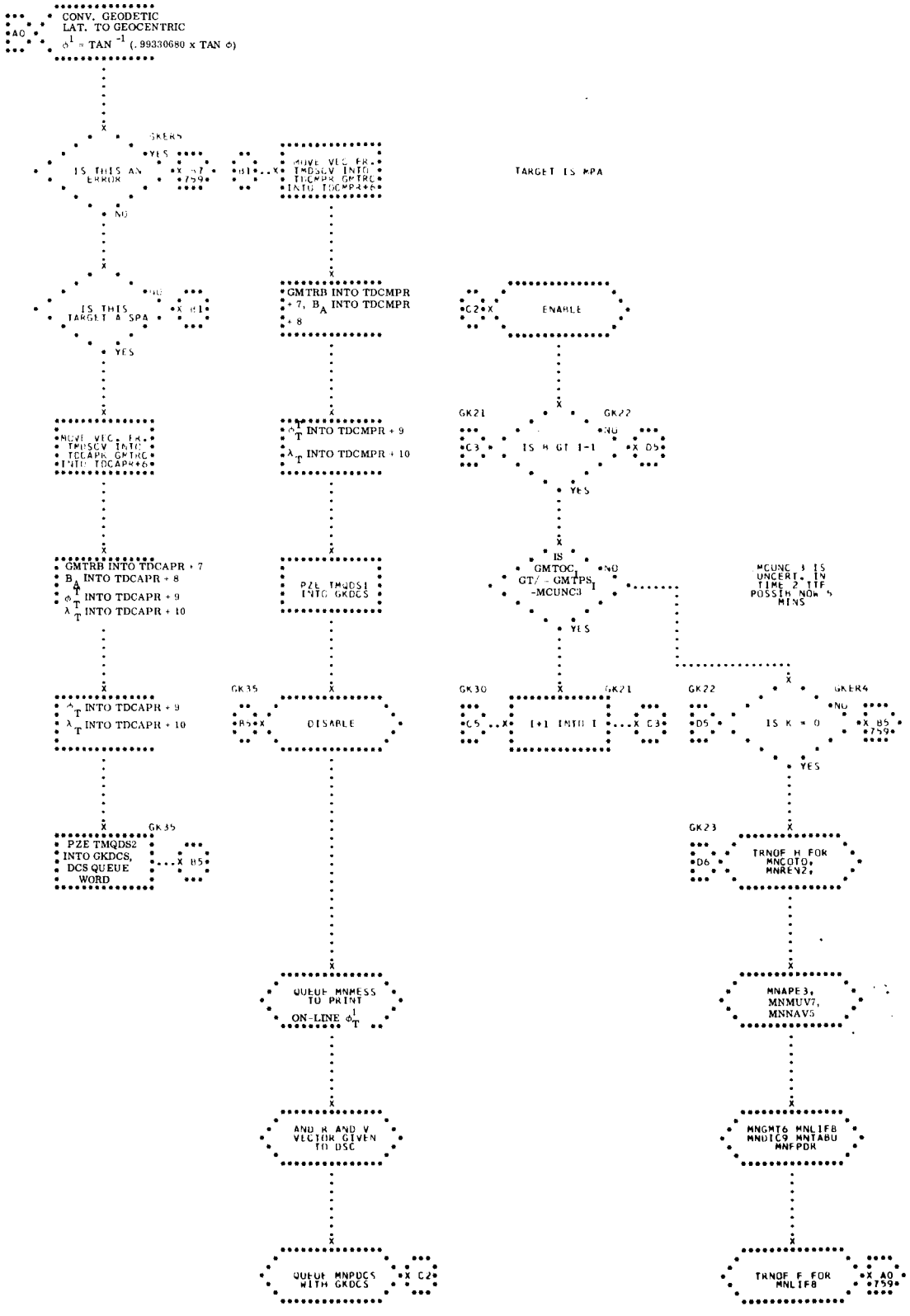

DETAILED FLOWCHART
GEMWGT COMPUTE ORBIT WEIGHT
DO
X AO 754
ENTER MOPPIC
TRNON A FOR MYTTF4
TRNON H FOR MNCUTO MNOUR1 MNREN2 MNAPE3 MGNAV5
MNGMT6 MNMUV7 MNLIF8 MNDIO9 MNEPPR MNTABV
GENBZ DISABLE ALL TRAPS
UNQUE THE INPUT QUEUE WORD INTO MCTYPE
GENBA ENABLE ALL TRAPS
SET K NOT = 0 I INTO I 1 INTO GKCNT
IS I GT B
YES
NO
GK4
X C3
X C1
VECTORS, IMANUV - MANEUVER TABLE
SUPRESS OTHER N.I. USERS ALSO MYTABU
ΔtB, ΔtF INSTRUCT INTEGRATOR HOW FAR BACK AND HOW FAR FORWARD TO INTEGRATE
I IS A COUNTER KA SWITCH
F IS NO. OF BURNS IN MANEUVER TABLE, B INTO MCMANC
C1
IS APRN I+1 LT LRN
NO
YES
GK30
IS APRN I+1 LT LRN
YES
X C5 758
NO
GK4
C3
PICK UP ITH MAN REFLECT VECTOR AND STORE INTO TMCPVC
GKERI
IS APRN I-LRN GT 22
YES
X A2 759
NO
MYTTF4 ERROR NO. 1
IS APRN I LT LRN
NO
YES
GK1
APRN I-LRN + 1 INTO ΔtB 0 INTO ΔtF
X C7
LRN - APRN I INTO ΔtF 0 INTO ΔtB
GK1
C7
IS VACUUM PERIGEE GT 80 N.M.
YES
NO
SET INTEG. INTRVL + METH. TO 1MIN. R. K.
X C9
SET INTEG INTERVAL AND METHOD TO 2 MIN COWELLS
C9
TIME TAG OF VECTOR IN TMCPVCIN SECS INTO AC
X DO

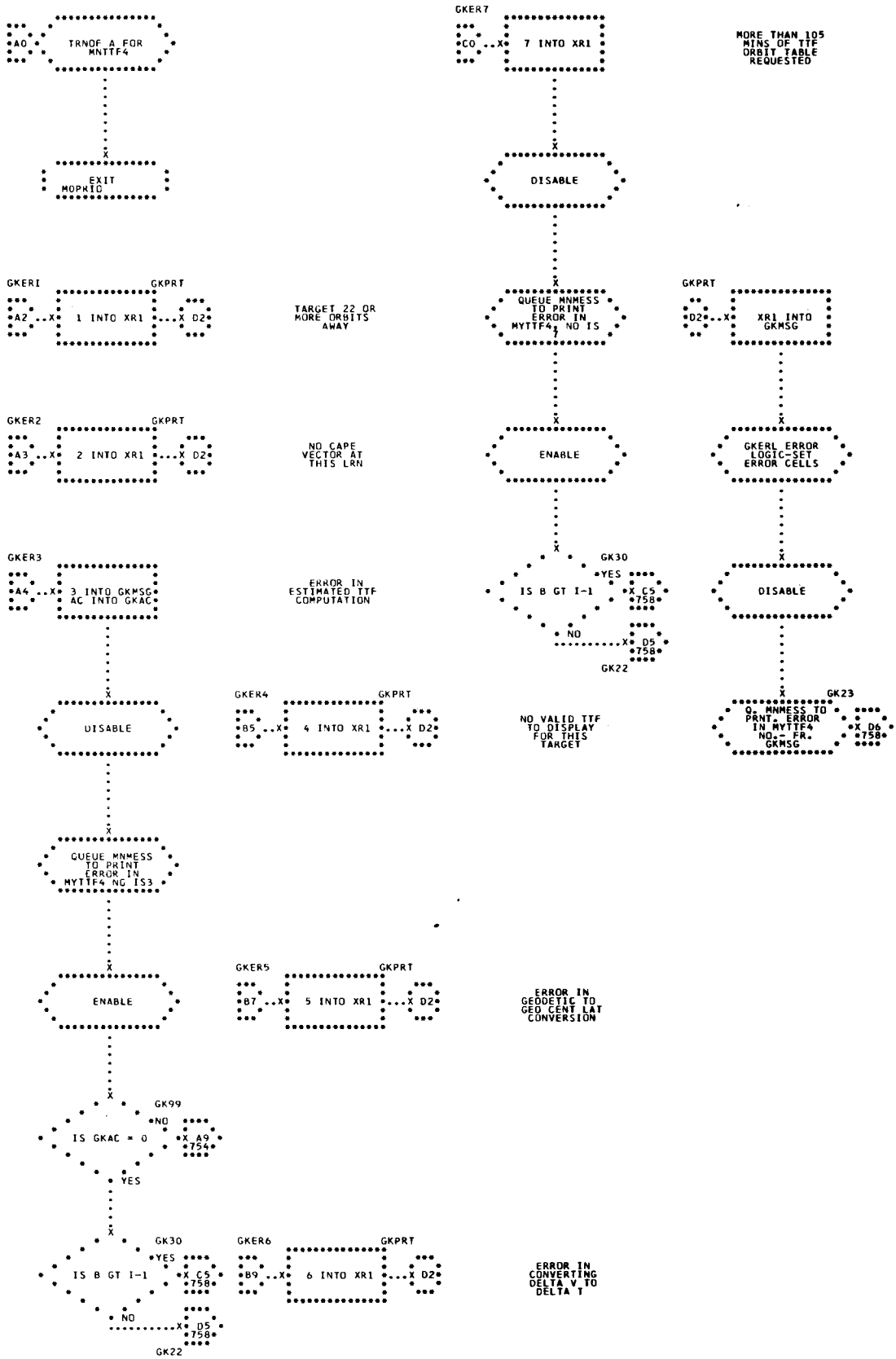

ORBIT WEIGHT INTO TMRCWT TMCPVC+12 TMCTRC+12
A0
QTSX MNCPNI, GKZ GKZ, PZE TMCPVC+17,1
INTEGRATE, STORE CAPE VECS. BASED ON MAN. REFL. VEC.
OUTPUT IN TMCAPE
GKER2
IS A(TMCAPE) LT CK = TO LRN LT CK = TU D(TMCAPE)
NO
YES
MYTTF4 ERROR NO. 2
A3 *759*
NOW SET TO 25 MIN.
PICK UP CAPE VECTOR IN LRNTH REV.
TRY NEXT MANUR REFL VECTOR, SINCE THERE MORE
ADD, STORE IN TMGTRC, TMGTRC+6
RGESTF COMP. EST. TIME-TO-FIRE, T_E TO HIT
NO NEED TO GET GMTRC FROM THIS VECTOR, SINCE
TARGET BASED ON VECTOR IN TMGTRC
T_E ALREADY INVALID
GKER3
IS THERE AN ERROR
YES
NO
MYTTF4 ERROR NO. 3
A4 *759*
NOW GEN TTF ORBIT TABLE AND COMP GMTRL USING RSRARF
GK99
STORE T_E INTO TMETTF (FIXED) TMETTY (FLTG)
A9
D0
SET DT = $T_E - T_C$ WHERE T_C IS FROM TMGTRC + 6 (TIME AT CAPE)
D0
IS I = 1
YES
NO
IS T_E LT GMTPC$_{I-1}$ -MCUNCR
NO
YES
GK30
B GT I-1
YES
C5 *758*
NO
GK16
B GT I-1
NO
YES
IS T_E GT GMTPS$_I$ + MCUNCR
GK30
YES
C5 *758*
NO
GK18
GK8
IS DT GT 0
NO
YES
A0 *755*
DT + MCUNC2 INTO $\Delta^t F$
B0 *755*

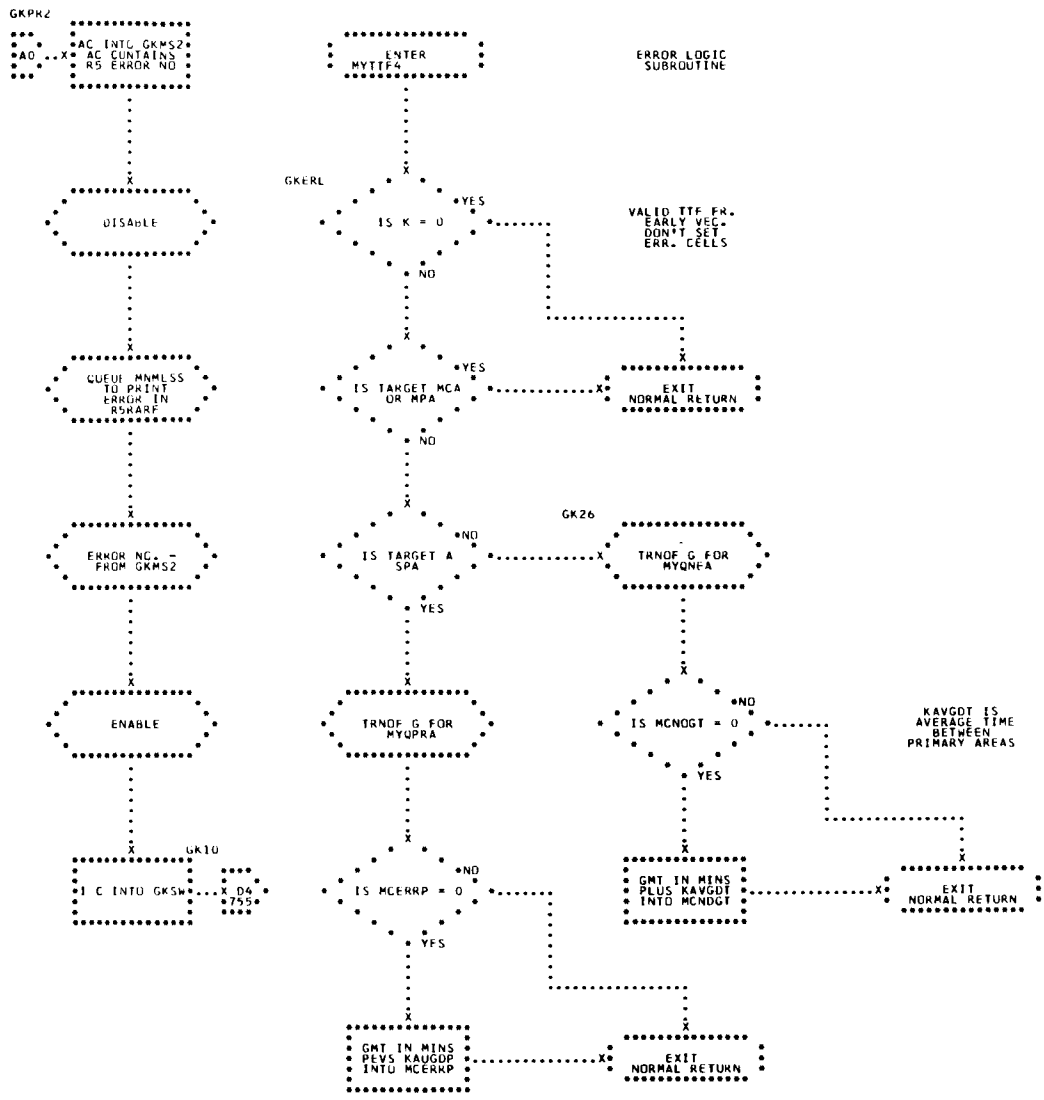

GK8
A0
-DT + MCUNC2 INTO $\Delta^t B$
B0
IS DT LT MCUNC2
NO
YES
MCUNC2 DOUBT. IN TE+4 MINS INTEPOL.
NORMAL
D0
ΔT INTO TMDELT
λ_T INTO AC
ϕ INTO MQ
IS -DT LT MCUNC2
NO
B4
YES
GK9
MCUNC2 - ΔT INTO $\Delta^t B$
A3
NOW 24 MINS.
R5RARF CALC GMTRC GMTRB, B, 0, AND
MCUNC2 + DT INTO $\Delta^t F$
0 INTO $\Delta^t B$
FULL LIFT IP'S AND MISS DISTANCE
GK9
A3
$\Delta^t B$ + $\Delta^t F$ GT 105 MINS
GKER7
YES
C0 759
NO
MYTTF4 ERROR NO.7
GKPR2
IS THERE AN ERROR
YES
A0 760
NO
CONVERT $\Delta^t B$, $\Delta^t F$ AND $^t A$ TO SECS.
GK9
B4
0 INTO $\Delta^t F$
A3
GK10
D4
TMRORF (GMTOC) INTO GKTTF IN FX PT
QTSX MNCPI, GK33 GK33 PZE TMGTRC+17,2
GEN TTF ORBIT TABLE AND STORE INTO TNINTG
GK19
IS GKSW = 0
NO
A2 756
YES
B_A INTO KBNKLE
GMTOC-GMT TO FIRE PRERETRO OAMS.
GK17
IS I = 1
YES
A0 756
NO
DELTA V INTO AC
GMTRC = GMTOC + 12 MINS.
GK21
IS $GMTOC_I$ GT/ = TO $GMTPC_{I+2}$
NO
C3 758
YES
A0 756
GK17
GKDET CONVERT DELTA V TO DELTA T
GKER6
IS THERE AN ERROR
YES
B9 759
NO
D0
NORMAL
MYTTF4 ERROR NO 6

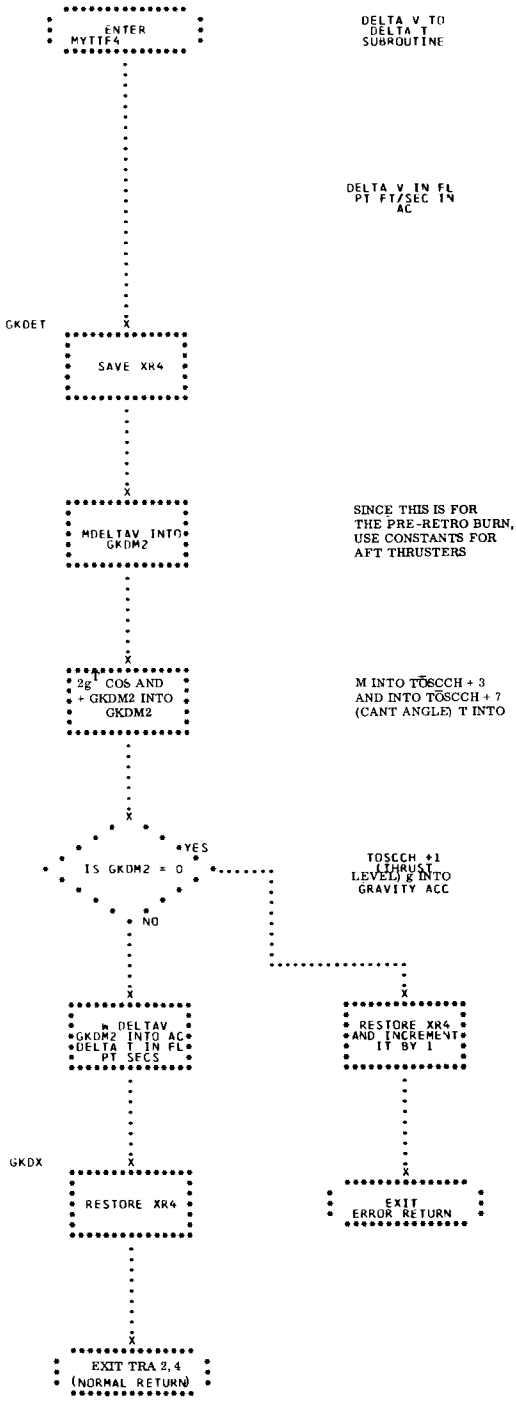

GK17
IS B GT I-1
NO
YES
IS GMTOC_I ≤ GMTPS_I
GK30
NO
C5 *758
YES
GK19
IS GKCNT = 0
NO
YES
DELAY 15 SEC BEFORE PRT. 2ND BLK FOR BUFF.
0 INTO GKCNT
IS IMOFLT = 0
YES
NO
GMTOC +12 INTO TYMEMS +12
GMTRC - GMTLO INTO TMYMES+13
FIX TMRORF+1 (GMTKU) INTO TMYMES+14
TMRORF + 12 (B_A) INTO TMYMES + 15
A0
A2
A7
B8
C0
D0
WRITE GMTCR, ETC ON-LINE
GMTOC INTO TMYMES+12
FIX TMRORF+3 (GMTIR) INTO TMYMES+16
TMRORF+4 (B) INTO TMYMES+17
TNREP (V) INTO TMYMES + 1, TNREEP + 1 (Y) INTO TMYMES + 2
TNREEP + 2 + 3 INTO TMYMES + 3, + 4, GMTB01, GMTB02 IN FX. PT.
TMIMPP INTO TMYMES + 5, GMT AT 10K IN FX. PT.
TMIMPP + 265 SECS, INTO TMYMES + 25 GMTLC
TMIMPP + 1, + 2, INTO TMYMES + 6, + 7 φ_IP', λ_IP IN FLTG. DEG.
TMRORF+20 +21 INTO TMYMES+8+9, TMRORF+22+23 INTO
TMYMES + 10, + 11 φ_ZL', λ_ZL' φ_ML', λ_ML
TMRORF + 18, + 19, TMYMES + 20, + 21 Δφ, Δλ IN N.M.
TNREEP + 5, + 14 INTO TMYMES + 18, + 19 φ_400K', λ_400K
TNREEP+7 INTO TMYMES GMT 400K
λ_T IN DEG INTO TMYMES + 23 φ_T IN DEG INTO TMYMES + 22
P, D (MCTYPE) INTO P, D (TMYMES+24) TYPE OF AREA
AND LRN BCI FOR RECOVERY AREA INTO A (TMYMES+24)
MNTTF4,K, INTO TMYMES+26 SUPPR INFO FOR MYMSC
TMRORF + 5, + 6 INTO TMDVHP, + 1 ΔV, PERIGEE
HEIGHT, FOR ON-LINE PRINT.
DISABLE TRAPS
QUEUE MNMESS GKW PRINT TTF BLOCK FROM TMYMES
X A0 *757*


A0
QUEUE MNMESS GKDVH PRINT ΔV AND HGT OF PERIGEE
ENABLE TRAPS
IS GKSW =0
NO
YES
0 INTO GKSW
GK30
B GT I-1
YES
X C5 *758*
NO
X D5 *758*
GK22
GK25
IS K = 0
GK21
YES
X C3 *758*
NO
DONT DISPLAY 2ND TTF FOR SAME TARGET
IS THIS A TARGET A SPA
NO
YES
SETUP TMORMD TO DRIVE DISPLAYS
GK41
GMT+2 MINS INTO MCYYYY
TRNOF G FOR MNGPRA
GMTLC INTO MCTSPA FIXED SECS/2
0 INTO MCERRP
X C1
GMTRC (MOD 24 HR) INTO TMORMC+20 ESTRC TU TMORMC+21
GMTLC (MOD 24 HR) INTO TMORMC+22 Φ IP INTO IP
*TMORMC + 23 *A IP INTO TMORMC + 24 (RAD)
GK43
YES
IS THIS TARGET A SCA
X D2
NO
X C0
GK42
C0
LRN, H INTO TMORMC+25 FOR RECOVERY AREA DISPLAY
X D7
MCP, MPA
GK42
C1
GMTRC (MOD 24 HR) TO TMORMC+18 ESTRC TU TMORMC+19
X D7
MERELY PRINTING BLOCK AFTER RSRARF ERROR
MCA, MPA, SCA
SET UP DCS INPUTS
TMBLST HAS R, V VECTORS AT GMTRD
GK43
D2
0 INTO MCNOGT *GMT +2 MINS INTO MCXXXX
TRNOF G FOR MNQENA
GMTOC INTO MCTNEA MCRECA INTO TMORMC +25 NAME OF
CONTINGENCY AREA FOR DISPLAY
GK42
D7
GK21
IS THIS TARGET SCA OR MCA
YES
X C3 *758*
NO
MOVE VECTOR FROM TMBLST INTO TMDCSV
X A0 *758*


A0
CONV. GEODETIC
LAT. TO GEOCENTRIC
$\phi^1 = TAN^{-1}$ (.99330680 x TAN ϕ)
GKER5
IS THIS AN ERROR
YES
NO
B1
MOVE VEC FR. TMDSCV INTO TDCMPR GMTRC INTO TDCMPR+6
TARGET IS MPA
IS THIS TARGET A SPA
B1
YES
GMTRB INTO TDCMPR + 7, B_A INTO TDCMPR + 8
C2
ENABLE
MOVE VEC. FR. TMDSCV INTO TDCAPR GMTRC INTO TDCAPR+6
ϕ_T INTO TDCMPR + 9
λ_T INTO TDCMPR + 10
GK21
GK22
C3
IS B GT I-1
D5
YES
GMTRB INTO TDCAPR + 7
B_A INTO TDCAPR + 8
ϕ_T INTO TDCAPR + 9
λ_T INTO TDCAPR + 10
PZE TMQDS1 INTO GKDCS
IS GMTOC$_I$ GT/ - GMTPS$_I$ -MCUNC3
MCUNC 3 IS UNCERT. IN TIME 2 TTF POSSIB NOW 5 MINS
YES
ϕ_T INTO TDCAPR + 9
λ_T INTO TDCAPR + 10
GK35
B5
DISABLE
GK30
GK21
C5
I+1 INTO I
C3
GK22
GKER4
D5
IS K = 0
B5
759
YES
GK35
PZE TMQDS2 INTO GKDCS, DCS QUEUE WORD
B5
GK23
D6
TRNOF H FOR MNCOTO, MNREN2,
QUEUE MNMESS TO PRINT ON-LINE ϕ_T^1
MNAPE3, MNMUV7, MNNAV5
AND R AND V VECTOR GIVEN TO DSC
MNGMT6 MNLIF8 MNDIC9 MNTABU MNFPDR
QUEUE MNPDCS WITH GKDCS
C2
TRNOF F FOR MNLIF8
A0
759


A0
TRNOF A FOR MNTTF4
EXIT MOPRTD
GKER1
GKPRT
A2
1 INTO XR1
D2
TARGET 22 OR MORE ORBITS AWAY
GKER2
GKPRT
A3
2 INTO XR1
D2
NO CAPE VECTOR AT THIS LRN
GKER3
A4
3 INTO GKMSG AC INTO GKAC
ERROR IN ESTIMATED TTF COMPUTATION
DISABLE
GKER4
GKPRT
B5
4 INTO XR1
D2
NO VALID TTF TO DISPLAY FOR THIS TARGET
QUEUE MNMESS TO PRINT ERROR IN MYTTF4 NO IS3
ENABLE
GKER5
GKPRT
B7
5 INTO XR1
D2
ERROR IN GEODETIC TO GEO CENT LAT CONVERSION
GK99
IS GKAC = 0
NO
X A9 *754*
YES
GK30
IS B GT I-1
YES
X C5 *758*
NO
X D5 *758*
GK22
GKER6
GKPRT
B9
6 INTO XR1
D2
ERROR IN CONVERTING DELTA V TO DELTA T
GKER7
C0
7 INTO XR1
MORE THAN 105 MINS OF TTF ORBIT TABLE REQUESTED
DISABLE
QUEUE MNMESS TO PRINT ERROR IN MYTTF4, NO IS 7
ENABLE
GK30
IS B GT I-1
YES
X C5 *758*
NO
X D5 *758*
GK22
GKPRT
D2
XR1 INTO GKMSG
GKERL ERROR LOGIC-SET ERROR CELLS
DISABLE
GK23
Q. MNMESS TO PRNT. ERROR IN MYTTF4 NO.- FR. GKMSG
X D6 *758*


GKPR2
AC INTO GKMS2
AC CONTAINS R5 ERROR NO
AO
DISABLE
QUEUE MNMLSS TO PRINT ERROR IN RSHARF
ERROR NO. FROM GKMS2
ENABLE
GK10
1 C INTO GKSW
D4
755
ENTER MYTIF4
GKERL
IS K = 0
YES
NO
IS TARGET MCA OR MPA
YES
NO
IS TARGET A SPA
NO
YES
TRNOF G FOR MYQPRA
IS MCERRP = 0
NO
YES
GMT IN MINS PEVS KAVGDP INTO MCERKP
EXIT NORMAL RETURN
VALID TTF FR. EARLY VEC. DON'T SET ERR. CELLS
EXIT NORMAL RETURN
GK26
TRNOF G FOR MYQNFA
IS MCNOGT = 0
NO
YES
GMT IN MINS PLUS KAVGDT INTO MCNOGT
KAVGDT IS AVERAGE TIME BETWEEN PRIMARY AREAS
EXIT NORMAL RETURN
ERROR LOGIC SUBROUTINE


ENTER MYTTF4
DELTA V TO DELTA T SUBROUTINE
DELTA V IN FL PT FT/SEC IN AC
GKDET
SAVE XR4
MDELTAV INTO GKDM2
SINCE THIS IS FOR THE PRE-RETRO BURN, USE CONSTANTS FOR AFT THRUSTERS
2g COS AND + GKDM2 INTO GKDM2
M INTO TŌSCCH + 3 AND INTO TŌSCCH + 7 (CANT ANGLE) T INTO
IS GKDM2 = 0
YES
NO
TOSCCH +1 (THRUST LEVEL) g INTO GRAVITY ACC
m DELTAV GKDM2 INTO AC DELTA T IN FL PT SECS
RESTORE XR4 AND INCREMENT IT BY 1
GKDX
RESTORE XR4
EXIT ERROR RETURN
EXIT TRA 2,4 (NORMAL RETURN)

15. MYNAV5 - ORBIT NAVIGATION VECTOR FOR DCS

MYNAV5 GENERATES THE MANEUVER-REFLECTED ORBIT NAVIGATION VECTOR AND QUEUES THE DIGITAL COMMAND SYSTEM LOAD PROGRAM, WHICH WILL FORMAT AND TRANSMIT IT. (SEE FLOWCHART)

INPUT

A) QUEUE WORD — CONTAINS GMTRC (GMT OF FUNCTION COMPUTED) IN FIXED POINT SECONDS

B) TABLES -

TMREFL	MANEUVER-REFLECTED VECTOR TABLE (SEE MYTTF4 FOR FORMAT.)
TMANUV	MANEUVER TABLE (SEE MYTTF4 FOR FORMAT.)
TNINTG	TABLE OF VECTORS SET UP BY NOCPNI IN ITS STANDARD OUTPUT FORMAT.
TNLAST	TABLE CONTAINING ONE VECTOR, THE LAST ONE WHILE INTEGRATING, IN THE NOCPNI FORMAT FOR INTEGRATING WITHOUT STORING.

C) COMMUNICATION CELLS -

MCHFSC	CURRENT GMT IN FIXED POINT HALF-SECONDS.
MCMANC	NUMBER OF MANEUVERS IN MANEUVER TABLE.

D) CONSTANTS -

K2100M K00100
K00005 K00004
K00008 K00001
K00060 KT0001

E) SYSTEM SUBROUTINES

GEMWGT COMPUTES SPACECRAFT WEIGHT.
UINTP LAGRANGE INTERPOLATOR.

F) MACROS -

QTSX QENBA
QUEUE TRNON
QENBZ TRNOF

OUTPUT

A) TMONAV TABLE CONTAINING THE INPUT TO NOCPNI IN THE STANDARD INTEGRATOR FORMAT. (SEE MPCPNI FOR FORMAT.)

B) TDCSON THE INPUT BLOCK TO OOPDCS FOR ORBIT NAVIGATION TYPE DCS UPDATES, CONSISTING OF X,Y,Z,$\dot{X}$,$\dot{Y}$,$\dot{Z}$ AND T IN INERTIAL COORDINATES, GEMINI UNITS.

C) TMDCON 1-WORD INPUT LOCATION FOR MPPDCS CONTAINING THE CODE (1) IN THE DECREMENT, INDICATING GENERATE TRANSLATION AND LOAD, AND THE TYPE (1) IN THE ADDRESS, INDICATING ORBIT NAVIGATION UPDATE.

METHOD

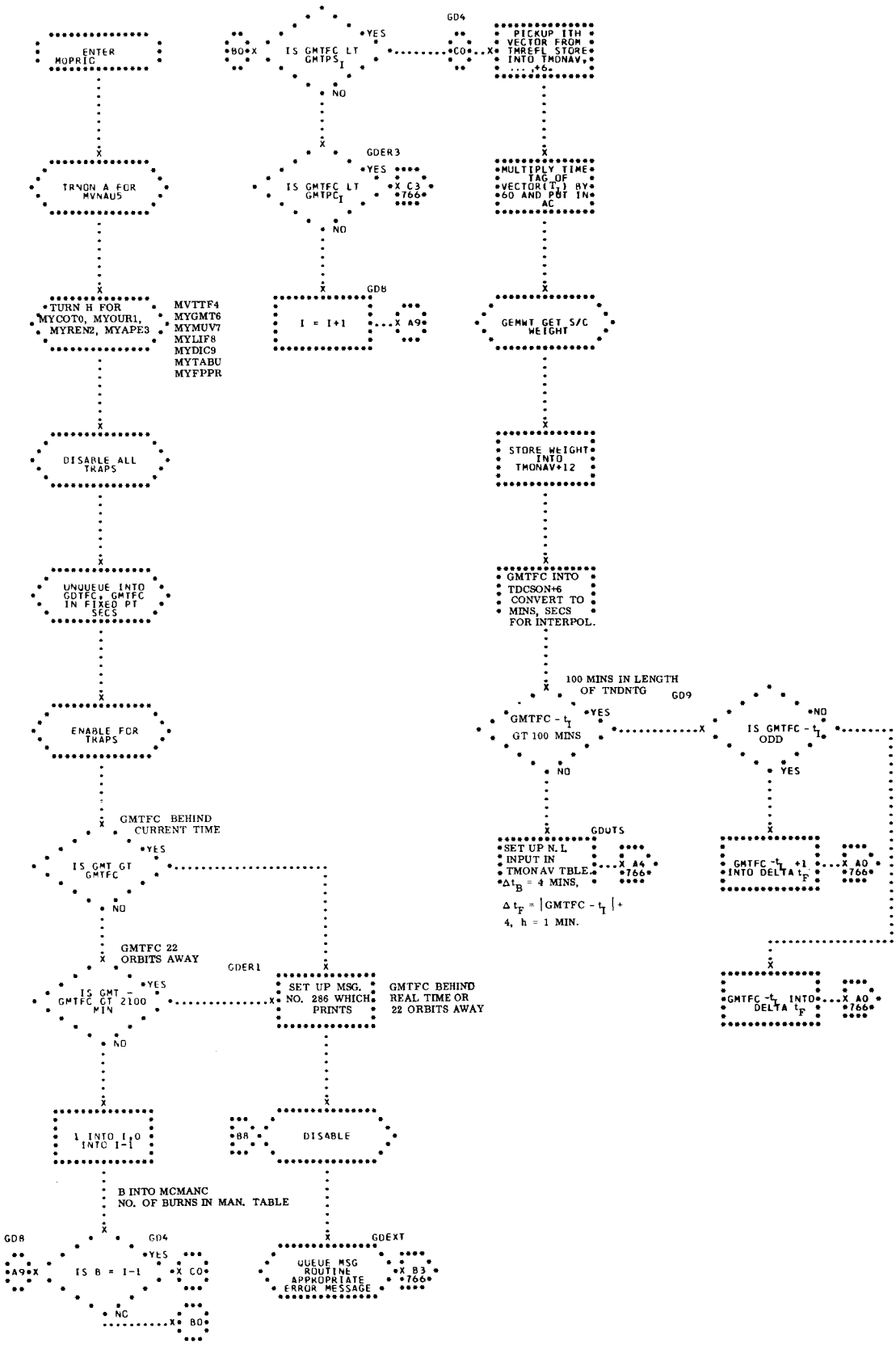
MYNAV5 IS QUEUED BY MFMN17, UPON MANUAL INSERTION OF A REQUESTED GMTFC (GMT OF FUNCTION COMPUTED). AT THIS TIME IT IS REQUIRED TO FIND THE CORRESPONDING POSITION AND VELOCITY VECTORS, TAKING PLANNED MANEUVERS INTO

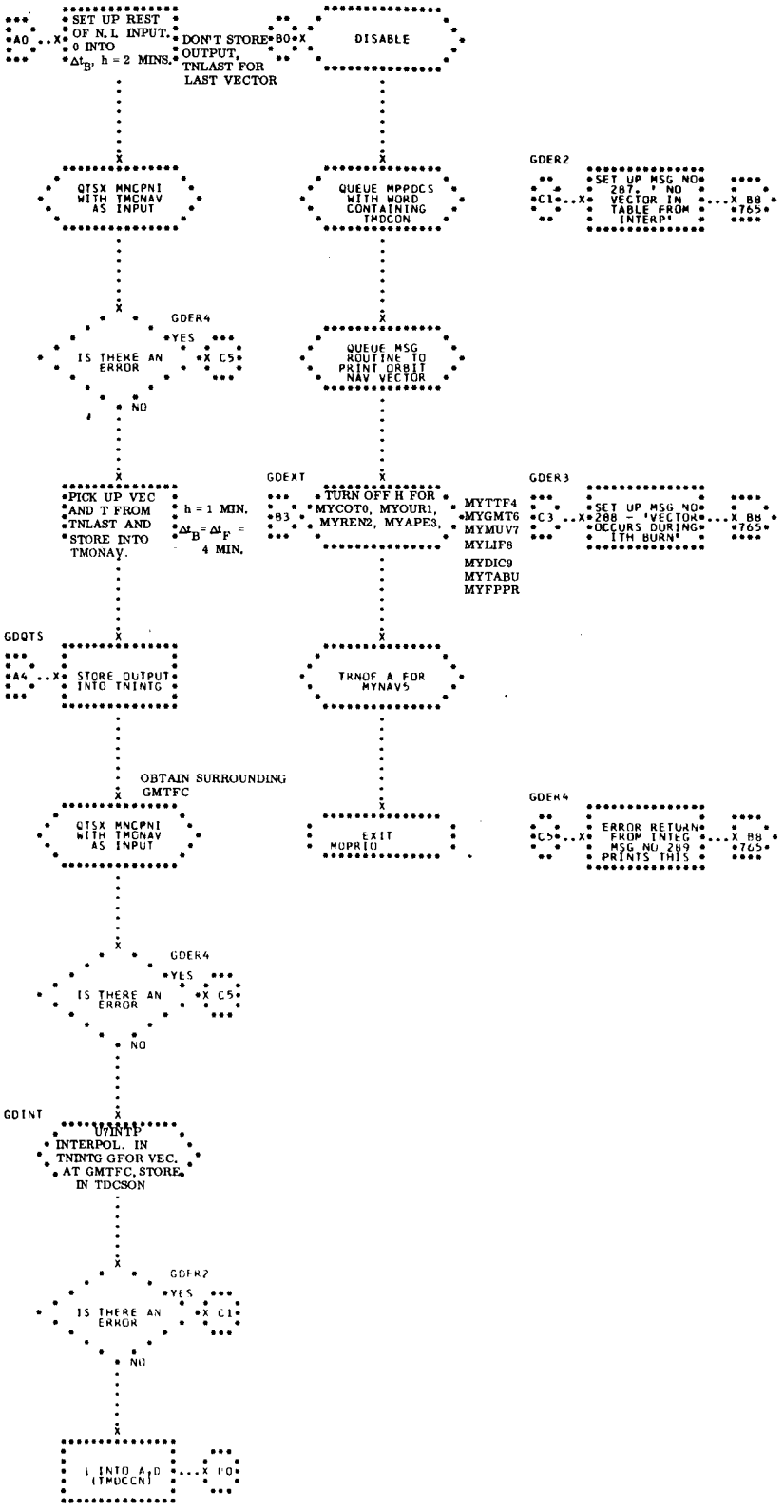
ACCOUNT. THE MANEUVER-REFLECTING VECTOR THAT LIES IMMEDIATELY BEFORE THIS GMTFC IS SELECTED FOR THE ANCHOR POINT FOR THE INTEGRATOR. IF THIS VECTOR IS WITHIN 100 MINUTES OF GMTFC, IT IS INTEGRATED DIRECTLY 4 MINUTES BEYOND GMTFC IN 1-MINUTE STEPS.

IF THIS VECTOR IS NOT WITHIN 100 MINUTES, WE FIRST INTEGRATE NEAR GMTFC IN 2-MINUTE STEPS WITHOUT STORING, THEN INTEGRATE IN 1-MINUTE STEPS, 4-MINUTES EITHER SIDE OF GMTFC. WE THEN INTERPOLATE IN THE N.I. OUTPUT TABLE, TNINTG, FOR THE VECTOR AT GMTFC AND STORE THIS IN TDCSON, THE ORBIT NAVIGATION VECTOR INPUT BLOCK FOR THE DCS PROGRAM. MPPDCS IS QUEUED BY THIS INPUT AND THE VECTOR COMPONENTS ARE PRINTED ON-LINE.

USAGE

STORAGE REQUIRED -
178 LOCATIONS


ENTER MOPRIC
TRNON A FOR MVNAUS
TURN H FOR MYCOT0, MYOUR1, MYREN2, MYAPE3
MVTTF4
MYGMT6
MYMUV7
MYLIF8
MYDIC9
MYTABU
MYFPPR
DISABLE ALL TRAPS
UNQUEUE INTO GDTFC, GMTFC IN FIXED PT SECS
ENABLE FOR TRAPS
GMTFC BEHIND CURRENT TIME
IS GMT GT GMTFC
YES
NO
GMTFC 22 ORBITS AWAY
IS GMT - GMTFC GT 2100 MIN
GDER1
SET UP MSG. NO. 286 WHICH PRINTS
GMTFC BEHIND REAL TIME OR 22 ORBITS AWAY
1 INTO I,0 INTO I-1
B INTO MCMANC
NO. OF BURNS IN MAN. TABLE
IS B = I-1
GD8
GD4
CO
B0
DISABLE
GDEXT
QUEUE MSG ROUTINE APPROPRIATE ERROR MESSAGE
B3
*766
IS GMTFC LT GMTPS I
GDER3
IS GMTFC LT GMTPC I
C3
I = I+1
GDB
A9
PICKUP ITH VECTOR FROM TMREFL STORE INTO TMONAV, ... ,+6
MULTIPLY TIME TAG OF VECTOR(T_I) BY 60 AND PUT IN AC
GEMWT GET S/C WEIGHT
STORE WEIGHT INTO TMONAV+12
GMTFC INTO TDCSON+6 CONVERT TO MINS, SECS FOR INTERPOL.
100 MINS IN LENGTH OF TNDNTG
GD9
GMTFC - t_I GT 100 MINS
IS GMTFC - t_I ODD
NO
YES
GDUTS
SET UP N.L INPUT IN TMONAV TBLE. Δt_B = 4 MINS,
Δt_F = |GMTFC - t_I| + 4, h = 1 MIN.
A4
GMTFC -t_I +1 INTO DELTA t_F
A0
GMTFC -t_I INTO DELTA t_F


A0
SET UP REST OF N.L. INPUT. 0 INTO Δtb, h = 2 MINS.
B0
DON'T STORE OUTPUT, TNLAST FOR LAST VECTOR
DISABLE
QTSX MNCPNI WITH TMCNAV AS INPUT
QUEUE MPPDCS WITH WORD CONTAINING TMDCON
GDER2
C1
SET UP MSG NO 287, ' NO VECTOR IN TABLE FROM INTERP'
B8
765
GDER4
IS THERE AN ERROR
YES
C5
NO
QUEUE MSG ROUTINE TO PRINT ORBIT NAV VECTOR
PICK UP VEC AND T FROM TNLAST AND STORE INTO TMONAV.
h = 1 MIN. Δtb = Δtp = 4 MIN.
GDEXT
B3
TURN OFF H FOR MYCOT0, MYOUR1, MYREN2, MYAPE3, MYTTF4 MYGMT6 MYMUV7 MYLIF8 MYDIC9 MYTABU MYFPPR
GDER3
C3
SET UP MSG NO 288 - 'VECTOR OCCURS DURING ITH BURN'
GDQTS
A4
STORE OUTPUT INTO TNINTG
TRNOF A FOR MYNAV5
OBTAIN SURROUNDING GMTFC
EXIT MOPRIO
C5
ERROR RETURN FROM INTEG MSG NO 289 PRINTS THIS
GDINT
U7INTP
INTERPOL. IN TNINTG GFOR VEC. AT GMTFC, STORE IN TDCSON
GDER2
IS THERE AN ERROR
YES
C1
NO
L INTO A,D (TMDCCN)
F0

16. MYGMT6 - CALCULATE GMT

THE PURPOSE OF THE PROGRAM IS TO CALCULATE A GMT FOR GIVEN LONGITUDE AND LANDING REVOLUTION NUMBER . (SEE FLOWCHART.)

INPUT

A) SUBROUTINE -

A3MSCP

B) PROGRAMS -

GEMWGT

MPCPNI

C) CONSTANTS -

K00001	K00060
K00002	K00007
K00008	K001.0
K060.0	K095.0
K00005	K00003
K00004	K00006
KCH233	KLCAPE

D) TABLES -

TMREFL	X
+1	Y
+2	Z
+3	$\dot{X}$
+4	$\dot{Y}$
+5	$\dot{Z}$
+6	TIME
+7	ANCHOR POINT REVOLUTION NUMBER

TJKINT - A 17-LOCATION TABLE CORRESPONDING TO
TMBBNI (INPUT TABLE FOR INTEGRATOR)

TMANUV GMTPS - TIME TO INITIATE THE MANEUVER
+1 GMTPC - TIME OF COMPLETION OF MANEUVER
+2 A(ΔT) ,,D(THRUSTER INDICATOR)
+3 PITCH
+4 YAW

E) SYSTEM MACROS -

QENBA
UNQUE
QEWBA
QTSX
QUEUE

F) COMMUNICATION CELLS -

MCMANC CONTAINS THE NUMBER OF MANEUVERS TO BE PERFORMED

G) INPUT CELL -

JKINP A(ADDRESS OF DESIRED LONGITUDE),,
D(LANDING REVOLUTION NUMBER)

H) CONDITIONS ON ENTRY -

NO PREREQUISITES

OUTPUT

THE GMT AT THE DESIRED LONGITUDE AND LANDING REVOLUTION NUMBER IS PRINTED ON LINE.

TMCAPE CAPE VECTOR STORAGE AREA TABLE

METHOD

THE PROPER MANUEVER REFLECTING VECTOR IS SELECTED FROM TMREFL. THIS VECTOR IS USED TO INITIALIZE THE INTEGRATOR AND FROM THIS POINT CAPE VECTORS ARE GENERATED UP TO THE REQUESTED LANDING REVOLUTION NUMBER MINUS ONE (1).

THE INTEGRATOR IS THEN INITIALIZED TO SEARCH FOR THE DESIRED LONGITUDE AT WHICH POINT THE TIME THIS LONGITUDE IS REACHED IS PRINTED ON LINE.

USAGE

MYGMT6 IS QUEUED BY MANUAL INSERTION AND NORMALLY EXITS TO MOPRIO.

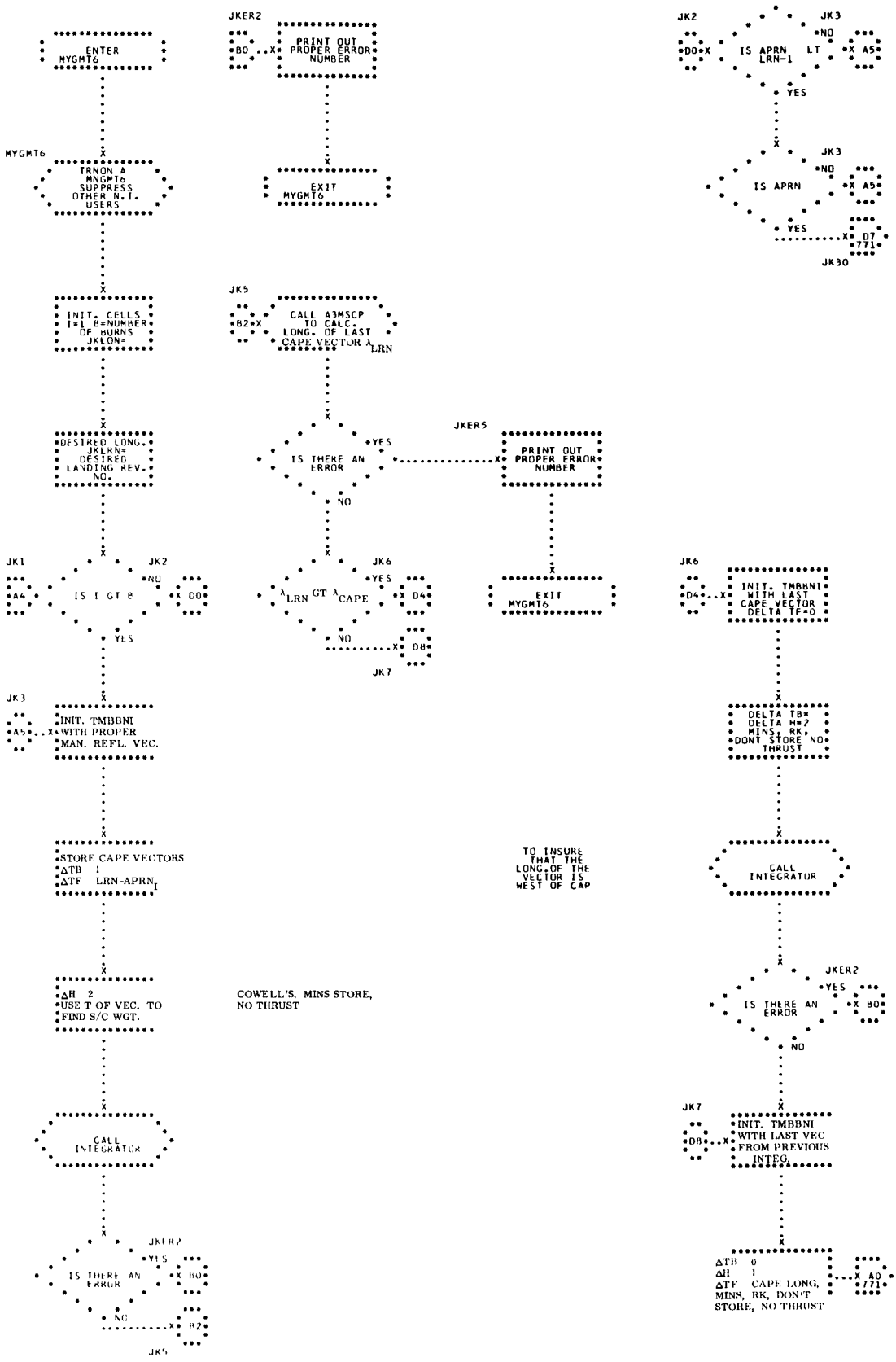
A) STORAGE REQUIRED -

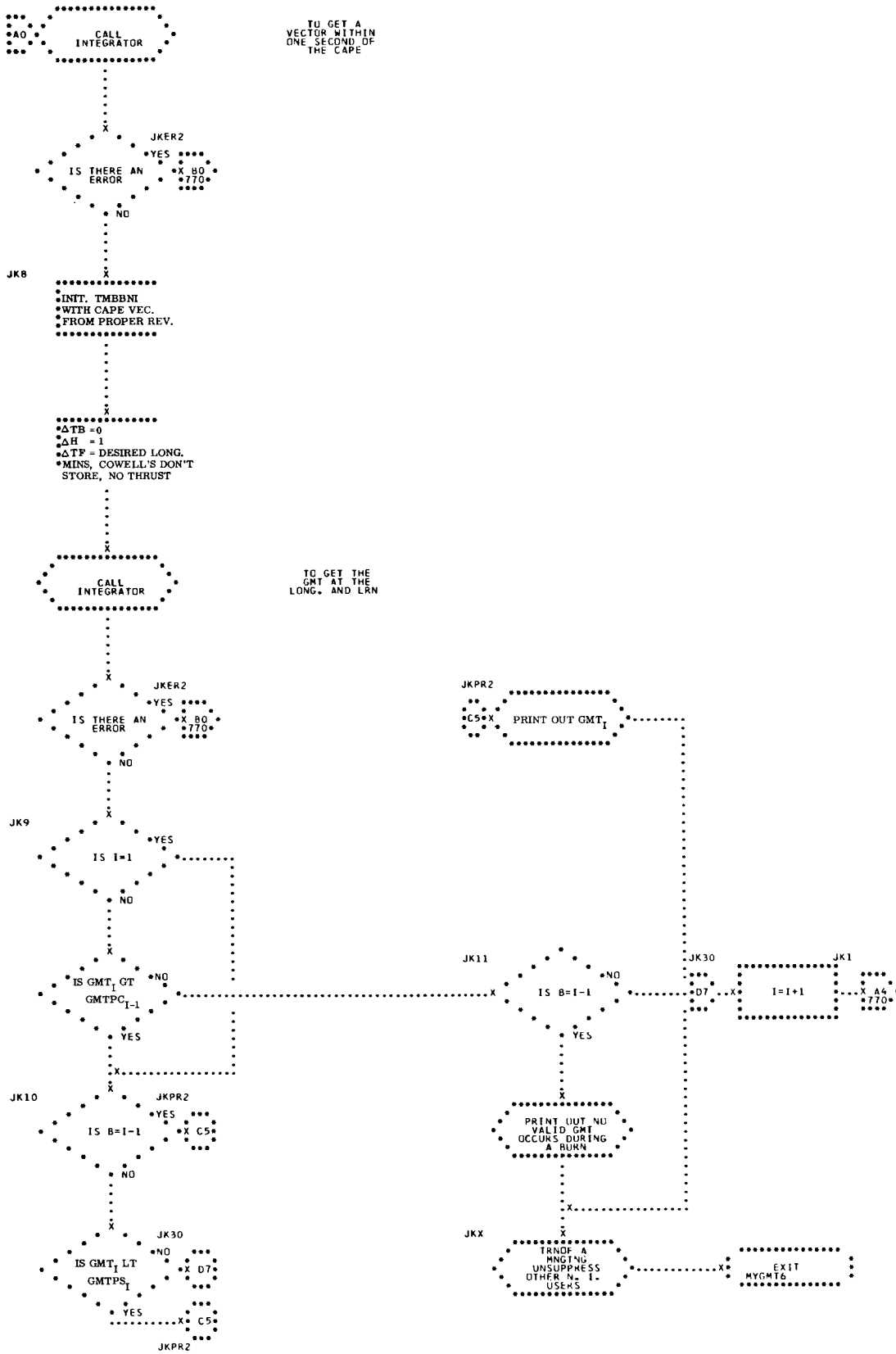
407 LOCATIONS USED FOR THE FOLLOWING -

1) 346 INSTRUCTIONS
2) 14 CONSTANTS
3) 47 CELLS

B) ERROR CODES -

1 = REQUESTED LANDING REVOLUTION NUMBER GREATER THAN 23
2 = ERROR IN INTEGRATION
3 = IMPACT DURING INTEGRATION
4 = NOT ENOUGH CAPE VECTORS IN TABLE
5 = ERROR IN A3MSCP


ENTER MYGMT6
MYGMT6
TRNON A MNGMT6 SUPPRESS OTHER N.I. USERS
INIT. CELLS I=1 B=NUMBER OF BURNS JKLON=
DESIRED LONG. JKLRN= DESIRED LANDING REV. NO.
JK1
IS I GT B
NO
YES
JK2
JK3
INIT. TMBBNI WITH PROPER MAN. REFL. VEC.
STORE CAPE VECTORS ΔTB 1 ΔTF LRN-APRN I
ΔH 2 USE T OF VEC. TO FIND S/C WGT.
CALL INTEGRATOR
IS THERE AN ERROR
YES
NO
JKER2
B2
JK5
JKER2
PRINT OUT PROPER ERROR NUMBER
EXIT MYGMT6
JK5
CALL A3MSCP TO CALC. LONG. OF LAST CAPE VECTOR λLRN
IS THERE AN ERROR
YES
NO
JKER5
JK6
λLRN GT λCAPE
YES
NO
JK7
JKER5
PRINT OUT PROPER ERROR NUMBER
EXIT MYGMT6
TO INSURE THAT THE LONG. OF THE VECTOR IS WEST OF CAP
COWELL'S, MINS STORE, NO THRUST
JK2
IS APRN LRN-1 LT
NO
YES
JK3
IS APRN
NO
YES
D7 771
JK30
JK6
INIT. TMBBNI WITH LAST CAPE VECTOR DELTA TF=0
DELTA TB= DELTA H=2 MINS, RK, DONT STORE NO THRUST
CALL INTEGRATOR
IS THERE AN ERROR
YES
NO
JKER2
JK7
INIT. TMBBNI WITH LAST VEC FROM PREVIOUS INTEG.
ΔTB 0
ΔH 1
ΔTF CAPE LONG, MINS, RK, DON'T STORE, NO THRUST
A0 771


A0
CALL INTEGRATOR
TO GET A VECTOR WITHIN ONE SECOND OF THE CAPE
JKER2
IS THERE AN ERROR
YES
X B0 *770*
NO
JK8
INIT. TMBBNI
WITH CAPE VEC.
FROM PROPER REV.
ΔTB =0
ΔH = 1
ΔTF = DESIRED LONG.
MINS, COWELL'S DON'T STORE, NO THRUST
CALL INTEGRATOR
TO GET THE GMT AT THE LONG. AND LRN
JKER2
IS THERE AN ERROR
YES
X B0 *770*
NO
JK9
IS I=1
YES
NO
IS GMT$_I$ GT GMTPC$_{I-1}$
NO
YES
JK10
JKPR2
IS B=I-1
YES
X C5
NO
JK30
IS GMT$_I$ LT GMTPS$_I$
NO
X D7
YES
X C5
JKPR2
JKPR2
C5 X
PRINT OUT GMT$_I$
JK11
IS B=I-1
NO
YES
PRINT OUT NO VALID GMT OCCURS DURING A BURN
JK30
D7
X
I=I+1
JK1
X A4 *770*
JKX
TRNOF A MNGING UNSUPPRESS OTHER N. I. USERS
X
EXIT MYGMT6

17. MYMUV7 - MANEUVER PARAMETER COMPUTATION CONTROL PROCESSOR

MYMUV7 PROVIDES THE CONTROL LOGIC NECESSARY TO COMPUTE THE PARAMETERS NEEDED TO PERFORM PRESCRIBED ORBITAL SPACECRAFT MANEUVERS. THIS INCLUDES THE SELECTION OF THE PROPER MANEUVER REFLECTING VECTOR FOR INTEGRATION UP TO THE PREDETERMINED TIME TO INITIATE THE MANEUVER, THE ON-LINE PRINTING AND DISPLAYING ON THE PROPULSION PANEL OF THE COMPUTED PARAMETERS, AND THE ON-LINE PRINTING OF ANY ERROR NUMBERS WHICH MAY ARISE DURING THE COURSE OF THE PROGRAM. (SEE FLOWCHART.)

INPUT

A) SUBROUTINES -

VOMANV

GEMWGT

B) PROGRAMS -

MPCPNI

MOPRIO

C) CONSTANTS -

K02000	K00060	K00008
K00120	K00000	K00001
K80NMI	K00030	K.1RAD
K00.P1	KKK2PI	K86400
K00002	K00003	

D) TABLES -

TMREFL DEFINED IN PROGRAM DESCRIPTION FOR GVMMRF

TMANUV DEFINED IN PROGRAM DESCRIPTION FOR GVMMRF

TIPROP DEFINED IN VOMANV
TUMANV N. I. INPUT BUFFER
TXINTK N. I. OUTPUT BUFFER
TMORMC PROPULSION PANEL DISPLAY BUFFER
TMUV7K ON-LINE PRINT BUFFER

E) MACROS -

QUEUE
QTSX
TRNON
TRNOF
QENBZ
QENBA

F) COMMUNICATION CELLS -

MCMANC NUMBER OF MANEUVERS
MCVPER VACUUM PERIGEE OF CURRENT ORBIT DEFINING VECTORS

OUTPUT

THE CALCULATED PITCH, YAW, TIME TO INITIATE THE MANEUVER, ΔT, AND ΔV ARE PRINTED ON-LINE AND DISPLAYED ON THE PROPULSION PANEL.

METHOD

THE PROPER MANEUVER REFLECTING VECTOR IS CHOSEN FROM TMREFL. THIS VECTOR IS USED IN INITIATING INTEGRATION UP TO THE TIME TO INITIATE THE MANEUVER. THE RESULTANT VECTOR FROM THIS INTEGRATION IS THEN USED IN THE SUBROUTINE VOMANV WHICH ACTUALLY CALCULATES THE MANEUVER PARAMETERS. CONTROL IS THEN RETURNED TO MYMUV7 AND THE COMPUTED QUANTITIES ARE PRINTED ON-LINE AND DISPLAYED ON THE PROPULSION. ERROR NUMBERS COULD ALSO BE PRINTED AT THIS TIME IF ANY ERROR CONDITION HAD

ARISEN. CONTROL IS THEN RETURNED TO MOPRIO.

USAGE

MYMUV7 IS QUEUED BY MANUAL INSERTION AND NORMALLY EXITS TO MOPRIO.

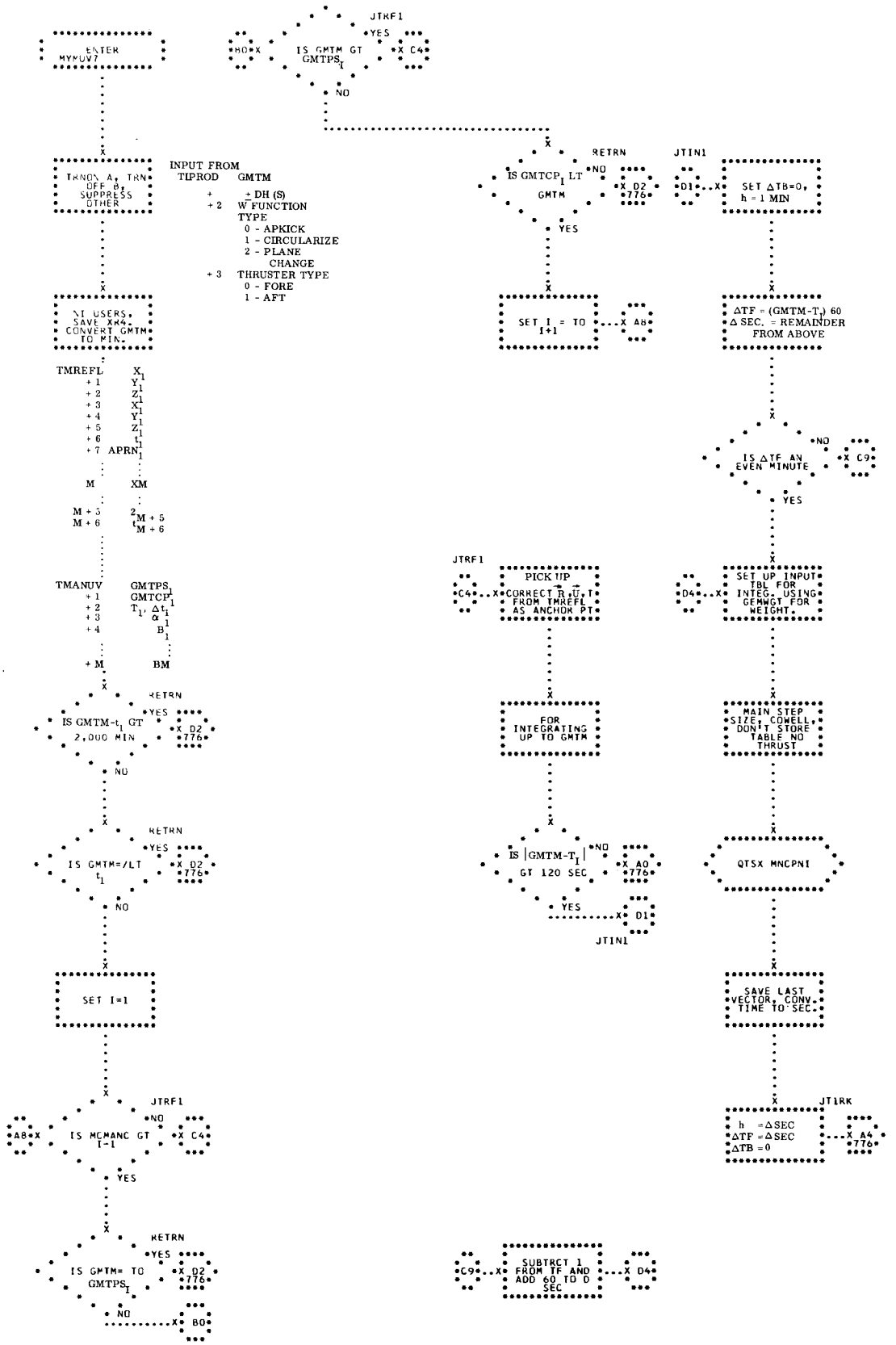
A) STORAGE REQUIRED -

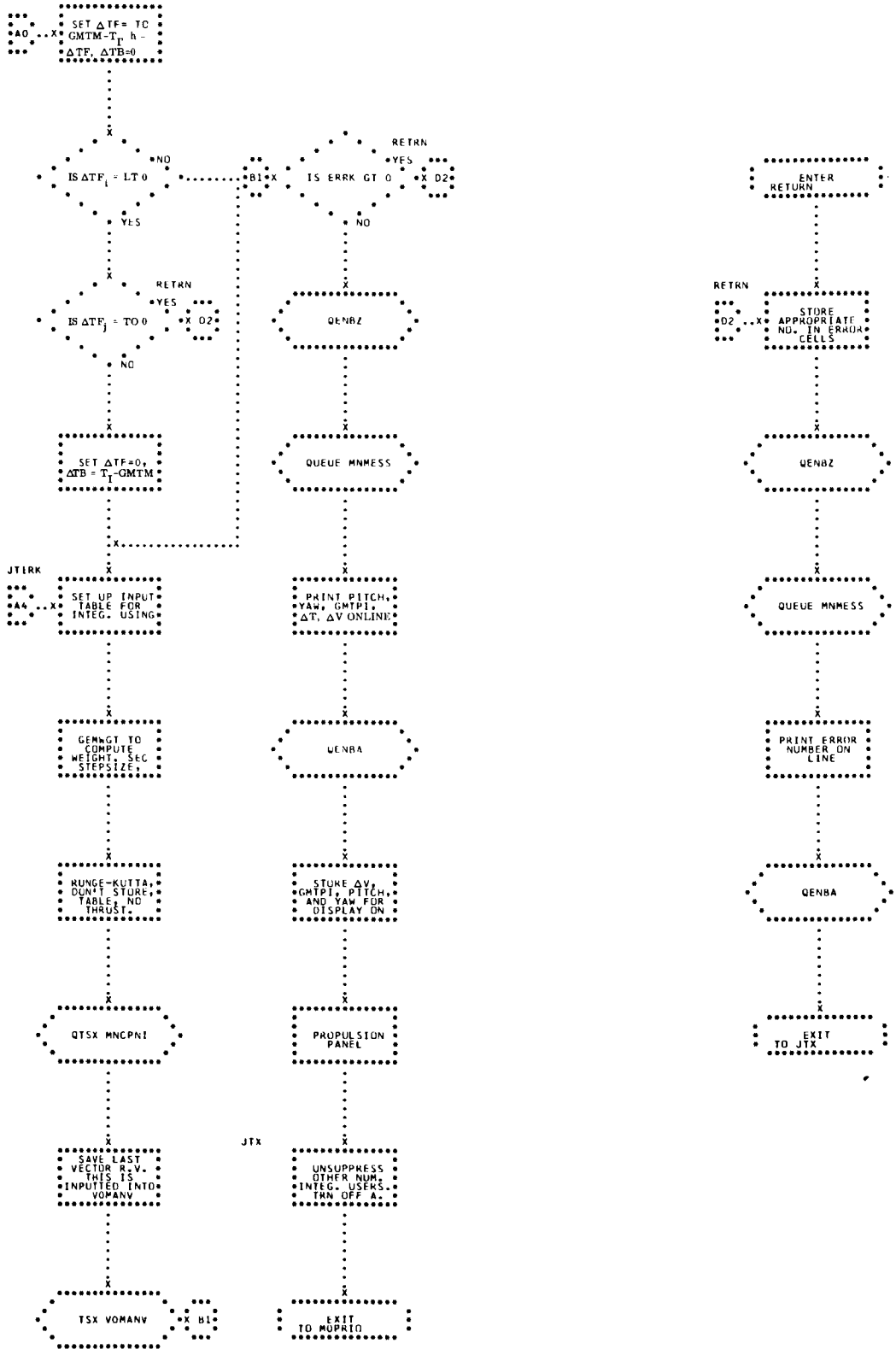
253 LOCATIONS

B) ERROR CODES -

1. INTEGRATION ERROR OR MANEUVER REQUEST MORE THAN 22 ORBITS AWAY.
2. MANEUVER REQUESTED DURING PLANNED BURN
3. GMTM IS LESS THAN THE TIME OF INITIAL VECTOR IN TMREFL

15. $\Delta V = 0$
18. LOOPING ERROR - NO CONVERGENCE
20. TIME ERROR FROM INTEGRATOR


ENTER MYMUV7
TRNON A, TRN OFF B, SUPPRESS OTHER
NI USERS, SAVE XR4, CONVERT GMTM TO MIN.
TMREFL
+1 X1
+2 Y1
+3 Z1
+4 X1
+5 Y1
+6 Z1
+7 APRN1
M XM
M+5 2M+5
M+6 tM+6
TMANUV
+1 GMTPSI
+2 GMTCPI
+3 T1, Δt1
+4 α1
B1
+M BM
IS GMTM-t1 GT 2,000 MIN
IS GMTM=/LT t1
SET I=1
IS MCMANC GT I-1
IS GMTM= TO GMTPSI
JTRF1
IS GMTM GT GMTPSI
INPUT FROM TIPROD GMTM
+ ± DH (S)
+2 W FUNCTION TYPE
0 - APKICK
1 - CIRCULARIZE
2 - PLANE CHANGE
+3 THRUSTER TYPE
0 - FORE
1 - AFT
IS GMTCPI LT GMTM
SET I = TO I+1
JTIN1
SET ΔTB=0, h = 1 MIN
ΔTF = (GMTM-T1) 60 Δ SEC. = REMAINDER FROM ABOVE
IS ΔTF AN EVEN MINUTE
PICK UP CORRECT R, V, T FROM TMREFL AS ANCHOR PT
FOR INTEGRATING UP TO GMTM
IS |GMTM-TI| GT 120 SEC
SET UP INPUT TBL FOR INTEG. USING GENWGT FOR WEIGHT.
MAIN STEP SIZE, COWELL, DON'T STORE TABLE NO THRUST
QTSX MNCPNI
SAVE LAST VECTOR, CONV. TIME TO SEC.
h =ΔSEC
ΔTF =ΔSEC
ΔTB =0
JT1RK
SUBTRCT 1 FROM TF AND ADD 60 TO D SEC
RETRN
YES
NO


A0
SET ΔTF= TC GMTM-T_I h - ΔTF, ΔTB=0
IS ΔTF_i = LT 0
NO
YES
B1
IS ERRK GT 0
RETRN
YES
D2
NO
IS ΔTF_j = TO 0
RETRN
YES
D2
NO
QENBZ
SET ΔTF=0, ΔTB = T_I-GMTM
QUEUE MNMESS
JTIRK
A4
SET UP INPUT TABLE FOR INTEG. USING
PRINT PITCH, YAW, GMTPI, ΔT, ΔV ONLINE
GENGT TO COMPUTE WEIGHT, SEC STEPSIZE,
QENBA
RUNGE-KUTTA, DON'T STORE, TABLE, NO THRUST.
STORE ΔV, GMTPI, PITCH, AND YAW FOR DISPLAY ON
QTSX MNCPNI
PROPULSION PANEL
JTX
SAVE LAST VECTOR R,V. THIS IS INPUTTED INTO VOMANV
UNSUPPRESS OTHER NUM. INTEG. USERS. TRN OFF A.
TSX VOMANV
B1
EXIT TO MOPRIO
ENTER RETURN
RETRN
D2
STORE APPROPRIATE NO. IN ERROR CELLS
QENBZ
QUEUE MNMESS
PRINT ERROR NUMBER ON LINE
QENBA
EXIT TO JTX

18. MYLIF8 - REVOLUTION CAPABILITY MONITOR CONTROL PROCESSOR

MYLIF8 IS USED TO PROVIDE AN ESTIMATE OF THE REVOLUTION CAPABILITY OF AN EARTH-ORBITING SPACECRAFT BASED UPON AN INPUT TIME AND A POSITION AND VELOCITY VECTOR. THIS CAPABILITY IS DETERMINED IN CONJUNCTION WITH THREE PARAMETERS, LIFETIME DENSITY FACTOR, REVOLUTION LIMIT AND IMPACT ALTITUDE. THE DENSITY FACTOR ESTABLISHES THE ATMOSPHERIC DRAG CHARACTERISTICS, REVOLUTION LIMIT PROVIDES A BOUND FOR USE AS A CAPABILITY VERIFIER AND IMPACT ALTITUDE SETS A LIMIT BELOW WHICH THE SENSUOUS ATMOSPHERE RAPIDLY DECAYS THE ORBITAL TRAJECTORY. WITH THIS INFORMATION, THE INPUT ORBIT IS INTEGRATED FORWARD UNTIL IT EITHER REACHES THE REVOLUTION LIMIT OR THE IMPACT ALTITUDE. THE RESULTS ARE THEN FORMATTED FOR OUTPUT TO THE APPROPRIATE DISPLAY UNITS. (SEE FLOWCHART)

INPUT

THE 8 BIT IS TURNED ON BY MYOUR1 AND MFMN10.

A) TABLES -

TMORBD	CONTAINING THE PRESENT ORBIT DEFINING VECTOR
TIDRAG	CONTAINING LIFETIME PARAMETERS

B) COMMUNICATION CELLS -

MCOWGT	CONTAINING PRESENT S/C WEIGHT

OUTPUT

A) TABLES -

TMCAPB	ADDRESS CONTAINS REVOLUTION NUMBER OF INPUT VECTOR. DECRE-

MENT CONTAINS REVOLUTION NUMBER OF LAST GENERATED CAPE VECTOR.

TMORMC+15 CONTAINS COMPUTED REVOLUTION CAPABILITY

TPLIF8 USED FOR PRINTING ERROR INFORMATION

B) COMMUNICATION CELL -

MCNIAC CONTAINS ERROR NUMBER IN EVENT OF AN INTEGRATOR ERROR

C) ERROR CONDITION -

MYMESS IS QUEUED WITH AN ERROR MESSAGE IN EVENT OF A PROGRAM ERROR.

MYMESS IS QUEUED WITH A MESSAGE STATING THE INTEGRATED REVOLUTION CAPABILITY.

METHOD

UPON PROGRAM ENTRY, ALL OTHER INTEGRATOR USERS ARE H SUPPRESSED AND TMORBD IS SET UP AS INPUT TO THE INTEGRATOR. A QTSX IS THEN MADE TO THE INTEGRATOR INDICATING A TYPE NUMBER 5 INTEGRATION. WHEN COMPLETED, OUTPUT CELLS ARE UPDATED FOR DISPLAY AND MYMESS IS QUEUED FOR AN ON-LINE PRINT. THE H SUPPRESSED ROUTINES ARE RELEASED AND CONTROL RETURNS TO MOPRIO.

IN THE EVENT OF AN INTEGRATOR ERROR, MYMESS IS QUEUED TO PRINT ON-LINE THE ROUTINE NAME AND THE ERROR CODE.

IN DETERMINING REVOLUTION CAPABILITY, THE COWELL'S DOUBLE-PRECISION INTEGRATOR, WITH A TWO-MINUTE STEP SIZE, IS USED. CAPE VECTORS ARE COUNTED UNTIL EITHER THE REQUESTED LIMIT IS REACHED OR IMPACT OCCURS. THE OUTPUT INFORMATION IS THEN PLACED INTO TMCAPB.

USAGE

ENTRY FROM AND EXIT TO MOPRIO

MYLIF8 USES -

A) COMMUNICATION CELLS -

MCWOGT

MCNIAC

B) TABLES -

TMCRBD

TIDRAG

TMORMC

TMCAPB

TPLIF8

C) MACROS -

TRNON

TRNOF

QTSX

QENBZ

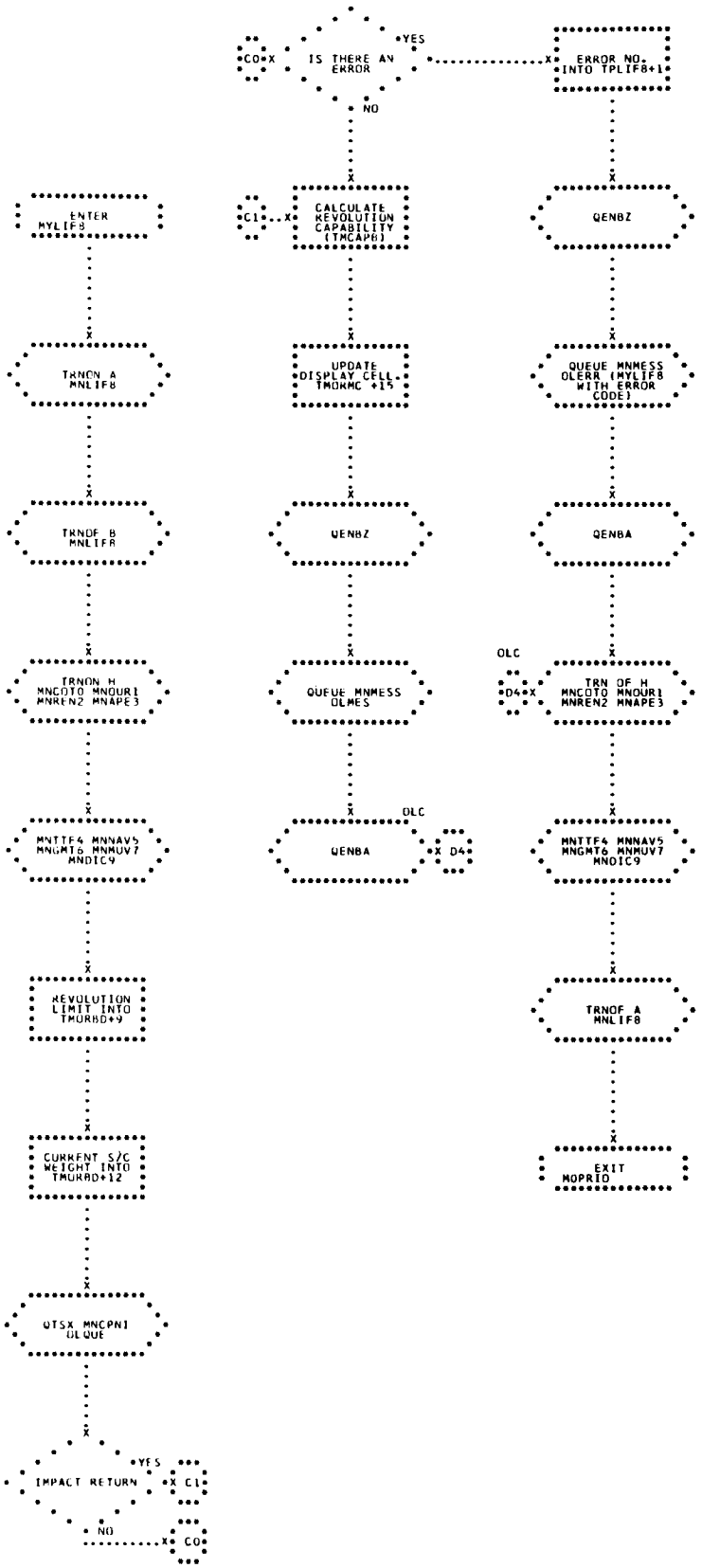
QENBA

QUEUE

D) PARAMETERS -

MNCOTO	MNMUV7
MNOUR1	MNLIF8
MNREN2	MNDIC9
MNAPE3	MNCPNI
MNTTF4	MNMESS
MNNAV5	MNBBNI
MNGMT6	A, B, H

E) STORAGE REQUIRED - 60 LOCATIONS


IS THERE AN ERROR
YES
NO
CO
ERROR NO. INTO TPLIF8+1
ENTER MYLIF8
CALCULATE REVOLUTION CAPABILITY (TMCAP8)
C1
QENBZ
TRNON A MNLIF8
UPDATE DISPLAY CELL TMORMC +15
QUEUE MNMESS OLERR (MYLIF8 WITH ERROR CODE)
TRNOF B MNLIF8
QENBZ
QENBA
TRNON H MNCOTO MNOUR1 MNREN2 MNAPE3
QUEUE MNMESS OLMES
OLC
D4
TRN OF H MNCOTO MNOUR1 MNREN2 MNAPE3
MNTTF4 MNNAV5 MNGMT6 MNMUV7 MNDIC9
QENBA
REVOLUTION LIMIT INTO TMORBD+9
TRNOF A MNLIF8
CURRENT S/C WEIGHT INTO TMORBD+12
EXIT MOPRIO
QTSX MNCPN1 OLQUE
IMPACT RETURN

19. MYDIC9 - DIFFERENTIAL COORECTION MONITOR CONTROLLER

MYDIC9 INITIATES CONTROL FUNCTIONS FOR NOCPNI WHEN DIFFERENTIAL CORRECTION REQUIRES THE NEED OF A VECTOR OR A TABLE OF VECTORS. (SEE FLOWCHART.)

INPUT

INPUT CONSISTS OF A UNQUEUE WORD.

OUTPUT

OUTPUT CONSISTS OF A QUEUE GIVEN TO THE NUMERICAL INTEGRATOR, NOCPNI.

METHOD

SEE FLOW CHART.

USAGE

A) MYDIC9 ENTERS FROM AND EXIT TO MOPRIO

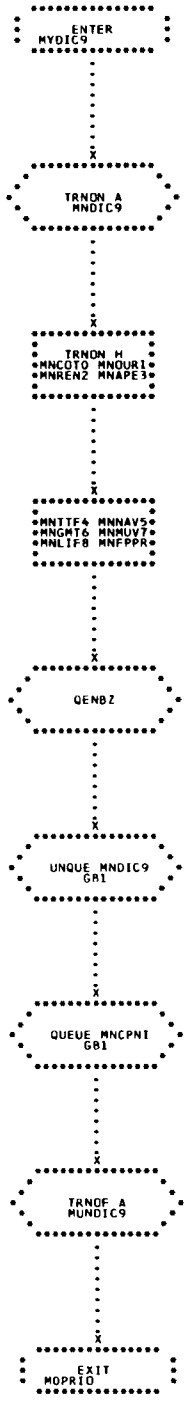
B) MACROS
   - TRNON
   - TRNOF
   - QENBZ
   - QUEUE
   - UNQUE

C) TABLE -
   - TMPRIO

D) PARAMETERS -
   - A
   - H
   - MNCOTO
   - MNOUR1

MNREN2
MNAPE3
MNTTF4
MNNAV5
MNGMT6
MNMUV7
MNLIF8
MNDIC9
MNFPPR
MNCPNI

E) STOARGE REQUIRED - 21 LOCATIONS


ENTER
MYDIC9
TRNON A
MNDIC9
TRNON H
MNCOTO MNOUR1
MNREN2 MNAPE3
MNTTF4 MNNAV5
MNGMT6 MNMUV7
MNLIF8 MNFPPR
QENBZ
UNQUE MNDIC9
GR1
QUEUE MNCPNI
GR1
TRNOF A
MUNDIC9
EXIT
MOPRIO

20. MYQPRA - MONITOR QUEUE NEXT PRIMARY AREA PROCESSOR

MYQPRA QUEUES MYTTF4 TO INITIATE A TIME-TO-FIRE RETRO COMPUTATION FOR THE NEXT PRIMARY AREA IN THE TABLE. (SEE FLOWCHART)

INPUT

A) TABLES -

TMLMPT TABLE OF LONGITUDES AND LATITUDES OF ALL PRIMARY RECOVERY AREAS, STORED SUCCESSIVELY IN RADIANS, LONGITUDES BETWEEN 0 AND 2π.

TMPLON TABLE ASSOCIATED WITH TMLMPT, CONTAINING THE REVOLUTION NUMBER OF THE PRIMARY AREA IN THE DECREMENT, AND SOME IDENTIFYING NUMBER IN THAT REVOLUTION IN THE ADDRESS.

TMDELV TABLE OF ΔV'S PER REVOLUTION, INDICATING THE AMOUNT OF THE PRE-RETRO BURN FOR ALL TARGETS IN THAT REVOLUTION.

TMORMC INPUT TABLE TO THE OUTPUT ROUTINE FOR DISPLAYS AND PLOTS. ONLY THE SECOND WORD (LONGITUDE OF PRESENT POSITION) IS USED HERE.

TDRDWT TABLE OF TTF PARAMETERS. THE NINTH WORD CONTAINS THE TIME (7 MINUTES) BEFORE THE CURRENTLY DISPLAYED GMTRC SHOULD BE UPDATED. THE 10TH WORD CONTAINS THE DELTA LONGITUDE TO BE USED ON A RESTART IN DETERMINING WHICH TARGETS ARE TOO LATE TO HIT.

TOSCCH S/C CHARACTERISTC TABLE. THE 32ND WORD CONTAINS THE BANK ANGLE TO BE USED IN COMPUTING GMTRC FOR CONTINGENCY AND PRIMARY AREAS.

B) COMMUNICATION CELLS -

MCREVN CURRENT REVOLUTION NUMBER.

MCTSPA CURRENTLY DISPLAYED GMTRC FOR THE STORED PRIMARY AREA.

MCHFSC CURRENT GMT TIME IN HALF-SECONDS

MCMINS CURRENT GMT TIME IN MINUTES

MCERRP IF AN ERROR OCCURRED WHILE CALCULATING A PRIMARY AREA GMTRC, THIS CELL CONTAINS THE GMT AT THAT TIME, OTHERWISE ZERO.

C) PARAMETER -

MNNUPA EQUAL TO THE NUMBER OF RECOVERY AREAS IN TMLMPT

D) CONSTANTS -

KLCAPE

KKK2PI

E) MACROS -

UNQUE

QUEUE

TRNON

TRNOF

QENBZ

QENBA

OUTPUT

A) AN ENTRY IS PUT INTO THE QUEUE FOR MYTTF4, CONSISTING OF A TABLE LOCATION IN THE ADDRESS, REVOLUTION NUMBER OF THE TARGET IN THE DECREMENT, + 0 IN THE PREFIX.

B) TABLE -

TMPRIO INPUT TABLE TO MYTTF4, CONSISTING OF λ_T, ϕ_T, B, ΔV IN THAT ORDER

C) COMMUNICATION CELL -

MCRECP CONTAINS THE RECOVERY AREA CODE, LRN AND NAME OF TARGET.

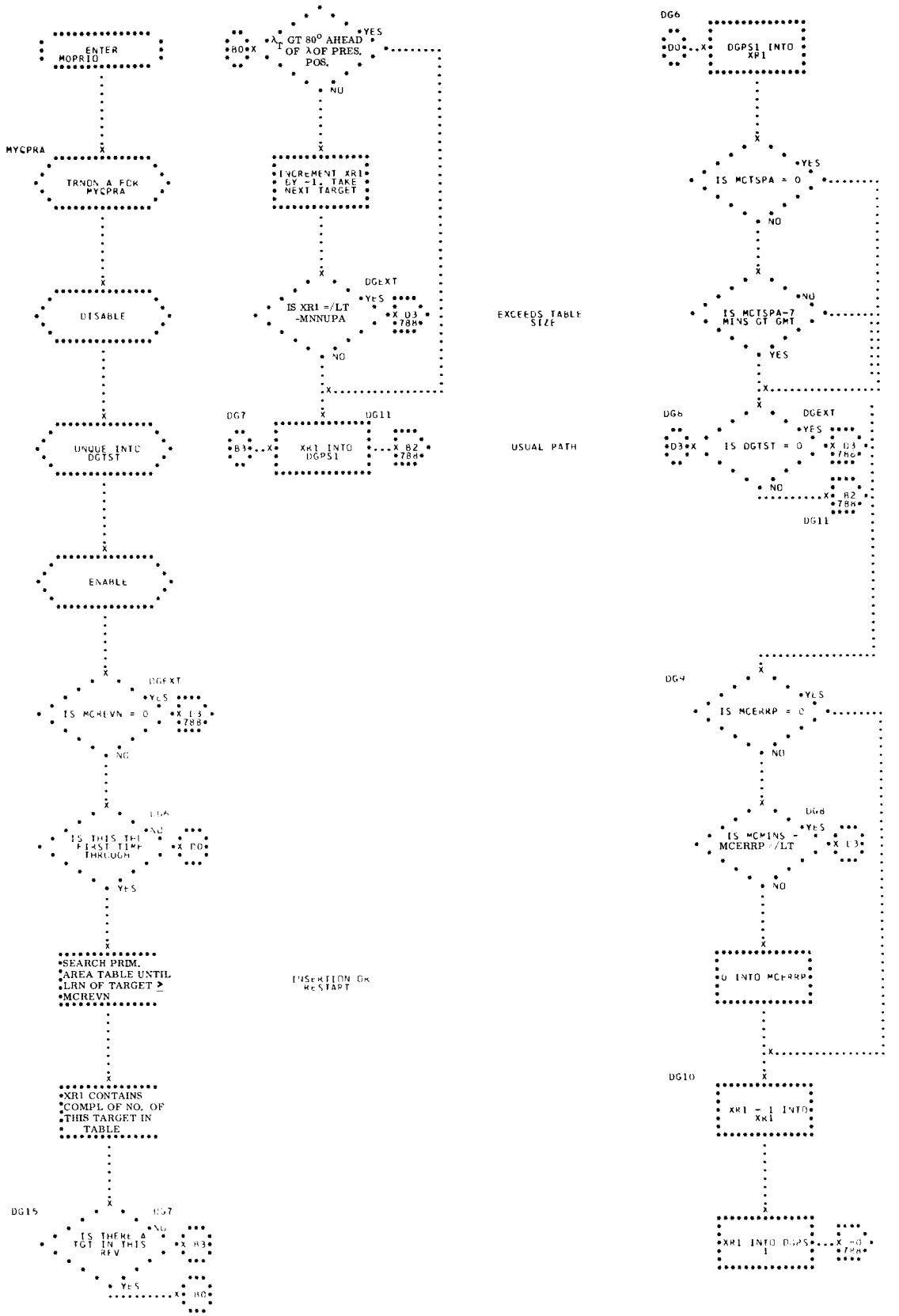
METHOD

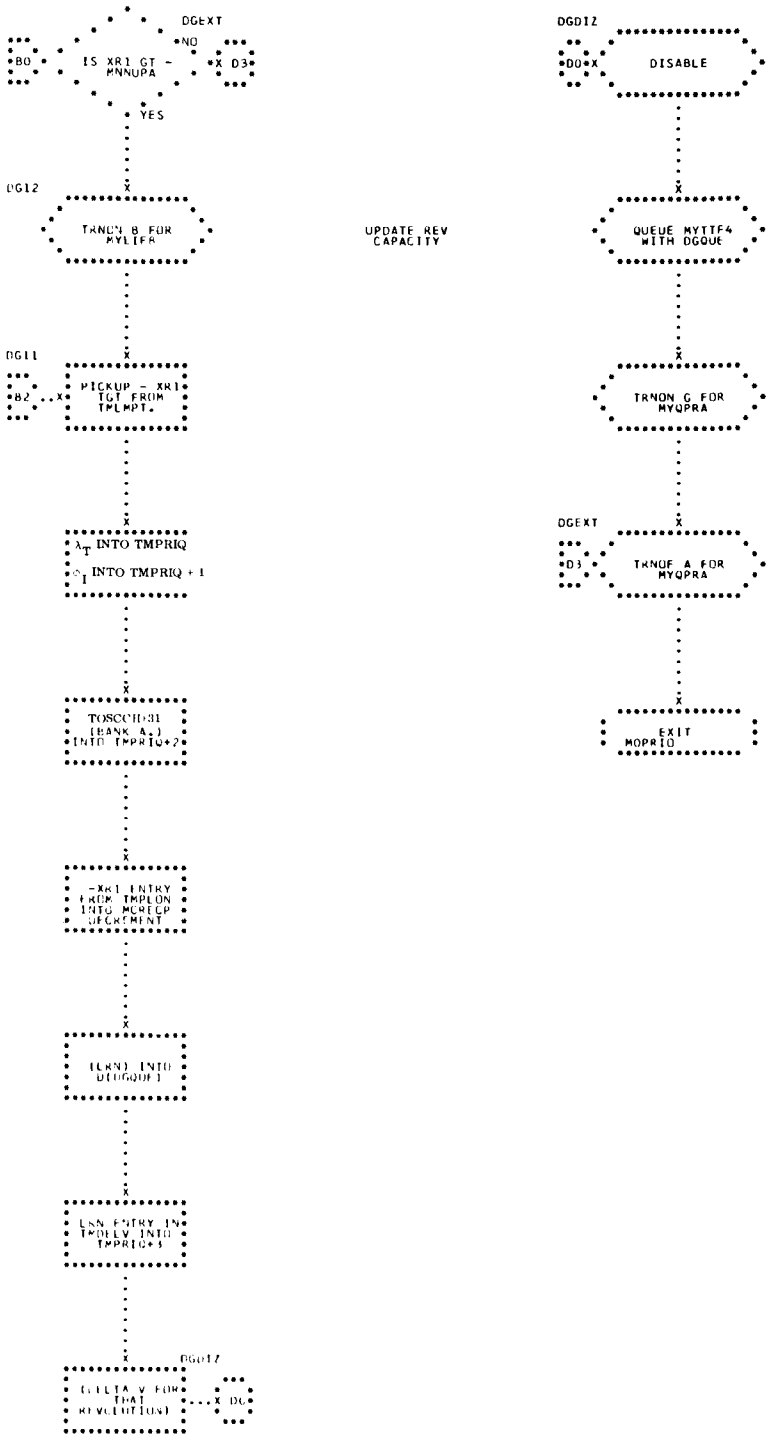
IF IN A RESTART OR ON INSERTION INTO ORBIT, MYQPRA WILL SEARCH THE TABLE OF PRIMARY AREAS SELECTING THE FIRST ONE WHICH HAS A LRN GREATER OR EQUAL TO THE CURRENT ORBIT NUMBER, UNLESS THE LONGITUDE OF THE TARGET IS WITHIN 80° OF THE LONGITUDE OF PRESENT POSITION, IN WHICH CASE THE NEXT TARGET IN THE TABLE IS CHOSEN. IF NO RESTART, THE CURRENTLY DISPLAYED GMTRC FOR THE STORED PRIMARY AREA IS CHECKED TO SEE IF IT IS WITHIN 7 MINUTES OF CURRENT GMT. IF IT IS NOT AND THE ORBIT HAS CHANGED, MYTTF4 IS QUEUED TO UPDATE THE CURRENT GMTRC. IF NO ORBIT CHANGE, MERELY EXIT. IF IT IS WITHIN 7 MINUTES, THE SUCCEEDING TARGET IN THE TABLE IS CHOSEN AND THE INPUT QUEUE FOR MYTTF4 IS FORMED AND MYTTF4 IS QUEUED.

USAGE

A) STORAGE REQUIRED - 100 CELLS

B) TIMING - MAXIMUM, WITH THE EXCEPTION OF FIRST TIME THROUGH ON RESTART - .120 MS


ENTER MOPRIO
MYCPRA
TRNON A FOR MYCPRA
DISABLE
UNQUE INTO DGTST
ENABLE
DGEXT
IS MCREVN = 0
YES
NO
DG6
IS THIS THE FIRST TIME THROUGH
SEARCH PRIM. AREA TABLE UNTIL LRN OF TARGET ≥ MCREVN
XR1 CONTAINS COMPL OF NO. OF THIS TARGET IN TABLE
DG15
DG7
IS THERE A TGT IN THIS REV
B0
B3
λ_T GT 80° AHEAD OF λOF PRES. POS.
INCREMENT XR1 BY -1. TAKE NEXT TARGET
DGEXT
IS XR1 =/LT -MNNUPA
D3 788
DG7
DG11
XR1 INTO DGPS1
B2 788
EXCEEDS TABLE SIZE
USUAL PATH
INSERTION OR RESTART
DG6
DGPS1 INTO XR1
IS MCTSPA = 0
IS MCTSPA-7 MINS GT GMT
DGEXT
IS DGTST = 0
D3 788
DG11
DG9
IS MCERRP = 0
DG8
IS MCMINS - MCERRP =/LT
D3
0 INTO MCERRP
DG10
XR1 - 1 INTO XR1
XR1 INTO DGPS 1
B0 788


DGEXT
NO
IS XR1 GT - MNNUPA
BO
X D3
YES
DG12
TRNON B FOR MYLIFR
DG11
PICKUP - XR1 TGT FROM TMLMPT.
B2
X
λ_T INTO TMPRIQ
Δ_I INTO TMPRIQ + 1
TOSCCH031 (BANK A.) INTO TMPRIQ+2
-XR1 ENTRY FROM TMPLON INTO MCREGP DECREMENT
(ERN) INTO D(DGQUE)
LNN ENTRY IN TMDELV INTO TMPRIQ+5
DGDIZ
DELTA V FOR THAT REVOLUTION
DG
UPDATE REV CAPACITY
DGDIZ
DO
X
DISABLE
QUEUE MYTIF4 WITH DGQUE
TRNON G FOR MYQPRA
DGEXT
D3
TRNOF A FOR MYQPRA
EXIT MOPRIO

21. MYQNEA - MONITOR QUEUE NEXT EMERGENCY AREA PROCESSOR

MYQNEA SELECTS THE APPROPRIATE CONTINGENCY RECOVERY AREA FROM THE STORED TABLE OF CONTINGENCY AREAS AND QUEUES MYTTF4 TO INITIATE A GMTRC COMPUTATION FOR THAT AREA. (SEE FLOWCHART.)

INPUT

A) TABLES -

TMAREA	TABLE CONTAINING LONGITUDES OF ALL CONTINGENCY (EMERGENCY) RECOVERY AREAS FOR THE MISSION.
TMPNRA	TABLE ASSOCIATED WITH TMAREA CONTAINING CORRESPONDING REVOLUTION NUMBER OF THE RECOVERY AREAS ALONG WITH AN INDICATION OF THE NAME OF THE AREA AND AN ASSOCIATED DELAY IN DISPLAYING THEM.
TMDELV	TABLE OF ΔV'S (VELOCITY INCREMENTS) PER REVOLUTION TO BE USED IN THE PRE-RETRO OAMS MANEUVER.
TDRDWT	TABLE OF PARAMETERS, THE 11TH WORD CONTAINING 80° IN RADIANS TO BE USED IN DETERMINING WHICH TARGETS TO TAKE.
TOSCCH	TABLE OF MISSION PARAMETERS, THE 32ND WORD CONTAINING THE BANK ANGLE TO BE USED IN TTF COMPUTATION.
TMORMC	SECOND WORD CONTAINS LONGITUDE OF PRESENT POSITION, USED HERE.

B) COMMUNICATION CELLS -

MCREVN	CONTAINS CURRENT REVOLUTION NUMBER.
MCHFSC	CONTAINS CURRENT TIME IN HALF-SECONDS.
MCMINS	CONTAINS CURRENT TIME IN MINUTES.
MCTNEA	CONTAINS CURRENTLY DISPLAYED GMTRC FOR THE NEXT CONTINGENCY AREA.
MCNOGT	IF AN ERROR OCCURED WHILE COMPUTING GMTRC FOR THE NEXT CONTINGENCY AREA, THIS CELL CONTAINS THE GMT+ SOME AVERAGE TIME BETWEEN AREAS.

C) CONSTANTS -

KLCAPE

KKK2KI

D) MACROS -

UNQUE

QUEUE

QENBZ

QENBA

TRNON

TRNOF

E) PARAMETER -

MNPNRA	EQUAL TO THE NUMBER OF CONTINGENCY AREAS.

OUTPUT

A) TABLE -

TMLOCC	TABLE OF INPUT FOR MYTTF4, CONTAINING LONGITUDE OF THE TARGET, ϕ_T, BANK ANGLE AND ΔV IN THAT ORDER.

B) QUEUE WORD FOR MYTTF4, CONTAINING MZE IN THE PREFIX, REVOLUTION NUMBER OF THE TARGET IN THE DECREMENT AND TMLOCC IN THE ADDRESS.

C) COMMUNICATION CELL -

MCRECA CONTAINS THE RECOVERY AREA DIGITAL DISPLAY BITS.

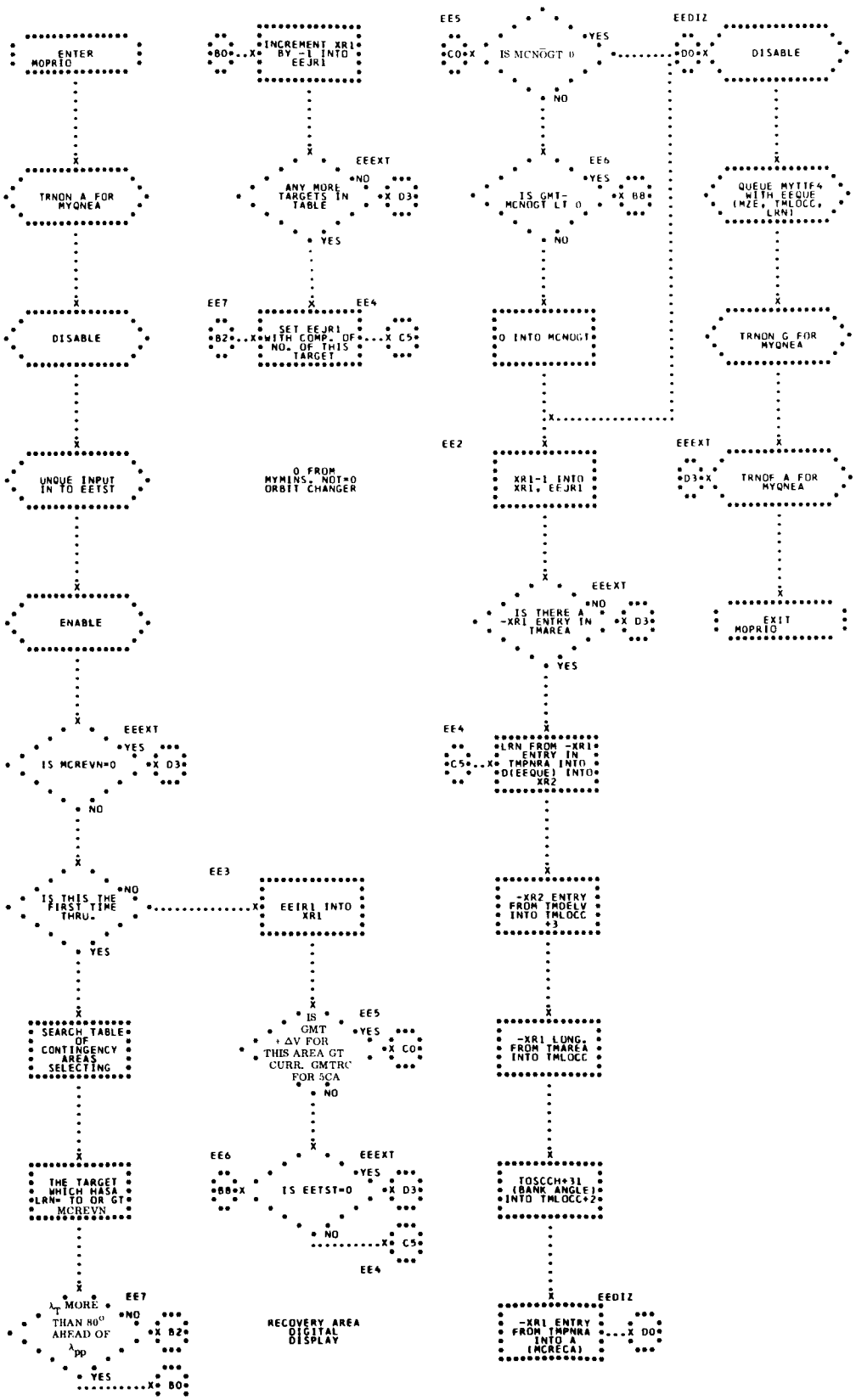
METHOD

IF IN A RESTART OR ON INSERTION INTO ORBIT, MYQNEA WILL SEARCH THE TABLE OF CONTINGENCY AREAS SELECTING THE FIRST ONE WHICH HAS A LRN GREATER OR EQUAL TO THE CURREN ORBIT NUMBER, UNLESS THE LONGITUDE OF THE TARGET IS WITH IN 80° OF THE LONGITUDE OF PRESENT POSITION, IN WHICH CASE THE NEXT TARGET IN THE TABLE IS CHOSEN. IF NO RESTART, THE CURRENTLY DISPLAYED GMTRC FOR THE STORED CONTINGENCY AREA IS CHECKED TO SEE IF IT IS WITHIN SOME VARIABLE DELAY OF THE CURRENT GMT. IF IT IS NOT AND THE ORBIT HAS CHANGED, MYTTF4 IS QUEUED TO UPDATE THE CURREN GMTRC. IF NO ORBIT CHANGE, MERELY EXIT. IF IT IS WITHI 7 MINUTES, THE SUCCEEDING TARGET IN THE TABLE IS CHOSEN AND THE INPUT QUEUE FOR MYTTF4 IS FORMED AND MYTTF4 IS QUEUED.

USAGE

STORAGE REQUIRED - 98 LOCATIONS

TIMING - MAXIMUM (EXCEPT FIRST TIME THROUGH ON RESTART) - .120 MS


ENTER MOPRIO
TRNON A FOR MYQNEA
DISABLE
UNQUE INPUT IN TO EETST
ENABLE
EEEXT
IS MCREVN=0
YES
D3
NO
IS THIS THE FIRST TIME THRU.
NO
YES
SEARCH TABLE OF CONTINGENCY AREAS SELECTING
THE TARGET WHICH HASA LRN= TO OR GT MCREVN
EE7
λT MORE THAN 80° AHEAD OF λpp
NO
B2
YES
B0
INCREMENT XR1 BY -1 INTO EEJR1
B0
EEEXT
ANY MORE TARGETS IN TABLE
NO
D3
YES
EE7
EE4
SET EEJR1 WITH COMP. OF NO. OF THIS TARGET
B2
C5
0 FROM MYMINS, NOT=0 ORBIT CHANGER
EEIR1 INTO XR1
EE3
EE5
IS GMT + ΔV FOR THIS AREA GT CURR. GMTRC FOR 5CA
YES
C0
NO
EE6
EEEXT
IS EETST=0
YES
D3
NO
C5
EE4
RECOVERY AREA DIGITAL DISPLAY
EE5
IS MCNOGT 0
YES
C0
NO
EE6
IS GMT-MCNOGT LT 0
YES
B8
NO
0 INTO MCNOGT
EE2
XR1-1 INTO XR1, EEJR1
EEEXT
IS THERE A -XR1 ENTRY IN TMAREA
NO
D3
YES
EE4
LRN FROM -XR1 ENTRY IN TMPNRA INTO D(EEQUE) INTO XR2
C5
-XR2 ENTRY FROM TMOELV INTO TMLOCC +3
-XR1 LONG. FROM TMAREA INTO TMLOCC
TOSCCH+31 (BANK ANGLE) INTO TMLOCC+2
EEDIZ
-XR1 ENTRY FROM TMPNRA INTO A (MCRECA)
D0
EEDIZ
D0
DISABLE
QUEUE MYTIF4 WITH EEQUE (MZE, TMLOCC, LRN)
TRNON G FOR MYQNEA
EEEXT
D3
TRNOF A FOR MYQNEA
EXIT MOPRIO

22. MPLED1 - MONITOR PREFIX TO EOLED1

MPLED1 PREPARES THE EDITING PARAMETERS FOR EOLED1 BY PROVIDING THE LOCATION OF INPUT RADAR MESSAGES. SPECIAL CONSIDERATION IS GIVEN TO BERMUDA AND THE DOWNRANGE STATIONS, I.E., CAPE KENNEDY, SAN SALVADOR AND GRAND BAHAMA. FOR FIRST-ORBIT BERMUDA DATA, AN INDICATOR IS SET SO THAT MFLED1 WILL SCREEN OUT POWERED-FLIGHT DATA. THEREFORE, FIRST-ORBIT DOWNRANGE DATA WILL NOT BE PROCESSED FURTHER. ON SUBSEQUENT ORBITS, THE ONLY DOWNRANGE RADAR DATA PROCESSED IS THAT FROM THE FIRST STATION COMPLETING TRANSMISSION. (SEE FLOWCHART.)

INPUT

A) AN ENTRY IN THE INPUT QUEUE TABLE FOR EOLED1 CONTAINING IN ITS ADDRESS THE BASE ADDRESS OF THE TMRMXX RADAR MESSAGE BLOCK PROVIDING THE INPUT.

B) MCHFSC CONTAINS CURRENT GMT IN FIXED POINT HALF-SECONDS.

C) MCGTLO CONTAINS GMT OF LIFTOFF IN FIXED POINT SECONDS.

D) MCMINS CONTAINS CURRENT GMT IN FIXED POINT MINUTES.

OUTPUT

A) MCLED1 CONTAINS IN ITS ADDRESS THE INTERNAL STATION NUMBER OF THE DATA TO BE EDITED.

B) TMRMXX TABLE OF INPUT DATA FOR EDITING.

C) MCLEDD CONTAINS IN ITS ADDRESS THE LOCATION OF THE TMRMXX INPUT TABLE.

D) MCCAP1 SET TO NONZERO TO INDICATE THAT FIRST-PASS BERMUDA DATA IS TO BE SCREENED FOR POWERED-FLIGHT BY MFLED1.

METHOD

MPLED1 RECEIVES CONTROL AFTER THE LOW-SPEED INPUT PROCESSOR HAS DETERMINED THE NEED TO EDIT THE RADAR DATA (THE END-OF-TRANSMISSION HAS BEEN DETECTED OR THE EIGHT MINUTES MAY HAVE ELAPSED SINCE TRANSMISSION BEGAN FOR THIS RADAR STATION). THE A INDICATOR FOR EOLED1 IS TURNED ON AND MYREST IS SUPPRESSED UNTIL THIS EDITING ENTRY HAS BEEN COMPLETED.

MPLED1 UNQUEUES THE OLDEST ENTRY FROM THE EDIT INPUT QUEUE TABLED AND FROM THIS ENTRY PROVIDES THE EDITING PARAMETERS FOR EOLED1.

IF THE INPUT DATA HAS BEEN RECEIVED FROM ANY STATION OTHER THAN THE THREE KENNEDY DOWNRANGE STATIONS, MPLED1 GIVES CONTROL DIRECTLY TO EOLED1 TO EDIT THE DATA. IF 80 MINUTES HAS NOT ELAPSED SINCE LIFTOFF AND IF THE DATA IS FROM THE BERMUDA AN/FPS-16 RADAR, MCCAP1 IS SET TO NONZERO TO INDICATE INITIAL, FOR FIRST PASS, BERMUDA DATA. FOR THE DOWNRANGE STATIONS THE DATA IS EDITED IF 80 MINUTES HAVE PASSED SINCE LIFTOFF OR REENTRY PHASE HAS BEEN ENTERED. OTHERWISE, ZERO IS STORED IN THE SECOND LOCATION OF THE TMRMXX TABLE TO INDICATE THAT NO INPUT DATA IS AVAILABLE FOR EDITING, AND MPLED1 GIVES CONTROL TO MFLED1, BYPASSING EDITING FOR THIS ENTRY.

USAGE

MPLED1 IS ENTERED FROM MOPRIO AND EXITS TO EOLED1 TO MFLED1.

A) STORAGE REQUIRED -

41 LOCATIONS

B) MACROS -

QENBA
QENBZ
QUEUE
TRNOF
TRNON

C) PARAMETERS -

A
D
MNLED1
MNREST

D) COMMUNICATION CELLS -

MCCAP1
MCGTLO
MCHFSC
MCLED1
MCLEDD
MCMINS

E) TABLE -

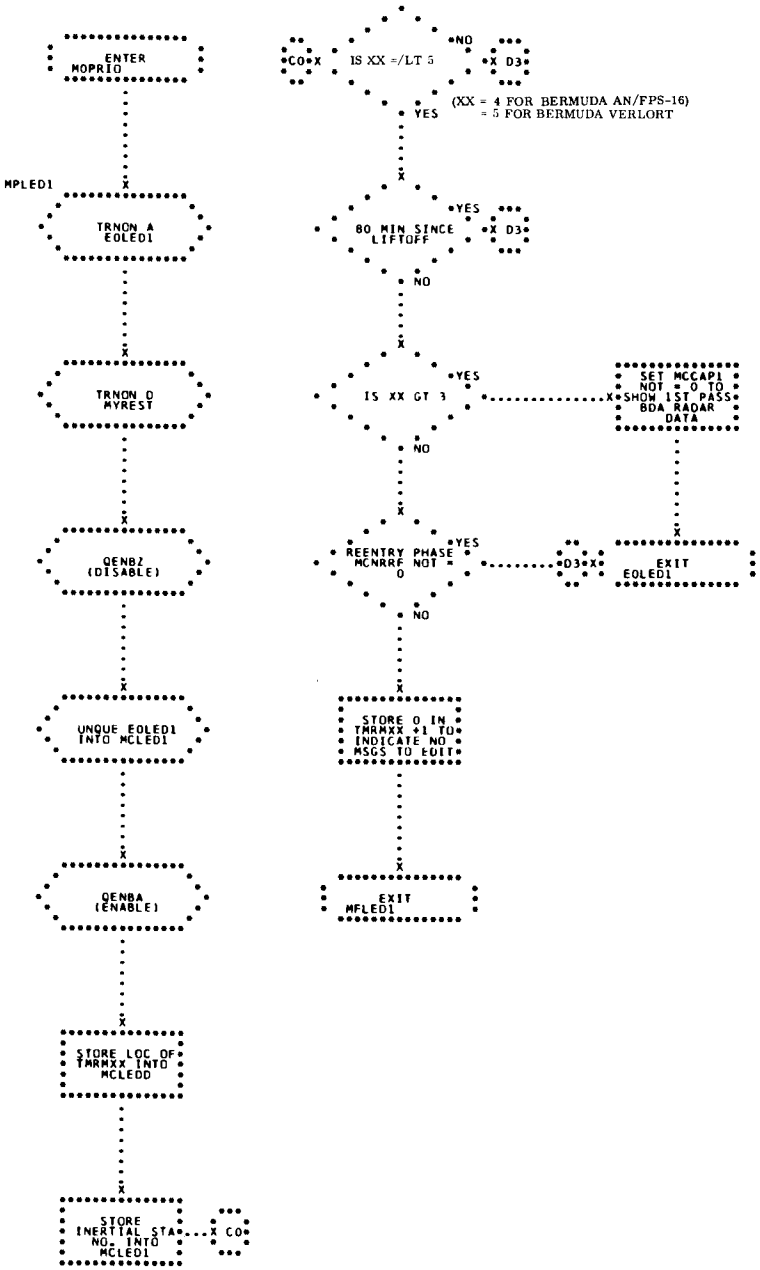
TMRMXX

F) CONSTANTS -

K00072
K01800

G) TIME REQUIRED -

MINIMUM - 0.191 MILLISECOND
MAXIMUM - 0.248 MILLISECOND


ENTER MOPRIO
MPLED1
TRNON A EOLED1
TRNON O MYREST
QENBZ (DISABLE)
UNQUE EOLED1 INTO MCLED1
QENBA (ENABLE)
STORE LOC OF TMRMXX INTO MCLEDD
STORE INERTIAL STA NO. INTO MCLED1
CO
X
IS XX =/LT 5
NO
X D3
YES
(XX = 4 FOR BERMUDA AN/FPS-16)
= 5 FOR BERMUDA VERLORT
80 MIN SINCE LIFTOFF
IS XX GT 3
SET MCCAPI NOT = 0 TO SHOW 1ST PASS BDA RADAR DATA
REENTRY PHASE MCNRRF NOT = 0
EXIT EOLED1
STORE 0 IN TMRMXX +1 TO INDICATE NO MSGS TO EDIT
EXIT MFLED1

23. MPLED2 - MONITOR PREFIX TO EOLED1

MPLED2 IS USED WHEN THE PHASE CHANGES FROM ORBIT TO REENTRY. IT SERVES AS A QUEUE FOR CERTAIN RADAR MESSAGE BLOCKS WHICH MAY BE ACCEPTABLE REENTRY INPUT, BUT WHICH HAVE BEEN QUEUED AS ORBIT INPUT FOR DIFFERENTIAL CORRECTION BUT NOT PROCESSED BY DIFFERENTIAL CORRECTION AS ORBITAL INPUT. MPLED2 ALLOWS MFLED1 TO REQUEUE DIFFERENTIAL CORRECTION IN THE PROPER ORDER WITH THE MESSAGE BLOCKS, AFTER MAKING THE BLOCKS VALID FOR REENTRY INPUT. (SEE FLOWCHART.)

INPUT

INPUT TO MPLED2 INCLUDES - TMRMXX - THE BASE ADDRESS OF THE RADAR MESSAGE BLOCK TMRMXX (WHERE THE XX RANGES FROM 01 TO 12) IS ENTERED IN THE QUEUE FOR MPLED2.

OUTPUT

A) MCCHEK IF THE QUEUE TABLE FOR MPLED2 HAS BEEN EMPTIED, MCCHEK IS SET TO NONZERO.

B) MCLEDD CONTAINS IN ITS ADDRESS THE BASE LOCATION OF THE RADAR MESSAGE BLOCK, TMRMXX, TO BE PROCESSED BY MFLED1.

METHOD

MPLED2 SUPPRESSES MYREST, WHICH IS UNSUPPRESSED WHEN MFLED1 RECEIVES THE INDICATION THAT THE QUEUE FOR MPLED2 HAS BEEN EMPTIED. THE SUPPRESSION IS NECESSARY TO ENSURE THAT RADAR MESSAGE BLOCKS ARE QUEUED FOR DIFFERENTIAL CORRECTION IN THE PROPER ORDER.

MPLED2 THEN UNQUEUES AN ENTRY FROM ITS QUEUE AND

STORES THE ENTRY IN MCLEDD. IF THE QUEUE IS EMPTY, AN INDICATION IS SET IN MCCHEK, AND CONTROL IS TRANSFERRED TO MFLED1.

USAGE

MPLED2 IS ENTERED FROM MOPRIO AND EXITS TO MFLED1.

A) STORAGE REQUIRED -
15 LOCATIONS

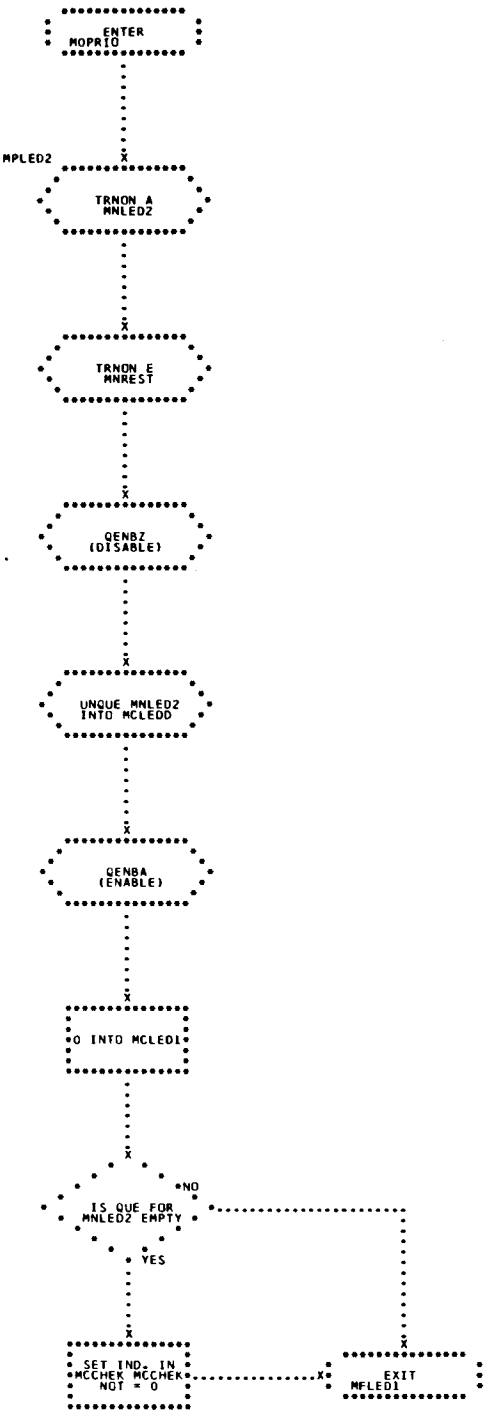
B) MACROS -
QENBA
QENBZ
TRNON
UNQUE

C) PARAMETERS -
MNLED2
MNREST

C) COMMUNICATION CELLS -
MCCHEK
MCLEDD

E) TABLES -
TMRMXX (WHERE THE XX RANGES FROM 01 TO 12) AND THE QUEUE TABLE FOR MPLED2.

F) TIME REQUIRED -
MINIMUM - 0.093 MILLISECOND
MAXIMUM - 0.392 MILLISECOND


ENTER
MOPRIO
MPLED2
TRNON A
MNLED2
TRNON E
MNREST
QENBZ
(DISABLE)
UNQUE MNLED2
INTO MCLEDD
QENBA
(ENABLE)
0 INTO MCLED1
NO
IS QUE FOR
MNLED2 EMPTY
YES
SET IND. IN
MCCHEK MCCHEK
NOT = 0
EXIT
MFLED1

24. EOLED1 - EDIT

DURING ORBIT, A NETWORK OF TRACKING SITES LOCATED AROUND THE WORLD MONITORS THE SPACECRAFT'S POSITION AND THE STATUS AND PERFORMANCE OF ITS HUMAN OCCUPANT. SUMMARIES OF THIS DATA ARE SENT BY TELETYPE TO GODDARD. THE DATA IS PROCESSED BY THE INPUT PROGRAM (IOTTIN), AND ONLY THE POSITIONAL DATA IS RETAINED IN MEMORY UNTIL SUCH TIME AS MONITOR INSTRUCTS THE EOLED1 PROGRAM TO EDIT THE INFORMATION. (SEE FLOWCHART)

THE EOLED1 PROGRAM -

A) DELETES RADAR MESSAGES HAVING ERRONEOUS TIMES.

B) FINDS THE TIME OF CLOSEST APPROACH (t_{ca}) TO EACH SITE.

INPUT

EOLED1 RECEIVES ITS INPUT FROM -

A)	MCLED1	THE ADDRESS FIELD OF MCLED1 CONTAINS THE INTERNAL STATION NUMBER, j.
B)	TMSTCH	THE ADDRESS FIELD OF TMSTCH + j CONTAINS THE FIRST LOCATION OF THE STATION CHARACTERISTICS BLOCK JTH STATION MINUS 1.
C)	MCLEDD	CONTAINS THE BASE ADDRESS OF THE MESSAGE BLOCK OF STATION j, TMRMXX + 203 TO BE EDITED.

THE DATA BLOCK IS ARRANGED IN THE FOLLOWING FORMAT -

TMRMXX + 203	STATION NUMBER ,, PASS NUMBER
+1	NUMBER OF RADAR MESSAGE IN THIS BLOCK N (≥ 0)

+2	BLANK
+3	TIME t_1, FIXED POINT (MIN,SEC)
+4	RANGE R_1 FLOATING POINT, GEMINI UNIT
+5	AZIMUTH A_1 FLOATING POINT, RADIANS
+6	ELEVATION E_1 FLOATING POINT, RADIANS
+7	t_2 SECOND RADAR MESSAGE
+8	R_2 SECOND RADAR MESSAGE
+9	A_2 SECOND RADAR MESSAGE
+10	E_2 SECOND RADAR MESSAGE
....	ETC., UP TO 50 MESSAGES

OUTPUT

THE OUTPUT DATA OF EOLED1 IS THE SAME AS THAT OF THE INPUT DATA BLOCK AND CONTAINS THE EDITED MESSAGES IN THE FOLLOWING FORMAT -

TMRMXX + 203	STATION NUMBER,,PASS NUMBER
+1	NUMBER OF RADAR MESSAGE RETAINED AFTER EDITING, NL $\leq$ N
+2	LOCATION RELATIVE TO TMRMXX + 203 OF t_{CA}
+3	t_1 FIXED POINT (MIN,,SEC)
+4	R_1 FLOATING POINT, GEMINI UNIT
+5	A_1 FLOATING POINT, RADIANS
+6	E_1 FLOATING POINT, RADIANS,
....	ETC.

METHOD

THE TIME SEQUENCE CHECK DIVIDES THE TIME DATA INTO SEQUENCES OF CONSECUTIVE TIME DATA, EACH SEQUENCE

SATISFYING THE FOLLOWING CONDITIONS -

A) THE SEQUENCE IS MONOTONICALLY INCREASING

B) THE SPAN OF THE SEQUENCE (INCREMENT BETWEEN THE FIRST AND THE LAST ELEMENTS) DOES NOT EXCEED SEVEN MINUTES. THE LONGEST OF THESE SEQUENCES IS ACCEPTED AS VALID TIME DATA.

EACH OF THOSE ELEMENTS NOT IN THE ACCEPTED SEQUENCE IS THEN EXAMINED. IF THE ACCEPTED SEQUENCE IS EXTENDED TO INCLUDE THIS ELEMENT AND THE EXTENDED SEQUENCE STILL MEETS THE CONDITION (A AND B, ABOVE) THE ELEMENT IS ACCEPTED AS VALID. OTHERWISE, THE RADAR MESSAGE CONTAINING IT IS DELETED.

TIME OF CLOSEST APPROACH

THE TIME OF CLOSEST APPROACH, t_{ca} IS THE TIME ASSOCIATED WITH THE MAXIMUM ELEVATION ANGLE.

LOCAL VERTICAL CORRECTION

$$\Delta E = \Delta\lambda \cdot \sin A_A - \Delta\varphi \cdot \cos A_A \quad (|\Delta E|) < 4 \cdot \frac{\pi}{180^o}$$

$$\approx 0.00116$$

$$\Delta A = \frac{\Delta\lambda \cdot \cos A_A + \Delta\varphi \cdot \sin A_A \quad (|\Delta A|)}{\Delta E - \cot E_A} < 5.16^o \cdot \frac{\pi}{180^o}$$

THUS- $R_G = R_A$

$A_G = A_A + \Delta A \pmod{2\pi}$

$E_G = E_A + \Delta E$

SUBSCRIPTS A AND G REFER TO ASTRONOMIC (OR GEOID) AND GEODETIC (OR GEMINI ELLIPSOID) QUANTITIES RESPECTIVELY. $\Delta\lambda$ AND $\Delta\phi$ ARE THE PRIME VERTICAL AND MERIDIAN COMPONENTS OF THE DEFLECTION OF THE (LOCAL) VERTICAL (OR ASTRONOMIC

ZENITH) FROM THE TRUE VERTICAL (OR GEODETIC ZENITH), NEGATIVE VALUES INDICATE DEFLECTION TO THE EAST OR THE SOUTH.

$$|\Delta\varphi| \cdot |\Delta\varphi| < 2' \times \frac{\pi}{180^o} \approx 0.00058$$

REFRACTION CORRECTION

$$\Delta R = 4K\,(n_0 - 1)/\sin E_G$$

$$\Delta E = (n_0 - 1)\cot E$$

FOR E_G <.175 RADIANS THE FOLLOWING EQUATIONS ARE USED TO COMPUTE Δ F—

$$\tau = \left[1.03585796 - \frac{.01072014}{E_G} + \frac{.1279119 \times 10^{-7}}{E_G^{\,2}} - \frac{.1227363 \times 10^{-7}}{E_G^{\,3}} \right] (n_0 - 1)\cot E_G$$

$$r^2 = R_G^{\,2} + r_0^{\,2} + 2R_G r_0 \sin (E_G - \tau)$$

$$K = n_0 r_0 \cos E_G$$

$$\tan \theta_1 = \sqrt{\frac{r^2 - K^2}{K}}$$

$$\delta = \frac{N_0 \times 10^{-b} - \tau \tan E_G}{\tau - \tan E_G + n_0 \tan \theta_1}$$

$$\Delta E = \tau - \delta$$

WHERE-

E_G = OBSERVED ELEVATION ANGLE IN RADIANS

τ = BENDING ANGLE OF RAY

ΔE = ELEVATION ANGLE ERROR DUE TO REFRACTION

n_0 = INDEX OF REFRACTION AT RADAR

N_0 = $(n_0 - 1) \times 10^6$ = REFRACTIVITY AT RADAR

E = TRUE ELEVATION ANGLE

R_G = SLANT RANGE OBSERVATION

r_0 = RADIAL DISTANCE FROM EARTH'S CENTER TO RADAR

r = APPROXIMATE RADIAL DISTANCE FROM EARTH'S CENTER TO TARGET

θ_1 = LOCAL ELEVATION ANGLE OF RAY AT TARGET

δ = $\tau - \Delta E$ = ANGLE USED TO COMPUTE ΔE

THUS -

R = $R_G - \Delta R$

A = A_G

E = $E_G - \Delta E$

USAGE

A) STORAGE REQUIRED -

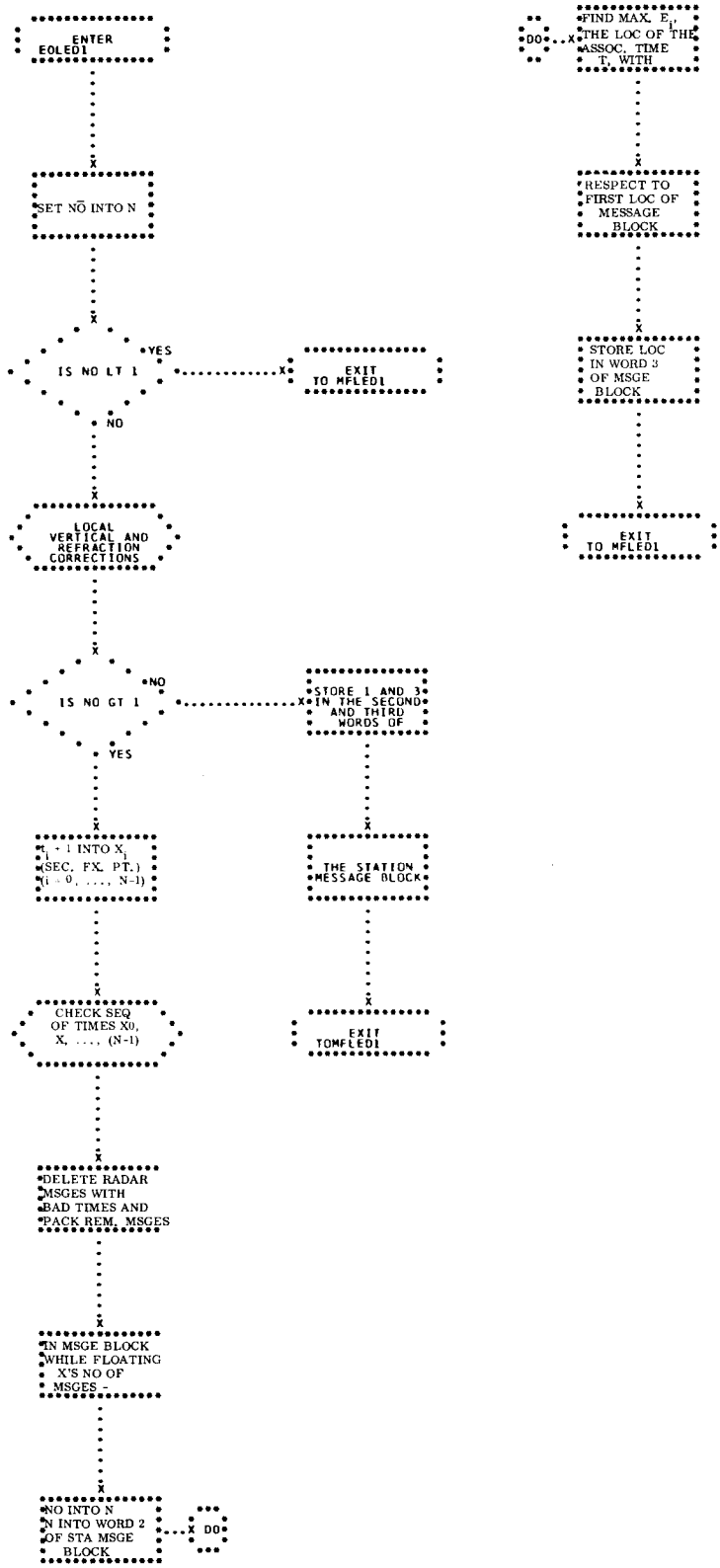
INSTRUCTIONS	- 432
TEMP	- 246
CONSTANT	- 7
TOTAL	685 LOCATIONS

B) TIME REQUIRED -

THE COMPUTER RUNNING TIME REQUIRED BY EOLED1 IS A FUNCTION OF THE QUANTITY AND QUALITY OF THE INPUT.

C) CHECKOUT -

THE REVISED EOLED1 HAS BEEN OPERATING IN THE GEMINI PROGRAM SYSTEM SINCE MARCH 1965.


ENTER EOLED1
SET NO INTO N
IS NO LT 1
YES
EXIT TO MFLED1
NO
LOCAL VERTICAL AND REFRACTION CORRECTIONS
IS NO GT 1
NO
STORE 1 AND 3 IN THE SECOND AND THIRD WORDS OF
YES
t_i - 1 INTO X_i (SEC. FX. PT.) (i = 0, ..., N-1)
THE STATION MESSAGE BLOCK
CHECK SEQ OF TIMES X0, X, ..., (N-1)
EXIT TOMFLED1
DELETE RADAR MSGES WITH BAD TIMES AND PACK REM. MSGES
IN MSGE BLOCK WHILE FLOATING X'S NO OF MSGES -
NO INTO N N INTO WORD 2 OF STA MSGE BLOCK
DO
DO
FIND MAX. E_i, THE LOC OF THE ASSOC. TIME T, WITH
RESPECT TO FIRST LOC OF MESSAGE BLOCK
STORE LOC IN WORD 3 OF MSGE BLOCK
EXIT TO MFLED1

25. MFLED1 - MONITOR SUFFIX TO EOLED1

MFLED1 DIRECTS ALL EXIT CONTROL FUNCTIONS ASSOCIATED WITH THE LOW-SPEED EDIT PROGRAM, EOLED1. MFLED1 PROCESSES THE EDITED RADAR MESSAGE BLOCK CONTAINING FIRST-PASS BERMUDA DATA TO ELIMINATE ALL POWERED-FLIGHT DATA AND, AFTER RETROFIRE HAS OCCUREED, PROCESSES THE EDITED RADAR MESSAGE BLOCK FOR REENTRY INPUT. ALL MESSAGES IN THE RADAR MESSAGE BLOCK THAT HAVE A TIME TAG EARLIER THAN THE END OF POWERED FLIGHT ARE DELETED. DURING THE REENTRY PHASE, ALL MESSAGES CONTAINING A TIME TAG EARLIER THAN BURNOUT ARE DELETED. THE NUMBER-OF-MESSAGES INDICATOR FOR THE MESSAGE BLOCK IS UPDATED, IF NECESSARY, AS IS THE TIME OF MINIMUM RANGE. DURING THE CHANGE OF PHASE FROM ORBIT TO REENTRY, MFLED1, WITH MPLED2, HAS THE ADDED FUNCTION OF RE-QUEUEING DIFFERENTIAL CORRECTION (DC) IN THE PROPER ORDER WITH THE MESSAGE BLOCKS WHICH ARE VALID REENTRY INPUT, BUT WHICH HAVE BEEN QUEUED AS ORBITAL INPUT TO DIFFERENTIAL CORRECTION BUT HAVE NOT PROCESSED BY DIFFERENTIAL CORRECTION AS ORBITAL INPUT. BEFORE REQUEUEING, THE RADAR MESSAGE BLOCK IS PROCESSED AS DESCRIBED ABOVE TO BE MADE VALID FOR REENTRY INPUT. (SEE FLWOCHART.)

INPUT

A) THE RADAR MESSAGE BLOCK TMRMXX SPECIFIED IN THE ADDRESS OF MCLEDD, WHERE XX VARIES FROM 31 TO 12 CORRESPONDING TO ONE OF 12 BLOCKS IN A BUFFER POOL. THE RADAR BLOCKS HAVE THE FOLLOWING FORMAT -

WORD	CONTENTS
1	STATION IDENTIFICATION NUMBER IN THE ADDRESS FIELD.
2	NUMBER OF RADAR MESSAGES IN THIS BLOCK.
3	ADDRESS FIELD CONTAINS LOCATION OF TIME OF MINIMUM RANGE IN THIS BLOCK RELATIVE TO THE FIRST WORD. FOR EXAMPLE - IF THE FIRST TIME (THE TIME IS WORD 4) IS THE TIME OF MINIMUM RANGE, THE ADDRESS OF WORD 3 CONATINS A 3 (WORD 1+3). IF THE SECOND TIME IS THE TIME OF MINIMUM RANGE, THE ADDRESS OF WORD 3 CONTAINS A 7.
4	TIME (T_1) IN FIXED-POINT, MINUTES IN ADDRESS FIELD, SECONDS IN DECREMENT FIELD.
5	RANGE (R_1) IN FLOATING-POINT GEMINI UNITS.
6	AZIMUTH (A_1) IN FLAOTING-POINT RADIANS.
7	ELEVATION (E_1) IN FLOATING-POINT RADIANS
8	T_2 2ND RADAR MESSAGE
9	R_2 2ND RADAR MESSAGE
10	A_2 2ND RADAR MESSAGE
11	E_2 2ND RADAR MESSAGE
•	
•	
•	
203	(UP TO 50 T, R, A, E, MESSAGES)

T_n IS ALWAYS GREATER THAN T_{n-1}.

B) MCLEDD CONTAINS IN ITS ADDRESS THE STARTING LOCATION OF THE RADAR MESSAGE BLOCK FOR EDITING, TMRMXX (WHERE THE XX VARIES FROM 01 TO 12).

C) MCNRRF CONTAINS ZERO DURING ORBIT, SET TO NONZERO DURING REENTRY.

D) MCBNOT CONTAINS IN THE REENTRY PHASE GMT OF BURNOUT (SECONDS IN ADDRESS, MINUTES IN DECREMENT), SET EITHER BY MYREST OR MYSEEK.

E) MCPFLD CONTAINS GMT OF THE TERMINATION OF POWERED FLIGHT (SECONDS IN ADDRESS, MINUTES IN DECREMENT), SET IN LAUNCH.

F) MCCAP1 LOCATION INITIALLY ZERO, BUT SET TO NONZERO BY MPLED1 TO INDICATE TO MFLED1 THAT THE BLOCK BEING EDITED CONTAINS FIRST PASS BERMUDA DATA. MCCAP1 IS RESET TO ZERO BY MFLED1.

G) MCCHEK LOCATION INITIALLY ZERO, BUT SET TO NONZERO BY MPLED2 TO INDICATE TO MFLED1 THAT THE QUEUE FOR MPLED2 HAS BEEN EMPTIED, IT IS THEN RESET TO ZERO BY MFLED1.

OUTPUT

A) TMRMXX

WORD	CONTENTS
1	STATION IDENTIFICATION IN THE ADDRESS FIELD.
2	IF THE MESSAGE BLOCK CONTAINS FIRST-PASS BERMUDA DATA OR IF RETROFIRE HAS OCCURRED, WORD 2 CONTAINS THE NUMBER OF

	MESSAGES IN THE BLOCK HAVING A TIME GREATER THAN OR EQUAL TO MCBNOT.	
3	IF THE MESSAGE BLOCK CONTAINS FIRST-PASS BERMUDA DATA OR IF RETROFIRE HAS OCCURRED, WORD 3 CONTAINS IN ITS ADDRESS FIELD THE LOCATION OF THE TIME OF MINIMUM RANGE RELATIVE TO THE FIRST WORD.	
4	TIME (T_1) IN FIXED-POINT, MINUTES IN ADDRESS FIELD, SECONDS IN THE DECREMENT FIELD.	1ST MESSAGE IN ORIGINAL BLOCK HAVING A TIME GREATER THAN MCBNOT
5	RANGE (R_1) IN FLOATING-POINT GEMINI UNITS.	
6	AZIMUTH (A_1) IN FLOATING-POINT RADIANS.	
7	ELEVATION (E_1) IN FLOATING-POINT RADIANS.	
8	T_2	2ND MESSAGE HAVING A TIME GREATER THAN MCBNOT
9	R_2	
10	A_2	
11	E_2	
•		
•		
•		
203	(UP TO 50 MESSAGES)	

B) MCCHEK SET TO NONZERO BY MPLED2 (TO INDICATE TO MFLED1 THAT THE QUEUE FOR MPLED2 HAS BEEN EMPTIED), RESET TO ZERO BY MFLED1.

C) MCCAP1 SET TO ZERO BY MFLED1.

D) MCBNOT IF THE RADAR BLOCK BEING EDITED CONTAINS FRIST-PASS BERMUDA DATA, MFLED1 STORES IN MCBNOT THE GMT OF THE END-OF-POWERED-FLIGHT (MINUTES IN DECREMENT, SECONDS IN ADDRESS).

E) ONE OF TWO MESSAGES IS QUEUED - EITHER ''0 MESSAGES IN RADAR MESSAGE BLOCK-DC BY-PASSED'' OR ''THE NUMBER OF OBSERVATIONS PRESENTED TO DC IS _____.'' WITH THE SECOND MESSAGE, DIFFERENTIAL CORRECTION IS QUEUED WITH THE EDITED RADAR MESSAGE BLOCK.

METHOD

WHEN MFLED1 IS ENTERED, THE SECOND WORD OF THE RADAR MESSAGE BLOCK, TMRMXX, WHICH CONTAINS THE NUMBER OF RADAR MESSAGES ACCEPTED BY THE EDIT PROGRAM IS TESTED. IF THIS WORD IS ZERO, DIFFERENTIAL CORRECTION IS BYPASSED, AND A MESSAGE TO THIS EFFECT IS QUEUED. OTHERWISE, A TEST IS MADE TO DETERMINE WHETHER THE INPUT IS FIRST-PASS BERMUDA DATA, FOR WHICH THE TIME FOR SCREENING THE RADAR MESSAGES IS SET TO EQUAL TO THE TIME OF THE END-OF-POWERED-FLIGHT. THE SCREENING PROCESS THEN PROCEEDS AS DESCRIBED BELOW.

IF THE INPUT IS NOT FIRST-PASS BERMUDA DATA, A TEST IS MADE TO DETERMINE WHETHER REENTRY IS IN PROCESS. FOR THE ORBIT PHASE, ALL MESSAGES IN THE BLOCK ARE VALID, AND THUS DC IS QUEUED, AS IS AN ON-LINE MESSAGE WHICH STATES THE NUMBER OF OBSERVATIONS PRESENTED TO DC. FOR THE REENTRY PHASE, THE TIME FOR SCREENING THE RADAR MESSAGES

HAS BEEN PRESET EQUAL TO THE TIME OF BURNOUT BY MYREST OR MYSEEK.

THE TIME OF THE FIRST MESSAGE IN THE RADAR BLOCK IS COMPARED TO THE SCREENING TIME. IF IT IS GREATER THAN OR EQUAL TO THE SCREENING TIME, ALL THE MESSAGES IN THE BLOCK ARE VALID, SINCE T_n IS ALWAYS GREATER THAN T_{n-1} IN THE EDITED RADAR MESSAGE BLOCK. IF THE TIME OF THE FIRST MESSAGE IN THE RADAR BLOCK IS LESS THAN THE SCREENING TIME, THE BLOCK IS SEARCHED FROM THE LAST MESSAGE RECEIVED FOR THE LATEST MESSAGE HAVING A TIME LESS THAN THE SCREENING TIME. WHEN THIS TIME IS FOUND, THE CORRESPONDING MESSAGE, AS WELL AS THOSE PRECEDING, IS REJECTED. THE NUMBER OF MESSAGES IN THE RADAR BLOCK IS UPDATED TO INDICATE ONLY THE NUMBER OF MESSAGES HAVING AN ASSOCIATED TIME GREATER THAN OR EQUAL TO THE TIME USED FOR SCREENING. IF ALL MESSAGES ARE REJECTED, THE SAME PROCEDURE IS FOLLOWED AS WHEN THERE ARE NO INPUT MESAGES. HOWEVER, IF THERE ARE SOME VALID MESSAGES IN THE BLOCK, THE VALID MESSAGES ARE MOVED FORWARD SO THAT THE FIRST MESSAGE IN THE BLOCK IS THE FIRST VALID MESSAGE.

A TEST IS MADE TO DETERMINE WHETHER THE MESSAGE WITH THE TIME OF MINIMUM RANGE WAS ACCEPTED. IF THIS MESSAGE WAS ACCEPTED, WORD 3 IN THE RADAR MESSAGE BLOCK IS UPDATED WITH THE NEW RELATIVE LOCATION OF THE MINIMUM RANGE TIME. IF, HOWEVER, THE MESSAGE WAS REJECTED, WORD 3 IN THE RADAR BLOCK IS SET TO REFERENCE THE LOCATION OF THE TIME OF THE FIRST MESSAGE IN THE BLOCK.

IF MCNRRF IS ZERO, INDICATING ORBIT PHASE, THE ORBIT NUMBER OF THE DATA IS COMPUTED BASED UPON THE TIME OF MINIMUM RANGE (t_{ca}) OF THE DATA COMPARED TO THE TIME OF ENTRY INTO THE CURRENT ORBIT (MCTPSN). ANY DATA WHOSE t_{ca} IS WITHIN 15 MINUTES OF MCTPSN AND IS BDA OR THE CAPE

COMPLEX, OR EXCEEDS THE 15 MINUTE INTERVAL AND IS EARLIER THAN THE TIME IN MCTPSN IS ASSIGNED THE CURRENT ORBIT NUMBER IN MCPASN. IF THESE CONDITIONS ARE NOT MET THE DATA IS ASSIGNED THE CURRENT ORBIT NUMBER (MCPASN) PLUS ONE. THE ORBIT NUMBER OF BDA OR THE CAPE COMPLEX, BEING EAST OF THE DIVIDING LINE, IS QUESTIONABLE AND A FLAG IS SET IN THE FIRST WORD OF THE RADAR MESSAGE BLOCK. THE ORBIT NUMBER OF EGLIN OR TEXAS, BEING WEST OF THE DIVIDING LINE IS QUESTIONABLE AND A FLAG IS SET IN THE FIRST WORD OF THE RADAR MESSAGE BLOCK.

DIFFERENTIAL CORRECTION IS QUEUED, AS IN A MESSAGE STATING ''THE NUMBER OF OBSERVATIONS PRESENTED TO DC IS _____.'' BEFORE EXIT IS MADE TO MOPRIO (BY MPLED1 WHEN ITS QUEUE HAS BEEN EMPTIED), IF MCCHEK IS NONZERO, MCCHEK IS RESET TO ZERO. MYREST IS UNSUPPRESSED AFTER HAVING BEEN SUPPRESSED IN MPLED2, AND MPLED1 IS UNSUPPRESSED AFTER HAVING BEEN SUPPRESSED IN MYREST. THESE SUPPRESSIONS ENSURE THAT THE QUEUES FOR DIFFERENTIAL CORRECTION ARE ACCEPTED IN THE PROPER ORDER. WHETHER MCCHEK IS OR IS NOT EQUAL TO ZERO, MCCAP1 IS RE-INITIALIZED TO ZERO, THE SUPPRESSION OF MYREST SET IN MPLED1 IS UNSUPPRESSED, AND THE A INIDICATOR TURNED OFF FOR BOTH MPLED1 AND MPLED2. CONTROL IS THEN TRANSFERRED TO MOPRIO. (NOTE - MPLED1 AND MPLED2 ARE NEVER IN PROCESS AT THE SAME TIME, THE B INIDICATOR FOR MPLED2 IS TURNED ON ONLY IN MYREST, WHICH CANNOT BE ENTERED WHEN MPLED1 IS IN PROCESS, AND WHICH SUPPRESSES MPLED1. THIS SUPPRESSION REMAINS UNTIL THE QUEUE FOR MPLED2 IS EMPTIED).

USAGE

MFLED1 IS ENTERED FROM EITHER MPLED1, MPLED2, OR EOLED1 AND EXITS TO MOPRIO.

A) STORAGE REQUIRED -

183 LOCATIONS

B) MACROS -

QENBA
QENBZ
QUEUE
TRNOF

C) PARAMETERS -

A	MNREST
MNLED1	MNDIFC
MNLED2	
MNMESS	

D) COMMUNICATION CELLS -

MCBNOT
MCCAP1
MCCHEK
MCLEDD
MCNRRF
MCPFLD
MCPASN
MCDCIP
MCTPSN

E) TABLES -

TMRM01	TMRM07
TMRM02	TMRM08
TMRM03	TMRM09
TMRM04	TMRM10
TMRM05	TMRM11
TMRM06	TMRM12

F) CONSTANTS -

K00000

K00001

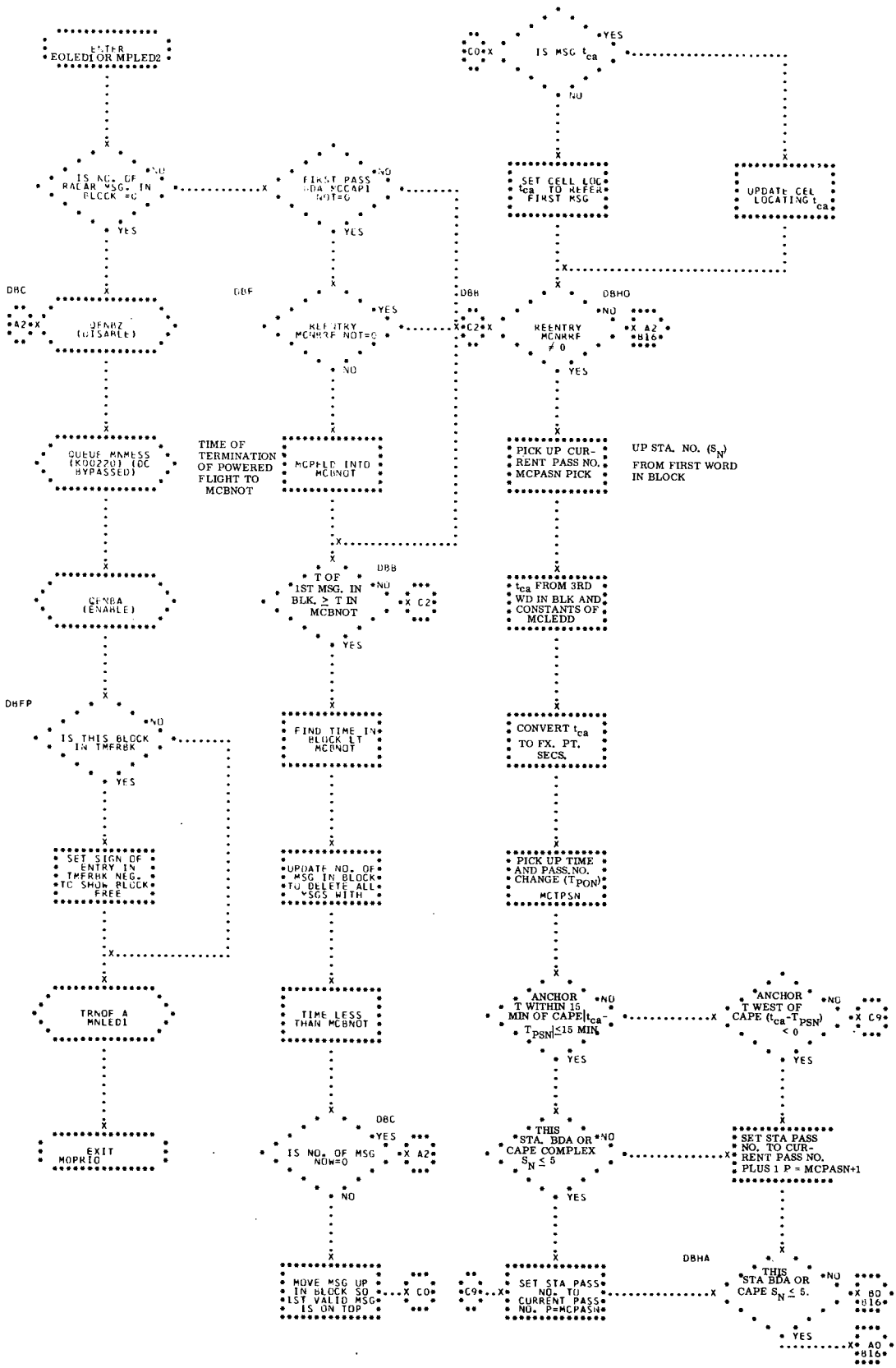
K00003

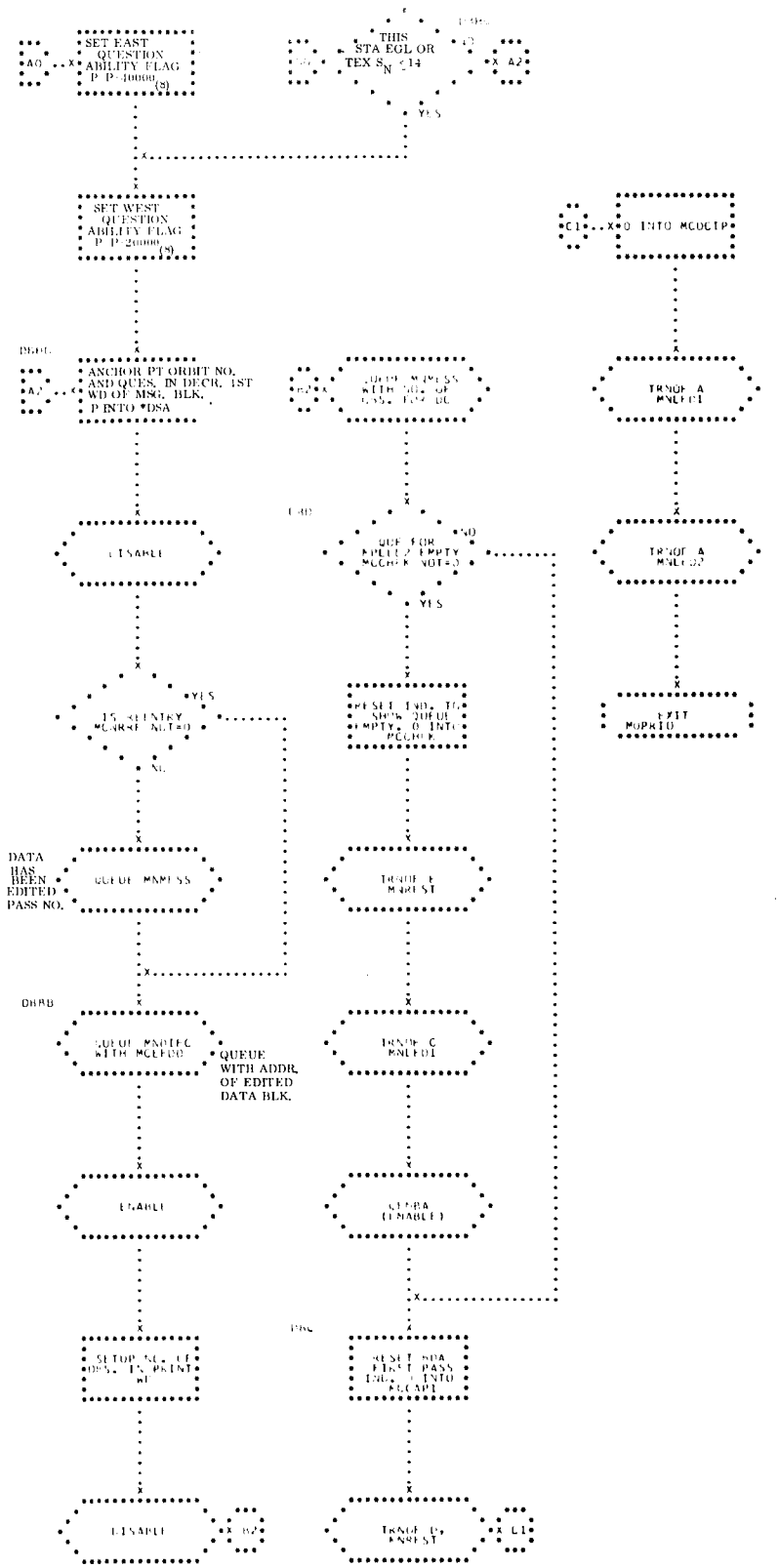
K00060

K0090

K00220

KNSTKA


ENTER EOLED1 OR MPLED2
IS NO. OF RADAR MSG. IN BLOCK =0
NO
YES
DBC
A2
QENBZ (DISABLE)
QUEUE MNMESS (K00220) (DC BYPASSED)
QENBA (ENABLE)
DBFP
IS THIS BLOCK IN TMFRBK
SET SIGN OF ENTRY IN TMFRBK NEG. TO SHOW BLOCK FREE
TRNOF A MNLED1
EXIT MOPRIO
FIRST PASS BDA MCCAP1 NOT=0
DBF
REENTRY MCNRRF NOT=0
TIME OF TERMINATION OF POWERED FLIGHT TO MCBNOT
MCPELD INTO MCBNOT
T OF 1ST MSG. IN BLK. ≥ T IN MCBNOT
DBB
C2
FIND TIME IN BLOCK LT MCBNOT
UPDATE NO. OF MSG IN BLOCK TO DELETE ALL MSGS WITH
TIME LESS THAN MCBNOT
DBC
IS NO. OF MSG NOW=0
A2
MOVE MSG UP IN BLOCK SO 1ST VALID MSG IS ON TOP
C0
C9
IS MSG t_{ca}
C0
SET CELL LOC t_{ca} TO REFER FIRST MSG
UPDATE CEL LOCATING t_{ca}
DBB
C2
REENTRY MCNRRF ≠ 0
DBHO
A2 B16
PICK UP CURRENT PASS NO. MCPASN PICK
UP STA. NO. (S_N) FROM FIRST WORD IN BLOCK
t_{ca} FROM 3RD WD IN BLK AND CONSTANTS OF MCLEDD
CONVERT t_{ca} TO FX. PT. SECS.
PICK UP TIME AND PASS.NO. CHANGE (T_{PON}) MCTPSN
ANCHOR T WITHIN 15 MIN OF CAPE $|t_{ca}-T_{PSN}| \leq 15$ MIN
ANCHOR T WEST OF CAPE ($t_{ca}-T_{PSN}$) < 0
C9
THIS STA. BDA OR CAPE COMPLEX $S_N \leq 5$
SET STA PASS NO. TO CURRENT PASS NO. PLUS 1 P = MCPASN+1
SET STA PASS NO. TO CURRENT PASS NO. P=MCPASN
DBHA
THIS STA BDA OR CAPE $S_N \leq 5$.
B0 B16
A0 B16


SET EAST QUESTION ABILITY FLAG P P-40000 (8)
THIS STA EGL OR TEX S_N ≤14
YES
SET WEST QUESTION ABILITY FLAG P P-20000 (8)
0 INTO MCDCTP
ANCHOR PT ORBIT NO. AND QUES. IN DECR. 1ST WD OF MSG. BLK. P INTO *DSA
TRNOF A MNLEDI
DISABLE
QUE FOR MPELL2 EMPTY MCCHCK NOT=0
NO
YES
TRNOF A MNLED2
IS REENTRY MCNRRF NOT=0
YES
NO
RESET IND. TO SHOW QUEUE EMPTY, 0 INTO MCCHCK
EXIT MOPRIO
DATA HAS BEEN EDITED PASS NO.
QUEUE MNMESS
TRNOF E MNREST
QUEUE MNDIEC WITH MCEFDD
QUEUE WITH ADDR. OF EDITED DATA BLK.
TRNOF C MNLEDI
ENABLE
QENBA (ENABLE)
SETUP NO. OF DRS. IN PRINT WD
RESET BDA FIRST PASS IND. 0 INTO MCCAPI
DISABLE
TRNOF D, MNREST

26. MPCPNI - MONITOR PREFIX TO CONTROL PROGRAM NUMERICAL INTEGRATION

MPCPNI SETS CELLS FOR THE NUMERICAL INTEGRATION PROCESSOR AND SETS UP TMBBNI, THE INTEGRATORS INPUT BLOCK. (SEE FLOWCHART.)

INPUT

A) QUEUE -

THE DECREMENT CONTAINS AN INTEGER THAT DETERMINES THE USE OF INTEGRATION. (SEE TABLE 1.) NUMBERS BETWEEN 0 AND 49, INCLUSIVE, INDICATE ORBIT TYPE INTEGRATION, 50 OR GREATER MEANS REENTRY TYPE INTEGRATIONS. THE ADDRESS CONTAINS THE LAST LOCATION PLUS ONE OF THE INPUT BLOCK TO THE INTEGRATOR. (THE FORMAT IS GIVEN IN TABLE 2.)

B) TABLES -

TIDRAG	THE FIRST WORD CONTAINS THE DENSITY MULTIPLYING FACTOR TO USE IN GENERAL ORBIT AND REENTRY INTEGRATIONS, THE SECOND WORD CONTAINS THE SAME FACTOR FOR ORBIT CAPABILITY.

C) PARAMETER -

MNBBNI	PARAMETER EQUAL TO LENGTH OF TMBBNI, CURRENTLY SET TO 17.

D) CONSTANTS -

K00054
K00050
K00010
K00008

K00005
K00001

OUTPUT

A) COMMUNICATION CELLS -

MCZRWX	CONTAINS THE QUEUE WORD TO MPCPNI
MCIVSS	THIS CELL IS SET IF THE SIGN OF TMBBNI+10 IS NEGATIVE (SET BY MSVTCA,) INDICATING UNEQUALLY SPACED INTERVALS.
MCIVOS	SAME USE AS MCIVOS, ADDRESS SET WITH DECREMENT OF TMBBNI+10.
KRORRV	ZERO INDICATES TO NOCPNI TO STORE POSITION VECTORS (R) ONLY. NON-ZERO INDICATES BOTH R AND V.
KCAVEC	NON-ZERO INDICATES TO NOCPNI TO STORE CAPE VECTORS ONLY.
K.MUTE	CONTAINS THE DENSITY MUTILATION CO-EFFICIENT FOR THE INTEGRATOR.

B) TABLES -

TMBBNI	NOCPNI INPUT TABLE IS BUFFERED INTO HERE FROM THE ADDRESS OF THE QUEUE WORD.

USAGE

A) STORAGE REQUIRED - 50 LOCATIONS
B) TIMING - .208 MS, MAXIMUM.

METHOD

THE METHOD IS SHOWN IN THE FLOWCHART.

TABLE 1.

D1MCZRWX)	USE OF INTEGRATION
0	ORBIT DISPLAY AND ACQUISITION TABLES
1	CAPE VECTORS
2	TTF ORBIT TABLE, ORBIT NAVIGATION VECTORS
3	DC VECTORS
4	DC INTEGRATION TABLES
5	ORBITAL CAPABILITY
6	NOMINAL VECTORS
7	INTEGRATION THRU RETROFIRE
8	MANEUVER REFLECTING VECTORS
9	APOGEE, PERIGEE COMPUTATIONS
10	INTEGRATION ON DESIRED LONGITUDE
11	MANEUVER COMPUTATIONS (PHASE ANGLE)
12	INTEGRATION THRU OAMS BURN
50	NDC TABLES BASED ON GUIDANCE PARAMETERS
51	REENTRY TABLE BASED ON GUIDANCE PARAMETERS
52	IMPACT POINT
53	TERMINATE INTEGRATION IF LOG $(V^2/A) \leq D$
54	NDC TABLES BASED ON ON-BOARD GUIDANCE EQUATIONS
55	REENTRY TABLE BASED ON ON-BOARD GUIDANCE EQUATIONS

TABLE 2.

INTEGRATOR INPUT BLOCK	QUANTITY	NAME AND UNITS
T-----	X	X-COMPONENT OF POSITION, FLTG. PT. GLU
+1	Y	Y-COMPONENT OF POSITION, FLTG. PT. GLU
+2	Z	Z-COMPONENT OF POSITION, FLG. PT. GLU
+3	X	X-COMPONENT OF VELOCITY, FLTG. PT. GVU
+4	Y	Y-COMPONENT OF VELOCITY, FLTG. PT. GVU
+5	Z	Z-COMPONENT OF VELOCITY, FLTG. PT. GVU
+6	t	TIME OF ABOVE VECTOR
+7	h	INTEGRATION STEP SIZE
+8	Δt_b	LENGTH OF TIME TO INTEGRATE BACKWARDS
+9	Δt_f	LENGTH OF TIME TO INTEGRATE FORWARDS
+10	PZE TL,,N	N-NUMBER OF INTERVALS PER OUTPUT TL-OUTPUT TABLE LOCATION
+11	P,A,T,D	P,+0=MIN. A,0=NO THRUST +1=SEC 1=THRUST T, 0=STORE D,0=COWELL 1=DON'T STORE 1=R.K.
+12	W	SPACECRAFT WEIGHT, FLTG. PT. LBS
	D(MCZRWX)=1,5,8 =10 =11 =7,12	
T----+13	APRN	λ_D Θ_D TXXXXX

+14

+15

+16

D(MCZRWX)=50,51,52 =53 =54,55

TABLE 2. (CONTINUED)

T-----+13	t_{RB}	$\dot{t}_{RB}$	λ_t
+14	t_{IR}	t_{IR}	ϕ_t
+15	B	B	
+16		D	

EXPLANATION AND FORMAT OF THESE QUANTITIES FOLLOWS -

APRN - ANCHOR POINT REVOLUTION NUMBER FOR USE WITH CAPE VECTOR INTEGRATION. FIXED POINT B35.

λ_0 DESIRED LONGITUDE TO STOP INTEGRATING ON. FLOATING PT RADIANS.

Θ_0 DESIRED PHASE ANGLE TO STOP INTEGRATING ON - FLOATING PT RADIANS.

TXXXXX - FIRST LOCATION OF A TABLE OF INPUT QUANTITIES TO THE THRUST SUBROUTINE. FORMAT BELOW.

$*t_{RB}$ GMT TO REVERSE BANK ANGLE. FLOATING PT SECS.

$*t_{IR}$ GMT TO INITIATE ROLL. FLOATING PT SECS.

B - BANK ANGLE TO BE USED IN LIFT COMPUTATION. FLOATING PT. DEGREES.

D - DENSITY FUNCTION. FLOATING PT.

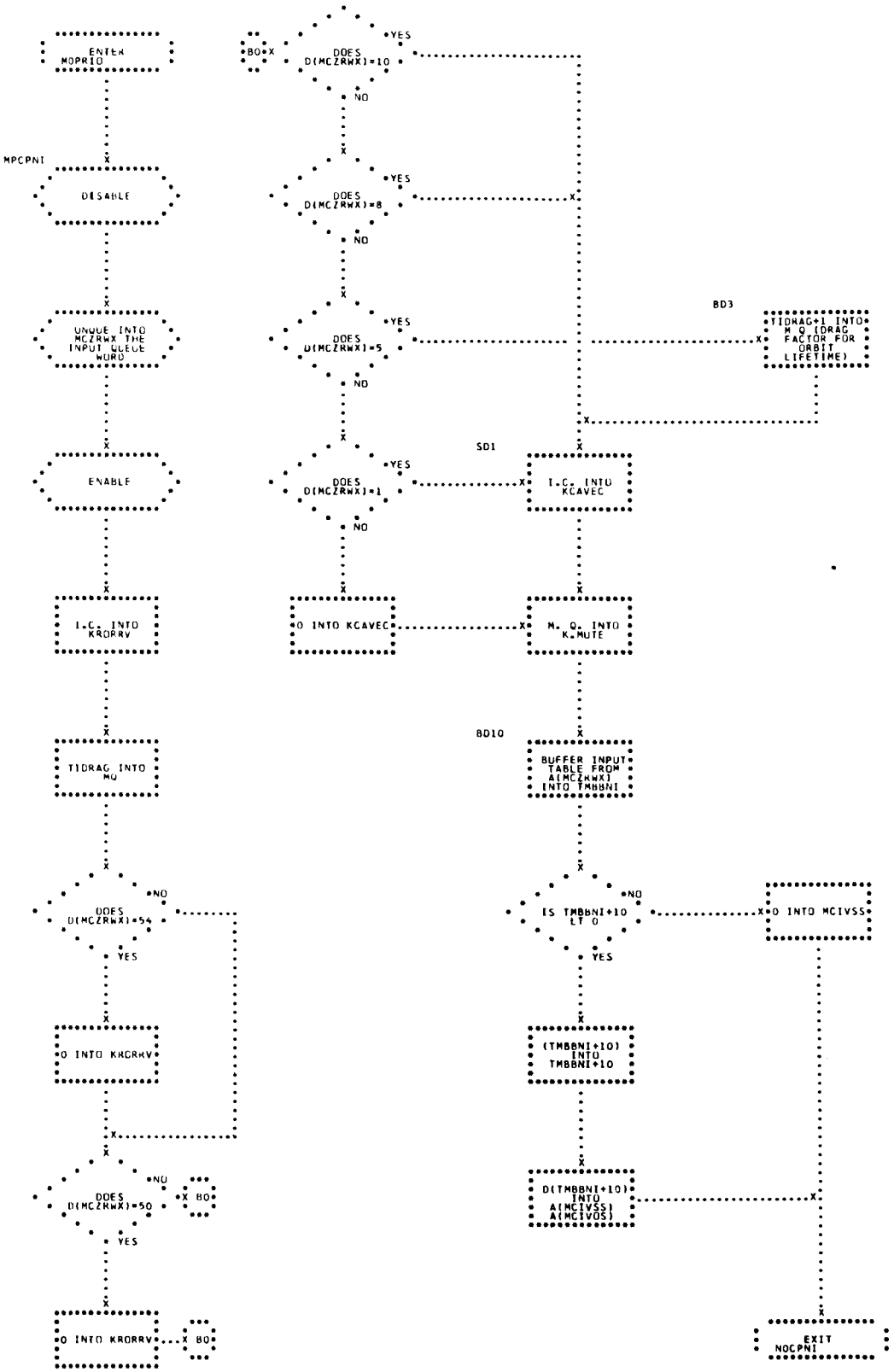
λ_t LONGITUDE OF TARGET. FLOATING PT. RADIANS.

ϕ_t LATITUDE (GEODETIC) OF TARGET. FLOATING PT. RADIANS.

*IF, WHILE INTEGRATING A REENTRY TRAJECTORY USING THE

GUIDANCE PARAMETERS, IT IS DESIRED NOT TO REVERSE BANK ANGLE (OR INITIATE ROLL), THEN t_{RB} (OR t_{IR}) SHOULD BE SET NEGATIVE.

THRUST INPUT	BLOCK	QUANTITY	UNITS
TXXXXX		TIME	FL PT. SEC
+1			FL PT. SEC
+2			FL PT. SEC
+3			FL PT. SEC
+4	GMTF		FL PT. SEC
+5	GMTF		FL PT. SEC
+6	GMTF		FL PT. SEC
+7	GMTF		FL PT. SEC
+8		THRUST LEVEL	FL PT. LBS
+9		TL	FL PT. LBS
+10		TL	FL PT. LBS
+11		TL	FL PT. LBS
+12		ORIENTATION ANGLE	FL PT. RAD
+13		OA	FL PT. RAD
+14		OA	FL PT. RAD
+15		OA	FL PT. RAD
+16		CANT ANGLE	FL PT. RAD
+17		CA	FL PT. RAD
+18		CA	FL PT. RAD
+19		CA	FL PT. RAD
+20		PITCH	FL PT. RAD
+21		ROLL	FL PT. RAD
+22		YAW	FL PT. RAD
+23		THRUST IND.	FX PT
+24		MASS EXPULSION RATE	FL LBS/SEC


ENTER MOPRIO
MPCPNI
DISABLE
UNQUE INTO MCZRWX THE INPUT QUEUE WORD
ENABLE
I.C. INTO KRORRV
TIDRAG INTO MQ
DOES D(MCZRWX)=54
YES
NO
0 INTO KRORRV
DOES D(MCZRWX)=50
YES
NO
BO
0 INTO KRORRV
BO
DOES D(MCZRWX)=10
YES
NO
DOES D(MCZRWX)=8
YES
NO
DOES D(MCZRWX)=5
YES
NO
DOES D(MCZRWX)=1
YES
NO
0 INTO KCAVEC
SD1
I.C. INTO KCAVEC
M. Q. INTO K.MUTE
BD10
BUFFER INPUT TABLE FROM A(MCZRWX) INTO TMBBNI
IS TMBBNI+10 LT 0
NO
YES
(TMBBNI+10) INTO TMBBNI+10
D(TMBBNI+10) INTO A(MCIVSS) A(MCIVOS)
0 INTO MCIVSS
BD3
TIDRAG+1 INTO M.Q. (DRAG FACTOR FOR ORBIT LIFETIME)
EXIT NOCPNI

27. NOCPNI - NUMERICAL INTEGRATION CONTROL PROGRAM

NOCPNI INTEGRATES THE THREE NEWTONIAN EQUATIONS OF MOTION TO OBTAIN AN ORBIT PREDICTION TABLE CONTAINING $\vec{r}$ AND $\vec{v}$ VECTORS FOR SPECIFIED TIMES. NOCPNI SERVES AS THE BASIC CONTROL PROGRAM FOR THE PROGRAMMED NUMERICAL INTEGRATION SYSTEM. N1SUFT DEVELOPS THE INITIAL CONVERGED SECOND-DERIVATIVE FUNCTION TABLE, USING N2EXCR TO EXTRAPOLATE AND CORRECT EACH NEW FUNCTION ENTRY IN THE TABLE. THE N3SDCL, N4DRAG, N4PDEN, N4COEFF, AND N6OPEP SUBROUTINES ARE USED TO CALCULATE THE SECOND DERIVATIVE, N5OTBI AND N5OTBL DEVELOP THE RESULTING PREDICTION OUTPUT TABLE WHICH CONTAINS THE CONVERGED $\vec{r}$ AND $\vec{v}$ VECTORS. N5CAPI AND N5CAPS DEVELOP THE OUTPUT TABLE IF CAPE VECTORS ARE USED, N7VARS DECREASES STEP SIZE AND CALCULATES F VALUES FOR HALF-STEPS. (SEE FLOWCHART.)

THE VECTOR EQUATION OF MOTION IS -

$$\vec{r} = -\frac{\partial V}{\partial \vec{r}} + \text{Drag} + \text{Lift} + \text{Thrust}$$

WHERE THE POTENTIAL (V) IS APPROXIMATED BY -

$$V = \frac{1}{r}\left\{1 + \frac{J}{3r^2}\left[1 - 3\left(\frac{Z}{r}\right)^2\right] - \frac{H}{5r^3}\left[3 - 5\left(\frac{Z}{r}\right)^2\right]\frac{Z}{r}\right.$$

$$\left. + \frac{K}{35\,r^4}\left[3 - 30\left(\frac{Z}{r}\right)^2 + 35\left(\frac{Z}{r}\right)^4\right] + \ldots\ldots\right\}$$

THUS -

$$\frac{\partial V}{\partial X}$$

$$\frac{\partial V}{\partial r} = \frac{\partial V}{\partial Y}$$

$$\frac{\partial V}{\partial Z}$$

WHERE

$$\frac{\partial V}{\partial X} = -\frac{X}{r^3}\left\{\frac{J}{r^2}\left[1-5\left(\frac{Z}{r}\right)^2\right]+\frac{HZ}{r^4}\left[3-7\left(\frac{Z}{r}\right)^2\right]+\frac{K}{7r^4}\left[3+63\left(\frac{Z}{r}\right)^4-42\left(\frac{Z}{r}\right)^2\right]+\frac{X}{r^3}\right\}$$

$$\frac{\partial V}{\partial Y} = -\frac{Y}{r^3}\left\{\frac{J}{r^2}\left[1-5\left(\frac{Z}{r}\right)\right]^2+\frac{HZ}{r^4}\left[3-7\left(\frac{Z}{r}\right)^2\right]+\frac{K}{7r^4}\left[3+63\ \frac{Z}{r}^4-42\left(\frac{Z}{4}\right)^2\right]+\frac{Y}{r^3}\right\}$$

$$\frac{\partial V}{\partial Z} = -\frac{1}{r^3}\left\{\frac{H}{r^2}\left[6\left(\frac{Z}{r}\right)^2-7\left(\frac{Z}{r}\right)^2-\frac{3}{5}\right]+\frac{JZ}{r^2}\left[3-5\left(\frac{Z}{r}\right)^2\right]+\frac{KZ}{7r^4}\left[15+63\left(\frac{Z}{r}\right)^4-70\left(\frac{Z}{r}\right)\right]^2+Z\right\}$$

INPUT

A. NOCPNI USES THE FOLLOWING PROGRAMS -

N1SUFT N2EXCR N3SOCL N3TRB N4DRAG
N4SALT N4PDEN N4COEF N5CAPI N5CAPS
N5OTBI N5OTBL N6OPEP N7VARS N7PCAL
N8OTST N9IRRB A3TSUP A3MSCP TADAPT
ROCKNR

B) LIBRARY ROUTINES USED BY NOCPNI -

U1EXPE U1SQRT U1ATAB U1LOGE

C) 1) K000.X EQU TMBBNI THE X COORDINATE IN FLOATING POINT.

2) K000.Y EQU TMBBNI+1 THE Y COORDINATE IN FLOATING POINT.

3) K000.Z EQU TMBBNI+2 THE Z COORDINATE IN FLOATING POINT.

4) K00.VX EQU TMBBNI+3 THE VELOCITY ALONG THE X AXIS, IN FLOATING POINT.

5) K00.VY EQU TMBBNI+4 THE VELOCITY ALONG THE Y AXIS, IN FLOATING POINT.

6) K00.VZ EQU TMBBNI+5 THE VELOCITY ALONG THE Z AXIS, IN FLOATING POINT.

7) K000TA EQU TMBBNI+6 THE ANCHOR TIME ASSOCIATED WITH THE ABOVE $\vec{r}$ AND $\vec{v}$ VALUES EXPRESSED IN FIXED OR FLOATING POINT MINUTES OR SECONDS, DEPENDING ON K000TU, MEASURED FROM MIDNIGHT PRECED-

ING THE LAUNCH.

8) KOOOOH EQU TMBBNI+7 THE TIME STEP, IN FIXED OR FLOATING POINT, AT WHICH ORBIT POINTS ARE TO BE CALCULATED.

9) KOODTB EQU TMBBNI+8 THE LENGTH OF TIME TO INTEGRATE BACKWARD, EXPRESSED IN FIXED OR FLOATING POINT MINUTES OR SECONDS, DEPENDING ON KOOOTU, OR THE NUMBER OF CAPE VECTORS TO GENERATE FROM THE ANCHOR POINT REVOLUTION NUMBER (APRN).

10) KOODTF EQU TMBBNI+9 THE LENGTH OF TIME TO INTEGRATE FORWARD, EXPRESSED AS KOODTB.

11) KOOOTL EQU TMBBNI+10 THE ADDRESS CONTAINS THE ADDRESS, IN FIXED POINT, OF THE OUTPUT TABLE DESIRED. THE DECREMENT CONTAINS THE NUMBER OF INTEGRATION STEPS, IN FIXED POINT, PER OUTPUT. IT MAY BE ANY POSITIVE INTEGER.

12) KOOINP EQU TMBBNI+11 CONTAINS FOUR INTEGRATION PARAMETERS, INDICATED BY A BIT IN THE

```
                                        PREFIX, ADDRESS, TAG,
                                        AND ADDRESS.
                                        PREFIX -      +0,MINUTES
                                                      INTEGRA-
                                                      TION
                                                      +1,SECONDS
                                                      INTEGRA-
                                                      TION
                                        ADDRESS -     0, NO
                                                      THRUST
                                                      1, THRUST
                                        TAG -         0,STORE
                                                      OUTPUT
                                                      TABLE
                                                      1,DO NOT
                                                      STORE OUT-
                                                      PUT TABLE
                                        DECREMENT -   0,USE CO-
                                                      WELL INTE-
                                                      GRATOR
                                                      1,USE
                                                      RUNGE KUT-
                                                      TA INTE-
                                                      GRATOR
13) KOOINW  EQU TMBBNI+12               SPACECRAFT WEIGHT IN
                                        FLOATING POINT POUNDS.
14) KTHRST  EQU TMBBNI+13               THE ADDRESS OF THE IN-
                                        PUT TABLE FOR THE
                                        THRUST PROGRAM, IN
                                        FIXED POINT.
```

15) KCAPON EQU TMBBNI+13 CONTAINS THE APRN IN FIXED POINT IF THE REVOLUTION NUMBER IS TO BE UPDATED.

16) KGMTRB EQU TMBBNI+13 CONTAINS THE TIME TO REVERSE BANK IN FLOATING POINT IF THIS REENTRY MODE IS USED.

17) KLONGD EQU TMBBNI+13 CONTAINS THE LONGITUDE TO TERMINATE INTEGRATION ON IN FLOATING POINT RADIANS.

18) KTHET1 EQU TMBBNI+13 THE ARCTANGENT (Y/X) $+180^{O}$ FOR PHASE ANGLE TESTS.

19) KGMTIR EQU TMBBNI+14 CONTAINS THE TIME TO INITIATE ROLL IN FLOATING POINT IF THIS REENTRY MODE IS USED. FOR OTHER TYPES OF REENTRY, THIS CELL IS SET NEGATIVE.

20) KLONGT EQU TMBBNI+13 THE LONGITUDE OF THE TAGET IN FLOATING POINT RADIANS WHEN THE SLOCED LOOP GUIDANCE MODE IS USED.

21) KDLATT EQU TMBBNI+14 THE GEODETIC LATITUDE OF THE TARGET IN FLOATING POINT RADIANS WHEN THE CLOSED LOOP GUIDANCE MODE IS USED.

22) KTHET3 EQU TMBBNI+14 THE ARC TAN. (Y/X) $+360^{O}$ FOR PHASE ANGLE TESTS.

23) KBANKA EQU TMBBNI+15 THE INITIAL BANK ANGLE IN FLOATING POINT DEGREES.

24) KMANCL EQU TMBBNI+16 A FLOATING POINT INDICATOR CELL FOR PHASE ANGLE TESTS.

25) KDENSF EQU TMBBNI+16 A DENSITY FUNCTION TO TERMINATE INTEGRATION ON FOR V^2/A COMPUTATIONS.

26) KOVECT EQU TMBBNI+16 INDICATOR WHICH IS SET NON-ZERO IN REENTRY WHEN INTEGRATING THE CONVERGED TRAJECTORY.

27) KSNLIT EQU TMBBNI+16 A FIXED POINT INDICATOR CELL FOR PHASE ANGLE TESTS.

28) MCZRWX MONITOR CONTROL WORD WHOSE DECREMENT INDICATES THE TYPE OF TABLE BEING GENERATED.

D(MCZRWX) = 0, ORBIT DISPLAY AND ACQUISITION TABLES.

= 1, CAPE VECTORS.

= 2, TIME TO FIRE ORBIT TABLE, ORBIT NAVIGATION VECTORS.

= 3, DC VECTORS.

= 4, DC INTEGRATION TABLES.
= 5, ORBIT CAPABILITY.
= 6, NOMINAL VECTORS.
= 7, INTEGRATION THROUGH RETROFIRE.
= 8, MANEUVER REFLECTING VECTORS.
= 9, APOGEE, PERIGEE COMPUTATIONS.
= 10, INTEGRATION TO DESIRED LONGITUDE.
= 11, MANEUVER COMPUTATIONS (PHASE ANGLE).
= 12, INTEGRATION THROUGH DAMS BURN.
= 50, NDC TABLES BASED ON GUIDANCE PARAMETERS
= 51, REENTRY TABLES BASED ON GUIDANCE PARAMETERS
= 52, IMPACT POINT
= 53, INTEGRATION TERMINATED WHEN LOG $(V^2/A) \leq D$.
= 54, NDC TABLES BASED ON ON-BOARD GUIDANCE EQUATIONS.
= 55, REENTRY TABLE BASED ON ON-BOARD GUIDANCE EQUATIONS.
0- 49- ORBIT TYPE INTEGRATION
50- 99- REENTRY TYPE INTEGRATIONS.

THE THIRTEEN CELLS THAT MUST BE SET FOR ALL USES OF THE INTEGRATOR ARE AS FOLLOWS -

LOCATION	QUANTITY	NAME AND UNITS
TMBBNI	X	KOOO.X, FLOATING POINT GLU
+1	Y	KOOO.Y, FLOATING POINT GLU
+2	Z	KOOO.Z, FLOATING POINT GLU
+3	X	KOO.VX, FLOATING POINT GVU
+4	Y	KOO.VY, FLOATING POINT GVU
+5	Z	KOO.VZ, FLOATING POINT GVU
+6	t_A	KOOOTA
+7	h	KOOOOH } ALL EITHER FIXED
+8	Δt_B	KOODTB } OR FLOATING POINT
+9	Δt_F	KOODTF } MINUTES OR SECONDS
	(A, T)	TL- OUTPUT TABLE LOCATION
+10	TL, N	N - NUMBER OF INTERVALS PER OUTPUT.
+11	P: + 0-MIN, + 1-SEC; A: 0-NO THRUST, 1-THRUST; T: 0-STORE, 1-DON'T STORE; D: 0-COWELL, 1-R-K	
+12	W	KOOINW

D) IN ADDITION TO THE GENERAL INTEGRATION INPUT, CERTAIN USES OF INTEGRATION REQUIRE ADDITIONAL INPUT. THESE ARE SET AS FOLLOWS -

D(MCZRWX)				
=1,5,8	=10	=11	=7,12	

TMBBNI+13	KCAPON	KLONGD	KTHET1	KTHRUST
+14			KTHET3	
+15			KMANCL	
+16			KSNLTT	

	D(MCZRWX) =50,51,52	=53	=54,55
TMBBNI+13	KGMTRB	KGMTRB	KLONGT
+14	KGMTIR	KGMTIR	KDLATT
+15	KBANKA	KBANKA	KBANKA
+16	KOVECT	KDENSF	

OUTPUT

THE FINAL OUTPUT IS IN TABULAR FORMAT, EXCEPT WHEN UPDATING REVOLUTION NUMBERS.

A) IF ZERO, THE OUTPUT TABLE REPRESENTS A MINUTE'S INTEGRATION, IF NON-ZERO THE OUTPUT TABLE REPRESENTS A SECOND'S INTEGRATION.

B) THE NUMBER OF MINUTES OR SECONDS INTERVAL AT WHICH ENTRIES ARE MADE IN THE TABLE. THE QUANTITY IS EQUAL TO KOOOOH TIMES KINTVN.

C) THE TIME ASSOCIATED WITH THE FIRST SET OF ENTRIES IN THE TABLE, THE STARTING TIME.

D) THE TIME ASSOCIATED WITH THE LAST SET OF ENTRIES IN THE TABLE, THE ENDING TIME.

E) VECTORS ARE STORED X, Y, Z, $\dot{X}$, $\dot{Y}$, AND $\dot{Z}$ (SIX LOCATIONS) IN SUCCESSIVE STORAGE LOCATIONS AT INTERVALS DESCRIBED BY B), FROM STARTING TIME C) THROUGH ENDING TIME C).

THE FINAL OUTPUT TABLE FOR CAPE VECTORS AND ORBIT

CAPABILITY IS IN THE FOLLOWING FORMAT -

A) ADDRESS OF TMCAPE - CRBIT NUMBER OF THE FIRST CAPE VECTOR IN THE TABLE.
DECREMENT OF TMCAPE - ORBIT NUMBER OF THE LAST CAPE VECTOR IN THE TABLE.

B) VECTORS ARE STORED X,Y,Z,$\dot{X}$, $\dot{Y}$, $\dot{Z}$, t (SEVEN LOCATIONS) IN SUCCESSIVE STORAGE LOCATIONS AT ONE-ORBIT INTERVALS FROM ORBIT NUMBER AS DESCRIBED BY ADDRESS OF A) THRCUGH ORBIT NUMBER IN DECREMENT OF A).

THE FINAL OUTPUT TABLE FCR TERMINATING INTEGRATIONS AS A FUNCTION OF LONGITUDE IS IN THE FOLLOWING FORMAT -

A) IF ZERO, TIME IS IN MINUTES, IF NCN-ZERO, TIME IS IN SECONDS.
B) INTEGRATICN OUTPUT INTERVAL, NORMALLY ZERO.
C) BLANK.
D) GMT FOR TERMINAL LONGITUDE.
E) X
F) Y
G) Z
} $\vec{r}$ IN GEMINI UNITS OF TERMINAL LONGITUDE
H) XD
I) YD
J) ZD
} $\vec{v}$ IN GEMINI UNITS OF TERMINAL LONGITUDE.

THE FINAL OUTPUT TABLE FCR MANEUVER REFLECTING VECTORS IS IN THE FOLLCWING FORMAT -

A) X
B) Y
C) Z
} $\vec{r}$ IN GEMINI UNITS
D) XD
E) YD
F) ZD
} $\vec{v}$ IN GEMINI UNITS

G) t_E ENDING TIME OF INTEGRATION ASSOCIATED WITH THE ABOVE $\vec{r}$ AND $\vec{v}$ IN FIXED POINT.

H) O_N ORBIT NUMBER OF THE ABOVE $\vec{r}$ ANC $\vec{v}$ IN FIXED POINT.

IN THE OUTPUT TABLES, ALL VECTORS ARE IN FLOATING POINT GEMINI UNITS. THE STORE INTERVAL AND TIME ARE IN FIXED OR FLOATING POINT MINUTES OR SECONDS DEPENDENT ON WHETHER THE INPUT, TMBBNI+6 THROUGH TMBBNI+9, IS IN FIXED OR FLOATING POINT MINUTES OR SECONDS, EXCEPT IN SPECIAL CASES.

NOCPNI HAS TWO INTERMEDIATE OUTPUT TABLES, TNFUNK AND TEMNO, WHICH ARE DESCRIBED UNDER ''ERASABLE LOCATIONS.'' TNFUNK AND TEMNO FORMATS ARE SHOWN IN TABLES 1 AND 2, RESPECTIVELY. THE ERROR PRINTOUT IS SHOWN IMMEDIATELY FOLLOWING THE TABLES.

THE CONTENTS OF KOODTB AND THE CONTENTS OF KOODTF MUST BE DIVISIBLE BY THE CONTENTS OF KOOOOH AND ALSO BY THE QUANTITY KOOOOH + KINTVN. OTHERWISE, THE OUTPUT TABLE HAS BEEN SET UP WRONG.

METHOD

THE NUMBERICAL INTEGRATICN TECHNIQUE EMPLOYED IN THE MERCURY PROGRAMMING SYSTEM IS THE COWELL METHOD, WHICH IS IN MIXED PRECISION. THE RUNGE KUTTA METHOD, IN SINGLE PRECISION, IS ALSO USED. THE BASIC FORMULAS USED ARE TAKEN FROM CHAPTER 1 - THE CALCULUS OF FINITE DIFFERENCES, OF ''COMPUTATICN OF ORBITS,'' BY DR. P. HERGET. THE FOLLOWING TWO ESSENTIAL EQUATICNS ARE DERIVED FROM AND APPEAR ON PAGE 8 OF DR. HERGET'S BOOK -

$$(hD)^{-2} = \Delta^{-2} + \frac{1}{12} - \frac{1}{240}\,\Delta^{2} + \frac{31}{60480}\,\Delta^{4} - \frac{299}{3628800}\,\Delta^{6} + \ldots \quad (1)$$

$$(hD)^{-1} = \Delta^{-1} - \frac{1}{12} + \frac{11}{720}\,\Delta^{3} - \frac{191}{60480}\,\Delta^{5} + \ldots \quad (2)$$

THESE TWO OPERATORS ARE APPLIED TO THE FUNCTION $F = h^2 \ddot{\vec{r}}$ (h IS THE INTEGRATION TIME INTERVAL) TO OBTAIN THE FOLLOWING EXPRESSIONS FOR THE POSITION AND VELOCITY VECTORS -

$$(hD)^{-2} F = h^{-2} D^{-2} h^2 \ddot{\vec{r}} = \vec{r}$$

$$(hD)^{-1} F = h^{-1} D^{-1} h^2 \ddot{\vec{r}} = h\vec{v}$$

THUS, $\vec{r}$ AND $h\vec{v}$ CAN BE COMPUTED BY EVALUATING THE CORRESPONDING FINITE DIFFERENCES. IN THE FORM OF THE COWELL METHOD, WHICH IS EMPLOYED IN THE MERCURY PROGRAM, IT IS MORE CONVENIENT TO EXPRESS EACH OF THE DIFFERENCES (Δ^i), APPEARING IN THE RIGHTHAND SIDES OF EQUATIONS (1) AND (2), AS LINEAR COMBINATION OF F_i (BOTH ARE SHOWN IN THE N2EXCR WRITE-UP).

THE MIXED PRECISION SIXTH-ORDER COWELL INTEGRATION IS OBTAINED BY INTERNALLY COMPUTING THE FIRST SUMS, SECOND SUMS AND SUMMATION IN DOUBLE PRECISIONS AND THE DERIVATIVE IN SINGLE PRECISION. THERE IS A MAGNITUDE CHECK ON THE FIFTH DIFFERENCE OF THE ACCELERATION FOR AN ADDITIONAL CRITERION ON REDUCING THE STEP SIZE.

THE RUNGE KUTTA INTEGRATION IS A FOURTH-ORDER PROCESS INTEGRATING DIRECTLY THE SECOND DERIVATIVE EQUATION OF MOTION. THE EQUATIONS, TAKEN FROM ''NUMERICAL ANALYSIS'' BY J. B. SCARBOROUGH, APPEAR BELOW.

$$X''_{n,i} = f(t_n, X_{n,i}, X'_{n,i})$$

$$K_{1,i} = hf\,(t_n, X_{n,i}, X'_{n,i})$$

$$K_{2,i} = hf\left(t_n + \frac{h}{2}, X_{n,i} + \frac{h}{2} X'_{n,i} + \frac{h}{8} K_{1,i}, X'_{n,i} + \frac{K_{1,i}}{2}\right)$$

$$K_{3,i} = hf\left(t_n + \frac{h}{2}, X_{n,i} + \frac{h}{2} X'_{n,i} + \frac{h}{8} K_{1,i}, X'_{n,i} + \frac{K_{1,i}}{2}\right)$$

$$K_{4,i} = hf\left(t_n + h, X_{n,i} + hX'_{n,i} + \frac{h}{2} K_{3,i}, X'_{n,i} + K_{3,i}\right)$$

$$\Delta X_i = h\left[X'_{n,i} + \frac{1}{6}\left(K_{1,i} + K_{2,i} + K_{3,i}\right)\right]$$

$$\Delta X'_i = \frac{1}{6}\left(K_{1,i} + 2K_{2,i} + 2K_{3,i} + K_{4,i}\right)$$

where $n = 0, 1, 2, \ldots$

$i = 1, 2, 3, \ldots$

USAGE

A) CALLING SEQUENCE - NOCPNI IS ENTERED WITH TRA NOCPNI, THE PROGRAM EXITS WITH THE ACCUMULATOR SET TO ZERO AND TRA MFCPNI.

B) STORAGE REQUIRED - RUNGE KUTTA HAS 235 LOCATIONS, EXCLUDING TABLES AND CONSTANTS, THE COWELL METHOD HAS 256 LOCATICNS EXCLUDING TABLES AND CONSTANTS.

C) SPECIAL USAGE - WHEN THE HEIGHT FOR WHICH THE DRAG COEFFICIENT IS BEING CALCULATED IN N4DRAG FALLS BELOW 10,000 FEET, THE AC IS SET TO MINUS ZERO AND THE PROGRAM EXISTS WITH TRA MFCPNI. THIS IS ALSO THE EXIT WHEN DOING INTEGRATED ORBIT CAPABILITY AND THE ALTITUDE FALLS BELOW 400,000 FEET, THE IMPACT ALTITUDE FOR ORBIT CABABILITY.

D) CONSTANTS -

K000.H	PZE		TIME STEP IN FLOATING POINT
K00.2H	PZE		TIME STEP SQUARED IN FLOATING POINT
K00.SH	DEC	.0012394486	CONVERT SECONDS TO GEMINI UNITS
K00.MH	DEC	.074366916	CONVERT MINUTES TO GEMINI UNITS
K000ET	PZE		ENDING TIME

```
KOOOTI  PZE              PRESENT TIME
KOOOST  PZE              STARTING TIME
KOZERO  DEC   0
KOO2.0  DEC   2.0
KOO.25  DEC    .25
KOO6.0  DEC   6.0
KCSUMM  PZE              STORAGE FOR FIFTH
                         DIFF. SUM
KO.FOT  DEC    .25E4     (D1) MAGNITUDE TEST
                         FACTOR FOR FIFTH
                         DIFF.
KO.FD1  DEC   1.         D(1) FIFTH DIFF.
                         COEFF. FOR COWELL
KO.FD2  DEC  -5.         D(2) FIFTH DIFF.
                         COEFF. FOR COWELL
KO.FD3  DEC  10.         D(3) FIFTH DIFF.
                         COEFF. FOR COWELL
KO.FD4  DEC -10.         D(4) FIFTH DIFF.
                         COEFF. FOR COWELL
KO.FD5  DEC   5.         D(5) FIFTH DIFF.
                         COEFF. FOR COWELL
KO.FD6  DEC  -1.         D(6) FIFTH DIFF.
                         COEFF. FOR COWELL
KTSH.2  PZE              h/2 STORAGE FOR
                         RUNGE KUTTA
KTSH.8  PZE              h/8 STORAGE FOR
                         RUNGE KUTTA
K2ZERO  PZE              DOUBLE PRECISION
                         WORD (MSH)
K2ZERO  PZE              DOUBLE PRECISION
                         WORD (MSH)
K2ZER1  PZE                 OF ZERO      (LSH)
```

KDP4.0	OCT	203400000000	4.0 IN DOUBLE (MSH)
KDP4.X	PZE		PRECISION (LSH)
K.MUTE	DEC	1.0	MUTILATION COEFFICIENT
KCH233	OCT	233000000000	
MCIVSS	PZE		LOCATES DEFINING VECTOR TIME-WISE IN TMSTMS
MCIVOS	PZE		POSITICN IN TMSTMS BEING INTEGRATED TO
RKNI1	PZE	TDRKK1+3	
RKNI2	PZE	TDRKK2+3	
RKNI3	PZE	TDRKK3+3	
RKNI4	PZE	TDRKK4+3	
TEMNO+162	PZE		WORK CELL
TEMNO+156	PZE		STORAGE FOR KOOO.H
TEMNO+138	PZE		STORAGE FOR KOO.2H
TEMNO+146	PZE		=0, INTEGRATION FORWARD, ≠ 0, INT. BACKWARD
TEMNO=163	PZE		=0, DCING APPROXIMATION BYPASS ALTITUDE TESTS IN N4DRAG. ≠0, VECTOR GOOD, GO THROUGH ALTITUDE TESTS

E) TABLES -

TMSTMS			DESCRIBED IN DODIFC
TNSAVR	BSS	6,F	CONVERGED $\vec{r}, \vec{v}$ FOR LAST TIME STEPS T_{i-2}
TN7N12	PZE		IF 0, INITIALIZATION BY N1SUFT FOR N7VARS

			IF 1, ACTUAL INTE- GRATION FROM N2EXCR
	TN71CT	PZE	NUMBER OF TIMES IN- TERVAL HAD TO BE RE- DUCED A ONE ENTRANCE TO N2EXCR
	TNETIM	PZE	TIME OF LAST $\vec{r}, \vec{v}$ IN OUTPUT TABLE (FIXED POINT MINUTES OR SEC- ONDS AS KOOOTU)
	TN7HSI	PZE	IF TIME STEP IS LESS THAN UNIT, TIME STEP IS HERE IN FLOATING- POINT UNITS
	TNIFPT	PZE	IF TIME STEP IS LESS THAN UNITY, PRESENT TIME FOR $\vec{r}, \vec{v}$ IS STORED HERE IN FLOAT- ING POINT UNITS

F) ERASABLE LOCATIONS -

THE TNFUNK AND TEMNO TABLE FORMATS ARE SHOWN FOLLOWING -

TRKSRV	BSS	6,F	SAVE LOCATIONS FOR $\vec{r}, \vec{v}$ IN R-K
TDRKK1	BSS	3,F	$K_{1,i}$ STORAGE FOR R-K
TDRKK2	BSS	3,F	$K_{2,i}$ STORAGE FOR R-K
TDRKK3	BSS	3,F	$K_{3,i}$ STORAGE FOR R-K
TDRKK4	BSS	3,F	$K_{4,i}$ STORAGE FOR R-K
TRKDRR	BSS	3,F	DELTA R STORAGE FOR RUNGE KUTTA
TRKDVV	BSS	3,F	DELTA V STORAGE FOR RUNGE KUTTA

```
K000TI  EQU    TEMNO+152, PRESENT TIME (MSH)
K0000H  EQU    TEMNO+154, INTEGRATION INTERVAL
                                                (MSH)
```

G) TIME REQUIRED - RUNGE KUTTA, INTEGRATING FOR ONE POINT=1750+4(t_{N3SDCL}) + t_{N50TBI} + t_{N50TBL} (AVERAGE IN MILLISECONDS).

TABLE 1. TNFUNK TABLE FORMAT

LOCATION			CONTENTS	
TNFUNK+ 0	+26	+52	F_i/h^2	MSH
+ 1	+27	+53		LSH
+ 2	+28	+54	$F_{i\pm1}$	MSH
+ 3	+29	+55		LSH
+ 4	+30	+56	$F_{i\pm2}$	MSH
+ 5	+31	+57		LSH
+ 6	+32	+58	$F_{i\pm3}$	MSH
+ 7	+33	+59		LSH
+ 8	+34	+60	$F_{i\pm4}$	MSH
+ 9	+35	+61		LSH
+10	+36	+62	$F_{i\pm5}$	MSH
+11	+37	+63		LSH
+12	+38	+64	$F_{i\pm6}$	MSH
+13	+39	+65		LSH
+14	+40	+66	$''F_{i\pm1}$	MSH
+15	+41	+67		LSH
+16	+42	+68	$'F_{i\pm3/2}$	MSH
+17	+43	+69		LSH
+18	+44	+70	$'F_{i\pm1/2}$	MSH
+19	+45	+71		LSH
+20	+46	+72	$''F_i$	MSH
+21	+47	+73		LSH
+22	+48	+74	X_i	MSH
+23	+49	+75		LSH
+24	+50	+76	$\dot{X}_i$	MSH
+25	+51	+77		LSH

TABLE 2 TEMNO TABLE FORMAT (SHEET 1 OF 6)

LOCATION	CONTENTS	
TEMNO + 0	$F_{i\pm1/2}$ (X, Y, Z)	MSH
+ 1		LSH
+ 2	"F_i (X, Y, Z)	MSH
+ 3		LSH
+ 4	$\Sigma A_R F_{i\pm R}$, OR $\Sigma B_R F_{i\pm R}$	MSH
+ 5		LSH
+ 6	$\Sigma A'_R F_{i\pm R}$, OR $\Sigma B_R F_{i\pm R}$	MSH
+ 7		LSH
+ 8	X	MSH
+ 9		LSH
+10	Y } F_i FROM N3SDCL	MSH
+11		LSH
+12	Z	MSH
+13		LSH
+14	F_a	MSH
+15		LSH
+16	F_{a+1}	MSH
+17		LSH
+18	F_{a+2}	MSH
+19		LSH
+20	F_{a+3} X	MSH
+21		LSH
+22	F_{a+4}	MSH
+23		LSH
+24	F_{a+5}	MSH
+25		LSH
+26	F_{a+6}	MSH
+27		LSH

TABLE 2 TEMNO TABLE FORMAT (SHEET 2 OF 6)

LOCATION	CONTENTS		
TEMNO +28	F_a (1)		MSH
+29			LSH
+30	F_{a+1} (1)		MSH
+31			LSH
+32	F_{a+2} (1)		MSH
+33			LSH
+34	F_{a+3} (1)	Y	MSH
+35			LSH
+36	F_{a+4} (1)		MSH
+37			LSH
+38	F_{a+5} (1)		MSH
+39			LSH
+40	F_{a+6} (1)		MSH
+41			LSH
+42	F_a (1)		MSH
+43			LSH
+44	F_{a+1} (1)		MSH
+45			LSH
+46	F_{a+2} (1)		MSH
+47			LSH
+48	F_{a+3} (1)	Z	LSH
+49			LSH
+50	F_{a+4} (1)		MSH
+51			LSH
+52	F_{a+5} (1)		MSH
+53			LSH
+54	F_{a+6} (1)		MSH
+55			LSH

TABLE 2 TEMNO TABLE FORMAT (SHEET 3 OF 6)

LOCATION	CONTENTS		
TEMNO +56	F_a (1)		MSH
+57			LSH
+58	F_{a-1} (1)		MSH
+59			LSH
+60	F_{a-2} (1)		MSH
+61			LSH
+62	F_{a-3} (1)	X	MSH
+63			LSH
+64	F_{a-4} (1)		MSH
+65			LSH
+66	F_{a-5} (1)		MSH
+67			LSH
+68	F_{a-6} (1)		MSH
+69			LSH
+70	F_a (1)		MSH
+71			LSH
+72	F_{a-1} (1)		MSH
+73			LSH
+74	F_{a-2} (1)		MSH
+75			LSH
+76	F_{a-3} (1)	Y	MSH
+77			LSH
+78	F_{a-4} (1)		MSH
+79			LSH
+80	F_{a-5} (1)		MSH
+81			LSH
+82	F_{a-6} (1)		MSH
+83			LSH

TABLE 2 TEMNO TABLE FORMAT (SHEET 4 OF 6)

LOCATION	CONTENTS		
TEMNO + 84	F_a (1)	↑	MSH
+ 85			LSH
+ 86	F_{a-1} (1)		MSH
+ 87			LSH
+ 88	F_{a-2} (1)		MSH
+ 89			LSH
+ 90	F_{a-3} (1)	Z	MSH
+ 91			LSH
+ 92	F_{a-4} (1)		MSH
+ 93			LSH
+ 94	F_{a-5} (1)		MSH
+ 95			LSH
+ 96	F_{a-6} (1)		MSH
+ 97		↓	LSH
+ 98	$''F_a(X)(1)$		MSH
+ 99			LSH
+100	$'F_{a-1/2}(X)$		MSH
+101			LSH
+102	$''F_6^J(X)(1)$		MSH
+103			LSH
+104	$''F_6^{J-1}(X)$		MSH
+105			LSH
+106	R_{xa} (1)		MSH
+107			LSH
+108	V_{xa} (1)		MSH
+109			LSH
+110	N (1)		
+111	T_i		
+112	$''F_a(Y)(1)$		MSH

TABLE 2 TEMNO TABLE FORMAT (SHEET 5 OF 6)

LOCATION	CONTENTS	
TEMNO +113		LSH
+114	$'F_{a-1/2}(Y)$	MSH
+115		LSH
+116	$''F_6^J(Y)(1)$	MSH
+117		LSH
+118	$''F_6^{J-1}(Y)$	MSH
+119		LSH
+120	R_{Ya} (1)	MSH
+121		LSH
+122	V_{Ya} (1)	MSH
+123		LSH
+124	IR1 } N1SUFT	
+125	IR2 }	
+126	$''F_{a(Z)}$ (1)	MSH
+127		LSH
+128	$'F_{a-1/2}$ (Z)	MSH
+129		LSH
+130	$''F_6^J(Z)(1)$	MSH
+131		LSH
+132	$''F_6^{J-1}(Z)$	MSH
+133		LSH
+134	R_{Za} (1)	MSH
+135		LSH
+136	V_{Za} (1)	MSH
+137		LSH
+138	h^2 (1)	MSH
+139		LSH
+140	$G_1(F_1 + F_{-1})$ OR $H_1(F_1 - F_{-1})$	MSH
+141		LSH

TABLE 2 TEMNO TABLE FORMAT (SHEET 6 OF 6)

LOCATION		CONTENTS	
TEMNO	+142	$G_2(F_2+F_{-2})$ OR $H_2(F_2-F_{-2})$	MSH
	+143		LSH
	+144	$\Sigma G_k (F_k + E_k)$ OR $\Sigma H_k (F_k - E_k)$ k = 1, 2, 3	MSH
	+145		LSH
	+146	≠ 0=BACK INTEGRATION, 0=FORWARD INTEGRATION	
	+147	IR1 } N2EXCR	
	+148	IR2 } N2EXCR	
	+149	IR4 } N2EXCR	
	+150	} KOOOTI	
	+151	} KOOOTI	
	+152	} KOOOTI	MSH
	+153	} KOOOOH (FLT. MIN. OR SEC.)	LSH
	+154	} KOOOOH (FLT. MIN. OR SEC.)	MSH
	+155	} KOOOOH (FLT. MIN. OR SEC.)	LSH
	+156	h (1)	MSH
	+157		LSH
	+158	WCRK CELL	
	+159	WCRK CELL	
	+160	J (1)	
	+161	IR4 N1SUFT	
	+162	WCRK CELL	
	+163	≠ 0 GO THROUGH CALC. IN N4DRAG = 0 BYPASS ALTITUDE CALC. IN N4DRAG	

NUMERICAL INTEGRATICN ERROR RETURNS

THE FOLLOWING ARE THE TYPE OF ERRORS WHICH MAY OCCUR IN NUMERICAL INTEGRATICN.

101 ERROR IN FIXING TIME DURING BACKWARD INTEGRATICN. (NOCPNI)

102 ERROR IN FIXING TIME DURING FORWARD INTEGRATICN. (NOCPNI)

103 ERROR IN COMPUTING GEOCENTRIC LATITUDE. (NOCPNI)

104 ERROR IN COMPUTING LONGITUDE, LATITUDE AND ALTITUDE FOR ''CN-BOARD'' EQUATIONS. (NOCPNI)

105 ERROR IN COMPUTING VELOCITY MAGNITUDE. (N4DRAG)

106 ERROR IN COMPUTING LONGITUDE, LATITUDE, AND ALTITUDE FOR DRAG. (N4DRAG)

107 ERROR IN FIXING TIME FOR TOOINC. (N4DRAG)

108 ERROR IN COMPUTING COMPCNENTS OF LIFT. (N4DRAG)

109 ERROR IN COMPUTING LOG (V^2/A) (N4DRAG)

113 ERROR IN COMPUTING DENSITY. (N4PDEN)

115 ERROR IN COMPUTING VELOCITY OF SOUND. (N4COEF)

116 NO C_D OR C_L IN TABLE FOR MEDIUM ALTITUDE. (N4COEF)

117 ERROR IN COMPUTING C_D OR C_L FOR LOW ALTITUDE CASE. (N4COEF)

119 ERROR IN COMPUTING LONGITUDE, LATITUDE, AND ALTITUDE FOR VECTOR AT KOOOTA. (N5CAPI)

120 ERROR IN COMPUTING LONGITUDE, LATITUDE, AND ALTITUDE FOR CAPE VECTORS AT KOOOTI.

(N5CAPS)

121 ERROR IN COMPUTING RANGE. (N60PEP)

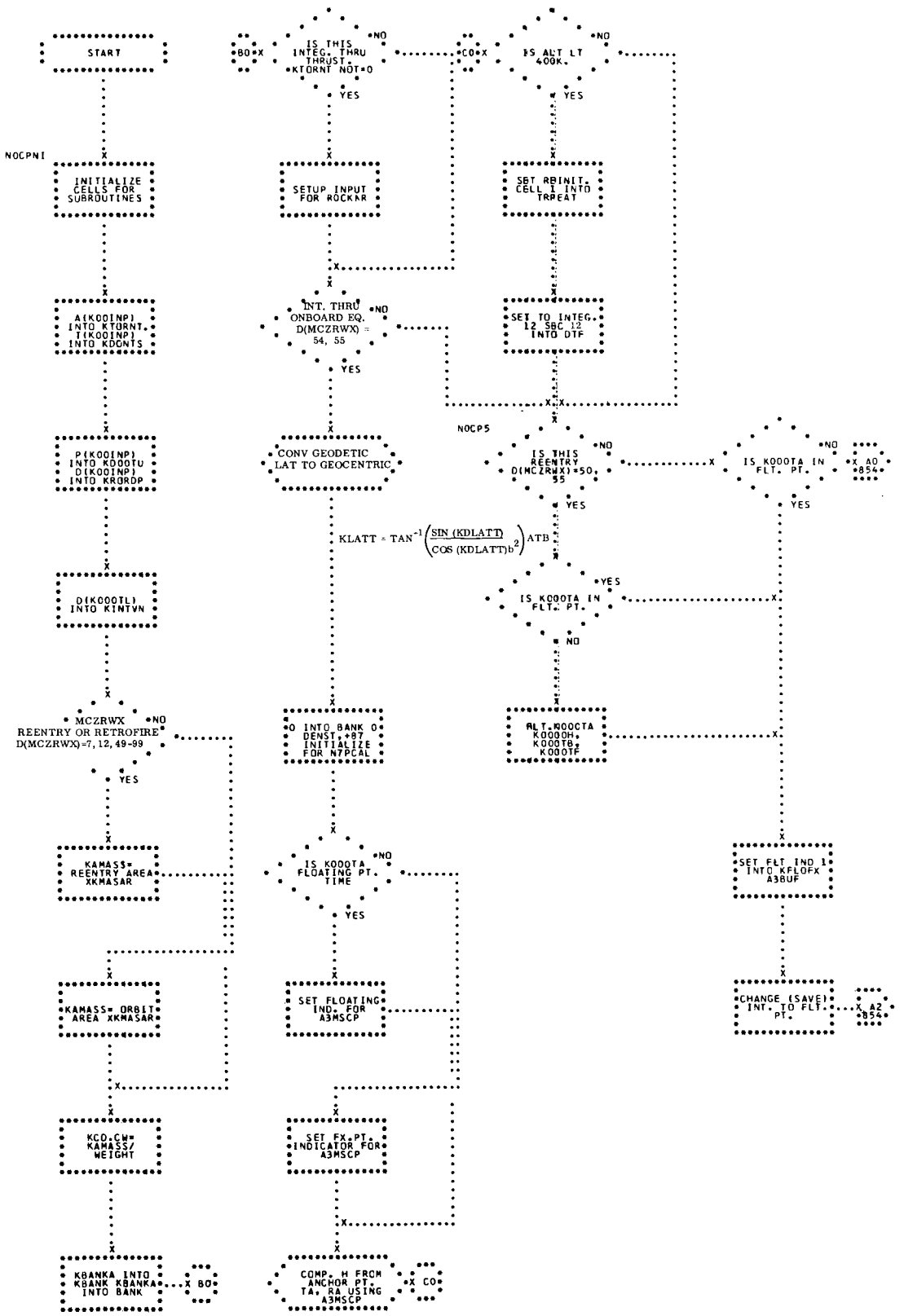
122 NON-CONVERGENCE CF INTEGRATICN PROCESS. (N7VARS)

123 ERROR IN CN-BOARC GUIDANCE EQUATIONS. (N7PCAL)

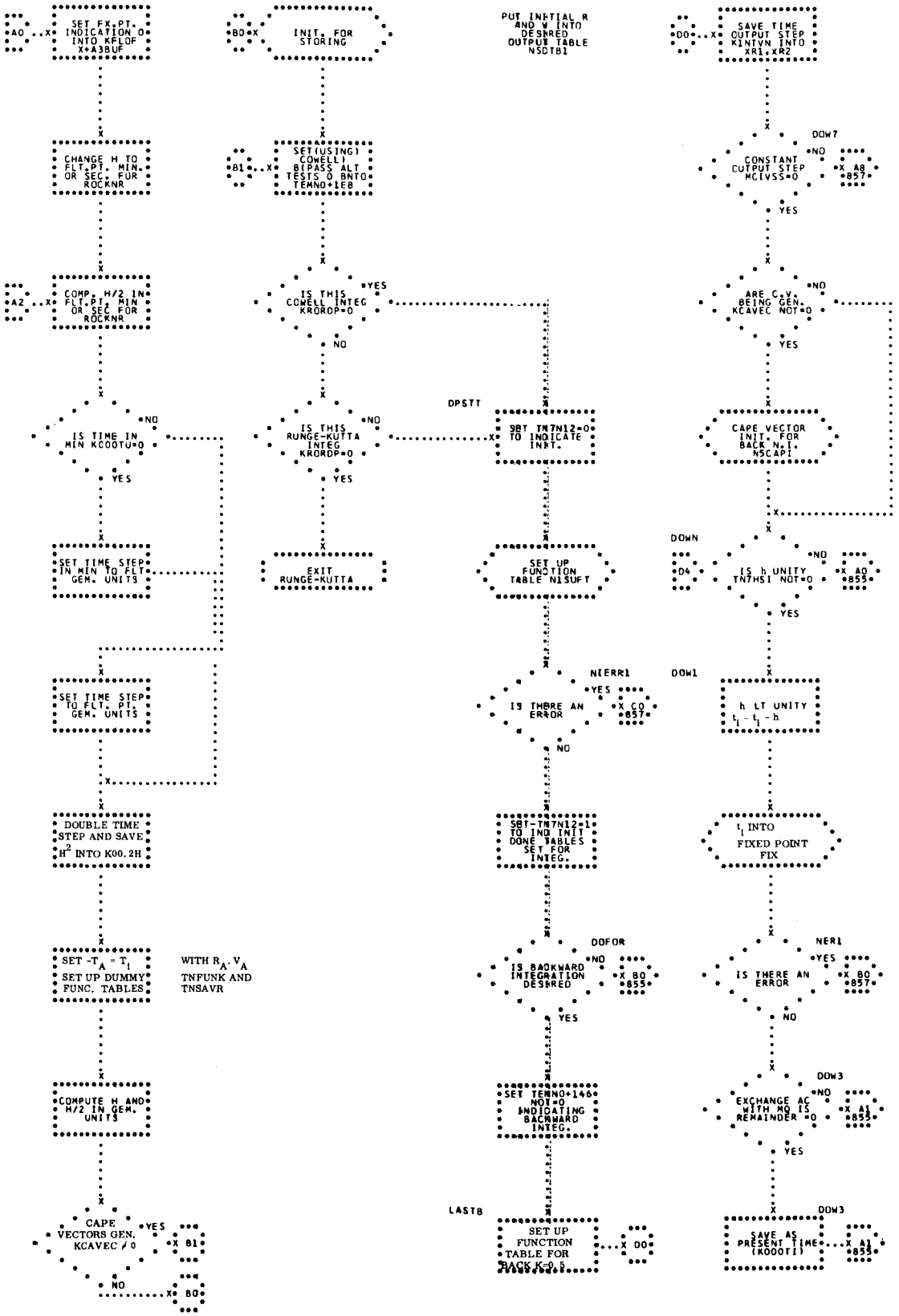
124 ERROR IN MANEUVER PHASE ANGLE TESTS. (N80TST)

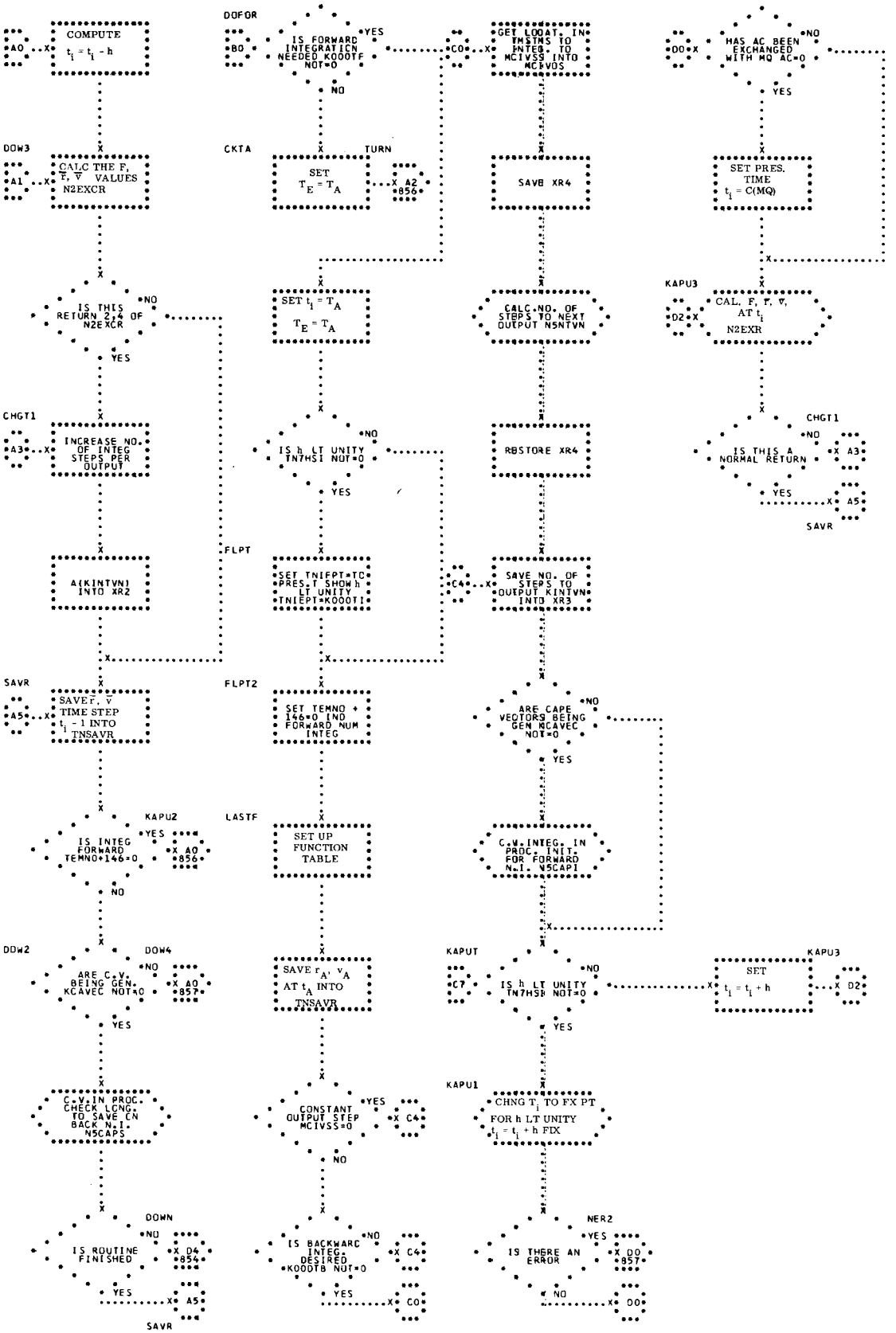
125 ERROR IN FIXING TIMES FOR OUTPUT TABLE. (N80TST)

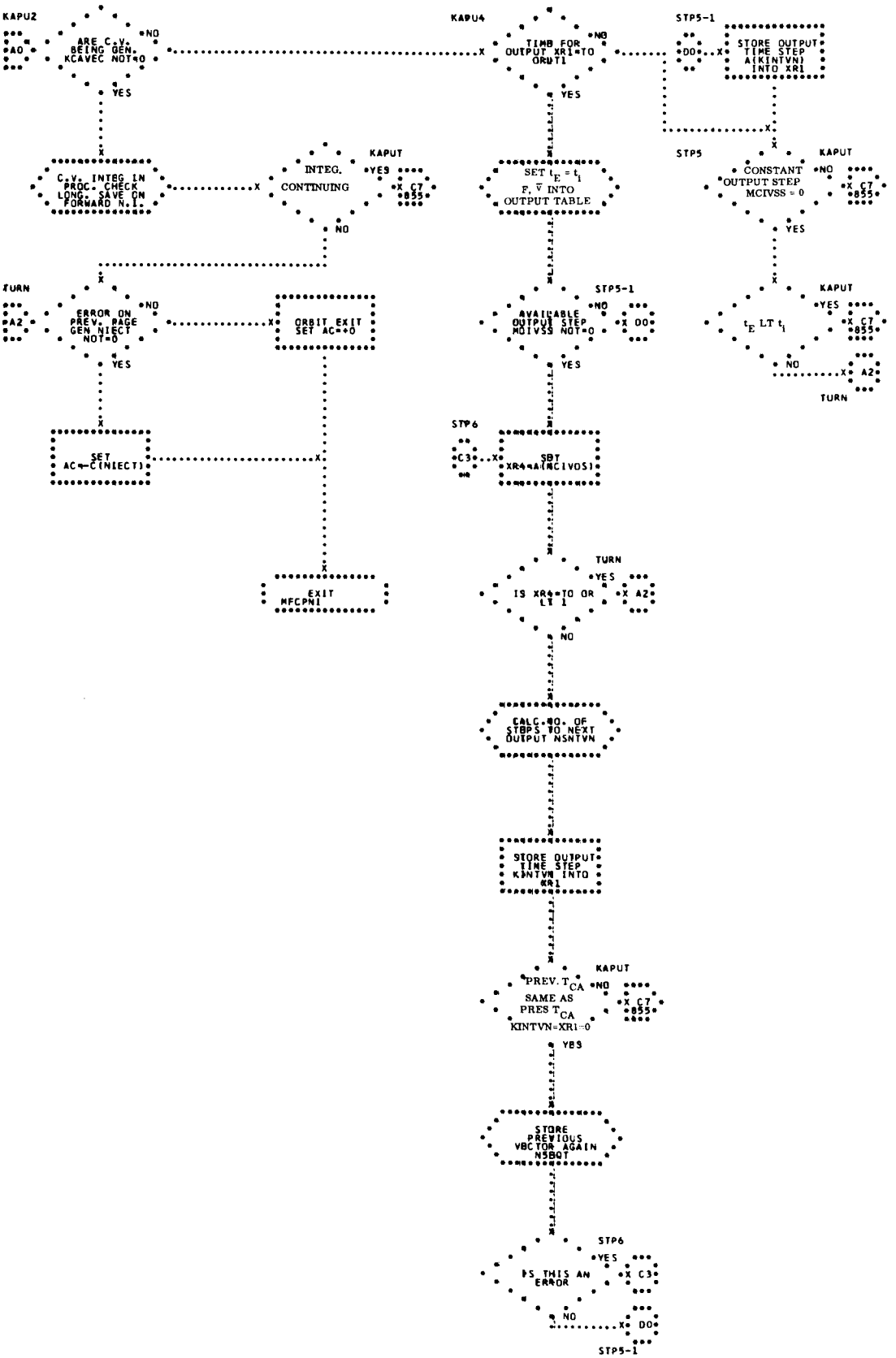
126 ERROR IN THRUSTER. (ROCRNR)

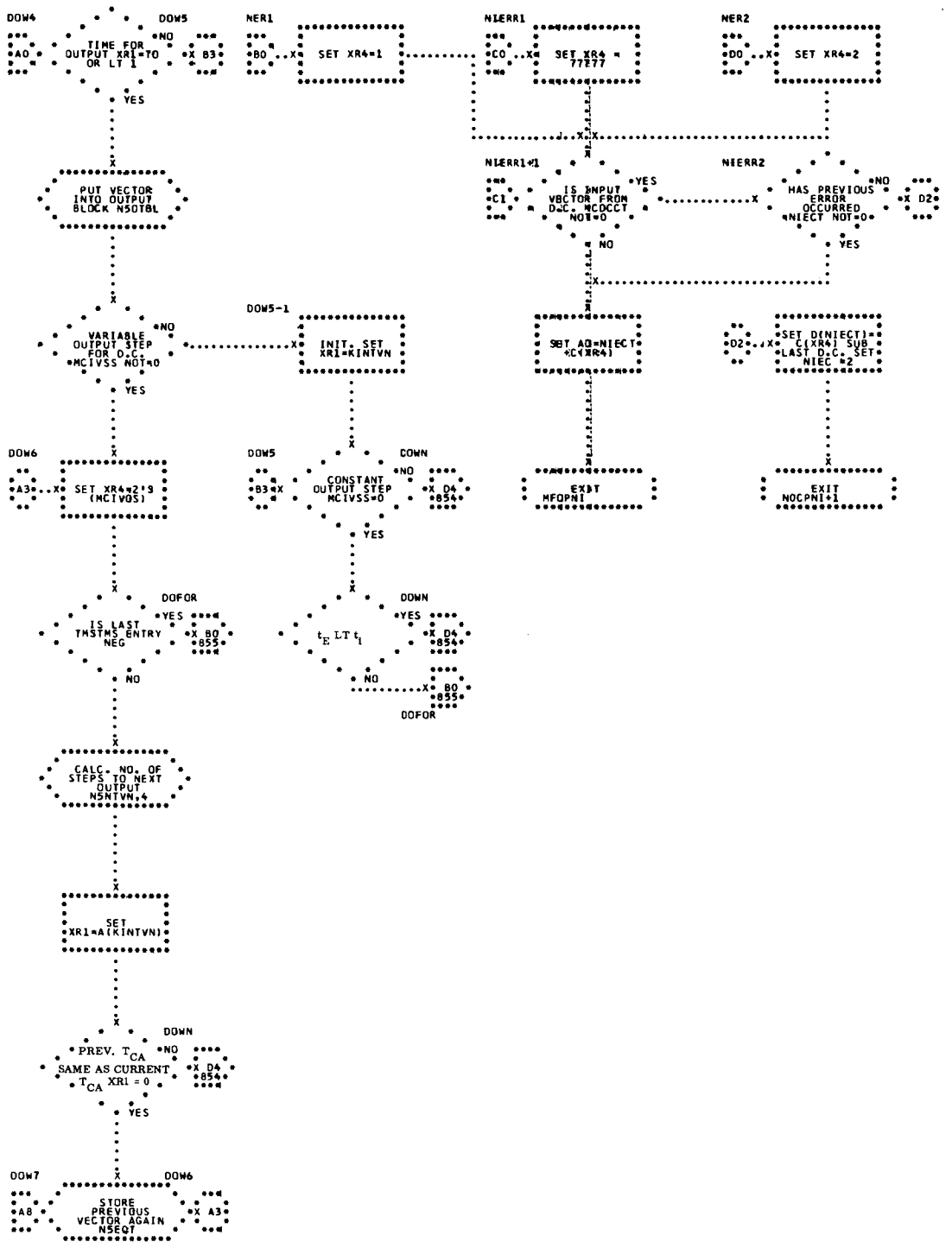


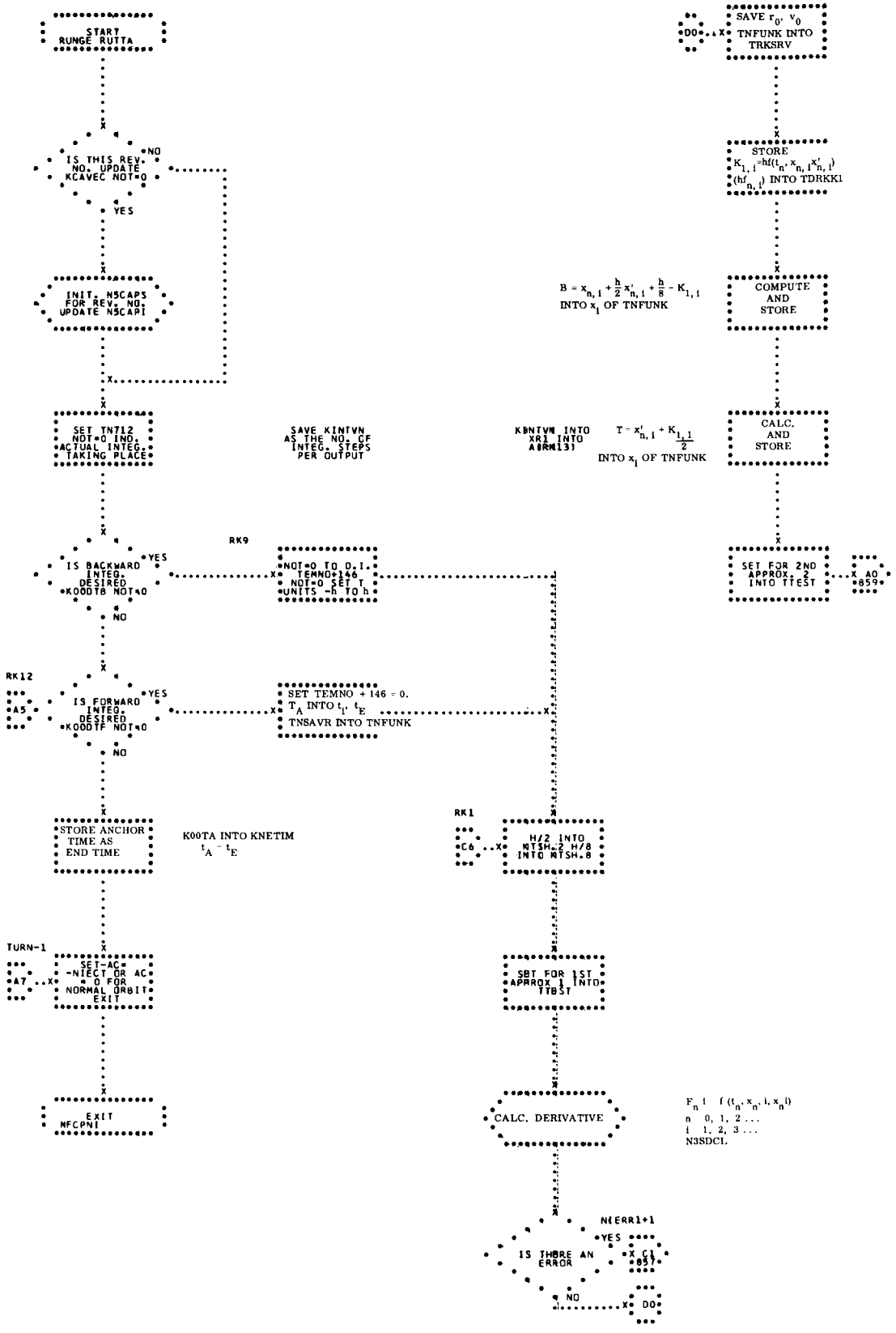
$$KLATT = TAN^{-1}\left(\frac{SIN\ (KDLATT)}{COS\ (KDLATT)b^2}\right) ATB$$

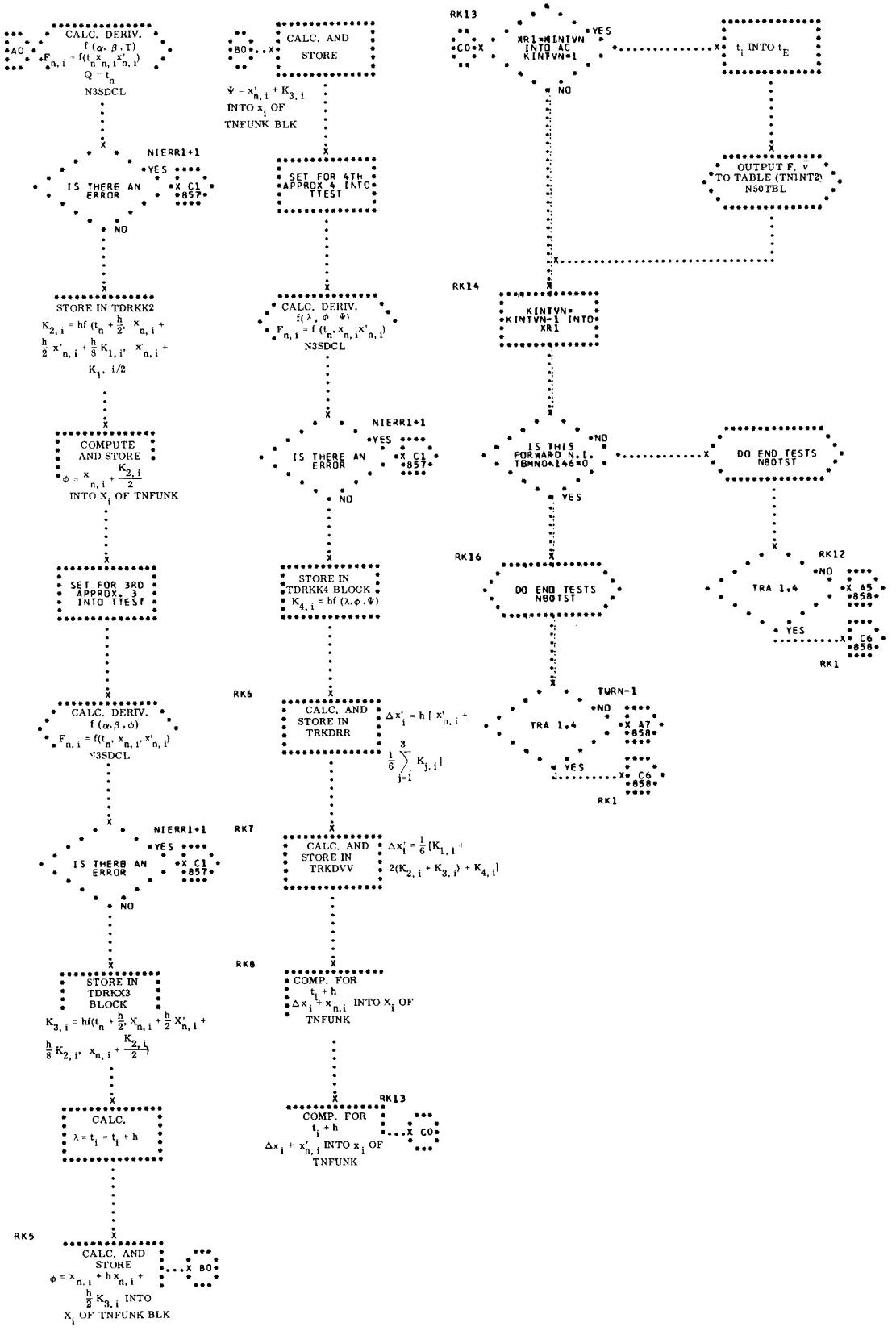

SET FX.PT. INDICATION=0 INTO KFLOF X+A3BUF
CHANGE H TO FLT.PT. MIN. OR SEC. FOR ROCKNR
COMP. H/2 IN FLT.PT. MIN OR SEC FOR ROCKNR
IS TIME IN MIN KCOOTU=0
NO
YES
SET TIME STEP IN MIN TO FLT. GEM. UNITS
SET TIME STEP TO FLT. PT. GEM. UNITS
DOUBLE TIME STEP AND SAVE H² INTO K00.2H
SET -T_A = T_I
SET UP DUMMY FUNC. TABLES
WITH R_A, V_A TNFUNK AND TNSAVR
COMPUTE H AND H/2 IN GEM. UNITS
CAPE VECTORS GEN. KCAVEC ≠ 0
YES
NO
X 81
X 80
INIT. FOR STORING
SET(USING) COWELL) BYPASS ALT TESTS O BNTO TEMNO+1E8
IS THIS COWELL INTEG KRORDP=0
YES
NO
IS THIS RUNGE-KUTTA INTEG KRORDP=0
NO
YES
EXIT RUNGE-KUTTA
PUT INITIAL R AND V INTO DESIRED OUTPUT TABLE N5OTB1
DPSTT
SBT TN7N12=0 TO INDICATE INIT.
SET UP FUNCTION TABLE N1SUFT
NIERR1
IS THERE AN ERROR
YES
X C0
NO
SBT-TN7N12=1 TO IND INIT DONE TABLES SET FOR INTEG.
DOFOR
IS BACKWARD INTEGRATION DESIRED
NO
YES
X 80
SET TEMNO+146 NOT=0 INDICATING BACKWARD INTEG.
LAST8
SET UP FUNCTION TABLE FOR BACK K=0.5
X D0
SAVE TIME OUTPUT STEP KINTVN INTO XR1, XR2
D0
DOW7
CONSTANT OUTPUT STEP MCIVSS=0
NO
YES
X A8
X 857
ARE C.V. BEING GEN. KCAVEC NOT=0
NO
YES
CAPE VECTOR INIT. FOR BACK N.I. N5CAPI
DOWN
IS h UNITY TN7HS1 NOT=0
NO
YES
X A0
X 855
DOW1
h LT UNITY
$t_I - t_I - h$
t_I INTO FIXED POINT FIX
NER1
IS THERE AN ERROR
NO
YES
X 80
X 857
DOW3
EXCHANGE AC WITH MQ IS REMAINDER =0
NO
YES
X A1
X 855
DOW3
SAVE AS PRESENT TIME (K000TI)
X A1
X 855


A0
COMPUTE
$t_i = t_i - h$
DOW3
A1
CALC THE F, $\dot{r}$, $\bar{v}$ VALUES N2EXCR
IS THIS RETURN 2,4 OF N2EXCR
NO
YES
CHGT1
A3
INCREASE NO. OF INTEG STEPS PER OUTPUT
A(KINTVN) INTO XR2
SAVR
A5
SAVE r, $\bar{v}$ TIME STEP t_i - 1 INTO TNSAVR
KAPU2
IS INTEG FORWARD TEMNO+146=0
YES
A0
856
NO
DDW2
DOW4
ARE C.V. BEING GEN. KCAVEC NOT=0
857
C.V.IN PROC. CHECK LONG. TO SAVE CN BACK N.I. N5CAPS
DOWN
IS ROUTINE FINISHED
D4
854
SAVR
A5
DOFOR
B0
IS FORWARD INTEGRATION NEEDED KOOOTF NOT=0
C0
NO
CKTA
SET $T_E = T_A$
TURN
A2
856
SET $t_i = T_A$
$T_E = T_A$
IS h LT UNITY TN7HSI NOT=0
YES
FLPT
SET TNIEPT=TO PRES.T SHOW h LT UNITY TNIEPT=KOOOTI
C4
SAVE r_A, v_A AT t_A INTO TNSAVR
FLPT2
SET TEMNO + 146=0 IND FORWARD NUM INTEG
LASTF
SET UP FUNCTION TABLE
CONSTANT OUTPUT STEP MCIVSS=0
C4
IS BACKWARD INTEG. DESIRED KOOOTB NOT=0
C0
GET LOOAT, IN TMSTMS TO INTEG. TO MCIVSS INTO MCIVOS
SAVB XR4
CALC.NO. OF STEPS TO NEXT OUTPUT N5NTVN
RBSTORE XR4
SAVE NO. OF STEPS TO OUTPUT KINTVN INTO XR3
ARE CAPE VECTORS BEING GEN KCAVEC NOT=0
YES
C.V.INTEG. IN PROC. INIT. FOR FORWARD N.I. N5CAPI
KAPUT
C7
IS h LT UNITY TN7HSB NOT=0
YES
KAPU1
CHNG T_i TO FX PT FOR h LT UNITY $t_i = t_i + h$ FIX
NER2
IS THERE AN ERROR
YES
D0
857
NO
D0
KAPU3
SET $t_i = t_i + h$
D2
HAS AC BEEN EXCHANGED WITH MQ AC=0
NO
YES
SET PRES. TIME t_i = C(MQ)
KAPU3
D2
CAL. F, $\dot{r}$, $\bar{v}$, AT t_i N2EXR
CHGT1
IS THIS A NORMAL RETURN
NO
A3
YES
A5
SAVR


KAPU2
ARE C.V. BEING GEN. KCAVEC NOT=0
NO
YES
AO
C.V. INTEG IN PROC. CHECK LONG. SAVE ON FORWARD N.I.
INTEG. CONTINUING
KAPUT
YES
X C7
855
NO
TURN
ERROR ON PREV. PAGE GEN NIECT NOT=0
A2
ORBIT EXIT SET AC=+0
SET AC←C(NIECT)
EXIT MFCPNI
KAPU4
TIMB FOR OUTPUT XR1=TO ORUT1
SET $t_E = t_i$ F, $\bar{V}$ INTO OUTPUT TABLE
STP5-1
AVAILABLE OUTPUT STEP MCIVSS NOT=0
DO
STP6
C3
SBT XR4=A(MCIVOS)
IS XR4=TO OR LT 1
CALC NO. OF STEPS TO NEXT OUTPUT NSNTVN
STORE OUTPUT TIME STEP KINTVN INTO XR1
PREV. T_{CA} SAME AS PRES T_{CA} KINTVN=XR1=0
STORE PREVIOUS VECTOR AGAIN NSBOT
IS THIS AN ERROR
STP5
STORE OUTPUT TIME STEP A(KINTVN) INTO XR1
CONSTANT OUTPUT STEP MCIVSS = 0
t_E LT t_i


DOW6
DOW5
NO
TIME FOR OUTPUT XR1=TO OR LT 1
A0
B3
YES
PUT VECTOR INTO OUTPUT BLOCK N5OTBL
VARIABLE OUTPUT STEP FOR D.C. MCIVSS NOT=0
NO
YES
DOW6
SET XR4=2'9 (MCIVOS)
DOFOR
IS LAST TMSTMS ENTRY NEG
YES
NO
B0
855
CALC. NO. OF STEPS TO NEXT OUTPUT N5NTVN,4
SET XR1=A(KINTVN)
DOWN
PREV. T_{CA} SAME AS CURRENT T_{CA} XR1 = 0
NO
YES
D4
854
DOW7
DOW6
STORE PREVIOUS VECTOR AGAIN NSEQT
A8
A3
NER1
B0
SET XR4=1
DOW5-1
INIT. SET XR1=KINTVN
DOWN
CONSTANT OUTPUT STEP MCIVSS=0
B3
NO
YES
DOWN
t_E LT t_1
NO
YES
B0
855
DOFOR
D4
854
NIERR1
C0
SET XR4 = 77777
NIERR1+1
IS INPUT VECTOR FROM D.C. MCDCCT NOT=0
C1
YES
NO
SET A0=NIECT+C(XR4)
EXIT MFOPNI
NER2
D0
SET XR4=2
NIERR2
HAS PREVIOUS ERROR OCCURRED NIECT NOT=0
NO
YES
D2
SET D(NIECT)= C(XR4) SUB LAST D.C. SET NIEC +2
EXIT NOCPNI+1


START RUNGE RUTTA
IS THIS REV. NO. UPDATE KCAVEC NOT=0
NO
YES
INIT. N5CAPS FOR REV. NO. UPDATE N5CAPI
SET TNT12 NOT=0 IND. ACTUAL INTEG. TAKING PLACE
SAVE KINTVN AS THE NO. OF INTEG. STEPS PER OUTPUT
KINTVN INTO XR1 INTO A(RM13)
IS BACKWARD INTEG. DESIRED *KOODTB NOT=0
YES
NO
RK9
*NOT=0 TO D.I. TEMNO+146 NOT=0 SET T UNITS -h TO h
RK12
IS FORWARD INTEG. DESIRED *KOODTF NOT=0
YES
NO
SET TEMNO + 146 = 0. T_A INTO t_I, t_E TNSAVR INTO TNFUNK
STORE ANCHOR TIME AS END TIME
KOOTA INTO KNETIM
$t_A - t_E$
TURN-1
SET-AC -NIECT OR AC = 0 FOR NORMAL ORBIT EXIT
EXIT NFCPNI
RK1
C6
H/2 INTO MTSH.2 H/8 INTO MTSH.8
SBT FOR 1ST APPROX. 1 INTO TTBST
CALC. DERIVATIVE
$F_{n\,i} \quad f(t_n, x_{n,i}, x_n)$
n 0, 1, 2 ...
i 1, 2, 3 ...
N3SDCL
N(ERR1+1)
IS THERE AN ERROR
YES
X C1 *857*
NO
DO
DO
SAVE r_0, v_0 TNFUNK INTO TRKSRV
STORE $K_{1,i} = hf(t_n, x_{n,i} x'_{n,i})$ $(hf_{n,i})$ INTO TDRKK1
$B = x_{n,i} + \frac{h}{2} x'_{n,i} + \frac{h}{8} - K_{1,i}$ INTO x_i OF TNFUNK
COMPUTE AND STORE
$T = x'_{n,i} + K_{1,1}/2$ INTO x_i OF TNFUNK
CALC. AND STORE
SET FOR 2ND APPROX. 2 INTO TTEST
X A0 *859*


CALC. DERIV. f (α, β, T) F n, i = f(t n, x n, i, x' n, i) Q = t n N3SDCL
AO
IS THERE AN ERROR
NIERR1+1
YES
X C1 *857*
NO
STORE IN TDRKK2 K 2, i = hf (t n + h/2, x n, i + h/2 x' n, i + h/8 K 1, i, x' n, i + K 1, i/2
COMPUTE AND STORE φ = x n, i + K 2, i/2 INTO X i OF TNFUNK
SET FOR 3RD APPROX, 3 INTO TTEST
CALC. DERIV. f (α, β, φ) F n, i = f(t n, x n, i, x' n, i) N3SDCL
IS THERB AN ERROR
STORE IN TDRKX3 BLOCK K 3, i = hf(t n + h/2, X n, i + h/2 X' n, i + h/8 K 2, i, x n, i + K 2, i/2)
CALC. λ = t i = t i + h
RK5
CALC. AND STORE φ = x n, i + h x n, i + h/2 K 3, i INTO X i OF TNFUNK BLK
X B0
B0
CALC. AND STORE Ψ = x' n, i + K 3, i INTO x i OF TNFUNK BLK
SET FOR 4TH APPROX 4 INTO TTEST
CALC. DERIV. f(λ, φ, Ψ) F n, i = f (t n, x n, i x' n, i) N3SDCL
STORE IN TDRKK4 BLOCK K 4, i = hf (λ, φ, Ψ)
RK6
CALC. AND STORE IN TRKDRR
Δx' i = h [x' n, i + 1/6 Σ j=1 3 K j, i]
RK7
CALC. AND STORE IN TRKDVV
Δx i = 1/6 [K 1, i + 2(K 2, i + K 3, i) + K 4, i]
RK8
COMP. FOR t i + h Δx i + x n, i INTO X i OF TNFUNK
RK13
COMP. FOR t i + h Δx i + x' n, i INTO x i OF TNFUNK
X C0
C0
RK13
XR1=KINTVN INTO AC KINTVN=1
YES
NO
t i INTO t E
OUTPUT F, v TO TABLE (TN1NT2) N50TBL
RK14
KINTVN=KINTVN-1 INTO XR1
IS THIS FORWARD N.I. TBMNO+146=0
NO
YES
DO END TESTS N80TST
RK16
DO END TESTS N80TST
TURN-1
TRA 1,4
X A7 *858*
YES
X C6 *858*
RK1
DO END TESTS N80TST
RK12
TRA 1,4
NO
X 45 *858*
YES
X C6 *858*
RK1

28. ROCKNR - THRUST PROGRAM

THIS SUBROUTINE COMPUTES THREE EARTH CENTERED COMPONENTS OF ACCELERATION DUE TO THRUST BASED UPON INPUTS OF THRUST DURATION AND MAGNITUDE FOR A VARIABLE N ROCKET. A CHANGING WEIGHT IS COMPUTED AT EACH TIME STEP. (SEE FLOWCHART.) ROCKNR USES SUBROUTINES -

U3MAT
SIN-COS
ARSIN
VMAG
U3DOT
UNITV
U3XPR

METHOD

A) DERIVATION OF THRUST ACCELERATION.

CONSIDER A ROCKET AT TIME t_0 AND A SYSTEM COMPOSED OF GAS PARTICLES AND A ROCKET AT TIME $t_0 + \Delta t$

	ROCKET	GAS PARTICLES	ROCKET
TIME	t_0	$t_0 + \Delta t$	
VELOCITY	V	$V + \Delta V - \Omega$	$V + \Delta V$
MASS	M	ΔM	$M - \Delta M$

THE LAW OF CONSERVATION OF LINEAR MOMENTUM GIVES

$$MV - [\Delta M(V + \Delta V - \omega) + (M - \Delta M)(V + \Delta V)] = -\Sigma F_{external} \Delta t$$

$$M\frac{\Delta V}{\Delta t} = \Sigma F_{ext.} - \omega \frac{\Delta M}{\Delta t}$$

$$\dot{V} = \frac{\Sigma F_{ext.}}{M} - \omega \frac{\dot{M}}{M}$$

THE ACCELERATION DUE TO THRUST IS

$$A_T = -\omega\frac{\dot{M}}{M}$$

WHERE WE ASSUME ω IS A CONSTANT GAS VELOCITY.
FOR AN INTEGRATION STEP

$$A_T \approx -\frac{\omega}{M}\frac{\Delta M}{\Delta t} \qquad 2)$$

ASSUMING A CONSTANT RATE OF CHANGE OF MASS $\dot{M}_c$ AND BURN DURATION Dt EQUATION 2 BECOMES, $A_T \approx -\omega(\dot{M}_c Dt)/(M\,\Delta t)$ 3)

LET $\omega M_c = T > 0$

THEN $A_T = \frac{T}{M}\frac{Dt}{\Delta t}$

FOR FURTHER ACCURACY WE CONSIDER A VARYING WEIGHT,

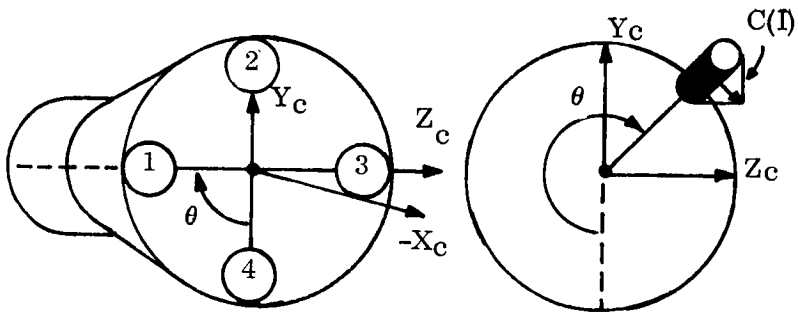
$$M_{AV} = \frac{M(I) + M(I+1)}{2}$$

$$M_{AV} = \frac{M(I) + [M(I) - \dot{M}_c Dt]}{2}$$

EQUATION 3 BECOMES,

$$A_T = \frac{T}{M_{AV}}\frac{Dt}{\Delta t} = g\frac{T}{W_{AV}}\frac{Dt}{\Delta t}$$

FOR THE CASE WHERE THERE ARE N ROCKETS POSITIONED AT ANGLES OF $\Theta(I)$ FROM A DIAMETER OF THE ROCKET, AND CANTED AT ANGLES C(I) WE ARRIVE AT THE FOLLOWING EQUATIONS AND CAPSULES AXES.



X Axis going into paper through longitudinal axis axis of rocket.

θ (1) = 90°
θ (2) = 180°
θ (3) = 270°
θ (4) = 0°

X AXIS THROUGH THE LONGITUDINAL AXIS OF THE ROCKET

Z AXIS IN THE VERTICAL PLANE

Y AXIS CHOSEN SUCH THAT $\bar{X} \times \bar{Y} = \bar{Z}$

$\Theta(I)$ = ANGULAR DISPLACEMENT OF THE ITH ROCKET FROM THE DIAMETER

C(I) = CANT ANGLE FOR THE ITH ROCKET

THE EQUATIONS IN THE CAPSULE COORDINATE SYSTEM ARE -

$$C_x = \frac{g}{\Delta t} \sum_{I=1}^{N} \frac{T(I)}{W(I)} \cos C(I) \times Dt(I)$$

$$C_z = +\frac{g}{\Delta t} \sum_{I=1}^{N} \frac{T(I)}{W(I)} \sin C(I) \times \sin \theta(I) \times Dt(I)$$

$$C_y = +\frac{g}{\Delta t} \sum_{I=1}^{N} \frac{T(I)}{W(I)} \sin C(I) \times \cos \theta(I) \times Dt(I)$$

WHERE

T(I) = THRUST IN LBS (SEE OPERATING INSTRUCTIONS, D)

W(I) = WEIGHT IN LBS = WEIGHT AVERAGE

C(I) = CANT ANGLE IN DEGREES

Dt(I) = BURN DURATION IN SECONDS

Δt = INTEGRATION TIME IN SECONDS

g = 1.00085551 - ACCELERATION DUE TO GRAVITY IN GEMINI UNITS

THE COMPONENETS OF ACCELERATION IN CAPSULE COORDINATES (C_x, C_y, C_z) ARE THEN CONVERTED BACK TO THE EARTH'S COORDINATE SYSTEM BY THE FOLLOWING ROTATIONS - (SEE OPERATING INSTRUCTIONS, E.)

$$\begin{pmatrix} C_{xo} \\ C_{yo} \\ C_{zo} \end{pmatrix} = \begin{pmatrix} C\alpha & 0 & +S\alpha \\ 0 & 1 & 0 \\ -S\alpha & 0 & C\alpha \end{pmatrix} \begin{pmatrix} C\psi & +S\psi & 0 \\ -S\psi & C\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & C\theta \\ 0 & -S\theta \end{pmatrix} \begin{pmatrix} 0 & C_x \\ S\theta & C_y \\ C\theta & C_z \end{pmatrix}$$

AND

$$(E_x, E_y, E_z) = (C_{xo}\ C_{yo}\ C_{zo}) \begin{pmatrix} -W_x & -W_y & -W_z \\ -U_x & -U_y & -U_z \\ -R_x & -R_y & -R_z \end{pmatrix}$$

WHERE

γ = FLIGHT PATH ANGLE

α = (γ + PITCH)

Θ_R = (T + ROLL) = (22.5° - R)

ψ = YAW

T = ROLL ZERO

$\overline{U}$ = $\overline{R}$ X $\overline{V}$ UNIT VECTOR

$\overline{W}$ $\overline{R}$ X ($\overline{R}$ X $\overline{V}$) UNIT VECTOR

$\overline{R}$ = RADIUS UNIT VECTOR

(C_{XO} C_{YO} C_{ZO}) = THRUST ACCELERATION ALONG $\overline{R}$, $\overline{R}$ X $\overline{V}$ AND $\overline{R}$ X ($\overline{R}$ X $\overline{V}$)

(E_X E_Y E_Z) =THRUST ACCELERATION IN EARTH CENTERED COORDINATES

REFERENCE - PAPER BY ED ROSS AND ART FUCHS
NOTES ON SUBROUTINE THRUST FOR THE GEMINI RUNGE KUTTA INTEGRATION PROGRAM.

B) ACCURACY - 7 DIGITS

USAGE

A) CALLING SEQUENCE -

```
TSX ROCKNR
PZE OUTPUT  ACCELERATION
            ERROR RETURN
            NORMAL RETURN
```

B) STORAGE REQUIRED -
396 + 8N LOCATIONS WHERE N IS THE NUMBER OF ROCKETS.

C) ERROR CODES -

THE ERROR RETURNS FROM ALL SUBROUTINES USED BY ROCKNR TRANSFER TO (2,4) IN THE CALLING SEQUENCE.

CODING INFORMATION

A) CONSTANTS -

K022.5	DEC	-.39269895	-22.5 DEGREES
KOODR.	DEC	.01745329252	DEG TO RADIANS
KOO.GG	DEC	1.00085551	CONVERTS ACCELERATION TO GEMINI UNITS

B) TIMING -

$$t = \{ (4 \text{ costs} + 250 \text{ cycles } 2.0 \times 10^{-6} \frac{\text{sec}}{\text{cycle}}) N$$

$$+444 \text{ cycles } 2.0 \times 10^{-6} \frac{\text{sec}}{\text{cycle}}$$

$$+ [6 \text{ cost} + U3DOTt + 2VMAGt + 3UNITVt + 4U3MATt + ARSINt$$

$$+ 2U3XPROT] \}$$

$t \approx 1.14N + 5$ millisec

for $N = 4$, $t \approx 9$ millisec

CHECKOUT

THREE HANDCHECKS WERE MADE, ONE OF WHICH COMPARED WITH A CHECK MADE BY THE ANALYSIS SECTION USING THEIR FORTRAN THRUST PROGRAM. THE OTHER TWO CHECKS COMPARED WITH THE HAND CHECKS TO 7 DIGITS.

OPERATING INSTRUCTION -

A) THRUSTER INPUT TABLE -

THE ADDRESS OF THE THRUST TABLE (TRUSTS IN EXAMPLE FOLLOWING) IS STORED IN LOCATION 14 OF THE INTEGER INPUT TABLE. FOLLOWING IS AN EXAMPLE OF INPUT TABLE (GT-3 VALUES).

	GENERAL INPUT	UNITS	OAMS NOMINALS	RETRO-FIRE NOMINALS
THRUSTS	$\Delta TIME_1$	FL. PT. SECS.	A(TMANUV+2)	5.5
+1	Δt_2		A(TMANUV+2)	5.5
+2	Δt_3	FL. PT. SECS.	0.	5.5
+3	Δt_4	FL. PT. SECS.	0.	5.5
+4	$GMTF_1$	FL. PT. SECS.	TMANUV	$GMTRF_1$
+5	$GMTF_2$	FL. PT. SECS.	TMANUV	$GMTRF_2$
+6	$GMTF_3$	FL. PT. SECS.	0.	$GMTRF_3$
+7	$GMTF_4$	FL. PT. SECS.	0.	$GMTRF_4$
+8	THRUST $LEVEL_1$	FL. PT. LBS.	94.5 OR-79.0	2490.
+9	TL_2	FL. PT. LBS.	94.5 OR-79.0	2490.
+10	TL_3	FL. PT. LBS.	0.	2490.
+11	TL_4	FL. PT. LBS.	0.	2490.
+12	ORIENTATION $ANGLE_1$	FL. PT. LBS.	270°	270°
+13	OA_2	FL. PT. RAD.	90°	180.°
+14	OA_3	FL. PT. RAD.	0.	90.°
+15	OA_4	FL. PT. RAD.	0.	0.°
+16	$CANT_1$	FL. PT. RAD.	0°	15°
+17	CA_2	FL. PT. RAD.	0°	15°
+18	CA_3	FL. PT. RAD.	0.	15°
+19	CA_4	FL. PT. RAD.	0.	15°
+20	PITCH	FL. PT. RAD.	TMANUV+3	16°
+21	ROLL	FL. PT. RAD.	0 °	0°
+22	YAW	FL. PT. RAD.	TMANUV+4	180 °
+23	THRUST IND.	FX. PT.	ZERO	NON-ZERO

+24	MASS EX-PULSION RATE	FL. LBS./SEC.	-0.7	-10.09

B) INPUT -

THE INPUT FROM THE THRUSTER INPUT TABLE ABOVE IS STORED IN THE FOLLOWING TABLES.

TT1MEO, TT1MEO+1,...,TT1MEO+N -	INITIAL FIRING TIME DIFFERENCE (FROM FIRING TIME) IN FLOATING POINT SECONDS
TT1MBO, TT1MBO+1,...,TT1MB)+N -	BURN DURATION IN FLOATING POINT SECONDS
THRUST, THRUST+1,...,THRUST+N -	THRUST IN LBS
TCANTL, TCANTL+1,...,TCANTL+N -	CANT ANGLE IN FLOATING POINT DEGREES
TTHETA, TTHETA+1,...,TTHETA+N -	C.C ANGLE FROM ROCKET TO REF AXIS
TIRET	TIME OF RETRO-FIRE IN FLOATING POINT SEC.

PITCH, ROLL, YAW IN FLOATING POINT RADIANS

TWEIGT	WEIGHT OF CAPSULE IN FLOATING POINT LBS.
TWEIGT+1	DW/Dt IN FLOATING POINT LB/SEC

TWEIGHT+2	SWITCH NON 0 FOR SETTING W(I)=W(I+1)
TIMONE	BEGINNING TIME FOR INTEGRATION STEP IN FLOATING POINT SECONDS
NUTIM	END TIME FOR INTEGRATION STEP IN FLOATING POINT SECONDS
F1XUV	SWITCH, 0 INITIALLY (FIXES REFERENCE PLANES)
REENPC	SWITCH 0 FOR PLANE CHANGE, NON ZERO FOR RETRO-FIRE
DTCSW	SWITCH, NON ZERO BY PASSES DT(I) COMPUTATION

THE FOLLOWING CARD IS ALSO NECESSARY -

MNROCK EQU N (WHERE N IS THE NUMBER OF ROCKETS)

C) ADAPTER INSTRUCTIONS (TADAPT) -

1. FOR USE WITH RUNGE KUTTA 4TH ORDER INTEGRATOR METHOD.

IN ADDITION TO THE ABOVE INPUT THE FOLLOWING INSTRUCTIONS MUST OCCUR -

PRIOR TO FIRST PASS THROUGH RUNGE KUTTA SET-
(SOLVING FOR K_1) TIMONE= t_0 =
TIME AT BEGINNING OF TIME STEP.

NUTIM = $t_0 + \frac{h}{2}$
DTCSW = 0
TWEIGHT+2 = 0

PRIOR TO SECOND PASS (SOLVING FOR K_2) SET -
TIMONE = t_0 (SAME AS FIRST PASS)
NUTIM = $t_0 + \frac{h}{2}$ (SAME AS FIRST PASS)
DTCSW = NON ZERO
TWEIGHT+2 = NON ZERO

PRIOR TO THIRD PASS (SOLVING FOR K_3) SET -
TIMONE = $t_0 + \frac{h}{2}$ = NUTIM
NUTIM = $t_0 + h$
DTCSW = 0
TWEIGHT+2 = 0

PRIOR TO FOURTH PASS (SOLVING FOR K_4 SET-

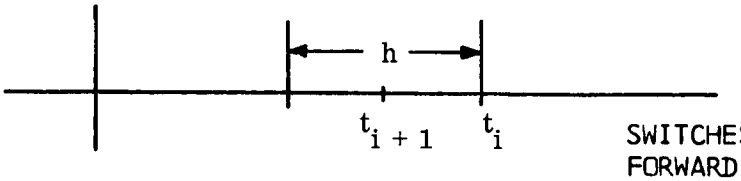
TIMONE = $t_0 + \frac{h}{2}$ (SAME AS THIRD PASS)
NUTIM = $t_0 + h$ SAME AS THIRD PASS)
DTCSW = NON ZERO
TWEIGHT+2 = NON ZERO

2. FOR USE WITH COWELLS INTEGRATION METHOD -
SET -

TIMONE = t_0
NUTIM = $t_0 + h$
DTCSW = 0
TWEIGHT+2 = NON ZERO

FOR BACKWARD INTEGRATION SEE STEP 3. REENPC MUST BE SET NON ZERO FOR RETROFIRE.

3. BACKWARDS INTEGRATION - (RUNGE KUTTA 4TH ORDER)



$h < 0$

WDOT > 0

SWITCHES ARE THE SAME AS FOR FORWARD INTEGRATION

1ST PASS -

TIMONE $= t_i + \frac{h}{2}$

NUTIM $= t_i$

2ND PASS -

SAME AS FIRST PASS.

3RD PASS -

NUTIM $= t_i + \frac{h}{2}$ = TIMONE

TIMONE $= t_{i+1}$ = TIMONE $+ \frac{h}{2}$

4TH PASS -

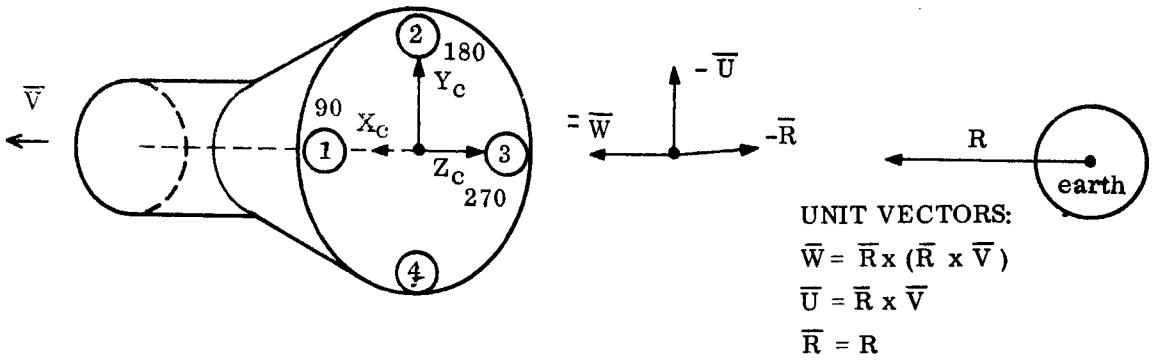
SAME AS THIRD PASS.

D) ROCKET THRUST SIGN NOMENCLATURE -

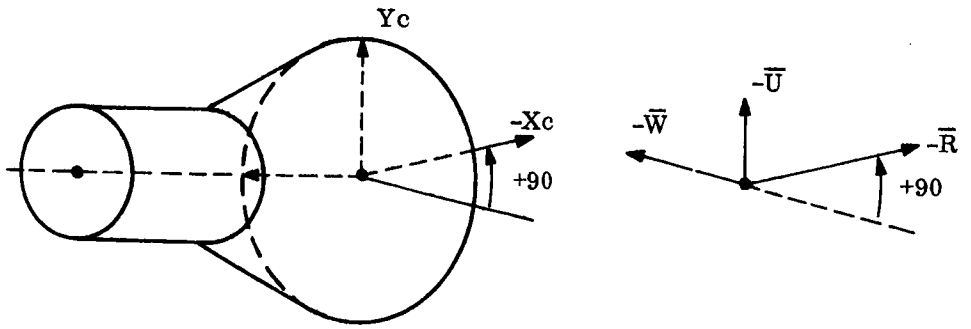
1. RETROFIRE ROCKETS - POSITIVE THRUST
2. AFT THRUSTERS - POSITIVE THRUST
3. FORWARD THRUSTERS - NEGATIVE THRUST

E) ROCKET ATTITUDE ANGLE DESCRIPTION -

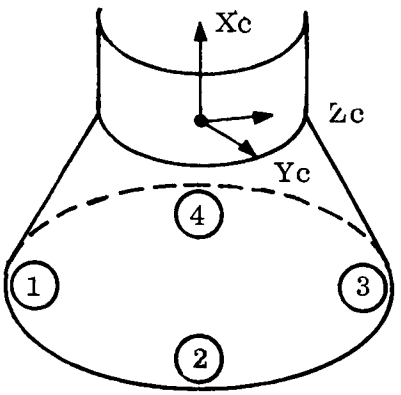
1. REFERENCE POSITION - 0 PITCH, ROLL, AND YAW.

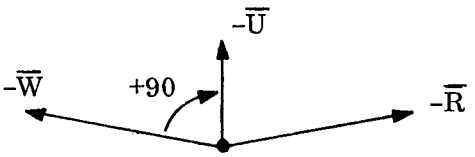


2. PITCH = 90° IN $\overline{R}$, $\overline{W}$ PLANE

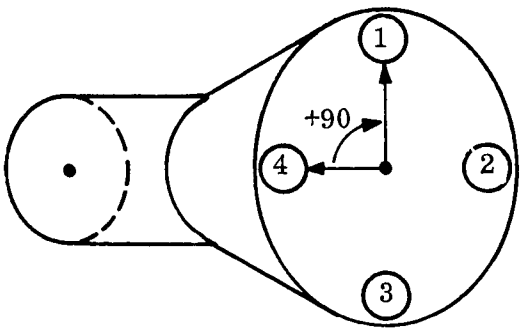


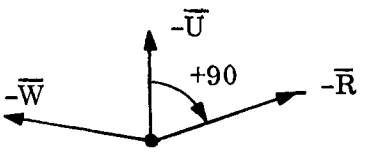
3. YAW = +90° IN $\bar{U}$, $\bar{W}$ PLANE

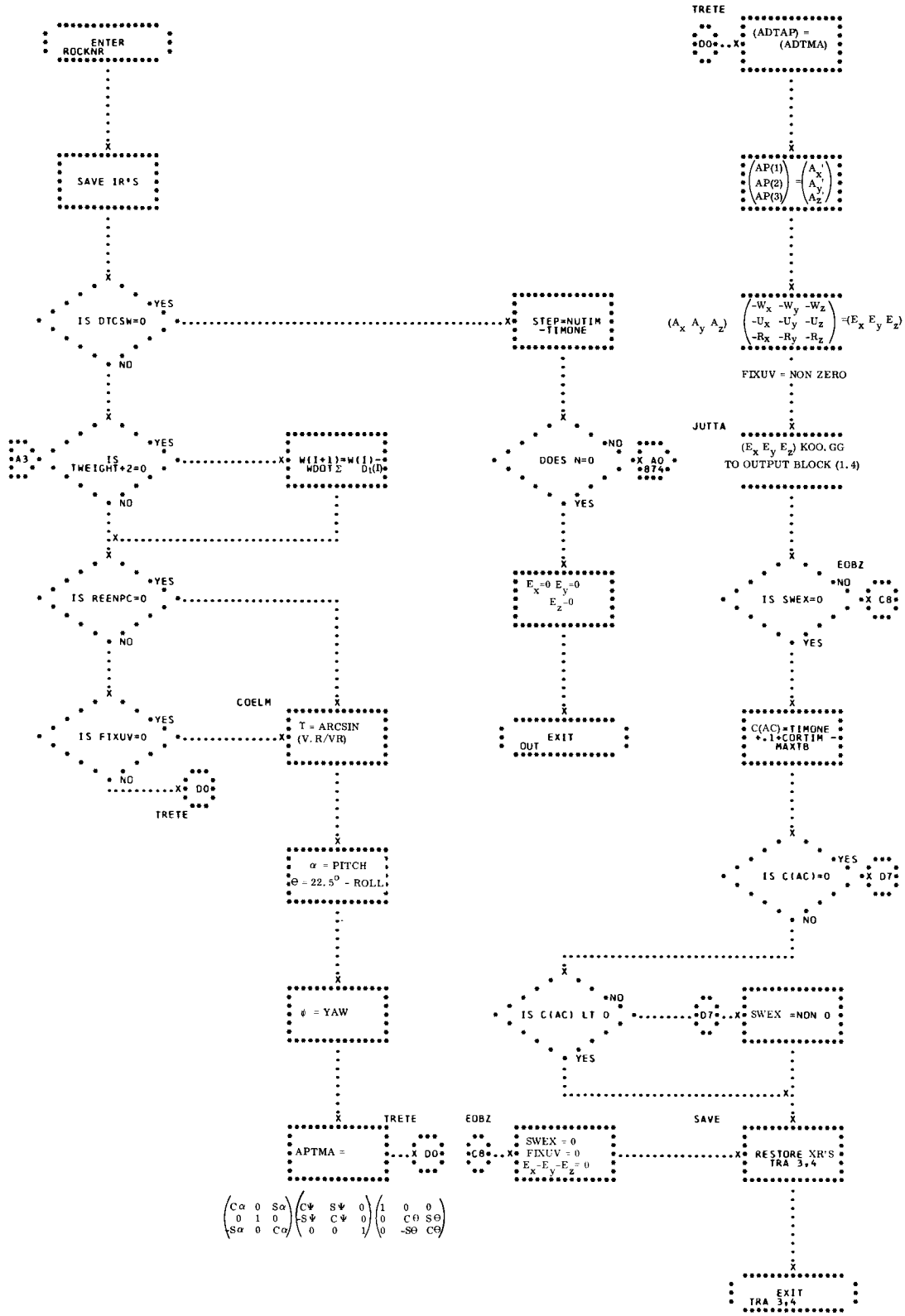


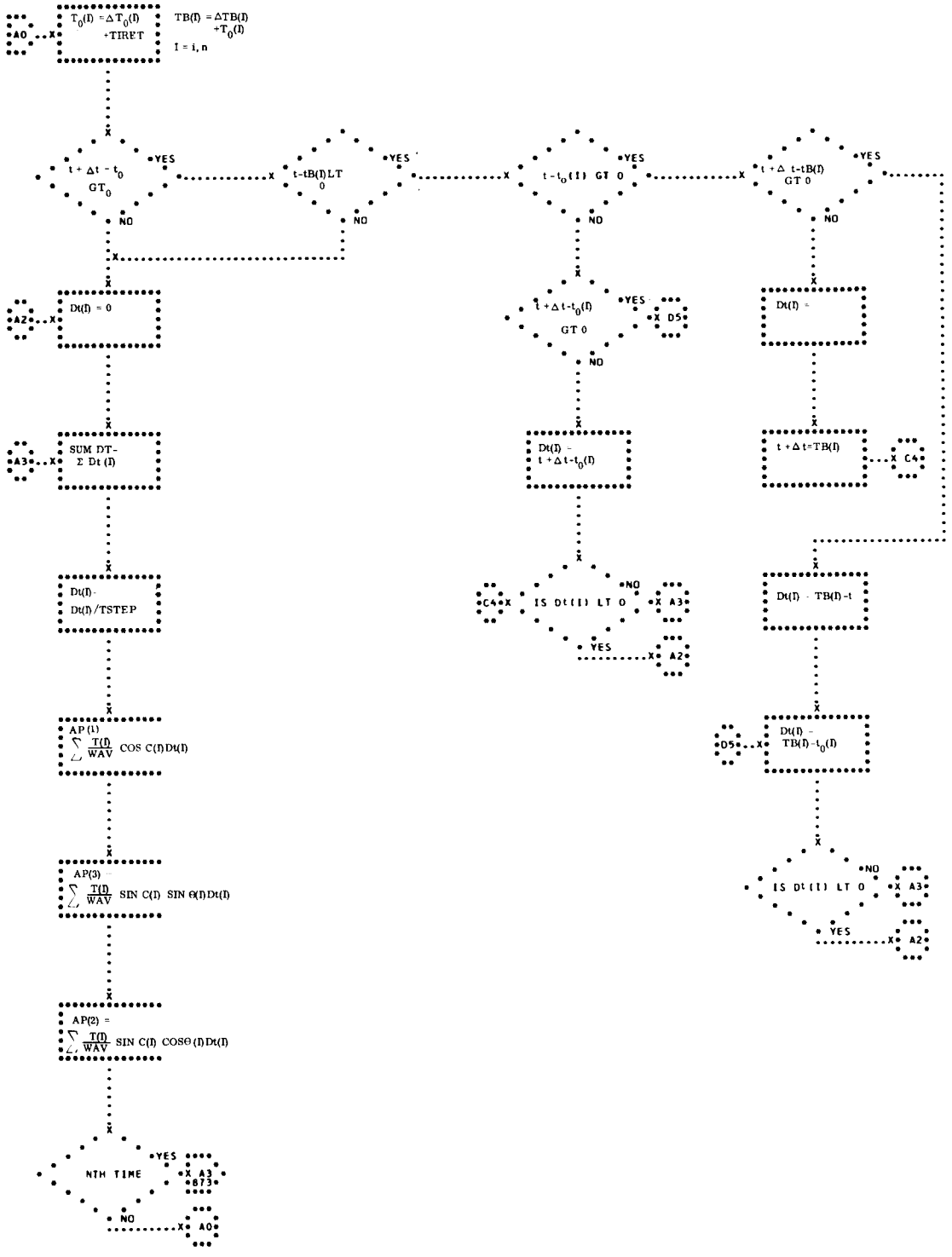


4. ROLL = +90° IN $\bar{U}$, $\bar{R}$ PLANE






ENTER ROCKNR
SAVE IR'S
IS DTCSW=0
YES
NO
IS TWEIGHT+2=0
A3
W(I+1)=W(I)-WDOT Σ Di(I)
IS REENPC=0
IS FIXUV=0
COELM
T = ARCSIN (V.R/VR)
DO
TRETE
α = PITCH
Θ = 22.5° - ROLL
φ = YAW
APTMA =
STEP=NUTIM -TIMONE
DOES N=0
A0 874
E_x=0 E_y=0 E_z=0
EXIT
OUT
TRETE
(ADTAP) = (ADTMA)
(AP(1) AP(2) AP(3)) = (A_x' A_y' A_z')
(A_x A_y A_z) (-W_x -W_y -W_z / -U_x -U_y -U_z / -R_x -R_y -R_z) =(E_x E_y E_z)
FIXUV = NON ZERO
JUTTA
(E_x E_y E_z) KOO. GG
TO OUTPUT BLOCK (1.4)
EOBZ
IS SWEX=0
C8
C(AC)=TIMONE +.1+CORTIM -MAXTB
IS C(AC)=0
D7
IS C(AC) LT 0
SWEX =NON 0
SWEX = 0
FIXUV = 0
E_x-E_y-E_z = 0
SAVE
RESTORE XR'S
TRA 3,4
EXIT
TRA 3,4


A0
$T_0(I) = \Delta T_0(I) + TIRET$
$TB(I) = \Delta TB(I) + T_0(I)$
$I = i, n$
$t + \Delta t - t_0$ GT 0
YES
NO
t-tB(I) LT 0
$t - t_0(I)$ GT 0
$t + \Delta t - tB(I)$ GT 0
A2
Dt(I) = 0
$t + \Delta t - t_0(I)$ GT 0
D5
Dt(I) =
A3
SUM DT = Σ Dt (I)
Dt(I) = $t + \Delta t - t_0(I)$
$t + \Delta t = TB(I)$
C4
Dt(I) = Dt(I)/TSTEP
IS Dt(I) LT 0
Dt(I) = TB(I) - t
AP(1) = $\sum \frac{T(I)}{WAV}$ COS C(I) Dt(I)
Dt(I) = TB(I) - $t_0(I)$
AP(3) = $\sum \frac{T(I)}{WAV}$ SIN C(I) SIN Θ(I) Dt(I)
AP(2) = $\sum \frac{T(I)}{WAV}$ SIN C(I) COSΘ(I) Dt(I)
NTH TIME
A3 873
A0

29. MFCPNI - MONITOR SUFFIX TO NUMERICAL INTEGRATION

MFCPNI TERMINATES THE USE OF THE INTEGRATOR BY SETTING UP THE QTSX RETURN. (SEE FLOWCHART.)

INPUT

A) TABLES -

TMSAVE	CONTAINS THE PANEL LOCATIONS OF ALL PROCESSORS IN PRIORITY ORDER. THE PANEL IS THE BLOCK OF MACHINE REGISTERS AND CELLS SAVED DURING TRAPPING.
COMMON	A 6-WORD BLOCK OF STORAGE SHARED BY LIBRARY SUBROUTINES. THE LAST WORD CONTAINS THE ROUTINE NUMBER OF ANY ROUTINE IN PROGRESS AT THE TIME OF A SIMULATED TRAP (QTSX).

B) COMMUNICATION CELL -

MCZRWX	CONTAINS THE QUEUE WORD TO MPCPNI. DECREMENT CONTAINS AN INTEGER THAT INDICATES THE USE OF THE INTEGRATION (SEE MPCPNI, SECTION 26.)

C) CONSTANTS -

K00054
K00050
K00006
K00004

K00003

D) MACROS -

TRNOF

QENBZ

OUTPUT

A)	MCNIAC	CONTAINS THE CONTENT OF THE ACCUMULATOR UPON EXIT FROM NOCPNI. IF THERE WAS AN ERROR IN INTEGRATION, THIS WILL BE THE INTEGRATOR'S ERROR NUMBER.
B)	TMSAVE	THE INSTRUCTION COUNTER AT THE TIME OF THE SIMULATED TRAP IS INCREMENTED ACCORDING TO THE RETURN FROM THE INTEGRATOR AND STORED BACK INTO THE APPROPRIATE PANEL LOCATION OF THE QTSXING ROUTINE.

METHOD

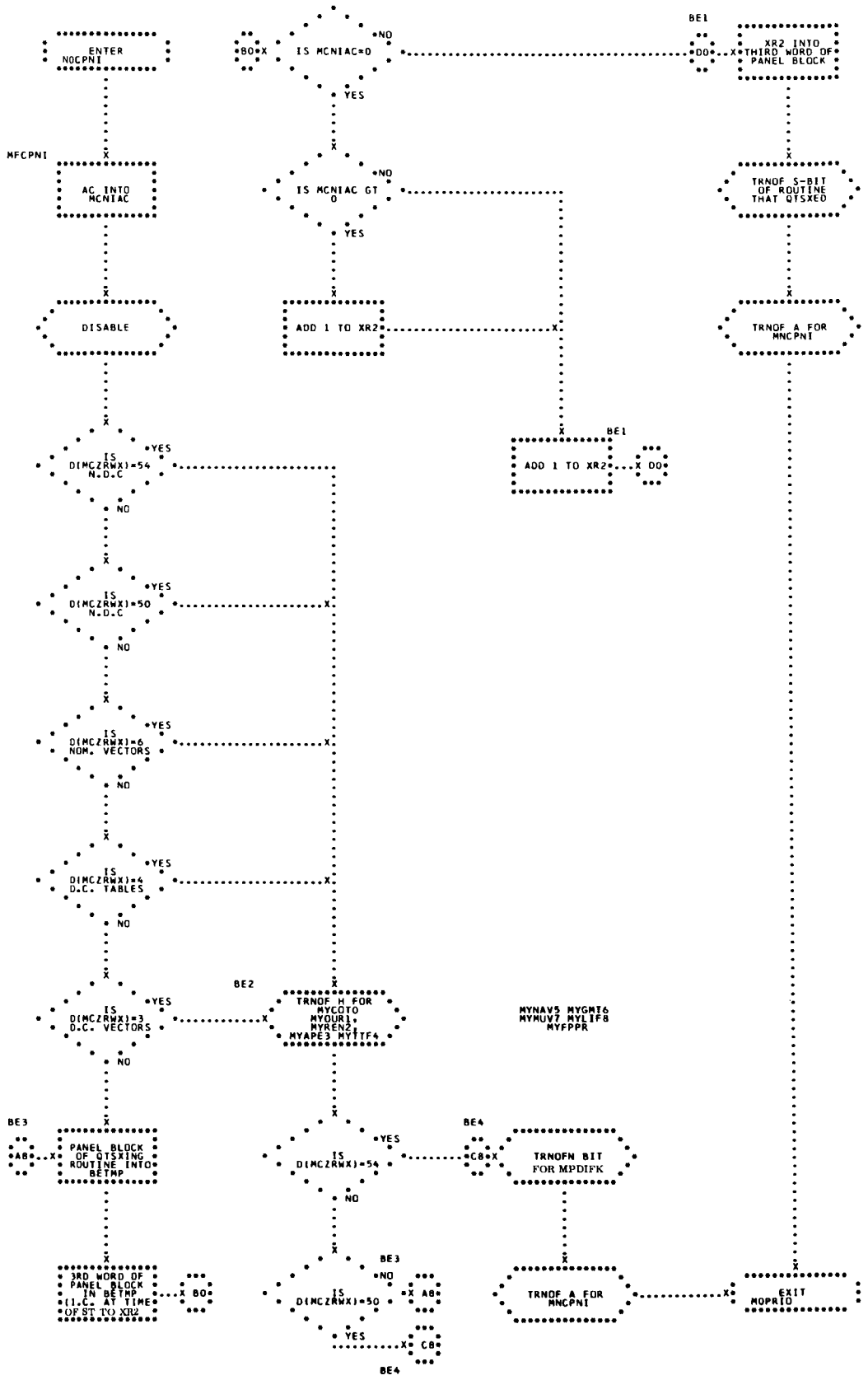
MFCPNI FIRST CHECKS FOR A QUEUE FROM MYDIC9 AND IN THOSE CASES, UNSUPPRESSES OTHER USES OF NUMERICAL INTEGRATION. THE REASON HERE IS THAT THERE IS NO RETURN TO MYDIC9, SINCE IT QUEUES INTEGRATION AND OTHER PROCESSORS QTSX. EVEN HERE THOUGH EXCEPT IN THE CASE OF N.D.C. IN REEENTRY, THE INTEGRATION IS INITIATED BY Q'SX. HENCE, MFCPNI SETS UP THE RETURN FROM NOCPNI TO THE NEXT LOCATION AFTER THE QTSX IF THERE WAS AN ERROR IN NOCPNI, THE SECOND LOATION AFTER THE QTSX IF THE TABLE IMPACTED, AND, THE THIRD LOCATION AFTER THE QTSX IF THE TABLE DOESN'T

IMPACT. THEN THE S BIT IS TURNED OFF FOR THE ROUTINE THAT QTSX'ED.

USAGE

A) STORAGE REQUIRED - 66 LOCATIONS

B) TIMING - .133 MS (MAXIMUM)


ENTER NOCPNI
IS MCNIAC=0
*NO
*YES
BE1
DO
XR2 INTO THIRD WORD OF PANEL BLOCK
MFCPNI
AC INTO MCNIAC
IS MCNIAC GT 0
TRNOF S-BIT OF ROUTINE THAT QTSXEQ
DISABLE
ADD 1 TO XR2
TRNOF A FOR MNCPNI
IS D(MCZRWX)=54 N.D.C
IS D(MCZRWX)=50 N.D.C
IS D(MCZRWX)=6 NOM. VECTORS
IS D(MCZRWX)=4 D.C. TABLES
IS D(MCZRWX)=3 D.C. VECTORS
BE2
TRNOF H FOR MYCOTO MYOUR1, MYREN2, MYAPE3 MYTTF4
MYNAV5 MYGMI6 MYMUV7 MYLIF8 MYFPPR
BE3
PANEL BLOCK OF QTSXING ROUTINE INTO BETMP
BE4
TRNOFN BIT FOR MPDIFK
3RD WORD OF PANEL BLOCK IN BETMP (I.C. AT TIME OF ST TO XR2
EXIT MOPRIO

30. N1SUFT - SET UP FUNCTION TABLE

N1SUFT ESTABLISHES A SECOND DERIVATIVE TABLE WHICH IS USED TO INITIATE THE NUMERICAL INTEGRATION FUNCTION. (SEE FLOWCHART.)

INPUT

OTHER PROGRAMS USED WITH THE N1SUFT PROGRAM ARE N2EXCR, N3SDCL, AND N7VARS. THE FOLLOWING PARAMETERS ARE INPUT REQUIREMENTS FOR THE N1SUFT PROGRAM -

A) K000.X - DESCRIBED IN NOCPNI
B) K000.Y - DESCRIBED IN NOCPNI
C) K000.Z - DESCRIBED IN NOCPNI
D) K00.VX - DESCRIBED IN NOCPNI
E) K00.VY - DESCRIBED IN NOCPNI
F) K00.VZ - DESCRIBED IN NOCPNI
G) K000TA - DESCRIBED IN NOCPNI
H) K0000H - DESCRIBED IN NOCPNI
I) K000.H - CONTAINS THE FLOATING-POINT EQUIVALENT OF THE CONTENTS OF K0000H IN GEMINI UNITS.
J) K00.2H - CONTAINS THE CONTENTS OF K000.H SQUARED

OUTPUT

THE OUTPUT OF N1SUFT IS TNFUNK (DESCRIBED AT THE END OF THE NOCPNI PROGRAM WRITEUP).

METHOD

THE EQUATIONS USED IN THIS PROGRAM, TAKEN FROM THE ''COMPUTATION OF ORBITS'' BY DR. P. HERGET, ARE -

$$A)\quad {}''F_a = R_a + G_a F_a + \sum_{k=1}^{3} G_k \left(F_k + F_{-k} \right)$$

$$B)\quad {}'F_{a-1/2} = (h)\, V_a + H_a F_a + \sum_{k=1}^{3} H_k \left(F_k - F_{-k} \right)$$

$$C)\quad {}''F_{a-1} = F_a - {}'F_{a-1/2}$$

THE OUTPUT TABLE (TNFUNK) IS NOT ASSUMED TO BE CORRECT UNTIL SUCCESSIVE CALCULATIONS OF THE $''F_6^t$ VALUE DIFFER BY LESS THAN 1E-15.

USAGE

A) CALLING SEQUENCE -

```
TSX     N1SUFT,4
        ERROR RETURN
        NORMAL RETURN
```

B) STORAGE REQUIRED - 261 LOCATIONS

C) CONSTANTS USED -

K00001	DEC 1	
K.TCTB	DEC 1E-15-CONVERGENCE CRITERION	
	$(''F^t - ''F^{t-1})^2$: E-15	
K000JB	PZE 30-NUMBER OF TRIES TO MAKE FOR CONVERGENCE BEFORE DECREASING INTERVAL	
K2ZERO	DESCRIBED IN NOCPNI	
TEMNO + 146	DESCRIBED IN NOCPNI	
TEMNO + 160	PZE STORAGE FOR J	

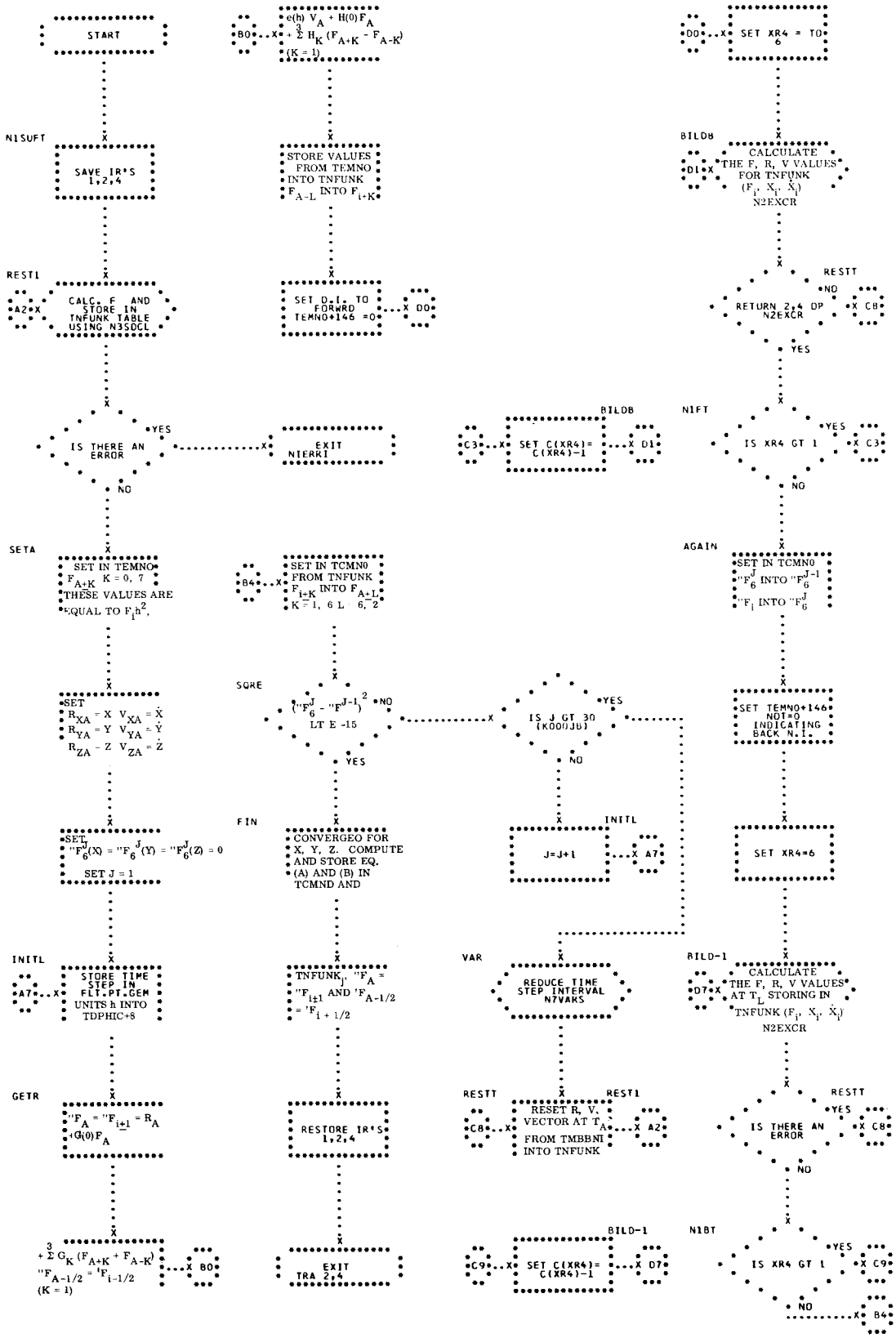
```
TDPGIC      DEC 0.0074115416        G(1)
            PZE
      + 2   DEC -0.0009904103       G(2)        K=1,3
            PZE
      + 4   DEC 0.0000796407        G(3)
            PZE
      + 6   OCT -175612454702       G(0)  -G(0)
TDPHIC      DEC 0.0648396164        H(1)
            PZE
      + 2   DEC 0.0139550265        H(2)        K=1,3
            PZE
      + 4   DEC 0.0015790344        H(3)
            PZE
      + 6   DEC -0.5                H(0) - H(0)
            PZE
      + 8   DEC STORAGE FOR K0000.H
            PZE
```

D) ERASABLE LOCATIONS - TEMNO

E) CHECKOUT - N1SUFT WAS CHECKED OUT BY COMPARING ITS OUTPUT WITH THAT OF IJKNI.

DIFFERENCE AND SUMMATION TABLE OF COWLES NUMERICAL INTEGRATION PROCEDURE

ARGUMENT	2nd SUM	1st SUM	FUNCTION	Δ	Δ^2	Δ^3	Δ^4	Δ^5
t_{i-6h}	$''F_{i-6}$		F_{i-6}					
		$'F_{i-11/2}$		$\Delta_{i-11/2}$				
t_{i-5h}	$''F_{i-5}$		F_{i-5}		Δ^2_{i-5}			
		$'F_{i-9/2}$		$\Delta_{i-9/2}$		$\Delta^3_{i-9/2}$		
t_{i-4h}	$''F_{i-4}$		F_{i-4}		Δ^2_{i-4}		Δ^4_{i-4}	
		$'F_{i-7/2}$		$\Delta_{i-7/2}$		$\Delta^3_{i-7/2}$		$\Delta^5_{i-7/2}$
t_{i-3h}	$''F_{i-3}$		F_{i-3}		Δ^2_{i-3}		Δ^4_{i-3}	
		$'F_{i-5/2}$		$\Delta_{i-5/2}$		$\Delta^3_{i-5/2}$		$\Delta^5_{i-5/2}$
t_{i-2h}	$''F_{i-2}$		F_{i-2}		Δ^2_{i-2}		Δ^4_{i-2}	
		$'F_{i-3/2}$		$\Delta_{i-3/2}$		$\Delta^3_{i-3/2}$		$\Delta^5_{i-3/2}$
t_{i-h}	$''F_{i-1}$		F_{i-1}		Δ^2_{i-1}		Δ^4_{i-1}	
		$'F_{i-1/2}$		$\Delta_{i-1/2}$		$\Delta^3_{i-1/2}$		$\Delta^5_{i-1/2}$
t_i	$''F_i$		F_i		Δ^2_i		Δ^4_i	
		$'F_{i+1/2}$		$\Delta_{i+1/2}$		$\Delta^3_{i+1/2}$		$\Delta^5_{i+1/2}$
t_{i+h}	$''F_{i+1}$		F_{i+1}		Δ^2_{i+1}		Δ^4_{i+1}	
		$'F_{i+3/2}$		$\Delta_{i+3/2}$		$\Delta^3_{i+3/2}$		$\Delta^5_{i+3/2}$
t_{i+2h}	$''F_{i+2}$		F_{i+2}		Δ^2_{i+2}		Δ^4_{i+2}	
		$'F_{i+5/2}$		$\Delta_{i+5/2}$		$\Delta^3_{i+5/2}$		$\Delta^5_{i+5/2}$
t_{i+3h}	$''F_{i+3}$		F_{i+3}		Δ^2_{i+3}		Δ^4_{i+3}	
		$'F_{i+7/2}$		$\Delta_{i+7/2}$		$\Delta^3_{i+7/2}$		$\Delta^5_{i+7/2}$
t_{i+4h}	$''F_{i+4}$		F_{i+4}		Δ^2_{i+4}		Δ^4_{i+4}	
		$'F_{i+9/2}$		$\Delta_{i+9/2}$		$\Delta^3_{i+9/2}$		
t_{i+5h}	$''F_{i+5}$		F_{i+5}		Δ^2_{i+5}			
		$'F_{i+11/2}$		$\Delta_{i+11/2}$				
t_{i+6h}	$''F_{i+6}$		F_{i+6}					


START
N1SUFT
SAVE IR'S 1,2,4
RESTI
CALC. F AND STORE IN TNFUNK TABLE USING N3SDCL
A2
IS THERE AN ERROR
YES
NO
EXIT N1ERR1
SETA
SET IN TEMNO F A+K K = 0, 7 THESE VALUES ARE EQUAL TO F1h2,
SET R XA = X V XA = Ẋ R YA = Y V YA = Ẏ R ZA = Z V ZA = Ż
SET "F6J(X) = "F6J(Y) = "F6J(Z) = 0 SET J = 1
INITL
STORE TIME STEP IN FLT.PT.GEM UNITS h INTO TDPHIC+8
A7
GETR
"FA = "Fi+1 = RA +G(0)FA
3 + Σ GK (FA+K + FA-K) "FA-1/2 = 'Fi-1/2 (K = 1)
B0
c(h) VA + H(0)FA + 3 Σ HK (FA+K - FA-K) (K = 1)
STORE VALUES FROM TEMNO INTO TNFUNK FA-L INTO Fi+K
SET D.I. TO FORWRD TEMN0+146 =0
D0
B4
SET IN TCMN0 FROM TNFUNK Fi+K INTO FA+L K = 1, 6 L = 6, 2
SQRE
("F6J - "F6J-1)2 LT E -15
NO
YES
FIN
CONVERGEO FOR X, Y, Z. COMPUTE AND STORE EQ. (A) AND (B) IN TCMND AND
TNFUNK, "FA = "Fi+1 AND 'FA-1/2 = 'Fi + 1/2
RESTORE IR'S 1,2,4
EXIT TRA 2,4
C3
BILDB
SET C(XR4)= C(XR4)-1
D1
IS J GT 30 (K00)JB)
YES
NO
INITL
J=J+1
A7
VAR
REDUCE TIME STEP INTERVAL N7VARS
RESTT
RESTI
C8
RESET R, V, VECTOR AT TA FROM TMBBNT INTO TNFUNK
A2
BILD-1
C9
SET C(XR4)= C(XR4)-1
D7
D0
SET XR4 = TO 6
BILDB
D1
CALCULATE THE F, R, V VALUES FOR TNFUNK (Fi, Xi, Ẋi) N2EXCR
RESTT
NO
RETURN 2,4 DP N2EXCR
C8
YES
N1FT
IS XR4 GT 1
YES
C3
NO
AGAIN
SET IN TCMN0 "F6J INTO "F6J-1 "Fi INTO "F6J
SET TEMNO+146 NOT=0 INDICATING BACK N.I.
SET XR4=6
BILD-1
D7
CALCULATE THE F, R, V VALUES AT TL STORING IN TNFUNK (Fi, Xi, Ẋi) N2EXCR
RESTT
YES
IS THERE AN ERROR
C8
NO
N1BT
IS XR4 GT 1
YES
C9
NO
B4

31. N2EXCR - EXTRAPOLATION AND CORRECTION EQUATION DERIVATION

N2EXCR CALCULATES F_i , R_i AND V_i FOR A GIVEN TIME T_i. (SEE FLOWCHART.)

INPUT

THE INPUT TO THE N2EXCR PROGRAM COMES FROM THE FOLLOWING LOCATIONS -

A) TNFUNK - DESCRIBED IN NOCPNI.

B) K000.H - CONTAINS THE FLOATING-POINT EQUIVALENT OF THE CONTENTS OF K0000H IN GEMINI UNITS.

C) K00.2H - CONTAINS THE CONTENTS K000.H SQUARED.

D) TEMNO + 51 - IF ZERO, INTEGRATION IS CARRIED FORWARD, IF 1, INTEGRATION IS CARRIED BACKWARD (FIXED-POINT).

OUTPUT

THE UPDATED TNFUNK TABLE IS THE OUTPUT OF N2EXCR.

METHOD

THE EQUATIONS USED IN THIS PROGRAM ARE TAKEN FROM THE COMPUTATION OF ORBITS BY DR. P. HERGET.

A) $'F_{i-1/2} = 'F_{i-3/2} + F_{i-1}$ (FOR FORWARD INTEGRATION)

B) $'F_{i+1/2} = 'F_{i+3/2} - F_{i+1}$ (FOR BACKWARD INTEGRATION)

C) $''F_i = ''F_{i-1} + 'F_{i-1/2}$ (FOR FORWARD INTEGRATION)

D) $''F_i = ''F_{i+1} - 'F_{i+1/2}$ (FOR BACKWARD INTEGRATION)

E) $R_i = {}''F_i + \sum_{k=1}^{6} A_k F_{i-k}$

F) $V_i = \frac{1}{h}\left[{}'F_{i-1/2} + \sum_{k=1}^{6} A'_k F_{i-k}\right]$

(FORWARD EXTRAPOLATION FOR R_i, V_i)

G) $R_i = {}''F_i + \sum_{k=1}^{6} A_k F_{i+k}$

H) $V_i = \frac{1}{h}\left[{}'F_{i+1/2} - \sum_{k=1}^{6} {}'A_k F_{i+k}\right]$

(BACKWARD EXTRAPOLATION FOR R_i, V_i)

I) $R_i = {}''F_i + \sum_{k=0}^{6} B_k F_{i-k}$

J) $V_i = \frac{1}{h}\left[{}'F_{i-1/2} + \sum_{k=0}^{6} {}'B_k F_{i-k}\right]$

(FORWARD CORRECTION FOR R_i, V_i)

K) $R_i = {}''F_i + \sum_{k=0}^{6} B_k F_{i+k}$

L) $V_i = \frac{1}{h}\left[{}'F_{i+1/2} - \sum_{k=0}^{6} {}'B_k F_{i+k}\right]$

(BACKWARD CORRECTION FOR R_i, V_i)

M) $\left|{}''F_6^J - {}''F_6^{J-1}\right|$: 1E - 15 USED TO CHECK CONVERGENCE

THE F TABLE VALUES CALCULATED ARE NOT ASSUMED TO BE CORRECT UNTIL SUCCESSIVE CALCULATION OF THE F_i VALUE DIFFER BY LESS THAN 1E-15.

USAGE

A) CALLING SEQUENCE -

```
TSX     N2EXCR,4
        REDUCE TIME STEP
        NORMAL RETURN
```

B) STORAGE REQUIRED - 234 LOCATIONS.

C) CONSTANTS -

K00001	DEC 1	
KO.CTB	DEC 1E-15	- CONVERGENCE CRITERION FOR F_i WHEN GENERATING A MINUTES OUTPUT TABLE.
KO.CTC	DEC 1E-15	- CONVERGENCE CRITERION FOR F_i WHEN GENERATING A SECONDS OUTPUT TABLE.
K000TU		IF A 1, SECONDS ARE REPRESENTED, IF ZERO, MINUTES ARE REPRESENTED
KO.FDT	DEC 0.25E-4	- MAGNITUDE TEST FACTOR FOR FIFTH CORRECTION
DCSUMM	PZE	STORAGE FOR FIFTH DIFFERENTIAL CORRECTION SUM
TEMN0+110	PZE	STORAGE FOR N
TEMN0+146	PZE	=0, INTEGRATION FORWARD $\neq$ 0, INTEGRATION BACKWARD
TEMN0+138	PZE	STORAGE FOR K0002H
TN7ICT	PZE	NUMBER OF TIMES INTERVAL HAD TO BE REDUCED
MCZRWX		MONITOR CONTROL WORD WHOSE DECREMENT INDICATES TYPE OF TABLE BEING GENERATED

```
THE FOLLOWING ARE IN DOUBLE PRECISION -
TDPAIC          DEC 0.46038333      A(1)
     +1         PZE
     +2         DEC -1.09306667     A(2)
     +3         PZE
     +4         DEC 1.4142667       A(3)
     +5         PZE
     +6         DEC -1.0424         A(4)
                PZE
     +8         DEC .41235          A(5)
                PZE
     +10)       DEC -0.0682         A(6)
                PZE
TDPA1P          DEC 2.28573082      AP(1)
                PZE
     +2         DEC -5.1099041      AP(2)
                PZE
     +4         DEC 6.55591931      AP(3)
                PZE
     +6         DEC -4,82397487     AP(4)
                PZE
     +8         DEC 1.90782077      AP(5)
                PZE
     +10        DEC -.31559193      AP(6)
                PZE
```

A_k, k = 1, 6

$'A_k$, k = 1, 6

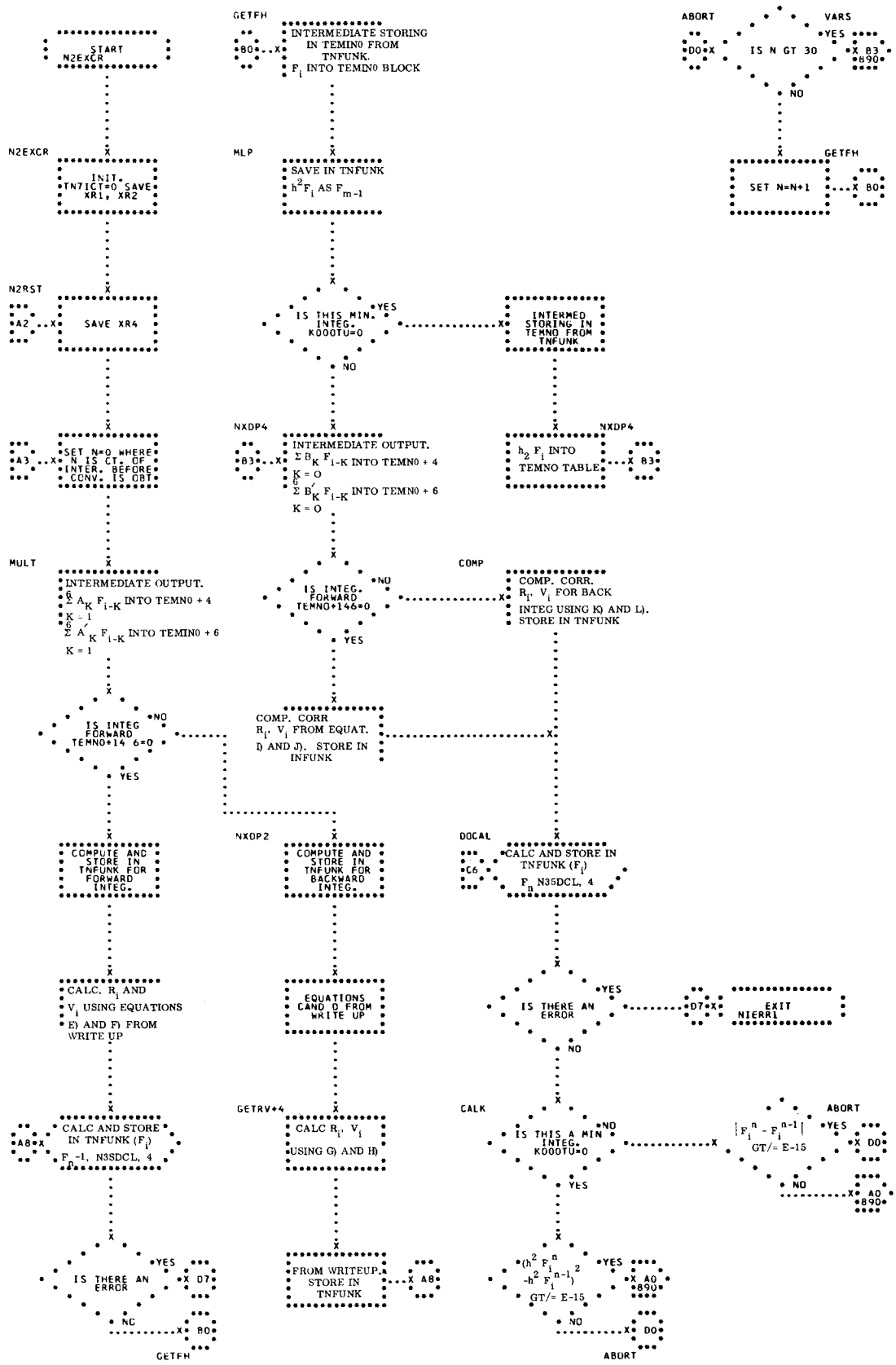
```
TDPBIC          DEC 0.065495756    B(0)
                PZE
    +2          DEC 0.067409063    B(1)
                PZE
    +4          DEC -0.110635752   B(2)
                PZE
    +6          DEC 0.104370596    B(3)
                PZE
    +8          DEC -0.05999091    B(4)
                PZE
    +10         DEC 0.019393189    B(5)
                PZE
    +12         DEC -0.002708609   B(6)
                PZE
TDPBIP          DEC 0.30422454     BP(0)
                PZE
    +2          DEC 0.46038360     BP(1)
                PZE
    +4          DEC -0.54653605    BP(2)
                PZE
    +6          DEC 0.47142857     BP(3)
                PZE
    +8          DEC -0.26060681    BP(4)
                PZE
    +10         DEC 0.8747355      BP(5)
                PZE
    +12         DEC -0.01136740    BP(6)
                PZE
```

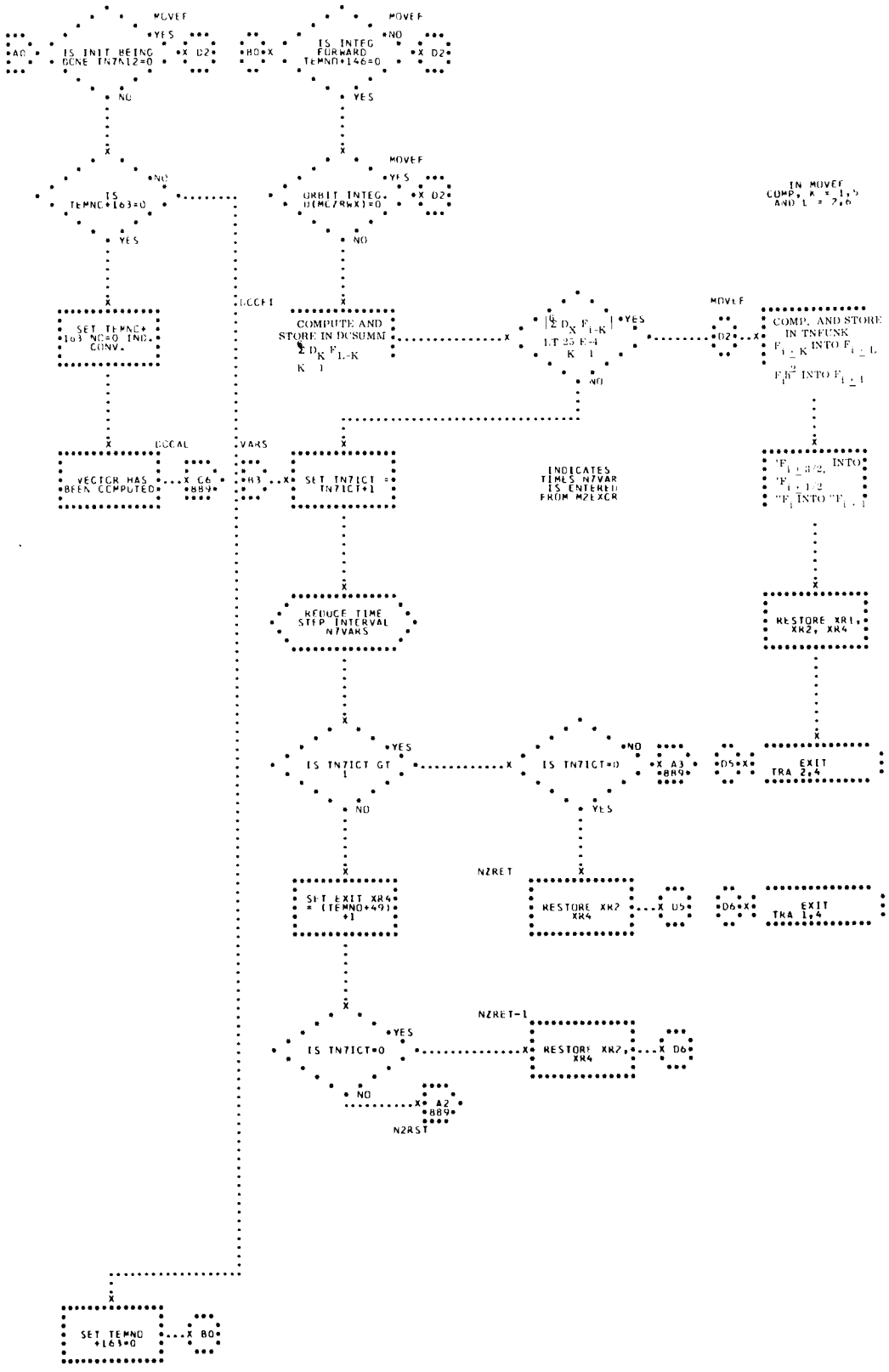
B_k, $k = 0, 6$

$'B_k$, $k = 0, 6$

D) ERASABLE LOCATIONS - TEMNO TABLE

E) PROGRAMS USES - N3SDCL AND N7VARS


START N2EXCR
N2EXCR
INIT. TN7ICT=0 SAVE XR1, XR2
N2RST
SAVE XR4
SET N=0 WHERE N IS CT. OF INTER. BEFORE CONV. IS OBT
MULT
INTERMEDIATE OUTPUT.
$\sum_{K=1}^{6} A_K F_{i-K}$ INTO TEMN0 + 4
$\sum_{K=1}^{6} A'_K F_{i-K}$ INTO TEMIN0 + 6
IS INTEG FORWARD TEMNO+14 6=0
NO
YES
COMPUTE AND STORE IN TNFUNK FOR FORWARD INTEG.
CALC. R_i AND V_i USING EQUATIONS E) AND F) FROM WRITE UP
CALC AND STORE IN TNFUNK (F_i)
F_n, N3SDCL, 4
IS THERE AN ERROR
YES
NO
D7
B0
GETFH
GETFH
INTERMEDIATE STORING IN TEMIN0 FROM TNFUNK. F_i INTO TEMIN0 BLOCK
B0
MLP
SAVE IN TNFUNK
$h^2 F_i$ AS F_{m-1}
IS THIS MIN. INTEG. K000TU=0
YES
NO
INTERMED STORING IN TEMNO FROM TNFUNK
NXDP4
NXDP4
INTERMEDIATE OUTPUT.
$\sum_{K=0}^{6} B_K F_{i-K}$ INTO TEMN0 + 4
$\sum_{K=0}^{6} B'_K F_{i-K}$ INTO TEMN0 + 6
B3
IS INTEG. FORWARD TEMNO+146=0
NO
YES
COMP. CORR R_i, V_i FROM EQUAT. I) AND J). STORE IN INFUNK
NXDP2
COMPUTE AND STORE IN TNFUNK FOR BACKWARD INTEG.
EQUATIONS C AND D FROM WRITE UP
GETRV+4
CALC R_i, V_i USING G) AND H)
FROM WRITEUP. STORE IN TNFUNK
A8
COMP
COMP. CORR. R_i, V_i FOR BACK INTEG USING K) AND L). STORE IN TNFUNK
DOCAL
CALC AND STORE IN TNFUNK (F_i)
F_n N35DCL, 4
C6
IS THERE AN ERROR
YES
NO
D7
EXIT NIERR1
CALK
IS THIS A MIN INTEG. K000TU=0
NO
YES
$|F_i^n - F_i^{n-1}|$ GT/= E-15
YES
NO
ABORT
A0
$(h^2 F_i^n - h^2 F_i^{n-1})^2$ GT/= E-15
YES
NO
ABORT
D0
ABORT
VARS
IS N GT 30
YES
NO
GETFH
SET N=N+1
B0
B3
B90


MOVEF
IS INIT BEING DONE TN7N12=0
YES
NO
A0
X D2
MOVEF
IS INTEG FORWARD TEMNO+146=0
NO
YES
H0
X D2
IS TEMNC*163=0
NO
YES
MOVEF
ORBIT INTEG. U(MC/RWX)=0
YES
NO
X D2
IN MOVEF COMP. K = 1,5 AND L = 2,6
SET TEMNC+163 NC=0 IND. CONV.
DCOEFI
COMPUTE AND STORE IN DCSUMM
$\sum_{K=1}^{6} D_K F_{L-K}$
$\left|\sum_{K=1}^{6} D_K F_{L-K}\right|$ LT 25 E-4
YES
NO
MOVEF
X D2
COMP. AND STORE IN TNFUNK
F_{L-K} INTO F_{L-L}
F_L^2 INTO F_{L-1}
VECTOR HAS BEEN COMPUTED
DCCAL
X G6 *889*
VARS
H3
SET TN7ICT = TN7ICT+1
INDICATES TIMES N7VAR IS ENTERED FROM M2EXCR
"$F_{L-3/2}$ INTO "$F_{L-1/2}$
"F_L INTO "F_{L-1}
REDUCE TIME STEP INTERVAL N7VARS
RESTORE XR1, XR2, XR4
IS TN7ICT GT 1
YES
NO
IS TN7ICT=0
YES
NO
X A3 *889*
D5
X
EXIT TRA 2,4
NZRET
SET EXIT XR4 = (TEMNO+49) +1
RESTORE XR2 XR4
X D5
D6
X
EXIT TRA 1,4
NZRET-1
IS TN7ICT=0
YES
NO
RESTORE XR2, XR4
X D6
X A2 *889*
N2RST
SET TEMNO +163=0
X B0

32. N3SDCL - SECOND DERIVATIVE CALCULATION CONTROL

N3SDCL CALCULATES F_i BY ADDING THE OUTPUT OF SUBROUTINE N60PEP AND N4DRAG. (SEE FLOWCHART.)

INPUT

THE N3SDCL PROGRAM USES PROGRAMS - N60PEP, N4DRAG, N3TRB, TADAPT, ROCKNR, AND N7PCAL.

A) THE X, Y, Z, VX, VY, AND VZ VALUES FROM THE FOLLOWING TNFUNK TABLE LOCATIONS -

TNFUNK + 22 = X
TNFUNK + 48 = Y
TNFUNK + 74 = Z
TNFUNK + 24 = VX
TNFUNK + 50 = VY
TNFUNK + 76 = VZ

B) K000TI THE TIME FOR WHICH R AND V VALUES ARE CURRENTLY BEING CALCULATED IN NOCPNI, THE PRESENT INTEGRATION TIME, EXPRESSED IN MINUTES OR SECONDS, DEPENDING ON K0000H.

C) K000TU IF A ONE, K0000H REPRESENTS SECONDS, IF ZERO, K0000H REPRESENTS MINUTES (BOTH IN FIXED-POINT).

D) KTORNT IF ZERO, TADAPT AND ROCKNR ARE NOT USED. IF NON-ZERO, TADAPT AND ROCKNR ARE USED.

E) K7CB IF ZERO, N7PCAL IS NOT USED
IF NON-ZERO, N7PCAL IS USED

OUTPUT

THE OUTPUT OF N3SDCL IS F_i AND PLACED IN TNFUNK.

TNFUNK + 0 = $\ddot{X}$

TNFUNK + 26 = $\ddot{Y}$

TNFUNK + 52 = $\ddot{Z}$

METHOD

THE EQUATIONS USED IN N3SDCL ARE TAKEN FROM THE COMPUTATION OF ORBITS' BY DR. P. HERGET.

$$X = \frac{\partial P}{X} + D_x + L_x + T_x$$

$$Y = \frac{\partial P}{Y} + D_y + L_y + T_y$$

$$Z = \frac{\partial P}{Z} + D_z + L_z + T_z$$

N3SDCL IS ACCURATE TO SEVEN SIGNIFICANT DIGITS.

USAGE

A) CALLING SEQUENCE -

```
ALPHA     TSX N3SDCL,4
ALPHA + 1    ERROR RETURN
ALPHA + 2    NORMAL RETURN
```

B) STORAGE REQUIRED - 39 LOCATIONS

C) CONSTANTS -

```
KDSUBX   SYN TDSUBX   D_x
KDSUBY   SYN TDSUBY   D_y    OUTPUT OF N4URAG
KDSUBZ   SYN TDSUBZ   D_z

KOO.XP   SYN TNINT    ∂P/x

KOO.YP   SYN TNIINT+1 ∂P/y   OUTPUT OF N60PEP
```

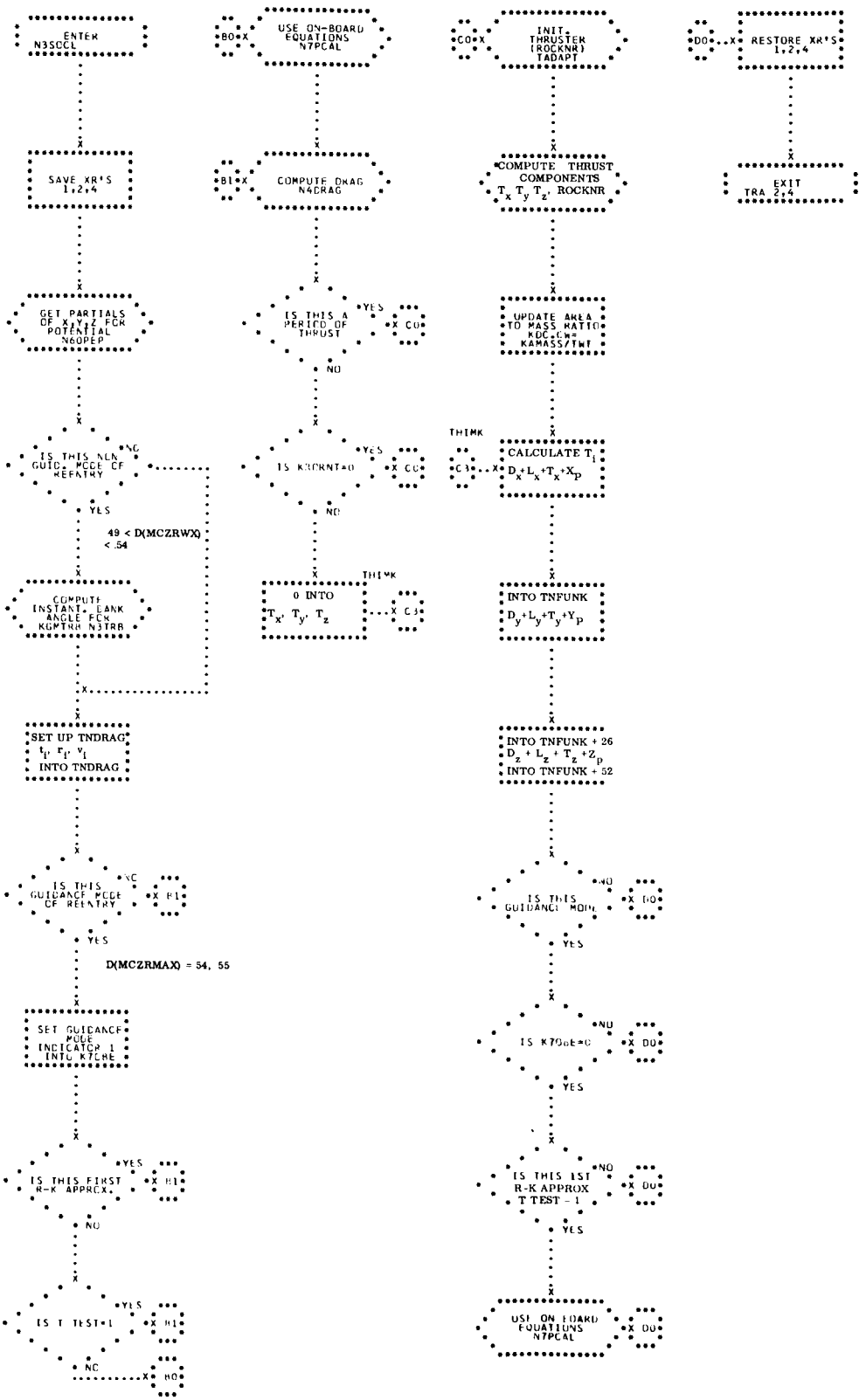
KCO.ZP SYN TNIINT+2 $\frac{\partial P}{z}$

TNDRAG BSS 7 INPUT TO N4DRAG

D) TIME REQUIRED - 539+D+P+N3TRB+N7PCAL CYCLES

WHERE D IS TIMING FOR N4DRAG

P IS TIMING FOR N60PEP


ENTER N3SCCL
SAVE XR'S 1,2,4
GET PARTIALS OF X,Y,Z FOR POTENTIAL N60PEP
IS THIS NLN GUID. MODE OF REENTRY
*NO
YES
49 < D(MCZRWX) < 54
COMPUTE INSTANT. BANK ANGLE FOR KGMTRR N3TRB
SET UP TNDRAG t_f, r_f, v_f INTO TNDRAG
IS THIS GUIDANCE MODE OF REENTRY
*NO
*X B1
YES
D(MCZRMAX) = 54, 55
SET GUIDANCE MODE INDICATOR 1 INTO K7GNE
IS THIS FIRST R-K APPROX.
*YES
*X B1
NO
IS T TEST=1
*YES
*X B1
NO
*X B0
USE ON-BOARD EQUATIONS N7PCAL
*X B1
COMPUTE DRAG N4DRAG
IS THIS A PERIOD OF THRUST
*YES
*X C0
NO
IS K7GRNT=0
*YES
*X C0
NO
THIMK
0 INTO T_x, T_y, T_z
*X C3
*X C0
INIT. THRUSTER (ROCKNR) TADAPT
COMPUTE THRUST COMPONENTS T_x T_y T_z, ROCKNR
UPDATE AREA TO MASS RATIO KDC.CM= KAMASS/TWT
THIMK
CALCULATE T_i D_x+L_x+T_x+X_p
*X C3
INTO TNFUNK D_y+L_y+T_y+Y_p
INTO TNFUNK + 26 D_z + L_z + T_z + Z_p
INTO TNFUNK + 52
IS THIS GUIDANCE MODE
*NO
*X D0
YES
IS K7GNE=0
*NO
*X D0
YES
IS THIS 1ST R-K APPROX T TEST - 1
*NO
*X D0
YES
USE ON BOARD EQUATIONS N7PCAL
*X D0
*X D0
RESTORE XR'S 1,2,4
EXIT TRA 2,4

33. N3TRB - COMPUTATION OF INSTANTANEOUS BANK ANGLE FOR TIME TO REVERSE BANK

INPUT

A)	TFTSWT	INDICATOR TO SET TO NOTE NOT THE FIRST TIME IN PROGRAM.
B)	RBNOW	=0, NOT TIME TO REVERSE ANGLE ≠0, IT IS TIME TO REVERSE ANGLE
C)	TTEST	DENOTES WHICH RUNGE-KUTTA APPROXIMATION
D)	KRATE	ROLL RATE
E)	KSEC.0	H IN FLOATING POINT SECONDS
F)	KSEC.2	H/2 IN FLOATING POINT SECONDS
G)	FBANK	FINAL BANK ANGLE DESIRED
H)	KBANK	BANK ANGLE AT END OF INTEGRATION STEP.

OUTPUT

A)	BANK	INTERMEDIATE BANK ANGLE FOR EACH APPROXIMATION.
B)	KBANK	BANK ANGLE AT END OF INTEGRATION STEP.
C)	TNIRRB	DENOTES FINAL BANK ANGLE HAS BEEN REACHED WHEN NON-ZERO

METHOD

$$B_2 = B_1 + \dot{B} \times h$$

$$B_2 = B_1 + \dot{B} \times h/2$$

USAGE

A) CALLING SEQUENCE -

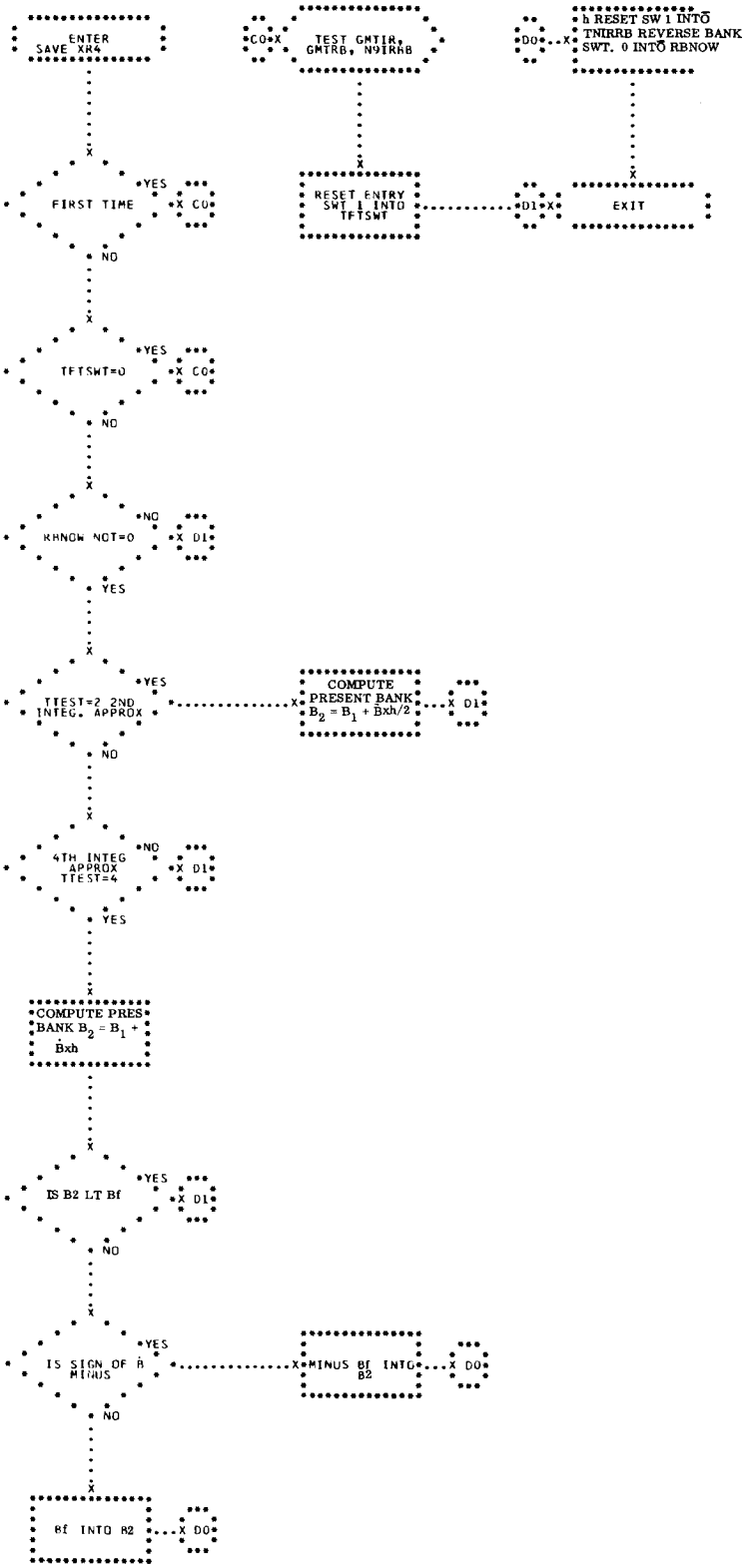
```
TSX     N3TRB,4
+1      NORMAL RETURN
```

B) CONSTANTS -

K00002 DEC 2

C) PROGRAMS - N9IRRB

D) STORAGE REQUIRED - 37 LOCATIONS


ENTER
SAVE XR4
TEST GMTIR, GMTRB, N91RRB
CO
DO
h RESET SW 1 INTŌ TNIRRB REVERSE BANK SWT. 0 INTŌ RBNOW
FIRST TIME
YES
NO
C0
RESET ENTRY SWT 1 INTO TFTSWT
D1
EXIT
TFTSWT=0
RRNOW NOT=0
TTEST=2 2ND INTEG. APPROX
COMPUTE PRESENT BANK $B_2 = B_1 + Bxh/2$
4TH INTEG APPROX TTEST=4
COMPUTE PRES BANK $B_2 = B_1 + \dot{B}xh$
IS B2 LT Bf
IS SIGN OF $\dot{h}$ MINUS
MINUS Bf INTO B2
Bf INTO B2

34. N4DRAG - CAPSULE DRAG FORMULATION

DURING THE MISSION, N4DRAG PRODUCES THE COMPONENTS OF DRAG ACCELERATION, THE TIMES OF BEGINNING AND ENDING OF BLACKOUT, CAPSULE IMPACT DATA, THE VECTORS SPANNING 400K FEET, AND THOSE SPANNING 350K FEET. (SEE FLOWCHART.)

INPUT

A)	TNDRAG	TIME, t_0 IN MINUTES OR SECONDS
B)	TNDRAG + 1 X TNDRAG + 2 Y TNDRAG + 3 Z TNDRAG + 4 VX TNDRAG + 5 VY TNDRAG + 6 VZ	$\bar{r}$ AND $\bar{v}$ VECTORS CORRESPONDING TO TIME, T_0 IN FLOATING POINT.
C)	TN7N12	INDICATOR SET TO DIFFERENTIATE INITIALIZATION FROM ACTUAL INTEGRATION - ZERO INDICATES INITIALIZATION, NONZERO INTEGRATION.
D)	TN4NEX	INDICATOR SET BY R2AINT (A SUBROUTINE OF DIFFERENTIAL CORRECTION) (NOTE - NORMAL STATUS +0). IF MINUS, THE N4DRAG PROGRAM WILL NOT STORE IMPACT DATA. ONE OF THE SIX TABLES NEEDED FOR THE NUMERICAL DIFFERENTIAL CORRECTION IS BEING GENERATED. IF MINUS ZERO, THE PROGRAM WILL NOT EXIT AT 10,000 FEET. THUS THE NUMERICAL DIFFERENTIAL COR-

RECTION TABLES (AFTER THE FIRST ONE) WILL ALL HAVE THE SAME ENDING.

E) TNCORK — ORBIT COUNTER DURING GENERATION OF CAPE VECTORS. NEEDED TO SET UP CAPE VECTOR COUNT IF 10,000 FEET IS REACHED DURING THIS INTEGRATION.

F) TEMNO + 146 — COMMON LOCATION -
+1 INDICATES BACKWARD INTEGRATION
+0 INDICATES FORWARD INTEGRATION

G) TEMNO+163 — COMMON LOCATION -
≠0 INDICATES A NEW VECTOR
=0 INDICATES A RUNGE-KUTTA APPROXIMATE

H) TN4SW1 — SWITCH, USED FOR STORING 2 VECTORS SPANNING 400,000 FEET
=0, THIS IS FIRST POINT
=1, LESS THAN 400,000 FEET
=2, GREATER THAN 400,000 FEET

I) TN4SW2 — SWITCH USED FOR TB1 (BEGINNING OF BLACKOUT) -
=0, THIS IS FIRST POINT
=1, LESS THAN 308,000 FEET
=2, GREATER THAN 308,000 FEET

J) TN4SW3 — SWITCH USED FOR TB2 (ENDING OF BLACKOUT) -
=0, THIS IS FIRST POINT
=1, LESS THAN 133,768 FEET
=2, GREATER THAN 133,768 FEET

K) TN4SW5 — SWITCH USED FOR STORING 2 VECTORS SPANNING 350,000 FEET.

		=0, THIS IS FIRST POINT =1, LESS THAN 350,000 FEET =2, GREATER THAN 350,000 FEET
L)	TN4SW6	SWITCH USED FOR TIME AT 325,000 FEET. =0, THIS IS FIRST POINT =1, LESS THAN 325,000 FEET =2, GREATER THAN 325,000 FEET
M)	KFLOFX	=0, FIXED POINT INTEGRATION IS IN PROCESS. ≠0, FLOATING POINT INTEGRATION IS IN PROCESS.
N)	KOVECT	=0, ITERATIONS FOR TTF ≠0, CONVERGED TRAJECTORY, DO ALL ALTITUDE TESTS.
O)	BANK	PRESENT BANK ANGLE, USED FOR LIFT COMPONENTS COMPUTATIONS.
P)	TVSQOA	COMPONENTS ACCELERATION.
Q)	KDENSF	DENSITY FUNCTION USED IN TESTS FOR LOG (V^2/A).

OUTPUT

	TDSUBX TDSUBY TDSUBZ	X,Y, AND Z COMPONENTS OF THE DRAG ACCELERATION IN FLOATING-POINT GEMINI UNITS.
	RETROFIRE IMPACT DATA IS STORED IN TMIMPP -	
A)	TMIMPP	FIRST TIME, t_1, $\leq$ 10,000 FEET, MIN/ ADDRESS, SEC/DECREMENT.
	TMIMPP + 1	Φ AT TIME t_1, IN FLOATING POINT.
	TMIMPP + 2	λ AT TIME t_1, IN FLOATING POINT.

TMIMPP + 3 LAST TIME, t_2, $\leq$ 450,000 FEET, MIN/ADDRESS, SEC/DECREMENT.

TMIMPP + 4 X

TMIMPP + 5 Y

TMIMPP + 6 Z $\bar{r}$ AND $\bar{v}$ PARAMETERS AT TIME t_1 IN FLOATING POINT

TMIMPP + 7 VX

TMIMPP + 8 VY

TMIMPP + 9 VZ

B) TLSUBX

C) TLSUBY X,Y,Z COMPONENTS OF LIFT ACCELERATION IN FLOATING GEMINI UNITS.

D) TLSUBZ

E) TRBTRV SIX VECTORS IN REVERSE FOR TTF.

F) TVVSQA SIX V^2/A FOR GMTRB.

G) TNREEP + 2 TIME, BEGINNING OF BLACKOUT.

H) TNREEP + 3 TIME, ENDING OF BLACKOUT.

I) TNREEP + 6 TIME, AT 325,000 FEET.

THE NORMAL STATE OF TODRAG IS ZERO. THE RETROFIRE PROGRAM SETS THIS INDICATOR TO A NONZERO VALUE ONLY DURING RETROFIRE CALCULATIONS - IT RESETS THE INDICATOR TO ZERO ON COMPLETION OF THESE CALCULATIONS.

METHOD

THE FOLLOWING EQUATIONS ARE USED BY THE N4DRAG PROGRAM -

$$V = (V_X + Y\omega_Z)^2 + (V_Y - X\omega_Z)^2 + V_Z^2$$

h (GEOP. METERS) = 6378145.0 $h_{\text{Gemini units}} / 1 + h_{\text{Gemini units}}$
FORMULATED FROM ICAO STANDARD ATMOSPHERE.

$$D_X = -1/2\rho C_d V \left(V_X + Y\omega_Z\right)$$

$$D_Y = -1/2\rho C_d V \left(V_Y - X\omega_Z\right)$$

$$D_Z = 1/2\rho C_d V \left(V_Z\right) - Y\omega_Z$$

$\omega_Z = 0.058833543$, ROTATIONAL VELOCITY OF THE EARTH

$$T_{B1} = T_L + \left(T - T_L\right) \left[\frac{308K - H_L}{H - H_L}\right]$$

$$T_{325K} = T_L + \left(T - T_L\right) \left[\frac{325K - H_L}{H - H_L}\right]$$

$$T_{B2} = T_L + \left(T - T_L\right) \left[\frac{127K - H_L}{H - H_L}\right]$$

$$\gamma = \sin^{-1} \left[\frac{\overline{r} \cdot \overline{v}}{rv}\right]$$

USAGE

A) CALLING SEQUENCE -

α	TSX	N4DRAG,4
α+ 1		ERROR RETURN
α+ 2		NORMAL RETURN

B) SPACE REQUIRED 294 LOCATIONS, EXCLUDING CONSTANTS AND TABLES, ONE OF WHICH IS TEMPORARY STORAGE LOCATION.

C) SPECIAL USAGE EXIT TO MFCPNI WILL NOT BE MADE FOR A 10,000 FEET INDICATION UNDER THE FOLLOWING CONDITIONS -

1) INITIAL FUNCTION TABLE SET UP BY N1SUFT IS PROCESS (TN7N12 = 0)
2) BACKWARD INTEGRATION IS IN PROCESS (TEMNO + 146 $\neq$ 0)
3) AFTER FIRST OF SIX NUMERICAL DIFFERENTIAL CORRECTION TABLES HAS GIVEN A TIME OF A 10,000 FEET EXIT, TN4NEX IS SET TO ZERO TO USE THIS TIME AS AN EXIT FOR THE REMAINING TABLES (NOT A HEIGHT INDICATION)

D) CONSTANTS -

KM0000	DEC	-0	
K00001	DEC	1	
K00002	DEC	2	
K00003	DEC	3	
K00060	DEC	60	
K000.5	DEC	.5	
K001.0	DEC	1.0	
K040.4	DEC	.01911521591	400,000 FEET IN GEMINI UNITS
K0.1E4	DEC	.000477880397	10,000 FEET IN GEMINI UNITS

	K325.3	DEC	.0155311129 325,000 FEET IN GEMINI UNITS
	KLOG10	DEC	.43429448 CONVERTS $LOG_e(N)$ TO $LOG_{10}(N)$
	K.IPOC	DEC	.020309917 IP FOR ORBIT CAPABILITY (425,000 FEET)
	KO.RAD	DEC	6378165 EQUATORIAL RADIUS OF EARTH (METERS)
	K.BLK1	DEC	.01471876232 BEGINNING OF BLACKOUT (308,000 FEET)
	K.BLK2	DEC	.0063925105 END OF BLACKOUT (133,768 FEET)
	K.1RAD	DEC	57.29577951
	K.OMEG	DEC	.058833780 ROTATIONAL VELOCITY OF THE EARTH (GEMINI UNITS)
	KCFTSC	DEC	25936.272 CONVERSION FACTOR TO FEET/SECOND
	KCH233	OCT	233000000000

E) TABLES -

TOOIND	TIME AT 450,000 FEET, MIN/ADDRESS SEC/DECREMENT
TOBLKT	TIME IN SECONDS TO SAVE AT BLACKOUT ALTITUDES
TODCHC	TIME AT 450,000 FEET TO BE USED BY DIFFERENTIAL CORRECTION
TCSUBD	COEFFICIENT OF DRAG
TCSUBL	COEFFICIENT OF LIFT
TN4LLH	THREE-WORD OUTPUT BLOCK FOR A3MSCP, CONTAINING LATITUDE (RADIANS), LONGITUDE (RADIANS), AND HEIGHT (GEMINI UNITS)

TNETIM	ENDING TIME OF INTEGRATION IF 10,000 FEET IS REACHED
TVELOC	VELOCITY MAGNITUDE (INCLUDING ROTATION OF EARTH)
TMBBNI	NUMERICAL INTEGRATION INPUT TABLE AND ASSOCIATED EQUATED K CONSTANTS
TVSQOA	COMPONENTS OF ACCELERATION

FOR OTHER TABLES USED, SEE SUB-SECTIONS TITLE INPUT AND OUTPUT.

F) COMMUNICATION CELLS -

MCNRRF	ORBIT OR REENTRY INDICATOR =0, ORBIT ≠0, REENTRY
MCZRWX	MONITOR CONTROL WORD WHOSE DECREMENT INDICATES TYPE OF TABLE BEING GENERATED.
DECREMENT	=0, ORBIT DISPLAY AND ACQUISITION TABLES =1, CAPE VECTORS =2, TIME TO FIRE ORBIT TABLE, ORBIT NAVIGATION VECTORS. =3, DC VECTORS =4, DC INTEGRATION TABLES =5, ORBIT CAPABILITY =6, NOMINAL VECTORS =7, INTEGRATION THROUGH RETROFIRE =8, MANUEVER REFLECTING VECTORS =9, APOGEE, PERIGEE COMPUTATIONS =10, INTEGRATION OR DESIRED LONGITUDE

=11, MANUEVER COMPUTAIONS (PHASE ANGLE)

=12, INTEGRATION THROUGH OAMS BURN

=50, NDC TABLES BASED ON GUIDANCE PARAMETERS

=51, REENTRY TABLE BASED ON GUIDANCE PARAMETERS

=52, IMPACT POINT

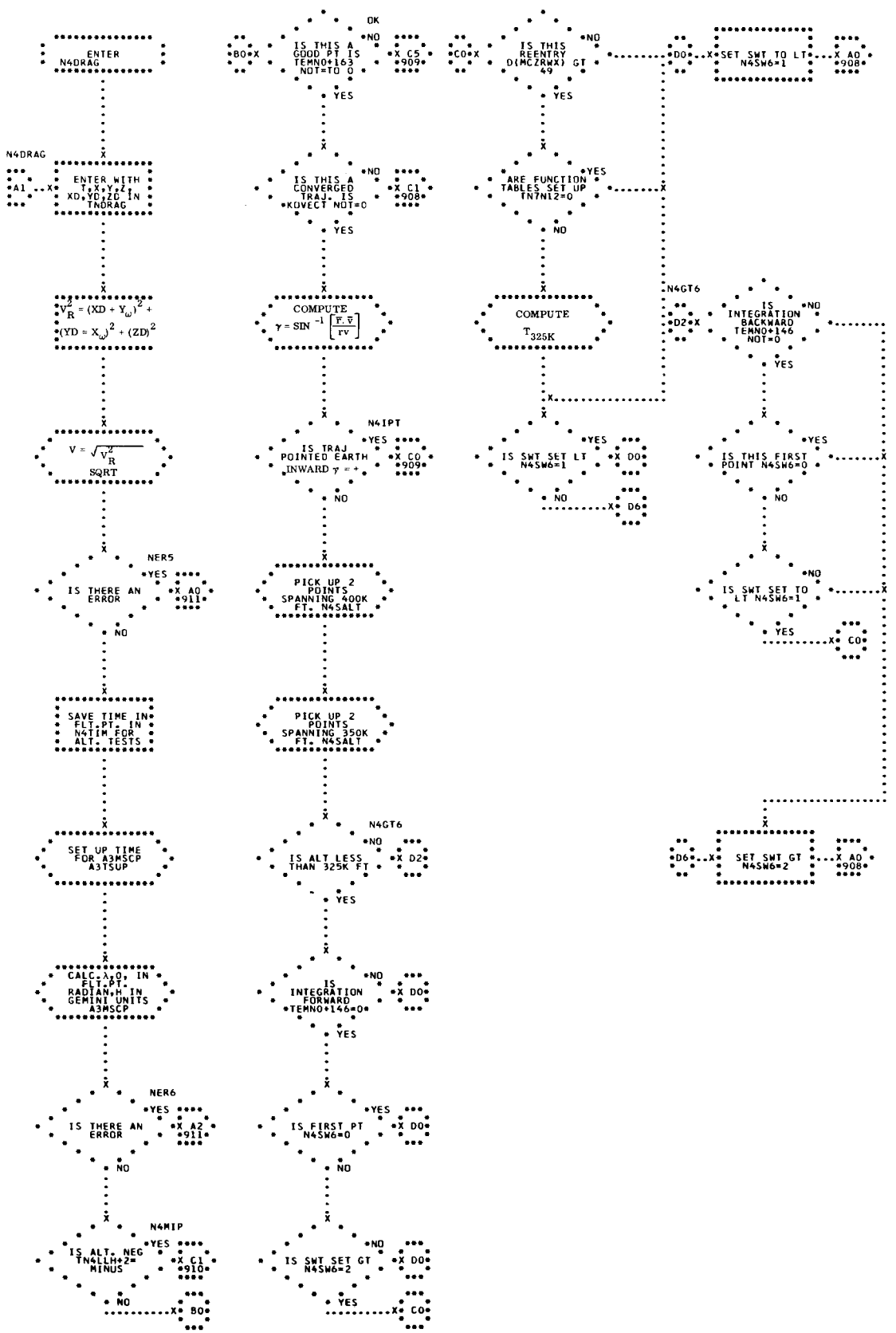
=53, TERMINATES INTEGRATION IF LOG (V 2/A) $\leq$ D

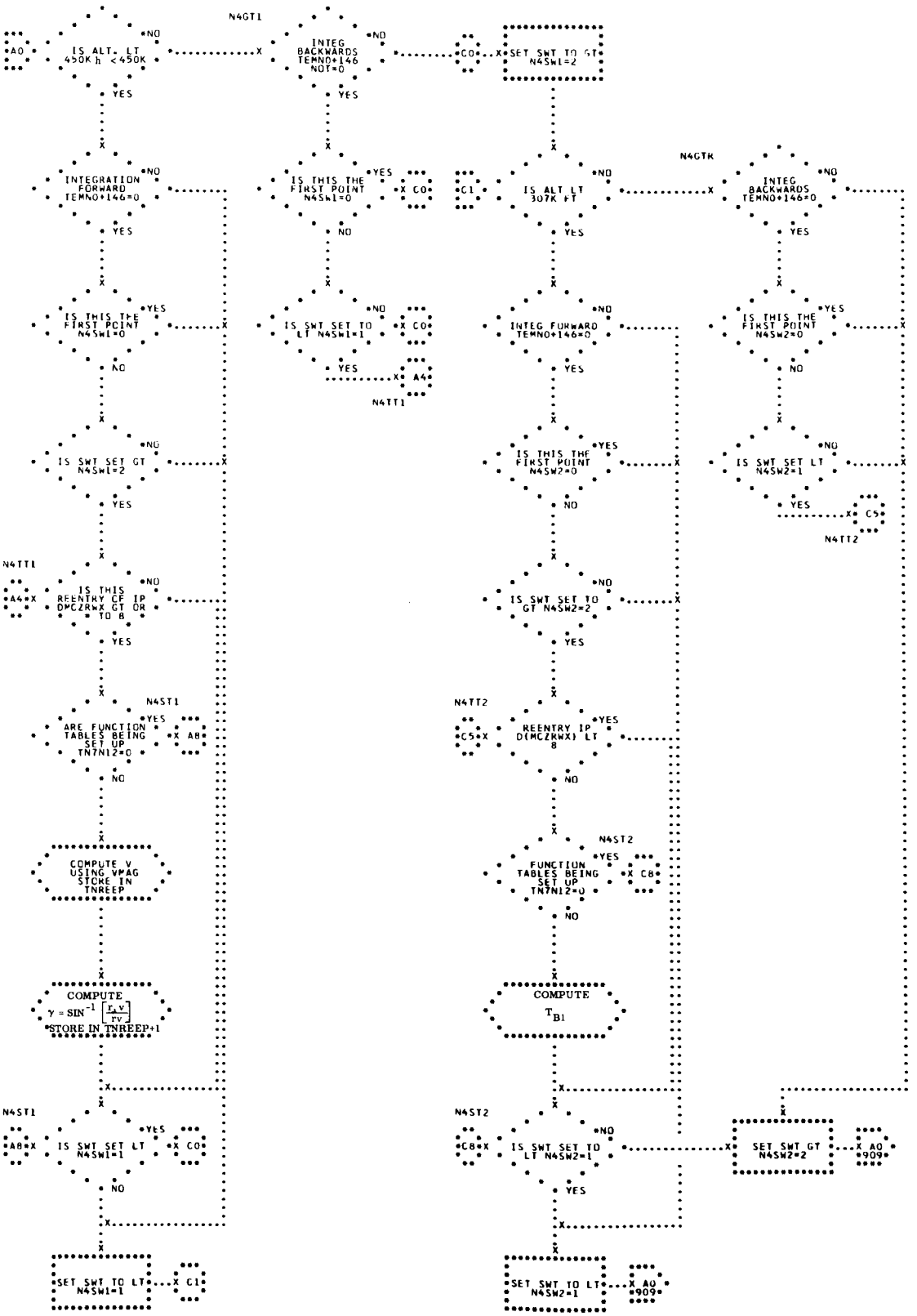
=54, NDC TABLES BASED ON ON-BOARD GUIDANCE EQUATIONS

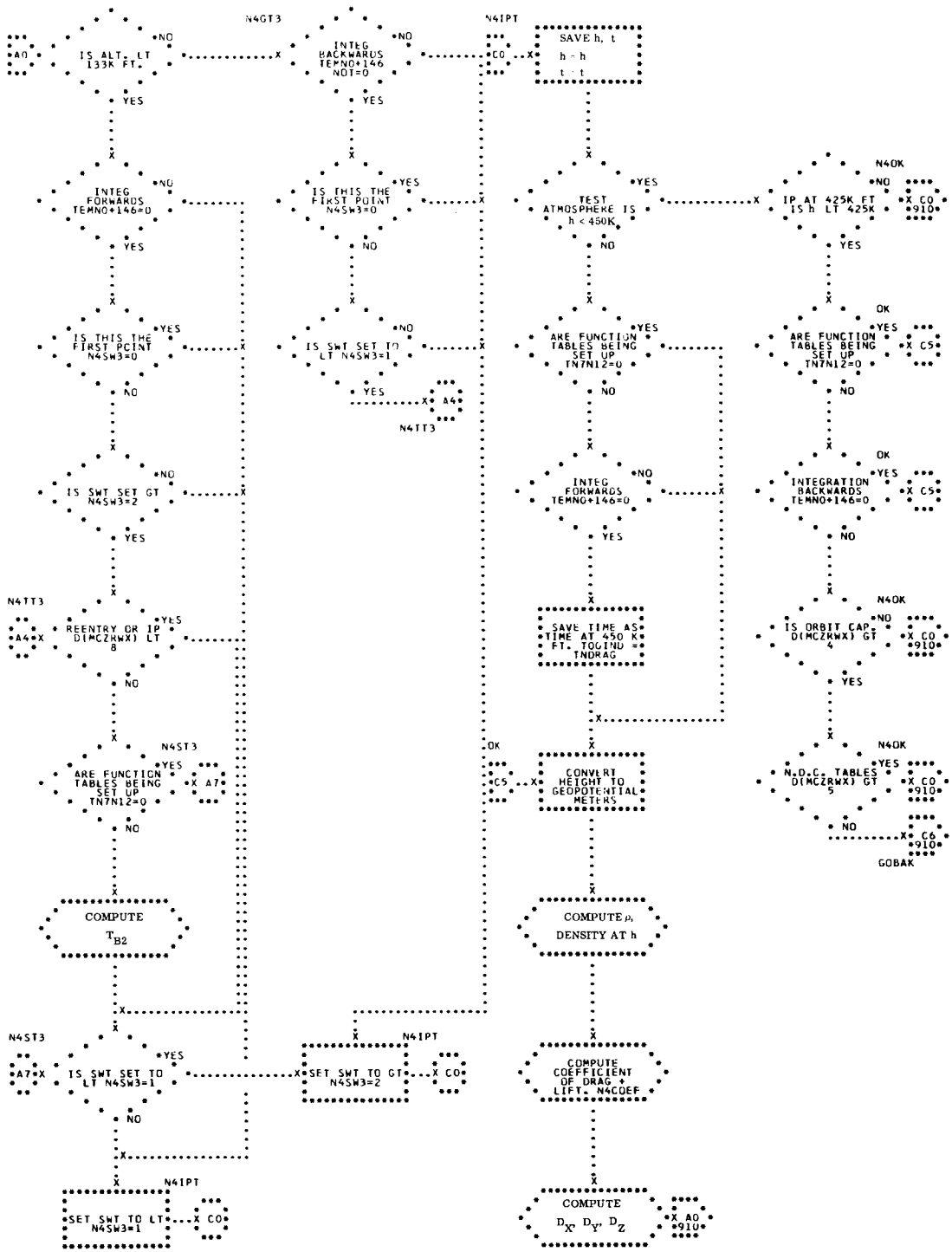
=55, REENTRY TABLE BASED ON ON-BOARD GUIDANCE EQUATIONS

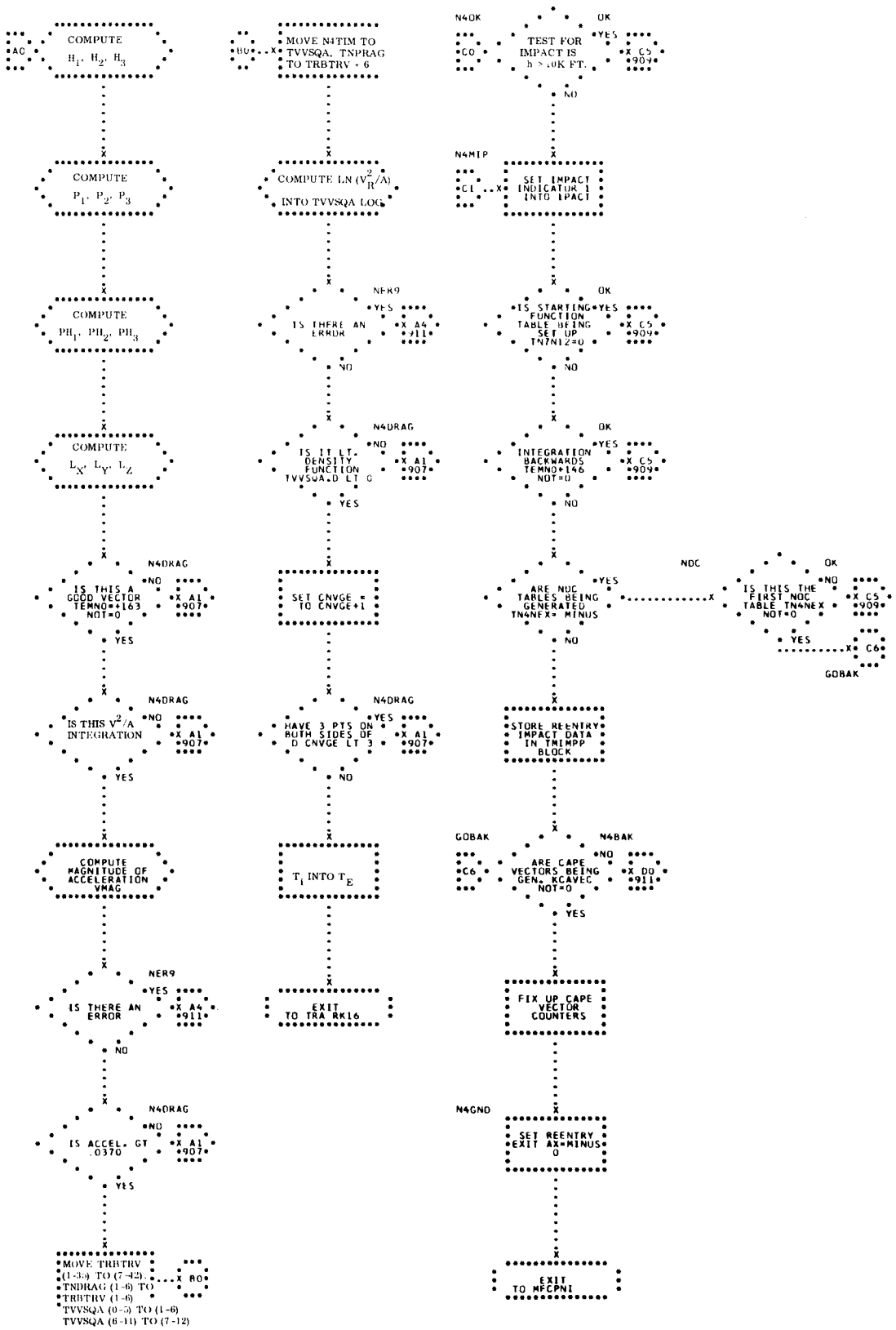
0=49 ARE ORBIT TYPE INTEGRATIONS

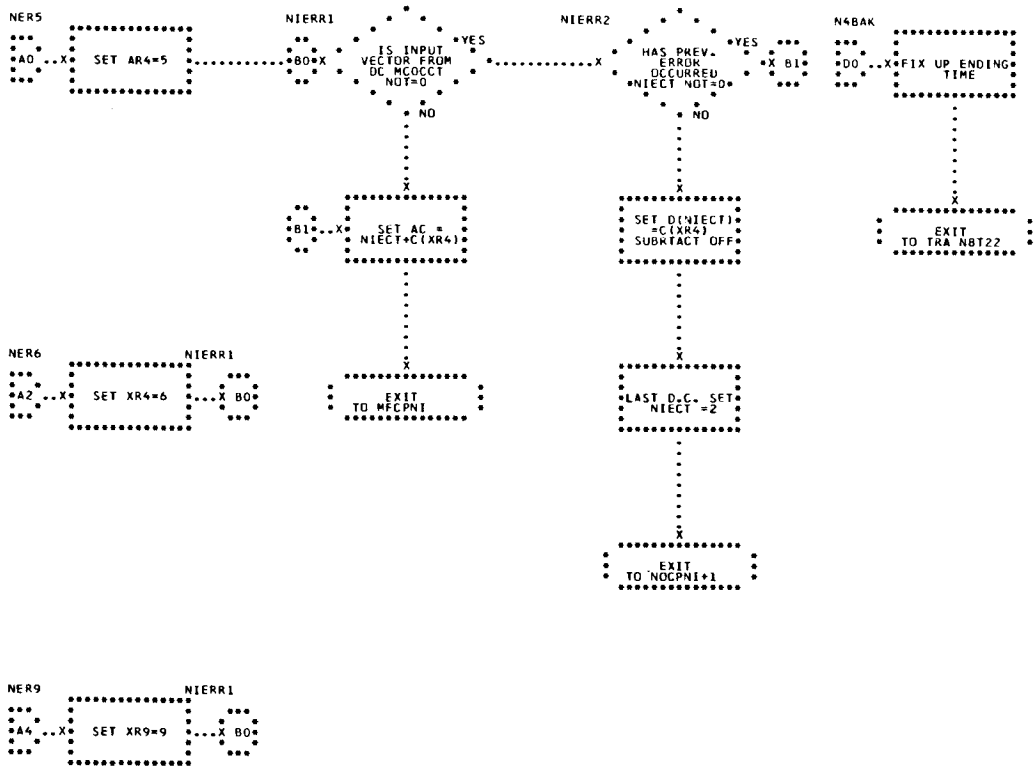
50-99 ARE REENTRY TYPES


ENTER N4DRAG
N4DRAG
A1
ENTER WITH T,X,Y,Z, XD,YD,ZD IN TNDRAG
$V_R^2 = (XD + Y_\omega)^2 +$
$(YD = X_\omega)^2 + (ZD)^2$
$V = \sqrt{V_R^2}$
SQRT
NER5
IS THERE AN ERROR
YES
X A0 911
NO
SAVE TIME IN FLT.PT. IN N4TIM FOR ALT. TESTS
SET UP TIME FOR A3MSCP A3TSUP
CALC. λ_0, IN FLT.PT. RADIAN,H IN GEMINI UNITS A3MSCP
NER6
IS THERE AN ERROR
YES
X A2 911
NO
N4MIP
IS ALT. NEG TN4LLH+2= MINUS
YES
X C1 910
NO
X B0
B0
OK
IS THIS A GOOD PT IS TEMNO+163 NOT=TO 0
NO
X C5 909
YES
IS THIS A CONVERGED TRAJ. IS KOVECT NOT=0
NO
X C1 908
YES
COMPUTE $\gamma = \sin^{-1}\left[\frac{\bar{r}\cdot\bar{v}}{rv}\right]$
N4IPT
IS TRAJ POINTED EARTH INWARD γ = +
YES
X C0 909
NO
PICK UP 2 POINTS SPANNING 400K FT. N4SALT
PICK UP 2 POINTS SPANNING 350K FT. N4SALT
N4GT6
IS ALT LESS THAN 325K FT
NO
X D2
YES
IS INTEGRATION FORWARD TEMNO+146=0
NO
X D0
YES
IS FIRST PT N4SW6=0
YES
X D0
NO
IS SWT SET GT N4SW6=2
NO
X D0
YES
X C0
C0
IS THIS ENTRY D(MCZRWX) GT 49
NO
X C0
YES
ARE FUNCTION TABLES SET UP TN7N12=0
YES
X D0
NO
COMPUTE T_{325K}
IS SWT SET LT N4SW6=1
YES
X D0
NO
X D6
D0
SET SWT TO LT N4SW6=1
X A0 908
N4GT6
D2
IS INTEGRATION BACKWARD TEMNO+146 NOT=0
NO
YES
IS THIS FIRST POINT N4SW6=0
YES
X D0
NO
IS SWT SET TO LT N4SW6=1
NO
YES
X C0
D6
SET SWT GT N4SW6=2
X A0 908


A0
IS ALT. LT 450K h < 450K
NO
YES
N4GT1
INTEG BACKWARDS TEMNO+146 NOT=0
C0
SET SWT TO GT N4SW1=2
INTEGRATION FORWARD TEMNO+146=0
IS THIS THE FIRST POINT N4SW1=0
IS SWT SET TO LT N4SW1=1
A4
N4TT1
C1
IS ALT LT 307K FT
N4GTR
INTEG BACKWARDS TEMNO+146=0
IS THIS THE FIRST POINT N4SW2=0
IS SWT SET GT N4SW1=2
IS SWT SET LT N4SW2=1
C5
N4TT2
N4TT1
IS THIS REENTRY CF IP DMCZRWX GT OR = TO 8
A4
N4ST1
ARE FUNCTION TABLES BEING SET UP TN7N12=0
A8
COMPUTE V USING VMAG STORE IN TNREEP
COMPUTE
$\gamma = \mathrm{SIN}^{-1}\left[\frac{r_{\cdot} v}{rv}\right]$
STORE IN TNREEP+1
N4ST1
IS SWT SET LT N4SW1=1
SET SWT TO LT N4SW1=1
C1
INTEG FORWARD TEMNO+146=0
IS THIS THE FIRST POINT N4SW2=0
IS SWT SET TO GT N4SW2=2
N4TT2
REENTRY IP D(MCZRWX) LT
C5
N4ST2
FUNCTION TABLES BEING SET UP TN7N12=0
C8
COMPUTE
T_{B1}
N4ST2
IS SWT SET TO LT N4SW2=1
SET SWT GT N4SW2=2
A0 909
SET SWT TO LT N4SW2=1
A0 909


N4GT3
N4IPT
A0
IS ALT. LT 133K FT.
NO
YES
INTEG BACKWARDS TEMNO+146 NOT=0
CO
SAVE h, t
h = h
t = t
INTEG FORWARDS TEMNO+146=0
IS THIS THE FIRST POINT N4SW3=0
TEST ATMOSPHERE IS h < 450K
N40K
IP AT 425K FT IS h LT 425K
IS SWT SET TO LT N4SW3=1
A4
N4TT3
ARE FUNCTION TABLES BEING SET UP TN7N12=0
OK
C5
INTEGRATION BACKWARDS TEMNO+146=0
IS SWT SET GT N4SW3=2
REENTRY OR IP D(MCZRWX) LT 8
IS ORBIT CAP D(MCZRWX) GT 4
SAVE TIME AS TIME AT 450 K FT. TOGIND = TNDRAG
N4ST3
N.D.C. TABLES D(MCZRWX) GT 5
CONVERT HEIGHT TO GEOPOTENTIAL METERS
C6
GOBAK
COMPUTE T_{B2}
COMPUTE ρ, DENSITY AT h
A7
IS SWT SET TO LT N4SW3=1
SET SWT TO GT N4SW3=2
COMPUTE COEFFICIENT OF DRAG + LIFT, N4COEF
SET SWT TO LT N4SW3=1
COMPUTE D_X, D_Y, D_Z
A0


COMPUTE H_1, H_2, H_3
COMPUTE P_1, P_2, P_3
COMPUTE PH_1, PH_2, PH_3
COMPUTE L_X, L_Y, L_Z
IS THIS A GOOD VECTOR TEMNO+163 NOT=0
IS THIS V^2/A INTEGRATION
COMPUTE MAGNITUDE OF ACCELERATION VMAG
IS THERE AN ERROR
IS ACCEL. GT .0370
MOVE TRBTRV (1-3) TO (7-42). TNDRAG (1-6) TO TRBTRV (1-6) TVVSQA (0-5) TO (1-6) TVVSQA (6-11) TO (7-12)
MOVE N4TIM TO TVVSQA, TNPRAG TO TRBTRV + 6
COMPUTE LN (V_R^2/A) INTO TVVSQA LOC
IS THERE AN ERROR
IS IT LT-DENSITY FUNCTION TVVSQA,D LT 0
SET CNVGE = TO CNVGE+1
HAVE 3 PTS ON BOTH SIDES OF D CNVGE LT 3
T_I INTO T_E
EXIT TO TRA RK16
TEST FOR IMPACT IS h > 10K FT.
SET IMPACT INDICATOR 1 INTO IPACT
IS STARTING FUNCTION TABLE BEING SET UP TN7N12=0
INTEGRATION BACKWARDS TEMNO+146 NOT=0
ARE NDC TABLES BEING GENERATED TN4NEX= MINUS
IS THIS THE FIRST NDC TABLE TN4NEX NOT=0
STORE REENTRY IMPACT DATA IN TMIMPP BLOCK
ARE CAPE VECTORS BEING GEN., KCAVEC NOT=0
FIX UP CAPE VECTOR COUNTERS
SET REENTRY EXIT AX=MINUS 0
EXIT TO MFCPNI
N4OK
N4MIP
N4DRAG
NER9
NDC
GOBAK
N4BAK
N4GND
OK
YES
NO
A0
B0
C0
C1
C6


NER5
AO
SET AR4=5
NIERR1
BO
IS INPUT VECTOR FROM DC MCOCCT NOT=0
YES
NO
NIERR2
HAS PREV. ERROR OCCURRED NIECT NOT=0
YES
X B1
NO
N4BAK
DO
FIX UP ENDING TIME
EXIT TO TRA N8T22
B1
SET AC = NIECT+C(XR4)
SET D(NIECT) =C(XR4) SUBTACT OFF
NER6
NIERR1
A2
SET XR4=6
BO
EXIT TO MFCPNI
LAST D.C. SET NIECT =2
EXIT TO NOCPNI+1
NER9
NIERR1
A4
SET XR9=9
BO
ERROR REQUTINE FROM SEC. 24-25

35. N4PDEN - COMPUTATION OF ATMOSPHERIC DENSITY

THIS PROGRAM PRODUCES, IN GEMINI UNITS, THE ATMOSPHERIC DENSITY CORRESPONDING TO A GIVEN ALTITUDE. IF THE ALTITUDE IS BELOW ONE METER THE DENSITY AT AN ALTITUDE OF ONE METER IS GIVEN.

INPUT

A) A ONE-LOCATION TABLE CONTAINING THE HEIGHT OF THE SPACECRAFT IN FLOATING POINT GEOPOTENTIAL METERS IS INPUT TO THE PROGRAM.

B) THE LIBRARY SUBROUTINES U1EXPE AND U1LOGE ARE USED.

C) CONSTANTS -

K001.0 DEC 1.0		FLOATING POINT ONE
K700K. DEC 700000.		FLOATING POINT SEVEN HUNDRED THOUSAND
K04.M5 DEC .000043425951		CONVERSION FACTOR kg/m^3 TO GEMINI DENSITY UNITS (GDU)

OUTPUT

A ONE-LOCATION TABLE CONTAINING THE DENSITY OF THE ATMOSPHERE IN FLOATING POINT GEMINI UNITS IS OUTPUT FROM THE PROGRAM.

METHOD

GIVEN AN H* (IN GEOPOTENTIAL METERS) THE FOLLOWING COMPUTATIONS ARE MADE† - (ρ) IS THE ATMOSPHERIC DENSITY IN kg/m^3.

A) IF $H^* \geq 700{,}000$ GEOPOTENTIAL METERS, $\rho = 0$.

B) IF $H^* < 700{,}000$ GEOPOTENTIAL METERS, THEN

$$\ln\rho = \ln\rho_b - \left[1 + \frac{m_0 g_0^*}{R^* L_m}\right] \cdot \ln\left[1 + \frac{L_m}{(T_m)_b}\ (H^* - H_b^*)\right]$$

WHERE

m_0 = 28.966, THE MOLECULAR WEIGHT OF AIR AT SEA LEVEL

g_0^* = 9.80665 $m^2/sec^2 m^*$ IS THE ACCELERATION DUE TO GRAVITY AT HEIGHT H*

R^* = 8.31439 $.10^3$ joules/^{0}KKg

H_b IS A BASE HEIGHT - ONE OF AN ELEVEN-LOCATION TABLE - EXPRESSED IN GEOPOTENTIAL METERS.

$(T_m)_b$ IS A TEMPERATURE - ONE OF AN ELEVEN-LOCATION TABLE - EXPRESSED IN ^{0}K AT THE VARIOUS BASE HEIGHTS.

---

† CF. AN INTRODUCTION TO ASTRODYNAMICS, BAKER AND MAKEMSON, ACADEMIC PRESS, INC., NEW YORK, 1960, PP 103-107.

$\ln\rho_b$ IS A NATURAL LOGARITHM - ONE OF AN ELEVEN-LOCATION TABLE - OF THE DENSITY AT THE VARIOUS BASE HEIGHTS.

$(L_m)_b = \frac{dT_m}{dH^*}$ IS A TEMPERATURE GRADIENT - ONE OF AN ELEVEN-LOCATION TABLE - FOR REGIONS BETWEEN THE VARIOUS BASE HEIGHTS.

IN SOME ISOTHERMAL REGIONS, $L_m = 0$. IN THESE REGIONS

$$\ln\rho = \ln\rho_b - \frac{m_0 g_0^*}{R^*} \frac{H^* - H_b^*}{(T_m)_b}$$

$$\rho\,(\text{in kb/m}^3) = e^{\ln}$$

$$\rho\,(\text{in GDU}) = 4.3425951 \times 10^{-5}\,\rho\,(\text{in kg/m}^3)$$

THE UNIT OF DENSITY FOR THE GEMINI SYSTEM IS DEFINED TO BE (THE MASS OF THE EARTH)/(THE EQUATORIAL RADIUS OF THE EARTH).^{3} THE MASS OF THE EARTH IS TAKEN AS 5.975×10^{24} KILOGRAMS AND THE EQUATORIAL DISTANCE AS 6,378,165 METERS.

HENCE,

$$\rho(\text{GDU}) = 4.3425951 \times 10^{-5}\,\rho(\text{kg/m}^3)$$

THE TABLES USED WERE OBTAINED FROM DATA TAKEN FROM ARDC MODEL 1959 ATMOSPHERE.

USAGE

A) CALLING SEQUENCE -

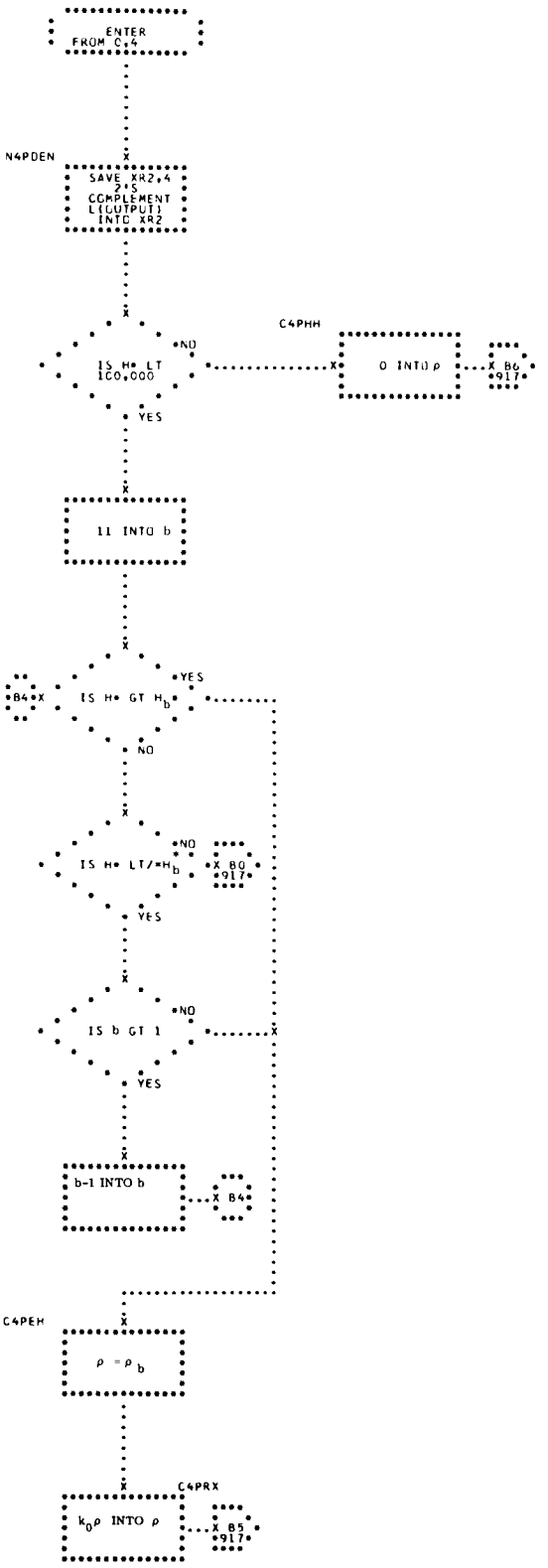
```
TSX         N4PDEN,4
PZE         L(INPUT),,L(OUTPUT)
ERROR       RETURN
NORMAL      RETURN
```

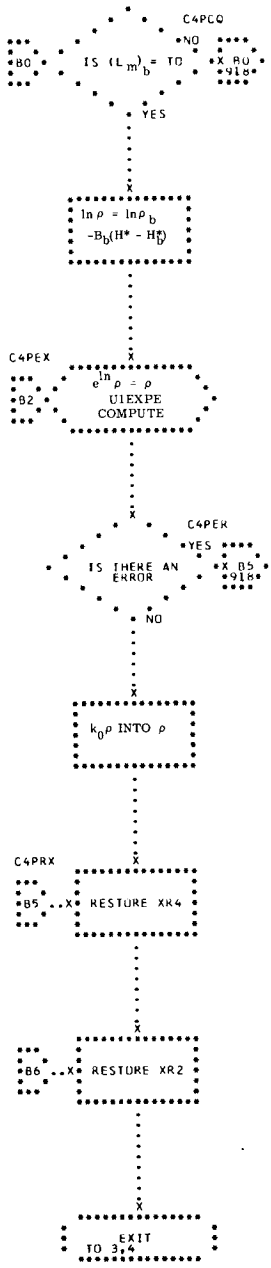
B) STORAGE REQUIRED - 115_{10} DECIMAL LOCATIONS EXCLUDING STORAGE REQUIRED BY SUBROUTINES AND EXTERNAL CONSTANTS.

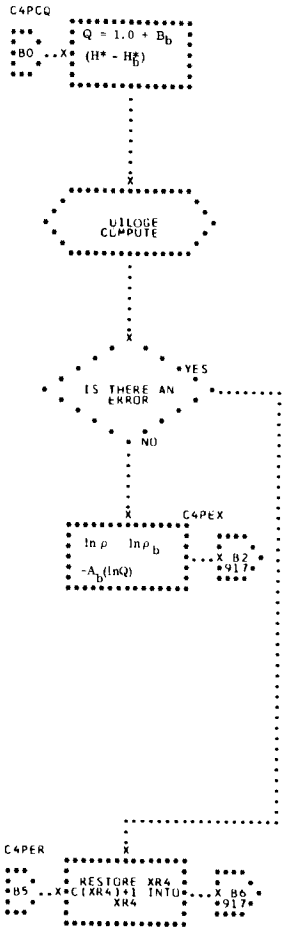
C) TIME REQUIRED - 0.520 MILLISECONDS (MAXIMUM) INCLUDING 0.128 MILLISECONDS FOR U1LOGE AND 0.200 MILLISECONDS FOR U1EXPE.

D) ERROR CONDITIONS - AN ERROR RETURN FROM N4PDEN IMPLIES AN ERROR RETURN FROM A LIBRARY SUBROUTINE.

E) CHECKOUT - RESULTS FROM N4PDEN WERE COMPARED WITH THOSE OF A DESK CALCULATOR. THESE WERE SATISFACTORY.


ENTER FROM C.4
N4PDEN
SAVE XR2,4 ?1'S COMPLEMENT L[OUTPUT] INTO XR2
IS H• LT 1C0,000
•NO
C4PHH
0 INTO ρ
X B6 •917•
• YES
11 INTO b
IS H• GT Hb
•YES
•B4•X
• NO
IS H• LT/•Hb
•NO
X B0 •917•
• YES
IS b GT 1
•NO
• YES
b-1 INTO b
X B4
C4PEH
ρ = ρb
C4PRX
k0ρ INTO ρ
X B5 •917•


C4PCQ
NO
B0
IS $(L_m)_b$ = TO
X B0
918
YES
$\ln\rho = \ln\rho_b$
$-B_b(H^* - H_b^*)$
C4PEX
B2
$e^{\ln\rho} = \rho$
U1EXPE
COMPUTE
C4PER
YES
IS THERE AN
ERROR
X B5
918
NO
$k_0\rho$ INTO ρ
C4PRX
B5
RESTORE XR4
B6
RESTORE XR2
EXIT
TO 3,4


C4PCQ
$Q = 1.0 + B_b$
$(H^* - H_b^*)$
BO
X
ULOGE COMPUTE
IS THERE AN ERROR
YES
NO
C4PEX
$\ln \rho \quad \ln \rho_b$
$-A_b(\ln Q)$
B2
917
C4PER
RESTORE XR4
C(XR4)+1 INTO XR4
B5
B6
917

36. N4COEF - COEFFICIENT OF DRAG CALCULATION

THE N4COEF PROGRAM CALCULATES THE COEFFICIENT OF ATMOSPHERIC DRAG ON THE SPACECRAFT AND THE COEFFICIENT OF LIFT FOR THE CAPSULE IN REENTRY. (SEE FLOWCHART)

INPUT

A) KODENS CONTAINS ALTITUDE IN GEOPOTENTIAL METERS, FLOATING-POINT.

B) TVELOC CONTAINS VELOCITY VECTOR MAGNITUDE GEMINI UNITS, FLOATING-POINT.

C) MCZRWX MONITOR CONTROL WORD WHOSE DECREMENT INDICATES THE TYPE OF INTEGRATION BEING GENERATED. SEE NOCPNI.

OUTPUT

LOCATION TCSUBD, WHICH CONTAINS THE COEFFICIENT OF DRAG IN GEMINI UNITS (AREA/MASS OF SPACECRAFT), IS THE OUTPUT FROM N4COEF.

METHOD

N4COEF USES THE FOLLOWING EQUATIONS -

A) $C_D = (C_D')\ M$

$C_L = (C_L')\ M$

$$KCD.CW = \frac{A_{F.A.\ of\ cap.}}{M_{cap.} * A_{earth}} * M_{earth} = \frac{412952008000}{M_{cap.} * 0.453923}$$

WHERE F.A. IS FRONTAL AREA

B) T_M IS A MOLECULAR SCALE TEMPERATURE DEFINED AS (ABSOLUTE TEMPERATURE AT ALTITUDE IN QUESTION) (MOLECULAR WEIGHT OF AIR AT SEA LEVEL)/(MOLECULAR WEIGHT OF AIR AT THE ALTITUDE IN QUESTION.) CF. AN INTRODUCTION TO ASTRODYNAMICS, BAKER AND MAKERNSON P. 104.

C) H_B IS ONE OF A 9-LOCATION TABLE OF BASE ALTITUDES.

D) $(T_M)_B$ IS THE VALUE OF T_M AT BASE ALTITUDE H_B .

E) $L_M = \frac{dt_M}{dH}$ = (GRADIENT) AND IS ONE OF A NINE LOCATION OF TABLE OF GRADIENTS.

F) FOR H < 90,147 GEOPOTENTIAL METERS (G.P. METERS) -

$$T_M = (T_M)_B + (L_M)_B (H-H_B)$$

$$C_S = 20,046333\ T_M$$

MACH = 7905,3764

C'_D AND C'_L ARE THEN SELECTED FROM A TABLE WHICH CORRESPONDS TO A MACH NUMBER TABLE.

G) FOR 150,000 < H $\leq$ 172,205

$$C'_D = 1.75 + .25 \cos \pi \frac{172205-H}{.30439}$$

$$C'_L = 0.0$$

H) FOR H > 17.2,205

$C'_D = 2.0$

$C'_L = 0.0$

I) FOR 90,147 < H < 150,000

C_S = 259.08 METERS/SEC

MACH = 7905.3764 (V/CS)

C'_D AND C_L ARE THEN SELECTED AS ABOVE.

J) TO SPECIFY CONSTANT DRAG

CCSUBD IS USED AS C'_D

K) TABLES

1) HUSBB (H_B) DEC 16.E4,105.E3 9.E4,79.E3, 53. E3, 47.E3, 25.E3, 11.E3, 0. TABLE OF ALTITUDES IN G.M.

2) LSUBM (L_M) DEC 0., .02, .004, 0., - .0045, 0., .003, 0., - .0065 TABLE OF CORRECTIVE FACTORS FOR DETERMINING T_M

3) N4CSD (C'_D) TABLE OF C'_D IN GEMINI UNITS, WHICH CHANGES EACH FLIGHT.

4) N4MCK (MACH) TABLE OF MACH NUMBERS WHICH CHANGES EACH FLIGHT.

5) N4CSL (C'_L) TABLE OF C'_L IN GEMINI UNITS WHICH CHANGES.

USAGE

A) CALLING SEQUENCE -

```
        TSX         N4COEF,4
    +1              ERROR RETURN
    +2
```

B) STORAGE REQUIRED 115 LOCATIONS INCLUDING INTERNAL TABLES.

C) CONSTANTS -

KCD.CW	PZE	AREA/MASS IN GEMINI UNITS (FLOATING-POINTING), WHERE WEIGHT FACTOR REPRESENTS WEIGHT OF SPACECRAFT. VARIABLE BECAUSE OF WEIGHT CHANGE.
K.MACH	DEC 7905.3757	GEMINI UNITS OF VELOCITY.
KSUND	DEC 20.0463333	PROPORTIONALITY CONTANTS, $C_s - T_m$ EQUATION.
K00.25	DEC .25	
K001.5	DEC 1.5	
K01.75	DEC 1.75	
K002.0	DEC 2.0	
K00.PI	DEC 3.14159265	
NER 15		ERROR IN COMPUTING VELOCITY OF SOUND.
NER 16		NO C_D IN TABLE FOR MEDIUM ALTITUDE CASE.
NER 17		ERROR IN COMPUTING C_D FOR LOW ALTITUDE CASE.
K010.0	DEC 10.0	
K30439	DEC 30439.0	
KGM3.5	DEC 90147.607	
KL.HGT	DEC 141766.0	
KT.GHT	DEC 172205.0	

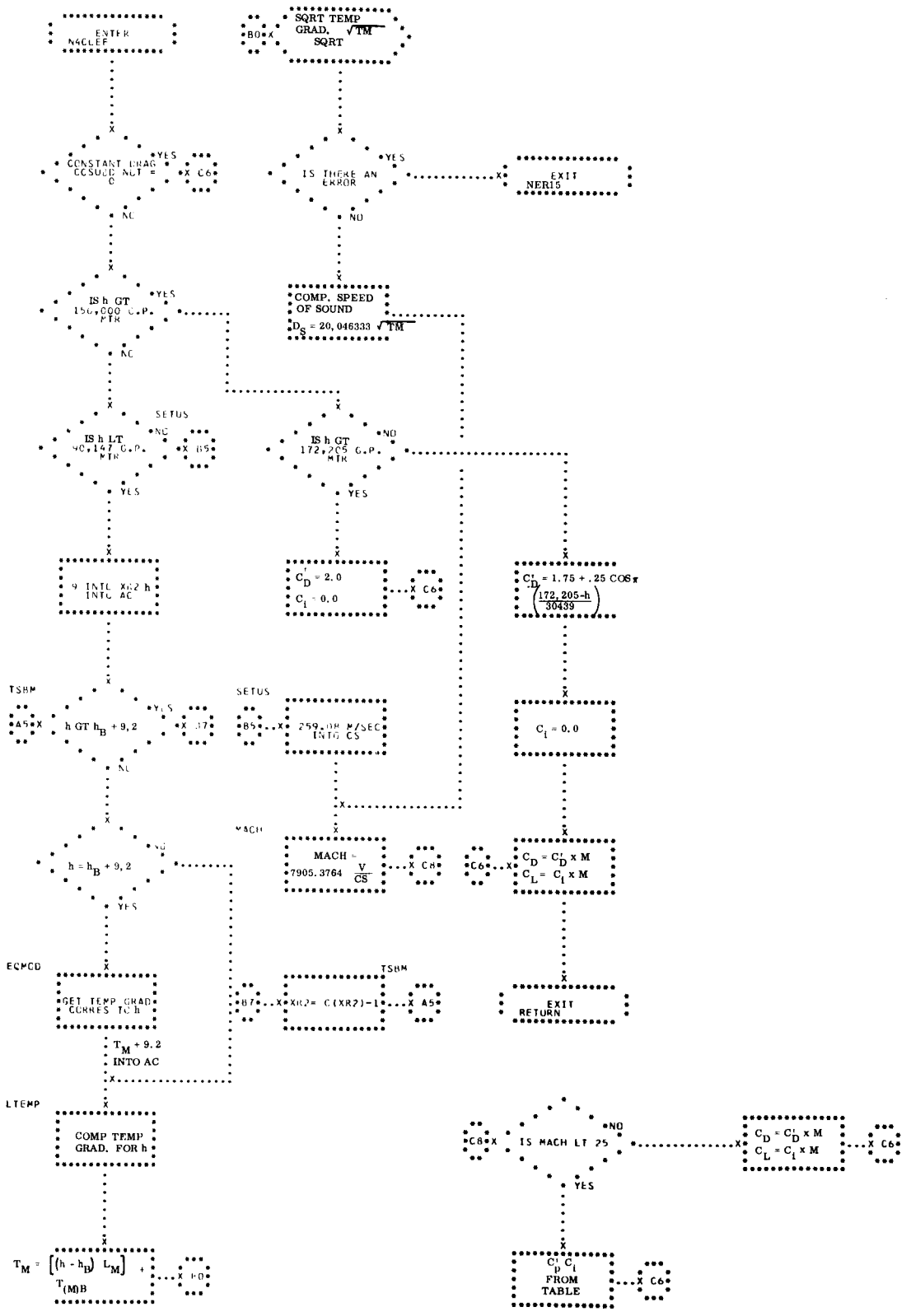
COMMON LOCATIONS - CA SYN STRGE

TIME REQUIRED -

1) NORMAL TIME = 0.672 MILLISECONDS

2) MAXIMUM TIME = (3.312 + .06 + .108) MILLI-SECONDS WHERE 1 = NUMBER OF LOOKUPS IN HSUBB TABLE j = NUMBER OF LOOKUPS IN MACH # TABLE

CHECKOUT- ALL POSSIBILITIES WERE TESTED AGAINST CHECK CALCULATIONS. N4COEF WAS RUN WITH NUMERICAL INTEGRATION AND COM-PARED WITH NOCPNI RUNS, USING A CON-STANT COEFFICIENT OF DRAG.


ENTER N4CLEF
B0 X
SQRT TEMP GRAD, $\sqrt{TM}$ SQRT
CONSTANT DRAG COEFF NOT = 0
YES
X C6
NO
IS THERE AN ERROR
YES
EXIT NER15
NO
IS h GT 156,000 G.P. MTR
YES
NO
COMP. SPEED OF SOUND
$D_S = 20,046333 \sqrt{TM}$
SETUS
IS h LT 90,147 G.P. MTR
NO
X B5
YES
IS h GT 172,205 G.P. MTR
NO
YES
9 INTO XR2 h INTO AC
$C_D' = 2.0$
$C_l = 0.0$
X C6
$C_D' = 1.75 + .25 \cos\pi\left(\frac{172,205-h}{30439}\right)$
TSBM
A5 X
h GT h_B + 9,2
YES
X 37
NO
SETUS
B5 X
259.08 M/SEC INTO CS
$C_l = 0.0$
MACH
$h = h_B + 9,2$
YES
MACH = 7905.3764 $\frac{V}{CS}$
X C4
C6 X
$C_D = C_D' \times M$
$C_L = C_l \times M$
EQMOD
GET TEMP GRAD CORRES TO h
TSBM
B7 X
XR2 = C(XR2)-1
X A5
EXIT RETURN
T_M + 9.2 INTO AC
LTEMP
COMP TEMP GRAD. FOR h
C8 X
IS MACH LT 25
NO
YES
$C_D = C_D' \times M$
$C_L = C_l \times M$
X C6
$T_M = \left[(h - h_B) L_M\right] + T_{(M)B}$
X F0
$C_p' C_l'$ FROM TABLE
X C6

37. N50TBI - NUMERICAL INTEGRATION OUTPUT TABLE INITIALIZATION

N50TBI STORES THE ANCHOR POINT IN THE INTEGRATION OUTPUT TABLE AND INITIALIZES QUANTITIES NEEDED BY N50TBL FOR STORING THE REMAINDER OF THE INTEGRATION OUTPUT TABLE. IN THE CASE OF SHORT ARC SOLUTIONS, N50TBI CALCULATES THE OUTPUT STEP INTERVAL FOR STATIONS IN D.C. AND STORES DEFINING VECTORS. (SEE FLOWCHART)

INPUT

A)	K0007A	ANCHOR POINT TIME, EXPRESSED IN FIXED OR FLOATING POINT SECONDS OR MINUTES, DEPENDING UPON K0007U AND KFL0FX.
B)	K0007L	CONTAINS ADDRESS, IN FIXED-POINT, OF THE START OF THE OUTPUT TABLE.
C)	K0000H	THE TIME STEP, IN FIXED OR FLOATING POINT AT WHICH THE POINTS ARE TO BE INTEGRATED.
D)	K0007U	IF NON-ZERO, K0000H REPRESENTS SECONDS, IF ZERO, K0000H REPRESENTS MINUTES.
E)	K00DTB	LENGTH OF TIME TO INTEGRATE BACKWARD, EXPRESSED IN FIXED OR FLOATING POINT MINUTES OR SECONDS DEPENDING ON K0007U.
F)	K00DTF	LENGTH OF TIME TO INTEGRATE FORWARD EXPRESSED IN FIXED OR FLOATING MINUTES OR SECONDS, DEPENDING ON K0007U.

G) KDORND — IF ZERO, THE N4DRAG PROGRAM IS USED, IF NOT ZERO, THE N4DRAG PROGRAM IS NOT USED.

H) KRORRV — IF ZERO, $\vec{r}$ VALUES ONLY ARE IN THE OUTPUT TABLE, IF ONE $\vec{r}$ AND $\vec{v}$ VALUES BOTH ARE IN THE OUTPUT TABLE (IN FIXED-POINT).

I) KINTVN — NUMBER OF INTEGRATION STEPS, IN FIXED-POINT, PER OUTPUT. THIS CAN BE ANY POSITIVE INTEGER.

J) KOOOTI — PRESENT TIME IN FIXED OR FLOATING POINT.

K) MCIVOS — LOCATES ENTRY IN TMSTMS, FIXED-POINT.

L) MCIVSS — INDICATES TIME STEP, ZERO FOR CONSTANT OUTPUT INTERVAL, NON-ZERO, VARIABLE INTERVAL.

M) TMBBNI + 6 — EQU KOOOTA

N) TMBBNI + 10 — EQU KOOOTL

O) TMSTMS — REFERENCE TABLE FOR DIFFERENTIAL CORRECTION.

P) TEMNO + 146 — INDICATES INTEGRATION DIRECTION - A ZERO TO INTEGRATE FORWARD - A NON-ZERO, BACKWARD.

OUTPUT

A) KOOOST — STARTING TIME OF THE OUTPUT TABLE, EXPRESSED IN FIXED OR FLOATING POINT MINUTES OR SECONDS DEPENDING ON KOOOTU.

B) KOOOET — ENDING TIME OF THE OUTPUT TABLE, EXPRESSED IN FIXED OR FLOATING POINT

MINUTES OR SECONDS DEPENDING ON KOOOTU.

C) KOOOTU — DEFINES OUTPUT TABLE AS MINUTES OR SECONDS TABLE, ZERO AND NON-ZERO, RESPECTIVELY.

D) A QUANTITY (H X N) WHICH DEFINES OUTPUT INTERVAL OF TABLE.

METHOD

N5OTBI DETERMINES WHETHER THE INTEGRATION STORING INTERVAL WILL BE CONSTANT OR VARIABLE BEFORE STORING THE INITIAL VECTOR AND DEFINING THE OUTPUT PARAMETERS. IF THE INTERVAL IS VARIABLE, THE ENTRY FOR THE VARIABLE INTERVAL IS TAKEN FROM TMSTMS AND THE NUMBER OF STEPS BEFORE THE NEXT INTEGRATION IS THEN COMPUTED.

USAGE

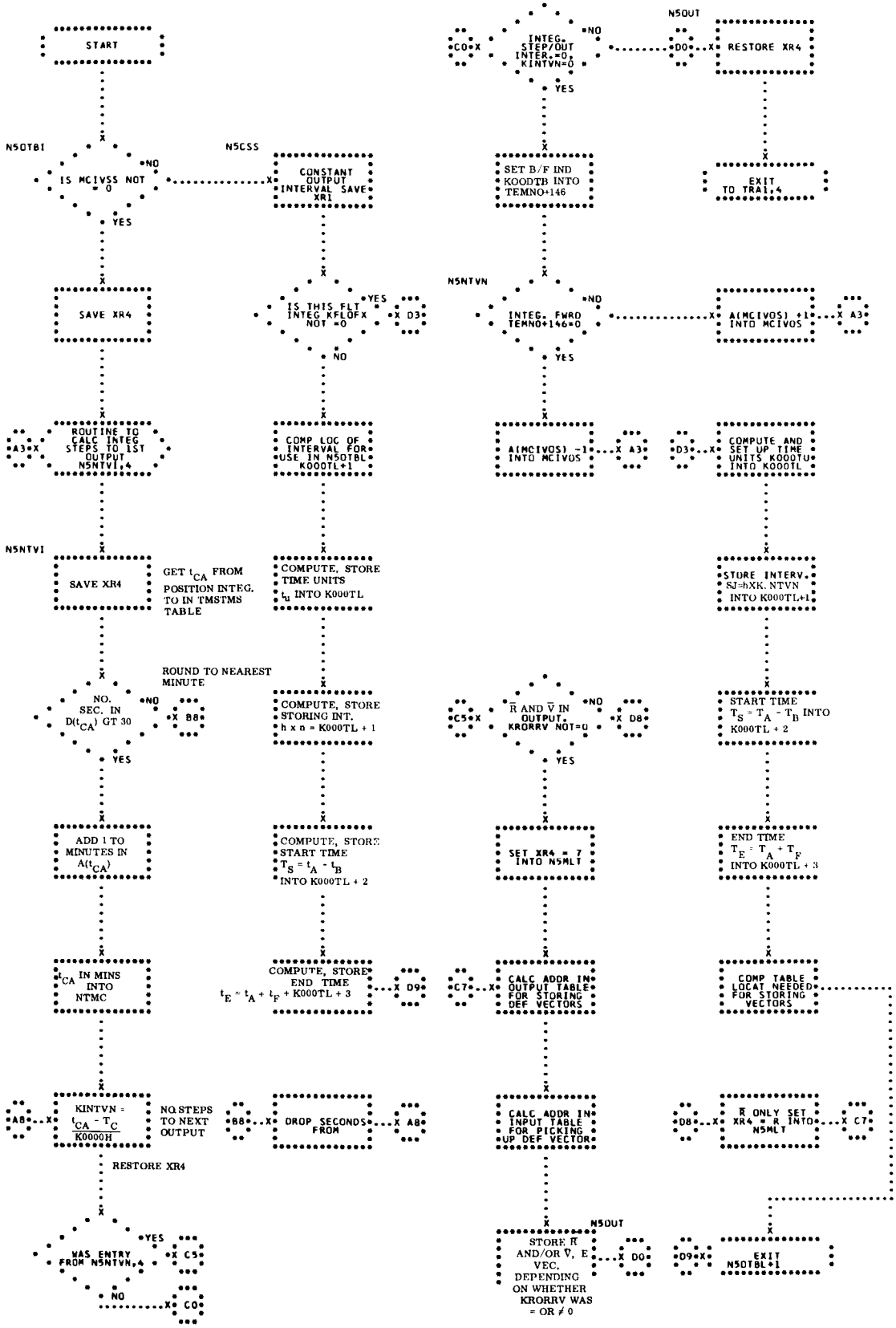
A) CALLING SEQUENCE -

```
ALPHA      TSX   N5OTBI,4
ALPHA+1           NORMAL RETURN
```

B) STORAGE REQUIRED - 19 LOCATIONS

C) CONSTANTS -

```
K00001  DEC   1
```


START
N5OTBI
IS MCIVSS NOT = 0
NO
YES
N5CSS
CONSTANT OUTPUT INTERVAL SAVE XR1
IS THIS FLT INTEG KFLOFX NOT =0
YES
NO
D3
SAVE XR4
ROUTINE TO CALC INTEG STEPS TO 1ST OUTPUT N5NTVI,4
A3
COMP LOC OF INTERVAL FOR USE IN N5OTBL K000TL+1
N5NTVI
SAVE XR4
GET t_{CA} FROM POSITION INTEG. TO IN TMSTMS TABLE
COMPUTE, STORE TIME UNITS t_u INTO K000TL
ROUND TO NEAREST MINUTE
NO. SEC. IN D(t_{CA}) GT 30
NO
YES
B8
COMPUTE, STORE STORING INT. h x n = K000TL + 1
ADD 1 TO MINUTES IN A(t_{CA})
COMPUTE, STORE START TIME $T_S = t_A - t_B$ INTO K000TL + 2
t_{CA} IN MINS INTO NTMC
COMPUTE, STORE END TIME $t_E = t_A + t_F$ + K000TL + 3
D9
A8
KINTVN = $\frac{t_{CA} - T_C}{K0000H}$
NO STEPS TO NEXT OUTPUT
B8
DROP SECONDS FROM
A8
RESTORE XR4
WAS ENTRY FROM N5NTVN,4
YES
C5
NO
C0
C0
INTEG. STEP/OUT INTER.=0, KINTVN=0
NO
YES
SET B/F IND KOODTB INTO TEMNO+146
N5NTVN
INTEG. FWRD TEMNO+146=0
NO
YES
A(MCIVOS) -1 INTO MCIVOS
A3
C5
$\bar{R}$ AND $\bar{V}$ IN OUTPUT, KRORRV NOT=0
NO
YES
D8
SET XR4 = 7 INTO N5MLT
C7
CALC ADDR IN OUTPUT TABLE FOR STORING DEF VECTORS
CALC ADDR IN INPUT TABLE FOR PICKING UP DEF VECTOR
N5OUT
STORE $\bar{R}$ AND/OR $\bar{V}$, E VEC. DEPENDING ON WHETHER KRORRV WAS = OR ≠ 0
D0
N5OUT
RESTORE XR4
EXIT TO TRA1,4
A(MCIVOS) +1 INTO MCIVOS
A3
COMPUTE AND SET UP TIME UNITS K000TU INTO K000TL
STORE INTERV. SJ=hXK. NTVN INTO K000TL+1
START TIME $T_S = T_A - T_B$ INTO K000TL + 2
END TIME $T_E = T_A + T_F$ INTO K000TL + 3
COMP TABLE LOCAT NEEDED FOR STORING VECTORS
D8
$\bar{R}$ ONLY SET XR4 = R INTO N5MLT
C7
D9
EXIT N5OTBL+1

38. N5OTBL - NUMERICAL INTEGRATION OUTPUT TABLE STORING

N5OTBL STORES $\dot{\vec{r}}$, OR $\vec{r}$ AND $\vec{v}$ VALUES INTO THE INTEGRATION OUTPUT TABLE ACCORDING TO THE OUTPUT FORMAT PRESENTED IN NOCPNI. (SEE FLOWCHART)

INPUT

A) KOOOTL — CONTAINS ADDRESS, IN FIXED-POINT, OF THE START OF THE OUTPUT TABLE.

B) KRORRV — IF ZERO, ONLY $\vec{r}$ VALUES ARE IN THE OUTPUT TABLE, IF ONE $\vec{r}$ AND $\vec{v}$ VALUES ARE IN THE OUTPUT TABLE (IN FIXED-POINT).

C) KOOOST — STARTING TIME OF THE OUTPUT TABLE EXPRESSED IN FIXED-POINT MINUTES OR SECONDS, DEPENDING UPON KOOOTU.

D) TNFUNK + 22-X
E) TNFUNK + 48-Y
F) TNFUNK + 74-Z
} $\vec{r}$

G) TNFUNK + 24-$\dot{X}$
H) TNFUNK + 50-$\dot{Y}$
I) TNFUNK + 76-$\dot{Z}$
} $\dot{\vec{r}}$ or $\vec{v}$

J) KOOOTI — TIME ASSOCIATED WITH THE ABOVE VECTORS.

K) KOONTS — IF ZERO, STORE TABLE, IF NON-ZERO, DO NOT STORE TABLE.

L) TEMNO + 146 — DIRECTION INDICATOR - IF ZERO, INTEGRATE FORWARD. IF NON-ZERO, BACKWARD.

M) MCIVOS — LOCATES ENTRY IN TMSTMS (FIXED-POINT).

N) MCIVSS — INDICATES TIME STEP - IF ZERO, INDICATES CONSTANT OUTPUT INTERVAL, IF NON-ZERO, VARIABLE OUTPUT INTERVAL.

O) KFLOFX — IF ZERO, INDICATES TIME IN FIXED POINT UNITS. IF NON-ZERO, INDICATES TIME IN FLOATING POINT UNITS.

OUTPUT

THE OUTPUT TABLE OF N5OTBL HAS THE FOLLOWING FORMAT-

A) INDICATION OF MINUTES OR SECONDS. ZERO FOR MINUTES, 1 FOR SECONDS.

B) INTEGRATION OUTPUT INTERVAL

C) STARTING TIME

D) ENDING TIME

E) X
F) Y
G) Z
— $\vec{r}$ VECTOR IN GEMINI UNITS

H) $\dot{X}$
I) $\dot{Y}$
J) $\dot{Z}$
— $\vec{v}$ VECTOR IN GEMINI UNITS

NOTE - IF VECTORS ONLY ARE DESIRED IN THE OUTPUT TABLE, WORD 8 CONTAINS THE FIRST COMPONENT OF THE $\vec{r}$ VECTOR.

METHOD

NOT APPLICABLE

USAGE

A) CALLING SEQUENCE -

```
        TSX N5OTBL,4
    +1      NORMAL RETURN
```

B) SPACE REQUIRED - 154 LOCATIONS

C) CONSTANTS -

K00001 DEC 1

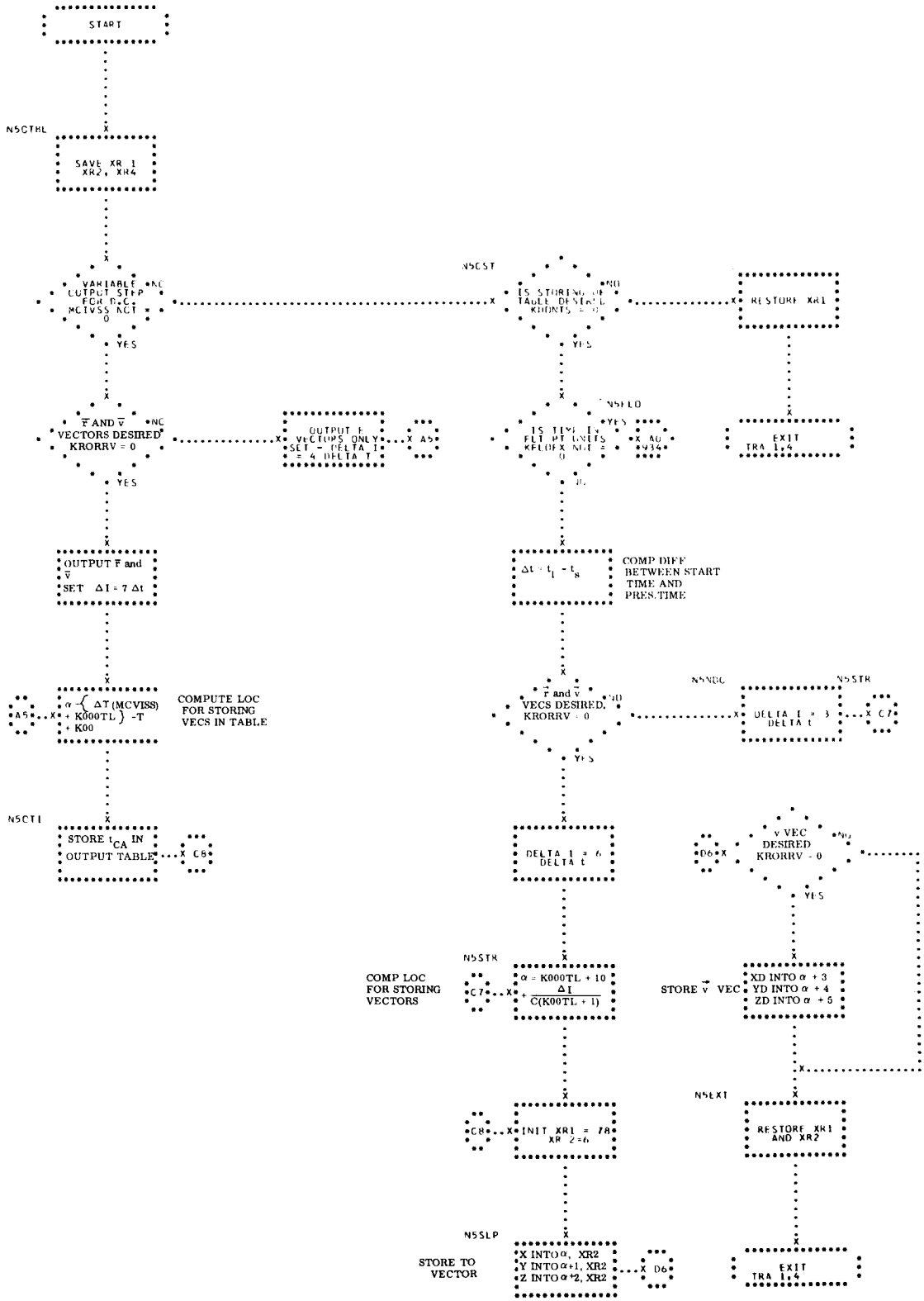
K00003 DEC 3

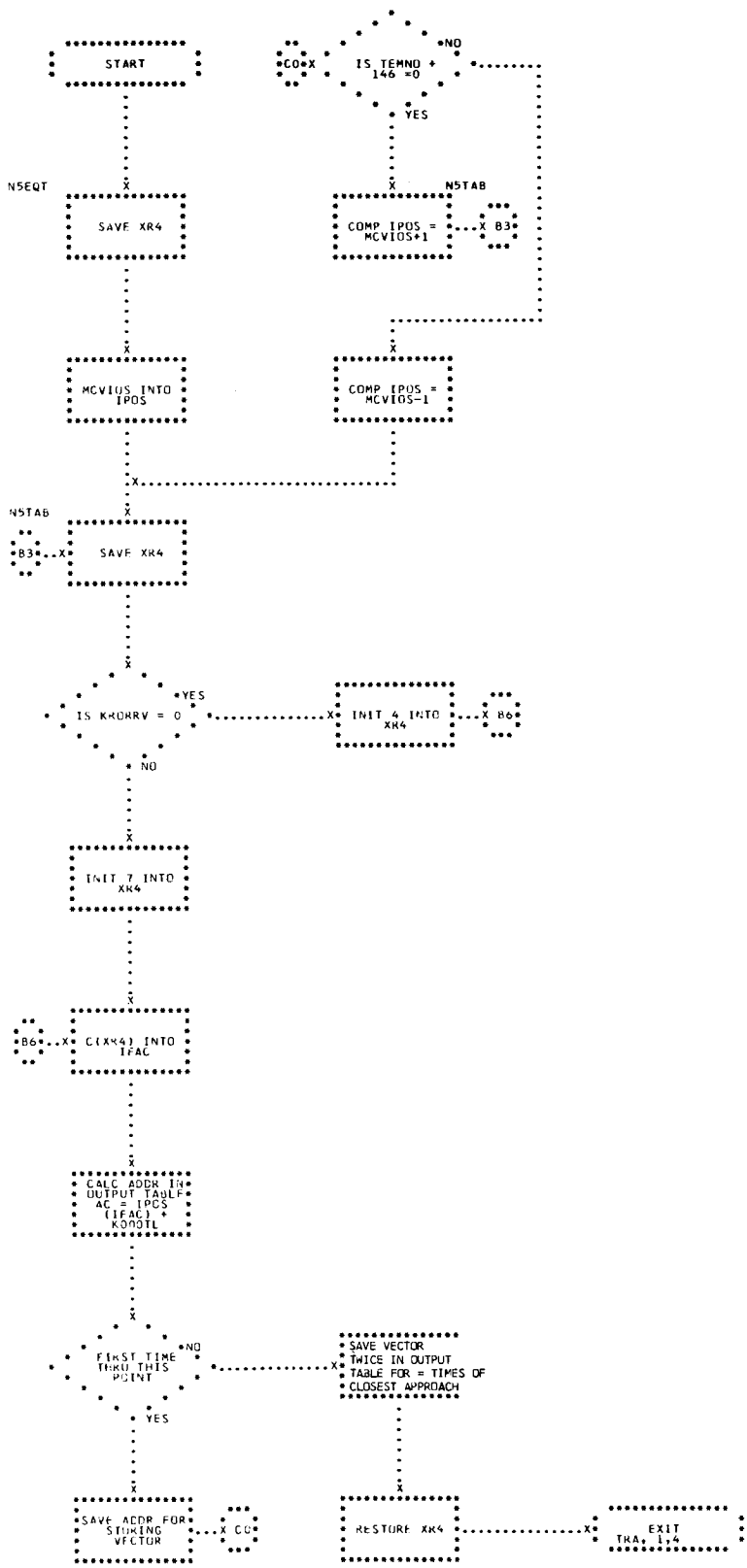
K00006 DEC 6

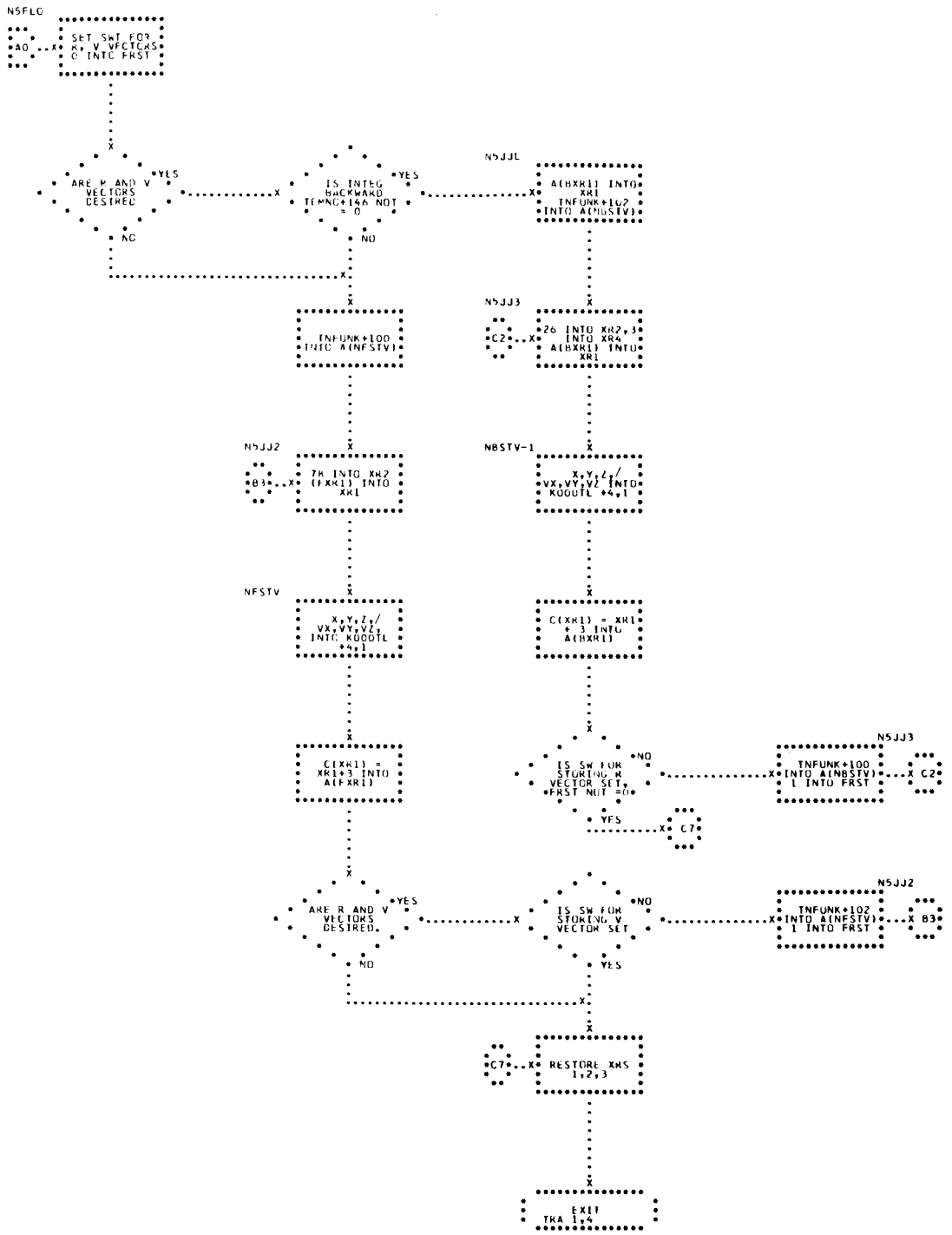
K00010 DEC 10

D) COMMON LOCATIONS -

TNFUNK (39 LOCATIONS).


START
N5CTBL
SAVE XR 1 XR2, XR4
VARIABLE OUTPUT STEP FOR D.C. MCIVSS NOT = 0
NO
YES
N5CST
IS STORING OF TABLE DESIRED KDONTS = 0
NO
YES
RESTORE XR1
EXIT TRA 1,4
$\bar{r}$ AND $\bar{v}$ VECTORS DESIRED KRORRV = 0
NO
YES
OUTPUT E VECTORS ONLY SET - DELTA I = 4 DELTA T
A5
N5FLO
IS TIME IN FLT PT UNITS KFLOFX NOT = 0
YES
NO
A0 934
OUTPUT $\bar{r}$ and $\bar{v}$ SET ΔI = 7 Δt
$\Delta t = t_1 - t_s$
COMP DIFF BETWEEN START TIME AND PRES.TIME
A5
α - { ΔT (MCVISS) + K000TL } -T + K00
COMPUTE LOC FOR STORING VECS IN TABLE
$\bar{r}$ and $\bar{v}$ VECS DESIRED KRORRV = 0
NO
YES
N5NDC
N5STR
DELTA I = 3 DELTA t
C7
N5CTI
STORE t_{CA} IN OUTPUT TABLE
C8
DELTA I = 6 DELTA t
D6
$\bar{v}$ VEC DESIRED KRORRV = 0
NO
YES
N5STR
COMP LOC FOR STORING VECTORS
C7
α = K000TL + 10 + ΔI / C(K00TL + 1)
STORE $\vec{v}$ VEC
XD INTO α + 3
YD INTO α + 4
ZD INTO α + 5
N5EXT
C8
INIT XR1 = 78 XR 2=6
RESTORE XR1 AND XR2
N5SLP
STORE TO VECTOR
X INTO α, XR2
Y INTO α+1, XR2
Z INTO α+2, XR2
D6
EXIT TRA 1,4


START
IS TEMNO + 146 =0
NO
YES
C0
X
N5EQT
N5TAB
SAVE XR4
COMP IPOS = MCVIOS+1
B3
MCVIOS INTO IPOS
COMP IPOS = MCVIOS-1
SAVE XR4
IS KRORV = 0
YES
NO
INIT 4 INTO XR4
B6
INIT 7 INTO XR4
C(XR4) INTO IFAC
CALC ADDR IN OUTPUT TABLE AC = IPOS (IFAC) + K0000TL
FIRST TIME THRU THIS POINT
SAVE VECTOR TWICE IN OUTPUT TABLE FOR = TIMES OF CLOSEST APPROACH
SAVE ADDR FOR STORING VECTOR
RESTORE XK4
EXIT TRA. 1,4


N5FLG
A0
SET SWT FOR R, V VECTORS 0 INTO FRST
ARE R AND V VECTORS DESIRED
YES
NO
IS INTEG BACKWARD TEMPC+146 NOT = 0
YES
NO
N5JJL
A(BXR1) INTO XR1
TNFUNK+102 INTO A(NBSTV)
N5JJ3
C2
26 INTO XR2,3 INTO XR4
A(BXR1) INTO XR1
TNFUNK+100 INTO A(NFSTV)
N5JJ2
B3
7R INTO XR2
(FXR1) INTO XR1
NBSTV-1
X,Y,Z,/ VX,VY,VZ INTO KOOUTL +4,1
NFSTV
X,Y,Z,/ VX,VY,VZ, INTO KOOUTL +4,1
C(XR1) = XR1 + 3 INTO A(BXR1)
C(XR1) = XR1+3 INTO A(FXR1)
IS SW FOR STORING R VECTOR SET, FRST NOT =0
NO
YES
N5JJ3
TNFUNK+100 INTO A(NBSTV) 1 INTO FRST
C2
C7
ARE R AND V VECTORS DESIRED
YES
NO
IS SW FOR STORING V VECTOR SET
NO
YES
N5JJ2
TNFUNK+102 INTO A(NFSTV) 1 INTO FRST
B3
C7
RESTORE XRS 1,2,3
EXIT
TRA 1,4

39. N5CAPI - NUMERICAL INTEGRATION CAPE VECTOR INITIALIZATION

N5CAPI INITIALIZES THOSE TESTS USED BY N5CAPS TO STORE CAPE KENNEDY VECTORS AND CORRECTS THE ANCHOR POINT ORBIT NUMBER IF NECESSARY. (SEE FLOWCHART)

INPUT

A)	K000.X	$\bar{r}_a$ -R VECTOR AT ANCHOR POINT
B)	K000.Y	
C)	K000.Z	
D)	K000TA t_a	TIME AT ANCHOR POINT
E)	K00DTB O_B	NUMBER AT CAPE VECTORS TO BE COMPUTED BACKWARD
F)	K00DTF O_F	NUMBER OF CAPE VECTORS TO BE COMPUTED FORWARD
G)	KCAPON O_A	ANCHOR POINT ORBIT NUMBER
H)	TEMNO + 146	DIRECTION INDICATOR. NON-ZERO FOR BACKWARD INTEGRATION, ZERO FOR FORWARD.
I)	KDONTS	WHEN ZERO, STORES TMCAPE TABLE AS DIRECTED
J)	KLCNGD	LONGITUDE TO INTEGRATE TO FOR A GMT.
K)	K000TL	CONTAINS ADDRESS OF OUTPUT TABLE.

OUTPUT

A)	TNCORK	CONTAINS THE ORBIT NUMBER OF THE NEXT VECTOR TO BE SAVED.
B)	NER19	ERROR IN COMPUTING LATITUDE, LONGITUDE, AND ALTITUDE FOR CAPE VECTORS.
C)	K000ST	STARTING TIME OF OUTPUT TABLE.

D) K0006ET ENDING TIME OF OUTPUT TABLE.

METHOD

N5CAPI USES SUB SPACECRAFT LONGITUDE COMPUTED IN A3MSCP TO DETERMINE WHETHER THE ANCHOR POINT IS EAST OR WEST OF THE CAPE, AND CORRECTS THE ANCHOR POINT ORBIT NUMBER IF NECESSARY. THIS PROGRAM ALSO INITIALIZES THE INDICATORS USED AS INPUT TO N5CAPS.

USAGE

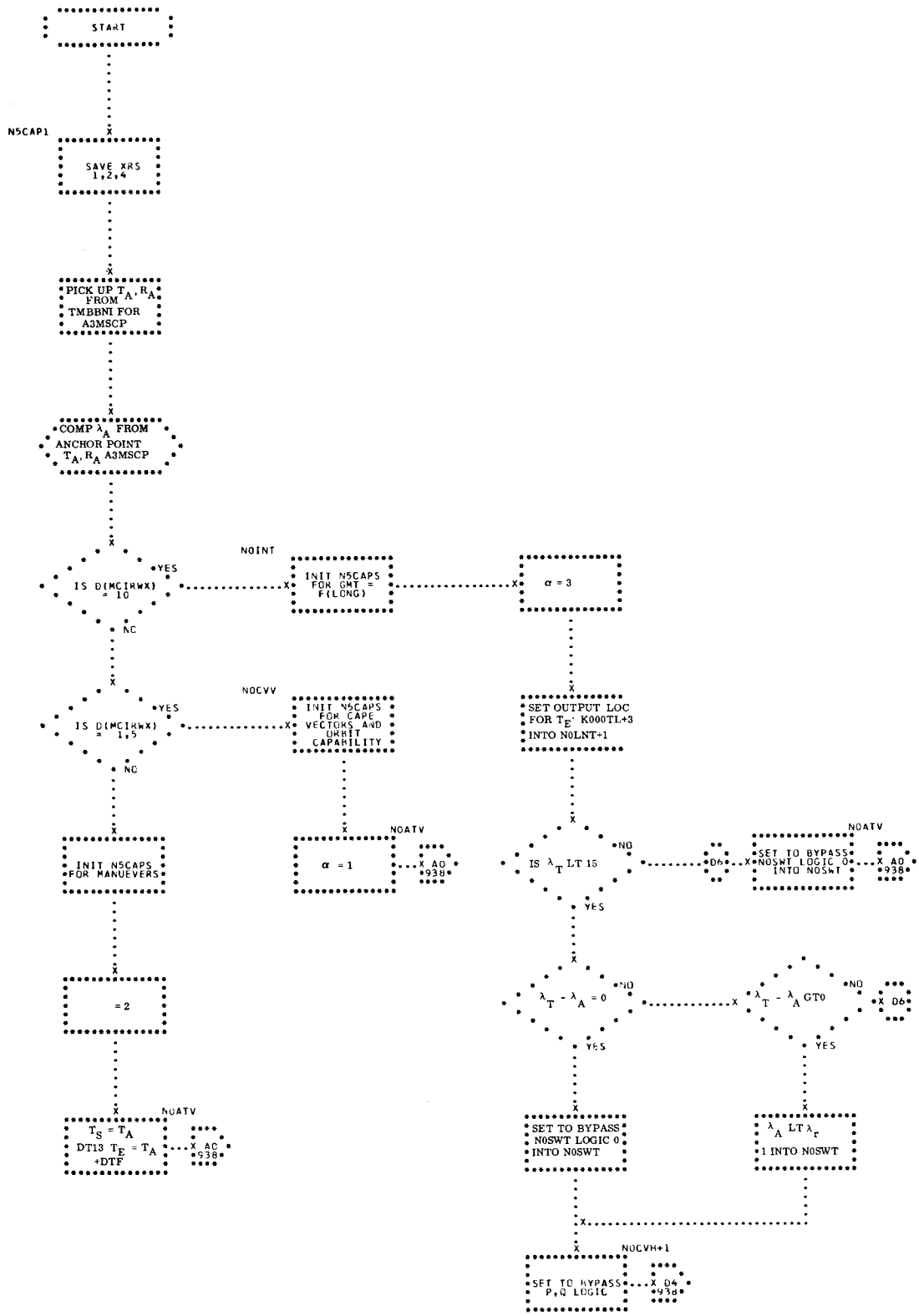
A) CALLING SEQUENCE -

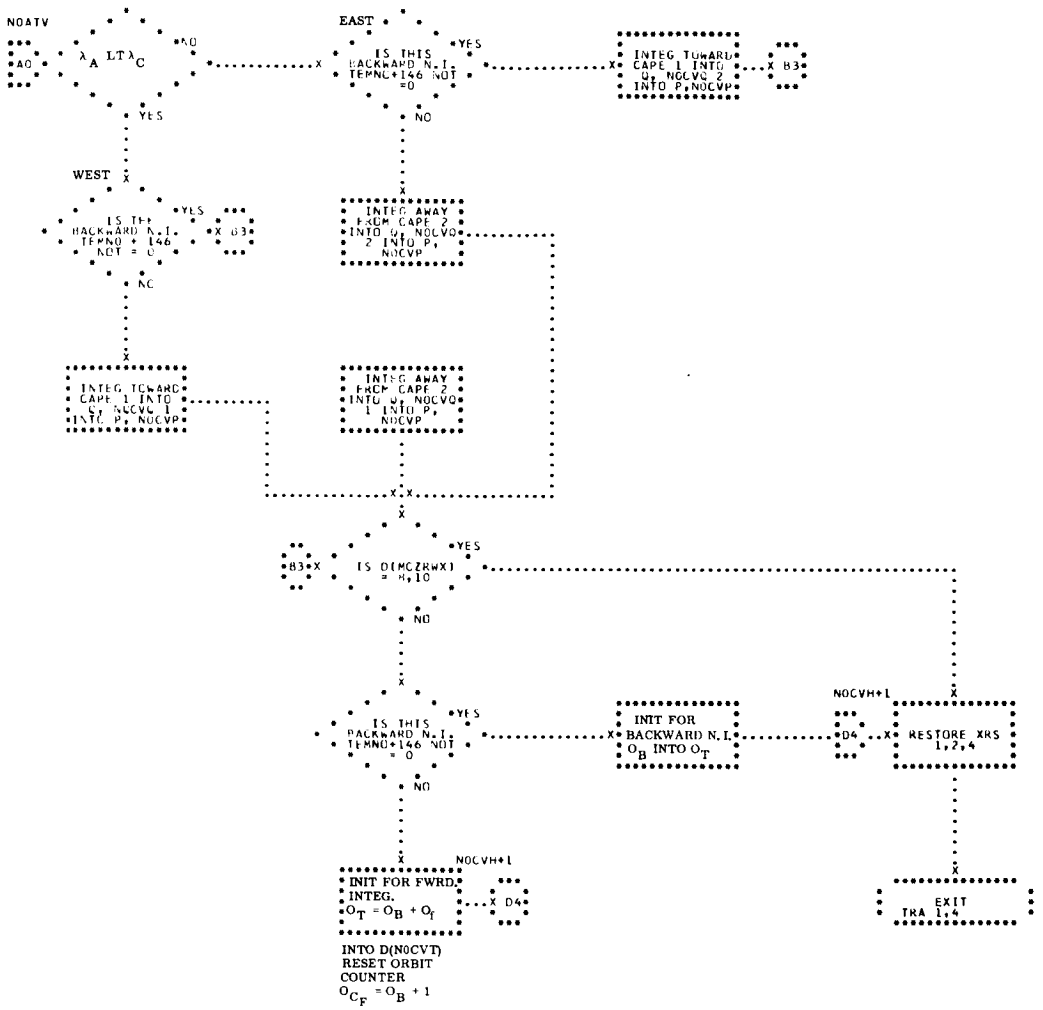
```
TSX     N5CAPI,4
        NORMAL RETURN
```

B) SPACE REQUIRED - 111 LOCATIONS

C) CONSTANTS -

```
K00001  DEC     1
K00002  DEC     2
KLCAPE  DEC     4.877148056     LONGITUDE OF CAPE
                                IN RADIANS
K015DG  DEC     .261799387      15 DEGREES IN
                                RADIANS
K00003  DEC     3
```


START
N5CAP1
SAVE XRS 1,2,4
PICK UP T_A, R_A FROM TMBBNI FOR A3MSCP
COMP λ_A FROM ANCHOR POINT T_A, R_A A3MSCP
IS D(MCIRWX) = 10
YES
NO
NOINT
INIT N5CAPS FOR GMT = F(LONG)
α = 3
IS D(MCIRWX) = 1,5
NOCVV
INIT N5CAPS FOR CAPE VECTORS AND ORBIT CAPABILITY
SET OUTPUT LOC FOR T_E K000TL+3 INTO N0LNT+1
INIT N5CAPS FOR MANUEVERS
α = 1
NOATV
X AO 938
IS λ_T LT 15
X D6
SET TO BYPASS N0SWT LOGIC 0 INTO N0SWT
= 2
$\lambda_T - \lambda_A = 0$
$\lambda_T - \lambda_A$ GT0
$T_S = T_A$ DT13 $T_E = T_A$ +DTF
SET TO BYPASS N0SWT LOGIC 0 INTO N0SWT
λ_A LT λ_T 1 INTO N0SWT
NOCVH+1
SET TO BYPASS P,Q LOGIC
X D4 938


NOATV
λ_A LT λ_C
A0
YES
WEST
IS THIS BACKWARD N.I. TERMNO+146 NOT = 0
B3
NO
INTEG TOWARD CAPE 1 INTO G, NOCVO 1 INTO P, NOCVP
INTEG AWAY FROM CAPE 2 INTO G, NOCVO 2 INTO P, NOCVP
EAST
IS THIS BACKWARD N.I. TERMNO+146 NOT =0
INTEG TOWARD CAPE 1 INTO G, NOCVO 2 INTO P, NOCVP
IS D(MCZRWX) = H,10
IS THIS BACKWARD N.I. TERMNO+146 NOT = 0
INIT FOR BACKWARD N.I. O_B INTO O_T
D4
NOCVH+1
RESTORE XRS 1,2,4
INIT FOR FWRD. INTEG. $O_T = O_B + O_f$
INTO D(N0CVT) RESET ORBIT COUNTER $O_{C_F} = O_B + 1$
EXIT TRA 1,4

40. N5CAPS - NUMERICAL INTEGRATION CAPE VECTOR STORING

N5CAPS TESTS THE CAPE GENERATED VECTORS, STORES THEM, AND TERMINATES NUMERICAL INTEGRATION WHEN THE LAST DESIRED CAPE VECTOR HAS BEEN STORED. (SEE FLOWCHART.)

INPUT

A)	TNFUNK + 22	$\dot{X}$	CURRENT $\vec{v}_i$
	+ 24	$\dot{Y}$	
	+ 26	$\dot{Z}$	
	+ 48	X	CURRENT $\vec{r}_i$
	+ 50	Y	
	+ 74	Z	
B)	KOOOTI - t_j		CURRENT TIME.
C)	KOCOTL -		CONTAINS ADDRESS OF START OF TMCAPE, AND OTHER OUTPUT TABLES.
D)	KOODTB - O_B		NUMBER OF CAPE VECTORS TO BE COMPUTED BACKWARD, FOR CAPE VECTORS OR ORBIT CAPABILITY, OTHERWISE LENGTH OF TIME TO INTEGRATE BACK.
E)	KOODTF - O_F		NUMBER OF CAPE VECTORS TO BE COMPUTED FORWARD, AS WITH KOODTB.
F)	KCAPON - O_A		ANCHOR POINT ORBIT NUMBER.
G)	TEMNO + 146		DIRECTION INDICATOR, NON-ZERO FOR BACKWARD INTEGRATION, ZERO FOR FORWARD.
H)	TNCORK		ORBIT NUMBER OF NEXT VECTOR TO BE SAVED.

I)	TNCORK + 1	ORBIT NUMBER OF NEXT VECTOR TO BE SAVED WHEN INTEGRATING BACKWARD DURING MANEUVERS.
J)	KLCAPE	DEC 4.877148056 LONG. OF CAPE IN RADIANS
K)	KDCNTS	PZE = 0 STORE CAPE VECTOR TABLE (TMCAPE) ≠ 0 DO NOT STORE CAPE VECTOR TABLE
L)	KOC001	DEC 1
M)	KOC007	DEC 7
N)	NER20	ERROR IN COMPUTING LAT., LONG., AND ALT. FOR CAPE VECTORS.
O)	KLCNGD	LONGITUDE FOR GMT
P)	KFLOFX	ZERO, TIME IS IN FIXED POINT UNITS, NON-ZERO, TIME IS IN FLOATING POINT UNITS.
Q)	KOOOTU	MINUTES OR SECONDS CODE = ZERO, TIME IS IN MINUTES = NON-ZERO, TIME IS IN SECONDS
R)	TRPEAT	= ZERO, N.I. HAS NOT BEEN RE-STARTED = NON-ZERO, RE-INITIALIZATION OF N.I. HAS OCCURRED.
S)	KO60.0	DEC 60.0
T)	KOC060	DEC 60
U)	KOO1.0	DEC 1.0

OUTPUT

THE OUTPUT TABLE (TMCAPE) USED FOR CAPE VECTORS AND ORBIT CAPABILITY HAS THE FOLLOWING FORMAT -

A)	DECREMENT	ORBIT NUMBER OF THE FIRST VECTOR IN THE TABLE.

B) ADDRESS — ORBIT NUMBER OF THE LAST VECTOR IN THE TABLE.

C) X, D) Y, E) Z — $\vec{r}$ VECTOR IN MERCURY UNITS

F) $\dot{X}$, G) $\dot{Y}$, H) $\dot{Z}$ — $\vec{v}$ VECTOR IN MERCURY UNITS

I) TIME IN MINUTES IN ADDRESS

J) ADDITIONAL VECTORS STORED AS IN PASTS C TO I .

THE OUTPUT TABLE FOR A GREENWICH MEAN TIME AS A FUNCTION OF LONGITUDE HAS THE FOLLOWING FORMAT -

A) INDICATION OF MINUTES OR SECONDS. ZERO FOR MINUTES, NON-ZERO FOR SECONDS.

B) INTEGRATION OUTPUT INTERVAL - NORMALLY ZERO

C) STARTING TIME-NOT NEEDED, MAY BE ZERO

D) ENDING TIME OR GMT FOR PRESENT LONGITUDE

E) X, F) Y, G) Z — $\vec{r}$ VECTOR IN GEMINI UNITS FOR PRESENT LONGITUDE

H) XD, I) YD, J) ZD — $\vec{v}$ VECTOR IN GEMINI UNITS FOR PRESENT LONGITUDE

THE OUTPUT TABLE FOR MANEUVERS HAS THE FOLLOWING FORMAT -

A) X, B) Y, C) Z — $\vec{r}$ VECTOR IN GEMINI UNITS

D) XD, E) YD, F) ZD — $\vec{v}$ VECTOR IN GEMINI UNITS

G) t_E — ENDING TIME

H) O_N ORBIT NUMBER OF PRESENT VECTOR

METHOD

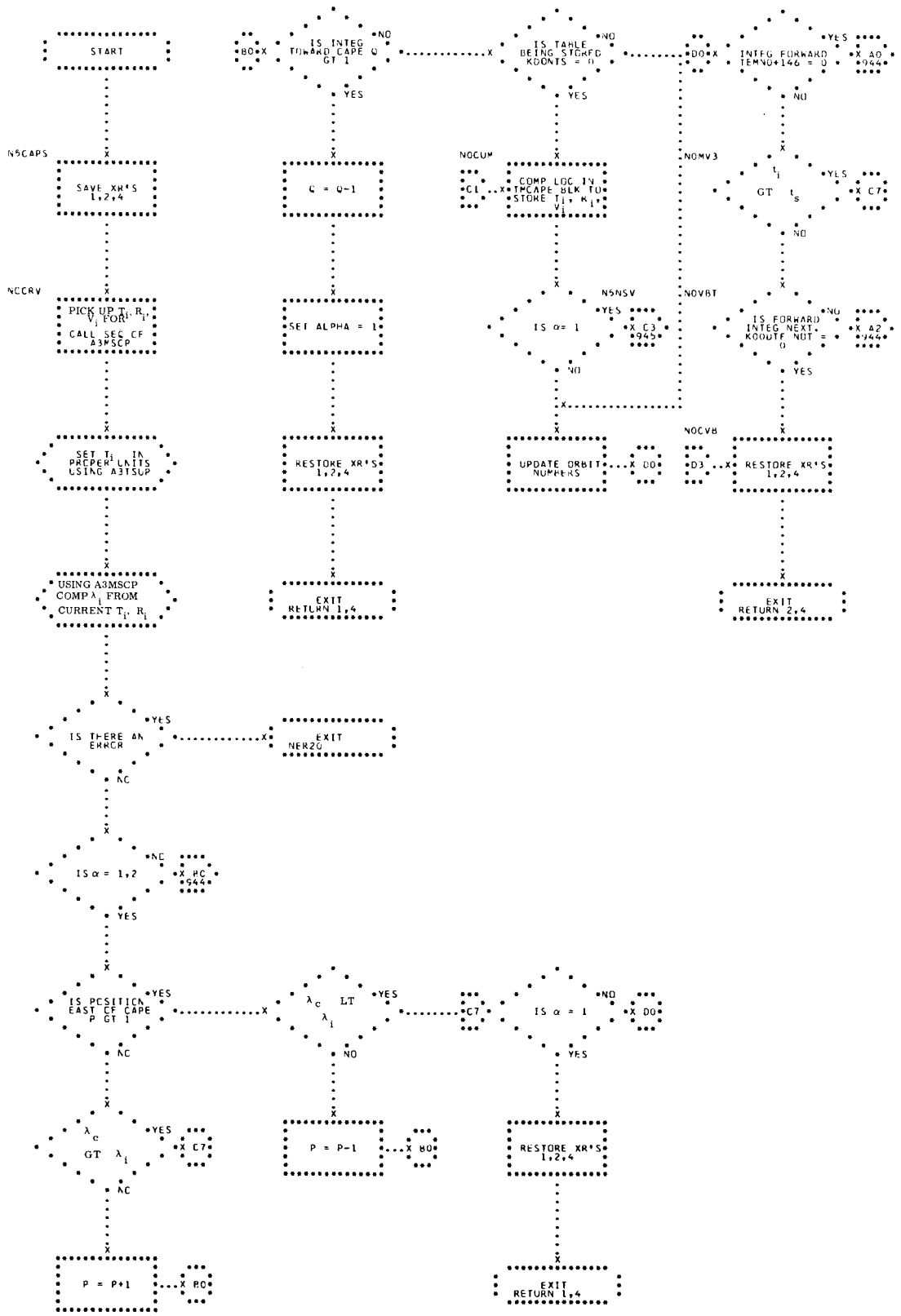
NCT APPLICABLE.

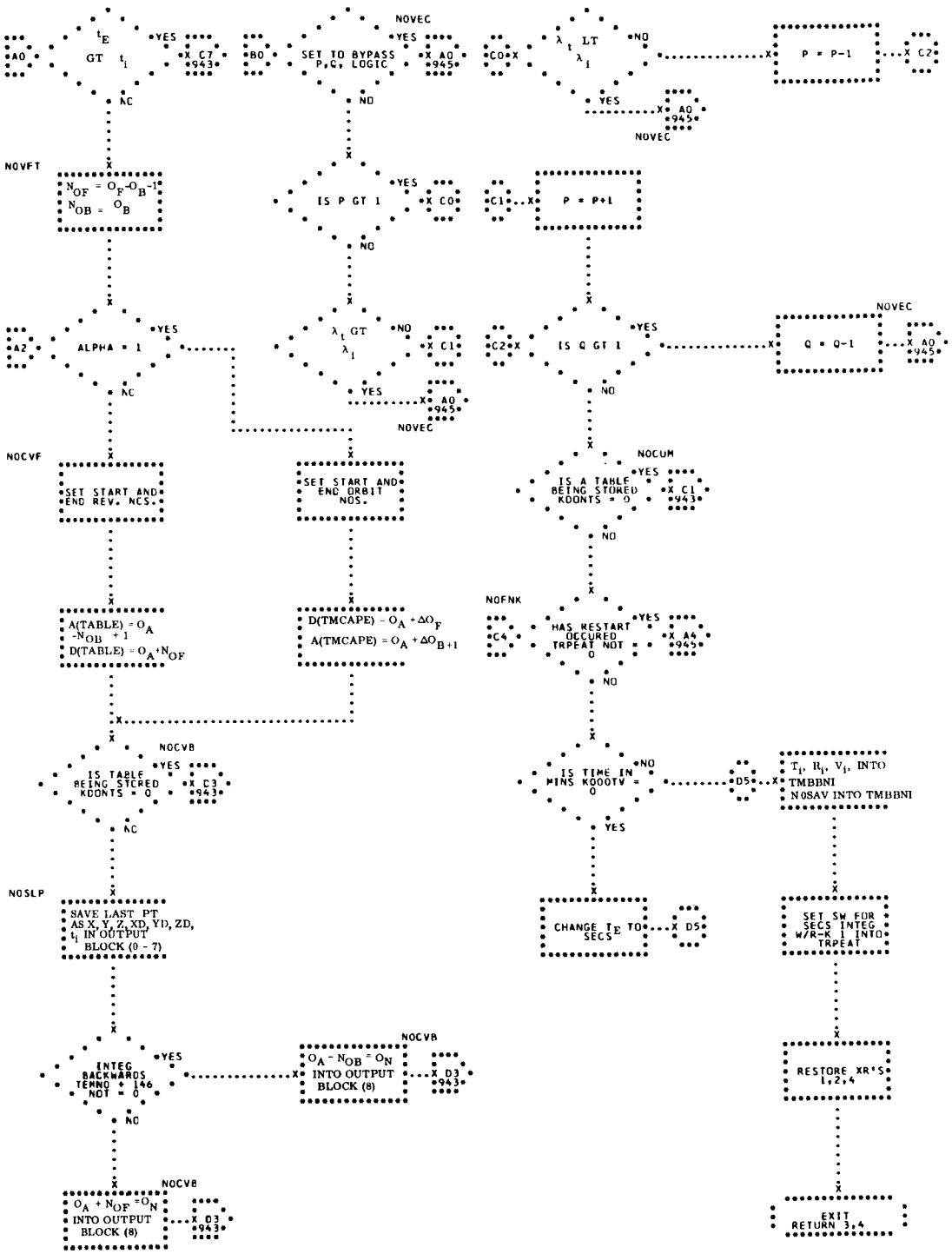
USAGE

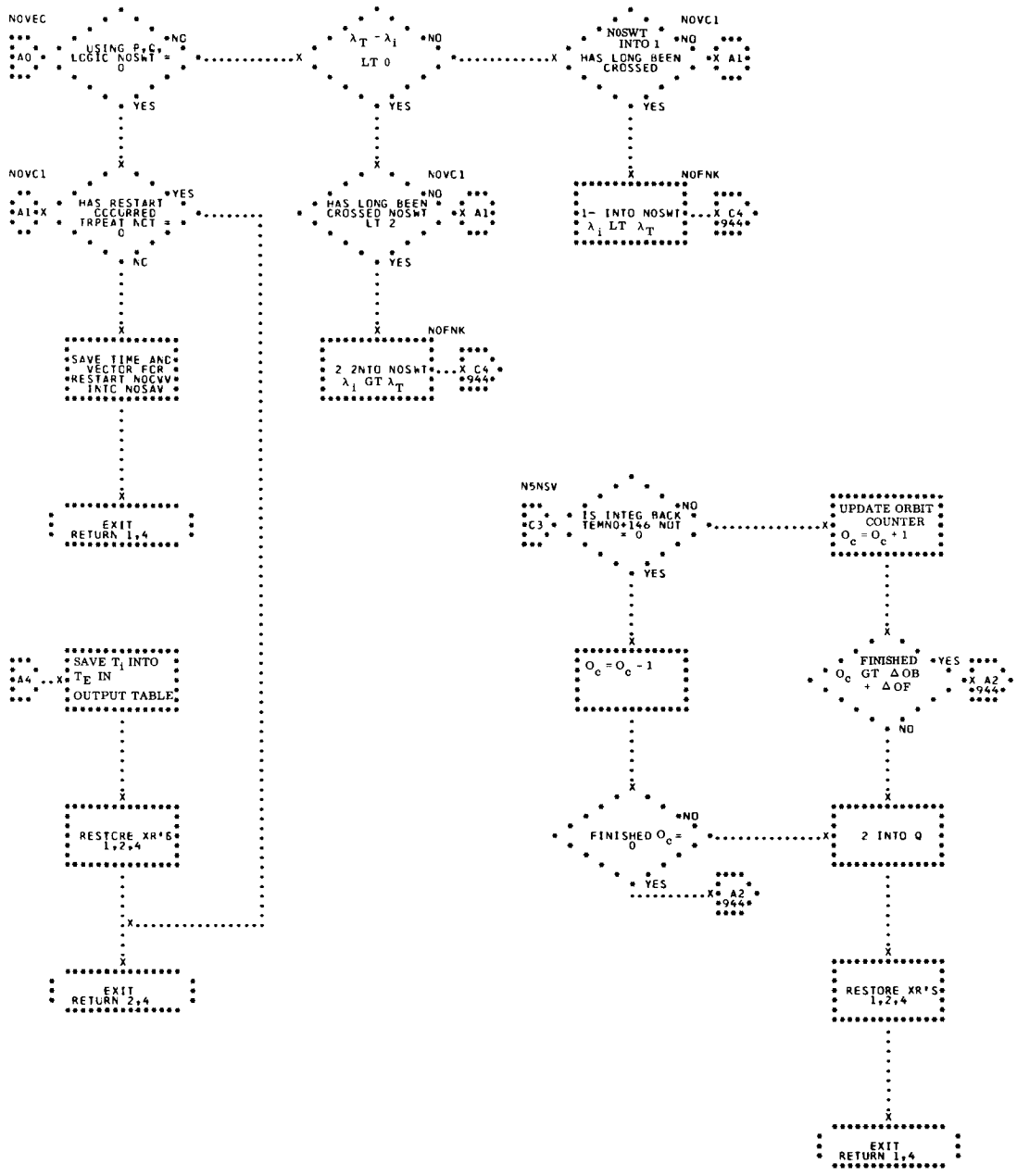
A) CALLING SEQUENCE -

```
        TSX     N5CAPS,4
  + 1           NORMAL RETURN CONTINUE INTE-
                GRATION.
  + 2           FINAL RETURN, INTEGRATION IN
                GIVEN DIRECTION IS COMPLETE.
  + 3           RESTART INTEGRATION PROCESS
```

B) SPACE REQUIRED - 228 LOCATIONS


START
IS INTEG FORWARD CAPE Q GT 1
NO
YES
B0
IS TABLE BEING STORED KDONTS = 0
NO
YES
D0
INTEG FORWARD TEMNO+146 = 0
YES
NO
X A0
X 944
N5CAPS
SAVE XR'S 1,2,4
Q = Q-1
NOCUM
C1
COMP LOC IN TMCAPE BLK TO STORE T_i, R_i, V_i
NOMV3
t_i GT t_s
YES
NO
X C7
NCCRV
PICK UP T_i, R_i, V_i FOR i CALL SEC OF A3MSCP
SET ALPHA = 1
N5NSV
IS α = 1
YES
NO
X C3
X 945
NOVBT
IS FORWARD INTEG NEXT, KOOUTF NOT = 0
NO
YES
X A2
X 946
SET T_i IN PREPER UNITS USING A3TSUP
RESTORE XR'S 1,2,4
UPDATE ORBIT NUMBERS
X D0
NOCVB
D3
RESTORE XR'S 1,2,4
USING A3MSCP COMP λ_i FROM CURRENT T_i, R_i
EXIT RETURN 1,4
EXIT RETURN 2,4
IS THERE AN ERROR
YES
NO
EXIT NER20
IS α = 1,2
NO
X B0
X 944
YES
IS POSITION EAST OF CAPE P GT 1
YES
NO
λ_c LT λ_i
YES
NO
C7
IS α = 1
NO
YES
X D0
λ_c GT λ_i
YES
NO
X C7
P = P-1
X B0
RESTORE XR'S 1,2,4
P = P+1
X B0
EXIT RETURN 1,4


A0
t_E GT t_i
YES
X C7 *943*
NO
B0
SET TO BYPASS P,Q, LOGIC
NOVEC
X A0 *945*
C0
λ_t LT λ_i
NO
P = P-1
X C2
YES
X A0 *945*
NOVEC
NOVFT
$N_{OF} = O_F - O_B - 1$
$N_{OB} = O_B$
IS P GT 1
YES
X C0
NO
C1
P = P+1
A2
ALPHA = 1
YES
NO
λ_t GT λ_i
NO
X C1
YES
X A0 *945*
NOVEC
C2
IS Q GT 1
YES
Q = Q-1
NOVEC
X A0 *945*
NO
NOCVF
SET START AND END REV. NOS.
SET START AND END ORBIT NOS.
NOCUM
IS A TABLE BEING STORED KOONTS = 0
YES
X C1 *943*
NO
A(TABLE) = $O_A - N_{OB} + 1$
D(TABLE) = $O_A + N_{OF}$
D(TMCAPE) = $O_A + \Delta O_F$
A(TMCAPE) = $O_A + \Delta O_{B+1}$
NOFNK
C4
HAS RESTART OCCURED TRPEAT NOT = 0
YES
X A4 *945*
NO
NOCVB
IS TABLE BEING STORED KOONTS = 0
YES
X C3 *943*
NO
IS TIME IN MINS KOOOTV = 0
NO
D5
YES
T_i, R_i, V_i, INTO TMBBNI
NOSAV INTO TMBBNI
NOSLP
SAVE LAST PT AS X, Y, Z, XD, YD, ZD, t_i IN OUTPUT BLOCK (0 - 7)
CHANGE T_E TO SECS
X D5
SET SW FOR SECS INTEG W/R-K 1 INTO TRPEAT
INTEG BACKWARDS TEMNO + 146 NOT = 0
YES
NO
NOCVB
$O_A - N_{OB} = O_N$ INTO OUTPUT BLOCK (8)
X D3 *943*
RESTORE XR'S 1,2,4
NOCVB
$O_A + N_{OF} = O_N$ INTO OUTPUT BLOCK (8)
X D3 *943*
EXIT RETURN 3,4


NOVEC
A0
USING P.C. LOGIC NOSWT = 0
NO
YES
$\lambda_T - \lambda_i$ LT 0
NO
YES
NOSWT INTO 1 HAS LONG BEEN CROSSED
NOVC1
NO
A1
YES
NOVC1
A1
HAS RESTART OCCURRED TRPEAT ACT = 0
YES
NO
HAS LONG BEEN CROSSED NOSWT LT 2
NOVC1
NO
A1
YES
NOFNK
1- INTO NOSWT λ_i LT λ_T
NOFNK
C4
944
NOFNK
2 2NTO NOSWT λ_i GT λ_T
C4
944
SAVE TIME AND VECTOR FOR RESTART NOCVV INTC NOSAV
EXIT RETURN 1,4
A4
SAVE T_i INTO T_E IN OUTPUT TABLE
RESTORE XR'S 1,2,4
EXIT RETURN 2,4
N5NSV
C3
IS INTEG BACK TEMNO+146 NOT = 0
NO
YES
UPDATE ORBIT COUNTER $O_c = O_c + 1$
$O_c = O_c - 1$
FINISHED O_c = 0
NO
YES
A2
944
FINISHED O_c GT $\Delta OB + \Delta OF$
YES
NO
A2
944
2 INTO Q
RESTORE XR'S 1,2,4
EXIT RETURN 1,4

41. N60PEP - EARTH'S EFFECT ON SPACECRAFT CALCULATION

N60PEP CALCULATES THE FORCE ON THE SPACECRAFT DUE TO EARTH'S POTENTIAL. (SEE FLOWCHART).

INPUT

THE SQUARE ROOT LIBRARY SUBROUTINE, U1SQRT, IS USED WITH THIS PROGRAM. THE FOLLOWING INFORMATION FROM THE TNFUNK TABLE SERVES AS INPUT TO N60PEP -

INPUT	SYMBOLIC LOCATION	
X	TNFUNK + 22	
Y	TNFUNK + 48	FLOATING POINT
Z	TNFUNK + 74	

OUTPUT

$\frac{\partial P}{\partial X}$	K00.XP	FLOATING POINT IN GEMINI UNITS
$\frac{\partial P}{\partial Y}$	K00.YP	
$\frac{\partial P}{\partial Z}$	K00.ZP	

METHOD

THE FOLLOWING EQUATIONS ARE USED BY THE N60PEP PROGRAM -

$$-\frac{\partial P}{X} = \frac{X}{r^3} - \frac{JX}{r^5}\left(1-\frac{5Z^2}{r^2}\right) - \frac{HXZ}{r^7}\left(3-\frac{7Z^2}{r^2}\right)$$

$$-\frac{DX}{7r^7}\left(3-\frac{42Z^2}{r^2}+\frac{63Z^4}{r^4}\right)$$

$$-\frac{\partial P}{\partial Y} = \frac{Y}{r^3} - \frac{JY}{r^5}\left(1 - \frac{5Z^2}{r^2}\right) - \frac{HYZ}{r^7}\left(3 - \frac{7Z^2}{r^2}\right) - \frac{DY}{7r^7}\left(3 - \frac{42Z^2}{r^2} + \frac{63Z^4}{r^4}\right)$$

$$-\frac{\partial P}{\partial Z} = \frac{Z}{r^3} - \frac{JZ}{r^5}\left(3 - \frac{5Z^2}{r^2}\right) - \frac{H}{r^5}\left[\left(6 - \frac{7Z^2}{r^2}\right)\frac{Z^2}{r^2} - \frac{3}{5}\right] - \frac{DZ}{7r^7}\left(15 - \frac{70Z^2}{r^2} + \frac{63Z^4}{r^4}\right)$$

$$r = \sqrt{X^2 + Y^2 + Z^2}$$

USAGE

A) CALLING SEQUENCE -

```
TSX     N60PEP,4
        NORMAL RETURN
```

(ERROR RETURN IN U1SQRT LIBRARY SUBROUTINE IS HANDLED BY MONITOR. RETURN IS A LOCATION IN MONITOR.)

B) SPACE REQUIRED - 64 LOCATIONS (MAIN PROGRAM) PLUS U1SQRT AND 10 TEMPORARY LOCATIONS DESIGNATED BY TEMN3.

C) CONSTANTS -

```
K0C000  0
K000.6  0.6
K0001.0 1.0
K002.0  2.0
```

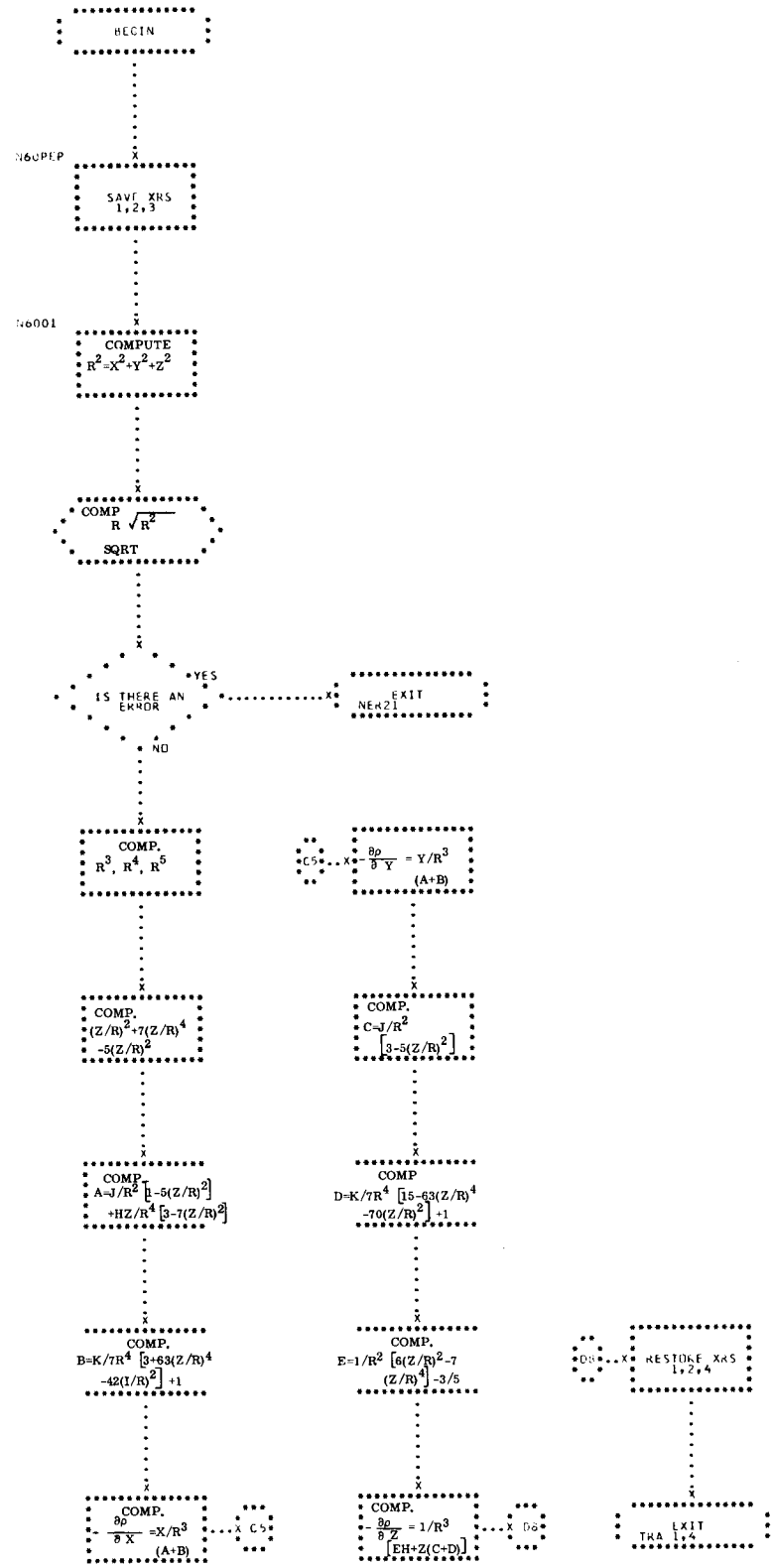
```
        K003.0   3.0
        K005.0   5.0
        K000J.   162345.E-8
        K000H.   -.575E-5
        K000K.   .7875E-5
        K006.0   6.0
        K009.0   9.0
```

D) TABLES USED -

```
        TNFUNK   80 LOCATIONS
        TEMN 3   10 LOCATIONS
```

E) TIMING - 5.78 MILLISECONDS + T_{SQRT} = AVERAGE TIME SYN CARDS FOR N60PEP

```
        K00.XP   SYN     TNINT
        K00.YP   SYN     TNINT + 1
        K00.ZP   SYN     TNIINT + 2
```


BEGIN
N60PEP
SAVE XRS 1,2,3
N6001
COMPUTE $R^2=X^2+Y^2+Z^2$
COMP $R\sqrt{R^2}$ SQRT
IS THERE AN ERROR
YES
EXIT NER21
NO
COMP. R^3, R^4, R^5
C5
$-\frac{\partial\rho}{\partial Y} = Y/R^3$ (A+B)
COMP. $(Z/R)^2+7(Z/R)^4 -5(Z/R)^2$
COMP. $C=J/R^2\left[3-5(Z/R)^2\right]$
COMP $A=J/R^2\left[1-5(Z/R)^2\right] +HZ/R^4\left[3-7(Z/R)^2\right]$
COMP $D=K/7R^4\left[15-63(Z/R)^4 -70(Z/R)^2\right]+1$
COMP. $B=K/7R^4\left[3+63(Z/R)^4 -42(I/R)^2\right]+1$
COMP. $E=1/R^2\left[6(Z/R)^2-7 (Z/R)^4\right]-3/5$
D8
RESTORE XRS 1,2,4
COMP. $-\frac{\partial\rho}{\partial X} = X/R^3$ (A+B)
C5
COMP. $-\frac{\partial\rho}{\partial Z} = 1/R^3 \left[EH+Z(C+D)\right]$
D8
EXIT THA 1,4

42. N7VARS - VARIABLE STEP INTEGRATION EQUATIONS

N7VARS DECREASES THE STEP TIME OF INTEGRATION BY ONE-HALF AND CALCULATES F VALUES FOR THE HALF-STEP. (SEE FLOWCHART)

INPUT

A) KOOOTI - PRESENT TIME OF INTEGRATION, IF TN7HSI IS $\neq$ 0, TIME IS TO THE NEAREST UNIT.

B) KOOOOH - INTEGRATION TIME STEP. IF TN7HSI IS $\neq$ 0, TIME STEP IS TO THE NEAREST UNIT.

C) KOOO.H - TIME STEP IN GEMINI UNITS.

D) KOO.2H - $(KOOO.H)^2$ IN GEMINI UNITS.

E) KINTVN - NUMBER OF STEPS PER OUTPUT.

F) TNFUNK - FUNCTION TABLE (SEE TABLE TNFUNK)

G) TN7N12 - INDICATOR = 0, ENTRANCE FROM N1SUFT, THEREFORE, HALF-STEP F VALUES ARE NOT CALCULATED. INDICATOR $\neq$ 0, ENTRANCE FROM N2EXCR.

H) TNSAVR - LAST $\overline{r}$ AND $\overline{v}$ WERE SAVED IN THESE LOCATIONS (F).

I) TN7ICT - COUNTER TO DETERMINE THE NUMBER OF TIMES THE TIME STEP WAS DECREASED FOR CONVERGENCE IN A PARTICULAR INTERVAL.

J) TN7HSI - INDICATOR = 0, TIME STEP IS A UNIT NUMBER. INDICATOR $\neq$ 0, TIME STEP IS LESS THAN UNITY AND IS EXPRESSED IN FLOATING-POINT UNITS.

K) TNIFPT - PRESENT TIME OF INTEGRATION IN FLOATING-POINT UNITS.

L) TEMNO + 146 - INDICATOR = 0, FORWARD INTEGRATION IN PROCESS. INDICATOR $\neq$ 0, BACKWARD INTEGRATION IN PROCESS.

M) KFLOFX - INDICATOR = 0, FIXED POINT INTEGRATION IN PROCESS. INDICATOR $\neq$ 0, FLOATING POINT INTEGRATION IN PROCESS.

OUTPUT

A) KOOOTI - FORMER T_i ADJUSTED FOR DECREASE IN TIME STEP.

TABLE. TNFUNK TABLE FORMAT

LOCATION				CONTENTS
TNFUNK	+0	+26	+52	F_i/h^2 MSH
	+1	+27	+53	LSH
	+2	+28	+54	$F_{i\pm1}$ MSH
	+3	+29	+55	LSH
	+4	+30	+56	$F_{i\pm2}$ MSH
	+5	+31	+57	LSH
	+6	+32	+58	$F_{i\pm3}$ MSH
	+7	+33	+59	LSH
	+8	+34	+60	$F_{i\pm4}$ MSH
	+9	+35	+61	LSH
	+10	+36	+62	$F_{i\pm5}$ MSH
	+11	+37	+63	LSH
	+12	+38	+64	$F_{i\pm6}$ MSH
	+13	+39	+65	LSH
	+14	+40	+66	"$F_{i\pm1}$ MSH
	+15	+41	+67	LSH
	+16	+42	+68	'$F_{i\pm3/2}$ MSH
	+17	+43	+69	LSH
	+18	+44	+70	'$F_{i\pm1/2}$ MSH
	+19	+45	+71	LSH
	+20	+46	+72	"F_i MSH
	+21	+47	+73	LSH
	+22	+48	+74	X_i MSH

+23	+49	+75		LSH
+24	+50	+76	X_i	MSH
+25	+51	+77		LSH

B) K0000H
C) K000.H
D) K00.2H
ALL VALUES ASSOCIATED WITH TIME STEP DIVIDED BY 2.

E) KINTVN - NUMBER OF STEPS PER OUTPUT INCREASED.

F) TNFUNK - IF ENTRANCE IS MADE FROM N2EXCR A RE-EVALUATED FUNCTION TABLE WITH HALF-LINE VALUE IS CALCULATED.

G) TEMNO + 156 — MSH
h
TEMNO + 157 — LSH

H) TEMNO + 138 — h MSH
TEMNO + 139 — LSH

METHOD

THE FOLLOWING EQUATIONS ARE USED BY THE N7VARS PROGRAM -

$$F_{i \pm 1/2} = \sum_{k=1}^{6} T_R F_{i-k}$$ SPECIALIZED DEVIATION (1)

$$F_{i \pm 3/2} = \sum_{k=1}^{6} T'_R F_{i-k}$$ OF LAGRANGIAN SIX POINT (2)

$$F_{i \pm 5/2} = \sum_{k=1}^{6} T''_R F_{i-k}$$ INTERPOLATION (3)

$$''F_i = R_i - \sum_{k=1}^{6} A_k F_{i \pm k} \qquad \text{(FORWARD OR BACKWARD)} \qquad (4)$$

$$'F_{i-1/2} = (h) V_i - \sum_{k=1}^{6} A'_i F_{i-k} \qquad \text{(FORWARD)} \qquad \text{FROM SUBSECTION 27} \qquad (5)$$

$$'F_{i \pm 1/2} = (h) V_i + \sum_{k=1}^{6} A'_k F_{i+k} \qquad \text{(BACKWARD)} \qquad (6)$$

USAGE

A) CALLING SEQUENCE -

```
TSX     N7VARS,4
        NORMAL RETURN
```

B) ERROR CODES - PROGRAM TRANSFERS TO ERROR RETURN IF INTERVAL IS REDUCED TOO SMALL (TN7ISI LESS THAN KOOHSI) OR IF THE NUMBER OF STEPS PER OUTPUT (KINTVN) EXCEEDS DECREMENT STORAGE (20 BITS).

C) SPACE REQUIRED - 150 LOCATIONS + 10 TEMPORARY LOCATIONS DESIGNATED BY TEMN7.

D) SPECIAL USAGE - THE FOLLOWING ARE THREE POSSIBLE CASES OF NONCONVERGENCE WHERE N7VARS WOULD BE USED -

1) N1SUFT WHILE SETTING UP FUNCTION TABLE (1) TN7N12 = 0 AS INPUT TO N7VARS, (2) C(AC) = 0 ON RETURN FROM N7VARS.

2) N2EXCR WHILE BEING USED TO SET UP FUNCTION TABLE (N1SUFT) (1) TN7N12 = 0 AS INPUT, (2)

C(AC) = 0 ON RETURN FROM N7VARS.

3) N2EXCR WHILE BEING USED TO UPDATE ALREADY ESTABLISHED FUNCTION TABLE (1) TN7N12 ≠ 0 AS INPUT, (2) C(AC) ≠ 0.

IN CASES (1) AND (2), N7VARS WILL ONLY CHANGE INTERVAL AND INITIALIZE CONSTANTS DEPENDING ON THE TIME STEP.

E) CONSTANTS -

```
KOZERO          DEC     0
KO02.0          DEC     2.0
KDP4.0          DEC     4.0     (DOUBLE PRECISION)
KOC003          DEC     3.0
KO00.5          DEC     0.5
KCH233          OCT     233000000000
KO0.A1...6

KO.A.1P...6P
TNVS.C          DEC     .24609375   \
TNVS.C +1       DEC     -.02734375  |
TNVS.C +2       DEC     .01171875   |  T'R, R = 0,5
TNVS.C +3       DEC     1.23046875  |
TNVS.C +4       DEC     .41015625   |
TNVS.C +5       DEC     -.09765625  /
TNVSC. +6       DEC     -.8203125   \
TNVS.C +7       DEC     .8203125    |  T'R, R = Q,5
TNVS.C +8       DEC     .5859375    |
TNVS.C +9       DEC     .4921875    /
TNVS.C +10      DEC     -.2734375   \
TNVS.C +11      DEC     .5859375    |  T'R, R = 0,5
TNVS.C +12      DEC     -.17578125  |
TNVS.C +13      DEC     .08203125   /
TNVS.C +14      DEC     -.09765625      T''R R = 0,5
```

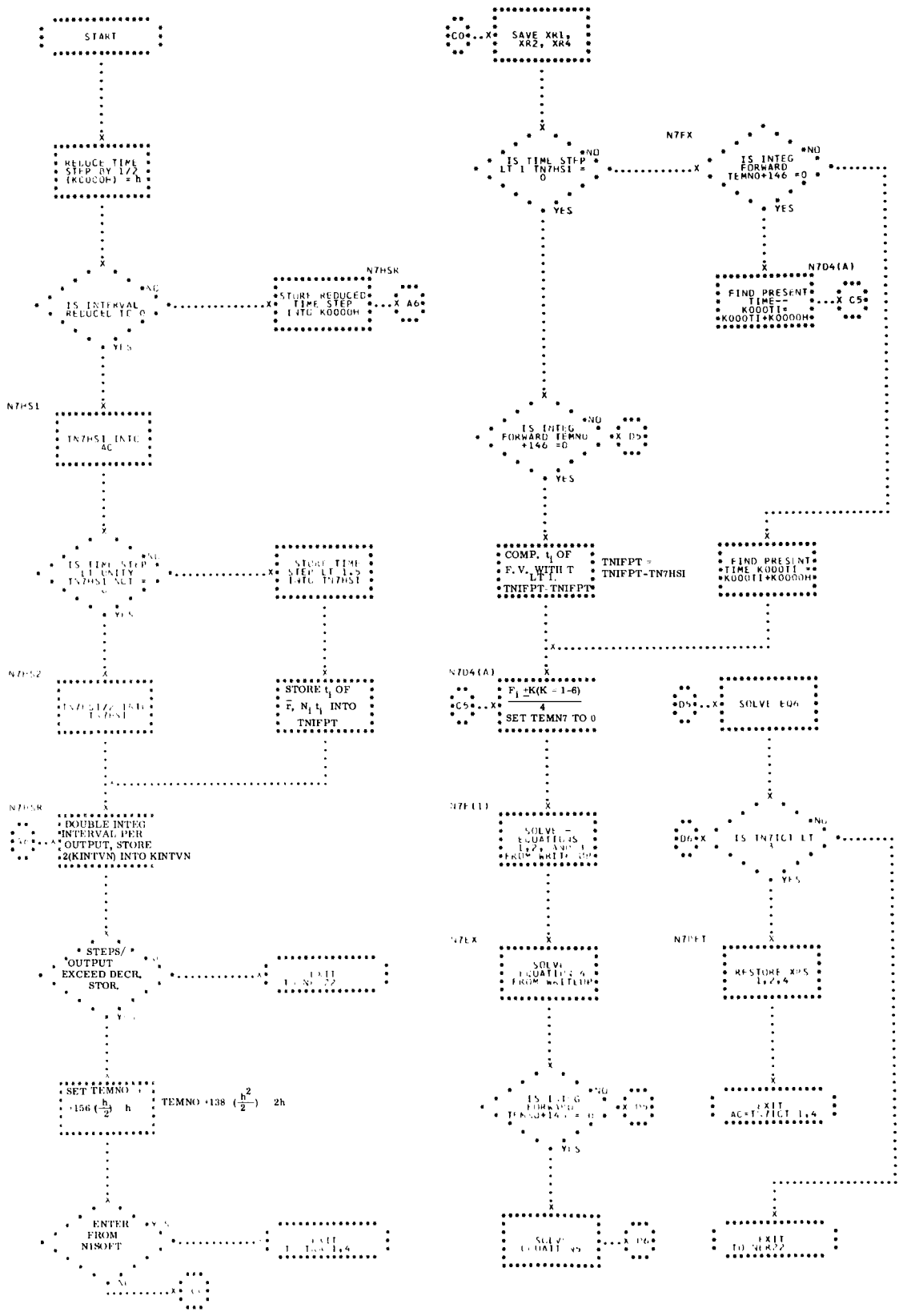
TNVS.C +15	DEC	.02734375	$'T''_R$ R = 0,5
TNVS.C +16	DEC	-.01171875	
TNVS.C +17	DEC	.01171875	
TEMN7	BSS	10	INTERMEDIATE TABLE FOR N7VARS
NER22		ERROR RETURN, NONCONVERGENCE 122 → XR(4)	

F) TIME REQUIRED -

ENTRANCE FROM N1SUFT

K0000H ≥ 1	59 CYCLES	(7094)
K0000H < 1	79 CYCLES	(7094)

ENTRANCE FROM N2EXCR


START
REDUCE TIME STEP BY 1/2 (KCOCOH) = h
IS INTERVAL REDUCED TO 0
NO
YES
STORE REDUCED TIME STEP INTO K0000H
N7HSR
A6
N7HS1
TN7HS1 INTO AC
IS TIME STEP LT UNITY
STORE TIME STEP LT 1.0 INTO TN7HST
N7HS2
STORE t_i OF $\bar{r}$, N_i t_i INTO TNIFPT
N7HSR
DOUBLE INTEG INTERVAL PER OUTPUT, STORE 2(KINTVN) INTO KINTVN
STEPS/ OUTPUT EXCEED DECR. STOR.
EXIT TO SCR22
SET TEMNO +156 $(\frac{h}{2})$ h
TEMNO +138 $(\frac{h^2}{2})$ 2h
ENTER FROM NISOFT
EXIT TO SCR22
C0
SAVE XK1, XR2, XR4
IS TIME STEP LT 1 TN7HS1 = 0
N7FX
IS INTEG FORWARD TEMNO+146 =0
N7D4(A)
FIND PRESENT TIME=- K0001I+K0000H
C5
IS INTEG FORWARD TEMNO +146 =0
D5
COMP. t_i OF F.V. WITH T LT 1. TNIFPT-TNIFPT
TNIFPT = TNIFPT-TN7HS1
FIND PRESENT TIME K0001I+K0000H
N7D4(A)
C5
$F_i \pm K(K-1-6)$ / 4
SET TEMN7 TO 0
D5
SOLVE EQ6
N7FLT
SOLVE EQUATIONS FROM K0001I
D6
IS IN7ICT LT
N7FEX
SOLVE EQUATIONS FROM K0001I
N7FET
RESTORE XPS 1,2,4
IS INTEG FORWARD TEMNO+146 = 0
EXIT AC=IN7ICT 1,4
SOLVE EQUATIONS
EXIT TO SCR22

43. N7PCAL - ON-BOARD COMPUTER EQUATIONS

PROGRAM N7PCAL SIMULATES THE ON-BOARD COMPUTER'S COMPUTATION OF BANK ANGLES WHICH DIRECT THE CAPSULE TO IMPACT AT THE TARGET. IT IS A SUBROUTINE OF THE INTEGRATOR. (SEE FLOWCHART.)

INPUT

KLATTT — TARGET LATITUDE

KLONGT — TARGET LONGITUDE

A) SUBROUTINES -

SQRT, ARSIN, ARCOS,
SIN, COS, ATNAB, LN

B) CONSTANTS -

K001.0	KFTSC2	KCFTSC	K.1RAD	K00001
K004.0	K002.0	K060.0	KGUFTT	
K015.0	K010.0	K003.0	K00019	
K.OMEG	K001.2	K005.0	KCH233	

C) TABLES -

TNFUNK

+22	X
+24	$\dot{X}$
+48	Y
+50	$\dot{Y}$
+74	Z
+76	$\dot{Z}$

TDVALS — TABLE OF GEMINI RANGE PREDICTION CONSTANTS

TCRDRE — CROSSRANGE ERROR IN N. M.

+1 — DOWNRANGE ERROR IN N. M.

TEMNO

+163 ≠ 0 MEANS FIRST RK APPROXIMATION

TN4LLH

+2 SPACECRAFT HEIGHT

D) CONDITICNS CN ENTRY -

NO PREREQUISITES

OUTPUT

COMPUTED BANK ANGLES ARE OUTPUTTED TO THE INTEGRATOR DURING THE SECONC AND THIRD RK APPROXIMATION.

METHCD

THE PRCGRAM IS CONTROLLED BY MYREN2. THE BURNOUT VECTCR IS INTEGRATED DOWN IN A 6-SECOND STEP SIZE UNTIL AN ALTITUDE OF 410K FEET IS REACHED. THE STEP SIZE IS THEN CHANGED TO 1.2 SECONDS IN ORDER TO MORE CLOSELY SIMULATE THE ACTUAL CYCLE OF THE ON BOARD COMPUTER. INTEGRATION CONTINUES, WITH A 1.2-SECOND STEP SIZE, UNTIL AN ALTITUDE CF 10K FEET IS REACHED. ACTUAL GUIDANCE IS INITIATED AT APPRCXIMATELY 305K FEET AT WHICH POINT THE AVERAGE SENSED ACCELERATION REACHES .4 FT/SEC^2 AND THE SPACECRAFT THUS ENCOUNTERS SUFFICIENT DYNAMIC PRESSURE TO ALLCW FCR TRAJECTORY CONTROL. BANK ANGLE COMMANDS ARE THEN GENERATED TC ALTER THE TRAJECTORY SUCH THAT THE VEHICLE WILL IMPACT AT THE TARGET.

EQUATICNS -

TRANSFCRMATION OF RECTANGULAR INERTIAL COORDINATES ($X, Y, Z, \dot{X}, \dot{Y}, \dot{Z}$) TC RADIAL DISTANCE (R), RELATIVE LONGITUDE λ RELATIVE HEADING FROM NORTH (β) RELATIVE VELOCITY (V) RELATIVE FLIGHT PATH ANGLE Υ LATITUDE ϕ EARTH RADIUS (R_E) HEIGHT (HT) -

1) $R = (X^2 + Y^2 + Z^2)^{1/2}$ ft.

2) $\lambda = \tan^{-1} \frac{Y}{X} - \Omega (t-t_{GHA}) - GHA$ deg.

3) $\beta = \tan^{-1} \frac{\dot{X}(\dot{Y}-\Omega X) - Y(\dot{X}+\Omega Y)}{RZ - ZV \sin}$ deg.

4) $V = ((\dot{X} + \Omega Y)^2 + (\dot{Y} - \Omega X)^2 + \dot{Z}^2)^{1/2}$ ft/sec

5) $\gamma = \sin^{-1} \frac{X\dot{X} + Y\dot{Y} + Z\dot{Z}}{RV}$ deg.

6) $\phi = \sin^{-1} \frac{Z}{R}$ deg.

7) $R_E = R_Q R_P / (R_P^2 \cos^2 \phi + R_Q^2 \sin^2 \phi)^{1/2}$ ft.

8) $HT = R - R_E$

WHERE -

Ω = EARTH ROTATION RATE
t_{GHA} = GREENWICH HOUR ANGLE AT t_{GHA}
GHA = GREENWICH HOUR ANGLE AT t_{GHA}
R_Q = EARTH EQUATORIAL RADIUS
R_P = POLAR EARTH RADIUS

CALCULATION OF DOWNRANGE ERROR (RN), CROSS RANGE ERROR (RC), PREDICTED BALLISTIC RANGE OF THE CAPSULE (RB)

1) $B = \cos \phi_T \sin(\lambda_T - \lambda)$

2) $E = \sin \phi_T \cos \phi - \cos \phi_T \sin \phi \cos(\lambda_T - \lambda)$

3) $RT = 60 \sin^{-1}(B^2 + E^2)^{1/2}$ N.M.

4) $\beta_T = \text{TAN}^{-1} \frac{B}{E}$

5) RC = RT SIN $(\beta - \beta_T)$ N.M.

6) RN = RT COS $(\beta - \beta_T)$ N.M.

7) D $= -.43429448$ LN XSA/V^2

HAVING CALCULATED D, FIND D_i AND D_{i-1} IN THE TABLE OF GEMINI RANGE PREDICTION CONSTANTS SUCH THAT $D_{i-1} < D < D_i$ THEN

8) $F1(D) = F1_{i-2} + \frac{D - D_{i-1}}{D_i - D_{i-1}} (F1_i - F1_{i-1})$

9) $F2(D) = F2_{i-1} + \frac{D - D_{i-1}}{D_i - D_{i-1}} (F2_i - F2_{i-1})$

10) $F0(D) = F0_i + 2(D-D_i)(F0_i - F0_{i-1} + 2(D-D_i)(D-D_{i-1})$

$(F0_{i+1} - 2F0_i + F0_{i-1})$

11) RB $= F0(D) + F1(D)\ \gamma + F2(D)\ V$ N.M.

CALCULATION OF BANK ANGLES AT .6 SECONDS (PHI6), 1 SECOND (PHI1), AND 1.2 SECONDS (PHI12), AND THE ACTUAL ROLL RATE AT 1.2 SECONDS (PDOT12).

1) PHI6 = PHI12 + .6* PDOT12 + .5* BDD $TAU6^2$
+ BDD* TAU6* (.6-TAU6) DEG.

2) PHI1 = PHI12 + PDOT12 + 5* BDD* $TAU1^2$
+ BDD* TAU1* (1-TAU1) DEG.

3) $PHI12 = PHI12 + 1.2 * PDOT12 + .5 * BDD * TAU12^2$
$+ BDD * TAU12 * (1.2-TAU12)$ DEG.

4) $PDOT12 = PDOT12 + TAU12 * BDD$ DEG/SEC

WHERE BDD = ROLL ACCELERATION

TAU6, TAU1, TAU12 = TIME JETS REMAIN ON

USAGE

A) ENTRY CALLED BY THE INTEGRATOR IN N3SDCL

B) EXIT EXITS TO THE INTEGRATOR

C) CALLING SEQUENCE -

```
TSX N7PCAL,4
+1 ERROR RETURN
+2 NORMAL RETURN
```

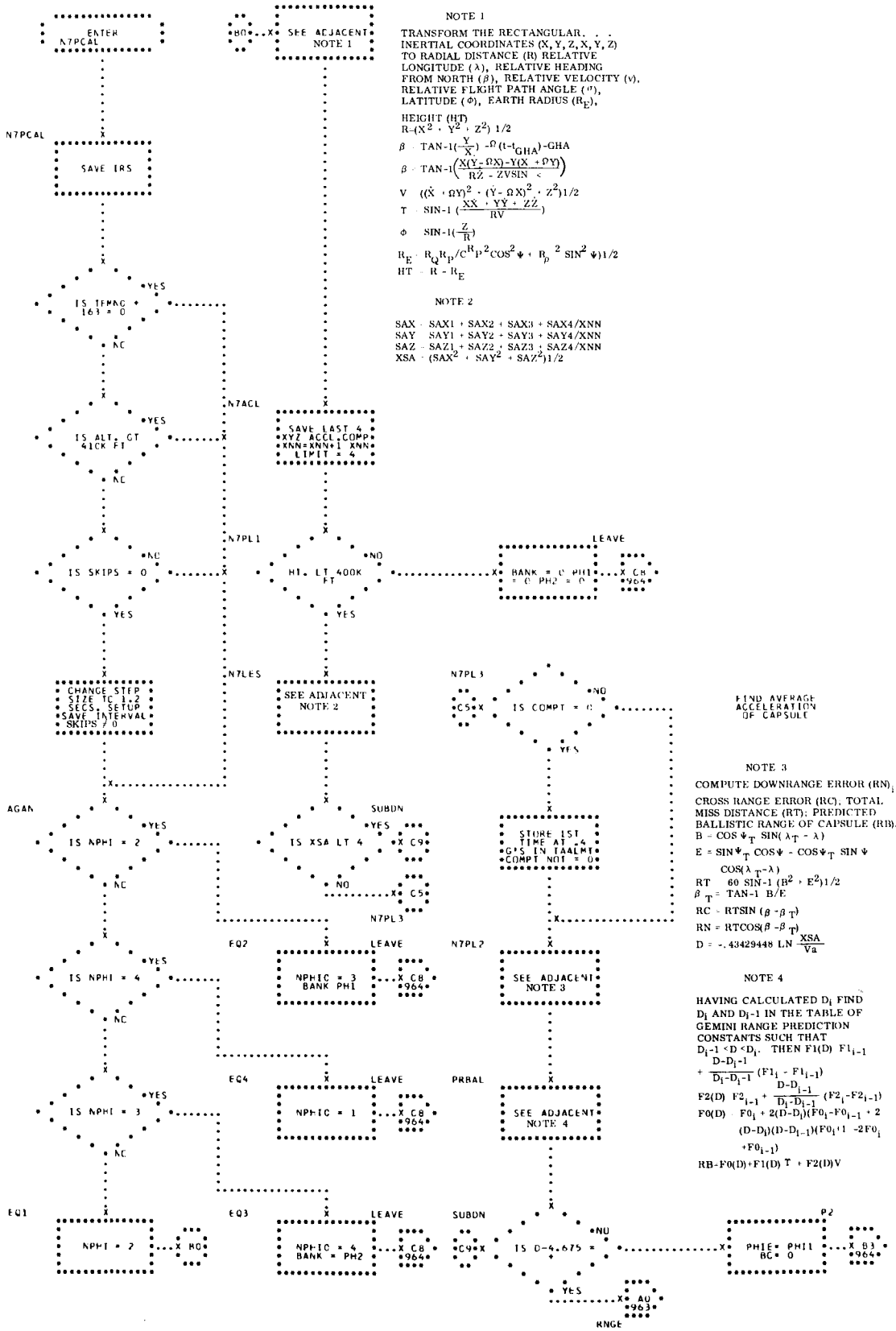
D) STORAGE REQUIRED -

960 LOCATIONS USED FOR THE FOLLOWING -

1) 778 INSTRUCTIONS
2) 87 STORAGE CELLS
3) 95 INTERNAL CONSTANTS

A) ERROR CODES -

123-ERROR RETURN FROM LIBRARY SUBROUTINE



NOTE 1

TRANSFORM THE RECTANGULAR. . . INERTIAL COORDINATES (X, Y, Z, X, Y, Z) TO RADIAL DISTANCE (R) RELATIVE LONGITUDE (λ), RELATIVE HEADING FROM NORTH (β), RELATIVE VELOCITY (v), RELATIVE FLIGHT PATH ANGLE (γ), LATITUDE (Φ), EARTH RADIUS (R_E),

HEIGHT (HT)

$R=(X^2 + Y^2 + Z^2)\,1/2$

$\beta = TAN\text{-}1(\frac{Y}{X}) - \Omega(t-t_{GHA})\text{-}GHA$

$\beta = TAN\text{-}1\left(\frac{X(\dot{Y}-\Omega X)-Y(\dot{X}+\Omega Y)}{R\dot{Z} - ZVSIN\ \gamma}\right)$

$V = ((\dot{X}+\Omega Y)^2 + (\dot{Y}-\Omega X)^2 + \dot{Z}^2)1/2$

$\gamma = SIN\text{-}1\ (\frac{X\dot{X}+Y\dot{Y}+Z\dot{Z}}{RV})$

$\Phi = SIN\text{-}1(\frac{Z}{R})$

$R_E = R_Q R_P / C^{R_P{}^2}\cos^2\Psi + R_P{}^2 \sin^2\Psi)1/2$

$HT = R - R_E$

NOTE 2

$SAX = SAX1 + SAX2 + SAX3 + SAX4/XNN$

$SAY = SAY1 + SAY2 + SAY3 + SAY4/XNN$

$SAZ = SAZ1 + SAZ2 + SAZ3 + SAZ4/XNN$

$XSA = (SAX^2 + SAY^2 + SAZ^2)1/2$

NOTE 3

COMPUTE DOWNRANGE ERROR (RN), CROSS RANGE ERROR (RC), TOTAL MISS DISTANCE (RT); PREDICTED BALLISTIC RANGE OF CAPSULE (RB).

$B = \cos\Psi_T \sin(\lambda_T - \lambda)$

$E = \sin\Psi_T \cos\Psi - \cos\Psi_T \sin\Psi \cos(\lambda_T-\lambda)$

$RT = 60\ SIN\text{-}1\ (B^2 + E^2)1/2$

$\beta_T = TAN\text{-}1\ B/E$

$RC = RTSIN\ (\beta - \beta_T)$

$RN = RTCOS(\beta - \beta_T)$

$D = -.43429448\ LN\ \frac{XSA}{Va}$

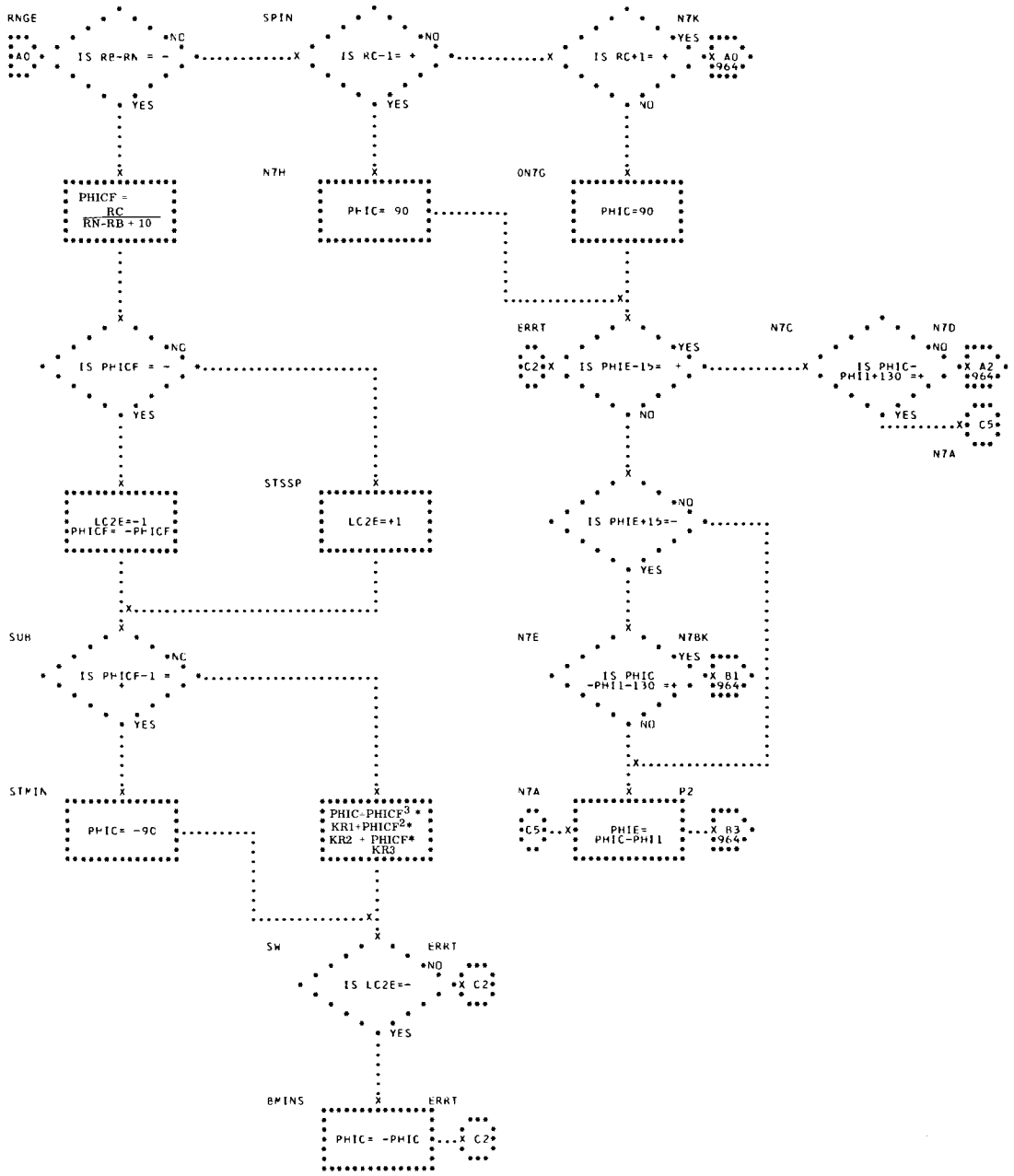
NOTE 4

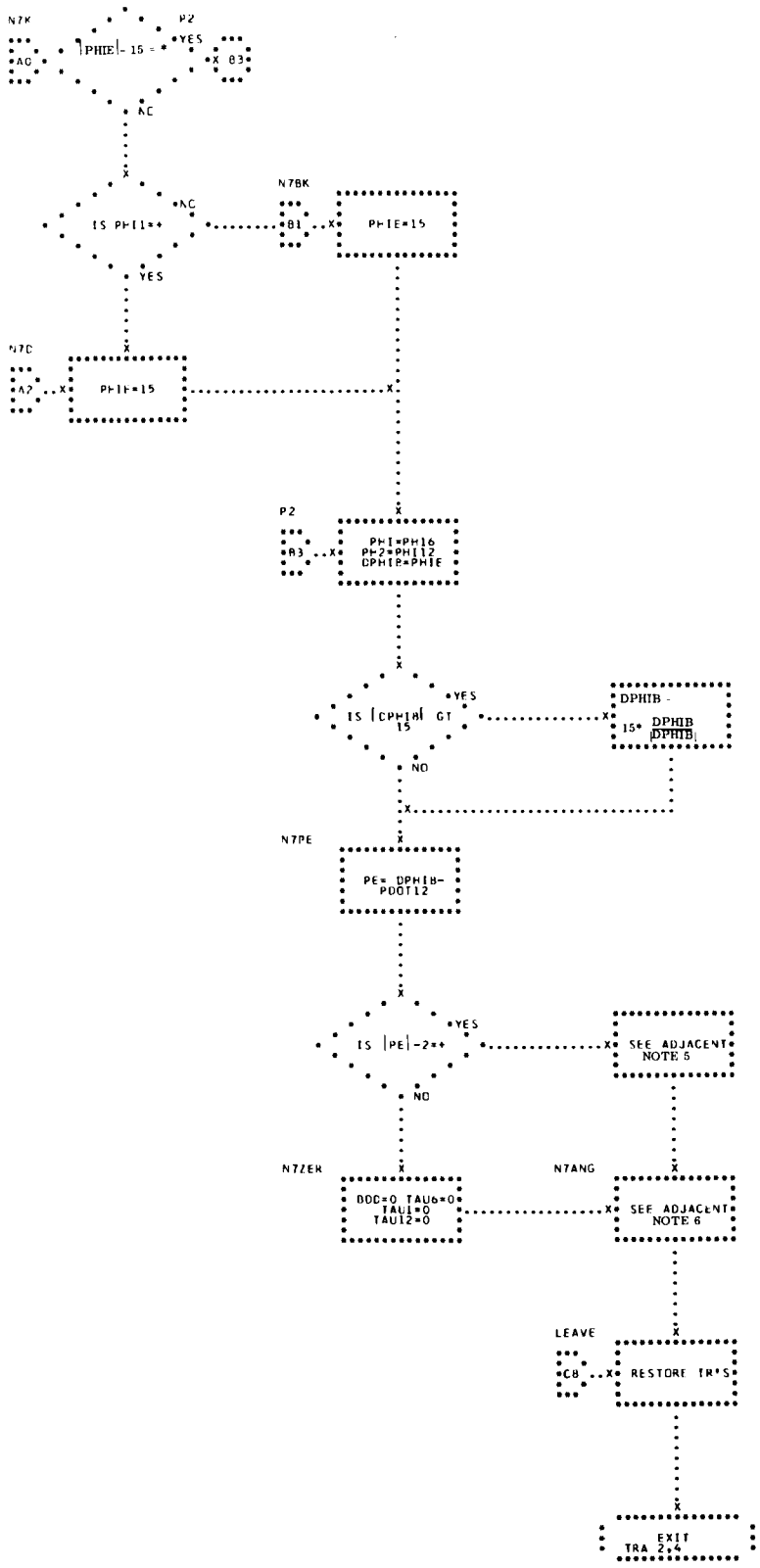
HAVING CALCULATED D_i FIND D_i AND D_i-1 IN THE TABLE OF GEMINI RANGE PREDICTION CONSTANTS SUCH THAT $D_{i-1} < D < D_i$. THEN $F1(D)\ F1_{i-1} + \frac{D-D_{i-1}}{D_i-D_{i-1}}(F1_i - F1_{i-1})$

$F2(D)\ F2_{i-1} + \frac{D-D_{i-1}}{D_i-D_{i-1}}(F2_i - F2_{i-1})$

$F0(D) = F0_i + 2(D-D_i)(F0_i - F0_{i-1} + 2(D-D_i)(D-D_{i-1})(F0_{i+1} - 2F0_i + F0_{i-1})$

$RB = F0(D) + F1(D)\ T + F2(D)V$


RNGE
IS RB-RN = -
*NO
*A0
YES
PHICF = RC / RN-RB + 10
SPIN
IS RC-1= +
*NO
YES
N7H
PHIC= 90
N7K
IS RC+1= +
*YES
*X A0 *964*
NO
ON7G
PHIC=90
IS PHICF = -
*NO
YES
LC2E=-1 PHICF= -PHICF
STSSP
LC2E=+1
ERRT
*C2*X
IS PHIE-15= +
*YES
NO
N7C
IS PHIC-PHI1+130 =+
*YES
N7D
*NO
*X A2 *964*
X C5
N7A
IS PHIE+15=-
*NO
YES
SUB
IS PHICF-1 = +
*NO
YES
N7E
IS PHIC -PHI1-130 =+
*YES
*X B1 *964*
NO
N7BK
STMIN
PHIC= -90
PHIC=PHICF3 * KR1+PHICF2* KR2 + PHICF* KR3
N7A
*C5*X
PHIE= PHIC-PHI1
P2
*X B3 *964*
SW
IS LC2E=-
*NO
X C2
YES
BMINS
PHIC= -PHIC
ERRT
X C2



NOTE 5

$BDD = 5 * PE/|PE|$

$TAU12 = \frac{/PE/-2}{5}$, LIMITED TO 1.2

TAU1 = TAU12, LIMITED TO 1.0

TAU6 = TAU1, LIMITED TO 0.6

NOTE 6

$PHI6 = PHI12 + .6 * PDOT12 + .5 * BDD * TAU6^2 + BDD * TAU6 * (.6 - TAU6)$

$PHI1 = PHI12 + PDOT12 + .5 * BDD * TAU1^2 + BDD * TAU1 * (1. TAU1)$

$PHI12 = PHI12 + 1.2 * PDOT12 + .5 * BDD * TAU12^2 + BDD * TAU12 * (1.2 - TAU12)$

$PDOT12 = PDOT12 + TAU12 * BDD$

BANK ANGLES AT .6, 1, 1.2 SECONDS AND ROLL RATE AT 1.2 SECONDS

44. MPDIFC - MONITOR PREFIX TO DODIFC

MPDIFC CONTROLS THE INITIAL PASS THROUGH DIFFERENTIAL CORRECTION (DC) WHEN NEW RADAR OBSERVATIONS ARE TO BE PROCESSED BY DIFFERENTIAL CORRECTION. (SEE FLOWCHART)

INPUT

A) A QUEUE WORD WHOSE ADDRESS CONTAINS THE INPUT LOCATION OF THE NEW EDITED RADAR MESSAGE BLOCK.

B) TMSTMS — A TABLE OF MNLTMS ENTRIES CONTAINING IN THE ADDRESS OF EACH ENTRY THE LOCATION OF THE TMDATA BLOCK ASSIGNED TO EACH RADAR STATION USED IN DC. ENTRIES ARE ORDERED BY THEIR RESPECTIVE TIMES OF CLOSEST APPROACH, FROM NEWEST TO OLDEST, AS THE TABLE PROCEEDS FROM TOP TO BOTTOM. THE OLDEST ENTRY IN DC IS NEGATIVE. REMAINING ENTRIES ARE NEGATIVE ZEROS.

C) MNLTMS — PARAMETER EQUATED WITH THE GREATEST POSSIBLE NUMBER OF ENTRIES IN TMSTMS.

D) MCNNRF — WHEN NONZERO, INDICATES RETRO-ROCKETS HAVE FIRED AND THE SYSTEM IS IN THE REENTRY PHASE.

OUTPUT

A) MCLENT — CONTAINS THE LOCATION OF THE NEW ENTRY IN TMSTMS, RELATIVE TO THE FIRST LOCATION IN TMSTMS. MCLENT

		RANGES FROM ZERO TO MNLTMS-1.
B)	MCDCRJ	SET TO NONZERO WHEN NEW OBSERVATIONS ARE TO BE OMITTED FROM THE DC INTERVAL. WHEN NONZERO, MCDCRJ INDICATES TO THE DC SUFFIX THAT DC WAS REJECTED IN THE PREFIX.
C)	AC	CONTAINS MCLENT AND IS POSITIVE ON TRANSFER TO DODIFC. THE AC IS NONZERO AND POSITIVE ON TRANSFER TO MDFIFC.
D)	TMSTMS	UPDATED
E)	TMDATA	UPDATED
F)	TMVTCA	DURING ORBIT, THIS TABLE MAY BE GENERATED FROM THE DEFINING VECTOR.

METHOD

MPDIFC HAS THE LOWEST PRIORITY OF THE THREE PREFIXES TO DODIFC. WHEN MPDIFC IS ENTERED, MPDIFM (MANUAL INTERVENTION PREFIX) MUST BE SUPPRESSED UNTIL THE FIRST PASS THROUGH DC IS COMPLETED. MPDIFK (THE HIGHEST PRIORITY) NEED NOT BE SUPPRESSED BECAUSE IT IS USED ONLY AFTER A FIRST PASS THROUGH DC HAS BEEN COMPLETED.

MPDIFC USES MCLCUC TO MOVE EDITED DATA FROM THE INPUT BUFFER TO THE 32K BUFFER MODULE. IN THE 32K BUFFER MODULE THE FOLLOWING TESTS ARE PERFORMED TO DETERMINE WHEN OBSERVATIONS CAN BE INCORPORATED INTO DC -

A) IS TIME OF CLOSEST APPROACH LATER THAN PRESENT TIME MINUS 225 MINUTES

B) ARE OBSERVATIONS OLDER THAN THOSE PRESENTLY IN CORE IF CORE IS FULL

C) ARE OBSERVATIONS FROM A RADAR WHICH IS ALTERNATE TO THE FIRST REPORTING RADAR (MULTI-RADAR SITE)

D) ARE OBSERVATIONS THE SECOND SET RECEIVED FROM SAME REPORTING RADAR DURING SAME ORBIT

IF ANY ONE OF THE ABOVE CITED CONDITIONS OCCURS, NEW OBSERVATIONS WILL NOT BE INCORPORATED AND AN ON-LINE MESSAGE LISTING THE CONDITION(S) WILL BE PRINTED. IF ALL CONDITIONS ARE SATISFIED, A CORRESPONDING ENTRY IS MADE IN TMSTMS (ORDERED CHRONOLOGICALLY BY t_{ca}). MSINSR INSERTS NEW ENTRIES IN TMSTMS AND MSTINT ADJUSTS THE TABLE TO EXCLUDE ENTRIES FOR OBSERVATIONS TIME-TAGGED OLDER THAN 225 (=MCMXDC) MINUTES FROM THE CURRENT GMT.

IF THE CORRESTION IS BEING MADE DURING THE ORBIT PHASE, A TABLE OF VECTORS (TMVTCA), DEFINING ORBITAL CONDITIONS AT EACH STATION IN THE CURRENT DC AT THEIR TIMES OF CLOSEST APPROACH (t_{ca}) IS GENERATED FROM THE DEFINING VECTOR (TDCDEF) BY MSVTCA. TO CONSERVE STORAGE, THE VECTORS DESCRIBING THE LOCAL ORBITAL ARCS ABOVE EACH STATION WILL BE GENERATED FROM TMVTCA AS THEY ARE NEEDED BY A SHORT ARC RUNGE-KUTTA INTEGRATION WITHIN THE DC CALCULATIONS. ALL VECTORS REQUIRED FOR DC DURING REENTRY ARE STORED IN THE REENTRY DISPLAY TABLE (TNINT2) BEFORE ENTRY TO MPDIFC. THIS IS A COMPLETE TABLE OF VECTORS OBTAINED BY INTEGRATION AT EQUALLY SPACED (SHORT) INTERVALS OF TIME THROUGHOUT THE REENTRY PHASE.

SUBSEQUENT TO A QUEUE FOR ANY PREFIX ERROR MESSAGE, CONTROL IS TRANSFERRED TO THE SUFFIX OF DC. IF THE CONDITIONS HAVE INDICATED THAT THE NEW RADAR OBSERVATIONS CAN BE USED IN A DIFFERENTIAL CORRECTION TO IMPROVE THE PREDICTED TRAJECTORY, CONTROL IS GIVEN TO DODIFC.

USAGE

MPDIFC IS ENTERED FROM MOPRIO AND EXITS TO EITHER DODIFC OR MFDIFC.

A) STORAGE REQUIRED -

107 LOCATIONS

B) MACROS -

GENBA

QUEUE

UNQUE

TRNCN

C) SUBROUTINES -

MSLCUC

MSINSR

MSTINT

MSVTCA

D) PARAMETERS -

MNDIFC

MNMESS

MNREST

MNDIFM

MNLTMS

E) COMMUNICATION CELLS -

MCDCRJ MCBKNO

MCLENT MCMXDC

MCDCIP MCMNOB

MCINST MCARCL

MCSTNC MCWGFC

MCNNRF

F) TABLES -

TMSTMS

TMSTCH

TMDATA (VIA TMSTMS)

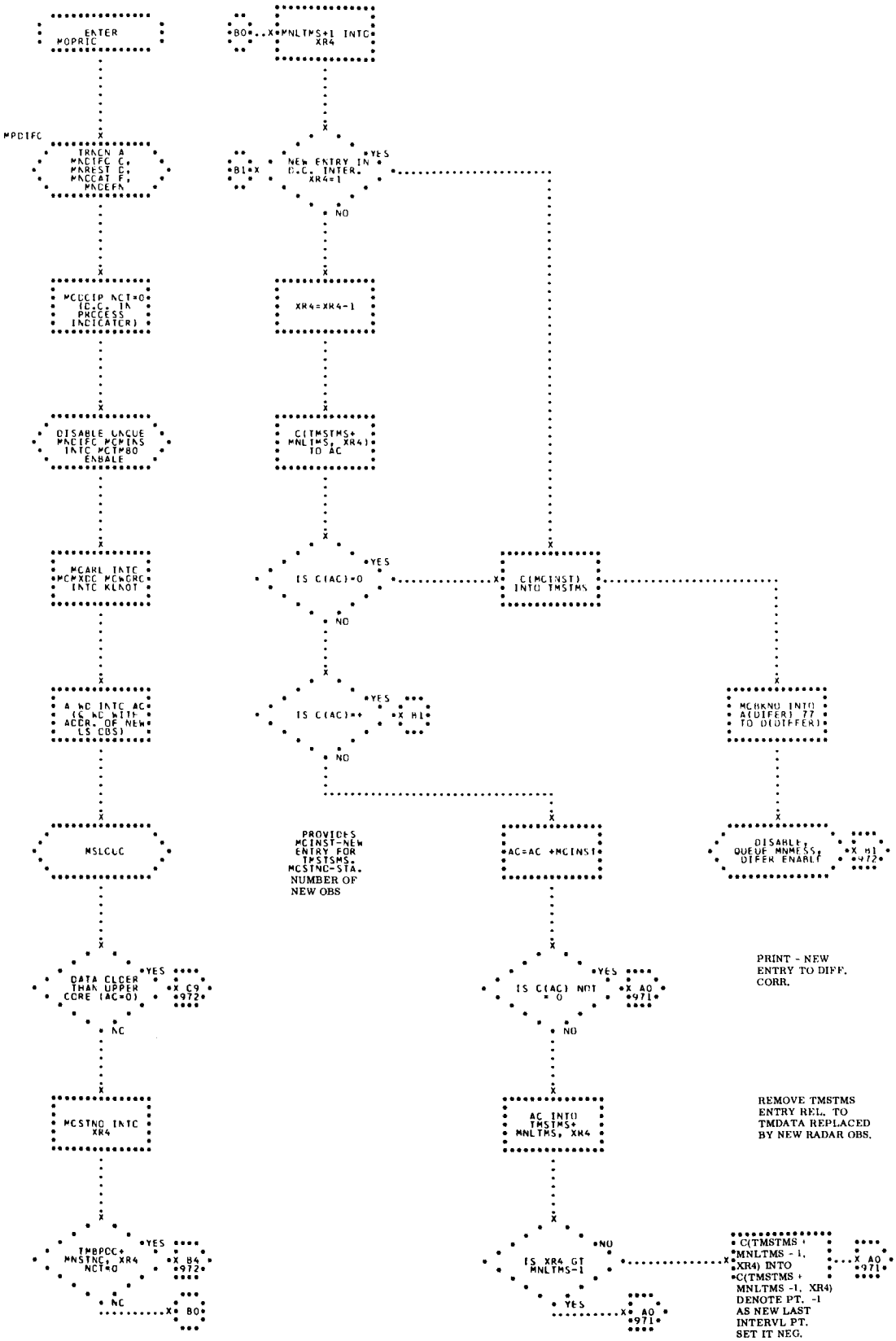
G) CONSTANTS -

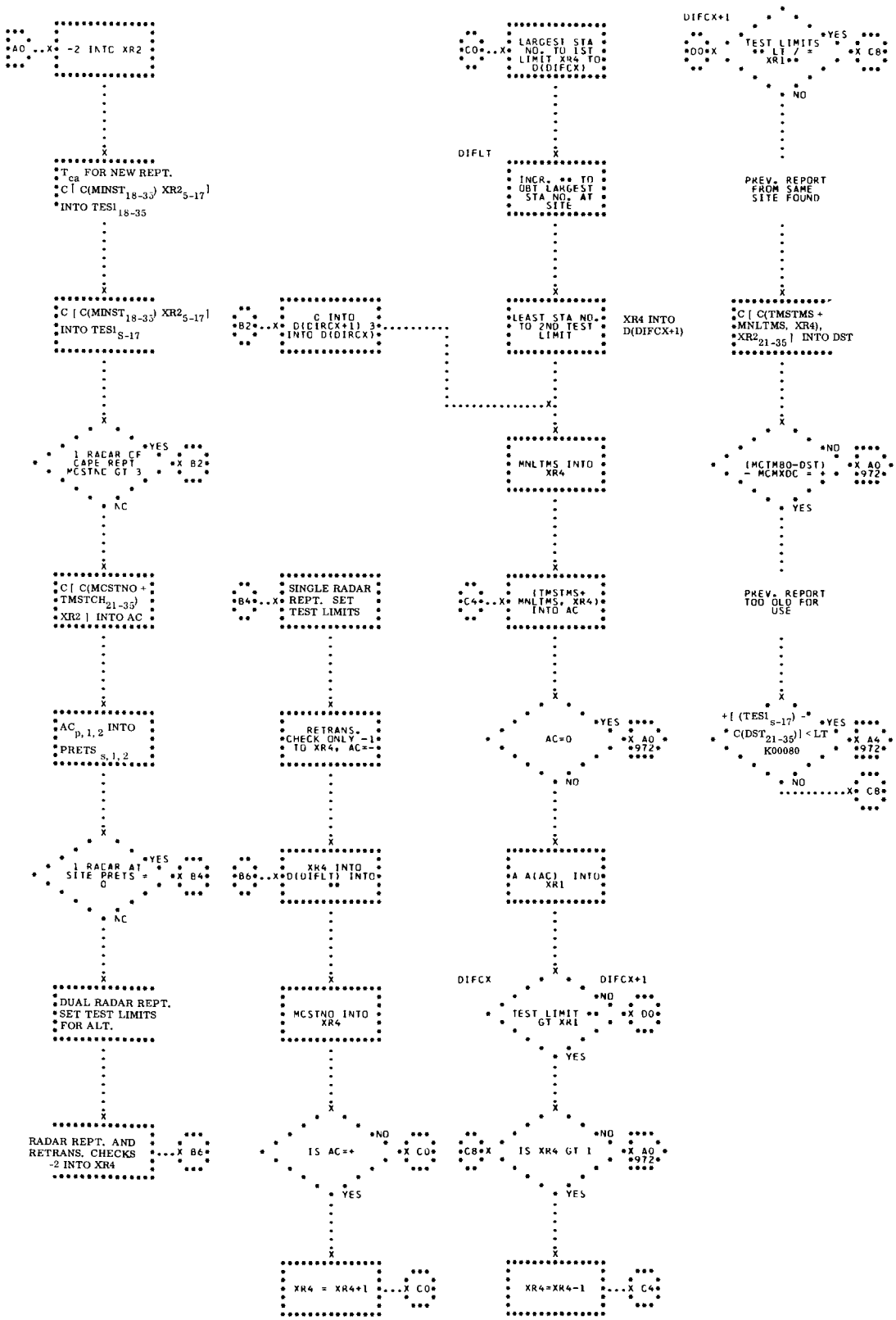
KC0080

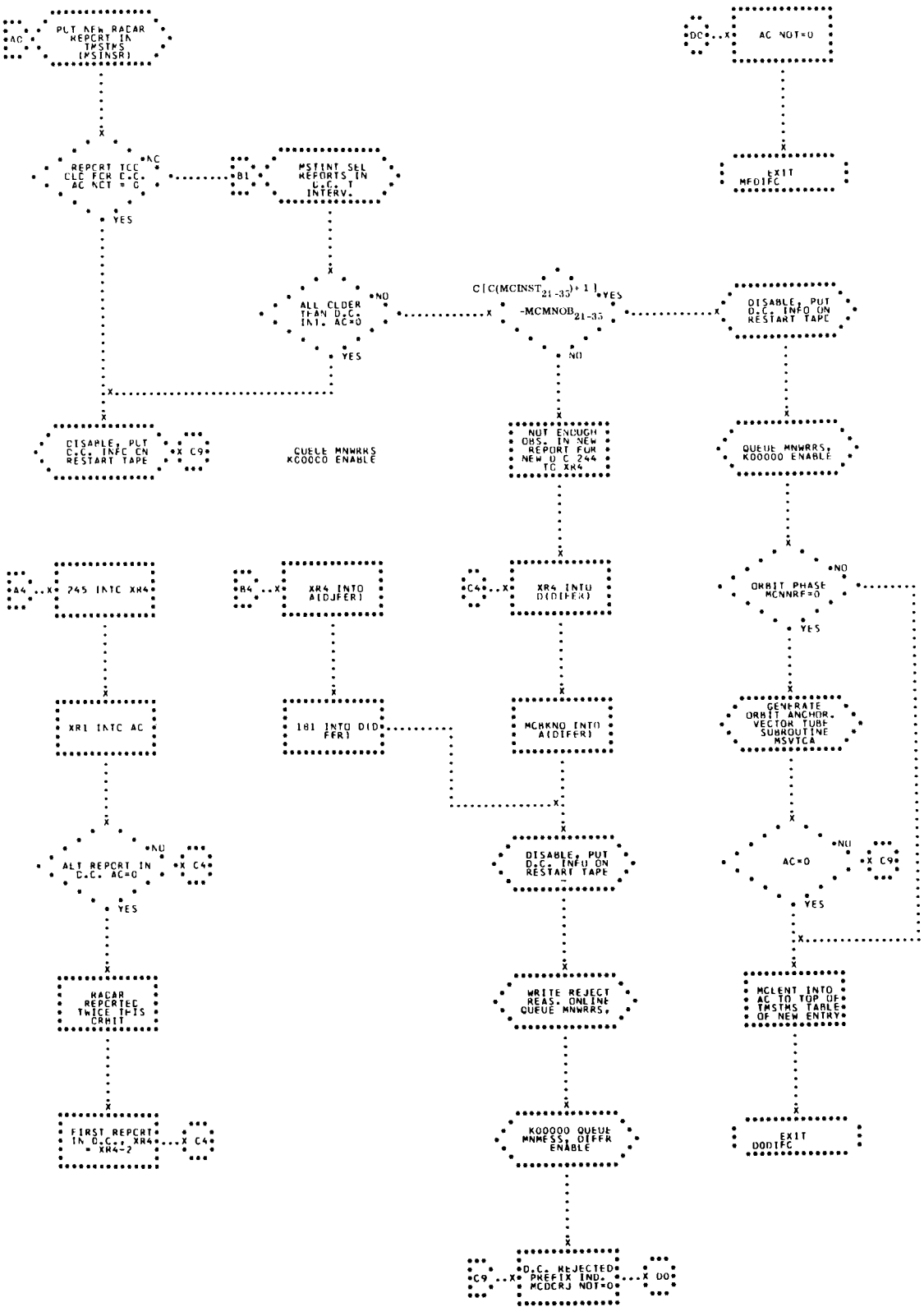
KLNO.7

H) TIME REQUIRED (INCLUDING SUBROUTINES) -

APPROXIMATELY 0.101 TO 2.48 MILLISECONDS.


ENTER MOPRTC
MPDIFC
TRACN A MNCIFC C, MNREST D, MNCCAT F, MNDEFA
MCDCIP NCT=0 (D.C. IN PROCESS INDICATER)
DISABLE UNCLUE MNCIFC MCPINS INTC MCTM80 ENBALE
MCARL INTC MCMXEC MCKDRC INTC KLNOT
A NO INTC AC (& NO WITH ACCR. OF NEW LS OBS)
MSLCLC
DATA CLOER THAN UPPER CORE (AC=0)
YES
NO
X C9 *972*
MCSTNO INTC XR4
TMBPOC* MNSTNC, XR4 NCT<0
X B4 *972*
X B0
B0
MNLTMS+1 INTO XR4
B1
NEW ENTRY IN D.C. INTER. XR4=1
XR4=XR4-1
C(TMSTMS+ MNLTMS, XR4) TO AC
IS C(AC)=0
IS C(AC)=+
X B1
PROVIDES MCINST-NEW ENTRY FOR TMSTSMS. MCSTNC-STA. NUMBER OF NEW OBS
C(MCINST) INTO TMSTMS
AC=AC +MCINST
IS C(AC) NOT = 0
X A0 *971*
AC INTO TMSTMS+ MNLTMS, XR4
IS XR4 GT MNLTMS-1
C(TMSTMS + MNLTMS - 1, XR4) INTO C(TMSTMS + MNLTMS -1, XR4) DENOTE PT. -1 AS NEW LAST INTERVL PT. SET IT NEG.
X A0 *971*
MC(KNO INTO A(DIFER) 77 TO D(DIFFER)
DISABLE, QUEUE MNMESS, DIFFR ENABLE
X B1 *972*
PRINT - NEW ENTRY TO DIFF. CORR.
REMOVE TMSTMS ENTRY REL. TO TMDATA REPLACED BY NEW RADAR OBS.


A0
-2 INTO XR2
T_{ca} FOR NEW REPT.
C[C(MINST$_{18-35}$) XR2$_{5-17}$]
INTO TES1$_{18-35}$
C [C(MINST$_{18-35}$) XR2$_{5-17}$]
INTO TES1$_{S-17}$
1 RADAR CF CAPE REPT MCSTNC GT 3
YES
X B2
NO
C[C(MCSTNO + TMSTCH$_{21-35}$) XR2] INTO AC
AC$_{p,1,2}$ INTO PRETS$_{s,1,2}$
1 RADAR AT SITE PRETS = 0
YES
X B4
NO
DUAL RADAR REPT. SET TEST LIMITS FOR ALT.
RADAR REPT. AND RETRANS. CHECKS -2 INTO XR4
X B6
B2
C INTO D(DIRCX+1) 3 INTO D(DIRCX)
B4
SINGLE RADAR REPT. SET TEST LIMITS
B6
XR4 INTO D(DIFLT) INTO
MCSTNO INTO XR4
IS AC=+
NO
X C0
YES
XR4 = XR4+1
X C0
C0
LARGEST STA NO. TO IST LIMIT XR4 TO D(DIFCX)
DIFLT
INCR. ** TO OBT LARGEST STA NO. AT SITE
LEAST STA NO. TO 2ND TEST LIMIT
XR4 INTO D(DIFCX+1)
MNLTMS INTO XR4
C4
(TMSTMS+ MNLTMS, XR4) INTO AC
AC=0
YES
X A0 *972
NO
A A(AC) INTO XR1
DIFCX
DIFCX+1
TEST LIMIT ** GT XR1
NO
X D0
YES
IS XR4 GT 1
NO
X A0 *972
C8
YES
XR4=XR4-1
X C4
DIFCX+1
D0
TEST LIMITS ** LT / = XR1**
YES
X C8
NO
PREV. REPORT FROM SAME SITE FOUND
C [C(TMSTMS + MNLTMS, XR4), XR2$_{21-35}$] INTO DST
(MCTM80-DST) - MCMXDC = +
NO
X A0 *972
YES
PREV. REPORT TOO OLD FOR USE
+[(TES1$_{s-17}$) - C(DST$_{21-35}$)] < LT K00080
YES
X A4 *972
NO
X C8


PUT NEW RADAR REPORT IN TMSTMS (MSINSR)
AC
REPORT TCC CLC FOR C.C. AC NOT = 0
*AC
YES
*B1
MSTINT SEL REPORTS IN C.C. T INTERV.
ALL OLDER THAN D.C. INT. AC=0
*NO
YES
C(C(MCINST21-35)+1) -MCMNOB21-35
*YES
NO
*DC
AC NOT=0
EXIT MFDIFC
DISABLE, PUT D.C. INFO ON RESTART TAPE
QUEUE MNWRRS, K00000 ENABLE
DISABLE, PUT D.C. INFO ON RESTART TAPE
X C9
QUEUE MNWRRS K00000 ENABLE
NOT ENOUGH OBS. IN NEW REPORT FOR NEW D C 244 TO XR4
ORBIT PHASE MCNNRF=0
*NO
YES
A4
245 INTO XR4
B4
XR4 INTO A(DIFFER)
C4
XR4 INTO D(DIFFER)
GENERATE ORBIT ANCHOR, VECTOR TUBE SUBROUTINE MSVTCA
XR1 INTO AC
181 INTO D(DIFFER)
MCHKNO INTO A(DIFFER)
ALT REPORT IN D.C. AC=0
*NO
X C4
YES
AC=0
*NO
X C9
YES
DISABLE, PUT D.C. INFO ON RESTART TAPE
RADAR REPORTED TWICE THIS ORBIT
WRITE REJECT REAS. ONLINE QUEUE MNWRRS,
MCLENT INTO AC TO TOP OF TMSTMS TABLE OF NEW ENTRY
FIRST REPORT IN D.C., XR4 = XR4=2
K00000 QUEUE MNMESS, DIFFR ENABLE
EXIT DODIFC
C9
D.C. REJECTED PREFIX IND. MCDCRJ NOT=0
X D0

45. MPDIFK - MONITOR PREFIX TO DODIFC

MPDIFK RECEIVES CONTROL AFTER DODIFC HAS REQUESTED A SUCCESSIVE DIFFERENTIAL CORRECTION USING THE SAME RADAR DATA. MPDIFK IS GIVEN CONTROL FOR AS MANY TIMES AS ANOTHER DIFFERENTIAL CORRECTION IS REQUESTED FOR A GIVEN RADAR STATION'S DATA. (SEE FLOWCHART.)

INPUT

A) AN INPUT QUEUE WORD WHOSE CONTENTS ARE ZERO IF THE PROCESSING IS FOR ORBIT PHASE AND NONZERO FOR ANY OTHER.

B) MCLENT CONTAINS THE LOCATION, RELATIVE TO TMTMS, OF THE LATEST ENTRY IN THE TMSTMS TABLE.

OUTPUT

A) IF THE PROCESSING IS FOR ORBIT PHASE, TMVTCA IS UPDATED USING THE SUBROUTINE MSVTCA.

B) BEFORE TRANSFER OF CONTROL TO DODIFC, THE CONTENTS OF MCLENT ARE STORED IN THE ACCUMULATOR AND THE SIGN IS SET TO NEGATIVE.

METHOD

MPDIFK PROVIDES DODIFC WITH THE RELATIVE LOCATION OF THE LATEST ENTRY IN THE TMSTMS TABLE BY PLACING THE CONTENTS OF MCLENT IN THE ACCUMULATOR. THE ACCUMULATOR SIGN IS SET TO NEGATIVE TO INDICATE THAT THIS IS A SUCCESSIVE ENTRY AND THAT NO NEW OBSERVATIONS ARE TO BE USED IN THE DIFFERENTIAL CORRECTION.

USAGE

MPDIFK IS ENTERED FROM MOPRIO AND EXITS TO DODIFC.

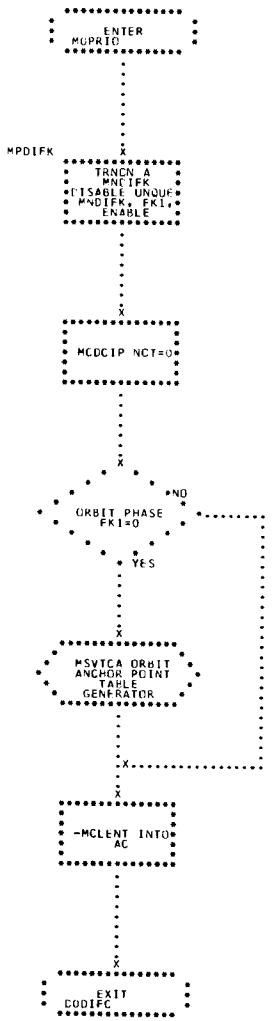
A) STORAGE REQUIRED -
18 LOCATIONS

B) MACROS -
GENBA
GENBZ
TRNCN
UNQUE

C) PARAMETER -
MNDIFK

D) COMMUNICATION CELLS -
MCDCIP
MCLENT

E) TIME REQUIRED -
APPROXIMATELY 0.061 MILLISECOND.

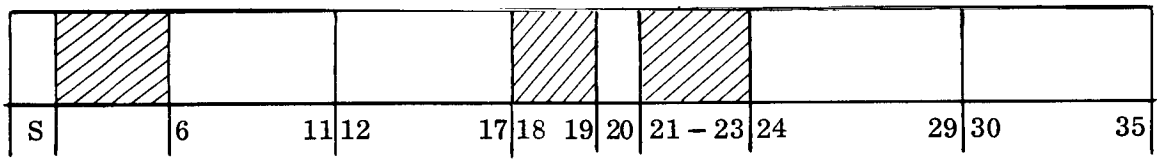

ENTER
MDPRIO
MPDIFK
TRNCN A
MNCIFK
DISABLE UNQUE
MNDIFK, FK1,
ENABLE
MCDCIP NCT=0
NO
ORBIT PHASE
FK1=0
YES
MSVTCA ORBIT
ANCHOR POINT
TABLE
GENERATOR
-MCLENT INTO
AC
EXIT
CODIFC

46. MPDIFM - MONITOR PREFIX TO DODIFC

MPDIFM IS ENTERED MYKEYS AS A RESULT OF A MANUAL REQUEST VIA THE PROGRAM CONTROL CONSOLE (PCC) OR FROM MYDCIT DURING THE ORBIT-TO-REENTRY INTERPHASE. (SEE FLOW CHART.)

INPUT

A) AN INPUT QUEUE ENTRY WORD IN THE FOLLOWING FORMAT -



EACH OF THE TWO 6-BIT SUBFIELDS (24-29, 30-35) IN THE ADDRESS FIELD CAN CONTAIN THE BLOCK NUMBER OF AN EDITED LOW-SPEED DATA BLOCK WHOSE CONTENTS ARE TO BE INCLUDED IN THE NEXT DIFFERENTIAL CORRECTION. EACH OF THE TWO SUBFIELDS (6-11, 12-17) IN THE DECREMENT FIELD CAN CONTAIN THE BLOCK NUMBER OF AN EDITED LOW-SPEED DATA BLOCK WHOSE CONTENTS ARE TO BE EXCLUDED FROM THE NEXT DIFFERENTIAL CORRECTION. A NEGATIVE SIGN INDICATES THAT A DIFFERENTIAL CORRECTION IS DESIRED IMMEDIATELY. THE ABSENCE OF A BIT IN POSITION 20 INDICATES THAT THE DIFFERENTIAL CORRECTION IS TO BE FORCED. THUS MCFORC IS SET TO NONZERO.

B) MCINST ADDRESS FIELD CONTAINS THE TMDATA ADDRESS CORRESPONDING TO THE LOW-SPEED EDITED DATA BLOCK WHOSE OBSERVATIONS

ARE TO BE INCLUDED OR EXCLUDED FROM DIFFERENTIAL CORRECTION.

C) MCMINS CONTAINS CURRENT GMT IN FIXED-POINT MINUTES.

D) MCNRRF WHEN NONZERO, INDICATES RETROROCKETS HAVE FIRED AND THE SYSTEM IS IN REENTRY PHASE.

E) MCTABT WHEN NONZERO, INDICATES THE SYSTEM IS IN THE HIGH-ABORT PHASE.

F) TMSTMS A TABLE OF MNLTMS ENTRIES, CONTAINING IN IN THE ADDRESS FIELD OF EACH ENTRY THE LOCATION OF THE TMDATA BLOCK ASSIGNED TO EACH REPORTING STATION WHICH IS TO BE USED IN THE DIFFERENTIAL CORRECTION. THE ENTRIES ARE ORDERED ACCORDING TO THE TIME OF MINIMUM RANGE (TMR) WITH THE ENTRY FOR THE MORE RECENT TIME FIRST. THE LAST ENTRY IN THE TABLE IS SET NEGATIVE - ALL PRECEDING ENTRIES ARE POSITIVE, WHILE ALL REMAINING LOCATIONS ARE NEGATIVE ZERO.

OUTPUT

A) MCBKNO ADDRESS FIELD CONTAINS BLOCK NUMBER CURRENTLY BEING PROCESSED.

B) MCDCIP SET TO NONZERO TO INDICATE DIFFERENTIAL CORRECTION IS IN PROCESS.

C) MCFORC SET TO NONZERO TO INDICATE THAT A DIFFERENTIAL CORRECTION HAS BEEN REQUESTED VIA MANUAL INSERTION.

D) MCLENT CONTAINS THE POSITION, RELATIVE TO THE TOP OF THE LATEST ENTRY IN THE TMSTMS

TABLE.

E) MCSTNO THE ADDRESS FIELD CONTAINS THE STATION NUMBER OF THE MOST RECENT ENTRY IN THE TMSTMS TABLE OR, IN THE CASE OF MANUAL INSERTION, THE STATION NUMBER ASSOCIATED WITH THE MOST RECENT TMR IN THE TMSTMS TABLE.

F) MCTM80 ADDRESS FIELD CONTAINS CURRENT GMT IN FIXED-POINT MINUTES.

THE ACCUMULATOR CONTAINS MCLENT WHEN MPDIFM EXITS TO DODIFC.

METHOD

WHEN PROCESSING THE INPUT QUEUE WORD, THE TWO 6-BIT SUBFIELDS OF THE DECREMENT ARE PROCESSED BEFORE THOSE OF THE ADDRESS FIELD. IF A SUBFIELD CONTAINS A ZERO, THE NEXT SUBFIELD IS EXAMINED. WHEN A SUBFIELD CONTAINS A NUMBER, A CHECK IS MADE TO ENSURE THAT THE NUMBER IS A LEGAL BLOCK NUMBER AND IF NOT, AN ERROR MESSAGE IS QUEUED BEFORE THE NEXT SUBFIELD IS EXAMINED.

WHEN A VALID BLOCK NUMBER IS FOUND, THE SUBROUTINE MSWTMS IS ENTERED TO GENERATE THE TMSTMS WORD FOR THE BLOCK NUMBER BEING PROCESSED. IF THE BLOCK NUMBER ORIGINATED IN A DECREMENT SUBFIELD, THE SUBROUTINE MSDELT IS ENTERED TO DELETE THE ENTRY IN THE TMSTMS TABLE CORRESPONDING TO THE TMSTMS WORD GENERATED BY MSWTMS. IF THE BLOCK NUMBER ORIGINATED IN AN ADDRESS SUBFIELD, THE SUBROUTINE MSINSR IS USED TO INSERT THE WORD, GENERATED BY MSWTMS, INTO THE TMSTMS TABLE.

AFTER ALL SUBFIELDS HAVE BEEN EXAMINED, THE SUBROUTINE IS ENTERED TO TIME-CHECK THE TMSTMS TABLE ENTRIES. IF THE RETURN FROM MSTINT INDICATES THERE ARE NO USABLE

ENTRIES IN THE TMSTMS TABLE, THE PROGRAM EXITS TO MOPRIO. WHEN THE RETURN FROM MSTINT INDICATES THERE ARE USABLE ENTRIES IN THE TMSTMS TABLE, THE INPUT QUEUE WORD IS EXAMINED FOR A NEGATIVE SIGN INDICATING A REQUEST FOR A DIFFERENTIAL CORRECTION. IF SUCH A REQUEST IS MADE IN THE ORBIT PHASE, THE SUBROUTINE MSVTCA MUST BE ENTERED TO PROVIDE THE VECTORS REQUIRED BY THE DIFFERENTIAL CORRECTION PROGRAM, OTHERWISE, MSVTCA IS BYPASSED.

IF THERE HAS NOT BEEN AN INSERTION IN THE TMSTMS TABLE FOR THIS PROGRAM ENTRY, THE CONTENTS OF MCLENT ARE SET TO ZERO BEFORE UPDATING THE STATION IDENTIFICATION CODE IN MCSTNO. IN EITHER CASE, MCLENT IS PLACED IN THE ACCUMULATOR BEFORE CONTROL IS GIVEN TO DODIFC. DODIFC IS NOTIFIED OF A FORCED CORRECTION BY SETTING MCFORC NOT EQUAL TO ZERO, UNLESS BIT 20 OF THE ENTRY QUEUE WORD IS SET TO ONE.

IF THE INPUT QUEUE WORD IS POSITIVE, INDICATING NO REQUEST FOR A DIFFERENTIAL CORRECTION, CONTROL GOES TO MOPRIO.

USAGE

MPDIFM IS ENTERED FROM MOPRIO AND EXITS TO EITHER DODIFC OR MOPRIO.

A) STORAGE REQUIRED -
   109 LOCATIONS EXCLUSIVE OF SUBROUTINES.

B) MACROS -
   QENBA
   QENBZ
   QUEUE
   TRNOP
   TRNON
   UNQUE

C) PARAMETERS -

A

C

D

MNDIFM

MNMESS

MNUCMB

MNOCAT

MNREST

D) SUBROUTINES -

MSDELT

MSINSR

MSTINT

MSVTCA

MSWTMS

E) COMMUNICATION CELLS -

MCARLC MCMXDC

MCBKNO MCNRRF

MCDCIP MCSTNO

MCFORC MCTABT

MCLENT MCTM80

MCINST MCWGFC

MCMINS

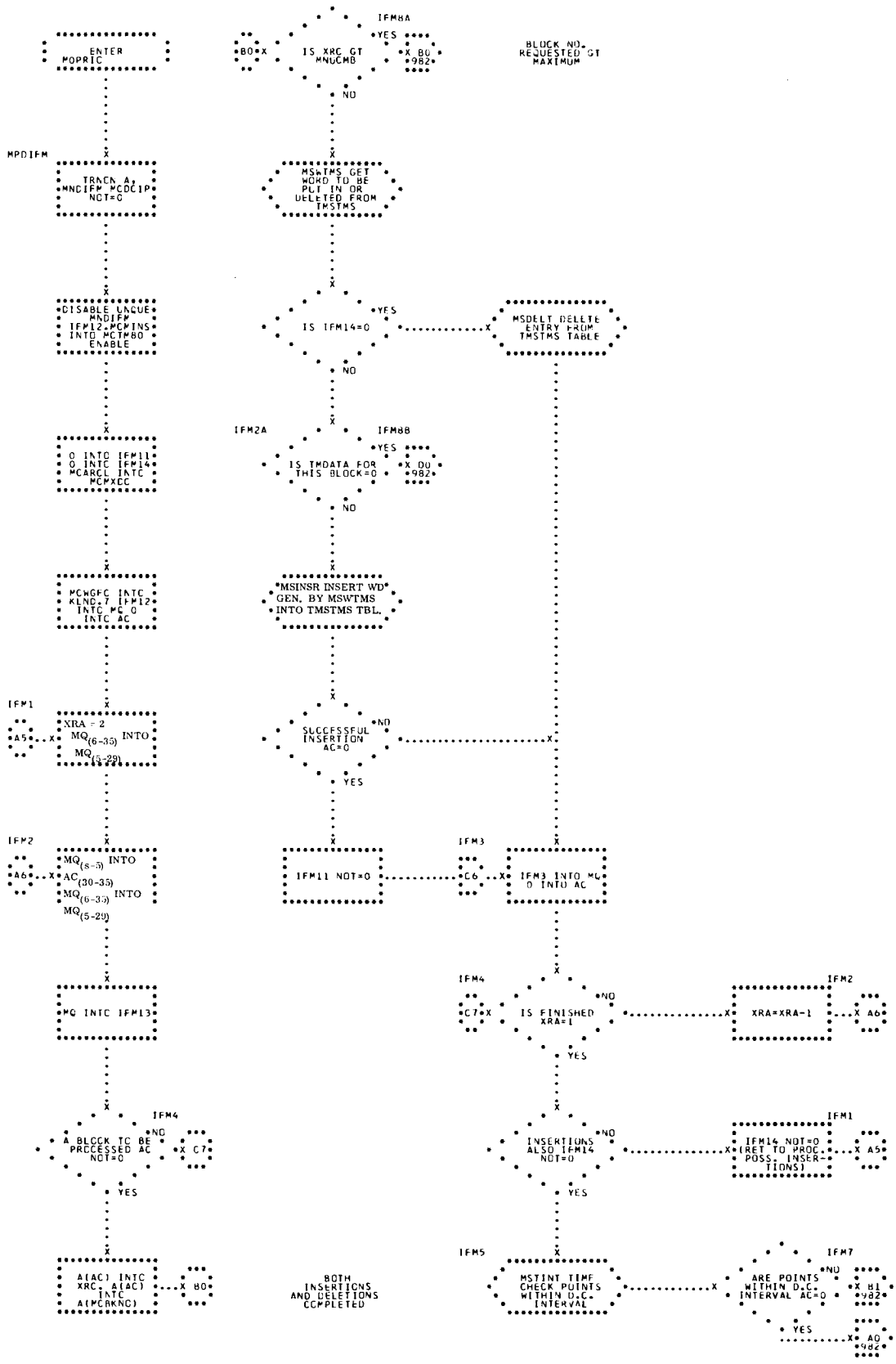
F) CONSTANTS -

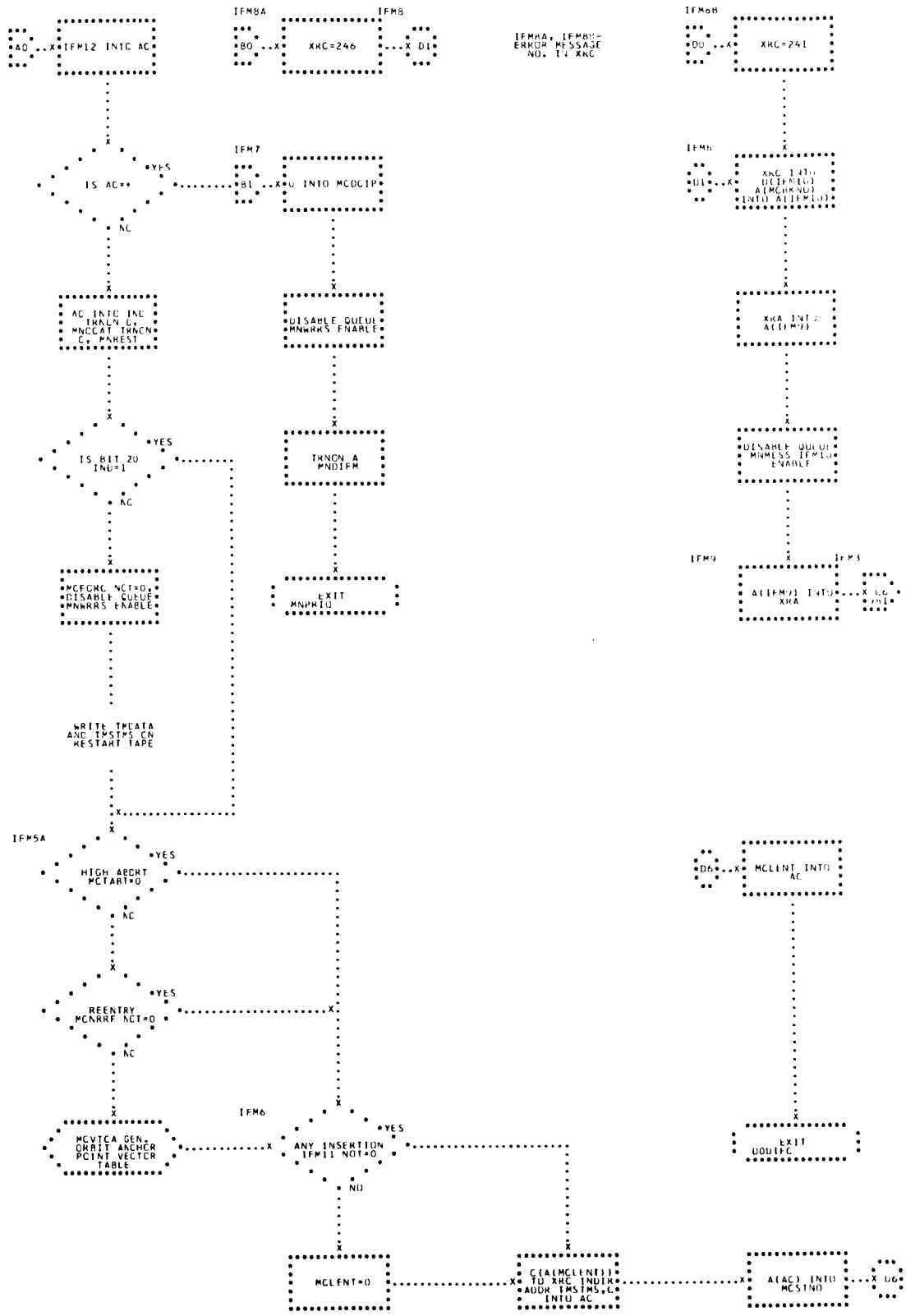
KCH233

KLNO.7

G) TIME REQUIRED -

0.16 TO 0.36 MILLISECOND APPROXIMATELY, EXCLUSIVE OF MACROS AND SUBROUTINES.


ENTER MOPRIC
IFM8A
IS XRC GT MNUCMB
YES
NO
BLOCK NO. REQUESTED GT MAXIMUM
MPDIFM
TRACN A, MNDIFM MCDC1P NOT=0
MSWTMS GET WORD TO BE PUT IN OR DELETED FROM TMSTMS
DISABLE UNQUE MNDIFM IFM12,MCMINS INTO MCTM80 ENABLE
IS IFM14=0
MSDELT DELETE ENTRY FROM TMSTMS TABLE
0 INTO IFM11 0 INTO IFM14 MCARCL INTC MCMXEC
IFM2A
IFM8B
IS TMDATA FOR THIS BLOCK=0
MCWGFC INTC KLNG.7 IFM12 INTC MC 0 INTC AC
MSINSR INSERT WD GEN. BY MSWTMS INTO TMSTMS TBL.
IFM1
XRA = 2 MQ(6-35) INTO MQ(5-29)
SUCCESSFUL INSERTION AC=0
IFM2
IFM3
MQ(8-5) INTO AC(30-35) MQ(6-35) INTO MQ(5-29)
IFM11 NOT=0
IFM3 INTO MQ 0 INTO AC
IFM4
IFM2
MQ INTC IFM13
IS FINISHED XRA=1
XRA=XRA-1
IFM4
A BLOCK TO BE PROCESSED AC NOT=0
IFM1
INSERTIONS ALSO IFM14 NOT=0
IFM14 NOT=0 (RET TO PROC. POSS. INSERTIONS)
IFM5
IFM7
A(AC) INTC XRC, A(AC) INTC A(MCRKNO)
BOTH INSERTIONS AND DELETIONS COMPLETED
MSTINT TIME CHECK POINTS WITHIN D.C. INTERVAL
ARE POINTS WITHIN D.C. INTERVAL AC=0


IFM8A
IFM8
IFM8A, IFM8B-
ERROR MESSAGE
NO. IN XRC
IFM6B
A0
IFM12 INTO AC
B0
XRC=246
D1
D0
XRC=241
IFM7
IS AC=+
YES
NO
B1
0 INTO MCDCIP
IFM6
D1
XRC INTO
D(IFM10)
A(MCBKNO)
INTO A(IFM10)
AC INTO IND
TRNCN C,
MNCCAT TRNCN
C, MNREST
DISABLE QUEUE
MNWRRS ENABLE
XRA INTO
A(IFM9)
IS BIT 20
IND=1
TRNON A
MNDIFM
DISABLE QUEUE
MNMESS IFM10
ENABLE
MCFCRC NCT=0,
DISABLE QUEUE
MNWRRS ENABLE
EXIT
MNPRIO
IFM9
IFM3
A(IFM9) INTO
XRA
WRITE TMDATA
AND TMSTMS ON
RESTART TAPE
IFM5A
HIGH ABORT
MCTABT=0
D6
MCLENT INTO
AC
REENTRY
MCNRRF NCT=0
IFM6
MCVTCA GEN.
ORBIT ANCHOR
POINT VECTOR
TABLE
ANY INSERTION
IFM11 NOT=0
EXIT
DODIFC
MCLENT=0
C(A(MCLENT))
TO XRC INDIR
ADDR TMSTMS,C
INTO AC
A(AC) INTO
MCSTNO
D6

47. DIFFERENTIAL CORRECTION METHOD

DURING A GEMINI MISSION, THE GODDARD IBM 7094 COMPUTERS ACCEPT RADAR TRACKING DATA FROM THE WORLDWIDE TRACKING NETWORK. THE COMPUTERS USE THIS DATA TO ESTABLISH THE PRESENT POSITION OF THE SPACECRAFT, TO IMPROVE THE ORBIT DETERMINATION SO SUCCESSIVE RADAR STATIONS CAN ACQUIRE THE SPACECRAFT, AND TO SUPERVISE REENTRY. DIFFERENTIAL CORRECTION USES SITE RADAR DATA TO CORRECT ORBITAL POSITION AND VELOCITY PARAMETERS. A BRIEF DISCUSSION OF THE MATHEMATICAL TECHNIQUES EMPLOYED IN DIFFERENTIAL CORRECTION FOLLOWS.

IF THE POSITION AND VELOCITY VECTORS OF A SPACECRAFT IN ORBIT ARE KNOWN, ITS COORDINATES AT ANY FUTURE TIME CAN BE PREDICTED. THIS COMPUTATION IS ACCOMPLISHED BY NUMERICALLY INTEGRATING THE THREE-SECOND-ORDER DIFFERENTIAL EQUATIONS OF MOTION USING COWELL'S METHOD. HOWEVER, THE ACCURACY OF THIS PREDICTION IS DEPENDENT UPON THE ACCURACY OF THE INITIAL VECTORS AND ON THE PRECISION WITH WHICH THE PERTURBATIONS DUE TO THE EARTH'S SHAPE AND ATMOSPHERE ARE REPRESENTED. THE ACCURACY OF THE INITIAL VECTORS, $\vec{r}_S$ AND $\vec{v}_S$ IS AFFECTED BY MEASUREMENT ERRORS IN RADAR EQUIPMENT. THE PERTURBATIONS ARE APPROXIMATED USING MODELS OF THE EARTH'S SHAPE AND ATMOSPHERE BASED ON THE LATEST AVAILABLE INFORMATION. HENCE, TO ACCOUNT FOR THESE DEVIATIONS TO THE ELLIPTICAL ORBIT, THE PREDICTIONS MUST BE CONTINUALLY IMPROVED OR UPDATED USING RADAR OBSERVATIONS. THE METHOD WHICH ACCOMPLISHES THE UPDATING TASK IS CALLED DIFFERENTIAL CORRECTION.

THE NUMERICAL INTEGRATION TABLE IS GENERATED BY NOCPNI USING COWELL'S METHOD BASED ON THE POSITION AND

VELOCITY VECTORS, $\vec{r}_s$ AND $\vec{v}_s$ AT TIME t_s. THE SIX COMPONENTS OF BASE VECTORS ARE REFERENCED TO AN INERTIAL COORDINATE SYSTEM. HENCE, A SET OF VALUES $t_i, x_{1i}, x_{2i}, x_{3i}, x_{4i}, x_{5i}, x_{6i}$, AND i RANGES OVER THE ORBIT LIFETIME ARE OBTAINED. IN ADDITION, ASSUME THAT A SET OF OBSERVATIONS $t_{jk}, R_{jk}, A_{jk}, E_{jk}$ (OR, IN VECTOR NOTATION, t_{jk}, F_{jk}) IN LOCAL COORDINATES ARE AVAILABLE WHERE -

R_{jk} = k^{th} OBSERVATION OF RANGE AT THE jth RADAR STATION

A_{jk} = k^{th} OBSERVATION OF AZIMUTH AT THE jth RADAR STATION

E_{jk} = k^{th} OBSERVATION OF ELEVATION AT THE jth RADAR STATION

t_{jk} = TIME OF THE kth OBSERVATION AT THE jth RADAR STATION

AT ANY t_{jk}, WE CAN INTERPOLATE INTO THE NUMERICAL INTEGRATION TABLE AND OBTAIN A CORRESPONDING $\vec{r}$. KNOWING THE POSITION OF THE jth STATION AT t_{jk}, THEN R'_{jk}, A'_{jk}, E'_{jk} ARE OBTAINED WHICH IS THE PREDICTED k^{th} OBSERVATION AT THE jth STATION BASED ON $\vec{r}_s$ AND $\vec{v}_s$.

$\Delta F = F_{jk} - F'_{jk}$ REPRESENTS, EXCEPT FOR MEASUREMENT ERRORS, THE DIFFERENCE BETWEEN THE ACTUAL POSITION OF THE SPACECRAFT AT t_{jk} AND ITS PREDICTED POSITION. SINCE F IS A FUNCTION OF THE SIX INDEPENDENT VARIABLES x_{ms}(m= 1,... 6), FIRST-ORDER DIFFERENTIAL EFFECTS ARE CONSIDERED BY USE OF THE TOTAL DIFFERENTIAL (CALLED AN EQUATION OF CONDITION) -

$$dF_{jk} = \sum_{m=1}^{6} \frac{\partial F_{jk}}{\partial x_{ms}} \Delta x_{ms} = \Delta F_{jk}.$$

WHERE EACH Δx_{ms} IS AN INCREMENTAL CHANGE IN THE COR-

RESPONDING COMPONENT OF THE POSITION AND VELOCITY VECTORS $\vec{r}$ AND $\vec{v}$. (NOTE - THESE VECTORS ARE NOT CONSTANT THROUGH AN ORBIT, BUT ARE SHIFTED AT EACH OF THE RADAR STATIONS.) IN THIS EQUATION, THE PARTIAL DIFFERENTIAL COEFFICIENTS AND ΔF_{jk} ARE KNOWN AT EACH t_{jk}. HENCE, A SYSTEM OF EQUATIONS WITH SIX UNKNOWNS IS OBTAINED. THESE EQUATIONS ARE WEIGHTED TO ACCOUNT FOR MEASUREMENT ERRORS WHICH VARY WITH THE RADAR STATIONS AND THE INDIVIDUAL COMPONENTS OF F_{jk}. A SOLUTION IS THEN OBTAINED BY THE WEIGHTED LEAST-SQUARES TECHNIQUE. AFTER ESTABLISHING THAT THIS SOLUTION IS ACTUALLY AN IMPROVEMENT, A NUMERICAL INTEGRATION TABLE IS GENERATED ABOUT THE NEW $\vec{r}_s$ AND $\vec{v}_s$ AND THE PROCESS IS REPEATED UNTIL THE Δx_{ms_u} ARE INSIGNIFICANT.

48. DODIFC - DIFFERENTIAL CORRECTION CONTROL PROGRAM

DODIFC IS THE CONTROL PROGRAM FOR 14 PROGRAMS WHICH PERFORM THE DIFFERENTIAL CORRECTION FUNCTION IN THE GEMINI PROGRAMMING SYSTEM. (SEE FLOWCHART.)

INPUT

TO ACCOMPLISH ITS FUNCTION, DODIFC REQUIRES THE FOLLOWING PROGRAMS AND LIBRARY ROUTINES -

PROGRAMS -

D1PCDC	D2CSFE	D2SLNE
D2SSMT	D2SUNE	D2WEQT
D3CDRM	D3WTLU	D4CFRV
R1NDCP	R2AINT	

LIBRARY SUBROUTINES -

U7INTP	U1SQRT	UA1LSC

THE FOLLOWING TABLES SERVE AS INPUT TO THE DODIFC PROGRAM -

A) TMSTMS (SET UP BY MPDIFC OF DATA PASSES TESTS.)

STA	S	DEC	T	ADD
1	+ +	0	,0,	Loc. of Sta. Info. in TMDATA
	.			
	.			
2	. -			

TMDATA (SET UP BY MPDIFC AT TIME UPPER CORE IS WRITTEN)

STA	S	DEC	T	ADD	
1	+	Pass No.	, 0,	St. Ident. No.	2.35 = 75 cells 3 words for each block in upper core, block 1 in TMDATA , TMDATA +2
		Loc. of St. Mess.	, 0,	No. of Radar Mess	
		Blk. in upper core	, 0,		
		Fix Pt. Sec of t_{ca},	0, fix pt. min. of t_{ca}		
2					

S	TMVTCA (Set up by M) DIFC using QTSX and TDCDEF)	
1	R Fl. Pt. M. U.	7x length of TMSTMS table
	V Fl. Pt. M. U.	
	t_{ca} in add. to nearest whole fix pt. min.	
2		

B) TNINT1 - A BLOCK CONTAINING AN ORBIT INTEGRATION TABLE IN THE FOLLOWING FORMAT -

TNINT1		FORM OF TABLE - 0=MINUTES 1=SECONDS
+ 1	FIXED-POINT	INTERVAL AT WHICH ENTRIES ARE MADE IN THE TABLE
+ 2		THE STARTING TIME OF THE TABLE
+ 3		THE ENDING TIME OF THE

		TABLE
+ 4		X
+ 5		Y
+ 6	FLOATING POINT	Z
+ 7		$\dot{X}$
+ 8		$\dot{Y}$
+ 9		$\dot{Z}$

ETC.

C) TNINT2 - REENTRY TABLE, SAME FORMAT AS ABOVE.

D) THE $\vec{r}$, $\vec{v}$ AND t FROM THE LAST DIFFERENTIAL CORRECTION FOR THE PRESENT STATION ARE AVAILABLE TO DODIFC IN LOCATIONS TDRANV TO TDRANV + 6.

E) MONITOR SUPPLIES THE STATION CHARACTERISTICS BLOCK IN THE FORMAT SHOWN IN TABLE OF STATION CHARACTERISTICS BLOCK CONTENTS.

F) TMSTCH - A 37-WORD REFERENCE TABLE WITH THE ADDRESS - 1 OF THE STATION CHARACTERISTIC BLOCK FOR EACH STATION. EACH WORD IN THE BLOCK CONTAINS THE LOCATION OF THE WORD PRECEDING THE FIRST WORD IN THE DESIRED STATION BLOCK, I.E., -

TMSTCH	BLANK
+ 1	LOC - 1 OF STATION CHAR. BLOCK NO. 1
+ 2	LOC - 1 OF STATION CHAR. BLOCK NO. 2
+ 3	LOC - 1 OF STATION CHAR. BLOCK NO. 3

ETC.

G) STATION MESSAGE BLOCKS IN THE FOLLOWING FORMAT -

WORD 1 - STATION IDENTIFICATION NUMBER IN ADDRESS, PASS NUMBER IN DECREMENT.

WORD 2 - NUMBER OF RADAR MESSAGES IN THIS BLOCK.

WORD 3 - LOCATION, RELATIVE TO WORD 1, OF TIME FOR MINIMUM RANGE IN THIS BLOCK. (ADDRESS FIELD)

STATION CHARACTERISTICS BLOCK CONTENTS

Word No.						
1	Prefix: Site type (0 = AN/FPS-16; 1 = Verlort; 2 = telemetry) Decrement: Earth sector number Address: DCC input subchannel number					
2	Prefix: Other radars at this site (7 = AN/FPS-16; 1 = Verlort; 0 = none) Address: Internal station number					
3	DCC Output Subchannel mask (Subchannel 10=PZE 0, 0, 16; Subchannel 11 = PZE 0, 0, 8)					
4, 5	Station Name: (in BCD)					
	Item	Source	Source Format	Source Unit	Converted Format	Converted Unit
6	Geodetic longitude (λ)	Given	Dec.	DMS*	Flt. -Pt.	Radians
7	Geodetic latitude (ϕ)	Given	Dec.	DMS*	Flt. -Pt.	Radians
8	Geocentric latitude (ϕ')	Computed			Flt. -Pt.	Radians
9	sin ϕ	Computed			Flt. -Pt.	Radians
10	cos ϕ	Computed			Flt. -Pt.	Radians
11	$R_s \sin(\phi-\phi')$	Computed			Flt. -Pt.	Earth radii
12	$R_s \cos(\phi-\phi')$	Computed			Flt. -Pt.	Earth radii
13	Radius from earth center to sta.(R_s)	Computed			Flt. -Pt.	Earth radii
14	Altitude above ellipsoid (H)	Given	Dec.	Meters	Flt. -Pt.	Earth radii
15	Local vertical deflection Longitude ($\Delta\lambda$)	Given	Dec.	DMS*	Flt. -Pt.	Radians
16	Local vertical deflection Latitude ($\Delta\phi$)	Given	Dec.	DMS*	Flt. -Pt.	Radians
17	Calibrated boresight elevation	Given	Dec.	DMS*	Flt. -Pt.	Radians
18	Calibrated boresight azimuth	Given	Dec.	DMS*	Flt. -Pt.	Radians
19	Inertial longitude at reference time	Computed			Flt. -Pt.	Radians
20	Air pressure at ground level (P_q)	Given	Dec.	Millibars	Flt. -Pt.	Millibars
21	Temperature at ground level (T_g)	Given	Dec.	°C	Flt. -Pt.	°K
22	Water vapor content at ground level (e_g)	Given	Dec.	Millibars	Flt. -Pt.	Millibars
23	Azimuth deviation from true north	Given	Dec.	DMS*	Flt. -Pt.	Radians
24	Square root of weight 1 (range)	Given	Dec.		Flt. -Pt.	-
25	Square root of weight 2 (azimuth)	Given	Dec.		Flt. -Pt.	-
26	Square root of weight 3 (elevation)	Given	Dec.		Flt. -Pt.	-
27	Modulus (n-1)	Computed			Flt. -Pt.	-
28	R sin ($\phi-\phi'$)	Computed			Flt. -Pt.	Earth radii
29	R cos ($\phi-\phi'$)	Computed			Flt. -Pt.	Earth radii
30	sin ϕ'	Computed			Flt. -Pt.	Radians
31	cos ϕ'	Computed			Flt. -Pt.	Radians
32	Boresight elevation correction (C_3)**	Computed			Flt. -Pt.	Radians
33	Boresight azimuth correction (C_2)**	Computed			Flt. -Pt.	Radians
34	Open for expansion					

*DMS - degrees, minutes and seconds
**Original input - raw radar reading

WORD 4 - TIME (T_1) FIXED-POINT, MINUTES IN ADDRESS, SECONDS IN DECREMENT.
WORD 5 - RANGE (R_1) FLOATING-POINT IN GEMINI UNITS.
WORD 6 - AZIMUTH (A_1) FLOATING-POINT IN RADIANS.
WORD 7 - ELEVATION (E_1) FLOATING-POINT IN RADIANS.
WORD 8 - T_2
WORD 9 - R_2 2ND RADAR MESSAGE
WORD 10 - A_2
WORD 11 - E_2
AN ERRONEOUS MESSAGE IS INDICATED BY + 1.

H) MCNRRF

= 0, DURING ORBIT

$\neq$ 0, DURING REENTRY

I) TFESAB

THE FIRST EVEN SECOND AFTER BURNOUT IN FIXED POINT SECONDS B35.

OUTPUT

THE FOLLOWING ARE OUTPUT REQUIREMENTS OF DODIFC -

A) AT THE COMPLETION OF DODIFC, A TRANSFER IS MADE TO MFDIFC WITH THE ACCUMULATOR IN THE FOLLOWING FORMAT -

AC = 0 PERFORM NUMERICAL INTEGRATION
AC $\neq$ 0 DO NOT PERFORM NUMERICAL INTEGRATION
S(AC)= + DO NOT PERFORM ANOTHER DIFFERENTIAL CORRECTION UNTIL NEW DATA HAS BEEN COLLECTED.
S(AC)= - PERFORM ANOTHER DIFFERENTIAL CORRECTION BEFORE SUPPLYING NEW DATA.
1 IN P(AC) INPUT FOR NUMERICAL INTEGRATION IS IN

TNNDCI (FOR 6 NDC TABLES).

0 IN P(AC) INPUT FOR NUMERICAL INTEGRATION IS IN TDRANV (NORMAL SETTING).

B) TDRANV - IF INDICATION HAS BEEN MADE TO PERFORM AN INTEGRATION, THE 15-LOCATION TABLE SHOWN IN TDRANV TABLE FORMAT IS SET AS INPUT.

C) TDPRNT - 140 WORD OUTPUT BLOCK USED FOR PRINTING ON-LINE MESSAGES OF DIFFERENTIAL CORRECTION RESULTS (SEE TDPRNT TABLE FORMAT)

METHOD

BEFORE A NEW SOLUTION IS ACCEPTED, IT IS COMPARED WITH THE PREVIOUS SOLUTION TO DETERMINE ITS RELATIVE RELIABILITY. THE FOLLOWING METHOD IS USED TO DETERMINE IF THE NEW SOLUTION WILL BE ACCEPTED. FOR PURPOSES OF THIS DISCUSSION, IT IS ASSUMED THAT A SOLUTION EXISTS AT THE LAST ''ANCHOR POINT,'' AND THAT A NEW SOLUTION HAS BEEN OBTAINED AT THE NEW ''ANCHOR POINT.'' EACH SOLUTION HAS A ''STANDARD DEVIATION'' OR ''PROBABLE ERROR OF UNIT WEIGHT,'' AND EACH UNKNOWN HAS A WEIGHT, Q_i, DEFINED BY -

PROBABALE ERROR OF $W_i = \left[W_i\right] = \left[\text{UNIT WEIGHT}\right] \sqrt{Q_i} =$ (STANDARD DEVIATION) $\sqrt{Q_i}$

$\left[X_i\right]$ IS READ AS ''THE PROBABLE ERROR OF X_i.''

THE ACCURACY WITH WHICH ANY POSITION CAN BE PREDICTED IS -

$$\left[\bar{\rho}\right]^2 = \sum_{i=1}^{6} \left(\frac{\partial \bar{\rho}}{\partial W_i}\right)^2 \left[W_i\right]^2$$

IN THE PRESENT CASE, THE INDEPENDENT VARIABLES, W_i, ARE COMPONENTS OF $\vec{r}_0$ AND $\vec{v}_0$. SINCE,

$$\vec{r}\,(t_i) = f_i\,\vec{r}_0 + g_i\,\vec{v}_0 \text{ and } \vec{\rho} = \vec{r} = \vec{R} \text{ (approximately)}$$

THEN, AT THE NEW "ANCHOR POINT," [R] = 0

$$[\bar{\rho}]^2 = [\vec{r}]^2 = (Q_1 + Q_2 + Q_3)\,(\text{S. E.})^2$$

TDRANV TABLE FORMAT

Location	Contents	
TDRANV	X Coordinate	Floating point
+ 1	Y Coordinate	
+ 2	Z Coordinate	
+ 3	Velocity in X direction	
+ 4	Velocity in Y direction	
+ 5	Velocity in Z direction	
+ 6	Time; Fixed point units refer to + 9	
+ 7	1) TNINT1 if integration table desired 2) TNINT2 is re-entry table desired	
+ 8	Time stepping interval (integer); units refer to + 9	
+ 9	1) If bit position 35 = 1, interval is seconds 2) If bit position 35 = 0, interval is minutes	
+ 10	1) length of time to integrate backwards (integer) 2) zeros if no backward integration desired	
+ 11	1) length of time to integrate forward 2) zeros if no forward integration desired	
+ 12	1) zeros; use N4DRAG program 2) $\neq$ zero; do not use N4DRAG program	
+ 13	1) + 1, $\bar{r}$ and $\bar{v}$ in output table 2) zero; $\bar{r}$ values only in output table	
+ 14	n number of integration steps per output	

TDPRNT TABLE FORMAT
SHEET 1 OF 2

Location	Contents
Word #1 Table	000 000 000 0XX (where XX is the number of stations giving data)
TDPRNT + 1 + 2 + 3	ΔR
+ 4 + 5 + 6	ΔV
+ 7	Sigma: standard error of fit
+ 8	K_{-1}
+ 9	K_0
+ 11 + 12	$\dfrac{\sqrt{\sum_{i=1}^{3} \left(S_e \sqrt{Q_i}\right)^2}}{\sqrt{\sum_{i=4}^{6} \left(S_e \sqrt{Q_i}\right)^2}}$
+ 13 . . . + 26	Internal Station Numbers
+ 27 . . . + 41	Equations per station 1st Edit (in the same order as station numbers)
+ 42 . . . + 56	2nd Edit

TDPRNT TABLE FORMAT
SHEET 2 OF 2

Location	Contents
TDPRNT +57	3rd Edit
+71	
+72	4th Edit
+86	
+87	5th Edit
+101	
+102	Residual/M
+116	
+117	A, E, I
+119	
+120	R
+122	
+123	V
+125	
+126	GMT associated with above R, V vectors in fixed point seconds.
+127	Blank
+140	

Note: If the first word in the above table is 0, no information will be printed.

HOWEVER, THIS IS INDEPENDENT OF $\left[\vec{v}_0\right]$ BECAUSE $g_0 = 0$.
THE FOLLOWING CRITERION IS USED AND IMPLIES THAT f =1= g -

$$K_0^2 = (S.E.)_0^2 \sum_1^6 Q_i = \sum_1^6 \left(S_e \sqrt{Q_i}\right)^2$$

DURING ORBIT, THIS FIGURE, ASSOCIATED WITH THE MOST RECENT SOLUTION, CHARACTERIZES THE QUALITY OF THE PREDICTIONS AT A TIME WHICH IS APPROXIMATELY 15 MINUTES LATER, THAN THE MOST RECENT ''ANCHOR POINT,'' I.E., $t_i = t_0$ +15 MIN. IN THE REENTRY PHASE, $t_i = T_0$, $g = 0$. THIS PRODUCES-

$$K_0^2 = \sum_1^3 \left(S_e \sqrt{Q_i}\right)^2.$$

USING THE SAME t_i , A FIGURE OF MERIT IS NOW DERIVED FROM THE SOLUTION AT THE LAST ''ANCHOR POINT'' TO BE COMPARED WITH THE K_0 GIVEN ABOVE. IT IS ASSUMED THAT AN OBSERVATION HAS BEEN MADE AT t_i , AND THEREFORE, THE EQUATIONS OF CONDITION WOULD HAVE TO BE COMPUTED AT THIS POINT. IN THE COURSE OF SUCH A CALCULATION, THE FOLLOWING EQUATION WOULD BE DERIVED -

$$\partial \vec{r} = \left[A_{ij}\right] \text{ , a 3x6 matrix.}$$

THEN SET

$$K^2_{-1} = (S.E.)^2_{-1} \sum_{j=1}^{6} \left(A_{1j}^2 + A_{2j}^2 + A_{3j}^2\right) Q_j.$$

THIS REPRESENTS $[\bar{\rho}]^2$ AT t_i , BASED UPON THE PREVIOUS SOLUTION. THEREFORE, IF $K_0 > b\ K_{-1}$ THE NEW SOLUTION IS REJECTED WHERE b IS SOME FACTOR FOR DETERMINING CONVERGENCE.

USAGE

A) CALLING SEQUENCE -

ALPHA TRA DODIFC

DODIFC RETURNS CONTROL TO MONITOR AT LOCATION MFDIFC.

B) SPACE REQUIRED - 629 DECIMAL.

C) ERROR RETURNS - IF AN ERROR OCCURRED WHILE PROCESSING A SINGLE OBSERVATION AT t_i , THE OBSERVATION t_i IS NEGLECTED AND THE PROGRAM ATTEMPTS TO CONTINUE. IN ALL OTHER CASES, DODIFC EXITS WITH AN INDICATION OF A REJECTED CORRECTION BUT NO PRINTING WILL RESULT ON-LINE (TDPRNT 0).

D) CONSTANTS -

```
KOZERO  DEC  0              FIXED-POINT 0
KM0000  MZE                 FIXED-POINT MINUS 0
K00001  DEC  1              FIXED-POINT 5
K00005  DEC  5              FIXED-POINT 5
K00018  DEC  18             FIXED-POINT 18
K00030  DEC  30             FIXED-POINT 30
K00060  DEC  60             FIXED-POINT 60
K00090  DEC  90             FIXED-POINT 90
K000.5  DEC  0.5            FLOATING-POINT .5
K0060.  DEC  60.            FLOATING-POINT 60.0
K01.E3  DEC  1000.          FLOATING-POINT 1000.
K00.SH  DEC  .12394486 E-2
KCH233  OCT  233 000 000 000
```

E) TABLES -

```
TAWTPC  BSS  88             WEIGHTED PARTIALS
                            MAX. NUMBER OF LOCA-
                            TIONS = C(K000N) *
                            C(K000MB).
TCORBP  BSS  16             NEAR CIRCULAR ORBIT
                            PARAMETER FUNCTIONS.
TD2SSM  BSS  15             SUM (DF**2)/M FOR
                            EACH STATION.
TDEQCT  BSS  24             THE PARTIAL COEFFI-
```

			CIENTS.
TDFNDC	PZE	-= NO K-1 CALC. += CALC. K-1	INDICATOR TO STOP K-1 CALCULATION AND COMPARING DURING NUMERICAL DIFFERENTIAL CORRECTION METHOD.
TDNIET	PZE		C(TNETIM-(2(TNINT2+1)) IN REENTRY.
TDNOBR	BSS	10	NUMBER OF OBSERVATIONS PROCESSED FOR EACH STATION IN DIFFERENTIAL CORRECTION INTERVAL IN REENTRY
TDRANV	SYN	KLAMDO + 1	15 LOCATIONS OF NUMERICAL INTEGRATION INPUT BLOCK.
TDTIMT	PZE		INDICATOR TO CALCULATE VALUE FOR END OF NUMERICAL INTEGRATION TABLE.
TDTOBS	PZE		TIME OF OBSERVATION FIXED-POINT SECONDS, REENTRY.
TEQCIN	BSS	27	INTERMEDIATE INPUT OR OUTPUT BLOCK.
TFESAB	PZE		FIRST SECOND AFTER BURNOUT IN FIXED-POINT
TLSSSE	BSS	17	SOLUTIONS TO LEAST SQUARES, SE, SE OF ELEMENTS.
TMRARF	BSS	15	INPUT TABLE FOR NUMERICAL INTEGRATION FOR

```
                                        FIRST TIME IN REENTRY.
TMSTCH  BSS  37                         REFERENCE TABLE FOR
                                        STATION CHARACTER-
                                        ISTICS TABLE.
TMSTMS  BSS  15                         REFERENCE TO RADAR
                                        MESSAGE BLOCKS.
TN4NEX  PON                             NONZERO EQUALS STOP
                                        NUMERICAL INTEGRATION
                                        AT 60,000 FT.  ZERO
                                        EQUALS CONTINUE NUMER-
                                        ICAL INTEGRATION.
TNDCIN  PZE                             ZERO EQUALS D1PCDE,
                                        NOT ZERO EQUALS NUMER-
                                        ICAL DIFFERENTIAL
                                        CORRECTION METHOD FOR
                                        PARTIALS.
TNEARM  PZE                             ANCHOR POINT FOR
                                        REENTRY IN FIXED-POINT
                                        SECONDS.
TNETIM  PZE                             TIME OF LAST ENTRY IN
                                        NUMERICAL INTEGRATION
                                        TABLE.
TNINT1  BSS  2200                       ORBIT INTEGRATION
                                        TABLE.
TNINT2  BSS  1800                       REENTRY INTEGRATION
                                        TABLE.
TNOMFS  BSS  15                         NUMBER OF OBSERVATIONS
                                        DESIRED FROM EACH
                                        STATION REPORTING.
TNSCOR  DEC  1,1,1,2,2,3,4,5,6,6,7,7,8,9,10,11,
                                        11,2,2
TNUMNI  PZE  0                          TEST FOR ALL 6 TABLES
```

				SET UP FOR NUMERICAL DIFFERENTIAL CORRECTION (0= NOT USE NUMERICAL DIFFERENTIAL CORRECTION, 6= ALL SET UP).
	TODCHC	PZE		TIME AT 450,000 FT. USED BY DIFFERENTIAL CORRECTION WITH MINUTES IN THE ADDRESS, SECONDS IN DECREMENT.
	TPWTDF	BSS	11	W(DF) MAXIMUM NUMBER OF LOCATIONS (K000MB).
	TRVTAP	BSS	13	ANCHOR POINT BLOCK (SEE SYN CARDS FOR FORMAT).
	TSTWTS	BSS	33	WEIGHT WORD TABLE.
	TSVSEE	BSS	6	SE -THE SIX STANDARD ERRORS OF THE ELEMENTS SAVED FROM THE PREVIOUS CORRECTION.

F) CONSTANTS -

	TEQCIN	BSS	27
KOSWTR	SYN	TEQCIN	SQUARE ROOT OF STATION WEIGHT FOR R.
KOSWTA	SYN	KOSWTR + 1	SQUARE ROOT OF STATION WEIGHT FOR A.
KOSWTW	SYN	KOSWTR + 2	SQUARE ROOT OF STATION WEIGHT FOR E.
KOOTOB	SYN	KOSWTR + 3	TIME OBSERVED.
KRVECX	SYN	KOOTOB + 1	R_X
KRVECY	SYN	KOOTOB + 2	R_Y

KRVECZ	SYN	KOOTOB + 3	R_Z
KSTCHL	SYN	KOOTOB + 4	LOCATION OF THE STATION CHARACTERISTICS BLOCK FOR STATION PROCESSED.
KTSUBI	SYN	KOSWTR + 8	t_i IN GEMINI UNITS (FLOATING-POINT).
KOOROB	SYN	KOSWTR + 9	RANGE OBSERVED.
KOOAOB	SYN	KOSWTR + 10	AZIMUTH OBSERVED.
KOOEOB	SYN	KOSWTR + 11	ELEVATION ANGLE OBSERVED.
KVVECX	SYN	KOSWTR + 12	V_X
KVVECY	SYN	KOSWTR + 13	V_Y
KVVECZ	SYN	KOSWTR + 14	V_Z
KOODRC	SYN	KOSWTR + 15	CALCULATED RANGE.
KOODAC	SYN	KOSWTR + 16	CALCULATED AZIMUTH.
KOODEC	SYN	KOSWTR + 17	CALCULATED ELEVATION ANGLE.
KMATRX	SYN	KOSWTR + 18	M MATRIX (USES 9 LOCATIONS).
		TDCCNT	BSS 24
KCCNTR	SYN	TDCCNT	COUNTS NUMBER OF TIMES IN LARGE LOOP.
KSRIND	SYN	TDCCNT + 1	MINUS = NO NEW STATION CAME IN.
KRESID	SYN	TDCCNT + 2	INDICATES TYPE OF RESIDUAL CHECK.
KD2CNT	SYN	TDCCNT + 3	INDICATES 1ST TIME IN D2SSMT.
KD2SUT	SYN	TDCCNT + 4	INDICATES 1ST TIME IN D2SUNE.
KDCSTT	SYN	TDCCNT + 5	INDICATES 1ST TIME IN

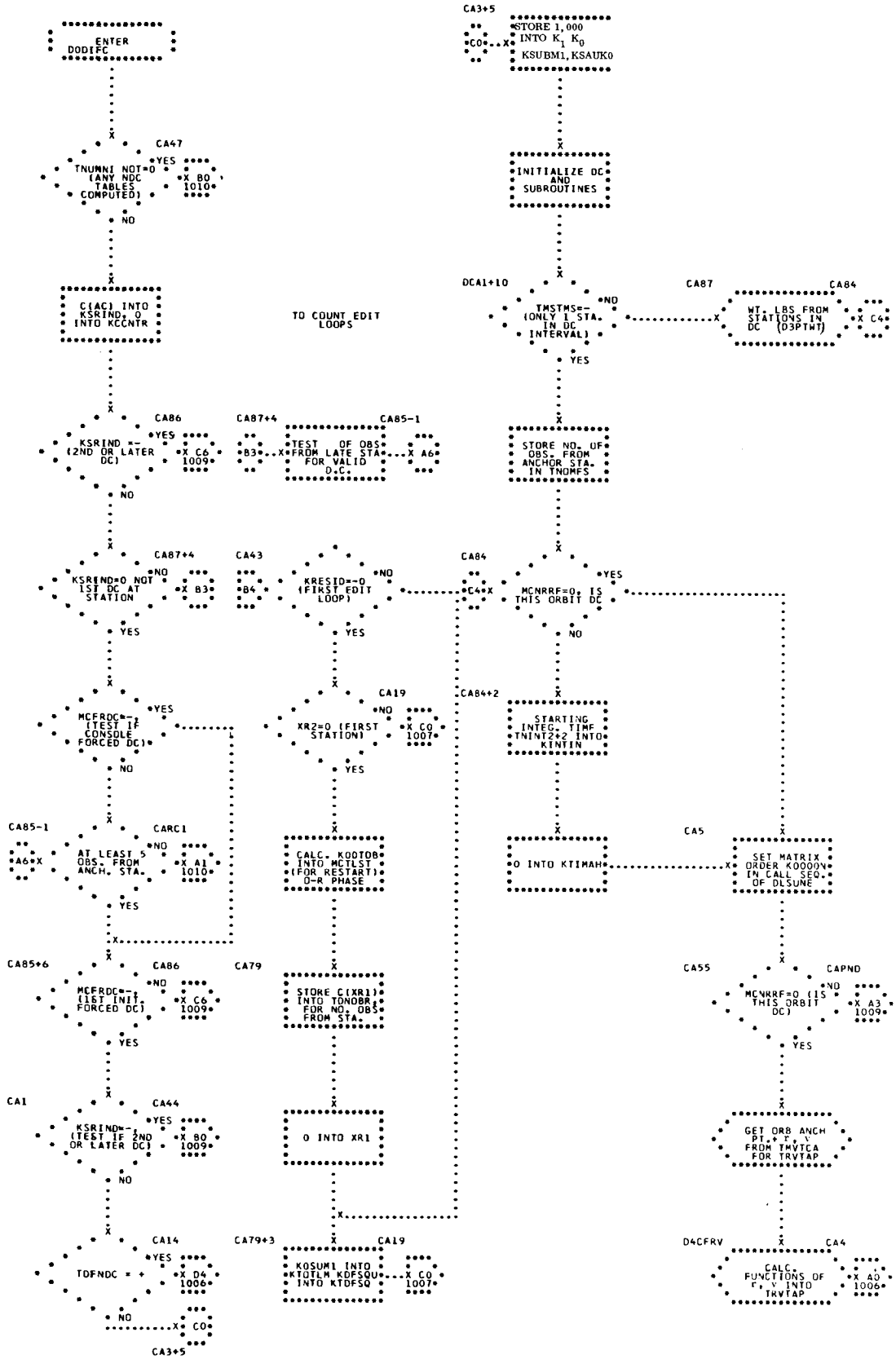
```
                                        D2WEQT - SET = 1.
KCDRMI   SYN   TDCCNT + 6               INDICATOR FOR D3CDRM
                                        - SET = 1 FOR EACH
                                        STATION PROCESSED IN
                                        INTERVAL.
KOLSMB   SYN   TDCCNT + 7               CONTAINS LOCATION OF
                                        STATION MESSAGE BLCOK.
KONDRM   SYN   TDCCNT + 8               NUMBER OF OBSERVATIONS
                                        TO BE TAKEN FROM EACH
                                        STATION.
KOOOON   SYN   TDCCNT + 9               NUMBER OF PARTIALS, 6.
KOOMCT   SYN   TDCCNT + 10              NUMBER OF EQUATIONS
                                        BOUND FOR D2SUNE.
KOSUMM   SYN   TDCCNT + 11              TOTAL NUMBER OF EQUA-
                                        TIONS FOR LEAST
                                        SQUARES SOLUTION.
KSUMWT   SYN   TDCCNT + 12              SUM OF THE WEIGHTS.
KDFSQU   SYN   TDCCNT + 13              SUM (DF) SQUARED.
KSAVSE   SYN   TDCCNT + 14              SAVED FROM PREVIOUS
                                        CORRECTION STANDARD
                                        ERROR.
KSEMPY   SYN   TDCCNT + 15              SE TIMES RESIDUAL
                                        CHECK FACTOR.
KINTIM   SYN   TDCCNT + 16
KRETBL   SYN   TDCCNT + 17              LOCATION TO INDEX
                                        TMORES TABLE FOR
                                        D2WEQT.
KTOTLM   SYN   TDCCNT + 18              ALL EQUATIONS FROM
                                        STATIONS IN INTERVAL
                                        USING ALL OBSERVATIONS
                                        NOT JUST DESIRED.
KTDFSQ   SYN   TDCCNT + 19              SUM W (DF) SQUARED FOR
```

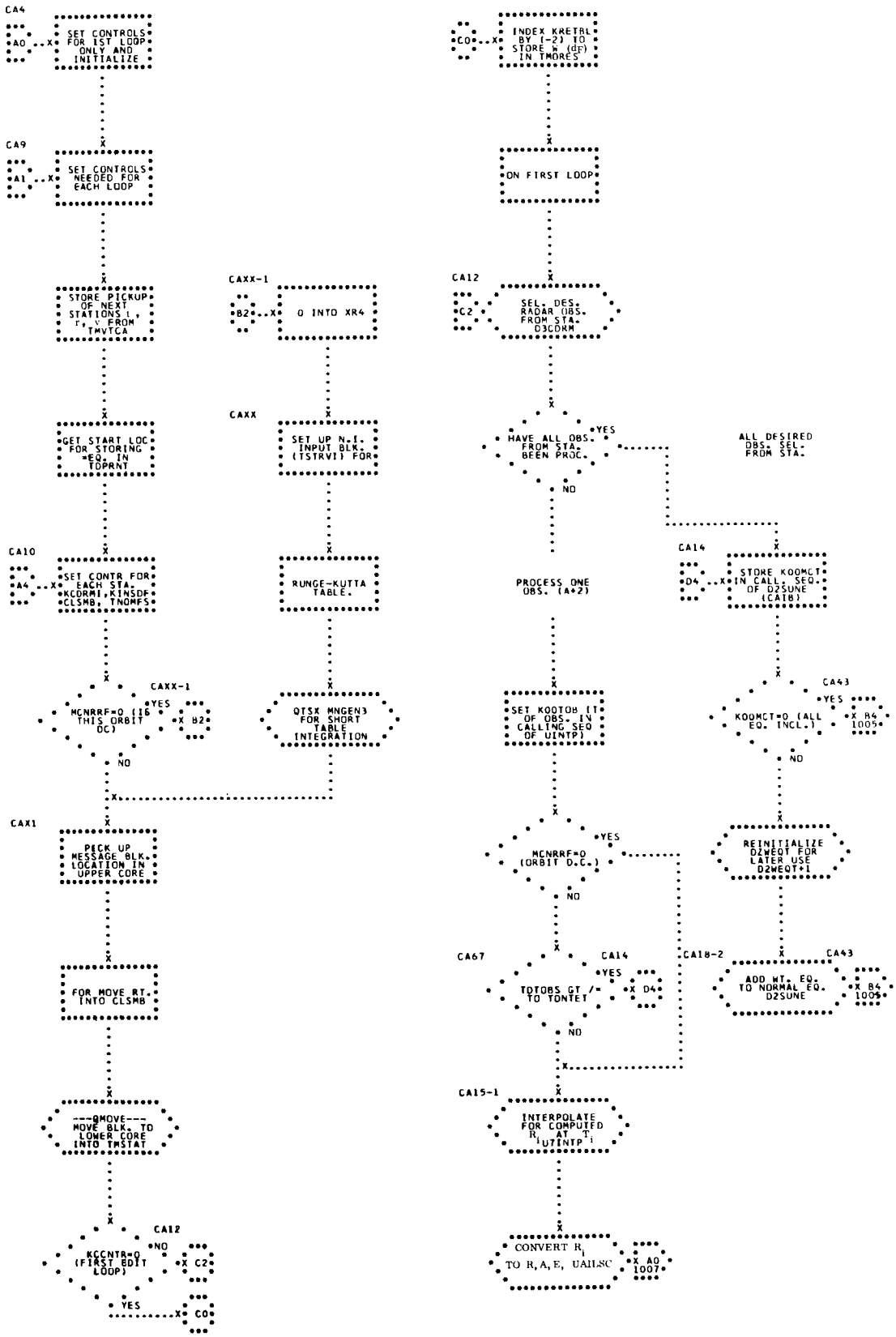
```
                                        STATION IN INTERVAL
                                        USING ALL OBSERVATIONS
                                        NOT JUST DESIRED.
KINSDF   SYN   TDCCNT + 20   INDICATOR TO D2WEQT TO
                             CALCULATE RESIDUALS.
KSUBM1   SYN   TDCCNT + 21   K-1
KSAVKO   SYN   TDCCNT + 22   K0 FROM LAST
                             CORRECTION.
KNEWKO   SYN   TDCCNT + 23   K0
               TSETCT  BSS   11
KLASTS   SYN   TSETCT        FACTOR FOR K-1 COMPA-
                             RISON.
KSAMES   SYN   TSETCT + 1    FACTOR FOR K0 COMPA-
                             RISON.
KWTIND   SYN   TSETCT + 2    ZERO - USE D3WTLU
                             (WEIGHT TABLE).
KNOBBD   SYN   TSETCT + 3    TOTAL NUMBER OF OBSER-
                             VATIONS DESIRED FROM
                             STATION.
KCPRVI   SYN   TSETCT + 4    ZERO = RV
KOOOMB   SYN   TSETCT + 5    UPPER BOUND OF TPWTDF
                             BLOCK.
KCCNBD   SYN   TSETCT + 6    DESIRED NUMBER OF
                             TIMES FOR DIFFERENTIAL
                             CORRECTION EDIT LOOPS.
KSEFTR   SYN   TSETCT + 7    RESIDUAL CHECK FACTOR
                             IN FLOATING POINT.
KTIMAH   SYN   TSETCT + 8    NUMBER AHEAD TO CAL-
                             CULATE K-1 .
KPERST   SYN   TSETCT + 9    NUMBER OF INTEGRATION
                             STEPS OUTPUT INTERVAL.
KRDELT   SYN   TSETCT + 10   OUTPUT INTERVAL FOR
```

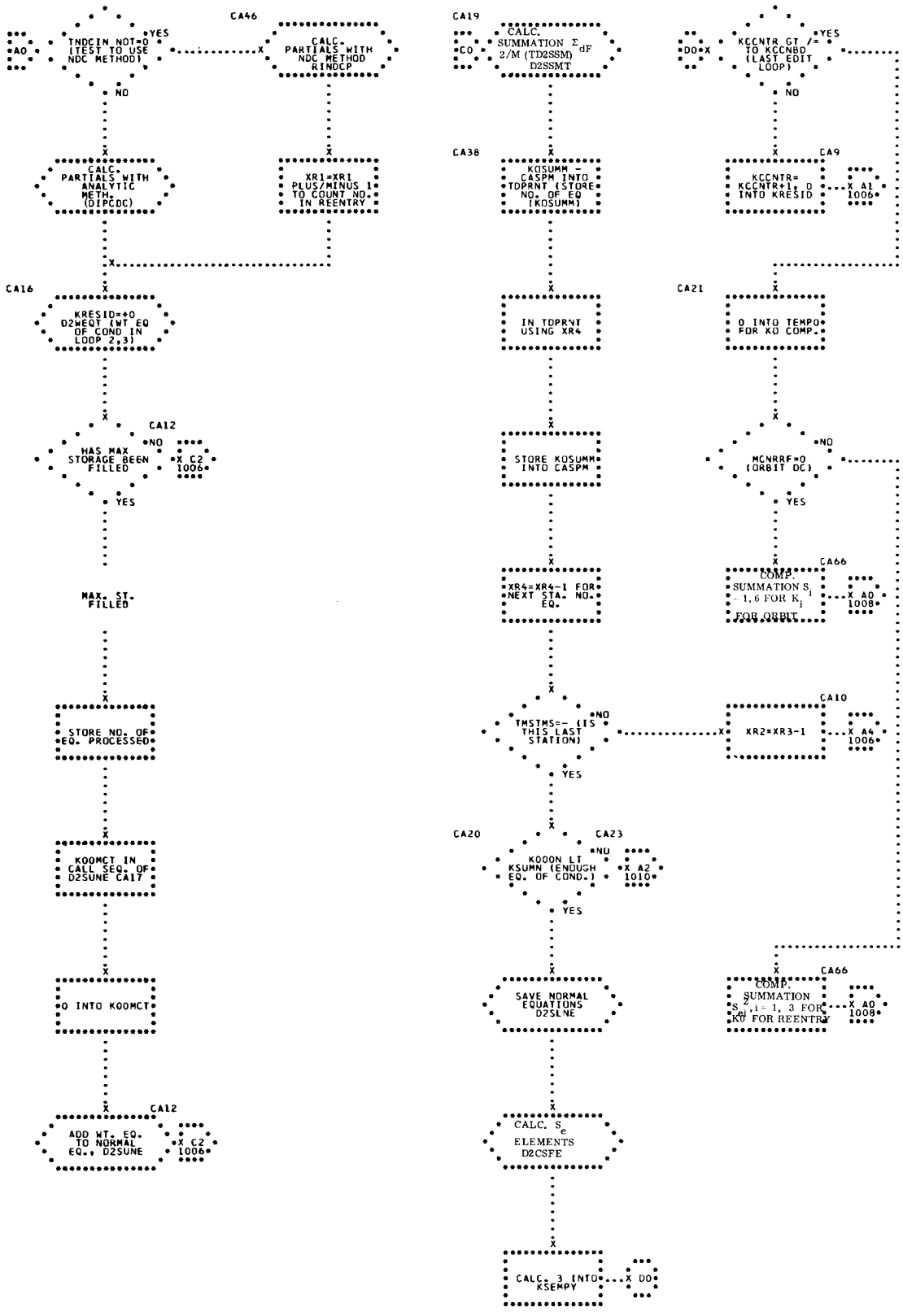
NUMERICAL INTEGRATION.

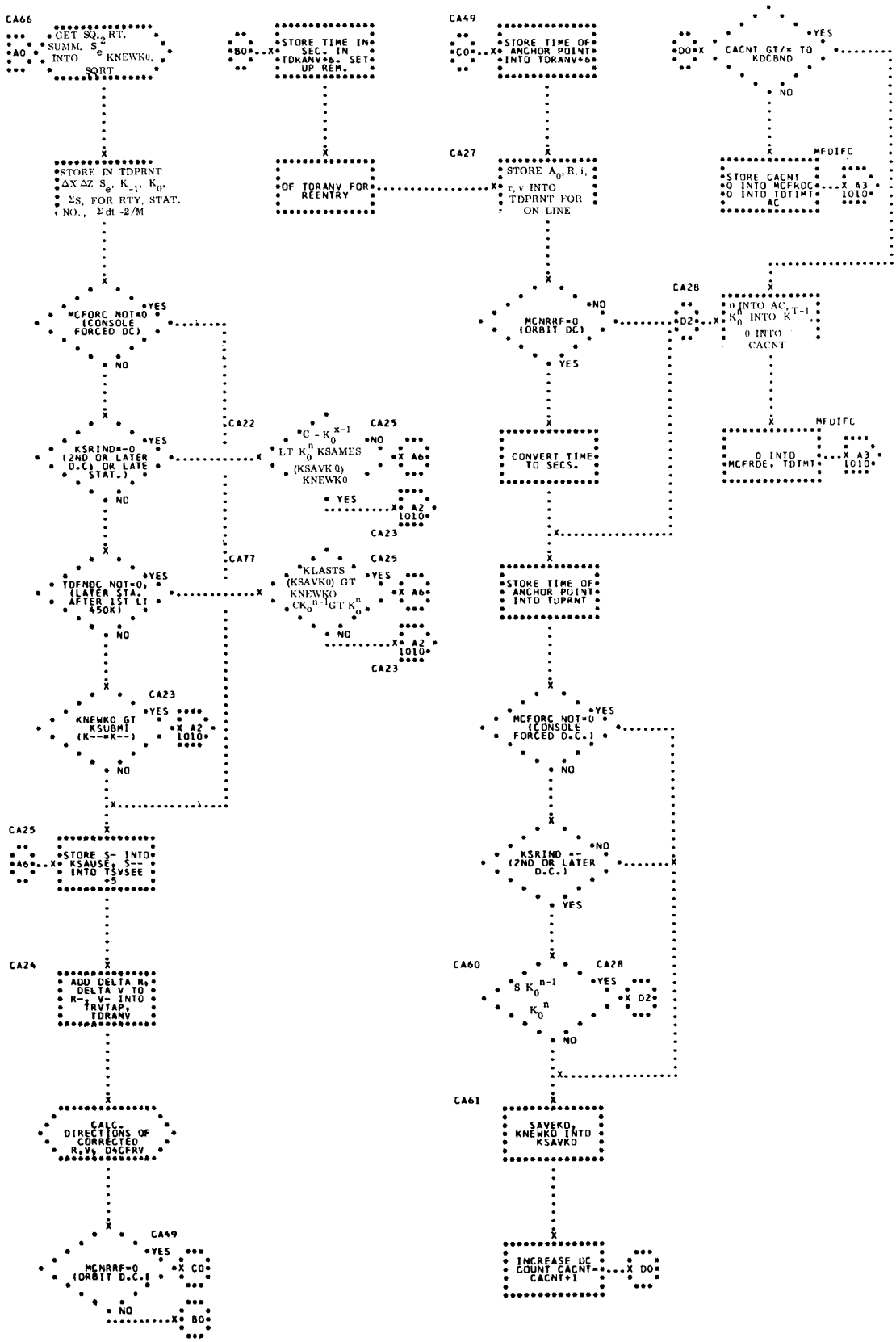
KDCBND SYN TSETCT + 11 BOUND FOR NUMBER OF DIFFERENTIAL CORRECTIONS PER STATION.

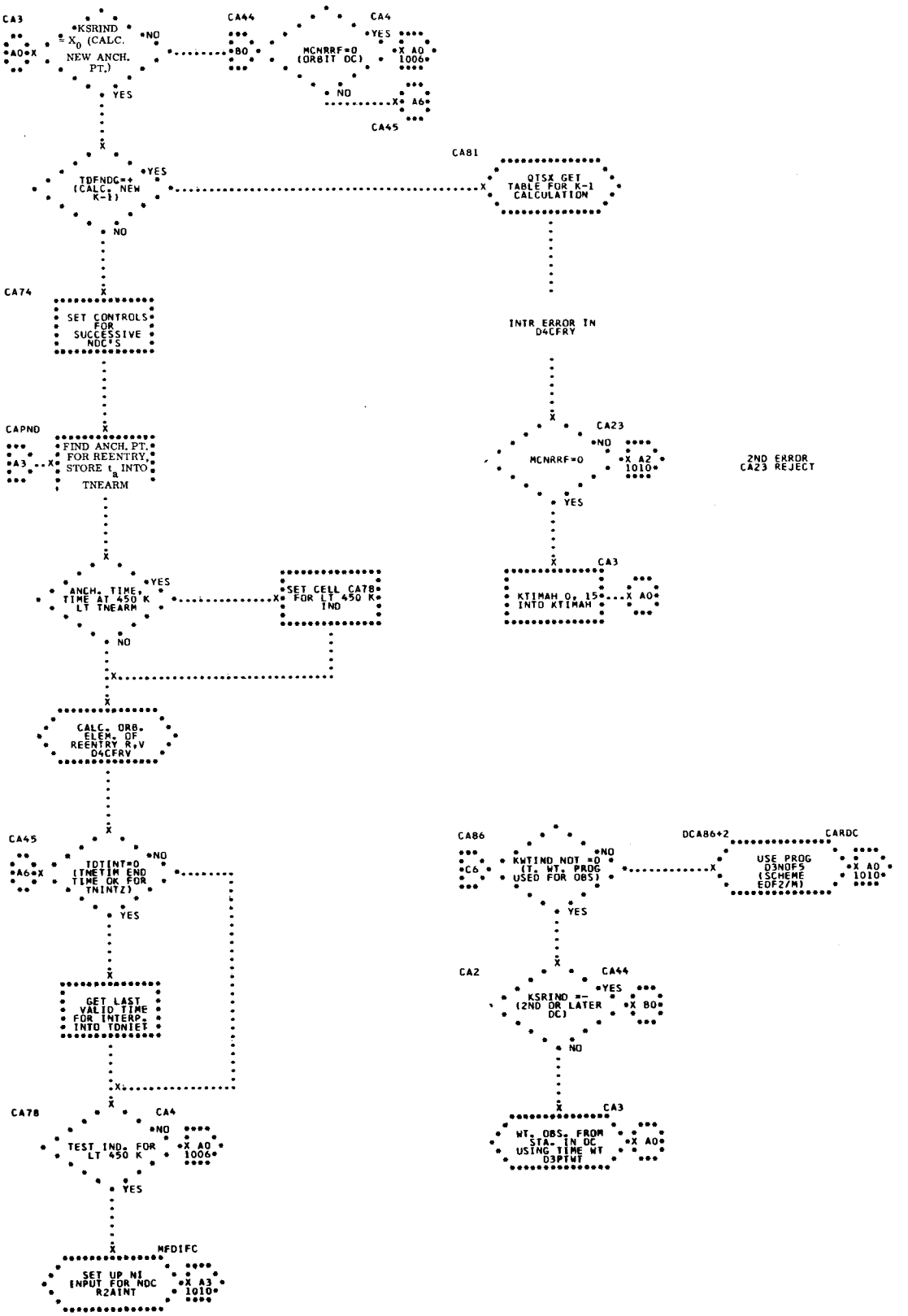
G) CHECKOUT - DODIFC HAS BEEN OPERATING IN THE GEMINI PROGRAM SYSTEM SINCE APRIL, 1964

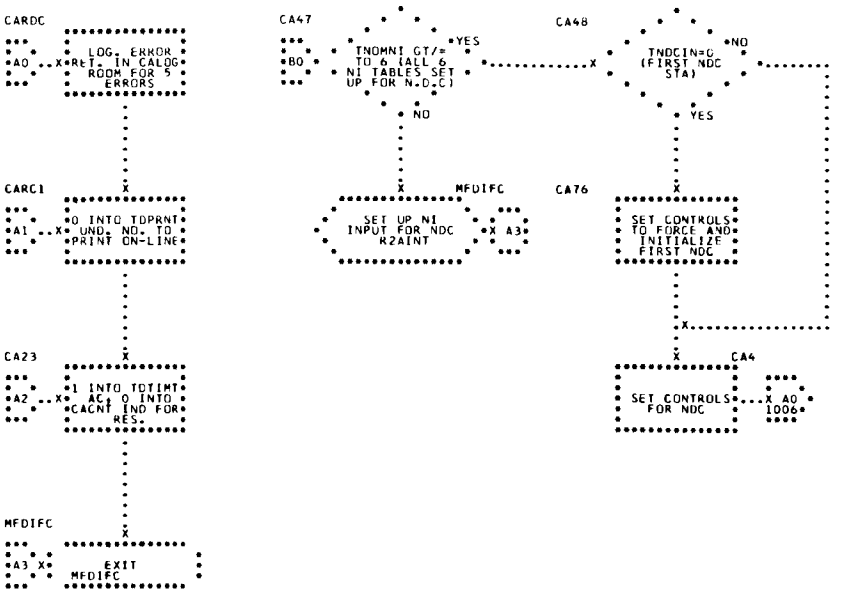

ENTER DODIFC
CA47
TNUMNI NOT=0 (ANY NDC TABLES COMPUTED)
YES
X BO 1010
NO
C(AC) INTO KSRIND, 0 INTO KCCNTR
TO COUNT EDIT LOOPS
CA86
KSRIND = (2ND OR LATER DC)
X C6 1009
CA87+4
CA85-1
TEST OF OBS FROM LATE STA FOR VALID D.C.
B3
X A6
KSRIND=0 NOT 1ST DC AT STATION
X B3
CA43
KRESID=-0 (FIRST EDIT LOOP)
B4
MCFRDC=- (TEST IF CONSOLE FORCED DC)
CA19
XR2=0 (FIRST STATION)
X C0 1007
CA85-1
AT LEAST 5 OBS. FROM ANCH. STA.
X A1 1010
A6
CARC1
CALC. KOOTOB INTO MCTLST (FOR RESTART) 0-R PHASE
CA85+6
CA86
MCFRDC=- (1ST INIT. FORCED DC)
X C6 1009
CA79
STORE C(XR1) INTO TONOBR FOR NO. OBS FROM STA.
CA1
CA44
KSRIND=- (TEST IF 2ND OR LATER DC)
X B0 1009
0 INTO XR1
CA14
TDFNDC = +
X D4 1006
X C0 1007
CA3+5
CA79+3
KOSUM1 INTO KTOTLM KDFSQU INTO KTDFSQ
CA3+5
STORE 1,000 INTO K1 K0
KSUBM1,KSAUK0
INITIALIZE DC AND SUBROUTINES
DCA1+10
TMSTMS=- (ONLY 1 STA. IN DC INTERVAL)
NO
YES
CA87
CA84
WT. LBS FROM STATIONS IN DC (D3PTWT)
X C4
STORE NO. OF OBS. FROM ANCHOR STA. IN TNOMFS
CA84
MCNRRF=0, IS THIS ORBIT DC
YES
NO
CA84+2
STARTING INTEG. TIME TNINT2+2 INTO KINTIN
0 INTO KTIMAH
CA5
SET MATRIX ORDER K0000N IN CALL SEQ. OF DLSUNE
CA55
CAPND
MCNRRF=0 (IS THIS ORBIT DC)
X A3 1009
GET ORB ANCH PT. F, V FROM TMVTCA FOR TRVTAP
D4CFRV
CA4
CALC. FUNCTIONS OF F, V INTO TRVTAP
X A0 1006


CA4
A0
SET CONTROLS FOR 1ST LOOP ONLY AND INITIALIZE
CA9
A1
SET CONTROLS NEEDED FOR EACH LOOP
STORE PICKUP OF NEXT STATIONS I, r, V FROM TMVTCA
GET START LOC FOR STORING =EQ. IN TOPRNT
CA10
A4
SET CONTR FOR EACH STA. KCDRMI, KINSDF CLSMB, TNOMES
CAXX-1
YES
MCNRRF=0 (1ST THIS ORBIT DC)
B2
NO
CAX1
PICK UP MESSAGE BLK. LOCATION IN UPPER CORE
FOR MOVE RT. INTO CLSMB
---QMOVE---
MOVE BLK. TO LOWER CORE INTO TMSTAT
CA12
NO
KCCNTR=0 (FIRST BDIT LOOP)
C2
YES
C0
CAXX-1
B2
0 INTO XR4
CAXX
SET UP N.I. INPUT BLK. (TSTRVI) FOR
RUNGE-KUTTA TABLE.
QTSX MNGEN3 FOR SHORT TABLE INTEGRATION
C0
INDEX KRETRL BY (-2) TO STORE R (dP) IN TMORES
ON FIRST LOOP
CA12
C2
SEL. DES. RADAR OBS. FROM STA. D3CORM
HAVE ALL OBS. FROM STA. BEEN PROC.
YES
NO
ALL DESIRED OBS. SEL. FROM STA.
PROCESS ONE OBS. (A*2)
CA14
D4
STORE KOOMCT IN CALL SEQ. OF D2SUNE (CA18)
SET KOOTOB (1ST OF OBS. IN CALLING SEQ OF UINTP)
CA43
YES
KOOMCT=0 (ALL EQ. INCL.)
B4
100$
NO
MCNRRF=0 (ORBIT D.C.)
YES
NO
REINITIALIZE D2WEQT FOR LATER USE D2WEQT+1
CA67
CA14
TOTOBS GT /= TO TONTET
YES
D4
NO
CA18-2
CA43
ADD HT. EQ. TO NORMAL EQ. D2SUNE
B4
100$
CA15-1
INTERPOLATE FOR COMPUTED R_i AT T_{UTINTP}
CONVERT R_i TO R,A,E, UAILSC
A0
100$


CA46
TNDCIN NOT=0 (TEST TO USE NDC METHOD)
YES
NO
CALC. PARTIALS WITH NDC METHOD RINDCP
CALC. PARTIALS WITH ANALYTIC METH. (DIPCDC)
XR1=XR1 PLUS/MINUS 1 TO COUNT NO. IN REENTRY
CA16
KRESID=+0 D2WEQT (WT EQ OF COND IN LOOP 2,3)
CA12
HAS MAX STORAGE BEEN FILLED
NO
X C2 1006
YES
MAX. ST. FILLED
STORE NO. OF EQ. PROCESSED
KOOMCT IN CALL SEQ. OF D2SUNE CA17
0 INTO KOOMCT
CA12
ADD WT. EQ. TO NORMAL EQ., D2SUNE
X C2 1006
CA19
CALC. SUMMATION Σ dF 2/M (TD2SSM) D2SSMT
CA38
KOSUMM = CASPM INTO TDPRNT (STORE NO. OF EQ IKOSUMM)
IN TDPRNT USING XR4
STORE KOSUMM INTO CASPM
XR4=XR4-1 FOR NEXT STA. NO. EQ.
TMSTMS=-(IS THIS LAST STATION)
NO
YES
XR2=XR3-1
X A4 1006
CA20
KOOON LT KSUMN (ENDUSH EQ. OF COND.)
CA23
NO
X A2 1010
YES
SAVE NORMAL EQUATIONS D2SLNE
CALC. Se ELEMENTS D2CSFE
CALC. 3 INTO KSEMPY
X DO
DO
KCCNTR GT /= TO KCCNBD (LAST EDIT LOOP)
YES
NO
KCCNTR= KCCNTR+1, 0 INTO KRESID
CA9
X A1 1006
CA21
0 INTO TEMPO FOR KO COMP.
MCNRRF=0 (ORBIT DC)
NO
YES
CA66
COMP. SUMMATION S_j, j = 1, 6 FOR K_j^1 FOR ORBIT
X A0 1008
CA10
XR2=XR3-1
X A4 1006
CA66
COMP. SUMMATION S_i, i = 1, 3 FOR K_0^1 FOR REENTRY
X A0 1008


CA66
GET SQ, RT.
SUMM. S_e INTO KNEWK0, SQRT.
STORE IN TDPRNT
ΔX ΔZ S_e, K_{-1}, K_0, ΣS, FOR RTY, STAT. (NO.), Σdt -2/M
MCFORC NOT=0 (CONSOLE FORCED DC)
YES
NO
KSRIND=-0 (2ND OR LATER D.C. OR LATE STAT.)
CA22
CA77
TDFNDC NOT=0, (LATER STA. AFTER 1ST LT 450K)
KNEWK0 GT KSU8MI (K--=K--)
CA23
CA25
STORE S- INTO KSAUSE, S-- INTO TSVSEE +5
CA24
ADD DELTA R_i, DELTA V TO R-f V- INTO TRVTAP, TDRANV
CALC. DIRECTIONS OF CORRECTED R, V, D4CFRV
CA49
MGNRRF=0 (ORBIT D.C.)
CO
BO
STORE TIME IN SEC. IN TDRANV+6, SET UP REM.
OF TDRANV FOR REENTRY
$C - K_0^{x-1}$ LT K_0^n KSAMES (KSAVK0) KNEWK0
A6
A2 1010
K_LASTS (KSAVK0) GT KNEWK0 CK_0^{n-1} GT K_0^n
CA27
STORE A_0, R_i, r_i, v INTO TDPRNT FOR ON-LINE
MGNRRF=0 (ORBIT DC)
CONVERT TIME TO SECS.
STORE TIME OF ANCHOR POINT INTO TDPRNT
MCFORC NOT=0 (CONSOLE FORCED D.C.)
KSRIND =- (2ND OR LATER D.C.)
CA60
$S K_0^{n-1}$ / K_0^n
CA28
D2
CA61
SAVEK0, KNEWK0 INTO KSAVK0
INCREASE DC COUNT CACNT=CACNT+1
DO
STORE TIME OF ANCHOR POINT INTO TDRANV+6
CACNT GT/= TO KDCBND
STORE CACNT 0 INTO MCFROG 0 INTO TDTIMT AC
MFDIFC
A3 1010
0 INTO AC, K_0^n INTO K_0^{n-1} 0 INTO CACNT
0 INTO MCFRDE, TDTMT


CA3
*KSRIND = X_0 (CALC. NEW ANCH. PT.)
*NO
CA44
*BO
CA4
*YES
MCNRRF=0 (ORBIT DC)
X A0 1006
*NO
X A6
CA45
*YES
TDFNDG=+ (CALC. NEW K-1)
*YES
CA81
QISX GET TABLE FOR K-1 CALCULATION
*NO
CA74
SET CONTROLS FOR SUCCESSIVE NDC'S
INTR ERROR IN D4CFRV
CAPND
*A3
FIND ANCH. PT. FOR REENTRY, STORE t_a INTO TNEARM
CA23
MCNRRF=0
*NO
X A2 1010
2ND ERROR CA23 REJECT
*YES
ANCH. TIME, TIME AT 450 K LT TNEARM
*YES
SET CELL CA78 FOR LT 450 K IND
*NO
CA3
KTIMAH 0, 15 INTO KTIMAH
X A0
CALC. ORB. ELEM. OF REENTRY R,V D4CFRV
CA45
A6
TDT(NT=0 (TNETIM END TIME OK FOR TNINTZ)
*NO
*YES
GET LAST VALID TIME FOR INTERP. INTO TDNIET
CA86
C6
KWTIND NOT =0 (T. WT. PROG USED FOR OBS)
*NO
DCA86+2
CARDC
USE PROG D3NOF5 (SCHEME EDF2/M)
X A0 1010
*YES
CA2
KSRIND =- (2ND OR LATER DC)
CA44
*YES
X B0
*NO
CA78
TEST IND. FOR LT 450 K
CA4
*NO
X A0 1006
*YES
CA3
WT. OBS. FROM STA. IN DC USING TIME WT D3PTWT
X A0
MFDIFC
SET UP NI INPUT FOR NDC R2AINT
X A3 1010


CARDC
LOG. ERROR RET. IN CALOG ROOM FOR 5 ERRORS
A0
CARC1
0 INTO TDPRNT UND. NO. TO PRINT ON-LINE
A1
CA23
1 INTO TDTIMT AC, 0 INTO CACNT IND FOR RES.
A2
MFDIFC
EXIT MFDIFC
A3
CA47
TNOMNI GT/= TO 6 (ALL 6 NI TABLES SET UP FOR N.D.C)
YES
NO
B0
MFDIFC
SET UP NI INPUT FOR NDC R2AINT
A3
CA48
TNDCIN=0 (FIRST NDC STA)
NO
YES
CA76
SET CONTROLS TO FORCE AND INITIALIZE FIRST NDC
CA4
SET CONTROLS FOR NDC
A0
1006

49. MFDIFC - MONITOR SUFFIX TO DODIFC

MFDIFC MAY BE ENTERED FOR ONE OF THE FOLLOWING REASONS - 1) AN ACCEPTABLE DIFFERENTIAL CORRECTION HAS BEEN MADE AND DODIFC HAS REQUESTED CONTROL FOR ANOTHER CORRECTION USING THE SAME RADAR DATA, 2) AN ACCEPTABLE DIFFERENTIAL CORRECTION HAS BEEN MADE, BUT DODIFC DOES NOT WISH TO CORRECT AGAIN USING THE SAME DATA, 3) THE DIFFERENTIAL CORRECTION HAS BEEN REJECTED BY DODIFC, AND 4) THE DIFFERENTIAL CORRECTION HAS BEEN REJECTED BY MPDIFC WHICH DETERMINED THAT THE NEW SET OF OBSERVATIONS IS NOT ACCEPTABLE FOR THE CURRENT DIFFERENTIAL CORRECTION. (SEE FLOWCHART)

INPUT

A) TDRANV - CONTAINS r AND v VECTORS OBTAINED FROM DIFFERENTIAL CORRECTION TOGETHER WITH NUMERICAL INTEGRATION CRITERIA.

B) TMSTMS - A TABLE OF MNLTMS ENTRIES, CONTAINING IN THE ADDRESS FIELD OF EACH ENTRY THE LOCATION OF THE TMDATA BLOCK ASSIGNED TO EACH REPORTING STATION WHICH WILL BE USED IN DIFFERENTIAL CORRECTION. ENTRIES ARE ORDERED ACCORDING TO THE TIME OF MINIMUM RANGE (TMR), WITH THE ENTRY FOR THE MOST RECENT TIME FIRST. THE LAST ENTRY IN THE TABLE IS SET TO NEGATIVE, ALL PRECEEDING ENTRIES ARE POSITIVE, WHILE ALL REMAINING LOCATIONS ARE NEGATIVE ZERO.

C) TMDATA - CONTAINS SUCCESSIVE 3-WORD BLOCKS SUCH THAT EACH BLOCK IS ASSOCIATED WITH A REPORTING STATION. THE FIRST WORD OF EACH 3-WORD BLOCK

CONTAINS THE STATION IDENTIFICATION NUMBER IN THE ADDRESS FIELD AND THE ORBIT NUMBER IN THE DECREMENT FIELD. THE SECOND WORD CONTAINS THE NUMBER OF RADAR MESSAGES RECEIVED FROM THIS STATION IN THE ADDRESS FIELD AND THE LOCATION OF THE STATION MESSAGE BLOCK IN UPPER CORE IN THE DECREMENT. THE THIRD WORD CONTAINS THE TIME OF MINIMUM RANGE WITH MINUTES IN THE ADDRESS AND SECONDS IN THE DECREMENT.

D) MCSTNO - CONTAINS THE IDENTIFICATION NUMBER OF THE STATION MOST RECENTLY PRESENTED TO DIFFERENTIAL CORRECTION.

E) MCNRRF - WHEN NONZERO, INDICATES RETROROCKETS HAVE FIRED AND THE SYSTEM IS IN THE REENTRY PHASE.

F) MCTABT - WHEN NONZERO, INDICATES THE SYSTEM IS IN THE HIGH-ABORT PHASE.

G) MCZZCC - WHEN ZERO, INDICATES THE DIFFERENTIAL CORRECTION WAS REJECTED BECAUSE OF THE ABSENCE OF r AND v ENTRIES IN THE TABLE.

H) MCDCRJ - WHEN NONZERO, INDICATES THAT THE CURRENT SET OF OBSERVATIONS WAS OMITTED FROM THE DIFFERENTIAL CORRECTION BY MPDIFC, HENCE REJECTING A DIFFERENTIAL CORRECTION.

I) MCTLST - CONTAINS THE TIME OF THE MOST RECENT OBSERVATION INCLUDED IN THE DIFFERENTIAL CORRECTION. THIS TIME IS WRITTEN ON THE RESTART TAPE.

J) MCHFSC - CONTAINS IN ITS ADDRESS FIELD THE CURRENT GMT IN FIXED-POINT HALF-SECONDS.

K) TNNDCI - CONTAINS THE STANDARD INTEGRATION FORMAT WHEN NUMERICAL DIFFERENTIAL CORRECTION IS

BEING USED.

OUTPUT

A) TMREST - DATA TO BE WRITTEN ON THE RESTART TAPE IS STORED IN THE RESTART TABLE.

B) TMLAST - THE BEST CURRENT VECTOR IS STORED IN THIS TABLE WHEN THE FINAL CORRECTION HAS BEEN ACHIEVED.

C) TDCDEF - IF A CORRECTION HAS BEEN ACCEPTED, THE NEW DEFINING VECTOR IS STORED IN THIS TABLE. NOTE THAT TDCDEF AND TMLAST WILL BE IDENTICAL AFTER THE LAST PASS THROUGH A CORRECTION CYCLE.

D) MCDCCT - SET TO NONZERO WHEN DIFFERENTIAL CORRECTION IS SUCCESSFUL. THIS CELL IS USED BY NOCPNI TO DETERMINE WHETHER THE PREVIOUS TABLE CAN BE REGENERATED IF AN ERROR OCCURS.

E) MCDCIP - SET TO ZERO WHEN THE FINAL DIFFERENTIAL CORRECTION IS ACCEPTED FOR THE CURRENT DATA TO INDICATE DIFFERENTIAL CORRECTION IS NO LONGER IN PROCESS.

F) MCDCRJ - RESET TO ZERO, IF NONZERO ON ENTRY TO MPDIFC.

G) MCFORC - SET TO ZERO UNLESS THE CAPSULE IS BELOW 450,000 FEET.

H) WHEN APPROPRIATE, ENTRIES ARE STORED IN THE QUEUE TABLES FOR NOCPNI AND MYMESS.

METHOD

MFDIFC CONTROLS THE PROGRAM FLOW BY PLACING APPROPRIATE ENTRIES IN THE INPUT QUEUE FOR NUMERICAL INTEGRATION FOR THOSE CASES IN WHICH INTEGRATION IS DESIRABLE. INTEGRATION IS QUEUED IF - 1) THE FINAL

DIFFERENTIAL CORRECTION FOR THE CURRENT SET OF RADAR OBSERVATIONS HAS BEEN ACCEPTED AND THE ORBIT PHASE IS IN PROGRESS, IN WHICH CASE AN ORBIT DISPLAY TABLE IS DESIRED, 2) THE DIFFERENTIAL CORRECTION FOR THE CURRENT SET OF OBSERVATIONS HAS BEEN ACCEPTED AND THE REENTRY PHASE IS IN PROGRESS, IN WHICH CASE A REENTRY TABLE IS DESIRED, 3) THE DIFFERENTIAL CORRECTION HAS BEEN REJECTED BY DODIFC FOLLOWING AN ACCEPTABLE DIFFERENTIAL CORRECTION AND THE ORBIT PHASE IS IN PROGRESS, IN WHICH CASE AN ORBIT DISPLAY TABLE IS DESIRED, 4) THE CAPSULE IS BELOW 450,000 FEET, IN WHICH CASE THE INTEGRATION OF TABLES FOR NUMERICAL PARTIALS IS DESIRED. ON-LINE MESSAGES ARE QUEUED TO PROVIDE AN EXTERNAL INDICATION OF THE PROGRAM FLOW.

IF THE ACCUMULATOR IS NONZERO ON ENTRY TO MFDIFC, THE CURRENT CORRECTION WAS REJECTED AND INTEGRATION IS NOT DESIRABLE UNLESS A PREVIOUS SUCCESSFUL DIFFERENTIAL CORRECTION HAS OCCURRED FOR THE CURRENT SET OF DATA. IN THIS CASE, AN ORBIT DISPLAY TABLE HAS NOT YET BEEN GENERATED FOR THE PREVIOUS PASS THROUGH DIFFERENTIAL CORRECTION TO SUCCESSFULLY PROVIDE A CORRECTED DEFINING VECTOR. IF THE ACCUMULATOR CONTAINS A POSITIVE ZERO, DIFFERENTIAL CORRECTION WAS ACCEPTED. HOWEVER, FURTHER CORRECTION WITH THE CURRENT DATA IS NOT NECESSARY. IF THE ACCUMULATOR CONTAINS A NEGATIVE ZERO, DIFFERENTIAL CORRECTION WAS ACCEPTED AND ANOTHER DIFFERENTIAL CORRECTION USING THE SAME DATA IS REQUESTED. IF THE ACCUMULATOR CONTAINS A NEGATIVE ZERO AND IF THE P BIT CONTAINS ONE, THE CAPSULE IS BELOW 450,000 FEET AND THE INTEGRATION OF TABLES FOR NUMERICAL PARTIALS IS DESIRED.

USAGE

MFDIFC IS ENTERED FROM DODIFC OR MPDIFC AND EXITS TO MOPRIO.

A) STORAGE REQUIRED - 147 LOCATIONS.

B) MFDIFC USES -

1) MACROS - QENBZ, QUEUE, TRNOF, AND TRNON

2) PARAMETERS - MNBBNI, MNCPNI, MNDIFC, MNDIFK, MNDIFM, MNGEN1, MNGEN2, MNGEN3, MNLPBD, MNMESS, MNOCAT, MNREST, AND MNWRRS

3) COMMUNICATION CELLS - MCDCRJ, MCDCCT, MCDCIP, MCFORC, MCGTIM, MCHFSC, MCNRRF, MCSTNO, MCTABT, MCTLST, AND MCZZCC

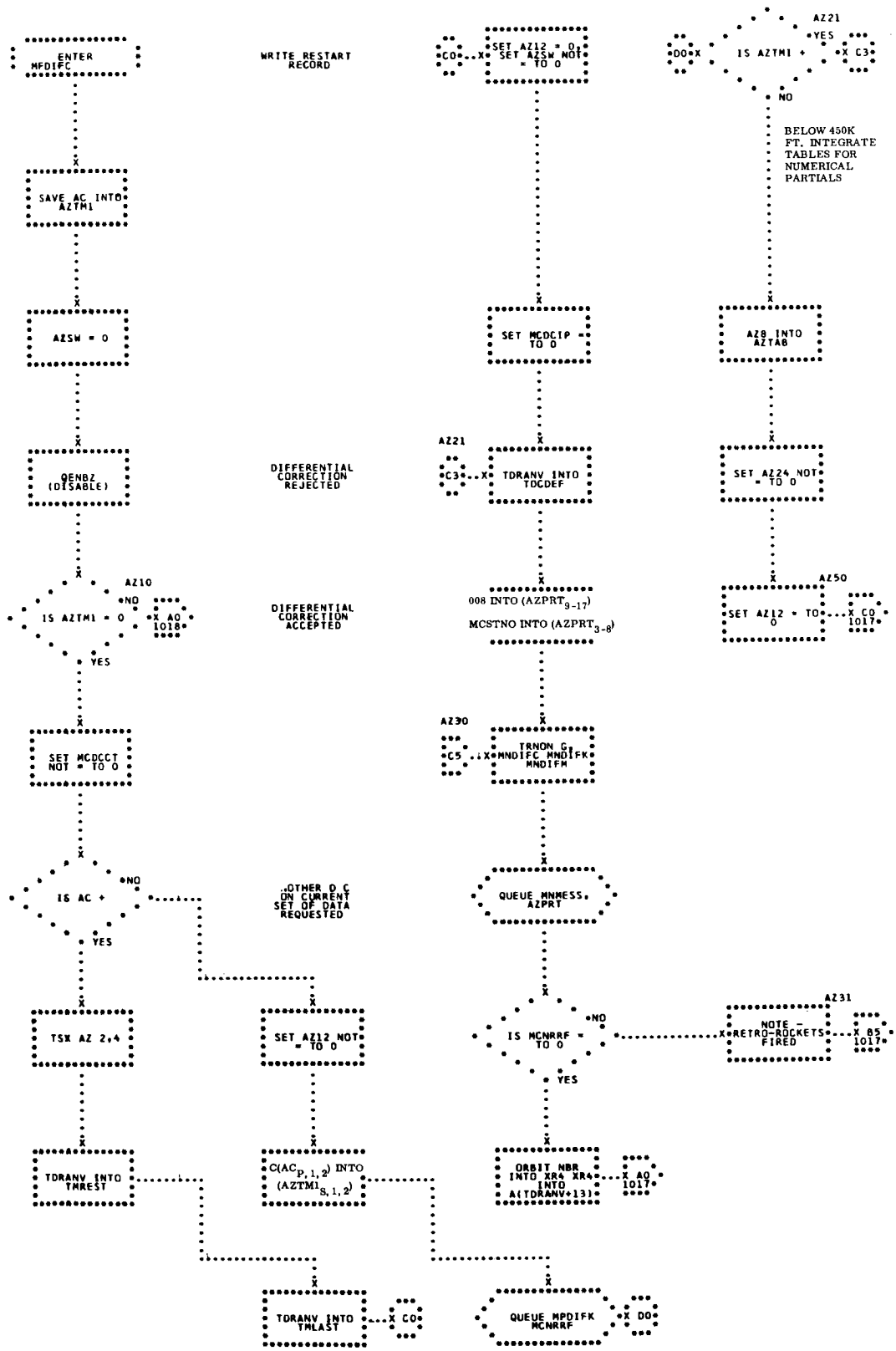
4) TABLES - TDCLLT, TDCEF, TDRANV, TMDATA, TMLAST, TMREST, TMSTMS, AND TNNDCI

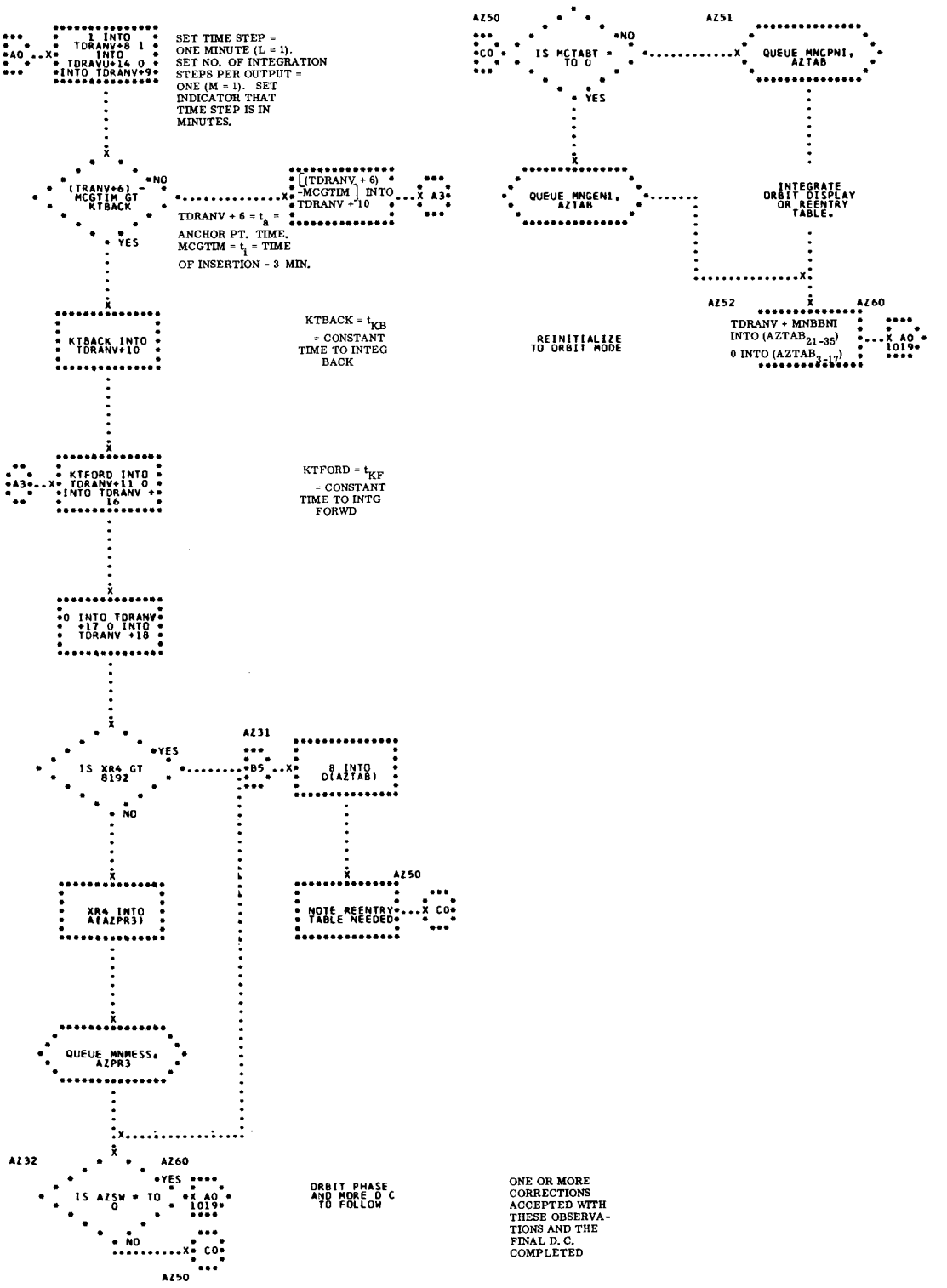
5) CONSTANTS - K00007, K00008, AND K00071

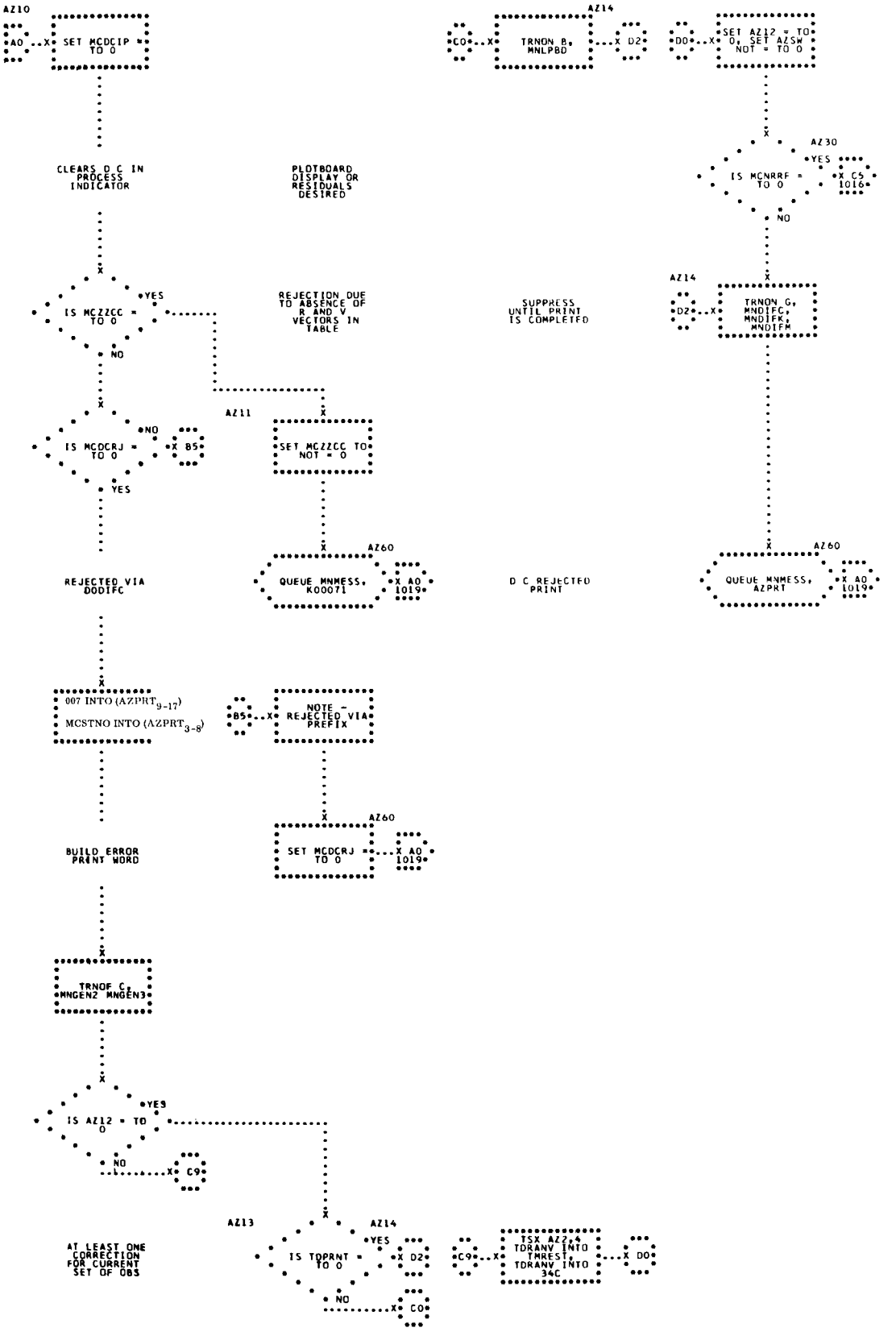
C) TIME REQUIRED -

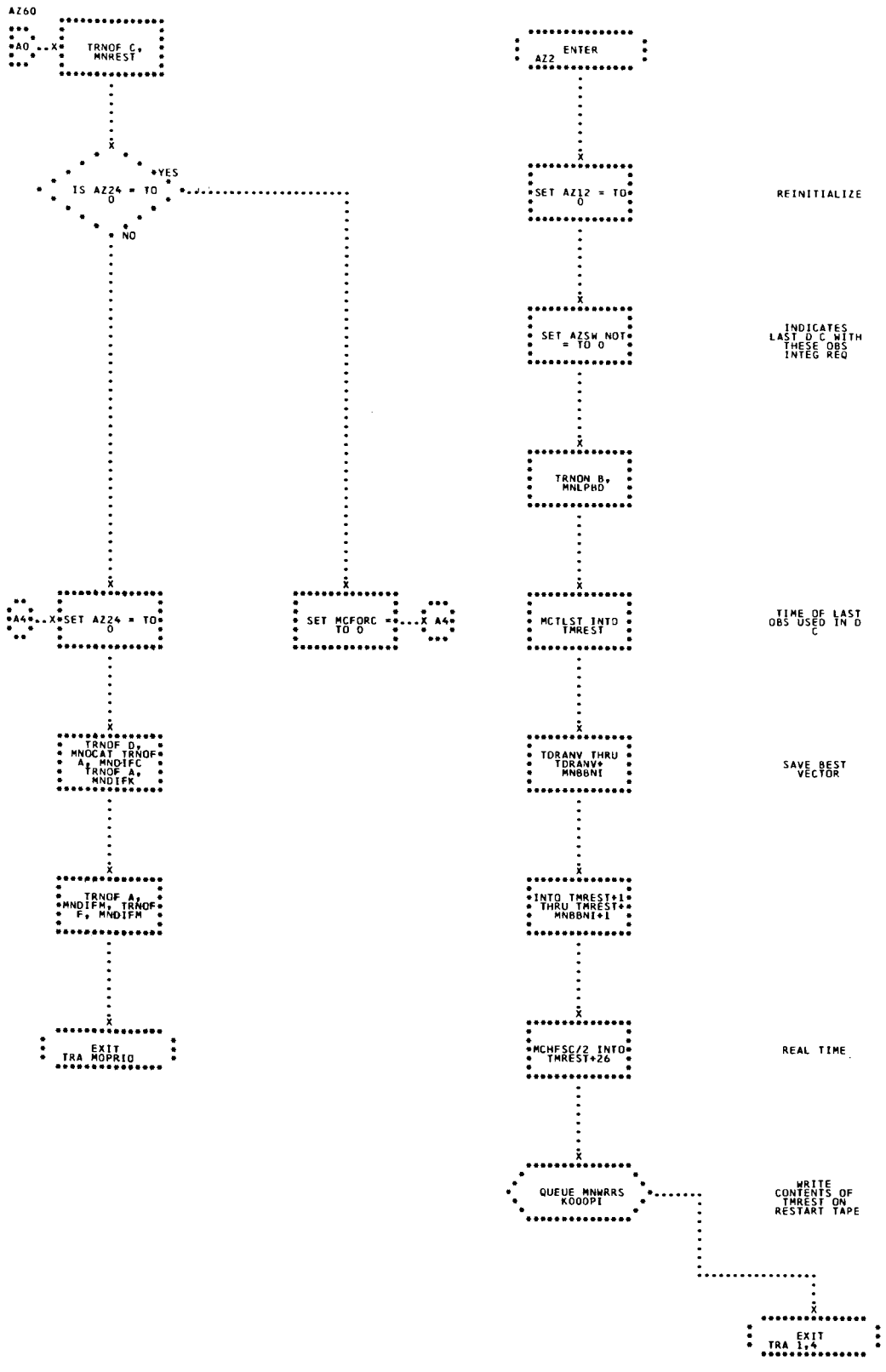
1) MINIMUM - 0.089 MILLISECOND

2) MAXIMUM - 0.436 MILLISECOND


ENTER MPDIFC
SAVE AC INTO AZTM1
AZSW = 0
QENBZ (DISABLE)
AZ10
IS AZTM1 = 0
NO
X AO 1018
YES
SET MCDCCT NOT = TO 0
IS AC +
NO
YES
TSX AZ 2,4
TDRANV INTO TMREST
WRITE RESTART RECORD
DIFFERENTIAL CORRECTION REJECTED
DIFFERENTIAL CORRECTION ACCEPTED
OTHER D C ON CURRENT SET OF DATA REQUESTED
SET AZ12 NOT = TO 0
C(AC P, 1, 2) INTO (AZTM1 S, 1, 2)
TDRANV INTO TMLAST
X CO
CO
SET AZ12 = 0, SET AZSW NOT = TO 0
SET MCDCIP = TO 0
AZ21
C3
TDRANV INTO TDCDEF
008 INTO (AZPRT 9-17)
MCSTNO INTO (AZPRT 3-8)
AZ30
C5
TRNON G, MNDIFC MNDIFK MNDIFM
QUEUE MNMESS, AZPRT
IS MCNRRF = TO 0
NO
YES
ORBIT NBR INTO XR4 XR4 INTO A(TDRANV+13)
X AO 1017
QUEUE MPDIFK MCNRRF
X DO
DO
AZ21
YES
IS AZTM1 +
X C3
NO
BELOW 450K FT. INTEGRATE TABLES FOR NUMERICAL PARTIALS
AZ8 INTO AZTAB
SET AZ24 NOT = TO 0
AZ50
SET AZ12 = TO 0
X CO 1017
AZ31
NOTE - RETRO-ROCKETS FIRED
X B5 1017


1 INTO TDRANV+8 1 INTO TDRAVU+14 0 INTO TDRANV+9
SET TIME STEP = ONE MINUTE (L = 1). SET NO. OF INTEGRATION STEPS PER OUTPUT = ONE (M = 1). SET INDICATOR THAT TIME STEP IS IN MINUTES.
(TRANV+6) - MCGTIM GT KTBACK
NO
YES
(TDRANV + 6) - MCGTIM INTO TDRANV +10
TDRANV + 6 = t_a = ANCHOR PT. TIME.
MCGTIM = t_1 = TIME OF INSERTION - 3 MIN.
KTBACK INTO TDRANV+10
KTBACK = t_{KB} = CONSTANT TIME TO INTEG BACK
KTFORD INTO TDRANV+11 0 INTO TDRANV + 16
KTFORD = t_{KF} = CONSTANT TIME TO INTG FORWD
0 INTO TDRANV +17 0 INTO TDRANV +18
IS XR4 GT 8192
YES
NO
AZ31
B5
8 INTO D(AZTAB)
XR4 INTO A(AZPR3)
AZ50
NOTE REENTRY TABLE NEEDED
CO
QUEUE MNMESS, AZPR3
AZ32
IS AZSW = TO 0
AZ60
YES
AO 1019
NO
CO
AZ50
IS MCTABT = TO 0
NO
YES
AZ51
QUEUE MNCPNI, AZTAB
QUEUE MNGENI, AZTAB
INTEGRATE ORBIT DISPLAY OR REENTRY TABLE.
REINITIALIZE TO ORBIT MODE
AZ52
AZ60
TDRANV + MNBBNI INTO ($AZTAB_{21-35}$)
0 INTO ($AZTAB_{3-17}$)
AO 1019
ORBIT PHASE AND MORE D C TO FOLLOW
ONE OR MORE CORRECTIONS ACCEPTED WITH THESE OBSERVATIONS AND THE FINAL D. C. COMPLETED


AZ10
A0
SET MCDCIP = TO 0
CLEARS D C IN PROCESS INDICATOR
PLOTBOARD DISPLAY OR RESIDUALS DESIRED
IS MCZZCC = TO 0
YES
NO
REJECTION DUE TO ABSENCE OF R AND V VECTORS IN TABLE
AZ11
SET MCZZCC TO NOT = 0
IS MCDCRJ = TO 0
NO
B5
YES
AZ60
QUEUE MNMESS, K00071
X A0 1019
REJECTED VIA DODIFC
007 INTO (AZPRT9-17)
MCSTNO INTO (AZPRT3-8)
NOTE - REJECTED VIA PREFIX
BUILD ERROR PRINT WORD
SET MCDCRJ = TO 0
TRNOF C, MNGEN2 MNGEN3
IS AZ12 = TO 0
YES
NO
C9
AT LEAST ONE CORRECTION FOR CURRENT SET OF OBS
AZ13
AZ14
IS TOPRNT = TO 0
YES
NO
D2
C0
TSX AZ2,4 TDRANV INTO TMREST, TDRANV INTO 34C
D0
AZ14
C0
TRNON B, MNLPBD
D2
D0
SET AZ12 = TO 0, SET AZSW NOT = TO 0
AZ30
IS MCNRRF = TO 0
YES
C5 1016
NO
SUPPRESS UNTIL PRINT IS COMPLETED
AZ14
D2
TRNON G, MNDIFC, MNDIFK, MNDIFM
AZ60
QUEUE MNMESS, AZPRT
X A0 1019
D C REJECTED PRINT


AZ60
A0
TRNOF C, MNREST
IS AZ24 = TO 0
YES
NO
A4
SET AZ24 = TO 0
TRNOF D, MNOCAT TRNOF A, MNDIFC TRNOF A, MNDIFK
TRNOF A, MNDIFM, TRNOF F, MNDIFM
EXIT TRA MOPRIO
SET MCFORC = TO 0
A4
ENTER AZ2
SET AZ12 = TO 0
REINITIALIZE
SET AZSW NOT = TO 0
INDICATES LAST D C WITH THESE OBS INTEG REQ
TRNON B, MNLPHD
MCTLST INTO TMREST
TIME OF LAST OBS USED IN D C
TDRANV THRU TDRANV+ MNBBNI
SAVE BEST VECTOR
INTO TMREST+1 THRU TMREST+ MNBBNI+1
MCHFSC/2 INTO TMREST+26
REAL TIME
QUEUE MNWRRS KOOOPI
WRITE CONTENTS OF TMREST ON RESTART TAPE
EXIT TRA 1,4

501 D1PCDC - PARTIAL COEFFICIENT CALCULATION

D1PCDC COMPUTES THE PARTIAL COEFFICIENTS FOR THE EQUATIONS OF CONDITION THAT ARISE FROM ONE OBSERVATION. (SEE FLOWCHART.)

INPUT

D1PCDC USES FIVE LIBRARY SUBROUTINES -

A) U1SQRT - SQUARE ROOT

B) U3DOTP - VECTOR DOT PRODUCT

C) U3MATM - MATRIX MULITPLICATION

D) U3XPRO - VECTOR CROSS PRODUCT

E) U1SICO - SINE/COSINE

THE FOLLOWING PARAMETERS SERVE AS INPUT TO THE D1PCDC PROGRAM.

KRAVC1 KRAVC2 DRAVC3	$\vec{r}_a$	(ANCHOR POINT r VECTOR)
KVAVC1 KVAVC2 KVAVC3	$\vec{v}_a$	(ANCHOR POINT v VECTOR)
KRVECX KRVECY KRVECZ	$\vec{r}_c$	(r VECTOR COMPUTED)
KMATRX	M MATRIX	(3 X 3, 9 LOCATIONS)
KASUBO	a_0	SEMIMAJOR AXIS
KKMESQ	$1 - e_0^2$	1 - ECCENTRICITY SQUARED
KNSUBO	n_0	$a_0^{-3/2}$
KRVEC3	COS i_0	i_0 = INCLINATION
KTSUBO	T_a	ANCHOR POINT TIME (FLOATING-POINT, GEMINI UNITS)

KTSUB1	t_i	TIME OBSERVED (FLOATING-POINT, GEMINI UNITS)
KRABSL	r_a	ABSOLUTE VALUE OF $\vec{r}$
AAA	A	AZIMUTH (KOODAC)
HHH	E	ELEVATION (KOOODEC)
ROW	R	RANGE (KOODRC)
JAAA	J=.00162348	CONSTANT

OUTPUT

THE FOLLOWING DATA IS OUTPUT BY THE D1PCDC PROGRAM -

TDEQCT 18 PARTIALS FOR DIFFERENTIAL CORRECTION

METHOD

THE FOLLOWING EQUATIONS (FROM THE COMPUTATION OF ORBITS BY DR. P. HERGET) ARE USED WITH THE D1PCDC PROGRAM -

$$f = \frac{\left(\vec{r}_c \times \vec{v}_a\right) \cdot \left(\vec{r}_a \times \vec{v}_a\right)}{\left(\vec{r}_a \times \vec{v}_a\right) \cdot \left(\vec{r}_a \times \vec{v}_a\right)} \tag{1}$$

$$g = \frac{r_a^2 \left(\vec{r}_c \cdot \vec{v}_a\right) - \left(\vec{r}_a \cdot \vec{v}_a\right)\left(\vec{r}_c \cdot \vec{r}_a\right)}{\left(\vec{r}_a \times \vec{v}_a\right) \cdot \left(\vec{r}_a \times \vec{v}_a\right)} \tag{2}$$

$$F = r_a (1-f) \tag{3}$$

$$G = \frac{g - F\left(\vec{r}_a \cdot \vec{v}_a\right)}{r_a} \tag{4}$$

$$T = t_i - t_0 \tag{5}$$

$$L = (3T - g - G_{ra}) \ / \ r_i \tag{6}$$

$$P_3 = (\ a/r_a \ ) \ \ (GL - 2F)$$

$$P_2 = \left[\left(F/r_i \right) G \right] \ /r_0 \tag{7}$$

$$P_1 = \left\{ \left[\left(Gr_a \right) \ /r_i \right] \ G + P_3 + F \right\} \ /r_a^{\ 3}$$

$$\vec{P} = (P_1, \ P_2, \ P_3)$$

$$q_3 = a(3g - 3T + FL) \tag{8}$$

$$q_2 = F^2/r_i$$

$$q_1 = q_3/r_0^{\ 3} + P_2$$

$$\vec{q} = (q_1, \ q_2, \ q_3)$$

$$df = \vec{P} \ Y, \ dg = \vec{q} \ \ Y \tag{9}$$

$$\text{where } Y = \begin{pmatrix} \vec{r}_0 & O_0 \\ \vec{v}_0 & \vec{r}_0 \\ O_0 & \vec{v}_0 \end{pmatrix} \ ; \text{ a (3 x 6 matrix)} \quad O = (0, \ 0, \ 0)$$

$$dr = \begin{pmatrix} f\ 0\ 0\ g\ 0\ 0 \\ 0\ f\ 0\ 0\ g\ 0 \\ 0\ 0\ f\ 0\ 0\ g \end{pmatrix} + \begin{pmatrix} \vec{r}_a \\ \vec{v}_0 \end{pmatrix}^T + \begin{pmatrix} df \\ dg \end{pmatrix} \tag{10}$$

$$RAD = \begin{pmatrix} \text{Cos E Sin A}, & \text{Cos E Cos A}, & \text{Sin E} \\ \dfrac{\text{Cos A}}{R} & \dfrac{-\text{ Sin A}}{R} & 0 \\ \dfrac{-\text{ Sin E Sin A}}{R} & \dfrac{-\text{ Sin E Cos A}}{R} & \dfrac{\text{Cos E}}{R} \end{pmatrix} \tag{11}$$

$$\xi = J \cdot n \left\{ \left(\cos i_0\right) / \left[a\left(1-e^2\right)\right]^{2} \right\} T \qquad (12)$$

$$B = \begin{pmatrix} \cos\xi & \sin\xi & 0 \\ -\sin\xi & \cos\xi & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad (13)$$

$$A = \left[(\mathrm{RAD})\, M \right] \quad B \qquad (14)$$

Calculate A (dr) (15)

USAGE

A) CALLING SEQUENCE -

```
TSX     DIPCDC,4
        ERROR IN UISICO
        NORMAL RETURN
```

B) STORAGE REQUIRED -

292 LOCATIONS
+ 100 TEMPORARY LOCATIONS
+ 22 K LOCATIONS
+ 18 LOCATION TABLE
+ SUBROUTINE

C) CONSTANTS -

KTIMTO PZE ($T_i - T_a$) IN FLOATING-POINT, GEMINI UNITS.

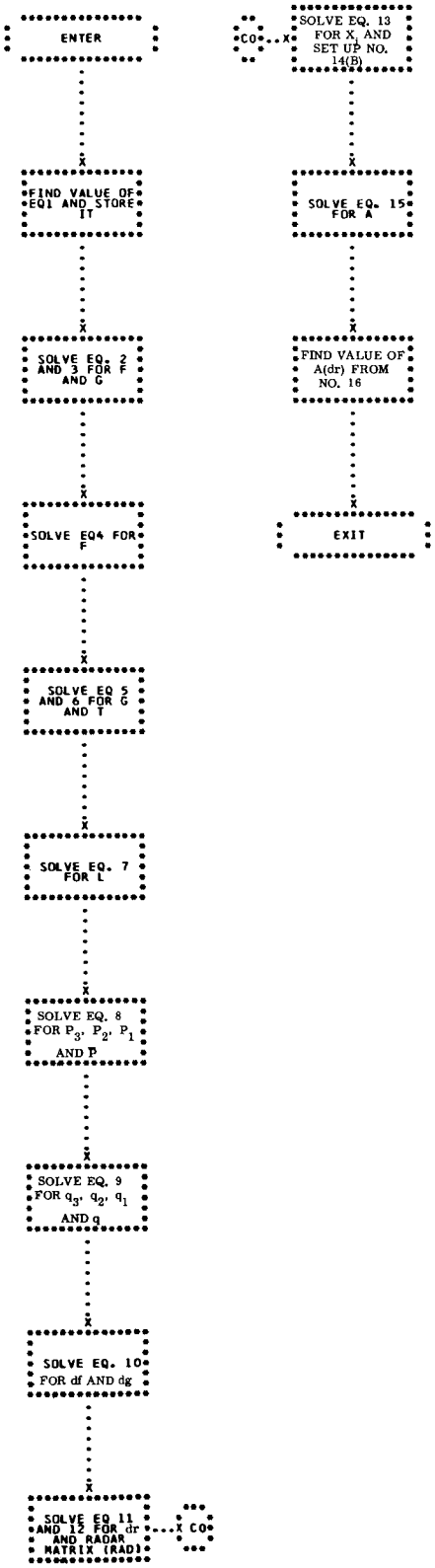
```
K001.0  DEC  1.0
K002.0  DEC  2.0
K003.0  DEC  3.0
```

PLUS THE CONSTANTS USED IN INPUT AND OUTPUT.

D) TIMING - 141.3 MILLISECONDS

E) CHECKOUT - CHECKS ON THIS PROGRAM WERE MADE TO

EIGHT DECIMAL PLACES. ACCURACY WAS EXACT EXCEPT IN A FEW INSTANCES WHERE THE VALUE WAS OFF IN THE EIGHTH PLACE, PROBABLY DUE TO ROUNDING.


ENTER
FIND VALUE OF EQ1 AND STORE IT
SOLVE EQ. 2 AND 3 FOR F AND G
SOLVE EQ4 FOR F
SOLVE EQ 5 AND 6 FOR G AND T
SOLVE EQ. 7 FOR L
SOLVE EQ. 8 FOR P_3, P_2, P_1 AND P
SOLVE EQ. 9 FOR q_3, q_2, q_1 AND q
SOLVE EQ. 10 FOR df AND dg
SOLVE EQ 11 AND 12 FOR dr AND RADAR MATRIX (RAD)
CO
X
SOLVE EQ. 13 FOR X, AND SET UP NO. 14(B)
SOLVE EQ. 15 FOR A
FIND VALUE OF A(dr) FROM NO. 16
EXIT

51. D2CSFE - CALCULATE STANDARD ERROR FIT

D2CSFE COMPUTES THE STANDARD ERROR OF FIT AND ELEMENTS. (SEE FLOWCHART.)

INPUT

D2CSFE USES THE SQUARE ROOT LIBRARY SUBROUTINE U1SQRT.

SYMBOLIC LOCATIONS — DEFINITIONS

A) KOOOON — NUMBER OF COEFFICIENTS (n)

B) KOSUMM — NUMBER OF EQUATIONS OF CONDITION

C) KDFSQU

$$\sum_{i=1}^{m} (dF_i)^2$$

D) TEMD2 — BLOCK OF TEMPORARY STORAGE GENERATED BY D2SLNE AND D2CSFE CONTAINING $\sum_{i=1}^{m} (dF)_i$ aij AND B_{ij}^{-1} (DIAGONAL ELEMENTS OF THE INVERSE MATRIX).

E) TLSSSE, TLSSSE + (N-1) - (n) COEFFICIENTS.

OUTPUT

TLSSSE, TLSSSE + 2N - (n) COEFFICIENTS IMMEDIATELY FOLLOWED BY STANDARD ERROR OF FIT AND STANDARD ERROR OF ELEMENTS.

METHOD

$$S_e = \sqrt{\frac{1}{(m-n)} \sum_{i=1}^{m} (dF_i)^2 - \sum_{j=1}^{n} X_j \left[\sum_{i=1}^{m} (dF)_i \, a_{ij} \right]}$$

SOLVING THE ABOVE EQUATION FOR THE FIT, $S_{(x)j}$ IS OBTAINED BY -

$$S_{x_j} = S_e \sqrt{B_{jj}^{-1}}$$

WHERE B_{jj}^{-1} IS THE TERM OF THE PRINCIPAL DIAGONAL OF THE INVERSE OF THE MATRIX REPRESENTING THE NORMAL EQUATIONS.

USAGE

A) CALLING SEQUENCE -

```
TSX     D2CSFE,4
        ERROR RETURN (ERROR IN U1SQRT)
        NORMAL
```

B) SPACE REQUIRED - 54 LOCATIONS

C) ERROR CODES - ALPHA + 1 ERROR IN U1SQRT.

D) SPECIAL USAGE - D2SUNE AND D2SLNE MUST BE EXECUTED PRIOR TO D2CSFE TO COMPUTE STANDARD ERROR.

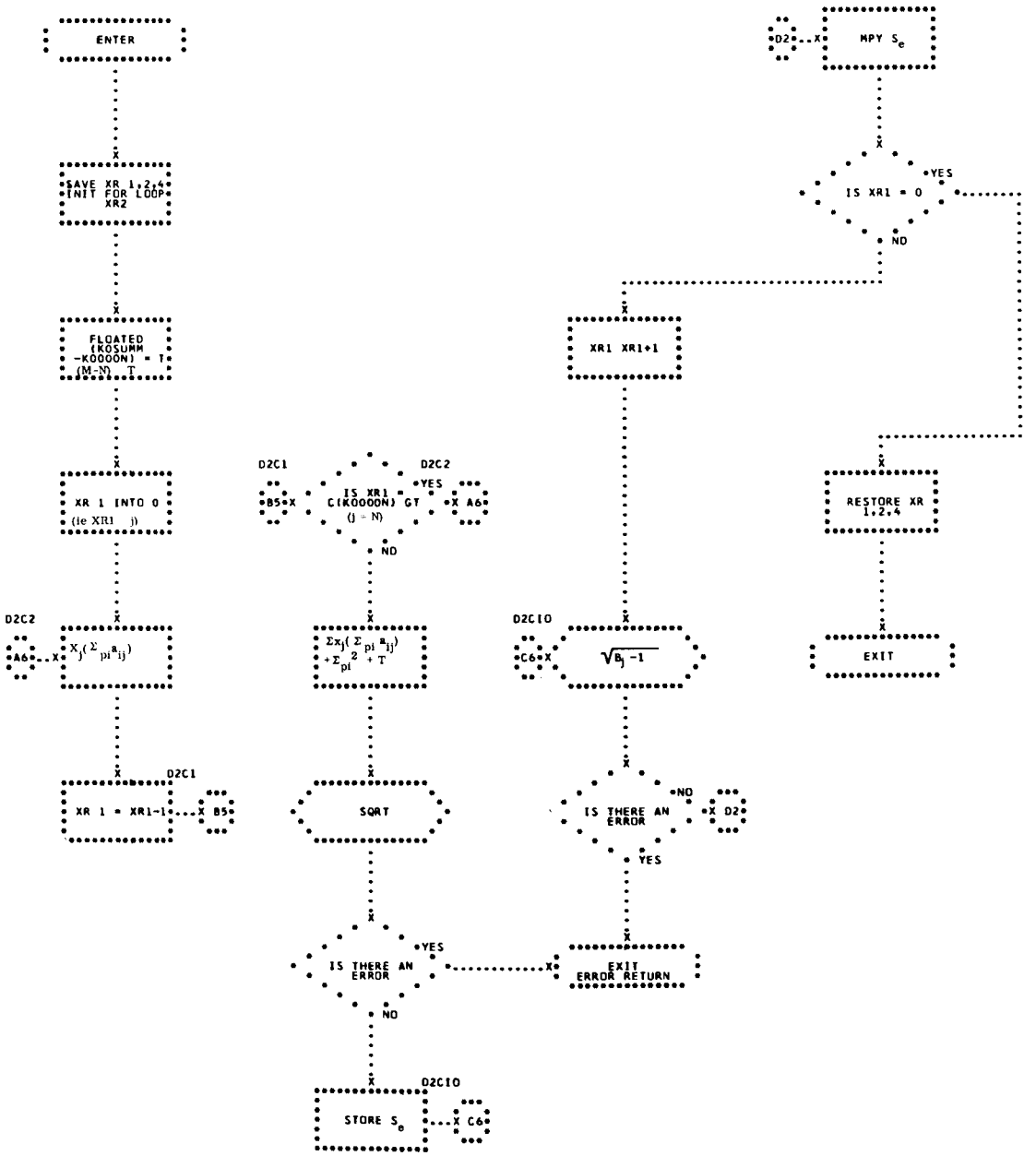
E) CONSTANTS USED -

```
K0ZERO  DEC 0
K00001  DEC 1
KCH233  OCT 233000000000
```

F) ERASABLE LOCATIONS - TEMD2, TEMD2 $\pm$ 20 N(2N+1) + 2 IS INPUT, ERASABLE AFTER COMPLETION OF D2CSFE.

G) TIMING - 14.928 MSEC FOR SIX UNKNOWNS.

H) CHECKOUT - C2CSFE WAS CHECKED OUT USING OUTPUT FROM D2SUNE AND D2SLNE AND HAND COMPUTING THE S_e AND S_{ej} .


ENTER
SAVE XR 1,2,4 INIT FOR LOOP XR2
FLOATED (KOSUMM -KOOOON) = T (M-N) T
XR 1 INTO 0 (ie XR1 j)
D2C2
A6
$X_j(\Sigma_{pi} a_{ij})$
D2C1
XR 1 = XR1-1
B5
D2C1
B5
IS XR1 = C(KOOOON) GT (j = N)
YES
D2C2
A6
NO
$\Sigma x_j(\Sigma_{pi} a_{ij}) + \Sigma_{pi}{}^2 + T$
SQRT
IS THERE AN ERROR
YES
EXIT ERROR RETURN
NO
D2C10
C6
STORE S_e
D2C10
C6
$\sqrt{B_j - 1}$
XR1 XR1+1
D2
MPY S_e
IS XR1 = 0
YES
NO
IS THERE AN ERROR
NO
D2
RESTORE XR 1,2,4
EXIT

52. D2SLNE - SOLVE NORMAL EQUATIONS

D2SLNE SOLVES M NORMAL EQUATIONS AND OBTAINS DIAGONALS OF THE INVERSE MATRIX. D2SLNE MUST BE USED AFTER GONALS OF THE INVERSE MATRIX. D2SLNE MUST BE USED AFTER D2SUNE WHICH SETS UP THE NORMAL EQUATIONS. D2CSFE THEN COMPUTES THE STANDARD ERROR OF FIT AND ELEMENTS. (SEE FLOWCHARTS.)

INPUT

THE FOLLOWING PARAMETERS SERVE AS INPUT TO THE D2SLNE PROGRAM -

VALUES	SYMBOLIC LOCATION
N = NUMBER OF UNKNOWNS	D(TEMD2 + 1)
M = NUMBER OF EQUATIONS	A(TEMD2 + 1)

NORMAL EQUATIONS MATRIX IS STORED BY ROWS BEGINNING IN TEMD2 + 20.

THE SIZE OF TEMD2 IS DEPENDENT UPON THE SIZE OF N, WHICH CANNOT BE GREATER THAN 10. (REFER TO WRITE-UP OF D2SUNE FOR FORMAT OF TEMD2).

OUTPUT

THE FOLLOWING ITEMS ARE OUTPUT FROM THE D2SLNE PROGRAM -

A) $\Delta X_1, \Delta X_2 \ldots, \Delta X_n$ - STORED CONSECUTIVELY BEGINNING IN TLSSSE.

B) B_{ij} - TERMS ON THE PRINCIPAL DIAGONAL OF THE INVERSE OF THE MATRIX (B) (TO BE USED LATER IN D2CSFE).

METHOD

THE NORMAL EQUATIONS ARE SOLVED BY MEANS OF ELIMINATION. THE DIAGONALS OF THE INVERSE MATRIX ARE THEN COMPUTED. THIS PROGRAM MUST BE RUN WITH D2SUNE AND D2CSFE.

USAGE

A) CALLING SEQUENCE -

```
TSX D2SLNE,4
PZE TLSSE
    ERROR RETURN (RESULT OF ZERO IN PIVOT
    ELEMENT)
    NORMAL RETURN
```

B) STORAGE REQUIRED -

135 LOCATIONS (MAIN PROGRAM)

156 LOCATIONS (MAIN PROGRAM, K LOC, TLSSE)

TOTAL = 156 + TEMD2 + 20 + N(2N + 1) + 1 (NONERASABLE) WHERE N IS THE NUMBER OF EQUATIONS.

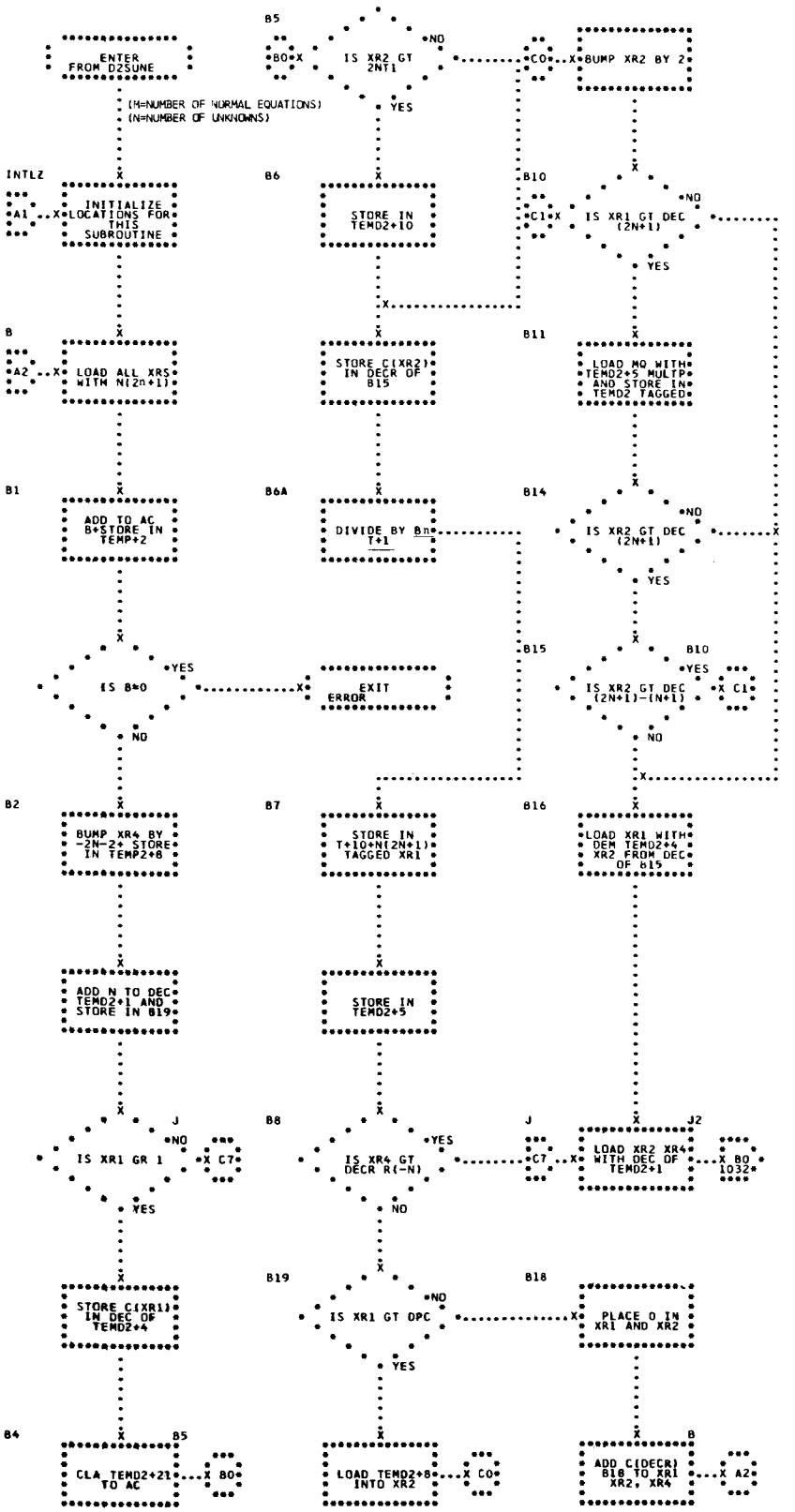
C) ERROR CODE - AN ERROR WILL RESULT IF THERE IS A ZERO IN THE PIVOT ELEMENT.

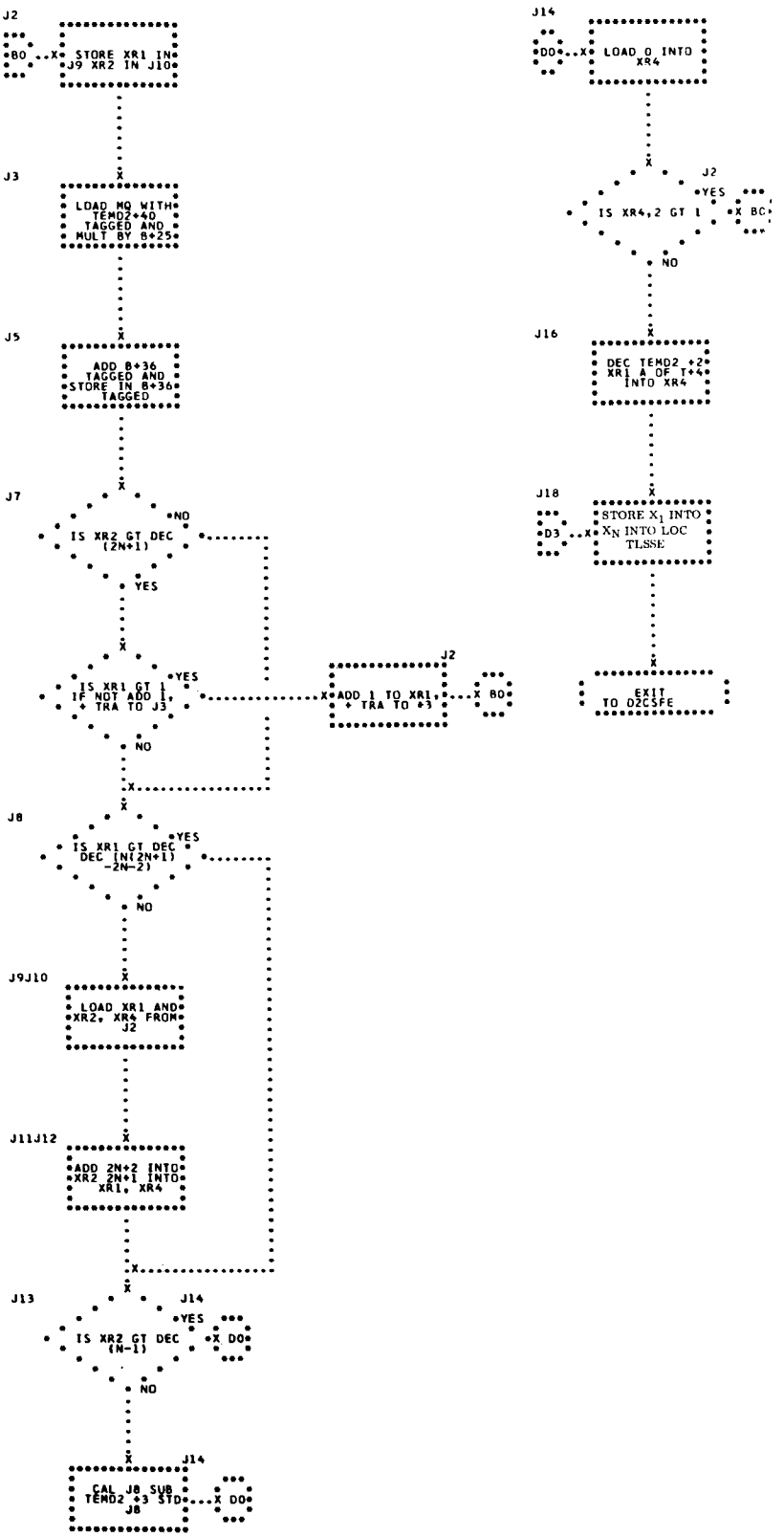
D) CONSTANTS USED -

```
KOZERO  DEC 0,
K00001  DEC 1
KD0001  PZE 0,0,1
KD0002  PZE 0,0,2
```

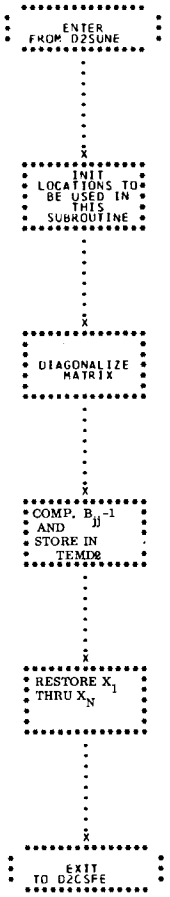
E) COMMON LOCATIONS - [TEMD2 + 20 + N(2N + 1) + 1] NONERASABLE STORAGE.

F) CHECKOUT - D2SLNE WAS CHECKED OUT WITH D2SUNE BY COMPARING RESULTS WITH THOSE OF LSQ2 (SEE SHARE LIBRARY).


ENTER FROM D2SUNE
(M=NUMBER OF NORMAL EQUATIONS)
(N=NUMBER OF UNKNOWNS)
INTLZ
INITIALIZE LOCATIONS FOR THIS SUBROUTINE
A1
B
LOAD ALL XRS WITH N(2n+1)
A2
B1
ADD TO AC B+$TORE IN TEMP+2
IS B*0
YES
NO
EXIT ERROR
B2
BUMP XR4 BY -2N-2+ STORE IN TEMP2+8
ADD N TO DEC TEMD2+1 AND STORE IN B19
IS XR1 GR 1
J
NO
YES
X C7
B4
B5
CLA TEMD2+21 TO AC
X B0
B5
B0
IS XR2 GT 2NT1
NO
YES
B6
STORE IN TEMD2+10
STORE C(XR2) IN DECR OF B15
B6A
DIVIDE BY Bn+T+1
B7
STORE IN T+10+N(2N+1) TAGGED XR1
STORE IN TEMD2+5
B8
IS XR4 GT DECR R(-N)
YES
NO
B19
IS XR1 GT DPC
NO
YES
LOAD TEMD2+8 INTO XR2
X C0
C0
BUMP XR2 BY 2
B10
C1
IS XR1 GT DEC (2N+1)
NO
YES
B11
LOAD MQ WITH TEMD2+5 MULTP AND STORE IN TEMD2 TAGGED
B14
IS XR2 GT DEC (2N+1)
NO
YES
B15
B10
IS XR2 GT DEC (2N+1)-(N+1)
YES
X C1
NO
B16
LOAD XR1 WITH DEM TEMD2+4 XR2 FROM DEC OF B15
J
J2
C7
LOAD XR2 XR4 WITH DEC OF TEMD2+1
X B0 1032
B18
PLACE 0 IN XR1 AND XR2
B
ADD C(DECR) B18 TO XR1 XR2, XR4
X A2


J2
B0
STORE XR1 IN J9 XR2 IN J10
J3
LOAD MQ WITH TEMD2+40 TAGGED AND MULT BY 8+25
J5
ADD B+36 TAGGED AND STORE IN B+36 TAGGED
J7
IS XR2 GT DEC (2N+1)
NO
YES
IS XR1 GT 1 IF NOT ADD 1, + TRA TO J3
YES
NO
J2
ADD 1 TO XR1, + TRA TO +3
B0
J8
IS XR1 GT DEC DEC (N(2N+1) -2N-2)
YES
NO
J9J10
LOAD XR1 AND XR2, XR4 FROM J2
J11J12
ADD 2N+2 INTO XR2 2N+1 INTO XR1, XR4
J13
IS XR2 GT DEC (N-1)
J14
YES
NO
CAL J8 SUB TEMD2 +3 STD J8
J14
D0
J14
LOAD 0 INTO XR4
J2
IS XR4,2 GT 1
YES
B0
NO
J16
DEC TEMD2 +2 XR1 A OF T+4 INTO XR4
J18
D3
STORE X_1 INTO X_N INTO LOC TLSSE
EXIT TO DZCSFE

GENERAL FLOW CHART



THESE ARE THE DIAGONALS OF THE INVERSE OF THE MATRIX

53. D2SSMT - STATION SUM OF SQUARES OVER M

D2SSMT COMPUTES $\frac{\Sigma(dF)^2}{\text{NO. OF EQUATION}}$ FOR EACH STATION IN A DIFFERENTIAL CORRECTION INTERVAL AND SETS THE RESULTS IN TABULAR FORM. (SEE FLOWCHART.)

INPUT

THE FOLLOWING DATA SERVES AS INPUT TO THE D2SSMT PROGRAM -

SYMBOLIC LOCATION	DEFINITION
KTOTLM	NUMBER OF EQUATIONS OF CONDITION (IN FIXED-POINT).
KTDFSQ	$\Sigma(dF)^2$, SUM OF THE SQUARE OF THE RIGHTHAND SIDES TO THE EQUATIONS OF CONDITION (IN FLOATING-POINT).
KD2CNT	ZERO FOR EACH SERIES OF STATIONS PROCESSED. NONZERO AFTER INITIALIZATION FOR THE SERIES HAS TAKEN PLACE.

OUTPUT

TD2SSM	TABLE CONTAINING $\Sigma(dF)^2$ FOR EACH REPORTING STATION IN THE ORDER THAT THE STATIONS WERE PROCESSED FOR THE OVERALL DIFFERENTIAL CORRECTION (IN FLOATING-POINT).

METHOD

ΣdF^2 AND ΣM FOR INDIVIDUAL STATIONS ARE OBTAINED

AS FOLLOWS -

$$(\Sigma dF^2)_S - (\Sigma dF^2)_{S-1} = (\Sigma dF^2)_{1\ STATION}*$$

THIS PROGRAM IS ENTERED AND $\frac{\Sigma(dF)^2}{m}$ IS COMPUTED AND STORED IN ITS PLACE IN TD2SSM AFTER EACH STATION IN THE INTERVAL HAS BEEN PROCESSED.

USAGE

A) CALLING SEQUENCE -

```
TSX             D2SSMT,4
                NORMAL RETURN
```

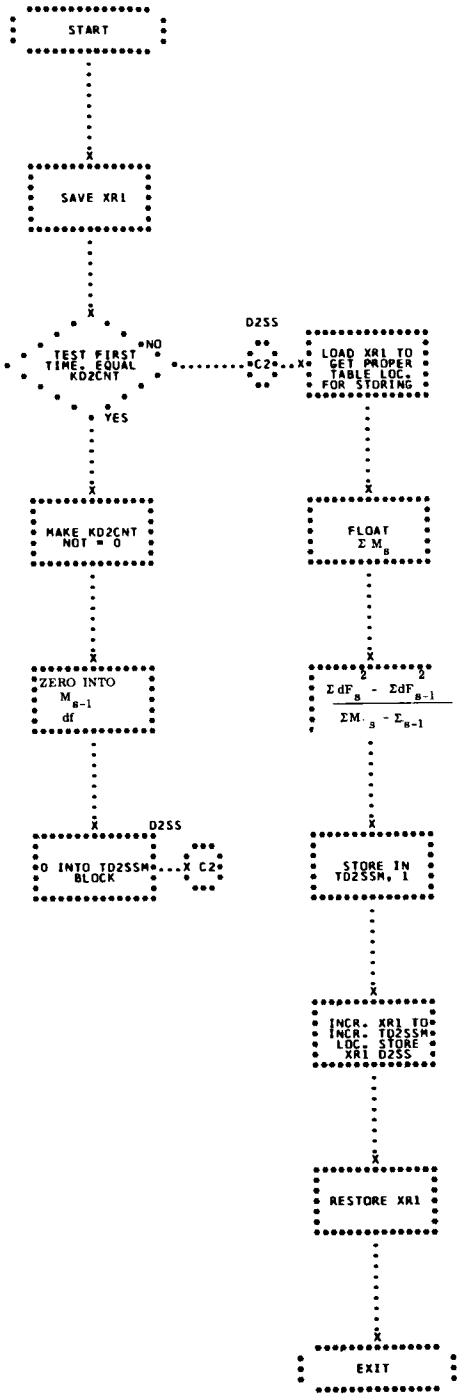
B) SPACE REQUIRED - 31 LOCATIONS.

C) CONSTANTS USED -

```
KOZERO = 0
K00001 = 1
KCH233 = 2330000000000 OCTAL
TD2SSM   BSS    17
```

D) TIMING - FIRST STATION REQUIRES 1.02 MILLISECONDS AND .78 MILLISECONDS EACH TIME THEREAFTER.

E) CHECKOUT - THE PROGRAM WAS ENTERED FIVE TIMES, EACH TIME STORING $\frac{dF^2}{m}$ IN ITS APPROPRIATE LOCATION. RESULTS SHOWED CORRECT ANSWERS.


START
SAVE XR1
TEST FIRST TIME, EQUAL KD2CNT
NO
YES
D2SS
C2
LOAD XR1 TO GET PROPER TABLE LOC. FOR STORING
MAKE KD2CNT NOT = 0
FLOAT Σ M_s
ZERO INTO M_s-1 df
Σ dF_s^2 - ΣdF_s-1^2 / ΣM_s - Σ_s-1
D2SS
0 INTO TD2SSN BLOCK
C2
STORE IN TD2SSN, I
INCR. XR1 TO INCR. TD2SSN LOC. STORE XR1 D2SS
RESTORE XR1
EXIT

54. D2SUNE - SET UP NORMAL EQUATIONS

D2SUNE OBTAINS NORMAL EQUATIONS FROM A GROUP OF M EQUATIONS IN N UNKNOWNS. (SEE FLOWCHART)

INPUT

SYMBOLIC LOCATIONS	DEFINITIONS
TAWTPC	= CORE LOCATION OF INDEPENDENT VARIABLE MATRIX.
TPWTDF	= CORE LOCATION OF FIRST OF M VALUES OF THE DEPENDENT VARIABLE.
C(KOOMCT)	= NUMBER OF EQUATIONS.
C(KCOOON)	= NUMBER OF UNKNOWN COEFFICIENTS

THE ABOVE INPUTS ARE SET IN THE CALLING SEQUENCE (DODIFC).

KD2SUT — INDICATOR SET BY CONTROL PROGRAM TO AVOID REPEATED INITIALIZATION OF CELLS DEPENDENT UPON N (NUMBER OF UNKNOWNS).
NONZERO - INITIALIZATION
ZERO - SKIP INITIALIZATION

OUTPUT

A) KOSUMM - TOTAL NUMBER OF EQUATIONS ADDED TO NORMAL EQUATIONS BLOCK THUS FAR.

B) NORMAL EQUATIONS BLOCK SET UP TO BE SOLVED BY D2SLNE STARTING AT LOCATION TEMD2 + 20.

TEMD2+20	TEMD2+20+1	TEMD2+20+(n-1)	TEMD2+20+n
$\Sigma(a_{i1})^2$	$\Sigma a_{i1}\, a_{i2}$	$\ldots \Sigma a_{i1}\, a_{in}$	$\Sigma a_{i1}\, p_i$
	$\Sigma(a_{i2})^2$	$\ldots \Sigma(a_{i2}\, a_{in})$	$\Sigma a_{i2}\, p_i$
		TEMD2+20+n(2n+1)-2	TEMD2+20+n(2n+1)-1
		$\Sigma(a_{in})^2$	$\Sigma a_{in}\, p_i$

The normal equations are stored by rows beginning with the first row in TEMD2 + 20. Each of the following rows is stored in n locations following a floating-point one in locations TEMD2 + 20 + n + 1, TEMD2 + 20 + n + 1+ 2n+ 2.. TEMD2 + 20 + 20 + n(2n + 1)

Method

Using the fundamental equations

$$a_{11} x_1 + a_{12} x_2 + \ldots + a_{1n} x_n = p_1$$

$$a_{21} x_1 + a_{22} x_2 + \ldots + a_{2n} x_n = p_2$$

$$\vdots$$

$$a_{m1} x_1 + a_{m2} x_2 + \ldots + a_{mn} x_n = p_m$$

where a = independent variables (partial coefficients)

p = dependent variables

$m > n$.

WHERE a = INDEPENDENT VARIABLES (PARTIAL COEFFICIENTS)
p = DEPENCENT VARIABLES
m GREATER THAN n.

TEMD2	Σm	
+1	Decrement	n
	Address	m
+2	Decrement	n + 1
	Address	m + 1
+3	Decrement	2n + 1
	Address	n(2n + 1) + 1
+4		n
+5		
+6	Address	TEMD2+20 + n(2n + 1)+1
+7	Decrement	m(n + 1)
	Address	p
+8		
+9		
+10	$\Sigma a_{i1} p_i$	
+11	$\Sigma a_{i2} p_i$	
+12	$\Sigma a_{i3} p_i$	
.	.	
.	.	
.	.	
+19*	$\Sigma a_{in} p_i$	
+20**	$\Sigma (a_{i1})^2$	
+21	$\Sigma a_{i1} a_{i2}$	
+22	$\Sigma a_{i1} a_{i3}$	
+23	$\Sigma a_{i1} a_{i4}$	
+24	$\Sigma a_{i1} p_i$	
+25	1.0	
+26		
+27		
+28		
+29		
+30	$\Sigma (a_{i2})^2$	
+31	$\Sigma a_{i2} a_{i3}$	

TEMD2	
+32	$\Sigma a_{i2}\, a_{i4}$
+33	$\Sigma a_{i2}\, p_i$
+34	
+35	1.0
+36	
+37	
+38	
+39	
+40	$\Sigma (a_{i3})^2$
+41	$\Sigma a_{i3}\, a_{i4}$
+42	$\Sigma a_{i3}\, p_i$
+43	
+44	
+45	1.0
+46	
+47	
+48	
+49	
+50	$\Sigma (a_{i4})^2$
+51	$\Sigma a_{i4}\, p_i$
+52	
+53	
+54	
+55	1.0
+56	Σp_i^2

*NOTE - MAXIMUM n = 10, CAN BE CHANGED BY MAKING T + 20 = T + 10 + n.

**FOR CLARITY IN ILLUSTRATION OF STORAGE ASSUME N =4.

SINCE THE ABOVE MATRIX IS SYMMETRIC, ONLY THE DIAGONALS AND INFORMATION GIVEN ABOVE NEED BE STORED IN THE NORMAL EQUATIONS BLOCK.

SETTING THE EQUATIONS IN MATRIX FORM -

$$\begin{pmatrix} a_{i1} & a_{i2} \cdots & a_{in} \\ a_{21} & a_{22} \cdots & a_{2n} \\ \cdot & & \\ \cdot & & \\ \cdot & & \\ a_{m1} & a_{m2} \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} X_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{m2} \end{pmatrix} = \begin{pmatrix} p_i \\ p_2 \\ \cdot \\ \cdot \\ \cdot \\ p_m \end{pmatrix}$$

OR

$$[A] \quad \{X\} \quad = \quad \{P\}.$$

MULTIPLYING BOTH SIDES OF THE ABOVE EQUATION BY THE TRANSPOSE, $[A^T]$ OF $[A]$, THE FOLLOWING EQUATION IS OBTAINED -

$$[A^T][A] \quad \{X\} \quad = \quad [A^T]\{P\}$$

OR

$$[B] \quad \{X\} \quad = \quad [C]$$

WRITTEN AS -

$$\begin{pmatrix} \sum_{i=1}^{m} (a_{i1})^2 & \sum_{i=1}^{m} a_{11}a_{12} & \cdots & \sum_{i=1}^{m} a_{i1}a_{in} \\ \sum_{i=1}^{m} a_{12}a_{11} & \sum_{i=1}^{m} (a_{12})^2 & \cdots & \sum_{i=1}^{m} a_{i2}a_{in} \\ \vdots & \vdots & & \vdots \\ \sum_{i=1}^{m} a_{in}a_{i1} & \sum_{i=1}^{m} a_{in}a_{i2} & \cdots & \sum_{i=1}^{m} (a_{in})^2 \end{pmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{matrix} \sum_{i=1}^{m} a_{i1}p_i \\ \sum_{i=1}^{m} a_{12}p_i \\ \vdots \\ \sum_{i=1}^{m} a_{in}p_i. \end{matrix}$$

USAGE

A) CALLING SEQUENCE -

```
TSX       D2SUNE,4
          TAWTPC,0, TPWTDT
          C(KOOMCT), 0, C(KCCOON)
          NORMAL RETURN
```

B) SPACE REQUIRED - 126 LOCATIONS.

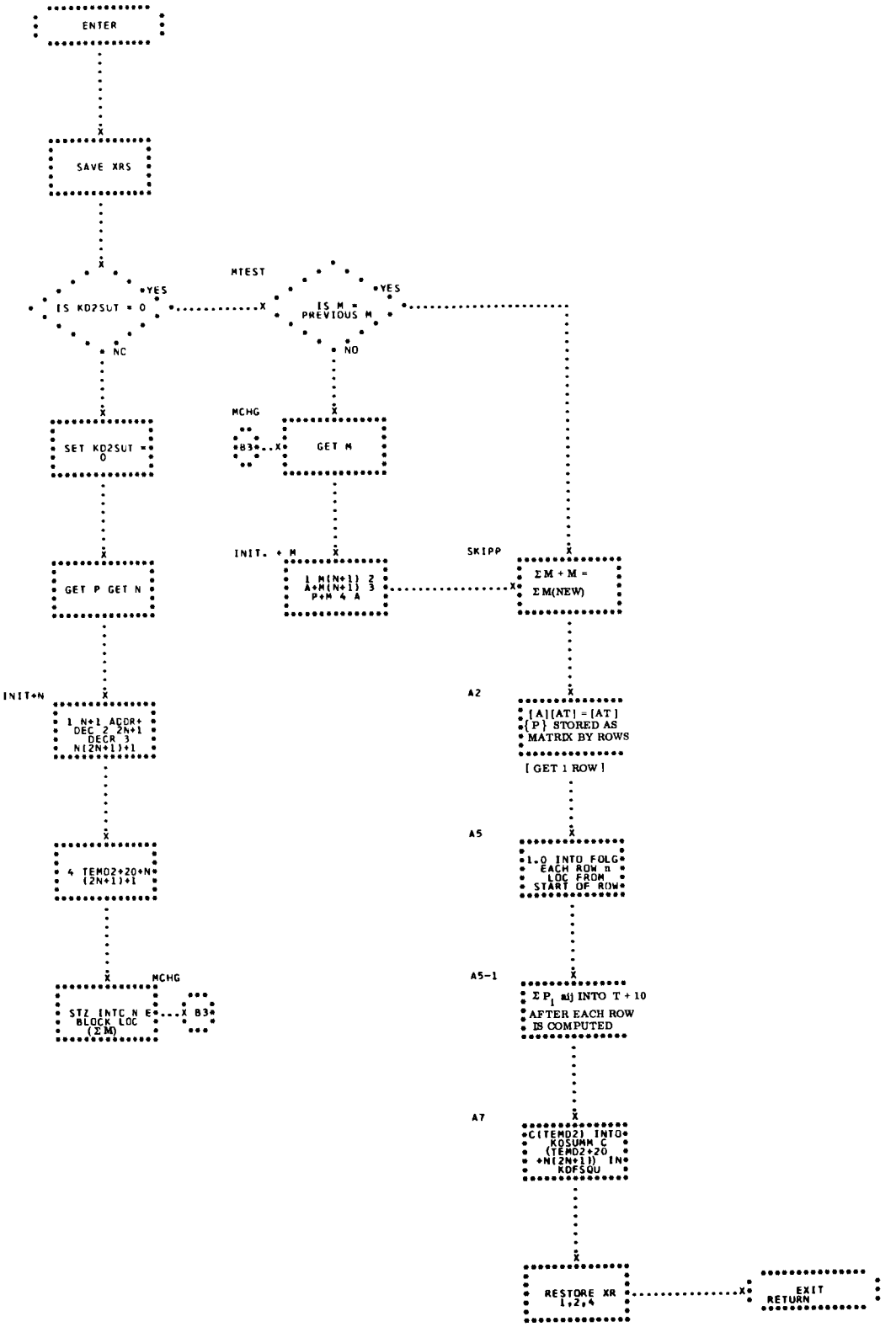
C) SPECIAL USAGE - D2SUNE MUST BE USED TO SET UP NORMAL EQUATIONS FOR SOLUTION USING D2SUNE.

D) CONSTANTS USED -

```
KOZERO    PZE
KOC001    DEC    1
KDC001    PZE    0,0,1
KOO1.0    DEC    1.
KD2SUT    PZE
KOSUMM    SUN    TDCCNT+11
KDFSQU    SYN    TDCCNT+13
```

E) ERASABLE LOCATIONS -
TEMD2, TEMD2 + 20 + N(2N + 1)
THIS TEMPORARY STORAGE CONTAINS THE MAJOR OUTPUT OF THE PROGRAM (NORMAL EQUATIONS). IT MUST BE MAINTAINED UNTIL SOLUTION BY D2SLNE IS COMPLETED AND IF DESIRED, THE STANDARD ERROR OF FIT AND ELEMENTS HAS BEEN COMPUTED BY D2CSFE.

F) CHECKOUT - D2SUNE WAS CHECKED OUT WITH D2SLNE BY COMPARING RESULTS FROM LSQ2. (SEE SHARE LIBRARY)


ENTER
SAVE XRS
MTEST
IS KD2SUT = 0
YES
NO
IS M = PREVIOUS M
YES
NO
SET KD2SUT = 0
MCHG
B3
GET M
GET P GET N
INIT. + M
1 M(N+1) 2 A+M(N+1) 3 P+M 4 A
SKIPP
ΣM + M = ΣM(NEW)
INIT+N
1 N+1 ADDR+ DEC 2 2N+1 DECR 3 N(2N+1)+1
A2
[A][AT] = [AT] {P} STORED AS MATRIX BY ROWS
[GET 1 ROW]
4 TEMD2+20+N (2N+1)+1
A5
1.0 INTO FOLG EACH ROW n LOC FROM START OF ROW
MCHG
SIZ INTC N E BLOCK LOC (2M)
B3
A5-1
ΣP1 aij INTO T + 10 AFTER EACH ROW IS COMPUTED
A7
C(TEMD2) INTO KOSUMM C (TEMD2+20 +N(2N+1)) IN KDFSQU
RESTORE XR 1,2,4
EXIT RETURN

55. D2WEQT - WEIGHTING THE EQUATIONS OF CONDITION

D2WEQT WEIGHTS PARTIAL COEFFICIENTS, COMPUTES AND WEIGHTS THE RIGHT SIDES OF THE EQUATIONS OF CONDITION, AND TESTS RESIDUALS BY COMPARISON WITH SOME CHECK FACTOR TIMES THE STANDARD ERROR. (SEE FLOWCHART.)

INPUT

THE D2WEQT PROGRAM USES THE SINE/COSINE LIBRARY SUBROUTINE UISICO. THE FOLLOWING PARAMETERS SERVE AS INPUT TO D2WEQT -

A) KDCSTT- SET = 1 TO INITIALIZE ON FIRST ENTRANCE.

B) KRESID= -0 WEIGHT PARTIALS AND RIGHT SIDES
+0 PERFORM RESIDUAL CHECK AND WEIGHT PARTIALS AND RIGHT SIDES.
+1 COMPUTE AND WEIGHT (df), PERFORM W (df) CHECK
+2 WEIGHT PARTIALS ACCORDING TO RESULTS OF W(df) CHECK.

C) KOOOON- NUMBER OF COEFFICIENT (FIXED-POINT).

D) KOSWTR- SQUARE ROOT OF STATION WEIGHT FOR RANGE.

E) KOSWTA- SQUARE ROOT OF STATION WEIGHT FOR AZIMUTH.

F) KOSWTE- SQUARE ROOT OF STATION WEIGHT FOR ELEVATION.

G) KSEMPY- STANDARD ERROR MULTIPLIED BY RESIDUAL CHECK FACTOR.

H) KOOROB- OBSERVED RANGE.

I) KOOAOB- OBSERVED AZIMUTH.

J) KOOEOB- OBSERVED ELEVATION.

K) KOODRC- COMPUTED RANGE.

L) KOODAC- COMPUTED AZIMUTH.

M) KOODEC- COMPUTED ELEVATION.

N) KOOOMB- MAXIMUM NUMBER OF EQUATIONS THAT CAN GO TO NORMAL EQUATIONS BLOCK NOT INCLUDING TWO EXTRA LOCATIONS FOR OVERFLOW.

O) TDEQCT- TABLE WITH THREE SETS OF PARTIAL COEFFICIENTS CORRESPONDING TO R, A, AND E RESPECTIVELY.

P) TLSSSE- SOLUTION OF NORMAL EQUATIONS OBTAINED FROM PREVIOUS DIFFERENTIAL CORRECTION.

Q) KRETBL- LOCATION USED IN SETTING UP INTERMEDIATE TMORES TABLE WITH W (df). USED ONLY WHEN KRESID = -0 OR +1.

R) KINSDF- =0, DO NOT PERFORM ANY CHECKS OR WEIGHT PARTIALS. ONLY COMPUTE $\Sigma W(df)^2$, ΣM
NOT = 0 - PERFORM NORMAL CALCULATIONS.

S) KTOTLM- TOTAL NUMBER OF EQUATIONS. ONLY WHEN KRESID = -0 OR +1.

T) KTDFSQ- $\Sigma W(df)^2$.

OUTPUT

THE FOLLOWING PARAMETERS ARE THE OUTPUT FROM D2WEQT-

A) KOOMCT - NUMBER OF EQUATIONS TO BE USED IN D2SUNE (SETTING UP NORMAL EQUATIONS).

B) KSUMWT - SUM OF THE WEIGHTS THAT WERE ACCEPTED.

C) TPWTDF - BLOCK CONTAINING WEIGHTED (df), NUMBER IN THE BLOCK = c(KOOMCT).

D) TAWTPC - BLOCK CONTAINING WEIGHTED PARTIALS, NUMBER IN THE BLOCK= c(KOOMCT) [c(KOOOON)].

E) KTOTLM - TOTAL NUMBER OF EQUATIONS OF CONDITION.

F) KTDFSQ - $\Sigma W(df)^2$ FOR ALL OBSERVATIONS, EVEN THOSE NOT SELECTED FOR INTERVAL.

METHOD

A) KRESID = 1 GO TO (D).
TEST R_{ob}, A_{ob}, AND E_{ob} AS MEANINGFUL DATA
+0 ----0 1 INDICATES ERRONEOUS DATA

B) $\sqrt{W_1}(R_{ob} - Rc) = W_r(df)$
$\sqrt{W_2}(\cos E_c)(A_{ob} - A_c) = W_a(dF)$
$\sqrt{W_3}(E_{ob} - E_c) = W_e(dF)$
KRESID =+0 GO TO (C)
KRESID =-0 GO TO (D)
W (dF) CHECK
W_r (dF) - KSEMPY (ALSO FOR $W_a(dF)$, $W_e(dF)$)
IF GREATER, INDICATE BY +1 IN DWDF TEST LOCATION AND REJECT DATA
EXIT FROM ROUTINE FOR W (dF) CHECK
ALPHA + 1 - ALL DATA REJECTED
ALPHA + 2 - AT LEAST ONE W(DF) PASSED THE TEST

C) RESIDUAL TEST - (RESIDUALS - KSEMPY)

IF $R_1 = \sqrt{W_1}\left[\sum_{1}^{6}(A_i X_i) - dF\right]$ KSEMPY, R_2 AND R_3 AS ABOVE, DATA IS ACCEPTED.

WHERE

a_i =PARTIALS X_i =CORRECTIONS

D) WEIGHT AND STORE PARTIALS - IF ANY TESTS HAVE BEEN MADE, PARTIALS ARE REJECTED ACCORDING TO RESULTS OF TESTS.

E) EXIT FROM THE ROUTINE IS MADE AS FOLLOWS (EXCEPT WHEN W(df) CHECK WAS MADE, AS SHOWN BEFORE) -
ALPHA + 1 INDICATES THAT MORE WEIGHTED EQUATIONS CAN BE STORED BEFORE ADDING CONTRIBUTIONS TO NORMAL EQUATIONS, THEREFORE, AT THIS POINT MORE PARTIALS SHOULD BE COMPUTED.
ALPHA + 2 INDICATES THAT BLOCKS STORING WEIGHTED

EQUATIONS ARE FILLED, THEREFORE, GOT TO SET UP NORMAL EQUATIONS.

F) IF ENTRANCE WAS MADE ON TSX D2WEQT + 1, LOCATIONS ARE SET AS IF RETURN HAD BEEN MADE TO SET UP NORMAL EQUATIONS. RETURN IS AT ALPHA + 1. NO COMPUTATION IS PERFORMED.

G) IF KINSDF=0, ONLY $\Sigma W(df)^2$ AND ΣM ARE CALCULATED AND STORED IN KTDFSQ AND KTOTLM RESPECTIVELY. RETURN IS MADE TO ALPHA + 1 (ONLY ONE RETURN).

H) WHEN KRESID= -0 (WHEN NO RESIDUAL CHECKS ARE MADE), KRETBL IS USED IN INCREMENTING LOCATIONS OF TMORES TABLE TO SET UP AN INTERMEDIATE TABLE OF W(df) FOR ALL SELECTED OBSERVATIONS.

USAGE

A) CALLING SEQUENCE -

```
TSX  D2WEQT, 4              IF PERFORMING  W(df)
                            CHECK  RETURNS IN
                            CALLING SEQUENCE INDI-
                            CATE -
RETURN FOR PARTIALS         GET ANOTHER OBSERVATION
SET UP NORMAL EQUATIONS     COMPUTE PARTIALS
```

B) SPACE REQUIRED - 187 LOCATIONS.

C) SPECIAL USAGE - TSX D2WEQT+1 (REINITIALIZES LOCATIONS).

D) CONSTANTS -

```
KOZERO      PZE
K00001      DEC  1
K00003      DEC  3
```

E) ERASABLE STORAGE - TMPD2, TMPD2+1, D2WDT, D2WDT+2

F) TIMING (IN MILLISECONDS) - WHERE p=NUMBER OF EQUATIONS THAT PASSED ANY TYPE TEST (MAX. 3), n=NUMBER OF PARTIALS (6 OR 8).

1) NO CHECKS

(A) 0.188+P(1.248+0.324N)+2.3 FOR U1SICO

(B) MAXIMUM 14.108 MSEC INCLUDING U1SICO

2) W (df) CHECK

(A) $0.24 \pm .624P+2.3$ FOR U1SICO

(B) MAXIMUM 4.532 MSEC

3) WEIGHT PARTIALS AFTER W(df) CHECK

(A) 0.432+p(0.72+0.324n)

(B) MAXIMUM 10.378 MSEC

4) RESIDUAL CHECK

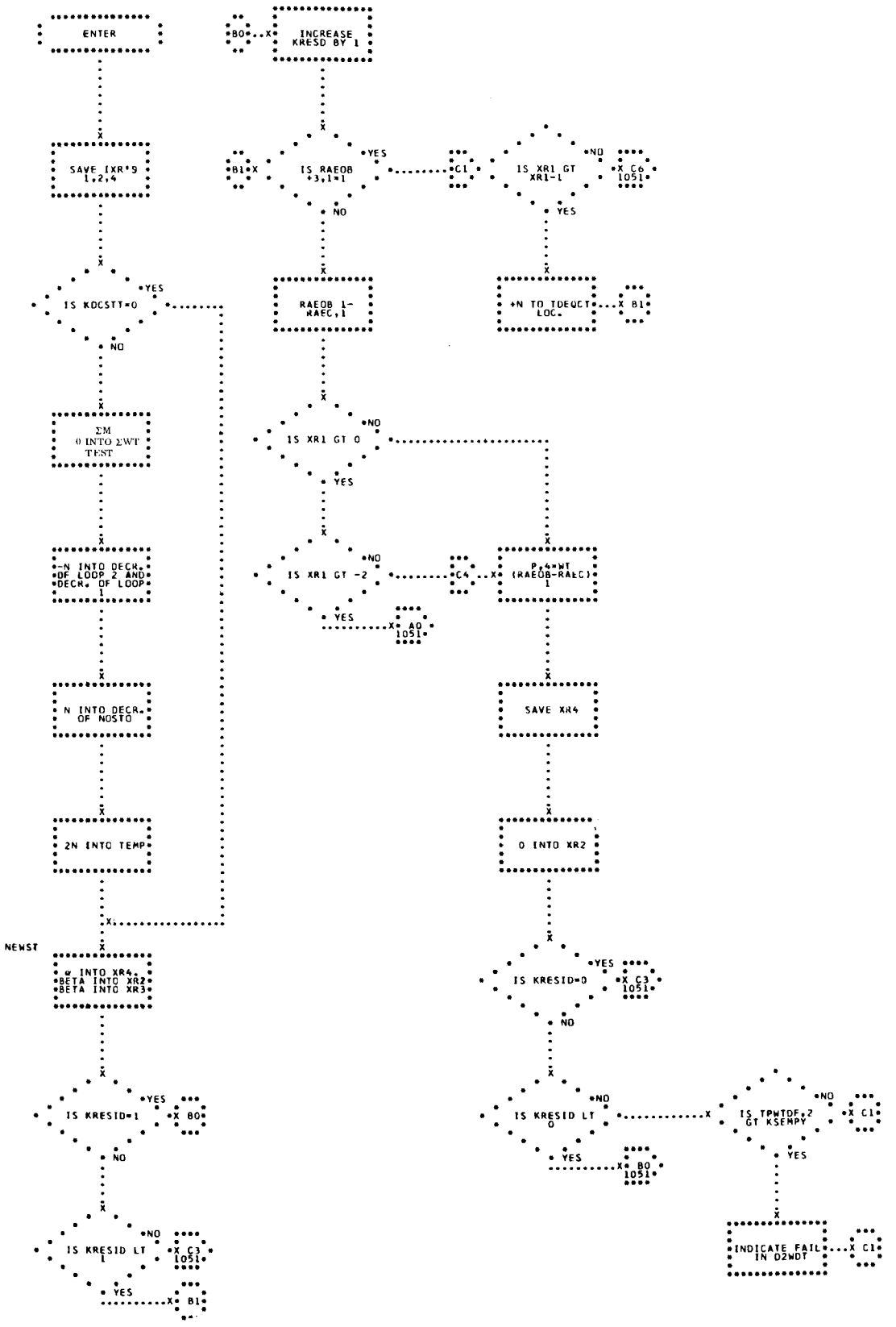
(A) 188+p(1.428+.66n)+2.3 FOR U1SICO

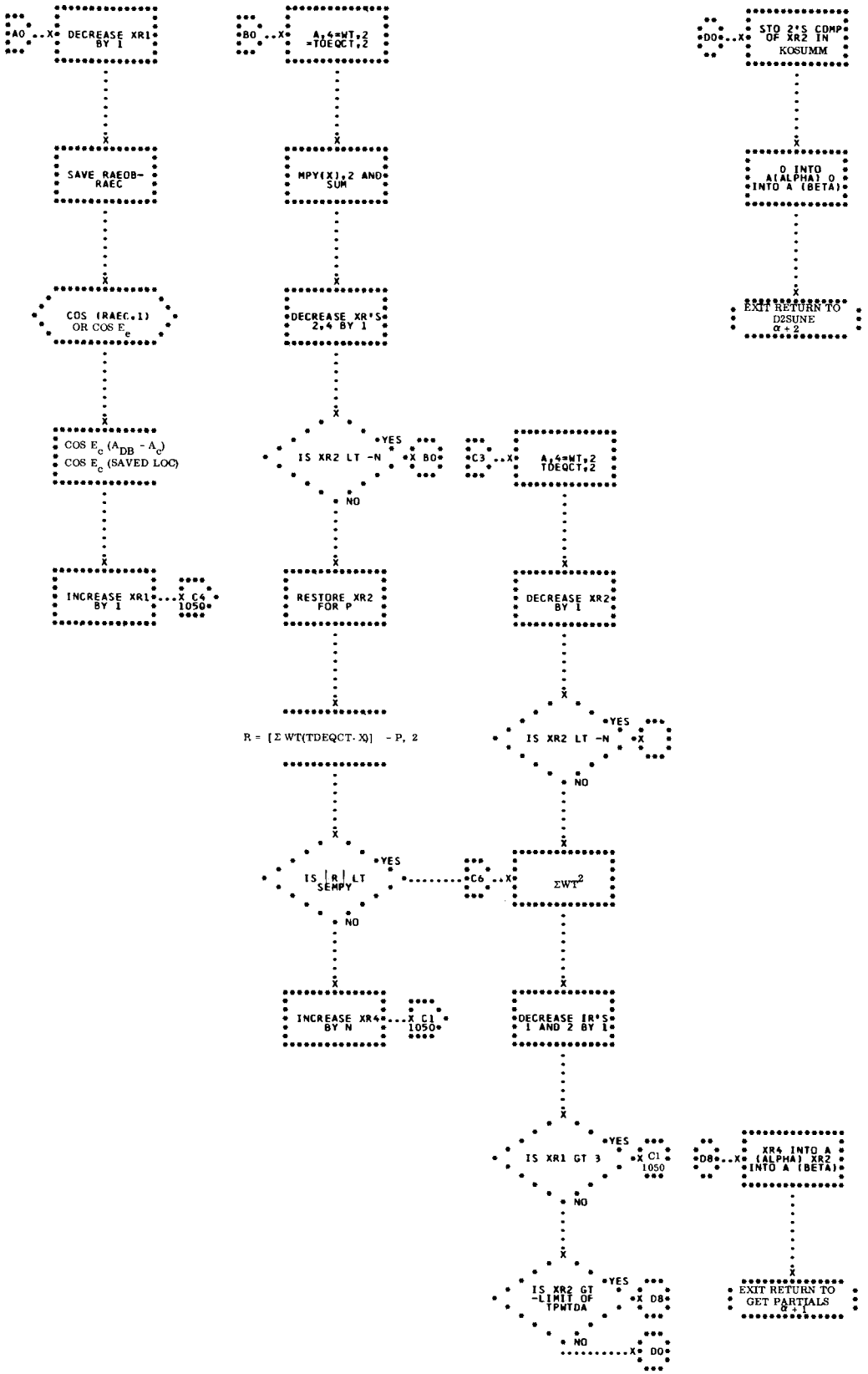
(B) MAXIMUM 22.712

5) TSX D2WEQT+1,4

(A) 0.12 MSEC

G) CHECKOUT - D2WEQT HAS BEEN SUCCESSFULLY CHECKED OUT USING ALL CASES LISTED IN METHOD.


ENTER
SAVE IXR*9 1,2,4
IS KDCSTT=0
YES
NO
ΣM 0 INTO ΣWT TEST
-N INTO DECR. OF LOOP 2 AND DECR. OF LOOP 1
N INTO DECR. OF NOSTO
2N INTO TEMP
NEWST
α INTO XR4, BETA INTO XR2 -BETA INTO XR3
IS KRESID=1
IS KRESID LT 1
B0
B1
C1
C3
1051
INCREASE KRESD BY 1
IS RAEOB +3,1=1
RAEOB 1- RAEC,1
IS XR1 GT 0
IS XR1 GT -2
A0
P,4=WT (RAEOB-RAEC) 1
SAVE XR4
0 INTO XR2
IS KRESID=0
IS KRESID LT 0
IS XR1 GT XR1-1
+N TO IDEQCT LOC.
C6
IS TPWTDF,2 GT KSEMPY
INDICATE FAIL IN D2WDT


A0
DECREASE XR1 BY 1
SAVE RAEOB-RAEC
COS (RAEC+1) OR COS E_c
COS E_c (A_DB - A_C) COS E_c (SAVED LOC)
INCREASE XR1 BY 1
C4 1050
B0
A,4=WT,2 =TDEQCT,2
MPY(X),2 AND SUM
DECREASE XR'S 2,4 BY 1
IS XR2 LT -N
YES
NO
X B0
RESTORE XR2 FOR P
R = [Σ WT(TDEQCT·X)] - P, 2
IS |R| LT SEMPY
YES
NO
INCREASE XR4 BY N
C1 1050
C3
A,4=WT,2 TDEQCT,2
DECREASE XR2 BY 1
IS XR2 LT -N
YES
NO
C6
ΣWT²
DECREASE IR'S 1 AND 2 BY 1
IS XR1 GT 3
YES
C1 1050
NO
IS XR2 GT -LIMIT OF TPWTDA
YES
D8
NO
D0
D0
STO 2'S COMP OF XR2 IN KOSUMM
0 INTO A(ALPHA) 0 INTO A (BETA)
EXIT RETURN TO D2SUNE α+2
D8
XR4 INTO A (ALPHA) XR2 INTO A (BETA)
EXIT RETURN TO GET PARTIALS α+1

56: D3CDRM - CHOOSE DESIRED RADAR MESSAGE

D3CDRM CHOOSES THE DESIRED NUMBER OF RADAR MESSAGES FROM THE STATION BLOCK AND STORES TIME, RANGE, AZIMUTH, AND ELEVATION FOR EACH MESSAGE STARTING WITH THE EARLIEST MESSAGE. (SEE FLOWCHART.)

INPUT

SYMBOLIC LOCATION	DEFINITION
KOLSMB SYN	TDCCNT + 7 ADDRESS CONTAINS LOCATION OF FIRST WORD OF STATION MESSAGE BLOCK.
KONDRM SYN	TDCCNT + 8 DESIRED NUMBER OF RADAR MESSAGES.
KCDRMI SYN	TDCCNT + 6 S, 1-34 BITS ALL ZEROS . IF 35=1, FIRST TIME THROUGH D3CDRM, =0 ALL SUBSEQUENT TIMES THROUGH D3CDRM.
STATION MESSAGE BLOCK	
Y	FIRST WORD OF STATION MESSAGE. ADDRESS CONTAINS STATION IDENTIFICATION NUMBER. DECREMENT CONTAINS PASS NUMBER.
Y + 1	NUMBER OF RADAR MESSAGES IN THIS BLOCK.
Y + 2	ADDRESS CONTAINS THE LOCATION, RELATIVE TO Y, OF THE

	TIME OF MINIMUM RANGE (T_m)
Y + 3	T_1 (TIME) FIXED-POINT, MINUTES IN ADDRESS, SECONDS IN DECREMENT.
Y + 4	R_1 (RANGE) FLOATING-POINT, IN GEMINI UNITS.
Y + 5	A_1 (AZIMUTH) FLOATING-POINT, IN RADIANS.
Y + 6	E_1 (ELEVATION) FLOATING-POINT, IN RADIANS.
Y + 7---Y + 10	SECOND RADAR MESSAGE.
Y + 4N - 1---Y + 4N+2	NTH RADAR MESSAGE.
TMSTCH	ADDRESS CONTAINS THE LOCATION-1 OF THE STATION CHARACTERISTICS BLOCK TO BE USED. (SEE)

OUTPUT

SYMBOLIC LOCATION		DEFINITION
KOSWTR SYN	TEQCIN	SQUARE ROOT OF STATION WEIGHT FOR RANGE.
KOSWTA SYN	TEQCIN + 1	SQUARE ROOT OF STATION WEIGHT FOR AZIMUTH.
KOSWTE SYN	TEQCIN + 2	SQUARE ROOT OF STATION WEIGHT FOR ELEVATION.
KOOTOB SYN	TEQCIN + 3	TIME OF RADAR MESSAGE JUST PROCESSED.
KOOROB SYN	TEQCIN + 9	RANGE OF RADAR MESSAGE JUST PROCESSED.
KOOAOB SYN	TEQCIN + 10	AZIMUTH OF RADAR MESSAGE JUST PROCESSED.
KOOEOB SYN	TEQCIN + 11	ELEVATION OF RADAR MESSAGE

		JUST PROCESSED.
KSTCHL SYN	TEQCIN + 7	ADDRESS CONTAINS THE NUMBER IDENTIFYING WHICH STATION CHARACTERISTICS BLOCK WAS USED.
KTSUBI SYN	TEQCIN + 8	TIME OF OBSERVATION IN GEMINI UNITS, FLOATING-POINT

METHOD

THE MESSAGE (T_m) AT THE TIME OF MINIMUM RANGE AND MAXIMUM ELEVATION AND THOSE MESSAGES CLOSEST IN TIME TO (T_m) -TOTALING THE DESIRED NUMBER OF MESSAGES -ARE STORED IN ORDER OF INCREASING TIME STARTING WITH THE EARLIEST.

IF, IN TESTING MESSAGES TO BE SELECTED ON THE BASIS OF TIME CLOSEST TO T_m THE FIRST MESSAGE OF THE STATION MESSAGE BLOCK IS ACCEPTED, THE PROGRAM NAMES IT ''THE FIRST DESIRED MESSAGE,'' AND AUTOMATICALLY SELECTS ENOUGH SEQUENTIAL MESSAGES AFTER T_m TO REACH THE NUMBER DESIRED.

IF, IN TESTING MESSAGES TO BE SELECTED ON THE BASIS OF TIME CLOSEST TO T_m THE LAST MESSAGE OF THE STATION BLOCK IS ACCEPTED, THE FIRST MESSAGE IS AUTOMATICALLY T-4 (NUMBER OF DESIRED MESSAGES-1), WHERE T IS THE NUMBER OF THE LAST MESSAGE. ALL MESSAGES BETWEEN T-4 AND T_m ARE THUS SELECTED.

AFTER SELECTING THE DESIRED MUMBER OF MESSAGES, THE FIRST ONE IS STORED AND CONTROL IS RETURNED TO 2, 4. ON EACH SUBSEQUENT TSX TO D3CDRM, ANOTHER OF THE SELECTED MESSAGES IS STORED, UNTIL ALL DESIRED MESSAGES HAVE BEEN STORED. THEN A FINAL RETURN IS MADE TO 1, 4.

USAGE

A) CALLING SEQUENCE -

```
TSX         D3CDRM,4
            ALL DESIRED MESSAGES HAVE BEEN
            STORED, FINAL RETURN.
            ANOTHER SELECTED MESSAGE HAS BEEN
            STORED, BUT NOT ALL OF THEM.
```

B) SPACE REQUIRED - 150 LOCATIONS USED (THE LAST 14 OF WHICH ARE ERASABLE) EXCLUDING CONSTANTS, INPUT, AND OUTPUT.

C) CONSTANTS USED -

```
K00001      DEC     1
K00002      DEC     2
K00003      DEC     3
K00004      DEC     4
K00005      DEC     6
K00.5H      DEC     .00123857
KCH233      OCT     233000000000
```

D) TIMING - D3CDRM REQUIRES 2.3 + N(1.2) MSEC FOR THE FIRST TIME THROUGH PLUS (1.3)N MSEC FOR THE REMAINING TIMES THROUGH (TOTAL), WHERE N = THE DESIRED NUMBER OF RADAR MESSAGES AND THE NUMBER OF TIMES CONTROL IS TRANSFERRED TO D3CDRM. TOTAL TIME IS 2.3 + N(2.5)MSEC.

E) CHECKOUT - TABLE 1 SHOWS TIMES FOR INPUT MESSAGES USED TO TEST D3CDRM. A BLOCK OF TEN MESSAGES WAS SET UP WITH TIME EQUAL TO 1342 OCTAL MINUTES AND SECONDS VARYING FROM 0.62 OCTAL. FOUR TESTS WERE MADE.

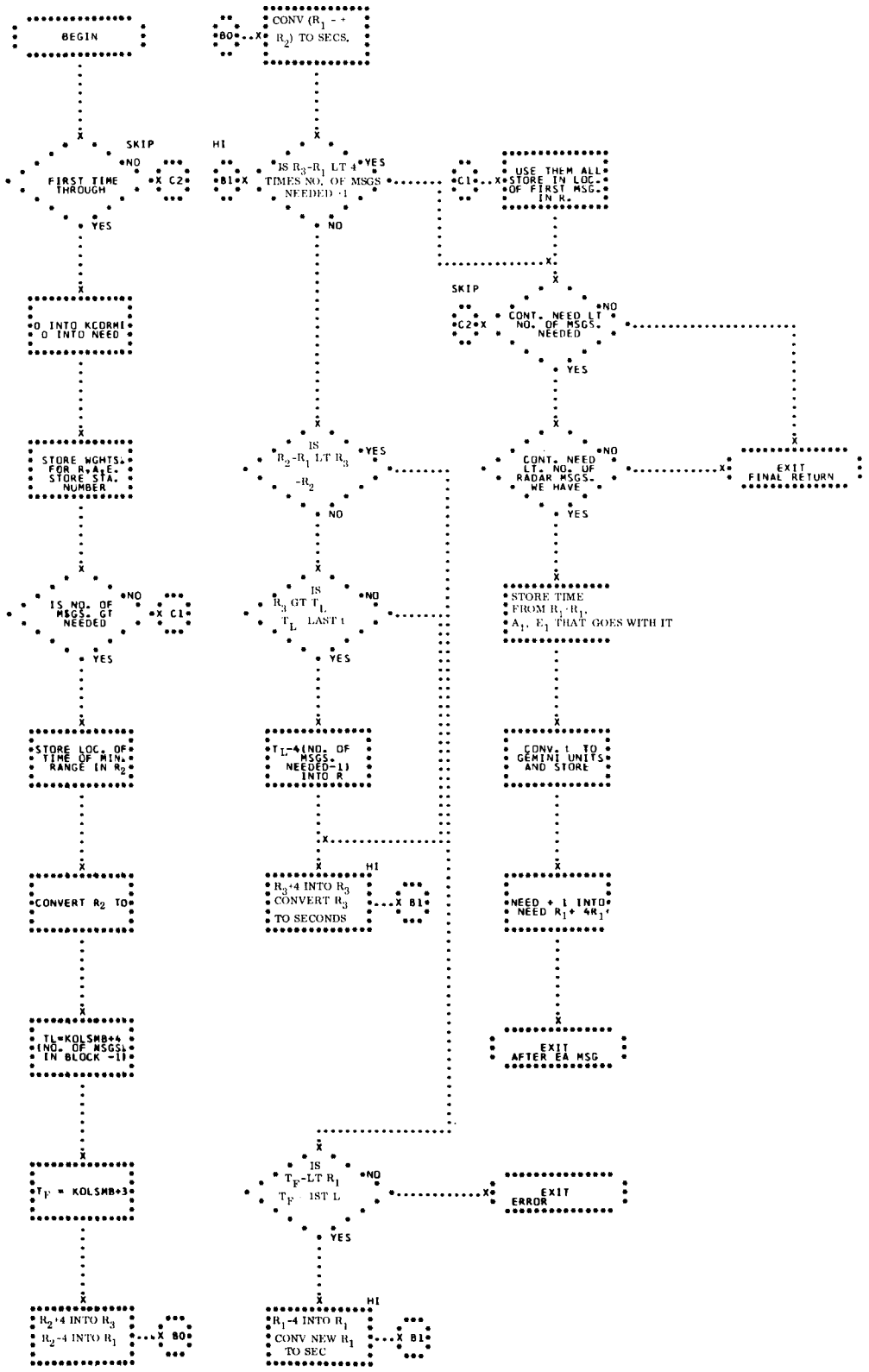
1) IN THE NORMAL CASE, ONLY PART OF THE BLOCK IS NEEDED AND THE MESSAGES ARE IN THE CENTER OF THE BLOCK.

2) ALL OF THE MESSAGES ARE NEEDED.
3) PART OF THE MESSAGES ARE NEEDED FROM THE END OF THE BLOCK.
4) PART OF THE MESSAGES ARE NEEDED FROM THE BEGINNING OF THE BLOCK.

TABLE 1

OCTAL SECS.	*DECR. SECS.	MSG.	TEST 1	TEST 2	TEST 3	TEST 4
			Desire 5 of the 10 msg. Min. range and max. elev. is 7.	Ten msgs. are desired from the blk. of 10. All are accepted, no tests are made.	Six of 10 are desired. Min. range & max. elev. is 9. When msg. No. 10 is hit, the program takes the last 6 msg. in the blk.	Same as No. 3 except min. range & max. elev. is No. 3.
0	0	1				14 4
6	6	2	Time dif. in sec.			8 9
16	14	3	22			0 1
23	19	4	17 The excepted 5			5 2
35	29	5	7 4		6	15 5
37	31	6	5 3		5	6
44	36	7	0 1		4	When msg. No. 1 was hit the program
51	41	8	5 2		6 3	took the first 6 msgs. in the blk.
57	47	9	11 5		0 1	
62	50	10	16		3 2	

*Time = 1342 min + x seconds


BEGIN
B0
CONV (R_1 - R_2) TO SECS.
SKIP
NO
FIRST TIME THROUGH
X C2
YES
HI
B1
IS R_3-R_1 LT 4 TIMES NO. OF MSGS NEEDED -1
YES
NO
C1
USE THEM ALL STORE IN LOC. OF FIRST MSG. IN R_1
SKIP
C2
CONT. NEED LT NO. OF MSGS. NEEDED
NO
YES
0 INTO KCORMI
0 INTO NEED
STORE WGHTS. FOR R, A, E. STORE STA. NUMBER
IS R_2-R_1 LT R_3 -R_2
YES
NO
CONT. NEED LT NO. OF RADAR MSGS. WE HAVE
NO
YES
EXIT FINAL RETURN
IS NO. OF MSGS. GT NEEDED
NO
X C1
YES
IS R_3 GT T_L T_L LAST t
NO
YES
STORE TIME FROM R_1-R_1, A_1, E_1 THAT GOES WITH IT
STORE LOC. OF TIME OF MIN. RANGE IN R_2
T_L-4(NO. OF MSGS. NEEDED-1) INTO R
CONV. t TO GEMINI UNITS AND STORE
CONVERT R_2 TO
R_3+4 INTO R_3 CONVERT R_3 TO SECONDS
HI
X B1
NEED + 1 INTO NEED R_1+ 4R_1'
TL=KOLSMB+4 (NO. OF MSGS) IN BLOCK -11
EXIT AFTER EA MSG
T_F = KOLSMB+3
IS T_F-LT R_1 T_F 1ST L
NO
YES
EXIT ERROR
R_2+4 INTO R_3 R_2-4 INTO R_1
X B0
R_1-4 INTO R_1 CONV NEW R_1 TO SEC
HI
X B1

57. D3PTWT - WEIGHT TABLE LOOKUP

D3PTWT DETERMINES THE NUMBER OF OBSERVATIONS DESIRED FROM EACH REPORTING STATION AND COMPARES THIS NUMBER TO THE NUMBER OF AVAILABLE OBSERVATIONS. THE SMALLER OF THE TWO NUMBER OF OBSERVATIONS IS USED IN THE DIFFERENTIAL CORRECTION (DC) PROGRAM FOR THAT STATION. (SEE FLOWCHART)

INPUT

A) TMSTMS - A TABLE OF MNLTMS ENTRIES, CONTAINING IN THE ADDRESS FIELD OF EACH ENTRY THE LOCATION OF THE TMDATA BLOCK ASSIGNED TO EACH STATION WHICH IS TO BE USED IN THE DC. THE ENTRIES ARE ORDERED ACCORDING TO TIME OF MINIMUM RANGE (TMR) WITH THE ENTRY FOR THE MOST RECENT TIME FIRST. THE LAST ENTRY IN THE TABLE IS SET TO NEGATIVE. ALL PRECEDING ENTRIES ARE POSITIVE, WHILE ALL REMAINING LOCATIONS ARE NEGATIVE ZERO.

B) TMDATA - A TABLE COMPOSED OF 3-WORD BLOCKS, ONE FOR EACH LOW-SPEED EDITED DATA BLOCK LOCATED IN 32K BUFFER MODULE. THE FIRST WORD CONTAINS THE STATION IDENTIFICATION NUMBER, THE SECOND WORD CONTAINS THE NUMBER OF RADAR MESSAGES FROM THE STATION, WHILE THE THIRD WORD CONTAINS THE ANCHOR TIME IN FIXED-POINT SECONDS IN THE DECREMENT AND FIXED-POINT MINUTES IN THE ADDRESS.

OUTPUT

THE OUTPUT OF D3PTWT IS TNOMFS, A TABLE CONTAINING THE NUMBER OF OBSERVATION TO BE USED FROM EACH STATION BY THE DC PROGRAM.

METHOD

D3PTWT FIRST COUNTS THE NUMBER OF REPORTING STATIONS FOR A DC, AND THEN DETERMINES, BY MEASURING THE TIME SPAN OF THE DATA, WHICH OF THE FOLLOWING THREE CONDITIONS EXIST -

A) MORE THAN ONE ORBIT

B) LESS THAN ONE ORBIT, MORE THAN TEN STATIONS

C) LESS THAN ONE ORBIT, TEN OR LESS STATIONS.

TO ACHIEVE THE BEST POSSIBLE CORRECTION, THE NUMBER OF OBSERVATIONS DESIRED FROM EACH STATION IS COMPUTED BASED ON TIME DIFFERENCE BETWEEN THE TIME OF MINIMUM RANGE (TMR) OF THE STATION AND THE TIME OF THE ANCHOR POINT VECTOR. THIS NUMBER IS THEN COMPARED TO THE NUMBER OF OBSERVATIONS AVAILABLE FROM THAT STATION AND THE SMALLER OF THE TWO NUMBERS IS USED.

USAGE

A) CALLING SEQUENCE

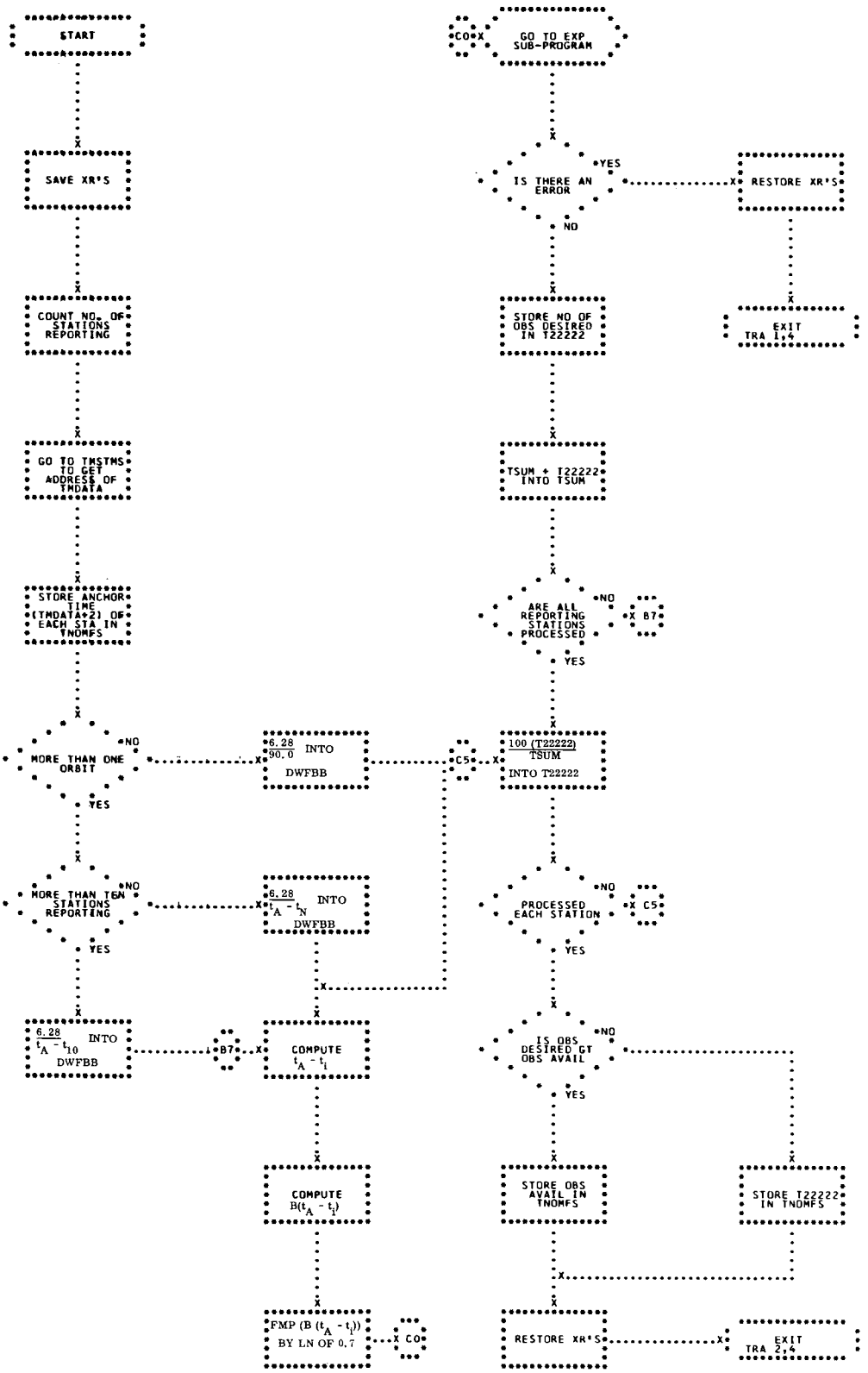
```
TSX     D3PTWT,4
ERROR RETURN
NORMAL RETURN
```

B) STORAGE REQUIRED - 168 LOCATIONS

C) CONSTANTS USED - KCH233, K00001, K090.0, KKK2PI, K00010, KLN0.7, K100.0

D) ROUTINES USED - U1EXPE

E) CHECKOUT - THIS PROGRAM HAS BEEN CHECKED BY USING A VARIETY OF OBSERVATIONS OVER DIFFERENT TIME PERIODS.


START
SAVE XR'S
COUNT NO. OF STATIONS REPORTING
GO TO TMSTMS TO GET ADDRESS OF TMDATA
STORE ANCHOR TIME (TMDATA+2) OF EACH STA IN TNOMFS
MORE THAN ONE ORBIT
NO
YES
6.28/90.0 INTO DWFBB
MORE THAN TEN STATIONS REPORTING
NO
YES
6.28/(t_A - t_N) INTO DWFBB
6.28/(t_A - t_10) INTO DWFBB
B7
COMPUTE t_A - t_i
COMPUTE B(t_A - t_i)
FMP (B (t_A - t_i)) BY LN OF 0.7
CO
GO TO EXP SUB-PROGRAM
CO
IS THERE AN ERROR
YES
NO
RESTORE XR'S
EXIT TRA 1,4
STORE NO OF OBS DESIRED IN T22222
TSUM + T22222 INTO TSUM
ARE ALL REPORTING STATIONS PROCESSED
NO
B7
YES
100 (T22222)/TSUM INTO T22222
C5
PROCESSED EACH STATION
NO
C5
YES
IS OBS DESIRED GT OBS AVAIL
NO
YES
STORE OBS AVAIL IN TNOMFS
STORE T22222 IN TNOMFS
RESTORE XR'S
EXIT TRA 2,4

58. D4CFRV - CALCULATE FUNCTIONS OF $\vec{r}$ AND $\vec{v}$

D4CFRV CALCULATES THE QUANTITITES (FUNCTIONS OF $\vec{r}, \vec{v}$) FROM A GIVEN SET OF SPACECRAFT POSITION AND VELOCITY VALUES. (SEE FLOWCHART.)

INPUT

THIS PROGRAM USES THREE LIBRARY SUBROUTINES U1SQRT, U3XPRO, AND U3DOTP.

A) KRAVC1 - X COORDINATE (ALL IN FLOATING-POINT)
B) KRAVC2 - Y COORDINATE
C) KRAVC3 - Z COORDINATE
D) KVAVC1 - VELOCITY ALONG THE X AXIS
E) KVAVC2 - VELOCITY ALONG THE Y AXIS
F) KVAVC3 - VELOCITY ALONG THE Z AXIS

OUTPUT

A) KASUBO - a_0 (ALL IN FLOATING-POINT)
B) KKMESQ - $1-e^2$
C) KRABSL - $\sqrt{X^2 + Y^2 + Z^2}$
D) BVABSL - $\sqrt{VX^2 + VY^2 + VZ^2}$
E) KECCEN - e
F) KNSUBO - $a_0^{-3/2}$
G) KREVEC1 - LARGE R VECTOR
H) KREVC2 - LARGE R VECTOR
I) KREVC3 - LARGE R VECTOR
J) KSRTPO - $P\sqrt{P}$

METHOD

THESE EQUATIONS ARE USED BY D4CFRV -

$$|\vec{r}_0| = \left(X^2 + Y^2 + Z^2\right)^{1/2} \tag{1}$$

$$|\vec{v}_0| = \left(VX^2 + VY^2 + VZ^2\right)^{1/2} \tag{2}$$

$$a_0 = \frac{r_0}{2 - \vec{r}_0 \vec{v}_0^{\,2}} \tag{3}$$

$$e^2 = \left(\frac{a_0 - r_0}{a_0}\right)^2 + \frac{\left(\vec{r}_0 \cdot \vec{v}_0\right)^2}{a_0} \tag{4}$$

$$N_0 = a_0^{-3/2} \tag{5}$$

$$\sqrt{P} = |\vec{r}_0 \times \vec{v}_0| \tag{6}$$

$$\vec{R} = \frac{\vec{r}_0 \times \vec{v}_0}{\sqrt{P}} \tag{7}$$

D4CFRV IS ACCURATE TO SEVEN SIGNIFICANT DIGTIS.

USAGE

A) CALLING SEQUENCE -

```
TSX          D4CFRV,4
             ERROR RETUNR
             NORMAL RETURN
```

B) SPACE REQUIRED - 107 LOCATIONS EXCLUDING LOCATIONS FOR LIBRARY SUBROUTINES AND TABLES.

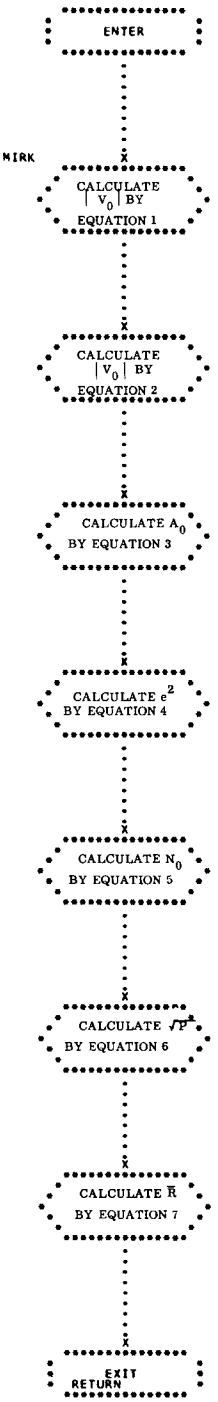
C) CONSTANTS USED -

```
K001.0  DEC  1.0
K002.0  DEC  2.0
```

D) ERASABLE LOCATIONS - TEMD4 THROUGH TEMD4+9.

E) TIMING REQUIRED - 12.9 MILLISECONDS.

F) CHECKOUT - D4CFRV WAS CHECKED OUT BY COMPARING ITS OUTPUT WITH RESULTS OF HAND COMPUTATIONS, AND WITH THE OUTPUT FROM THE COG PROGRAM.


ENTER
MIRK
CALCULATE $|V_0|$ BY EQUATION 1
CALCULATE $|V_0|$ BY EQUATION 2
CALCULATE A_0 BY EQUATION 3
CALCULATE e^2 BY EQUATION 4
CALCULATE N_0 BY EQUATION 5
CALCULATE $\sqrt{P^2}$ BY EQUATION 6
CALCULATE $\bar{R}$ BY EQUATION 7
EXIT RETURN

59. R1NDCP - NUMERICAL DIFFERENTIAL CORRECTION

R1NDCP COMPUTES BY NUMERICAL MEANS THE 18 PARTIAL DERIVATIVES APPEARING IN THE THREE EQUATIONS OF CONDITIONS ARISING FROM AN OBSERVATION. R1NDCP UTILIZES THE SIX INTEGRATION TABLES SET UP BY R2AINT. (SEE FLOWCHART)

INPUT

R1NDCP USES TWO LIBRARY SUBROUTINES - UA1LSC, U7INTP AND VANGUARD PROGRAM INTP3 ADAPTED FOR USE AS A CLOSED SUBROUTINE WITHIN R1NDCP. THE INPUT TO R1NDCP INCLUDES THE FOLLOWING -

A) A FIVE-WORD BLOCK BEGINNING AT LOCATION KOOTOB.

KOOTOB SYN TEQCIN + 3 TIME OF THE OBSERVATION -MINUTES IN THE ADDRESS, SECONDS IN THE DECREMENT.

+1
+2 } USED BY R1NDCP FOR TEMPORARY STORAGE.
+3

+4 STATION IDENTIFICATION

B) THE COMPUTED RANGE, AZIMUTH AND ELEVATION OF THE SPACECRAFT RELATIVE TO THE SPECIFIED STATION AT THE GIVEN TIME OF THE OBSERVATION BASED ON THE LATEST REENTRY TABLE.

KOODRC SYN TEQCIN +15 R_c
KOODAC SYN TEQCIN +16 A_c
KOODEC SYN TEQCIN +17 E_c

C) THE SIX PERTURBED INTEGRATION TABLES GENERATED IN THE PROGRAM R2AINT.

D) TDTOBS BSS 1 TIME OF THE OBSERVATION IN

			SECONDS
E)	TNETIM	BSS 1	TIME IN FIXED-POINT SECONDS OF THE LAST ENTRY IN THE LAST OF THE SIX REENTRY TABLES OF PREDICTED POSITION.

OUTPUT

EIGHTEEN PARTIAL DERIVATIVES ARE COMPUTED AND STORED BEGINNING IN LOCATION TDEQCT IN THE FOLLOWING ORDER -

$$\frac{\partial R}{\partial X}, \frac{\partial R}{\partial Y}, \frac{\partial R}{\partial Z}, \frac{\partial R}{\partial \dot{X}}, \frac{\partial R}{\partial \dot{Y}}, \frac{\partial R}{\partial \dot{Z}}, \frac{\partial A}{\partial X}, \ldots \frac{\partial A}{\partial \dot{Z}}, \frac{\partial E}{\partial X}, \ldots, \frac{\partial E}{\partial \dot{Z}}$$

METHOD

IN ADDITION TO THE LATEST REENTRY TRAJECTORY, THERE ARE AVAILABLE TO THE RINDCP PROGRAM SIX INTEGRATION TABLES WHICH HAVE BEEN GENERATED BY PERTURBING THE POSITION AND VELOCITY VECTORS AT THE LATEST ANCHOR POINT r_a AND V_a BY AMOUNTS ΔX_a, $\Delta Y_a \ldots . \Delta Z_a$, RESPECTIVELY. LET I REPRESENT ANY ONE OF THE VARIABLES X, Y, Z, $\dot{X}$, $\dot{Y}$, $\dot{Z}$. THE PARTIALS $\frac{\partial R}{\partial I}$, $\frac{\partial A}{\partial I}$, $\frac{\partial E}{\partial I}$ ARE CALCULATED AS FOLLOWS. FROM THE INTEGRATION TABLE GENERATED BY PERTURBING THE I COMPONENT OF r_a OR v_a BY THE AMOUNT ΔI_a, COMPUTE THE RANGE R_I, AZIMUTH A_I, AND ELEVATION E_I OF THE SPACECRAFT RELATIVE TO THE GIVEN STATION AT THE SPECIFIED OBSERVATION TIME. THEN

$$\frac{\partial R}{\partial I} \frac{R_I - R_C}{\Delta I_a}, \frac{\partial A}{\partial I} \frac{A_I - A_C}{\Delta I_a}, \frac{\partial E}{\partial I} \frac{E_I - E_C}{\Delta I_a}$$

WHERE R_c, A_c, E_c ARE THE COMPUTED RANGE, AZIMUTH, AND ELEVATION, RESPECTIVELY, BASED ON THE LATEST PREDICTED REENTRY TRAJECTORY.

IN THE FOLLOWING DISCUSSION, Δt IS THE TIME INTERVAL AT WHICH ENTRIES ARE MADE IN THE SIX NDC INTEGRATION TABLES, AND T_L IS THE TIME OF THE LAST ENTRY IN THE NUMERICAL DIFFERENTIAL CORRECTION TABLE.

TO CALCULATE BY 6 POINT LAGRANGIAN INTERPOLATION FOR AN OBSERVATION TIME WHICH OCCURS WITHIN THE TIME INTERVAL $t_L - 2\Delta t$ TO t_L, RINDCP IGNORES ANY OBSERVATION AT A TIME GREATER THAN t_L. IF AN OBSERVATION TIME FALLS BETWEEN $t_L - \Delta t$ TO $t_L - 2\Delta t$ OR BETWEEN t_L TO $t_L - \Delta t$, THE VANGUARD PROGRAM INTP3 IS USED.

THE FOLLOWING TESTS FOR DISCONTINUITY IN AZIMUTH NEAR 360° ARE MADE PRIOR TO COMPUTING $\frac{A_I - A_c}{\Delta I_a}$

$A_I < 90^\circ$, $A_c > 270^\circ$	Compute	$\frac{A_I + 2\pi - A_c}{\Delta I_a}$
$A_I < 90^\circ$, $A_c \leq 270^\circ$	Compute	$\frac{A_I - A_c}{\Delta I_a}$
$A_I \leq 270^\circ$ and $\geq 90^\circ$	Compute	$\frac{A_I - A_c}{\Delta I_a}$
$A_I > 270^\circ$, $A_c \geq 90^\circ$	Compute	$\frac{A_I - A_c}{\Delta I_a}$
$A_I > 270^\circ$, $A_c < 90^\circ$	Compute	$\frac{A_I - (A_c + 2\pi)}{\Delta I_a}$

RINDCP CAN BE USED ONLY WITH INTEGRATION TABLES THAT STORE $\vec{r}$ VECTORS ALONE.

USAGE

A) CALLING SEQUENCE -

```
TSX     RINDCP,4
```

ERROR RETURN

NORMAL RETURN

B) SPACE REQUIRED - 194_{10}

C) ERROR CODES - IF THE TIME IS BELOW t_L PROGRAM RETURNS TO ERROR RETURN.

D) CONSTANTS USED -

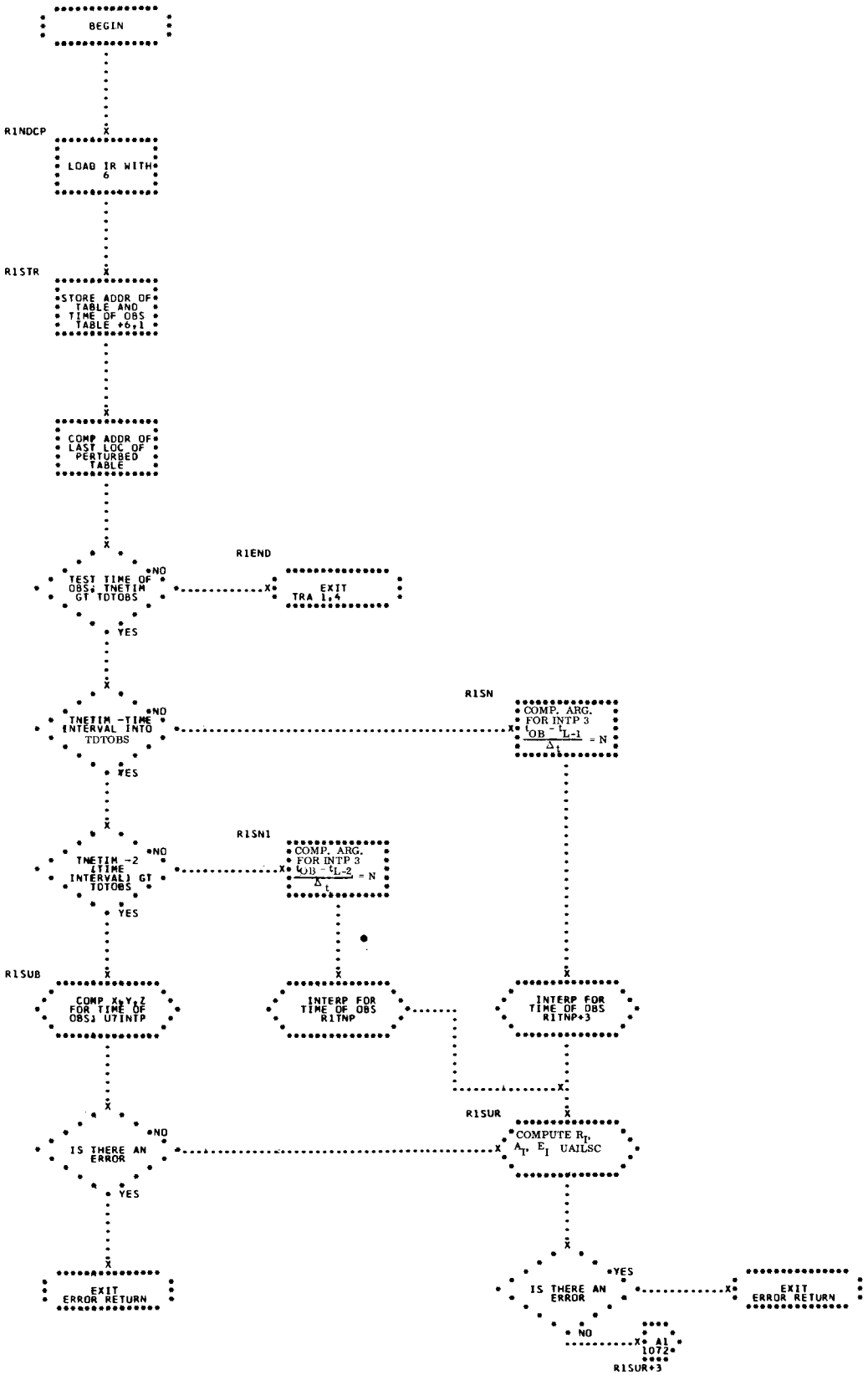
KOOTOB	SYN	TEQCIN+3	
KOODRC	SYN	TEQCIN+15	
KOODAC	SYN	TEQCIN+16	
KOODEC	SYN	TEQCIN+17	
KOO4.0	DEC	4.0	
KOO5.0	DEC	5.0	
KOO1.0	DEC	1.0	
KOOOO1	DEC	1	
KCH233	OCT	233000000000	
KOOOO3	DEC	3	
KOOOO7	DEC	7	
KOZERO	DEC	0	

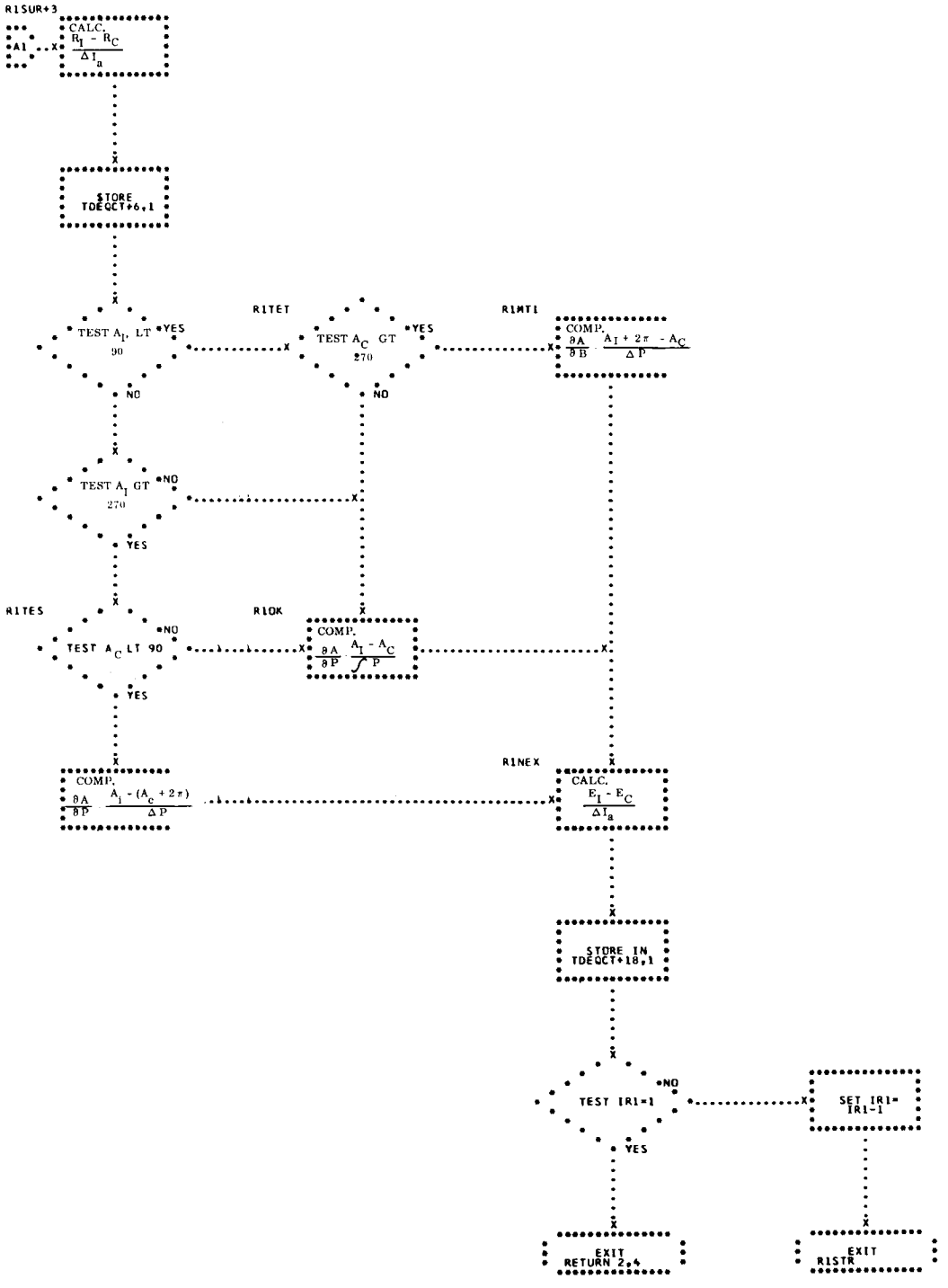
E) TABLES USED -

TPA.RX	BSS	6	AMOUNT TO PERTURB THE VECTORS BY.
TDEQCT	BSS	24	OUTPUT TABLE
TNINT3	BSS	112	X PERTURBED TABLE
TNINT4	BSS	112	Y PERTURBED TABLE
TNINT5	BSS	112	Z PERTURBED TABLE
TNINT6	BSS	112	PERTURBED TABLE
TNINT7	BSS	112	PERTURBED TABLE
TNINT8	BSS	112	PERTURBED TABLE
TNETIM	PZE		TIME OF LAST OBSERVATION OF LAST REENTRY TABLE.
TDTOBS	PZE		TIME OF OBSERVATION IN SECONDS.

F) TIMING - TIMING FOR RINDCP IS 458 MICROSECONDS.

G) CHECKOUT - RINDCP WAS CHECKED OUT IN THE GEMINI PROGRAMMING SYSTEM USING ACTUAL TEST DATA SUPPLIED BY NASA.


BEGIN
R1NDCP
LOAD IR WITH 6
R1STR
STORE ADDR OF TABLE AND TIME OF OBS TABLE +6,1
COMP ADDR OF LAST LOC OF PERTURBED TABLE
R1END
TEST TIME OF OBS; TNETIM GT TDTOBS
NO
YES
EXIT TRA 1,4
TNETIM -TIME INTERVAL INTO TDTOBS
R1SN
COMP. ARG. FOR INTP 3
$\frac{t_{OB} - t_{L-1}}{\Delta t} = N$
TNETIM -2 (TIME INTERVAL) GT TDTOBS
R1SN1
COMP. ARG. FOR INTP 3
$\frac{t_{OB} - t_{L-2}}{\Delta t} = N$
R1SUB
COMP X,Y,Z FOR TIME OF OBS; UTINTP
INTERP FOR TIME OF OBS R1TNP
INTERP FOR TIME OF OBS R1TNP+3
R1SUR
COMPUTE R_I, A_I, E_I UAILSC
IS THERE AN ERROR
YES
NO
EXIT ERROR RETURN
EXIT ERROR RETURN
A1 1072
R1SUR+3


R1SUR+3
CALC. $\frac{R_I - R_C}{\Delta T_a}$
STORE TDEQCT+6,1
TEST A_I, LT 90
YES
NO
R1TET
TEST A_C GT 270
YES
NO
R1MTI
COMP. $\frac{\partial A}{\partial B} \frac{A_I + 2\pi - A_C}{\Delta P}$
TEST A_I GT 270
NO
YES
R1TES
TEST A_C LT 90
NO
YES
R1OK
COMP. $\frac{\partial A}{\partial P} \frac{A_I - A_C}{\Delta P}$
COMP. $\frac{\partial A}{\partial P} \frac{A_I - (A_C + 2\pi)}{\Delta P}$
R1NEX
CALC. $\frac{E_I - E_C}{\Delta T_a}$
STORE IN TDEQCT+18,1
TEST IR1=1
NO
YES
SET IR1= IR1-1
EXIT RETURN 2,4
EXIT R1STR

60. R2AINT - NUMERICAL INTEGRATION INPUT TABLES TO BE USED FOR GENERATION OF PERTURBED TABLES FOR NUMERICAL PARTIAL DERIVATIVES

R2AINT SETS UP THE INPUT TABLES TO BE USED BY NOCPNI IN THE GENERATION OF INTEGRATION TABLES TO BE USED BY RINDCP IN THE CALCULATION OF PARTIAL DERIVATIVES BY NUMERICAL MEANS. IT IS A SUBROUTINE OF DODIFC, THE DIFFERENTIAL CORRECTION PROCESSOR. (SEE FLOWCHART)

INPUT

A) TABLES -

TRVTAP +1 X_a, +2 Y_a, +3 Z_a } $\bar{r}_a$ THE POSITION VECTOR AT ANCHOR TIME, t_a IN FLOATING POINT G.L.U.

+4 $\dot{X}_a$, +5 $\dot{Y}_a$, +6 $\dot{Z}_a$ } $\bar{v}_a$ THE VELOCITY VECTOR AT ANCHOR TIME, t_a IN FLOATING POINT G.V.U.

TNEARM t_a - ANCHOR TIME IN FIXED SECONDS

TNUMNI - INDICATOR USED TO TELL WHICH PERTURBED TABLE IS BEING WORKED ON. SET TO ZERO ON FIRST ENTRY

KRDELT h - INTEGRATION STEP SIZE TO BE USED - FIXED SECONDS

KPERST p - NUMBER OF INTEGRATION STEPS PER OUTPUT - FIXED POINT

TNETPT t_e - ENDING TIME OF PERTURBED TABLES

TPA.RX THRU TPA.RX+5 - PERTURBATIONS TO BE APPLIED TO

THE INITIAL CONDITIONS IN FLOATING POINT GEMINI UNITS

TMRCNP - A FOUR LOCATION TABLE SIGNIFYING THE MANNER IN WHICH THE D.C. REFERENCE TRAJECTORY WAS GENERATED. FORMAT IS AS FOLLOWS -

TMRCNP	mode	1	3	4	5
+1	1st parameter	λ_T	GMTRB	GMTIR	Zero
+2	2nd parameter	ϕ_T	Bank Angle	Bank Angle	Bank Angle
+3	3rd parameter	t_{325K}	t_{325K}	t_{325K}	t_{325K}

TABLE TMRCNP IS INTERPRETED AS FOLLOWS -

MODE 1 - ON BOARD GUIDANCE METHOD

MODE 3 - BANK ANGLE REVERSAL METHOD

MODE 4 - ROLL INITIATION AT SPECIFIED GMT METHOD

MODE 5 - CONTINGENCY METHOD (FIXED BANK ANGLE)

BANK ANGLE ARE IN DEGREES, ϕ_T AND λ_T IN RADIANS AND TIMES ARE IN GMT SECONDS.

MODE IS FIXED POINT B35, ALL OTHER QUANTITIES ARE IN FLOATING POINT

B) CONSTANTS

```
KMC000  DEC     -0
KOC000  DEC     0
K00001  DEC     1
K00035  DEC     35
K00050  DEC     50
K00054  DEC     54
```

KOP1D1	PCN	0,0,1	MASK WITH 1 IN PREFIX AND DECREMENT
MCRWGT			REENTRY WEIGHT IN FLOATING POINT POUNDS

OUTPUT

A) TABLES -

TNNDCI A SEVENTEEN LOCATION TABLE TO BE USED AS INPUT BY THE INTEGRATION PROGRAM NOCPNI IN THE FOLLOWING FORMAT -

TNNDCI	X_a	$\bar{r}_a$	+ DESIRED PERTURBATION
+1	Y_a		
+2	Z_a		
+3	$\dot{X}_a$	v_a	+ DESIRED PERTURBATION
+4	$\dot{Y}_a$		
+5	$\dot{Z}_a$		
+6	t_a		ANCHOR TIME (FIXED)
+7	h		STEP SIZE (FIXED)
+8	0		LENGTH OF TIME TO INTEGRATE BACKWARDS
+9	Δt_f		LENGTH OF TIME TO INTEGRATE FORWORDS (FIXED)
+10	TL, p		OUTPUT TABLE,,NUMBER INTERVALS PER OUTPUT
+11	ONE IN PREFIX AND DECREMENT		SECONDS, NO THRUST, STORE, RUNGE KUTTA
+12	W		REENTRY WEIGHT OF SPACECRAFT (FLOATING POINT)

+13 t_{RB} OR λ_T — TIME TO REVERSE BANK OR LONGITUDE OF TARGET IMPACT POINT (FLOATING POINT)

+14 t_{IR} OR ϕ_T — TIME TO INITIATE ROLL OR GEODETIC LATITUDE OF THE TARGET IMPACT POINT (FLOATING POINT)

+15 BANK ANGLE — BANK ANGLE (FLOATING POINT DEGREES)

+16 NOT USED

TNUMNI — UPDATED INDICATOR FOR TELLING WHICH PERTURBED TABLE IS BEING WORKED ON.

TN4NEX — INDICATOR WHICH TELLS NOCPNI TO IGNORE 10,COO FEET TEST (SET TO - 0)

B) COMMUNICATION CELL -

MCNDCQ - STATING THE TYPE OF NDC TABLE TO BE GENERATED

C) ACCUMULATOR -

UPON EXIT THE ACCUMULATOR IS SET TO -0 WITH 1 IN THE P BIT.

METHOD

UPON ENTRY R2AINT SAVES INDEX REGISTERS AND MOVES THE ANCHOR POSITION AND VELOCITY VECTORS TO THE OUTPUT TABLE. BASED ON THE SETTING OF TNUMNI IT PICKS UP THE INDICATED PERTURBATION FROM THE TPA.RX TABLE AND APPLIES IT TO THE INDICATED ANCHOR COMPONENT.

TNUMNI SETTING UPON ENTRY	INPUT VECTOR TO NOCPNI
0	$X_a + \Delta X, Y_a, Z_a, \dot{X}_a, \dot{Y}_a, \dot{Z}_a$
1	$X_a, Y_a + \Delta Y, Z_a, \dot{X}_a, \dot{Y}_a, \dot{Z}_a$
2	$X_a, Y_a, Z_a + \Delta Z, \dot{X}_a, \dot{Y}_a, \dot{Z}_a$
3	$X_a, Y_a, Z_a, \dot{X}_a + \Delta \dot{X}, \dot{Y}_a, \dot{Z}_a$
4	$X_a, Y_a, Z_a, \dot{X}_a, \dot{Y}_a + \Delta \dot{Y}, \dot{Z}_a$
5	$X_a, Y_a, Z_a, \dot{X}_a, \dot{Y}_a, \dot{Z}_a + \Delta \dot{Z}$

THE ANCHOR TIME AND STEP SIZE ARE THEN MOVED TO THE OUTPUT TABLE. THE TIME FOR BACKWARD INTEGRATION IS SET TO ZERO.

THE TIME FOR FORWARD INTEGRATION IS SET TO THE LESSER OF $t_e - t_a$ AND 35 HP. THIS TEST IS MADE TO INSURE THAT THERE IS NO POSSIBILITY OF THE PERTURBED TABLES OVERRUNNING THEIR ALLOCATED CORE STORAGE.

THE INTEGRATION OUTPUT TABLE ADDRESS IS THEN COMPUTED BASED ON THE TNUMNI SETTING AND STORED IN THE OUTPUT TABLE. THE NUMBER OF INTEGRATION STEPS PER OUTPUT IS ALSO STORED. INDICATIONS THAT INPUT TIMES ARE IN SECONDS, RUNGE KUTTA INTEGRATION IS TO BE USED WITHOUT THRUST AND THAT AN INTEGRATION OUTPUT TABLE IS DESIRED ARE STORED. THE REENTRY WEIGHT IS ALSO STORED IN THE OUTPUT TABLE.

A NEGATIVE ZERO IS PLACED IN TN4NEX TO SIGNAL THE INTEGRATOR THAT IT IS TO CONTINUE THE TABLE GENERATION UNTIL THE END OF THE TIME SPECIFIED FOR FORWARD INTEGRATION REGARDLESS OF THE ALTITUDE.

IF THE REENTRY MODE IS ONBOARD GUIDANCE, A FIXED POINT 54 IS PLACED IN THE DECREMENT OF MCNDCQ, OTHERWISE A 50 IS PLACED THERE.

IF THE REENTRY MODE IS ONBOARD GUIDANCE, THE GEODETIC LATITUDE AND LONGITUDE ARE PLACED IN THE OUTPUT TABLE. IF THE MODE IS BANK ANGLE REVERSAL, THE BANK ANGLE AND BANK ANGLE REVERSAL TIME ARE PLACED IN THE OUTPUT TABLE AND THE TIME TO INITIATE ROLL IS SET TO -0. IF THE MODE IS ROLL INITIATION AT A SPECIFIED GMT, THE BANK ANGLE TO BE FLOWN AND THE TIME TO INITIATE ROLL ARE PLACED IN THE OUTPUT TABLE AND THE TIME TO REVERSE BANK IS SET TO -0.

THE CONTENTS OF TNUMNI ARE INCREASED BY 1 IN PREPARATION FOR THE NEXT ENTRY TO R2AINT. A NEGATIVE ZERO AND A ONE IN THE P BIT ARE THEN PLACED IN THE ACCUMULATOR.

INDEX REGISTERS ARE RESTORED AND THE PROGRAM EXITS TO MFDIFC, THE SUFFIX OF THE DIFFERENTIAL CORRECTION PROCESSOR.

USAGE

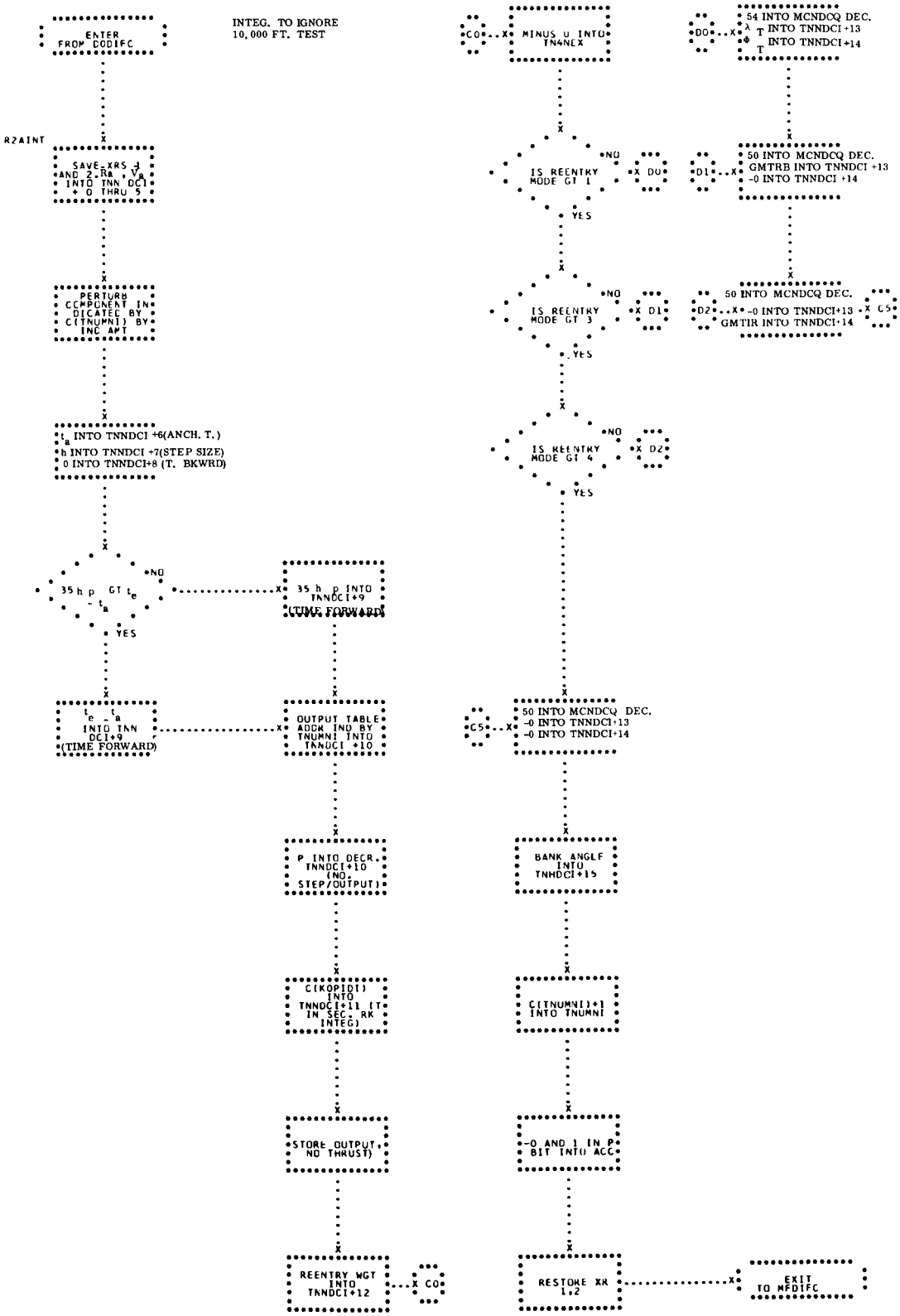
DODIFC REPETITIVELY ENTERS R2AINT BY A TRANSFER TO LOCATION R2AINT AFTER HAVING ESTABLISHED THE NECESSARY INPUT CONDITIONS. (IT SETS TNUMNI TO 0 WHEN THE CYCLE OF SIX ENTRIES FOR GENERATION OF THE SIX PERTURBED TABLES IS TO BE STARTED.)

THERE ARE NO ERROR RETURNS.

R2AINT TRANSFERS CONTROL TO MFDIFC UPON COMPLETION OF ITS TASK.

A) STORAGE REQUIRED - 82 LOCATIONS

B) TIMING - 0.23 MILLISECONDS


ENTER FROM DODIFC
INTEG. TO IGNORE 10,000 FT. TEST
R2AINT
SAVE XRS J AND Z, Ra, Va INTO TNN DCI + 0 THRU 5
PERTURB COMPONENT INDICATED BY C(TNUMNI) BY INC AMT
ta INTO TNNDCI +6(ANCH. T.)
h INTO TNNDCI +7(STEP SIZE)
0 INTO TNNDCI+8 (T. BKWRD)
35 h p GT te - ta
NO
YES
35 h p INTO TNNDCI+9 (TIME FORWARD)
te - ta INTO TNN DCI+9 (TIME FORWARD)
OUTPUT TABLE ADDR IND BY TNUMNI INTO TNNDCI +10
P INTO DECR. TNNDCI+10 (NO. STEP/OUTPUT)
C(KOPIDI) INTO TNNDCI+11 (T IN SEC. RK INTEG)
STORE OUTPUT, NO THRUST)
REENTRY WGT INTO TNNDCI+12
C0
MINUS U INTO TN4NEX
IS REENTRY MODE GT 1
D0
IS REENTRY MODE GT 3
D1
IS REENTRY MODE GT 4
D2
50 INTO MCNDCQ DEC. -0 INTO TNNDCI+13 -0 INTO TNNDCI+14
C5
BANK ANGLE INTO TNNDCI+15
C(TNUMNI)+1 INTO TNUMNI
-0 AND 1 IN P BIT INTO ACC
RESTORE XR 1,2
EXIT TO MFDIFC
54 INTO MCNDCQ DEC. λT INTO TNNDCI +13 ΦT INTO TNNDCI +14
50 INTO MCNDCQ DEC. GMTRB INTO TNNDCI +13 -0 INTO TNNDCI +14
50 INTO MCNDCQ DEC. -0 INTO TNNDCI+13 GMTIR INTO TNNDCI+14

61. R5RARF - TIME TO FIRE PROGRAM

R5RARF IS A SUBROUTINE USED BY MYTTF4 AND MYREN2 IN ORBIT AND REENTRY, RESPECTIVELY. IN THE ORBIT PHASE R5RARF COMPUTES THE TIME TO FIRE THE RETROROCKETS AND THE TIME TO REVERSE THE BANK ANGLE WITH A FIXED BANK ANGLE MAGNITUDE IN ORDER TO HIT A GIVEN TARGET WITHIN A PRESCRIBED TOLERANCE. IF TMDELT, THE DURATION OF OAMS BURN, IS ZERO R5RARF SOLVES FOR A TIME TO FIRE THE RETRO-ROCKETS. IF TMDELT IS NONZERO R5RARF SOLVES FOR A TIME TO FIRE THE OAMS ROCKETS IN ORDER TO HIT A 45-NM PERIGEE, INSURING A DECAY SAFE ORBIT. THE DURATION OF THE OAMS BURN IS SPECIFIED BY TMDELT WHICH IS COMPUTED BY MYTTF4. THEN THERE IS A FREE FLIGHT INTEGRATION AFTER WHICH RETROFIRE OCCURS AND LASTS FOR 22 SECONDS, FOLLOWED BY AN INTEGRATION TO IMPACT AT 10,000 FEET. THE DURATION OF THE FREE FLIGHT INTEGRATION IS 720 SECONDS MINUS THE CONTENTS OF TMDELT, THE DURATION OF THE OAMS BURN.

IN THE REENTRY PHASE RETROFIRE HAS OCCURRED AND EITHER A BURNOUT VECTOR OR A DIFFERENTIAL CORRECTION VECTOR IS INPUTTED TO R5RARF ALONG WITH THE LATITUDE AND LONGITUDE OF THE TARGET. R5RARF THEN COMPUTES A BANK ANGLE AND A TIME TO INITIATE ROLL METHOD OF HITTING THE TARGET AND BANK ANGLE AND TIME TO REVERSE BANK ANGLE METHOD NECESSARY TO HIT THE TARGET. (SEE FLOWCHART.)

INPUT

A) TABLES -

TMRARF

TMBLST

TMR5FF

TMR5OM

TNREEP

B) PROCESSOR -

NOCPNI

C) LIBRARY SUBROUTINES -

U3DOT	U3XPR
COS	VMAG
SIN	SQRT
ARCOS	ATNAB
ARSIN	

D) SYSTEM MACRO -

QTSX

E) CONSTANTS -

K00240	K002.0
KCH233	K003.8
K0060.	K030.0
K.1RAD	KRTNM.
K00INF	KCFTSC
K000.4	KCGUNM
K00.36	K00T1.
K060.0	K00C1.
K00.PI	K00D1.

F) COMMUNICATION CELLS -

MCZRWX

MCTYPE

MNCPNI

MCPWGT

G) CONDITIONS ON ENTRY -

AC-R5LNG	LONGITUDE OF TARGET - FL.PT RADIANS
MQ-R5LAT	LATITUDE OF TARGET - FL.PT. RADIANS
TMETTY	ESTIMATED TIME TO FIRE - FL.PT. SEC.
KBNKLE	FIXED BANK ANGLE - FL.PT. DEGREES

TMDELT DURATION OF OAMS BURN - FL.PT. SEC.

OUTPUT

A) TABLES -

TMBLST

TMRARF

TMR5FF

TMR5OM

TVVSQA

TMRORF

TNREEP

LOCATION	NAME	UNITS
TMRORF	TIME TO FIRE	FL. PT. SEC.
TMRORF+1	TIME TO REVERSE BANK	FL. PT. SEC.
+2	BANK ANGLE FOR TIME TO BANK BANK MOVE	FL. PT. SEC.
+3	TIME TO INITIATE ROLL	FL. PT. SEC.
+4	BANK ANGLE FOR B-TIR MODE	FL. PT. SEC.
+16	LATITUDE OF IMPACT	FL. PT. RADIANS
+17	LONGITUDE OF IMPACT	FL. PT. RADIANS
+18	LATITUDE ERROR	FL. PT. N. M.
+19	LONGITUDE ERROR	FL. PT. N. M.

B) COMMUNICATION CELLS -

NONE

C) PROGRAM PARAMETERS -

TNRF

TIMRB

BNKLE

GMTIR

D) CONDITIONS ON EXIT -

NORMAL RETURN - TMRORF TABLE CONTAINS THE STORED

OUTPUT PARAMETERS

ERROR RETURNS - AC CONTAINS A FIXED POINT ERROR NUMBER

WHICH MEANS THE FOLLOWING -

ERROR NUMBER	REASON
1	ZL IP WAS NOT GENERATED
2	60 BANK ANGLE WAS NOT GENERATED IN ELLIPTICAL BOUNDARY COMPUTATION.
3	GMTRB NOT GENERATED IN PARTIALS.
4	GMTIR NOT GENERATED IN PARTIALS.
5	NEITHER CONVERGE IN PARTIALS.
6	TTF, TIMRB ERROR IN PARTIALS.
7	ERROR - LONGITUDE ITERATION EXCEEDED MAXIMUM NUMBER.
8	ERROR IN TTF OPPOSITE BANK ANGLE.
9	ERROR IN TTF BOUNDARY COMPUTATION COMMON FUNCTION SUBROUTINE.
10	ERROR IN INTEGRATION TO D ALTITUDE SUBROUTINE.
31	ERROR IN INTEGRATING WITH THRUST IN LONGITUDE COMPUTATION.
12	ERROR IN ELLIPTICAL BOUNDARY COMPUTATION.
13	ERROR IN 1ST GUESS IMPACT FOR TIR, B MODE.
14	ERROR IN TIRN IMPACT GENERATION.
15	ERROR IN INTEGRATING TO BOUNDARY IMPACT FOR TARGET OUTSIDE IN CROSS RANGE.
16	TARGET OUTSIDE F.P. IN DOWN RANGE.
17	ERROR IN ELEMENTARY FUNCTION IN PROJ - R5ZML.
18	ERROR IN ELEMENTARY FUNCTION IN G60IM FOR CRMAX COMPUTATION.
19	ERROR IN ELEMENTARY FUNCTION IN TIR 1ST. GUESS.
20	ERROR IN ELEMENTARY FUNCTION IN TIRN ITERATION.
21	ERROR IN ELEMENTARY FUNCTION IN DTIR COMP IN

PARTIALS.

22 TARGET OUTSIDE FOOTPRINT IN CROSS RANGE, WE INTEGRATED TO HIT BOUNDARY HOWEVER.

23 TARGET LEFT OF FOOTPRINT IN DOWN RANGE.

24 TARGET OUTSIDE F.P. IN TTF MODE.

25 ERROR IN B, TIMRB INTEGRATION IF PARTIALS BLEW UP.

26 ERROR IN INRVB INTEGRATION (TIMRB 1).

27 MLIP WAS NOT GENERATED.

30 ERROR IN B, GMTIR INTEGRATION IF PARTIALS BLEW UP.

11 ERROR IN INTERPOLATION IN R5IWT IN LONGITUDE.

32 ERROR IN INTEGRATING TO IMPACT IN LONGITUDE ITERATION.

33 ERROR IN INTERPOLATING IN R5IWT AFTER ROUNDING.

34 ERROR IN INTEGRATING WITH THRUST IN R5IWT AFTER ROUNDING.

35 ERROR IN INTEGRATING TO IMPACT AFTER ROUNDING.

METHOD

THE METHOD IS BASED ON MR. J. RUTLAND'S PAPER, BACKUP REENTRY GUIDANCE AND CONTROL PROCEDURE FOR GEMINI, PUBLISHED BY NASA MANNED SPACE CRAFT CENTER, HOUSTON. THREE DIFFERENT MODES ARE COMPUTED.

1. TIME TO FIRE AND TIME TO REVERSE BANK WITH FIXED BANK ANGLE. (COMPUTED IN ORBIT)
2. TIME TO REVERSE BANK ANGLE AND BANK ANGLE WITH TIME TO FIRE FIXED. (COMPUTED IN REENTRY MODE)
3. TIME TO INITIATE ROLL AND BANK ANGLE WITH TIME TO FIRE FIXED. (COMPUTED IN REENTRY MODE.)

ALL THREE SETS OF PARAMETERS ARE COMPUTED USING SENSITIVITY COEFFICIENTS, I.E., DIFFERENTIALS RESULTING FROM A

PARTIAL TAYLOR'S EXPANSION AS FOLLOWS -

$$\phi_T = \phi_o + \frac{\partial \phi}{\partial tF} \Delta t_F + \frac{\partial \phi}{\partial t_{RB}} \Delta t_{RB}$$

$$\lambda_T = \lambda_o + \frac{\partial \lambda}{\partial t_F} \Delta t_F + \frac{\partial \lambda}{\partial t_{RB}} \Delta t_{RB}$$

$$d\phi_o = \frac{\partial \phi}{\partial t_F} \Delta t_F + \frac{\partial \phi}{\partial t_{RB}} \Delta t_{RB}$$

$$d\lambda_o = \frac{\partial \lambda}{\partial t_F} \Delta t_F + \frac{\partial \lambda}{\partial t_{RB}} \Delta t_{RB}$$

WHERE

ϕ_T = LATITUDE OF TARGET
λ_T = LONGITUDE OF TARGET
ϕ_0 = LATITUDE OF THE ''FIRST GUESS IMPACT''
λ_0 = LONGITUDE OF THE ''FIRST GUESS IMPACT'' POINT
Δt_F = UNKNOWN CORRECTION TO TIME TO FIRE FIRST GUESS
Δt_{RB} = UNKNWON CORRECTION TO TIME TO REVERSE BANK FIRST GUESS.

ϕ_0, λ_0 IS THE IP COMPUTED USING t_{FO} AND t_{RBO} WHICH MUST BE REASONABLY CLOSE TO THE FINAL SOLUTION IN ORDER TO INSURE CONVERGENCE. EQUATIONS 1 AND 2 ARE SOLVED FOR Δt_F AND Δt_{RB}. THE PARTIAL DERIVATIVES ARE SOLVED FOR BY PERTURBING TIME TO FIRE AND THEN BY PERTURBING TIME TO REVERSE BANK ANGLE. IN THE TIME TO REVERSE BANK ANGLE AND BANK ANGLE MODE, Δt_F IS REPLACED BY Δt_{RB} AND Δt_{RB} IS REPLACED BY ΔB.

IN THE TIME TO INITIATE ROLL AND BANK ANGLE MODE Δt_F IS REPLACED BY Δt_{IR} AND Δt_{RB} IS REPLACED BY ΔB.

A. TIME TO FIRE MODE -

IN THE TIME TO FIRE MODE, THE TIME TO FIRE MEANS THE TIME TO FIRE RETROS IF TMDELT IS ZERO AND THE TIME TO

FIRE OAMS ROCKETS IF TMDELT IS NON-ZERO. THE STEPS ARE

CONVERGE ON THE LONGITUDE OF THE TARGET WITHIN A PRESCRIBED TOLERANCE USING THE EQUATION

(3)

$$t_1^{RF} = t_0^{RF} - \frac{(\lambda_T - \lambda_o)}{(\omega_c - \omega_E)}$$

WHERE -

ω_c = ω capsule

ω_E = ω earth

WHERE -

λ_o IS OBTAINED BY INTEGRATING THROUGH RETROFIRE AND TO IMPACT USING THE ESTIMATED TIME TO FIRE t_o, WHICH IS AN INPUT TO R5RARF. THEN THE FOLLOWING ITERATIVE EQUATION IS USED UNTIL THE CONVERGENCE CRITERION IS MET. THERE IS A MAXIMUM LOOP COUNT OF 15.

(4)

$$t_{N+1}^{RF} = t_N^{RF} + \frac{(\lambda_T - \lambda_N)}{\lambda_N - \lambda_{N-1)}} (t_N^{RF} - t_{N-1}^{RF})$$

THE CONVERGED VALUE IS DENOTED AS t_o. NEXT, THE TARGET LOCATION IS CHECKED TO SEE IF IT IS IN OR OUT OF THE FOOTPRINT USING NINER'S METHOD.

IF THE TARGET IS IN THE FOOTPRINT A FIRST GUESS FOR TIME TO REVERSE BANK ANGLE IS OBTAINED USING A DENSITY ALTITUDE PARAMETER METHOD. THE FOLLOWING EQUATIONS ARE USED.

B = FIXED BANK ANGLE. $\phi \leq |B| \leq |90|$

C = CROSS RANGE DISTANCE FROM TARGET TO MIDDLE LINE OF FOOTPRINT.

W = MAXIMUM WIDTH OF FOOTPRINT.

$D = f(\frac{C}{W}, B)$ OBTAINED BY A TABLE LOOKUP (5)

WE NOTE THAT $D \equiv LN(V^2/a)$
WHERE

V IS THE RELATIVE VELOCITY AND a IS THE RELATIVE ACCELERATION

LN (V^2/a) IS INTEGRATED DOWN UNTIL IT MATCHES THE D VALUE FROM THE TABLE LOOKUP. THE CORRESPONDING TIME IS SET EQUAL TO tRB_0 OR THE FIRST GUESS FOR TIME TO REVERSE BANK ANGLE. USING t_0 AND tRB_0 WE INTEGRATE DOWN TO IMPACT. THE IMPACT POINT IS DENOTED AS λ_0 AND ϕ_0.

USING EQUATIONS 1 AND 2 THE DIFFERENTIALS Δt_F AND Δt_{RB} ARE SOLVED. INTEGRATION TO IMPACT MUST BE PERFORMED TWICE IN ORDER TO COMPUTE THE REQUIRED PARTIAL DERIVATIVES.

THE DELTAS COMPUTED IN STEP 5 ARE ADDED TO THE FIRST GUESS FOR TIME TO FIRE t_{FO} AND THE FIRST GUESS FOR TIME TO REVERSE BANK t_{RBO}. INTEGRATION TO IMPACT IS PERFORMED WITH THESE VALUES. IF THE IMPACT POINT IS WITHIN ONE MILE OF THE TARGET CONVERGENCE IS OBTAINED. IF CONVERGENCE HAS NOT BEEN OBTAINED, THE LAST VALUES COMPUTED AS A FIRST GUESS ARE USED AND REPEAT STEPS 5 AND 6 UNTIL WE DO CONVERGE. THERE IS A MAXIMUM LOOP COUNT OF 4.

AFTER CONVERGENCE THE OUTPUT PARAMETERS ARE STORED IN THE TMRORF TABLE AND THE NORMAL EXIT FROM R5RARF IS TAKEN.

B. TIME TO ROLL, BANK ANGLE MODE -

THIS METHOD OF NAVIGATION IS USED AFTER RETRO FIRE HAS OCCURRED. THE INPUT TO THE PROGRAM IS THE TARGET AND THE BURNOUT VECTOR OR A VECTOR FROM DIFFERENTIAL CORRECTION. THE STEPS INVOLVED IN THIS COMPUTATION ARE -

THE ZERO LIFT, MAX LIFT AND MINUS 60-DEGREE TRAJECTORIES ARE COMPUTED IN ORDER TO ESTIMATE AN APPROX-

IMATE FOOTPRINT.

A NORMALIZED FOOTPRINT IS COMPUTED BASED ON THE EQUATION OF AN ELLIPSE.

THE MAXIMUM BOUNDARIES OF THE ELLIPSE ARE -

$$YM \equiv CRMAX \sqrt{1 - \frac{(X - .4)^2}{.36}} \qquad (6)$$

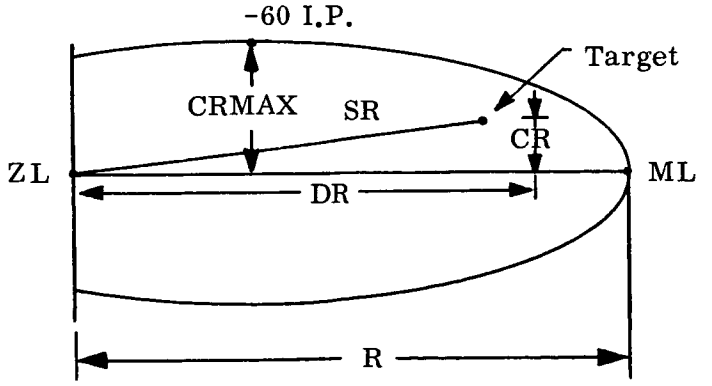
WHERE

X ≡ NORMALIZED DOWN RANGE OF THE TARGET

CRMAX ≡ MAXIMUM WIDTH OF FOOTPRINT (NORMALIZED)

YM ≡ NORMALIZED MAXIMUM CROSS RANGE AS A FUNCTION OF DOWN RANGE

THE TARGET PARAMETERS WITH RESPECT TO THE FOOTPRINT ARE COMPUTED. THEY ARE SLANT RANGE FROM THE ZERO LIFT I.P., DOWN RANGE AND CROSS RANGE AS INDICATED BELOW.



DR ≡ DOWN RANGE IN RADIANS.

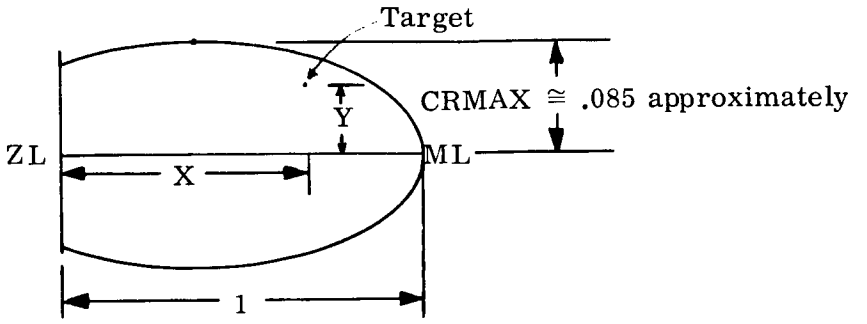
CR ≡ CROSS RANGE IN RADIANS.

R ≡ MAX. DOWN RANGE IN RADIANS.

SR ≡ SLANT RANGE IN RADIANS.

THE NORMALIZED FOOTPRINT IS OBTAINED BY DIVIDING THE PARAMETERS BY THE MAXIMUM DOWN RANGE. THE NORMALIZED

FOOTPRINT IS SHOWN BELOW.



$Y = \frac{CR}{R} \equiv$ NORMALIZED CROSS RANGE

$X = \frac{DR}{R} \equiv$ NORMALIZED DOWN RANGE.

A FIRST GUESS FOR BANK ANGLE IS COMPUTED

$$B_1 = f(x, y)$$

THIS IS A TABLE LOOK-UP FROM RUTLAND'S PAPER.

A FIRST GUESS FOR TIME TO INITIATE ROLL IS COMPUTED BASED ON CURVE FITS AND EQUATIONS BY NINER AND HOOVER.

$$X = (90 - B_1)/30 \qquad (7)$$

$Y_D = (((D_0 X + D_1) X + D_2) X) \equiv$ MAXIMUM NORMALIZED D.R.

$D_0 = -.0483$

$D_1 = .175$

$D_2 = .2433$

$$Y_C = (((C_0 X + C_1) X + C_2) X + C_3) \equiv \text{MAXIMUM CROSS RANGE NORMALIZED FOR } B_1 \qquad (8)$$

$C_0 = -.0037$

$C_1 = -.0035$

$C_2 = .02177$

$C_3 = .067$

$$R = (SR)/\sqrt{Y_C^2 + Y_D^2} \qquad (9)$$

$$Y_T = (((T_0 X+T_1)X+T_2)X+T_3) \quad (10)$$

$T_0 = 2.046$

$T_1 = -3.09$

$T_2 = 1.884$

$T_3 = .0584$

$RTM = f(THETR, B_1)$ REENTRY TIME (11)

THETR = LENGTH OF FOOTPRINT IN RADIANS

DTIR = (RTM) (Y_T) $\equiv$ TIME TO INITIATE ROLL REFERENCED FROM BEGINNING OF BLACKOUT. (12)

TIR = TBO + DTIR (13)

USING THIS VALUE FOR T_{NR} AND B_1 FOR BANK ANGLE INTEGRATION IS PERFORMED TO IMPACT.

THEN THE FOLLOWING ITERATIVE EQUATION IS USED TO CLOSE UP THE SLANT RANGE.

$$DTIR = (R_T - R_N) \frac{(TIR_N - TIR_{N-1})}{(R_N - R_{N-1})} \quad (14)$$

$TIR_N = TIR_N + DTIR$

WHERE

$R_T \equiv$ SLANT RANGE TO THE TARGET

$R_N \equiv$ SLANT RANGE TO NTH. I.P.

$R_{N-1} \equiv$ SLANT RANGE TO N-1ST. I.P.

$TIR_N \equiv$ TIME TO INITIATE ROLL FOR NTH ITERATION

$TIR_{N-1} \equiv$ TIME TO INITIATE ROLL FOR (N-1)ST. ITERATION

AFTER THE SLANT RANGE ERROR REACHES THE REQUIRED VALUE THE FOLLOWING QUANTITIES SET -

$$TIRO = TIR_N \quad (15)$$

$B_o = B_1$

$d\phi_o = \phi_T - \phi_o$

$d\lambda_o = \lambda_T - \lambda_o$

THESE VALUES ARE USED AS A FIRST GUESS FOR EQUATIONS 1 AND 2 I.E., THE DIFFERENTIAL OR PARTIAL DERIVATIVE ITERATIVE COMPUTATIONS USED TO REFINE THE VALUES FOR THE GUIDANCE PARAMETERS IN ORDER TO MEET THE REQUIRED CONVERGENCE CRITERION. THIS METHOD IS EXPLAINED IN A, TIME TO FIRE MODE. NOTE THE CORRESPONDING CHANGE IN VARIABLES

(16)

$$d\phi_o = \frac{\partial \phi}{\partial t_{IR}} \Delta t_{IR} + \frac{\partial \phi}{\partial B} \Delta B$$

$$d\lambda_o = \frac{\partial \lambda}{\partial t_{IR}} \Delta t_{RB} + \frac{\partial \lambda}{\partial B} \Delta B$$

THE NEXT STEP IS THE TIME TO REVERSE BANK ANGLE AND BANK ANGLE MODE OF COMPUTATION.

C. TIME TO REVERSE BANK, BANK ANGLE MODE -

THIS MODE OF NAVIGATION IS USED AFTER RETROFIRE HAS OCCURRED. THE INPUT AGAIN IS THE TARGET AND THE BURNOUT VECTOR OR A VECTOR FROM DIFFERENTIAL CORRECTIONS. THE PARAMETERS COMPUTED PERTAINING TO THE TARGET POSITION IN THE NORMALIZED FOOTPRINT ARE ALSO USED.

THE FOLLOWING STEPS AND EQUATIONS ARE USED.

$Y_B = (((B_1 3X+B_1)3X+B_2)3X+B_3) \equiv$ 1ST GUESS FOR BANK ANGLE (17)

$B_0 = 3$

$B_1 = -10$

$B_2 = 33$

$B_3 = -90$

$B_1 = f(x)$ - FROM RUTLAND'S PAPER

$D = f(x, y) \equiv LN(V^2/a)$ AS IN (5) (18)

AN INTEGRATION IS PERFORMED UNTIL $LN(V^2/a)$ MATCHES THE D VALUE FROM THE TABLE LOOK UP. THE CORRESPONDING TIME IS SET EQUAL TO TRB_o I.E., THE FIRST GUESS FOR TIME

TO REVERSE BANK ANGLE.

USING TRB_o AND $B_o = B_1$ INTEGRATION IS PERFORMED TO IMPACT. THE IMPACT POINT IS DENOTED AS λ AND ϕ

THESE VALUES ARE USED AS A FIRST GUESS FOR EQUATION 1 AND 2 I.E., THE DIFFERENTIAL OR PARTIAL DERIVATIVE ITERATIVE COMPUTATION USED TO REFINE THE VALUES FOR THE GUIDANCE PARAMETERS IN ORDER TO MEET THE REQUIRED CONVERGENCE CRITERION. THIS METHOD IS EXPLAINED IN A, TIME TO FIRE MODE. NOTE THE CORRESPONDING CHANGE IN VARIABLES.

(19)

$$d\phi_o = \frac{\partial \phi}{\partial t_{RB}} \Delta t_{RB} + \frac{\partial \phi}{\partial B} \Delta B$$

$$d\lambda_o = \frac{\partial \lambda}{\partial t_{RB}} \Delta t_{RB} + \frac{\partial \lambda}{\partial B} \Delta B$$

D. IF THE TARGET IS FOUND NOT TO BE INSIDE THE FOOTPRINT OR IF WE DO NOT ARRIVE AT A CONVERGED SOLUTION A COMPUTATION IS PERFORMED TO HIT THE BOUNDARY AT THE POINT WHICH IS THE CLOSEST DISTANCE FROM THE TARGET TO THE BOUNDARY. THE FOLLOWING EQUATIONS ARE USED EMPLOYIN NEWTON'S BOUNDARY VALUE METHOD OF SOLVING SIMULTANEOUS NON LINEAR EQUATIONS.

(20)

$$g = y^2 - b^2 \left(1 - \frac{(x-e)^2}{a^2}\right)$$

$$f = (x-c)^2 + (y-d)^2$$

IN ORDER TO MINIMIZE F WE USE THE LAGRANGE MULTIPLIER METHOD -

(21)

$$\frac{\partial f}{\partial x} + \lambda \frac{\partial g}{\partial x} = 0$$

$$\frac{\partial f}{\partial y} + \lambda \frac{\partial g}{\partial y} = 0$$

$$\frac{\partial f}{\partial x} = 2(x - c), \frac{\partial f}{\partial y} = 2(y - d)$$

$$\frac{\partial g}{\partial x} = \frac{2b^2}{a^2}(x - e), \frac{\partial g}{\partial y} = 2y$$

$$2(x - c) + \lambda \frac{2b^2}{a^2}(x - e) = 0$$

$$2(y - d) + \lambda 2y = 0$$

$$\lambda = \frac{2(d - y)}{2y} = \frac{2(c - x)}{\frac{2b^2}{a^2}(x - e)}$$

$$\frac{b^2}{a^2}(d - y)(x - e) - y(c - x) \equiv F \qquad (22)$$

$$y^2 - b^2\left(1 - \frac{(x - e)^2}{a^2}\right) \equiv G$$

EQUATIONS 22 ARE THEN SOLVED BY THE STANDARD RELATIONSHIPS OF NEWTON.

(23)

$$X_{i+1} = X_i - [F_i \frac{\partial G_j}{\partial Y} - G_i \frac{\partial F}{\partial Y_i}]/D$$

$$Y_{i+1} = Y_i - [G_i \frac{\partial F}{\partial X_i} - F_i \frac{\partial G}{\partial X_i}]/D$$

$$D = \frac{\partial F}{\partial X^i}\frac{\partial G}{\partial Y^i} - \frac{\partial F}{\partial Y^i}\frac{\partial G}{\partial X^i}$$

THE ABOVE ITERATIVE EQUATIONS GIVE THE APPROXIMATE FIRST GUESS FOR THE NORMALIZED BOUNDARY COORDINATES WHICH IN TURN GIVE THE FIRST GUESS FOR BANK ANGLE TO HIT THIS POINT ON THE EDGE OF THE FOOTPRINT.

USING THIS VALUE FOR BANK ANGLE INTEGRATION IS PERFORMED TO IMPACT. THE RADIAL ERROR FROM THE TARGET TO THE I.P. AND THE ASSOCIATED GUIDANCE PARAMETERS ARE STORED IN A TABLE. THESE SAME PARAMETERS FROM ALL PREVIOUS INTEGRATIONS HAVE BEEN STORED IN SEQUENCE.

A MINIMUM LOOK-UP IS NOW PERFORMED ON THE TABLE OF RADIAL ERRORS. THE MINIMUM VALUE AND ITS ASSOCIATED PARAMETERS ARE THEN CHOSEN AS THE POINT OF NEAREST APPROACH.

A PERTURBATION IN BANK ANGLE IS THEN MADE AROUND THIS MINIMUM POINT. BANK ANGLES ON EITHER SIDE OF THE MINIMUM ARE USED TO OBTAIN TWO MORE IMPACT POINTS.

THESE 3 POINTS ARE FITTED TO A QUADRATIC TO OBTAIN A NEW ESTIMATED MINIMUM AND CORRESPONDING BANK ANGLE. USING THIS BANK ANGLE ANOTHER INTEGRATION IS MADE AND STORED IN THE TABLE. A NEW MINIMUM SEARCH IS MADE IN THE TABLE AND THE RESULT MAY EITHER BE STORED IN THE OUTPUT BLOCK OR FURTHER PERTURBATIONS MAY BE MADE AROUND THE NEW MINIMUM POINT BE DECREASING THE SIZE OF THE DELTA BANK ANGLE.

USAGE

ENTRY FROM AND EXIT TO MYTTF4 IN ORBIT (TTF) MODE.
ENTRY FROM AND EXIT TO MYREN2 IN REENTRY MODE.

A) CALLING SEQUENCE -

1. REENTRY -

```
        TSX   R5RARF,4
      αMZE    VECZ0,0,0
   α+1                    ERROR RETURN
   α+2                    NORMAL RETURN
```

THE MINUS SIGN IN THE FIRST WORD OF THE CALLING SEQUENCE INDICATES REENTRY. IF THE DECREMENT OF THE

FIRST WORD IN THE CALLING SEQUENCE IS NON ZERO, THE ZERO AND MAXIMUM LIFT I.P.'S ARE NOT COMPUTED. IF THE DECREMENT IS ZERO THE ZERO LIFT AND MAXIMUM LIFT I.P.'S ARE GENERATED.

THE ADDRESS OF THE FIRST WORD OF THE CALLING SEQUENCE GIVES THE LOCATION OF THE BURNOUT VECTOR OR THE VECTOR FROM DIFFERENTIAL CORRECTIONS TO BE USED AS THE REFERENCE POINT.

2. ORBIT -

```
        TSX   R5RARF,4
       αPZE   **,,**
    α+1                    ERROR RETURN
    α+2                    NORMAL RETURN
```

THE PLUS SIGN INDICATES TIME TO FIRE MODE.

B) STORAGE REQUIRED -

1.	INSTRUCTIONS -	2773	
2.	CELLS -	532	
3.	SUBROUTINES -	382	LAGIN, DIN1 AND TIN1
4.	TABLES -	4076	DTBLE, DATBL, TLRCR, R5RTM, TBFDR
	TOTAL	7763	

C) ACCURACY -

NUMEROUS CASES HAVE BEEN SUCCESSFULLY RUN WITH TARGET POSITION SELECTED IN ALL AREAS OF THE FOOTPRINT. A LARGE NUMBER OF TARGET POSITION OUTSIDE THE FOOTPRINT WERE RUN TO TEST ERROR LOGIC AND THE COMPUTED PARAMETERS FOR CLOSEST APPROACH TO THE TARGET.

D) REFERENCES -

1. SUBROUTINES

GMDIN1 - DOUBLE INTERPOLATION SUBROUTINE SHARE #239.

UTNTP - FLOATING POINT INTERPOLATER BY KUNKO, GODDARD

GMTIN1 - TABLE INTERPOLATION SUBROUTINE OLS ZTVN

LAGIN - LAGRANGE INTERPOLATER SHARE

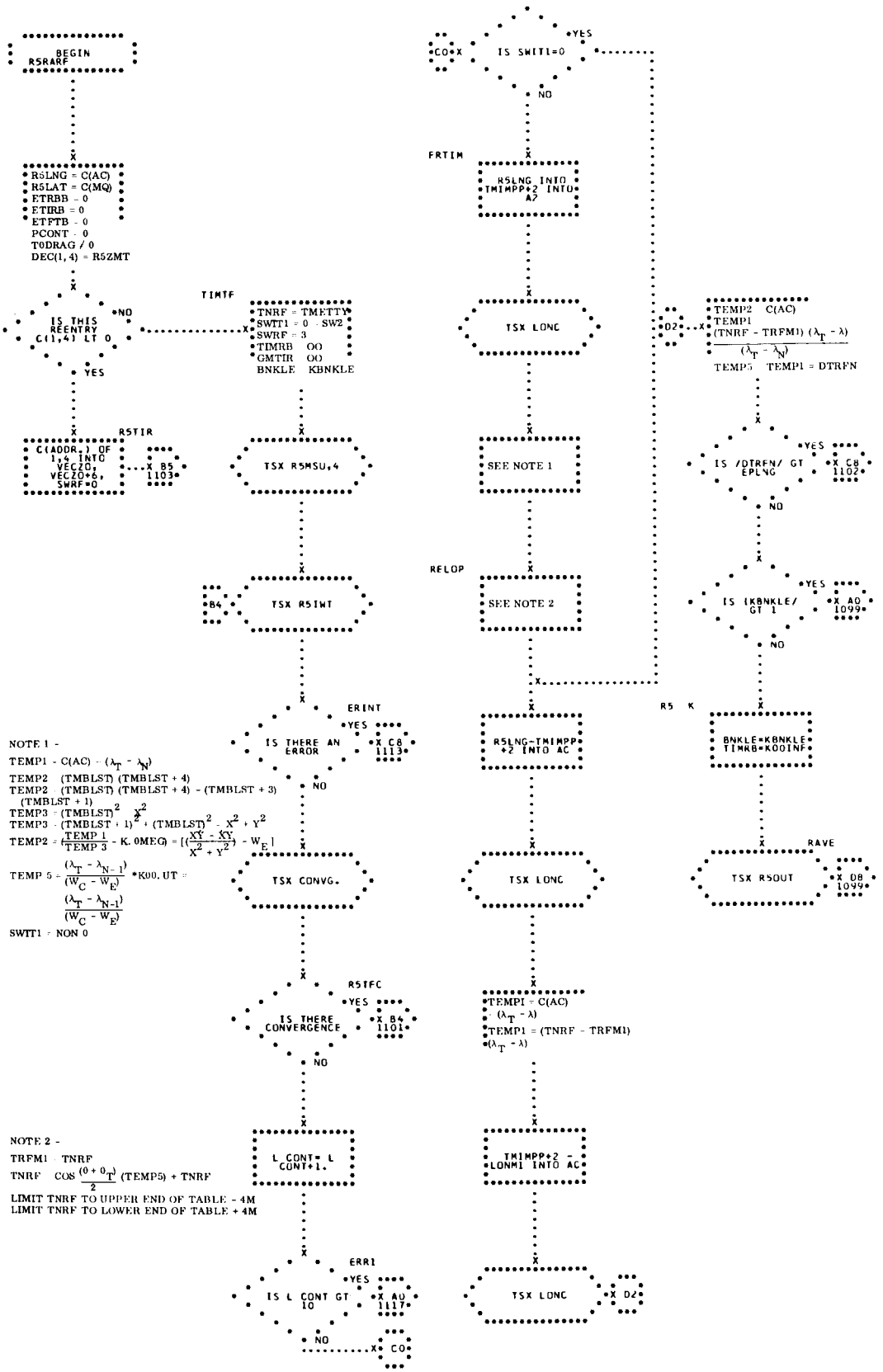
2. ANALYSIS -

''PAPER ON TTF RECTANGULAR BDRY LOGIC'' - MR. E. NINER, GODDARD

''DELTA COMPUTATIONS'' - MR. E. NINER, GODDARD

''BACKUP REENTRY GUIDANCE + CONTROL PROCEDURE FOR GEMINI'' BY MR. J. RUTLAND, HOUSTON

''ELLIPTICAL FOOTPRINT COMPUTATION AND MINIMUM RADIAL ERROR COMPUTATION'' BY MR. R. HOOVER, GODDARD



NOTE 1 -

TEMP1 - C(AC) - $(\lambda_T - \lambda_N)$

TEMP2 (TMBLST) (TMBLST + 4)

TEMP2 (TMBLST) (TMBLST + 4) - (TMBLST + 3) (TMBLST + 1)

TEMP3 = $(TMBLST)^2 \; X^2$

TEMP3 = $(TMBLST + 1)^2 + (TMBLST)^2 - X^2 + Y^2$

TEMP2 = $\left(\frac{TEMP\ 1}{TEMP\ 3} - K.\ 0MEG\right) = \left[\left(\frac{XY - XY}{X^2 + Y^2}\right) - W_E\right]$

TEMP 5 = $\frac{(\lambda_T - \lambda_{N-1})}{(W_C - W_E)}$ *K00. UT =

$\frac{(\lambda_T - \lambda_{N-1})}{(W_C - W_E)}$

SWTT1 = NON 0

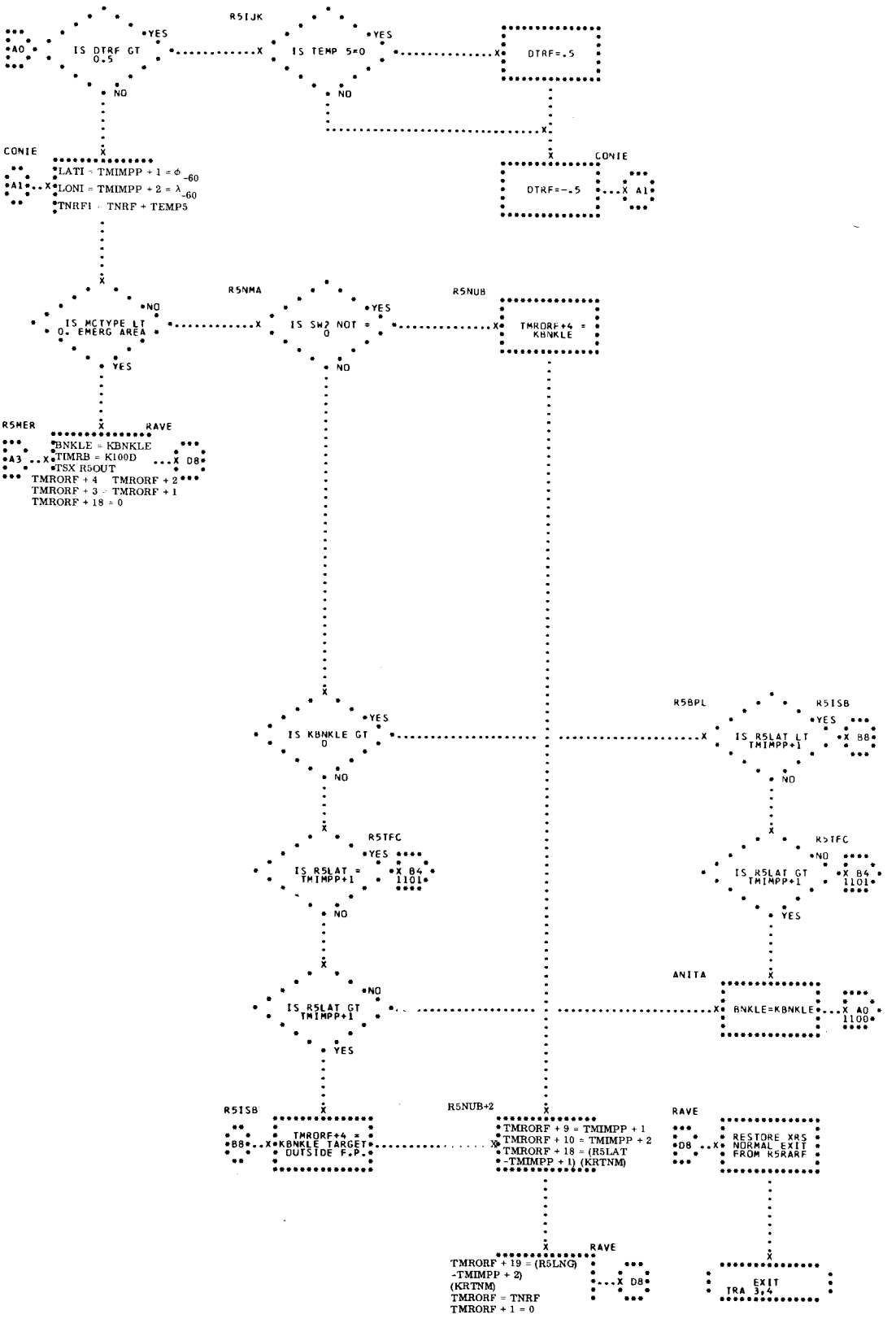
NOTE 2 -

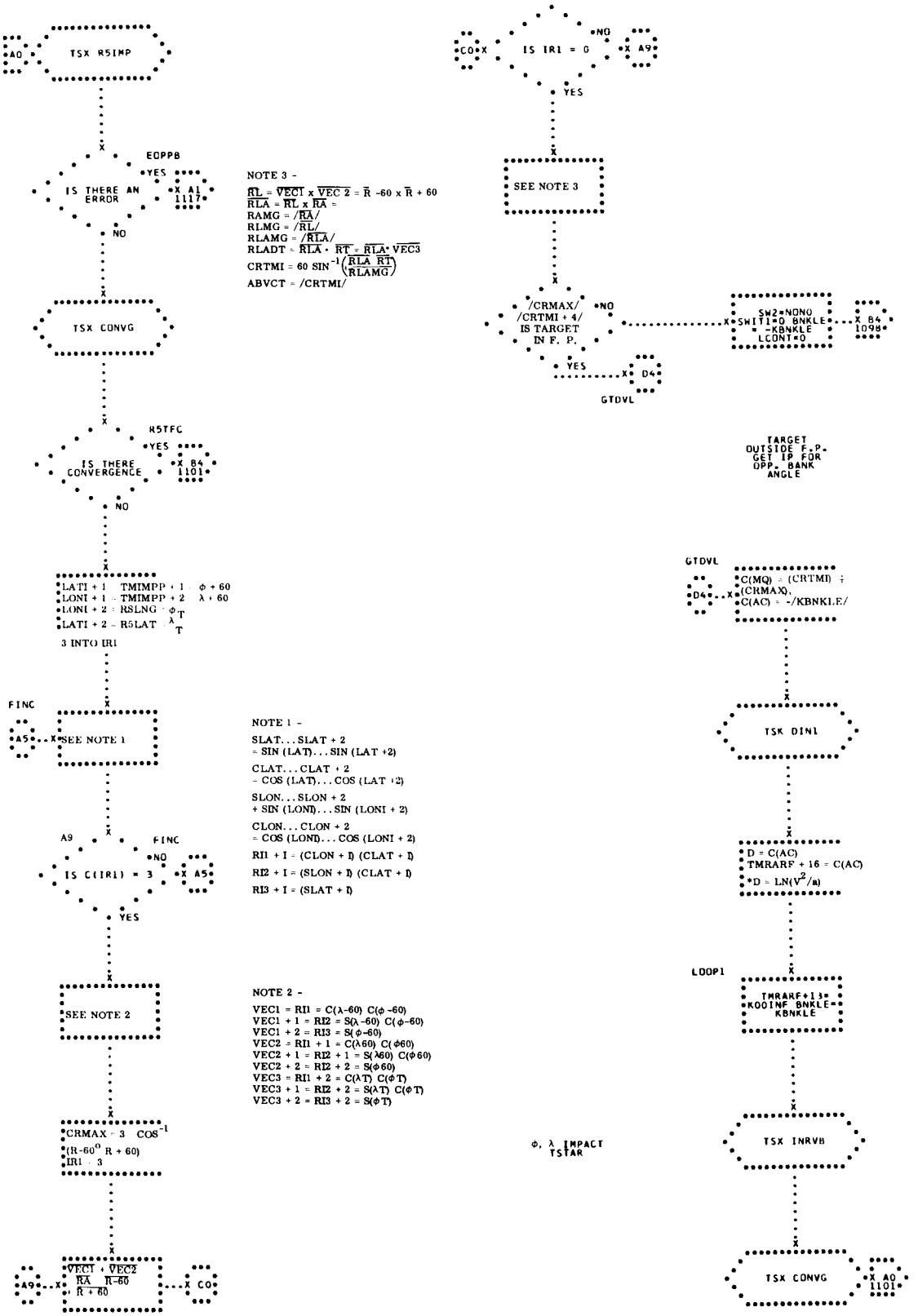
TRFM1 = TNRF

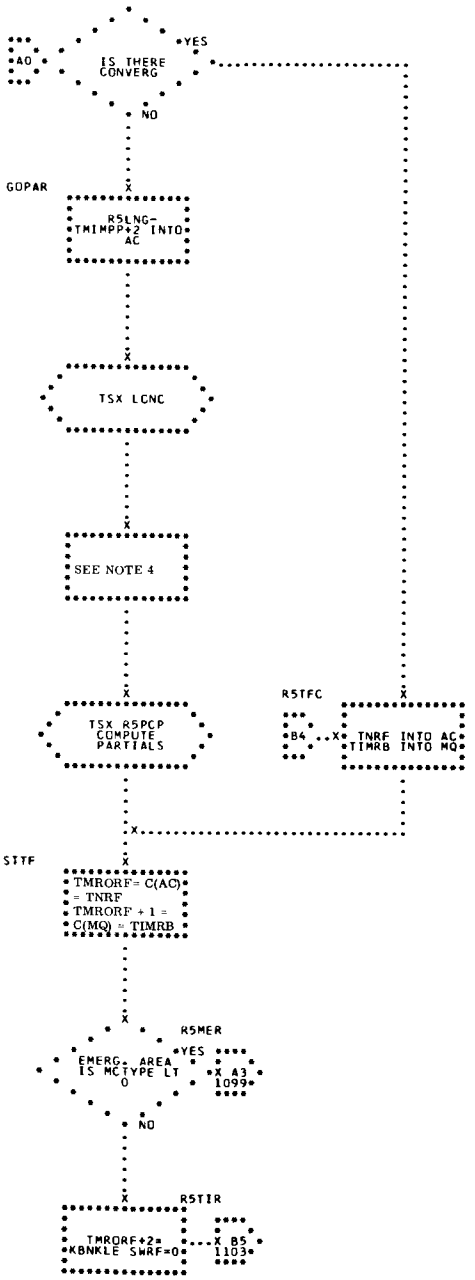
TNRF = $COS\ \frac{(0 + 0_T)}{2}$ (TEMP5) + TNRF

LIMIT TNRF TO UPPER END OF TABLE - 4M

LIMIT TNRF TO LOWER END OF TABLE + 4M

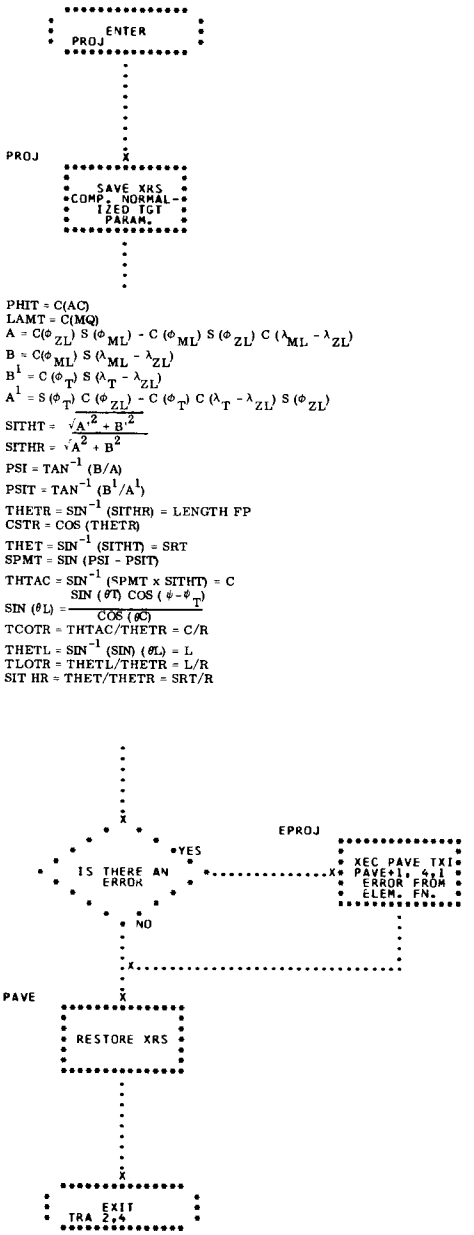

R5IJK
IS DTRF GT 0.5
YES
NO
AO
IS TEMP 5=0
YES
NO
DTRF=.5
DTRF=-.5
CONIE
A1
LATI = TMIMPP + 1 = φ -60
LONI = TMIMPP + 2 = λ -60
TNRF1 = TNRF + TEMP5
IS MCTYPE LT 0. EMERG AREA
NO
YES
R5NMA
IS SW2 NOT = 0
YES
NO
R5NUB
TMRORF+4 = KBNKLE
R5MER
RAVE
BNKLE = KBNKLE
TIMRB = K100D
TSX R5OUT
A3
D8
TMRORF + 4 = TMRORF + 2
TMRORF + 3 = TMRORF + 1
TMRORF + 18 = 0
IS KBNKLE GT 0
IS R5LAT = TMIMPP+1
B4
1101
IS R5LAT GT TMIMPP+1
R5BPL
R5ISB
IS R5LAT LT TMIMPP+1
B8
ANITA
BNKLE=KBNKLE
A0
1100
R5ISB
TMRORF+4 = KBNKLE TARGET OUTSIDE F.P.
B8
R5NUB+2
TMRORF + 9 = TMIMPP + 1
TMRORF + 10 = TMIMPP + 2
TMRORF + 18 = (R5LAT -TMIMPP + 1) (KRTNM)
RAVE
RESTORE XRS NORMAL EXIT FROM R5RARF
TMRORF + 19 = (R5LNG) -TMIMPP + 2) (KRTNM)
TMRORF = TNRF
TMRORF + 1 = 0
EXIT
TRA 3,4


TSX R5IMP
A0
EOPPB
IS THERE AN ERROR
YES
X A1 1117
NO
TSX CONVG
R5TFC
IS THERE CONVERGENCE
YES
X B4 1101
NO
LATI + 1 TMIMPP + 1 φ + 60
LONI + 1 TMIMPP + 2 λ + 60
LONI + 2 RSLNG φT
LATI + 2 R5LAT λT
3 INTO IR1
FINC
A5
SEE NOTE 1
A9
FINC
IS C(IR1) = 3
NO
X A5
YES
SEE NOTE 2
CRMAX = 3 COS-1
(R-60° R + 60)
IR1 = 3
VEC1 + VEC2
RA R-60
R + 60
A9
X CO
NOTE 3 -
RL = VEC1 x VEC 2 = R -60 x R + 60
RLA = RL x RA =
RAMG = /RA/
RLMG = /RL/
RLAMG = /RLA/
RLADT = RLA · RT = RLA· VEC3
CRTMI = 60 SIN-1(RLA RT / RLAMG)
ABVCT = /CRTMI/
NOTE 1 -
SLAT...SLAT + 2
= SIN (LAT)...SIN (LAT +2)
CLAT...CLAT + 2
= COS (LAT)...COS (LAT +2)
SLON...SLON + 2
= SIN (LONI)...SIN (LONI + 2)
CLON...CLON + 2
= COS (LONI)...COS (LONI + 2)
RI1 + I = (CLON + I) (CLAT + I)
RI2 + I = (SLON + I) (CLAT + I)
RI3 + I = (SLAT + I)
NOTE 2 -
VEC1 = RI1 = C(λ-60) C(φ -60)
VEC1 + 1 = RI2 = S(λ -60) C(φ-60)
VEC1 + 2 = RI3 = S(φ-60)
VEC2 = RI1 + 1 = C(λ60) C(φ60)
VEC2 + 1 = RI2 + 1 = S(λ60) C(φ60)
VEC2 + 2 = RI2 + 2 = S(φ60)
VEC3 = RI1 + 2 = C(λT) C(φT)
VEC3 + 1 = RI2 + 2 = S(λT) C(φT)
VEC3 + 2 = RI3 + 2 = S(φT)
CO
IS IR1 = 0
NO
X A9
YES
SEE NOTE 3
/CRMAX/
/CRTMI + 4/
IS TARGET IN F. P.
NO
SW2=NONO
SWIT1=0 BNKLE
= -KBNKLE
LCONT=0
X B4 1098
YES
X D4
GTDVL
TARGET OUTSIDE F.P. GET IP FOR OPP. BANK ANGLE
GTDVL
D4
C(MQ) = (CRTMI) ÷
(CRMAX),
C(AC) = -/KBNKLE/
TSK DIN1
D = C(AC)
TMRARF + 16 = C(AC)
*D = LN(V2/a)
LOOP1
TMRARF+1 =
KOOINF BNKLE=
KBNKLE
TSX INRVB
φ, λ IMPACT TSTAR
TSX CONVG
X A0 1101



NOTE 4 -

DELG1 = C(AC)
DLG1 = C(AC)
DEPG1 = $\theta_T - \theta$
DPG1 = $\theta_T - \theta$
SWRF = + 3
BNKLE = KBNKLE
C(MQ) = TSTAR
C(AC) = TNRF1



PHIT = C(AC)
LAMT = C(MQ)

$A = C(\phi_{ZL})\, S(\phi_{ML}) - C(\phi_{ML})\, S(\phi_{ZL})\, C(\lambda_{ML} - \lambda_{ZL})$

$B = C(\phi_{ML})\, S(\lambda_{ML} - \lambda_{ZL})$

$B^1 = C(\phi_T)\, S(\lambda_T - \lambda_{ZL})$

$A^1 = S(\phi_T)\, C(\phi_{ZL}) - C(\phi_T)\, C(\lambda_T - \lambda_{ZL})\, S(\phi_{ZL})$

$SITHT = \frac{\sqrt{A^{1\,2} + B^{1\,2}}}{\sqrt{A^2 + B^2}}$

$PSI = TAN^{-1}(B/A)$

$PSIT = TAN^{-1}(B^1/A^1)$

THETR = SIN^{-1} (SITHR) = LENGTH FP
CSTR = COS (THETR)

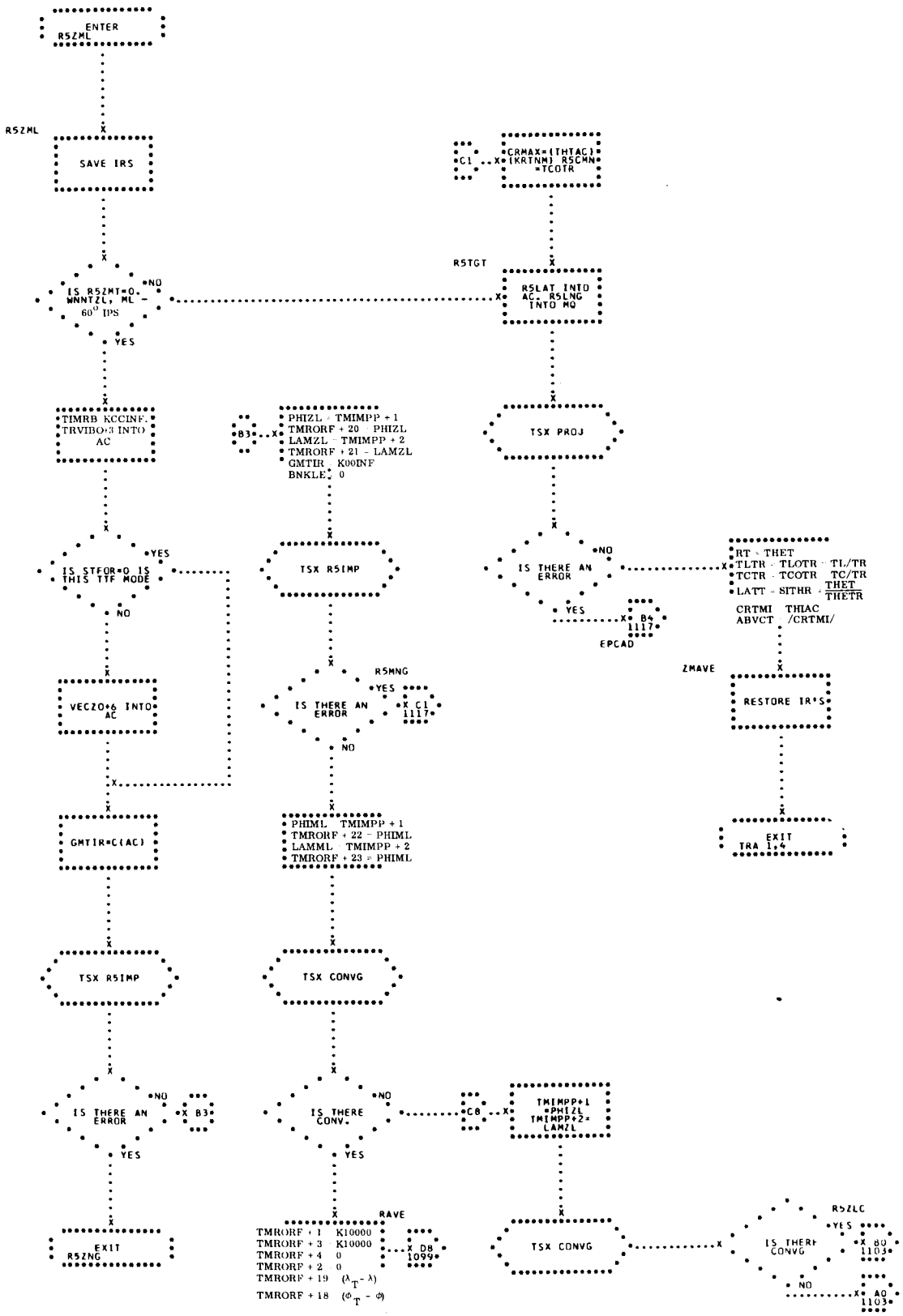
THET = SIN^{-1} (SITHT) = SRT
SPMT = SIN (PSI - PSIT)

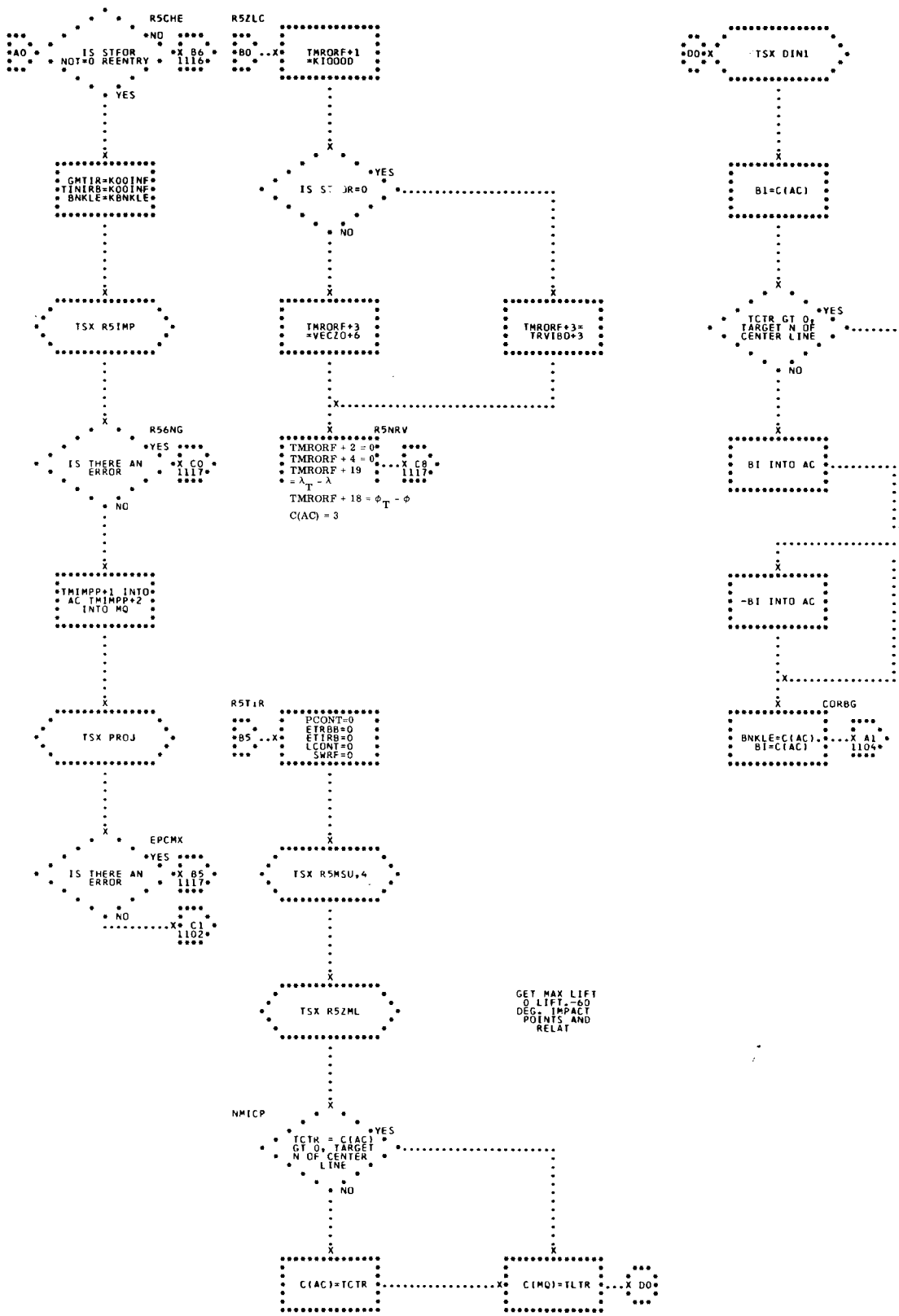
THTAC = SIN^{-1} (SPMT x SITHT) = C

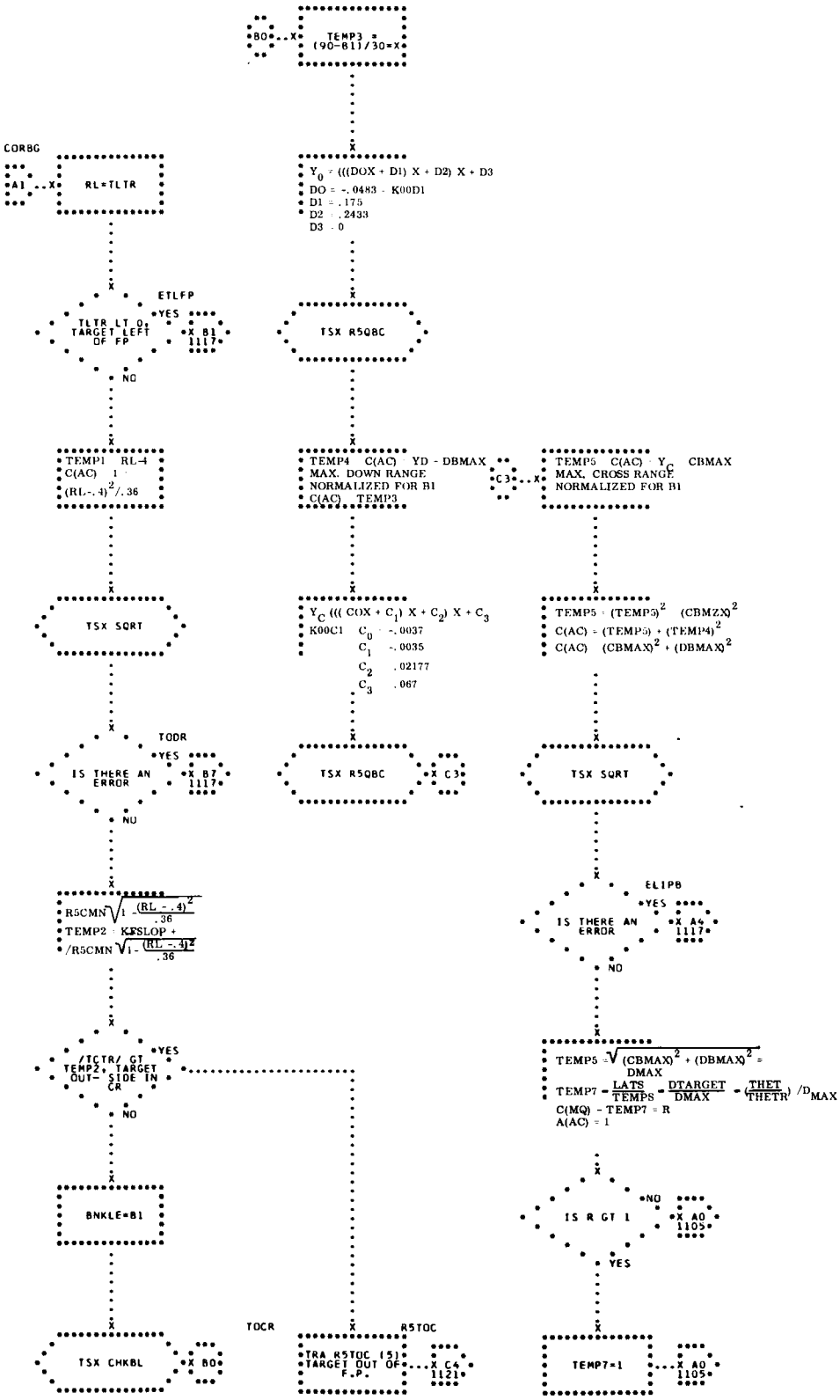
$SIN(\theta L) = \frac{SIN(\theta T)\, COS(\psi - \psi_T)}{COS(\theta C)}$

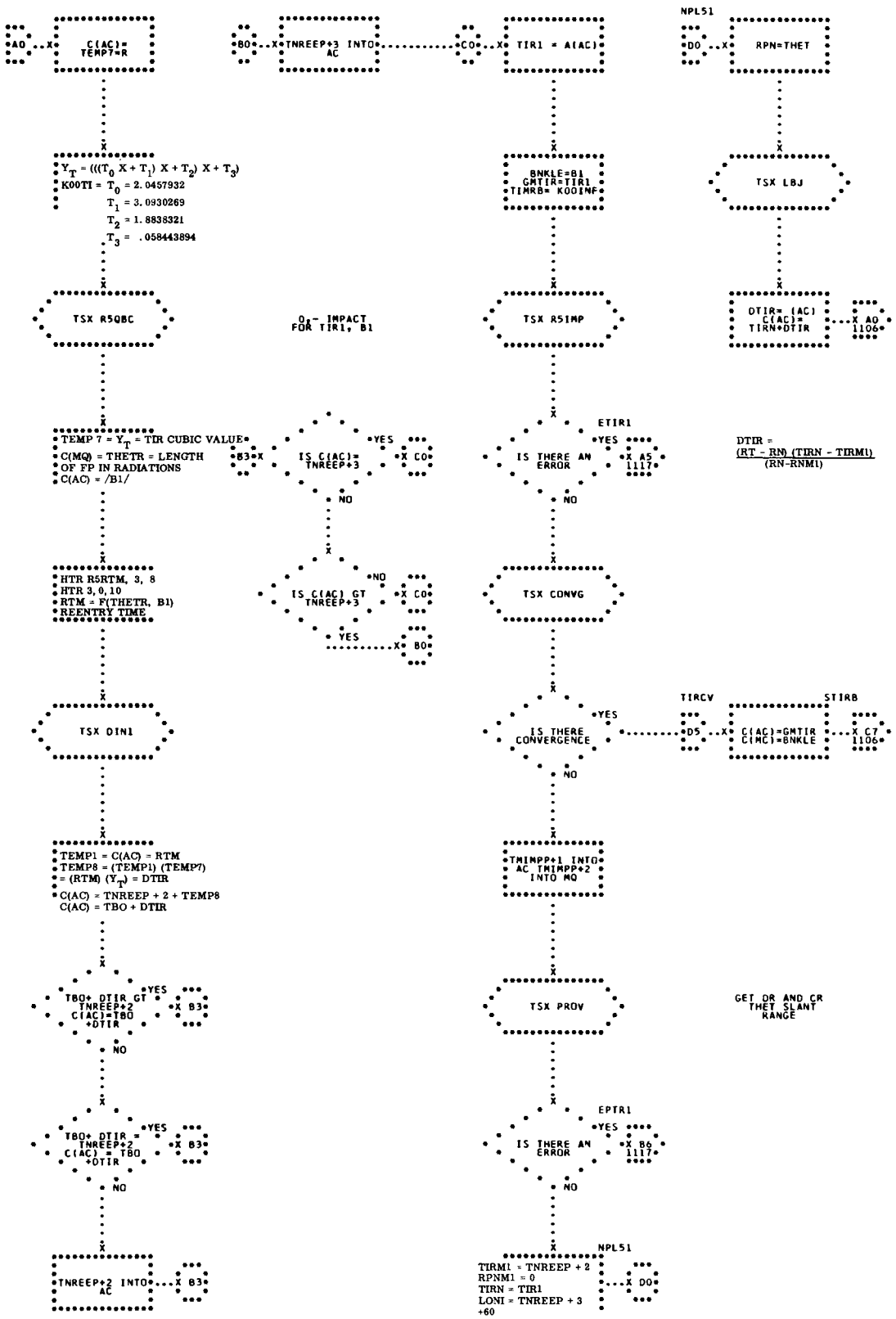
TCOTR = THTAC/THETR = C/R

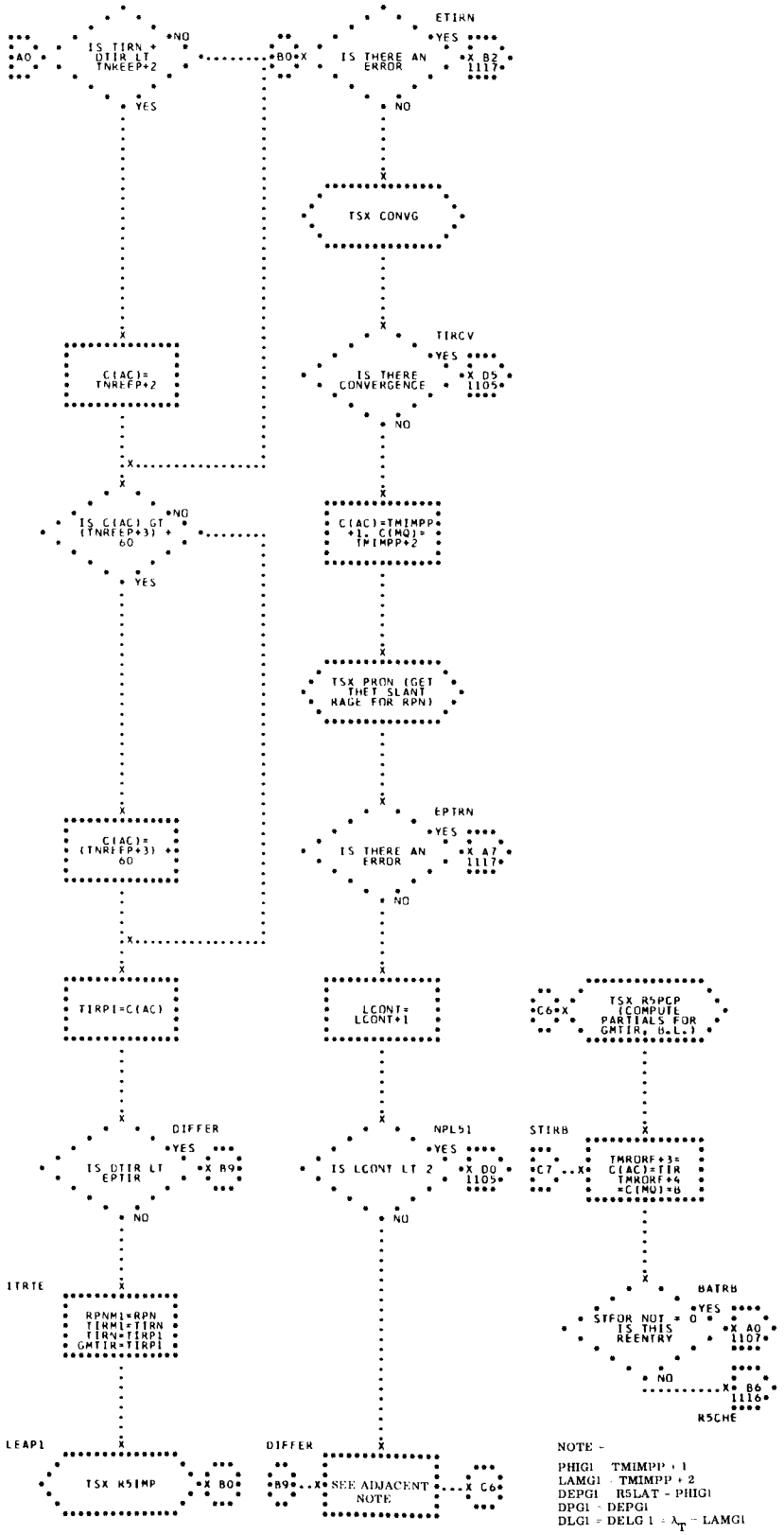
THETL = SIN^{-1} (SIN) (θL) = L
TLOTR = THETL/THETR = L/R
SIT HR = THET/THETR = SRT/R


ENTER R5ZML
R5ZML
SAVE IRS
IS R5ZMT=0. WNNTZL, ML - 60° IPS
NO
YES
TIMRB KCCINF, TRVIBO+3 INTO AC
IS STFOR=0 IS THIS TTF MODE
YES
NO
VECZO+6 INTO AC
GMTIR=C(AC)
TSX R5IMP
IS THERE AN ERROR
B3
EXIT R5ZNG
B3
PHIZL - TMIMPP + 1
TMRORF + 20 - PHIZL
LAMZL - TMIMPP + 2
TMRORF + 21 - LAMZL
GMTIR - K00INF
BNKLE - 0
TSX R5IMP
R5MNG
IS THERE AN ERROR
YES
C1 1117
NO
PHIML - TMIMPP + 1
TMRORF + 22 - PHIML
LAMML - TMIMPP + 2
TMRORF + 23 - PHIML
TSX CONVG
IS THERE CONV.
NO
YES
RAVE
TMRORF + 1 K10000
TMRORF + 3 K10000
TMRORF + 4 0
TMRORF + 2 0
TMRORF + 19 (λT - λ)
TMRORF + 18 (φT - φ)
D8 1099
C1
CRMAX=(THTAC)*(KRTNM) R5CMN*TCOTR
R5TGT
R5LAT INTO AC, R5LNG INTO MQ
TSX PROJ
IS THERE AN ERROR
NO
YES
B4 1117
EPCAD
C8
TMIMPP+1 *PHIZL TMIMPP+2* LAMZL
TSX CONVG
RT - THET
TLTR - TLOTR - TL/TR
TCTR - TCOTR - TC/TR
LATT - SITHR - THET/THETR
CRTMI THIAC
ABVCT - /CRTMI/
ZMAVE
RESTORE IR'S
EXIT TRA 1,4
R5ZLC
IS THERE CONVG
YES
NO
B0 1103
A0 1103


R5CHE
IS STFOR NOT=0 REENTRY
NO
YES
X B6 1116
A0
GMTIR=KOOINF
TINIRB=KOOINF
BNKLE=KBNKLE
TSX R5IMP
R56NG
IS THERE AN ERROR
YES
X C0 1117
NO
TMIMPP+1 INTO AC TMIMPP+2 INTO MQ
TSX PROJ
EPCMX
IS THERE AN ERROR
YES
X B5 1117
NO
X C1 1102
R5ZLC
B0
TMRORF+1 =KIOOOD
IS ST OR=0
YES
NO
TMRORF+3 =VECZO+6
TMRORF+3= TRVIBO+3
R5NRV
TMRORF + 2 = 0
TMRORF + 4 = 0
TMRORF + 19 = $\lambda_T - \lambda$
X C8 1117
TMRORF + 18 = $\phi_T - \phi$
C(AC) = 3
R5TIR
B5
PCONT=0
ETRBB=0
ETIRB=0
LCONT=0
SWRF=0
TSX R5MSU,4
TSX R5ZML
NMICP
TCTR = C(AC) GT 0, TARGET N OF CENTER LINE
YES
NO
C(AC)=TCTR
C(MQ)=TLTR
X D0
GET MAX LIFT Q LIFT,-60 DEG. IMPACT POINTS AND RELAT
D0
TSX DIN1
B1=C(AC)
TCTR GT 0, TARGET N OF CENTER LINE
YES
NO
BI INTO AC
-BI INTO AC
CORBG
BNKLE=C(AC), BI=C(AC)
X A1 1104


B0
TEMP3 = (90-B11/30=X
CORBG
A1
RL=TLTR
$Y_0 = (((DOX + D1) X + D2) X + D3$
DO = -.0483 - K00D1
D1 = .175
D2 = .2433
D3 = 0
ETLFP
TLTR LT 0, TARGET LEFT OF FP
YES
X B1 1117
NO
TSX R5QBC
TEMP1 RL-4
C(AC) 1
$(RL-.4)^2/.36$
TEMP4 C(AC) YD - DBMAX
MAX. DOWN RANGE
NORMALIZED FOR B1
C(AC) TEMP3
C3
TEMP5 C(AC) Y_C CBMAX
MAX. CROSS RANGE
NORMALIZED FOR B1
TSX SQRT
$Y_C = (((COX + C_1) X + C_2) X + C_3$
K00C1 C_0 -.0037
C_1 -.0035
C_2 .02177
C_3 .067
$TEMP5 = (TEMP5)^2 \quad (CBMZX)^2$
$C(AC) = (TEMP5) + (TEMP4)^2$
$C(AC) \quad (CBMAX)^2 + (DBMAX)^2$
TODR
IS THERE AN ERROR
YES
X B7 1117
NO
TSX R5QBC
X C3
TSX SQRT
$R5CMN\sqrt{1 - \frac{(RL - .4)^2}{.36}}$
$TEMP2 = KESLOP + /R5CMN\sqrt{1 - \frac{(RL - .4)^2}{.36}}$
ELIPB
IS THERE AN ERROR
YES
X A4 1117
NO
/TCTR/ GT TEMP2, TARGET OUT-SIDE IN CR
YES
NO
$TEMP5 = \frac{\sqrt{(CBMAX)^2 + (DBMAX)^2}}{DMAX}$
$TEMP7 = \frac{LATS}{TEMP5} = \frac{DTARGET}{DMAX} = (\frac{THET}{THETR}) /D_{MAX}$
C(MQ) - TEMP7 = R
A(AC) = 1
IS R GT 1
NO
X A0 1105
YES
BNKLE=B1
TOCR
R5TOC
TSX CHKBL
X B0
TRA R5TOC (5), TARGET OUT OF F.P.
X C4 1121
TEMP7=1
X A0 1105

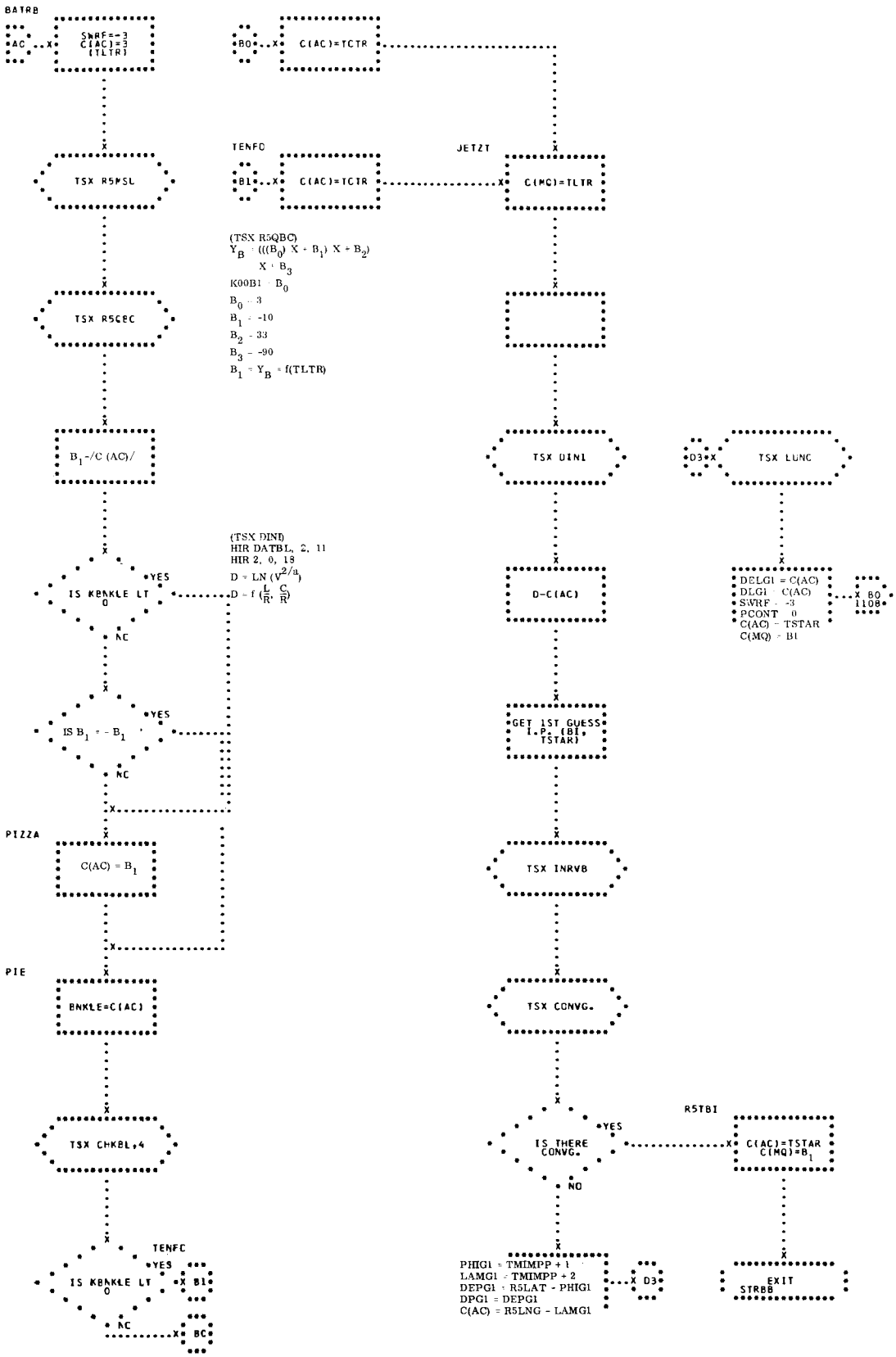

A0
C(AC)= TEMP7+R
$Y_T = (((T_0 X + T_1) X + T_2) X + T_3)$
K00TI = T_0 = 2.0457932
T_1 = 3.0930269
T_2 = 1.8838321
T_3 = .058443894
TSX R5QBC
TEMP 7 = Y_T = TIR CUBIC VALUE
C(MQ) = THETR = LENGTH OF FP IN RADIATIONS
C(AC) = /B1/
HTR R5RTM, 3, 8
HTR 3, 0, 10
RTM = F(THETR, B1)
REENTRY TIME
TSX DIN1
TEMP1 = C(AC) = RTM
TEMP8 = (TEMP1) (TEMP7)
= (RTM) (Y_T) = DTIR
C(AC) = TNREEP + 2 + TEMP8
C(AC) = TBO + DTIR
TBO+ DTIR GT TNREEP+2
C(AC)=TBO +DTIR
YES
B3
NO
TBO+ DTIR = TNREEP+2
C(AC) = TBO +DTIR
YES
B3
NO
TNREEP+2 INTO AC
B3
B0
TNREEP+3 INTO AC
C0
O_T- IMPACT FOR TIR1, B1
IS C(AC)= TNREEP+3
YES
C0
NO
IS C(AC) GT TNREEP+3
NO
C0
YES
B0
TIR1 = A(AC)
BNKLE=B1
GMTIR=TIR1
TIMRB= K00INF
TSX R51MP
ETIR1
IS THERE AN ERROR
YES
A5
1117
NO
TSX CONVG
IS THERE CONVERGENCE
YES
D5
NO
TMIMPP+1 INTO AC TMIMPP+2 INTO MQ
TSX PROV
EPTR1
IS THERE AN ERROR
YES
B6
1117
NO
NPL51
TIRM1 = TNREEP + 2
RPNM1 = 0
TIRN = TIR1
LONI = TNREEP + 3
+60
D0
NPL51
RPN=THET
TSX LBJ
DTIR= (AC) C(AC)= TIRN+DTIR
A0
1106
DTIR = (RT - RN) (TIRN - TIRM1) / (RN-RNM1)
TIRCV
C7
STIRB
C(AC)=GMTIR
C(MC)=BNKLE
1106
GET DR AND CR THET SLANT RANGE

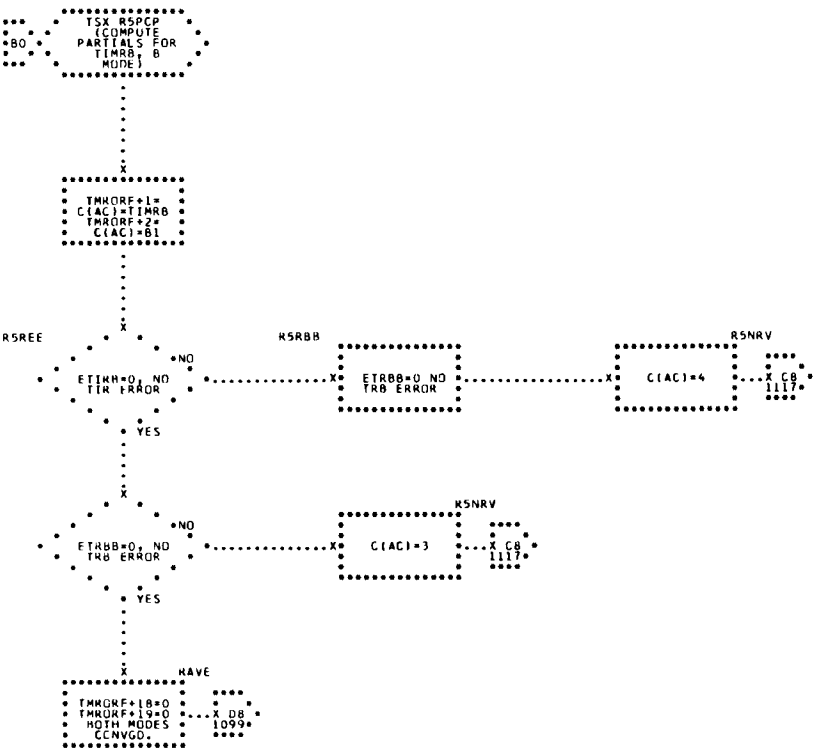


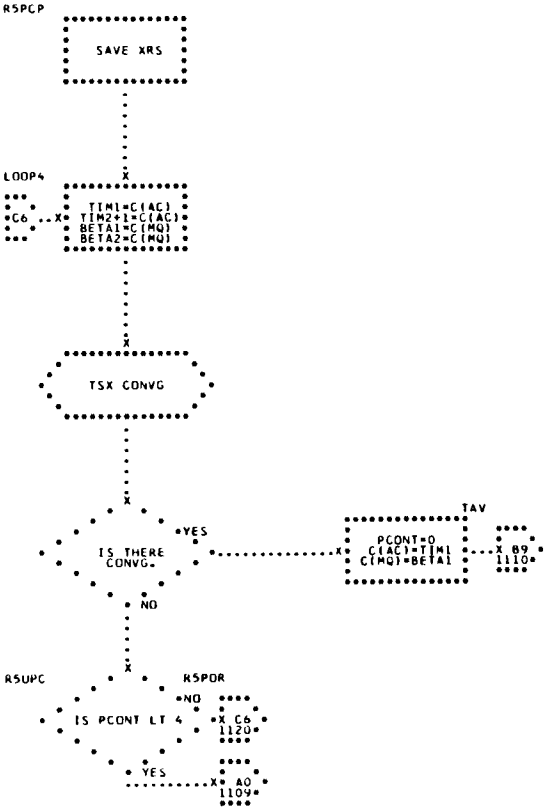
NOTE -

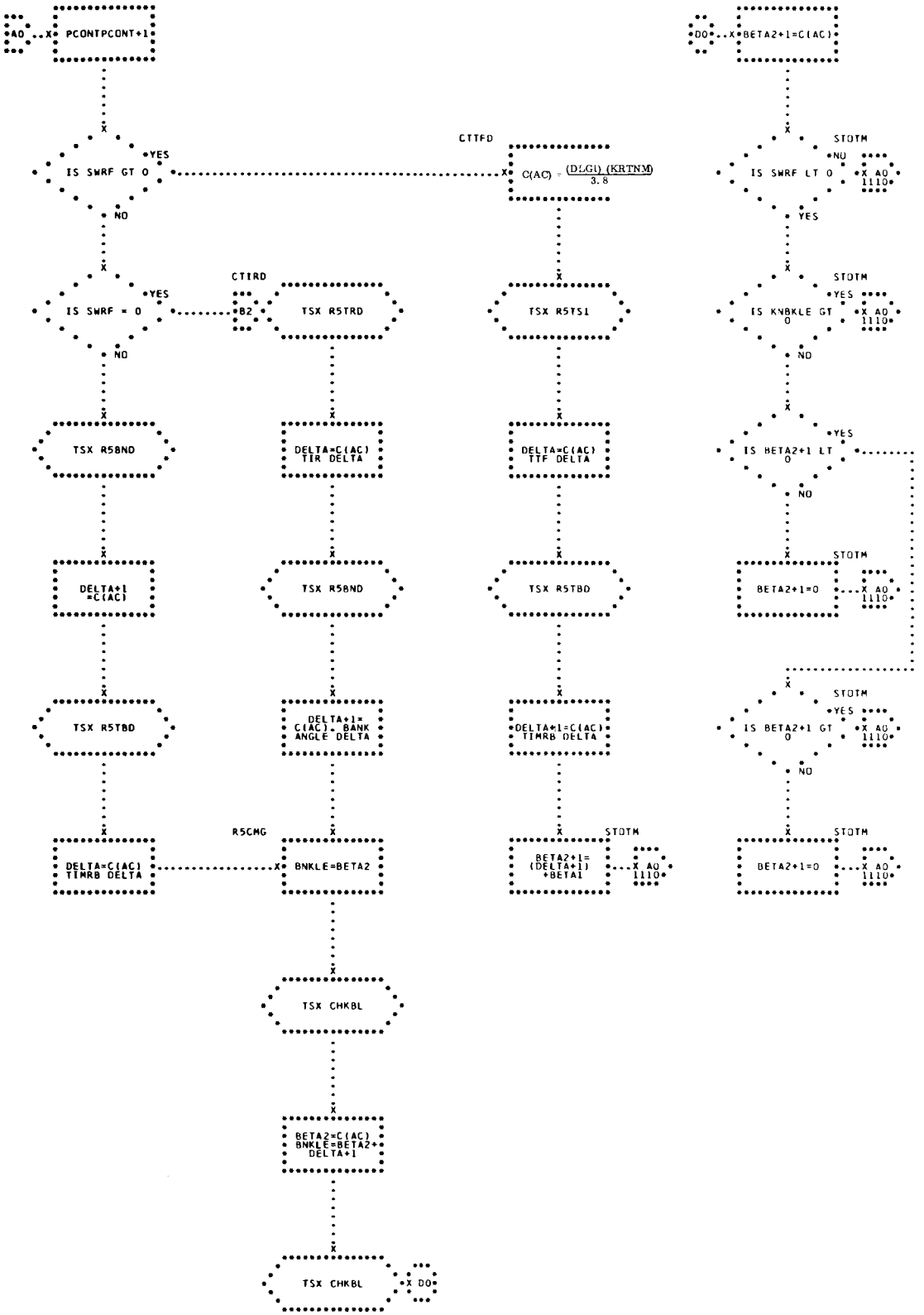
PHIG1 TMIMPP + 1
LAMG1 TMIMPP + 2
DEPG1 R5LAT - PHIG1
DPG1 DEPG1
DLG1 = DELG 1 = λ_T - LAMG1

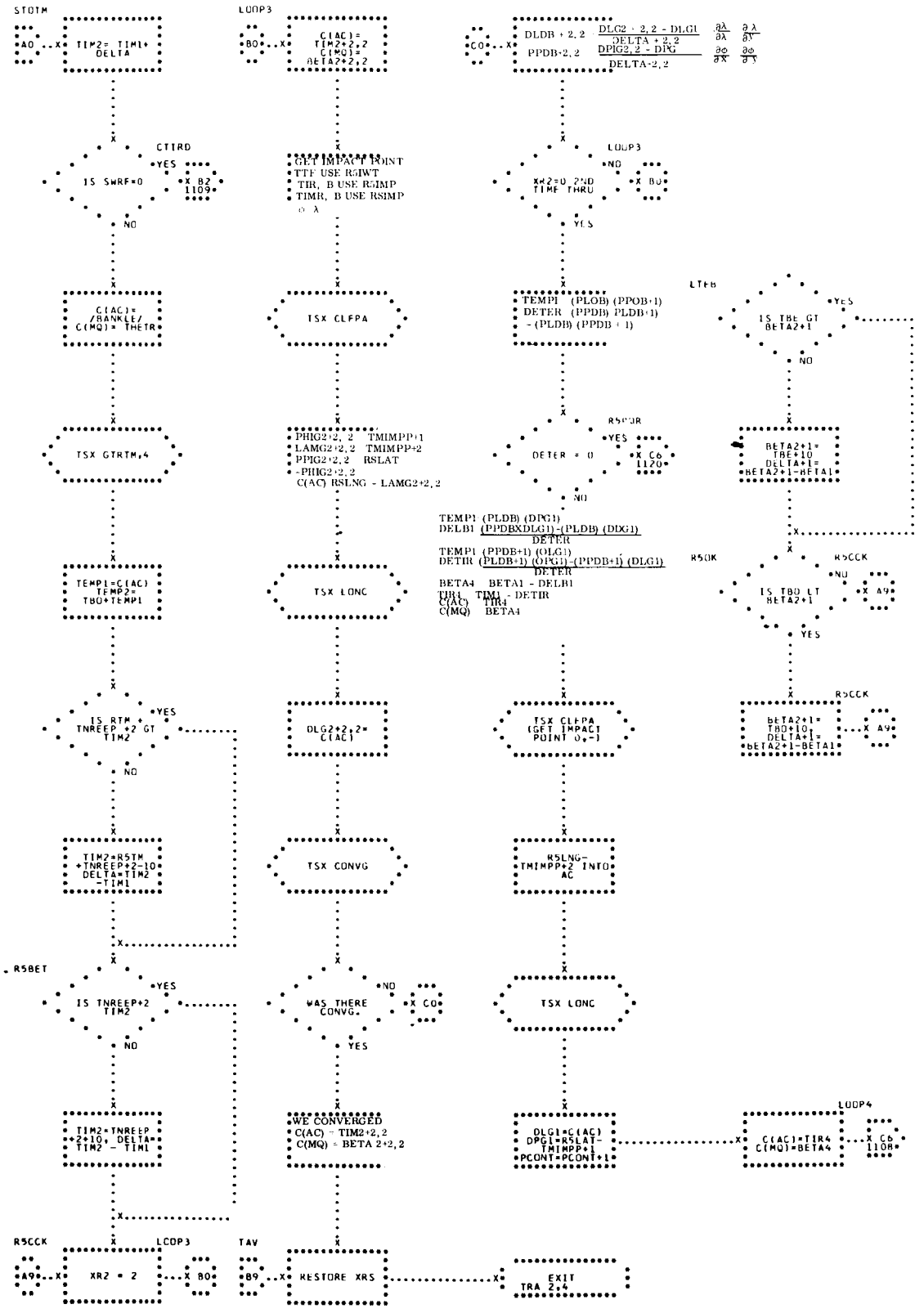
SWRF = 0
(C(AC) TIRN
C(MQ) = B1

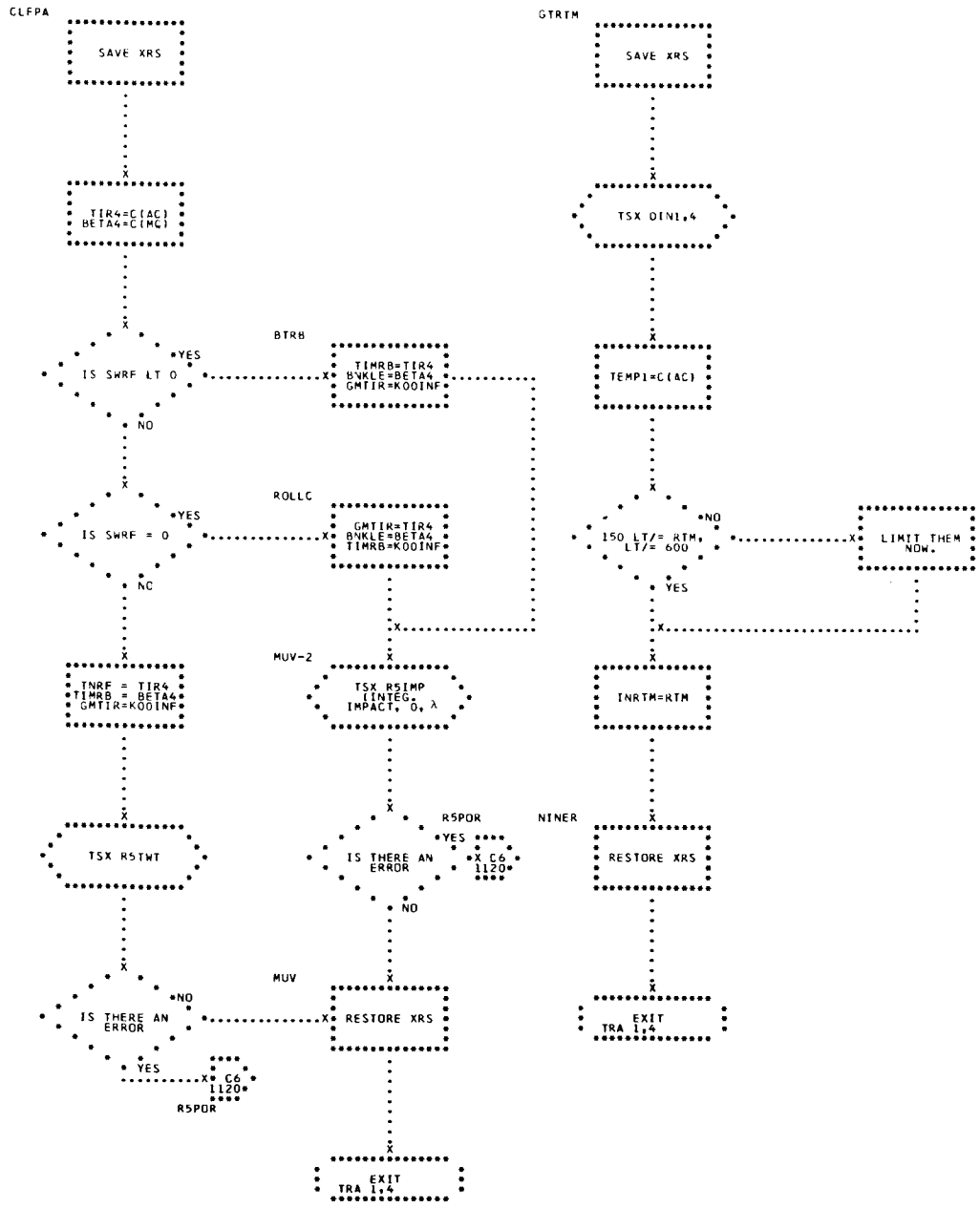

BATRB
AC
SWRF=-3
C(AC)=3
(TLTR)
B0
C(AC)=TCTR
TENFC
B1
C(AC)=TCTR
JETZT
C(MQ)=TLTR
TSX R5MSL
TSX R5QBC
(TSX R5QBC)
$Y_B = (((B_0) X + B_1) X + B_2) X + B_3$
K00B1 = B_0
B_0 = 3
B_1 = -10
B_2 = 33
B_3 = -90
$B_1 = Y_B + f(TLTR)$
B_1-/C (AC)/
TSX DIN1
D3
TSX LUNC
(TSX DIN)
HIR DATBL, 2, 11
HIR 2, 0, 18
$D = LN(V^{2/a})$
$D = f(\frac{L}{R}, \frac{C}{R})$
IS KBNKLE LT 0
YES
NO
D-C(AC)
DELG1 = C(AC)
DLG1 = C(AC)
SWRF = -3
PCONT = 0
C(AC) = TSTAR
C(MQ) = B1
X B0 1108
IS B_1 = - B_1
GET 1ST GUESS I.P. (B1, TSTAR)
PIZZA
C(AC) = B_1
TSX INRVB
PIE
BNKLE=C(AC)
TSX CONVG.
TSX CHKBL,4
IS THERE CONVG.
R5TB1
C(AC)=TSTAR
C(MQ)=B_1
TENFC
IS KBNKLE LT 0
X B1
NO
X B0
PHIG1 = TMIMPP + 1
LAMG1 = TMIMPP + 2
DEPG1 = R5LAT - PHIG1
DPG1 = DEPG1
C(AC) = R5LNG - LAMG1
X D3
EXIT
STRBB

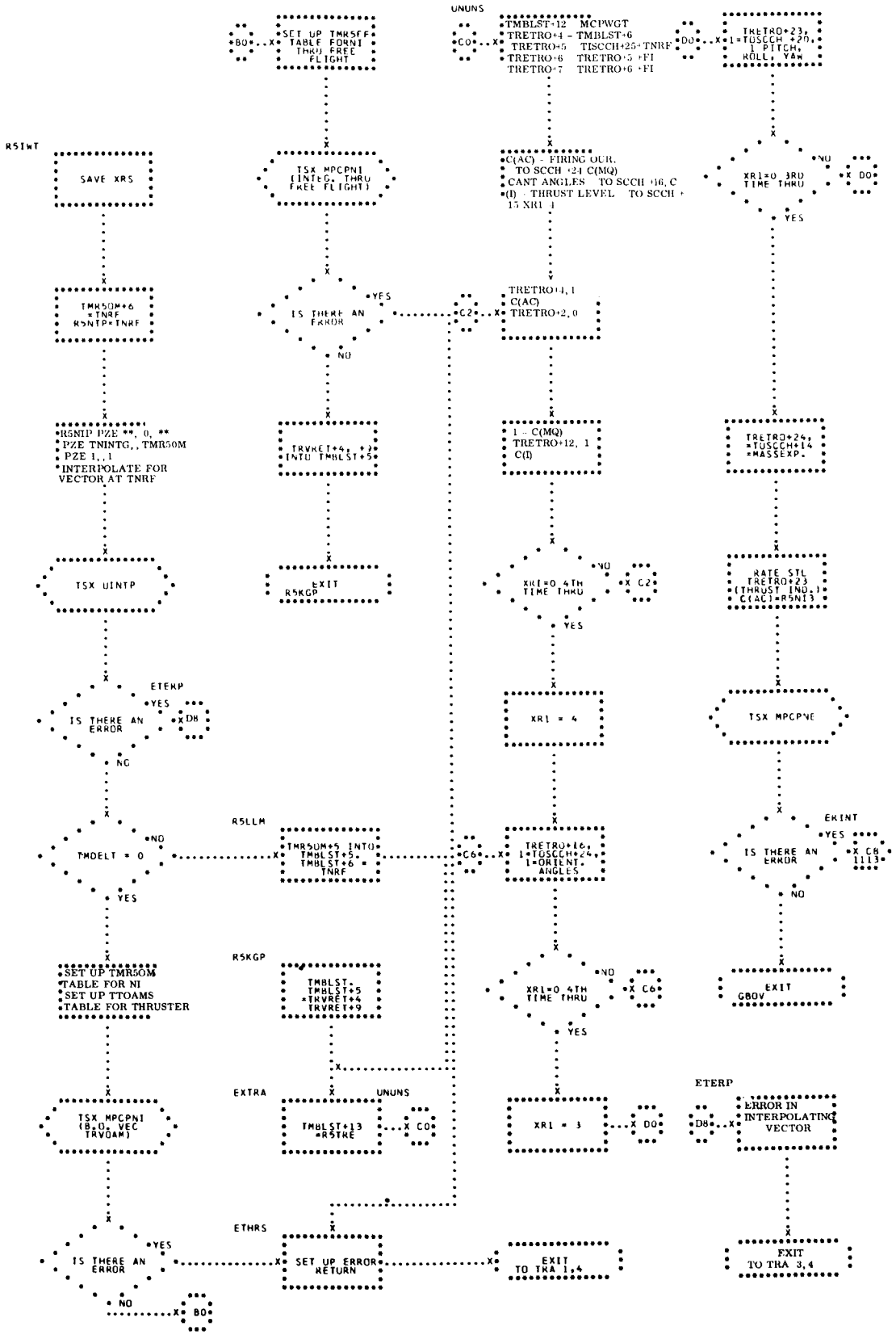

TSX R5PCP (COMPUTE PARTIALS FOR TIMRB, B MODE)
B0
TMRORF+1= C(AC)=TIMRB
TMRORF+2= C(AC)=B1
R5REE
ETIRH=0, NO TIR ERROR
NO
YES
R5RBB
ETRBB=0 NO TRB ERROR
R5NRV
C(AC)=4
X C8 1117
ETRBB=0, NO TRB ERROR
C(AC)=3
X C8 1117
R4VE
TMRORF+18=0 TMRORF+19=0 BOTH MODES CONVGD.
X D8 1099

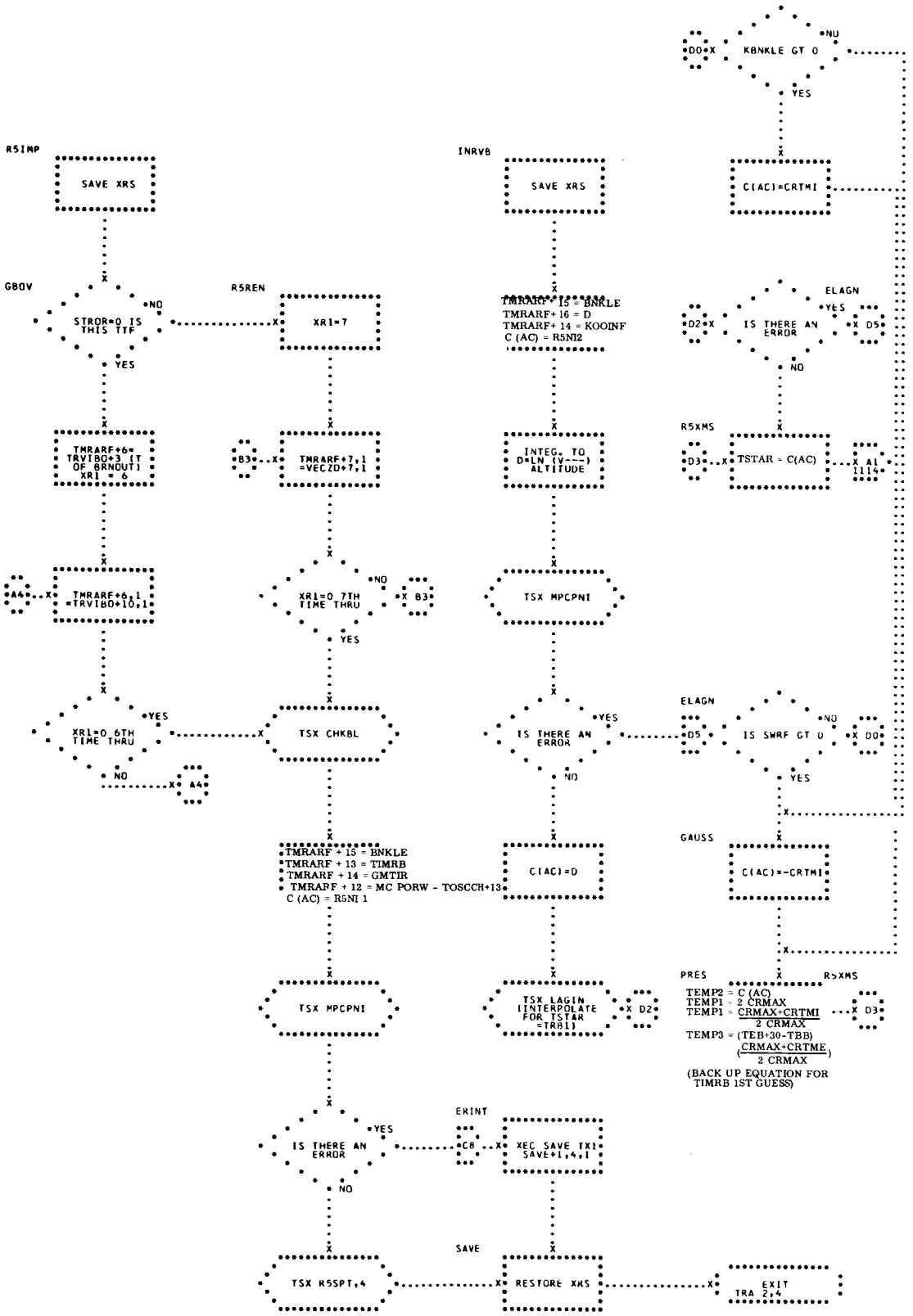

R5PCP
SAVE XRS
LOOP4
C6
TIM1=C(AC) TIM2=1=C(AC) BETA1=C(MQ) BETA2=C(MQ)
TSX CONVG
IS THERE CONVG.
YES
NO
TAV
PCONT=0 C(AC)=TIM1 C(MQ)=BETA1
X B9 1110
R5UPC
IS PCONT LT 4
R5POR
NO
X C6 1120
YES
A0 1109

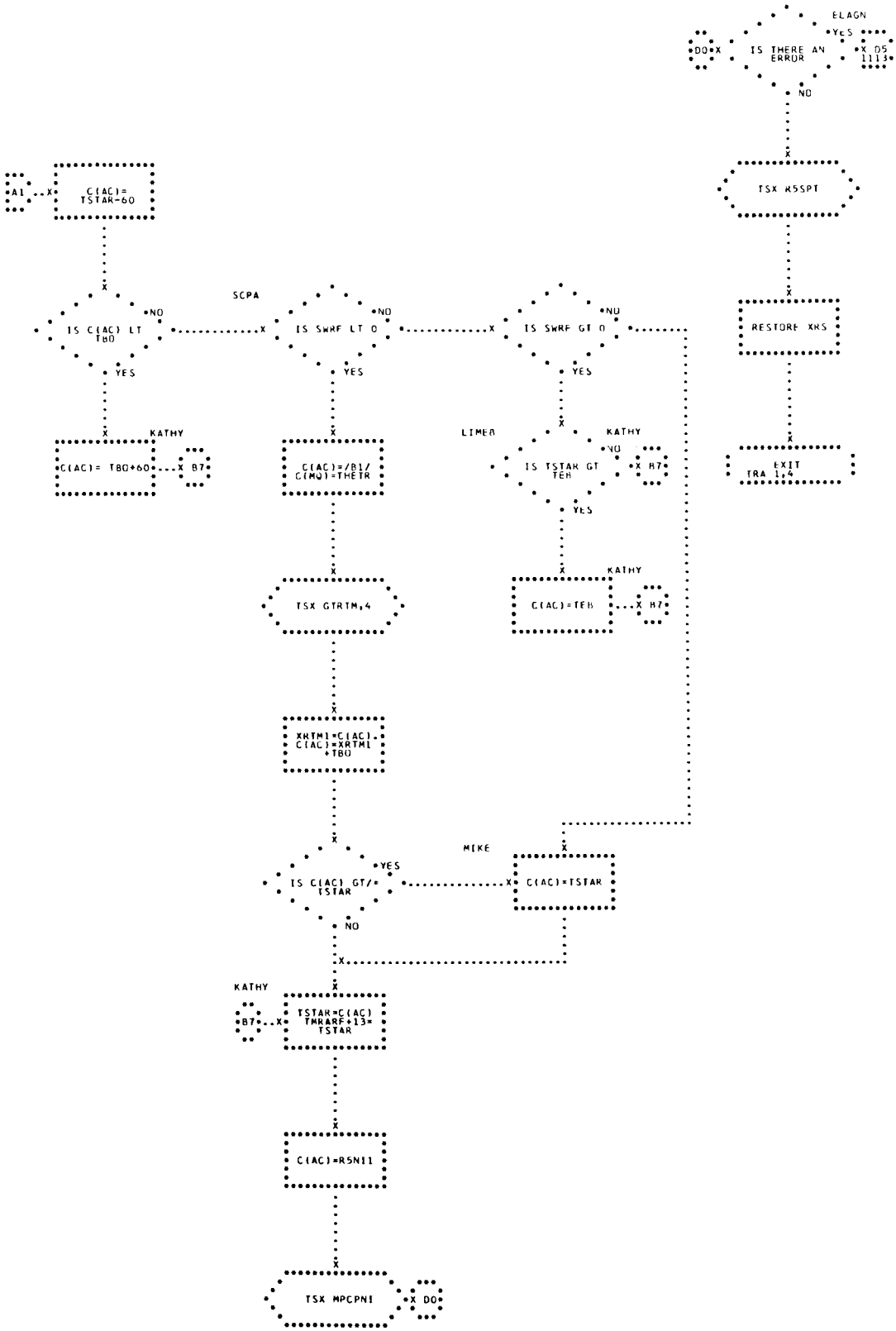

AO
PCONTPCONT+1
IS SWRF GT 0
YES
NO
CTTFD
C(AC) = (DLG1) (KRTNM) / 3.8
IS SWRF = 0
YES
NO
CTIRD
B2
TSX R5TRD
TSX R5TS1
TSX R58ND
DELTA=C(AC) TIR DELTA
DELTA=C(AC) TTF DELTA
DELTA+1 =C(AC)
TSX R58ND
TSX R5TBD
TSX R5TBD
DELTA+1= C(AC) - BANK ANGLE DELTA
DELTA+1=C(AC) + TTMRB DELTA
R5CMG
DELTA=C(AC) + TTMRB DELTA
BNKLE=BETA2
BETA2+1= (DELTA+1) +BETA1
STOTM
X AQ 1110
TSX CHKBL
BETA2=C(AC) BNKLE=BETA2+ DELTA+1
TSX CHKBL
X DO
DO
BETA2+1=C(AC)
IS SWRF LT 0
NO
YES
IS KNBKLE GT 0
IS BETA2+1 LT 0
BETA2+1=0
IS BETA2+1 GT 0
BETA2+1=0

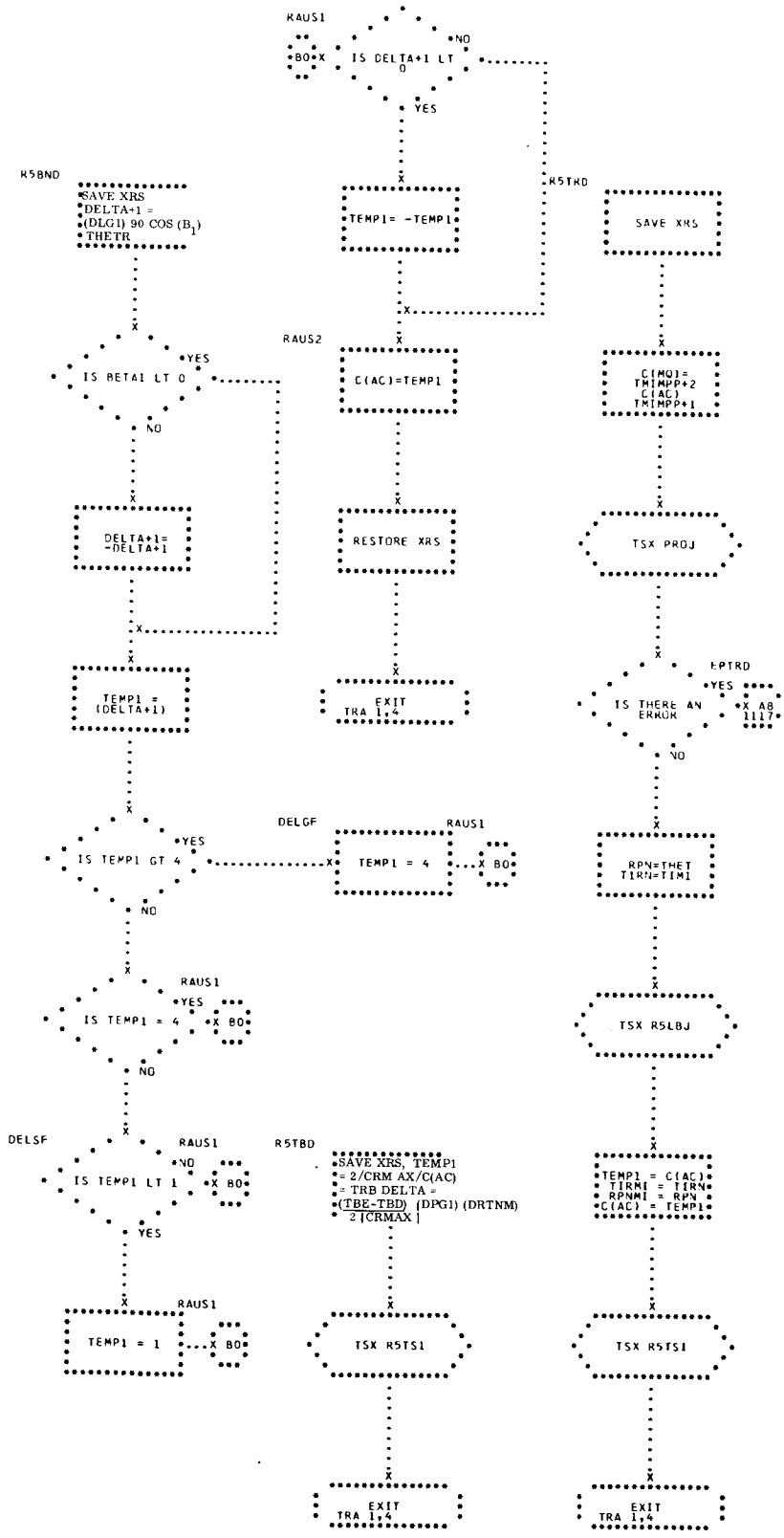

STOTM
A0
TIM2= TIM1+ DELTA
CTIRD
IS SWRF=0
YES
B2 1109
NO
C(AC)= /BANKLE/
C(MQ)= THETR
TSX GTRTM,4
TEMP1=C(AC)
TEMP2= TB0+TEMP1
IS RTM + TNREEP +2 GT TIM2
YES
NO
TIM2=R5TM +TNREEP+2-10 DELTA=TIM2 -TIM1
R5BET
IS TNREEP+2 TIM2
YES
NO
TIM2=TNREEP +2+10, DELTA= TIM2 - TIM1
R5CCK
A9
LCOP3
XR2 = 2
B0
LOOP3
B0
C(AC)= TIM2+2,2 C(MQ)= BETA2+2,2
TSX CLFPA
PHIG2+2,2 TMIMPP+1
LAMG2+2,2 TMIMPP+2
PPIG2+2,2 RSLAT
-PHIG2+2,2 C(AC) RSLNG - LAMG2+2,2
TSX LONC
DLG2+2,2= C(AC)
TSX CONVG
WAS THERE CONVG.
NO
C0
YES
WE CONVERGED
C(AC) = TIM2+2,2
C(MQ) = BETA 2+2,2
TAV
B9
RESTORE XRS
LOOP3
C0
DLDB + 2,2 (DLG2 - 2,2 - DLG1)/DELTA + 2,2 (∂λ/∂λ, ∂λ/∂λ)
PPDB-2,2 (DPIG2,2 - DPIG)/DELTA-2,2 (∂φ/∂λ, ∂φ/∂λ)
XR2=0 2ND TIME THRU
NO
B0
YES
TEMP1 (PLDB) (PPDB+1)
DETER (PPDB) PLDB+1)
- (PLDB) (PPDB + 1)
R5CDR
DETER = 0
YES
C6 1120
NO
TEMP1 (PLDB) (DIG1)
DELB1 (PPDBXDLG1)-(PLDB) (DIG1) / DETER
TEMP1 (PPDB+1) (OLG1)
DETIR (PLDB+1) (OPG1)-(PPDB+1) (DLG1) / DETER
BETA4 BETA1 - DELB1
TIR1 TIM1 - DETIR
C(AC) TIR4
C(MQ) BETA4
DLG1=C(AC) DPG1=R5LAT- TMIMPP+1 PCONT=PCONT+1
EXIT TRA 2,4
LTEB
IS TBE GT BETA2+1
YES
NO
BETA2+1= TBE+10 DELTA+1= BETA2+1-BETA1
R50K
R5CCK
IS TB0 LT BETA2+1
NO
A9
YES
BETA2+1= TB0+10, DELTA+1= BETA2+1-BETA1
R5CCK
A9
TSX CLFPA (GET IMPACT POINT 0,-)
R5LNG= TMIMPP+2 INTO AC
TSX LONC
LOOP4
C(AC)=TIR4 C(MQ)=BETA4
C6 1108

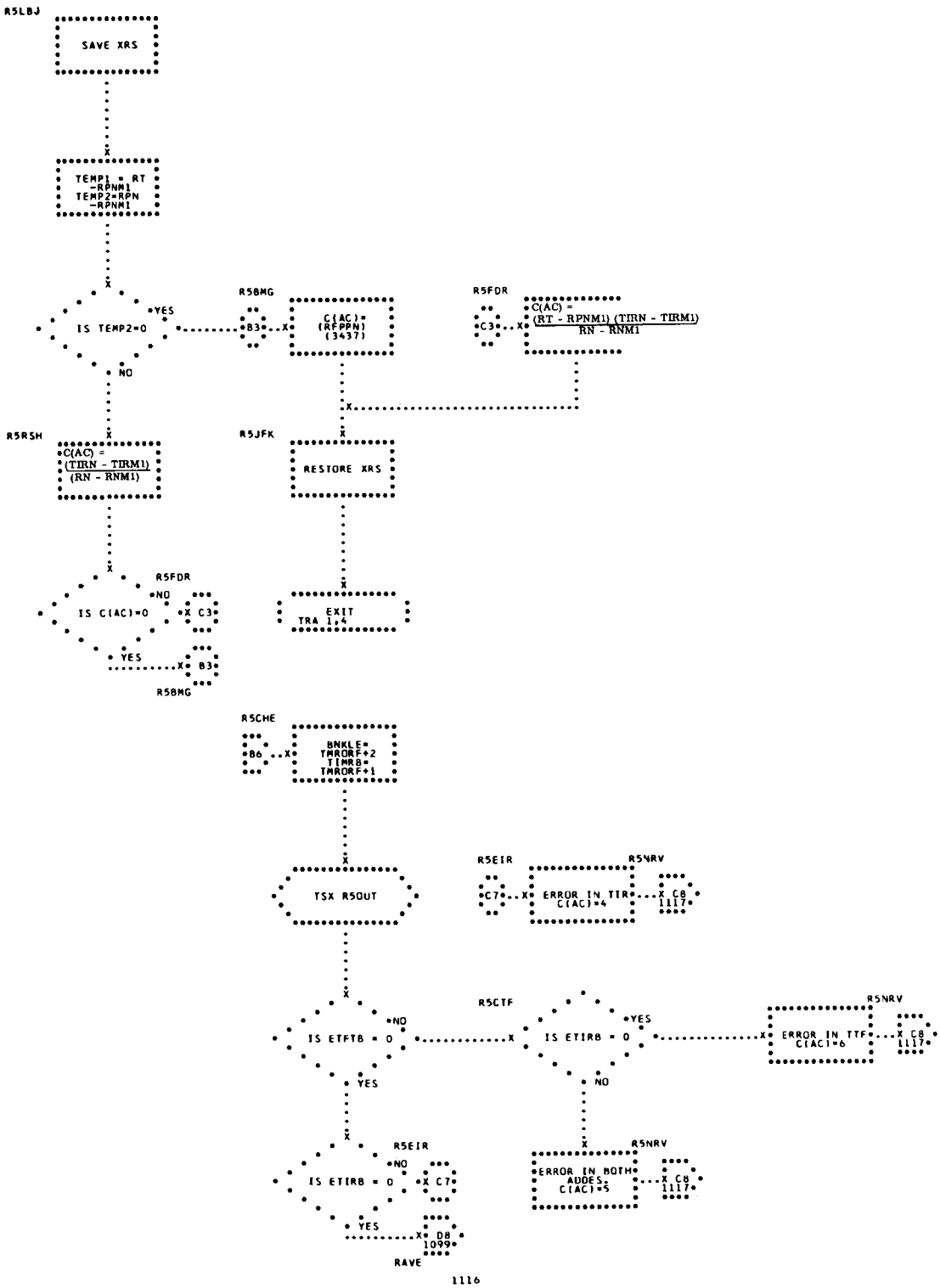

CLFPA
SAVE XRS
TIR4=C(AC)
BETA4=C(MQ)
IS SWRF LT 0
YES
NO
BTRB
TIMRB=TIR4
BNKLE=BETA4
GMTIR=K00INF
IS SWRF = 0
YES
NO
ROLLC
GMTIR=TIR4
BNKLE=BETA4
TIMRB=K00INF
TNRF = TIR4
TIMRB = BETA4
GMTIR=K00INF
MUV-2
TSX R5IMP
IINTEG,
IMPACT, 0, λ
TSX R5TWT
R5POR
IS THERE AN ERROR
YES
X C6 1120*
NO
MUV
IS THERE AN ERROR
NO
YES
X C6 1120*
R5POR
RESTORE XRS
EXIT
TRA 1,4
GTRTM
SAVE XRS
TSX DIN1,4
TEMP1=C(AC)
150 LT/= RTM, LT/= 600
NO
YES
LIMIT THEM NOW.
INRTM=RTM
NINER
RESTORE XRS
EXIT
TRA 1,4

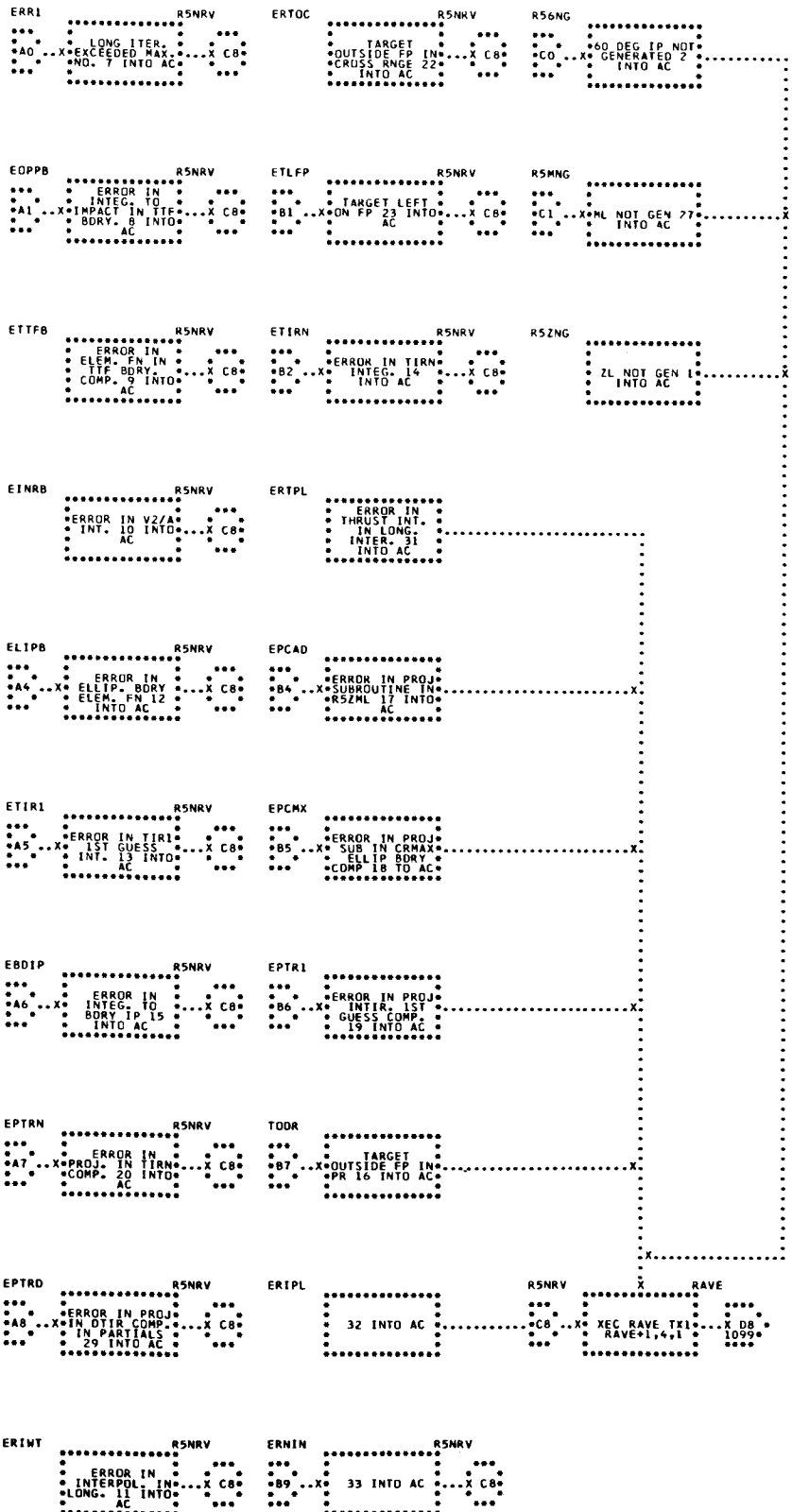

R5IWT
SAVE XRS
TMR50M+6 =TNRF R5NTP=TNRF
R5NIP PZE **, 0, **
PZE TNINTG,,TMR50M
PZE 1,,1
INTERPOLATE FOR VECTOR AT TNRF
TSX UINTP
ETERP
IS THERE AN ERROR
YES
DH
NO
TMDELT = 0
NO
YES
SET UP TMR50M
TABLE FOR NI
SET UP TTOAMS
TABLE FOR THRUSTER
TSX MPCPNI (B,0, VEC TRV0AM)
IS THERE AN ERROR
YES
NO
BO
SET UP TMR5FF TABLE FORNI THRU FREE FLIGHT
BO
TSX MPCPNI (INTEG, THRU FREE FLIGHT)
IS THERE AN ERROR
YES
NO
TRVRET+4, +2 =NTU TMBLST+5
EXIT R5KGP
R5LLM
TMR50M+5 INTO TMBLST+5, TMBLST+6 = TNRF
R5KGP
TMBLST, TMBLST+5 =TRVRET+4 TRVRET+9
EXTRA
UNUNS
TMBLST+13 =R5TRE
CO
ETHRS
SET UP ERROR RETURN
UNUNS
CO
TMBLST+12 MCPWGT
TRETRO+4 - TMBLST+6
TRETRO+5 TISCCH+25+TNRF
TRETRO+6 TRETRO+5 +FI
TRETRO+7 TRETRO+6 +FI
C(AC) - FIRING OUR. TO SCCH +21 C(MQ)
CANT ANGLES TO SCCH +16, C
(I) - THRUST LEVEL TO SCCH + 15 XR1 4
C2
TRETRO+1, 1 C(AC) TRETRO+2, 0
1 - C(MQ) TRETRO+12, 1 C(I)
XR1=0 4TH TIME THRU
NO
C2
YES
XR1 = 4
TRETRO+16, 1=TOSCCH+24, 1=ORIENT. ANGLES
XR1=0 4TH TIME THRU
NO
C6
YES
XR1 = 3
DO
EXIT TO TRA 1,4
DO
TRETRO+23, 1=TOSCCH +20, 1 PITCH, ROLL, YAW
XR1=0 3RD TIME THRU
NO
DO
YES
TRETRO+24, =TOSCCH+14 =MASSEXP.
RATE STL TRETRO+23 (THRUST IND.) C(AC)=R5N13
TSX MPCPNE
EKINT
IS THERE AN ERROR
YES
C8 1113
NO
EXIT GBOV
ETERP
D8
ERROR IN INTERPOLATING VECTOR
EXIT TO TRA 3,4

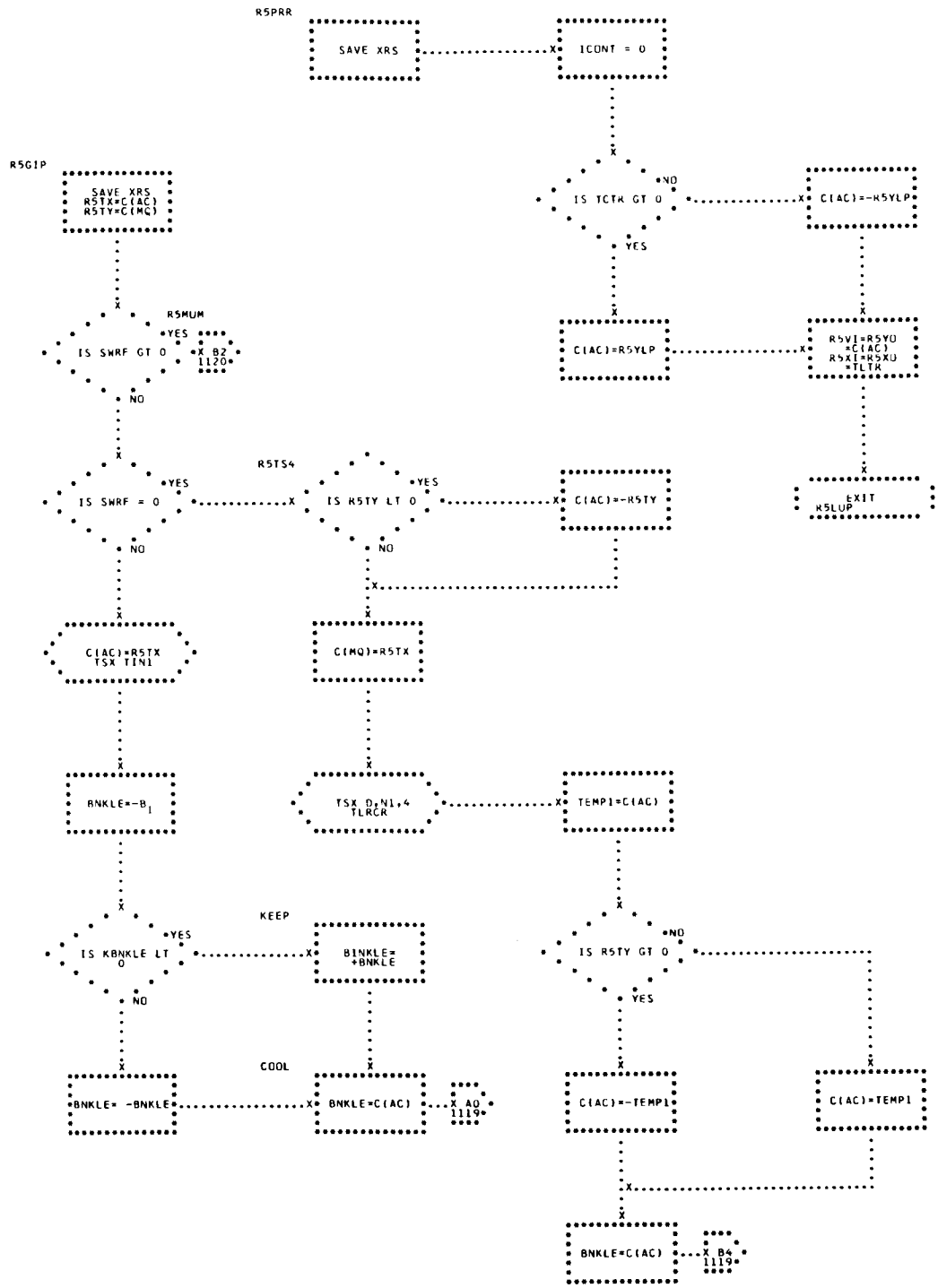

R5IMP
SAVE XRS
GBOV
STROR=0 IS THIS TTF
NO
YES
TMRARF+6=TRVIBO+3 (T OF BRNOUT) XR1 = 6
TMRARF+6,1 =TRVIBO+10,1
XR1=0 6TH TIME THRU
YES
NO
R5REN
XR1=7
TMRARF+7,1 =VECZD+7,1
XR1=0 7TH TIME THRU
YES
NO
TSX CHKBL
TMRARF + 15 = BNKLE
TMRARF + 13 = TIMRB
TMRARF + 14 = GMTIR
TMRARF + 12 = MC PORW - TOSCCH+13
C (AC) = R5NI 1
TSX MPCPNI
IS THERE AN ERROR
YES
NO
ERINT
XEC SAVE TXI SAVE+1,4,1
SAVE
TSX R5SPT,4
RESTORE XMS
EXIT TRA 2,4
INRVB
SAVE XRS
TMRARF + 15 = BNKLE
TMRARF+ 16 = D
TMRARF+ 14 = KOOINF
C (AC) = R5NI2
INTEG. TO D=LN (V---) ALTITUDE
TSX MPCPNI
IS THERE AN ERROR
YES
NO
C(AC)=D
TSX LAGIN (INTERPOLATE FOR TSTAR =TRBI)
PRES
TEMP2 = C (AC)
TEMP1 = 2 CRMAX
TEMP1 = (CRMAX+CRTMI)/(2 CRMAX)
TEMP3 = (TEB+30-TBB) (CRMAX+CRTME)/(2 CRMAX)
(BACK UP EQUATION FOR TIMRB 1ST GUESS)
R5XMS
GAUSS
C(AC)=-CRTMI
C(AC)=CRTMI
KBNKLE GT 0
NO
YES
ELAGN
IS THERE AN ERROR
YES
NO
R5XMS
TSTAR = C(AC)
ELAGN
IS SWRF GT 0
NO
YES

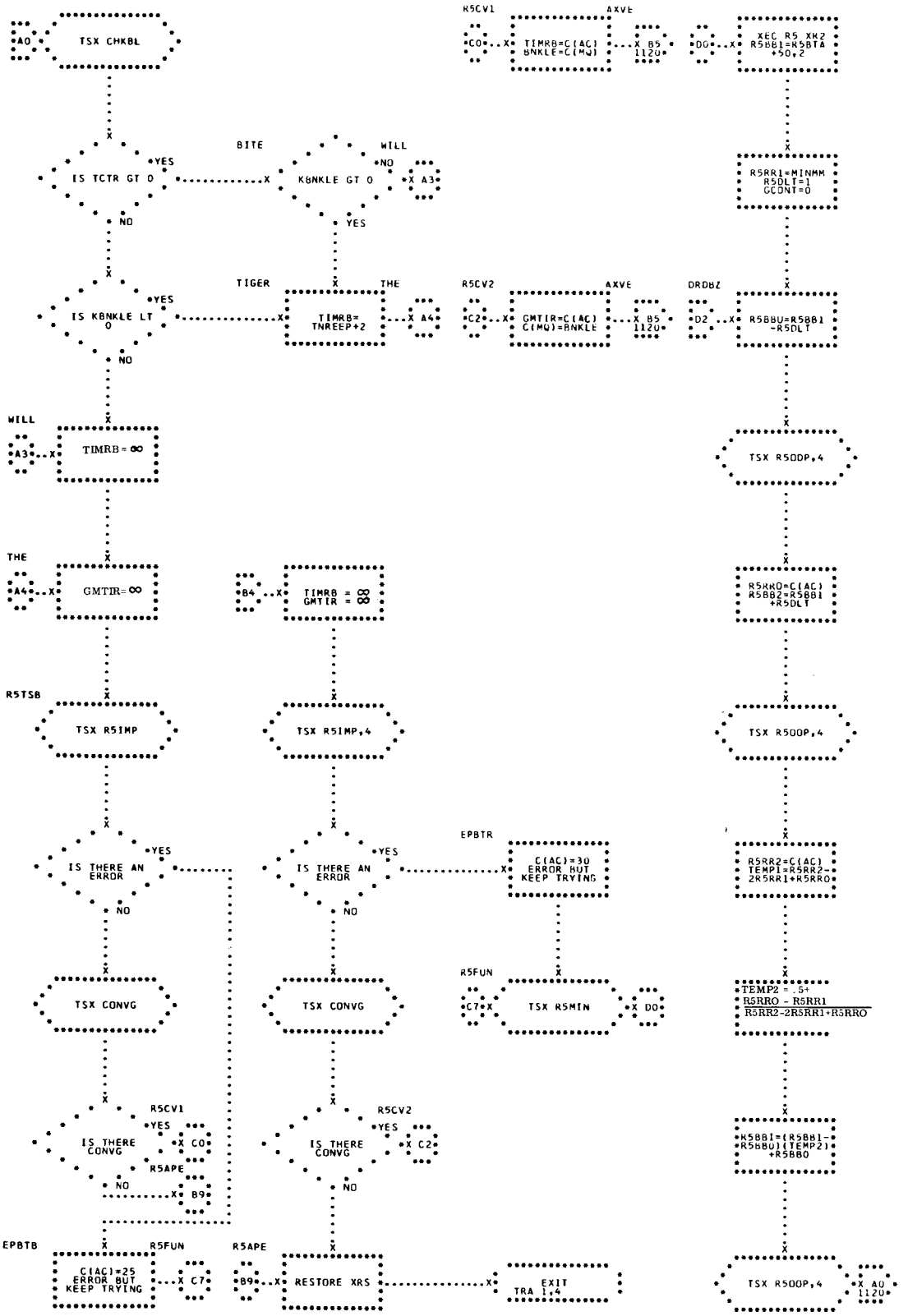

ELAGN
YES
DO
IS THERE AN ERROR
D5 1113
NO
TSX R5SPT
A1
C(AC)= TSTAR-60
RESTORE XRS
SCPA
NO
IS C(AC) LT TBO
IS SWRF LT 0
IS SWRF GT 0
YES
EXIT TRA 1,4
KATHY
LIMER
C(AC)= TBO+60
B7
C(AC)=/B1/ C(MQ)=THETR
IS TSTAR GT TEB
TSX GTRTM,4
C(AC)=TEB
XRTM1=C(AC). C(AC)=XRTM1 +TBO
MIKE
IS C(AC) GT/= TSTAR
C(AC)=TSTAR
TSTAR=C(AC) TMRARF+13= TSTAR
C(AC)=R5N11
TSX MPCPNI

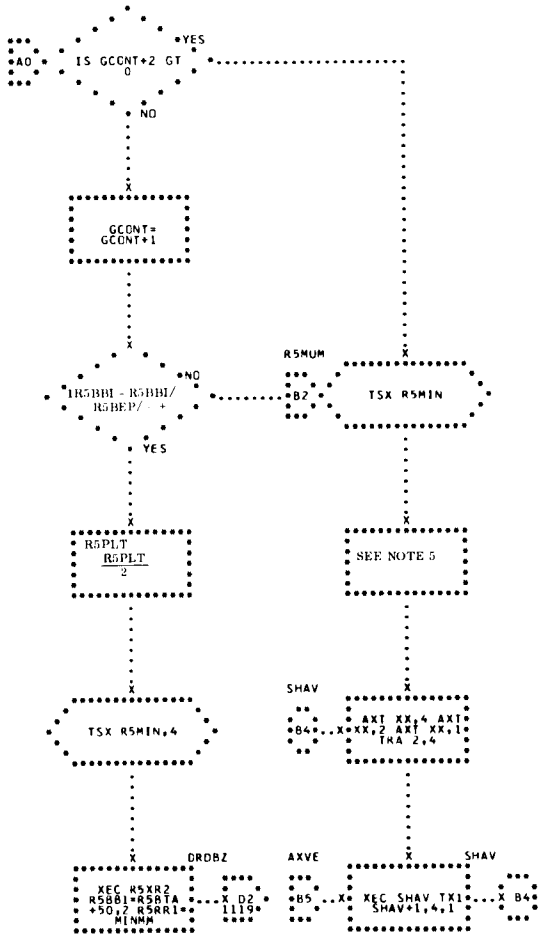

RAUS1
IS DELTA+1 LT 0
NO
BO
YES
R5BND
SAVE XRS
DELTA+1 =
(DLG1) 90 COS (B1)
THETR
TEMP1= -TEMP1
R5TRD
SAVE XRS
IS BETA1 LT 0
YES
NO
RAUS2
C(AC)=TEMP1
C(MQ)=
TMIMPP+2
C(AC)
TMIMPP+1
DELTA+1=
-DELTA+1
RESTORE XRS
TSX PROJ
TEMP1 =
(DELTA+1)
EXIT
TRA 1,4
EPTRD
IS THERE AN ERROR
YES
X A8
1117
NO
DELGF
RAUS1
IS TEMP1 GT 4
TEMP1 = 4
BO
RPN=THET
TIRN=TIM1
RAUS1
IS TEMP1 = 4
TSX R5LBJ
DELSF
RAUS1
IS TEMP1 LT 1
R5TBD
SAVE XRS, TEMP1
= 2/CRM AX/C(AC)
= TRB DELTA =
(TBE-TBD) (DPG1) (DRTNM)
2 |CRMAX|
TEMP1 = C(AC)
TIRN1 = TIRN
RPNM1 = RPN
C(AC) = TEMP1
TEMP1 = 1
TSX R5TS1
TSX R5TS1
EXIT
TRA 1,4
EXIT
TRA 1,4


R5LBJ
SAVE XRS
TEMP1 = RT -RPNM1
TEMP2=RPN -RPNM1
IS TEMP2=0
YES
NO
R5BMG
B3
C(AC)= (RPPN) (3437)
R5FDR
C3
C(AC) = (RT - RPNM1) (TIRN - TIRM1) / RN - RNM1
R5RSH
C(AC) = (TIRN - TIRM1) / (RN - RNM1)
R5JFK
RESTORE XRS
IS C(AC)=0
NO
YES
EXIT
TRA 1,4
R5CHE
B6
BNKLE= TMRORF+2
TIMRB= TMRORF+1
TSX R5OUT
R5EIR
C7
ERROR IN TIR C(AC)=4
R5NRV
C8 1117
IS ETFTB = 0
R5CTF
IS ETIRB = 0
ERROR IN TTF C(AC)=6
ERROR IN BOTH ADDES, C(AC)=5
D8 1099
RAVE

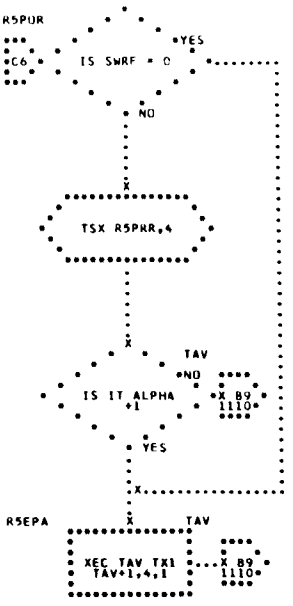

ERR1
R5NRV
LONG ITER. EXCEEDED MAX. NO. 7 INTO AC
A0
C8
ERTOC
R5NRV
TARGET OUTSIDE FP IN CROSS RNGE 22 INTO AC
C8
R56NG
60 DEG IP NOT GENERATED 2 INTO AC
C0
EOPPB
R5NRV
ERROR IN INTEG. TO IMPACT IN TTF BDRY. 8 INTO AC
A1
C8
ETLFP
R5NRV
TARGET LEFT ON FP 23 INTO AC
B1
C8
R5MNG
ML NOT GEN 27 INTO AC
C1
ETTF8
R5NRV
ERROR IN ELEM. FN IN TTF BDRY. COMP. 9 INTO AC
C8
ETIRN
R5NRV
ERROR IN TIRN INTEG. 14 INTO AC
B2
C8
R5ZNG
ZL NOT GEN 1 INTO AC
EINRB
R5NRV
ERROR IN V2/A INT. 10 INTO AC
C8
ERTPL
ERROR IN THRUST INT. IN LONG. INTER. 31 INTO AC
ELIPB
R5NRV
ERROR IN ELLIP. BDRY ELEM. FN 12 INTO AC
A4
C8
EPCAD
ERROR IN PROJ SUBROUTINE IN R5ZML 17 INTO AC
B4
ETIR1
R5NRV
ERROR IN TIR1 1ST GUESS INT. 13 INTO AC
A5
C8
EPCMX
ERROR IN PROJ SUB IN CRMAX ELLIP BDRY COMP 18 TO AC
B5
EBDIP
R5NRV
ERROR IN INTEG. TO BDRY IP 15 INTO AC
A6
C8
EPTR1
ERROR IN PROJ INTIR. 1ST GUESS COMP. 19 INTO AC
B6
EPTRN
R5NRV
ERROR IN PROJ. IN TIRN COMP. 20 INTO AC
A7
C8
TODR
TARGET OUTSIDE FP IN PR 16 INTO AC
B7
EPTRD
R5NRV
ERROR IN PROJ IN DTIR COMP. IN PARTIALS 29 INTO AC
A8
C8
ERIPL
32 INTO AC
R5NRV
C8
RAVE
XEC RAVE TX1 RAVE+1,4,1
X D8 1099
ERIWT
R5NRV
ERROR IN INTERPOL. IN LONG. 11 INTO AC
C8
ERNIN
R5NRV
33 INTO AC
B9
C8

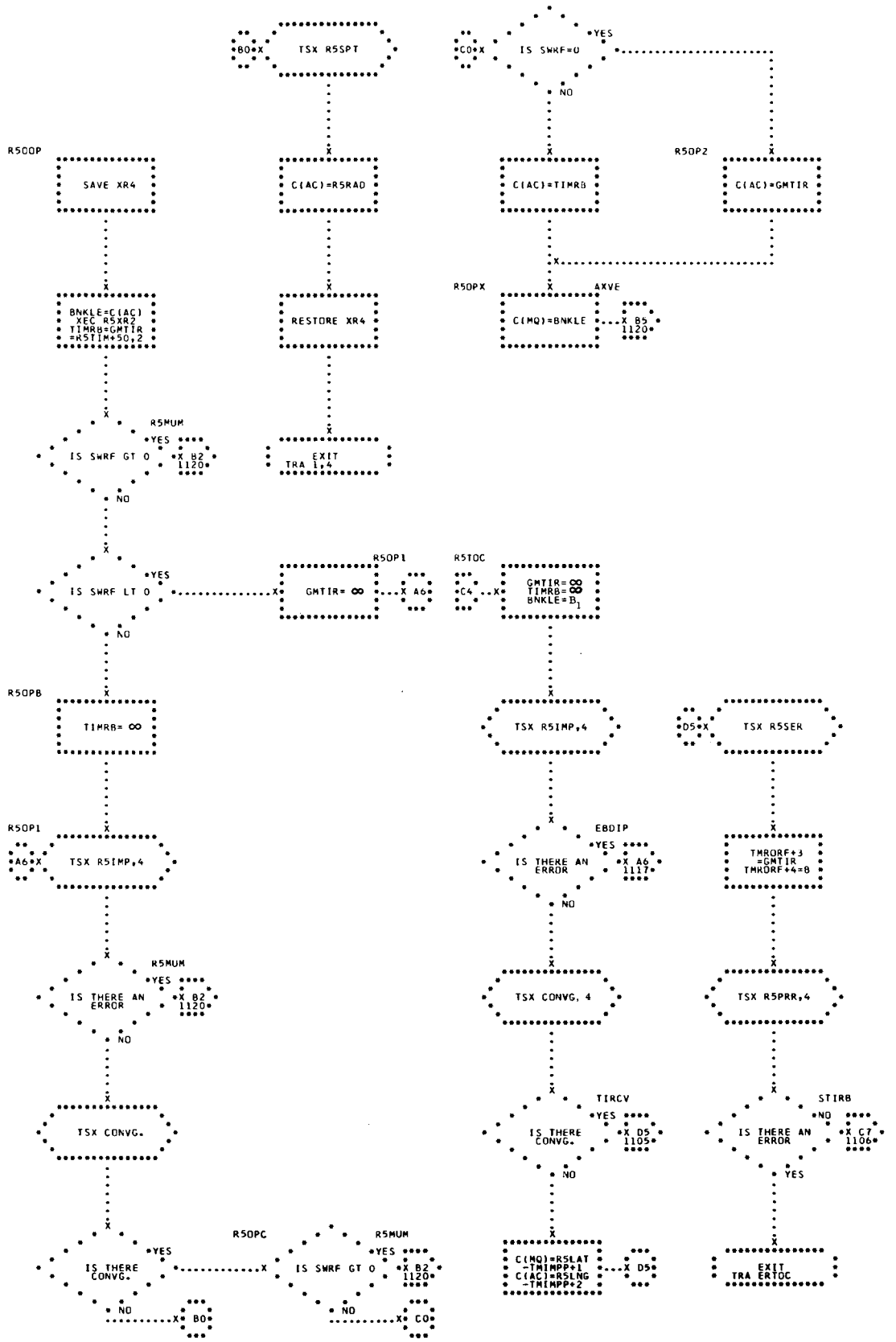

R5PRR
SAVE XRS
ICONT = 0
IS TCTR GT 0
NO
YES
C(AC)=-R5YLP
C(AC)=R5YLP
R5VI=R5YO
*C(AC)
R5XI=R5XO
+TLTR
EXIT
R5LUP
R5GIP
SAVE XRS
R5TX=C(AC)
R5TY=C(MQ)
R5MUM
IS SWRF GT 0
X BZ
1120
IS SWRF = 0
R5TS4
IS R5TY LT 0
C(AC)=-R5TY
C(AC)=R5TX
TSX TIN1
C(MQ)=R5TX
BNKLE=-B1
TSX D,N1,4
TLRCR
TEMP1=C(AC)
KEEP
IS KBNKLE LT
0
BINKLE=
+BNKLE
IS R5TY GT 0
COOL
*BNKLE= -BNKLE
BNKLE=C(AC)
X AO
1119
C(AC)=-TEMP1
C(AC)=TEMP1
BNKLE=C(AC)
X B4
1119

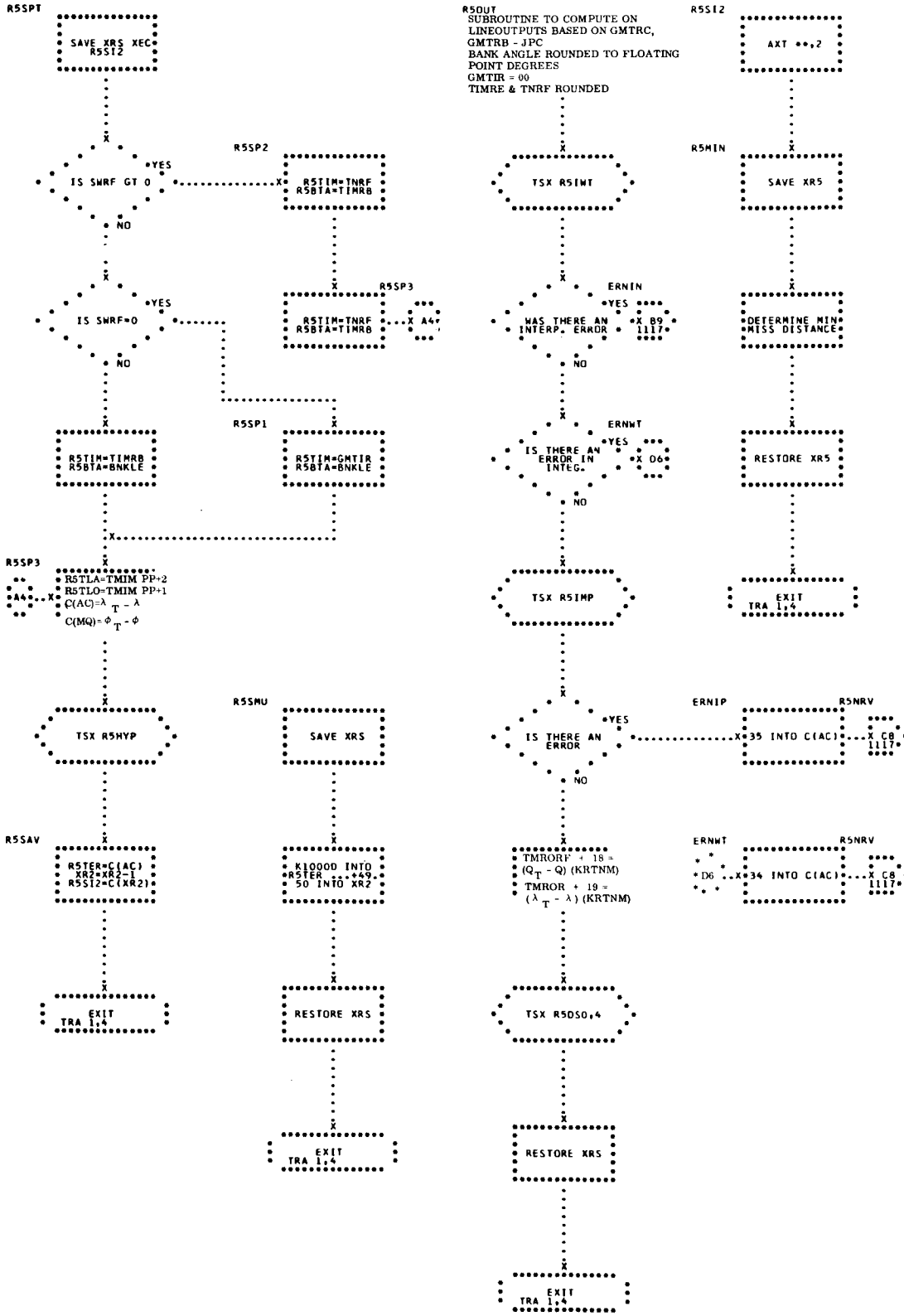

A0
TSX CHKBL
IS TCTR GT 0
YES
NO
BITE
WILL
KBNKLE GT 0
NO
YES
A3
IS KBNKLE LT 0
YES
NO
TIGER
THE
TIMRB= TNREEP+2
A4
WILL
A3
TIMRB = ∞
THE
A4
GMTIR= ∞
B4
TIMRB = ∞
GMTIR = ∞
R5TSB
TSX R5IMP
TSX R5IMP,4
IS THERE AN ERROR
YES
NO
IS THERE AN ERROR
YES
NO
TSX CONVG
TSX CONVG
R5CV1
IS THERE CONVG
YES
CO
R5APE
NO
B9
R5CV2
IS THERE CONVG
YES
C2
NO
EPBTB
R5FUN
C(AC)=25 ERROR BUT KEEP TRYING
C7
R5APE
B9
RESTORE XRS
EXIT TRA 1,4
R5CV1
AXVE
CO
TIMRB=C(AC) BNKLE=C(MQ)
B5 1120
DO
XEC R5 XR2 R5BB1=R5BTA +50,2
R5RR1=MINMM R5DLT=1 GCONT=0
R5CV2
AXVE
C2
GMTIR=C(AC) C(MQ)=BNKLE
B5 1120
DRDBZ
D2
R5BBU=R5BB1 -R5DLT
TSX R500P,4
R5RRO=C(AC) R5BB2=R5BB1 +R5DLT
TSX R500P,4
EPBTR
C(AC)=30 ERROR BUT KEEP TRYING
R5FUN
C7
TSX R5MIN
DO
R5RR2=C(AC) TEMP1=R5RR2-2R5RR1+R5RRO
TEMP2 = .5+ R5RRO - R5RR1 / R5RR2-2R5RR1+R5RRO
R5BB1=(R5BB1-R5BBU)(TEMP2) +R5BBO
TSX R500P,4
A0
1120

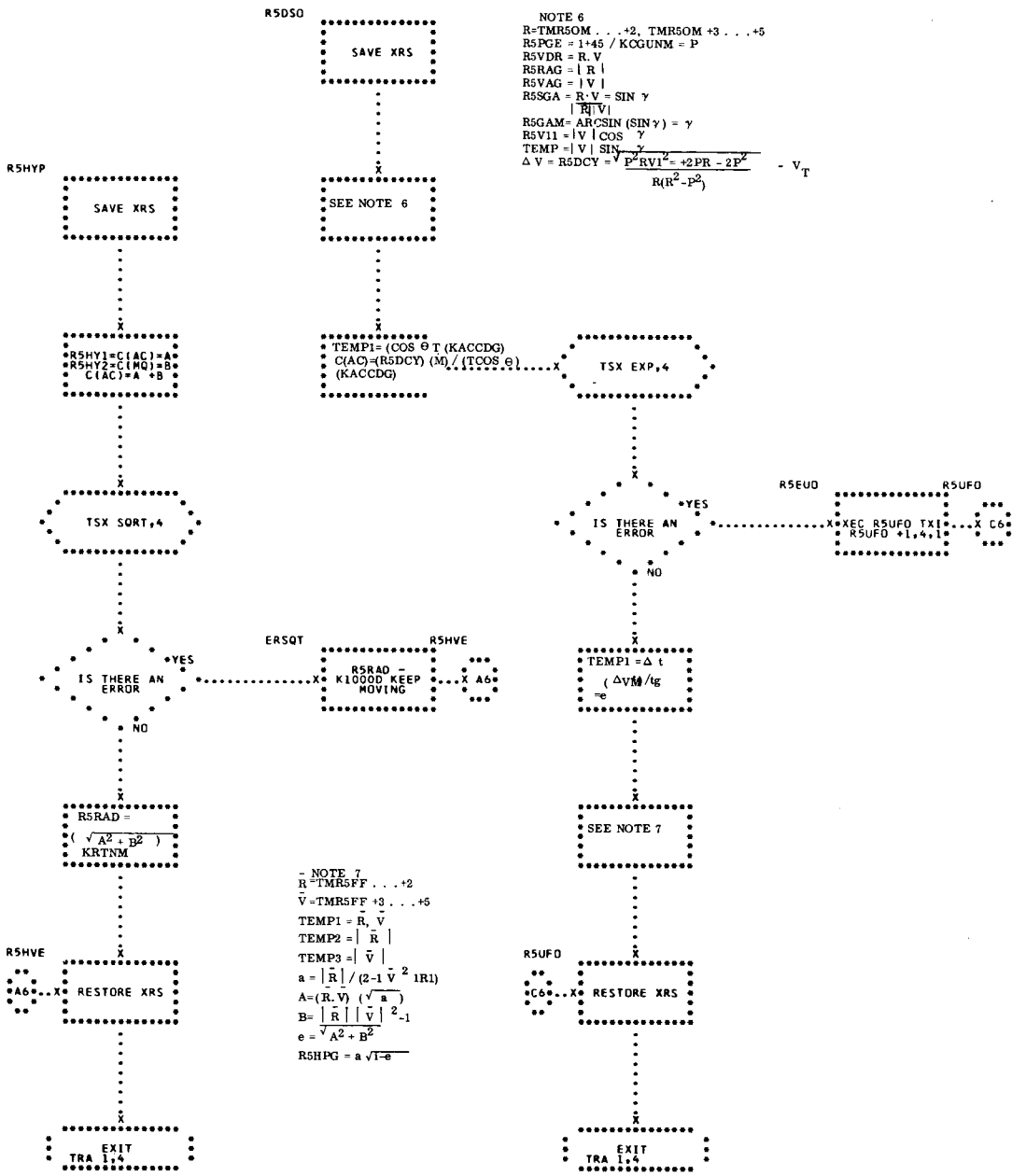


NOTE 5
XEC R5 x R2
TEMP3 = R5BTA+50, 2
TEMP1 = R5TER+50, 2
TEMP1 = R5TIM+50, 2
TMRORF+16 = R5TLA+50, 2
TMRORF+18 = (R5LAT-R5TLA+50, 2) (KRTNM)
TMRORF+17 = R5TLO+50, 2
TMRORF+19 = (R5LNG - R5TLO+50, 2) (KRTNM)




TSX R5SPT
B0
IS SWRF=0
YES
C0
NO
R50OP
SAVE XR4
C(AC)=R5RAD
C(AC)=TIMRB
R50P2
C(AC)=GMTIR
BNKLE=C(AC)
XEC R5XR2
TIMRB=GMTIR
=R5TIM+50,2
RESTORE XR4
R50PX
AXVE
C(MQ)=BNKLE
B5
1120
R5MUM
IS SWRF GT 0
B2
EXIT
TRA 1,4
R50P1
R5TOC
IS SWRF LT 0
GMTIR= ∞
A6
GMTIR= ∞
TIMRB= ∞
BNKLE=B1
C4
R50PB
TSX R5IMP,4
TSX R5SER
D5
EBDIP
IS THERE AN ERROR
TMRORF+3
=GMTIR
TMRORF+4=B
TSX CONVG, 4
TSX R5PRR,4
TIRCV
IS THERE CONVG.
STIRB
C7
1106
C(MQ)=R5LAT
-TMIMPP+1
C(AC)=R5LNG
-TMIMPP+2
EXIT
TRA ERTOC
TSX CONVG.
R50PC


R5SPT
SAVE XR5 XEC R5SI2
IS SWRF GT 0
YES
NO
R5SP2
R5TIM=TNRF
R5BTA=TIMRB
IS SWRF=0
R5SP3
R5TIM=TNRF
R5BTA=TIMRB
A47
R5SP1
R5TIM=TIMRB
R5BTA=BNKLE
R5TIM=GMTIR
R5BTA=BNKLE
R5SP3
A44
R5TLA=TMIM PP+2
R5TLO=TMIM PP+1
C(AC)=λT - λ
C(MQ)=ΦT - Φ
TSX R5HYP
R5SMU
SAVE XRS
R5SAV
R5TER=C(AC)
XR2=XR2-1
R5SI2=C(XR2)
K10000 INTO R5TER
50 INTO XR2
EXIT TRA 1,4
RESTORE XRS
EXIT TRA 1,4
R5OUT
SUBROUTINE TO COMPUTE ON LINEOUTPUTS BASED ON GMTRC,
GMTRB - JPC
BANK ANGLE ROUNDED TO FLOATING POINT DEGREES
GMTIR = 00
TIMRE & TNRF ROUNDED
TSX R5IWT
ERNIN
WAS THERE AN INTERP. ERROR
B9 1117
ERNWT
IS THERE AN ERROR IN INTEG.
D6
TSX R5IMP
ERNIP
IS THERE AN ERROR
35 INTO C(AC)
C8 1117
TMRORF + 18 = (QT - Q) (KRTNM)
TMROR + 19 = (λT - λ) (KRTNM)
ERNWT
D6
34 INTO C(AC)
C8 1117
TSX R5DS0,4
RESTORE XRS
EXIT TRA 1,4
R5SI2
AXT **,2
R5MIN
SAVE XR5
DETERMINE MIN MISS DISTANCE
RESTORE XR5
EXIT TRA 1,4
R5NRV
R5NRV


R5DSO
SAVE XRS
NOTE 6
R=TMR5OM . . .+2, TMR5OM +3 . . .+5
R5PGE = 1+45 / KCGUNM = P
R5VDR = R. V
R5RAG = | R |
R5VAG = | V |
R5SGA = R·V = SIN γ
| R | V |
R5GAM= ARCSIN (SIN γ) = γ
R5V11 = | V | COS γ
TEMP = | V | SIN γ
Δ V = R5DCY = √(P²RV1² +2PR - 2P²) / R(R²-P²) - V_T
R5HYP
SAVE XRS
SEE NOTE 6
R5HY1=C(AC)=A
R5HY2=C(MQ)=B
C(AC)=A +B
TEMP1= (COS Θ T (KACCDG)
C(AC)=(R5DCY) (M) / (TCOS Θ)
(KACCDG)
TSX EXP,4
TSX SQRT,4
IS THERE AN ERROR
YES
NO
R5EUO
R5UFO
XEC R5UFO TXI
R5UFO +1,4,1
X C6
ERSQT
R5HVE
R5RAD = K10000 KEEP MOVING
X A6
TEMP1 =Δ t
(ΔV M /tg
=e
R5RAD =
(√(A² + B²))
KRTNM
SEE NOTE 7
NOTE 7
R =TMR5FF . . .+2
V =TMR5FF +3 . . .+5
TEMP1 = R, V
TEMP2 = | R |
TEMP3 = | V |
a = | R | / (2-1 V ² 1R1)
A=(R. V) (√ a)
B= | R | | V | ²-1
e = √(A² + B²)
R5HPG = a √(1-e)
R5HVE
A6
RESTORE XRS
R5UFO
C6
RESTORE XRS
EXIT
TRA 1,4
EXIT
TRA 1,4

NOTE 8

$TEMP3 = B^2 = (R5CMN)^2$

$TEMP1 = (YI)^2 / B^2$

$TEMP4 = XI - .4$

$TEMP5 = (XI - .4)/.36$

$R5FI = [\ YI^2/B^2 + (XI - .4)\ /.36\ ] - 1$

$TEMP\ 1 = YI - TCTR$

$TEMP\ 1 = (TEMP1)\ (TEMP4)$

$TEMP\ 6 = (36)/TEMP\ 3$

$TEMP\ 2 = YI\ (TEMP6)$

$R5GI = (XI - .4)\ (YI - TCTR) + (A^2/B^2)\ YI\ (X^I - TCTR)$

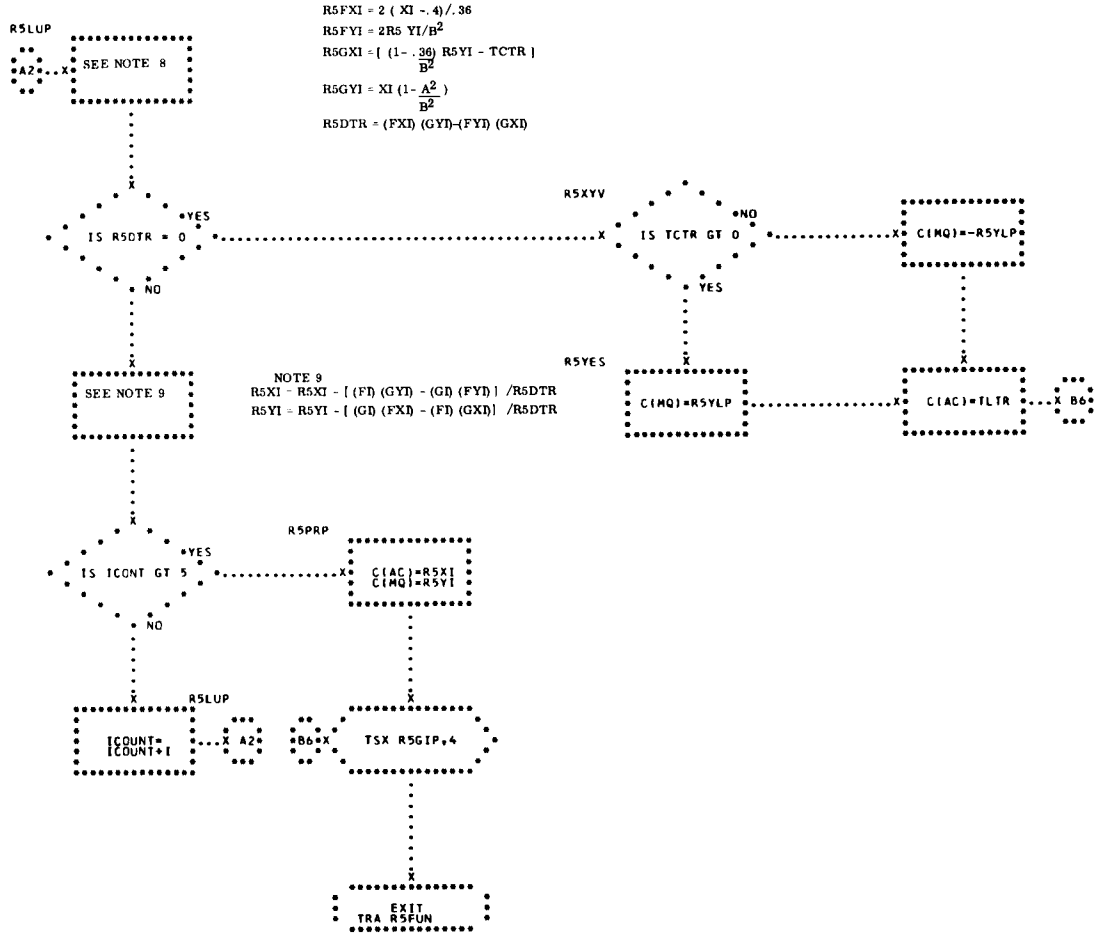
$R5FXI = 2\ (XI - .4)/.36$

$R5FYI = 2R5\ YI/B^2$

$R5GXI = [\ (1 - \frac{.36}{B^2})\ R5YI - TCTR\ ]$

$R5GYI = XI\ (1 - \frac{A^2}{B^2})$

$R5DTR = (FXI)\ (GYI) - (FYI)\ (GXI)$



62. AQDATA - SUBROUTINE TO GENERATE ACQUISITION DATA

POINTING DATA, FOR USE BY REMOTE STATIONS IN ACQUIRING A S/C, IS GENERATED BY THE SUBROUTINE AQDATA. GIVEN AN ESTIMATED TIME OF HORIZON CROSSING AND AN INTERNAL STATION NUMBER, AQDATA WILL DETERMINE IF THE SITE UNDER INTERROGATION IS ABLE TO OBSERVE THE S/C ON THIS STATION PASS. IF OBSERVATION IS NOT POSSIBLE, AN ERROR RETURN IS TAKEN. OTHERWISE, ACQUISITION DATA IS GENERATED.

THE DATA CONSISTS OF FIVE PARAMETERS FOR EACH DEFINING POINT, TIME, RANGE, AZIMUTH, ELEVATION AND DATA TAG. DEPENDING ON TYPE OF STATION PASS, ONE TO SIX SETS OF DEFINING PARAMETERS WILL BE GENERATED. AFTER CONSTRUCTING THE REQUIRED OUTPUT TABLES, THE SUBROUTINE NORMAL EXIT IS TAKEN. (SEE FLOWCHARTS.)

INPUT

A) TABLES -

TCRKO1	ORBIT EPHEMERIS GENERATED BY LAUNCH PROGRAM
TCRKO2	REENTRY EPHEMERIS GENERATED BY LAUNCH PROGRAM
TNDIAC	ORBIT EPHEMERIS GENERATED BY ORBIT PROGRAM
TNINT2	REENTRY EPHEMERIS GENERATED BY ORBIT PROGRAM
TRKO1H	CONTAINS TIME STEP OF TCRKO1
TRKR1H	CONTAINS TIME STEP OF TCRKO2
TMSTCH	CONTAINS BASE ADDRESSES OF STATION CHARACTERISTIC BLOCKS
TMANUV	S/C MANEUVER TABLE

TOBLK1 TIME OF BEGINNING OF BLACKOUT

TOBLK2 TIME OF ENDING OF BLACKOUT

B) COMMUNICATION CELLS -

MCACQ1 A (ESTIMATED TIME OF HORIZON CROSSING),,
D (INTERNAL STATION NUMBER)

MCPHSE PREFIX INDICATES MISSION PHASE

MCSUBR CORE EXECUTION INDICATOR

MCESAB TIME OF FIRST EVEN SECOND AFTER RETRO-
FIRE COMPLETION

MCMANC NUMBER OF MANEUVERS IN TMANUV

C) PARAMETERS -

MNLINS MAXIMUM NUMBER OF OUTPUT DATA LINES (PRESENTLY 6)

MNQANS NUMBER OF PARAMETERS PER DATA LINE (PRESENTLY 5)

OUTPUT

A) MCNOVL C(NUMBER OF OUTPUT DATA LINES),, A

B) TMSTAD

OUTPUT DATA BLOCK IN THE FOLLOWING FORMAT -

TMSTAD	TIME	A (FIXED POINT MINS.),,D (FIXED POINT SECS.)
+1	RANGE	FLOATING POINT KILOYARDS
+2	AZIMUTH	FLOATING POINT RADIANS
+3	ELEVATION	FLOATING POINT RADIANS
+4	DATA	TTY CODE
+5	TIME	A (FIXED POINT MINS.),, D (FIXED POINT SECS.)
•		•
•		•
•		•

+29	DATA TAG	TTY CODE

C) TMACQM

ON-LINE MESSAGE PRINT BLOCK IN THE FOLLOWING FORMAT

TMACQM	PASS TYPE	BCI
+1	PASS TYPE	BCI
+2	INITIAL HORIZON CORSSING	FIXED POINT SECS
+3	PASS DURATION	FIXED POINT SECS
+4	MAXIMUM ELEVATION	FLOATING POINT RADIANS

D) INTERPRETATION OF DATA TAGS IN TMSTAD -

A	ONE DEGREE IN ELEVATION
B	TEN DEGREES ALONG TRAJECTORY ARC
C	THIRTY DEGREES ALONG TRAJECTORY ARC
D	t_{ca} + TEN SECONDS
M	MARGINAL POINT
PM	PRE-POST MANEUVER POINT
OM	MANEUVER POINT
PR	PRE-REENTRY POINT
CB	COMMUNICATIONS BLACKOUT POINT
R	REENTRY POINT
LA	LAUNCH-ABORT POINT

METHOD

AQDATA IS USED DURING THREE MISSION PHASES, ORBIT, REENTRY AND HIGH ABORT. IN THE LATTER PHASE INTEGRATION TABLES SUPPLIED BY THE LAUNCH PROGRAM ARE USED, WHEREAS IN THE TWO FORMER PHASES, THE ORBIT OR REENTRY PROGRAM PROVIDES THE NECESSARY TABLES. HIGH-ABORT PHASE REQUIRES

THE REFORMATTING OF THE LAUNCH INTEGRATION TABLES TO CONFORM TO THE ORBIT AND REENTRY FORMAT, THIS IS DONE UPON ENTRY TO AQDATA.

USING THE INPUT ESTIMATED TIME (t_s), A SEARCH IS BEGUN TO DETERMINE INITIAL HORIZON CROSSING (ZERO DEGREES ELEVATION). IF ZERO DEGREES IS NOT REACHED WITHIN THE ALLOTTED TIME INTERVAL ($t_s \pm$ 15 MINUTES), A CHECK ON THE HIGHEST ELEVATION ACHIEVED DURING THIS INTERVAL IS MADE. IF IT IS GREATER THAN OR EQUAL TO -2° , THIS POINT IS SAVED AS THE SOLE PROGRAM OUTPUT, OTHERWISE THE ERROR EXIT (NO DATA) IS TAKEN.

AFTER ESTABLISHING FIRST HORIZON CORSSING (BY LINEAR INTERPOLATION, IF NECESSARY), AQDATA SEARCHES FOR THE TIME AT WHICH THE S/C IS 1° ABOVE THE HORIZON, COMPUTING t_1 BY LINEAR INTERPOLATION.

DURING THE SEARCH FOR 0° AND 1° , A CONTINUOUS CHECK IS MAINTAINED IN SELECTION OF THE PROPER INTEGRATION TABLE FOR SEARCH PURPOSES, WHETHER LAUNCH, ORBIT, MANEUVER OR REENTRY.

SECCND HORIZON CORSSING IS THEN DETERMINED AS A FUNCTION OF THE SINE OF THE ELEVATION ANGLE, SIN E = 0 CORRESPONDING TO 0° ELEVATION. AT THIS TIME THREE CONSECUTIVE VALUES OF RANGE, $\rho_a, \rho_b,$ AND ρ_c SATISFYING THE RELATICN $\rho_a \geq \rho_b \leq \rho_c$, ARE RETAINED FOR MAXIMUM ELEVATION CALCULATIONS.

ASSUMING A PARABOLIC FIT OF ρ AS A FUNCTION OF t, TIME OF CLOSET APPROACH (t_{CA}) IS COMPUTED.

$$t_{CA} = t_a - \frac{1}{2} \frac{4\rho_b - \rho_c - 3\rho_a}{\rho_a + \rho_c - 2\rho_b}$$

WHERE

t_a = TIME ASSOCIATED WITH ρ_a

NEXT, TIME SELECTION IS DETERMINED AS A FUNCTION OF STATION PASS TYPE -

A) NOMINAL - FREE FLIGHT ORBITAL TRAJECTORY FOR ENTIRE STATION PASS. (4 TIME POINTS)

1. t_1 - TIME OF 1 DEGREE IN ELEVATION
2. t_2 - TIME OF 10 DEGREES ALONG TRAJECTORY ARC
3. t_3 - TIME OF 30 DEGREES ALONG TRAJECTORY ARC
4. t_4 - TIME OF CLOSEST APPROACH + 10 SECONDS

t_2 AND t_3 ARE CALCULATED THRU THE USE OF A SECOND ORDER CURVE FIT AS FOLLOWS -

$$M = \rho_{CA} \cos {}^{\prime}E_{CA}$$

WHERE

ρ_{CA} AND E_{CA} ARE THE VALUES OF RANGE AND ELEVATION, RESPECTIVELY, AT t_{CA}.

$$t_2 = t_\phi + (101.6 - .092M)$$

$$t_3 = t_0 + (167.1 - .090M)$$

WHERE t_0

= TIME OF INITIAL HORIZON CORSSING

B) MANEUVER - DURING SOME PORTION OF THE STATION PASS, THE S/C IS ENGAGED IN A POWERED FLIGHT MANEUVER. (6 TIME POINTS)

1. t_1 - TIME OF 1 DEGREE IN ELEVATION
2. t_2 - TIME OF 1/6 OF STATION PASS DURATION
3. t_3 - TIME OF 2/6 OF STATION PASS DURATION
4. t_4 - TIME OF 3/6 OF STATION PASS DURATION
5. t_5 - TIME OF 4/6 OF STATION PASS DURATION
6. t_6 - TIME OF 5/6 OF STATION PASS DURATION

C) RETRO-FIRE - THE SITE OBSERVES THE RETRO-FIRE MANEUVER OF THE S/C. SIX TIME POINTS ARE SELECTED AS IN MANEUVER.

D) COMMUNICATIONS BLACKOUT -A TRACKING SITE SEES A PORTION OF REENTRY COMMUNICATIONS BLACKOUT. SIX TIME POINTS ARE SELECTED AS IN MANEUVER.

E) REENTRY - AS USED HERE, A REENTRY PASS IS DEFINED AS ONE EXCLUSIVE OF COMMUNICATIONS BLAKCOUT. FOUR TIME POINTS ARE SELECTED AS FOLLOWS -

1. t_1 - TIME OF 1 DEGREE IN ELEVATION
2. t_2 - TIME OF 1/4 OF STATION PASS DURATION
3. t_3 - TIME OF 2/4 OF STATION PASS DURATION
4. t_4 - TIME OF 3/4 OF STATION PASS DURATION

F) LAUNCH-ABORT - THE S/C TRAJECTORY IS REPRESENTED BY THE HIGH-ABORT DEFINING VECTOR. TIME POINTS ARE SELECTED IN 30-SECOND INCREMENTS OR FOLLOW THE REENTRY METHOD OF TIME SELECTION.

G) MARGINAL - IF THE S/C ELEVATION IS SUCH THAT, $-2^{\circ} \leq E < 1^{\circ}$ ONE SET OF DATA REFLECTING THE HIGHEST ELEVATION FALLING WITHIN THESE LIMITS WILL BE TRANSMITTED.

ONCE THE TIMES HAVE BEEN SELECTED, CONSTRUCTION OF THE OUTPUT BUFFER, TMSTAD, IS INITIATED. THE RANGE, AZIMUTH, ELEVATION AND DATA TAG ASSOCIATED WITH EACH TIME POINT ARE CALCULATED AND PLACED INTO THE BUFFER, WHEREUPON THE NORMAL SUBROUTINE EXIT IS TAKEN.

A SUBROUTINE IS INCLUDED WITHIN AQDATA TO COMPUTE RANGE AND SINE ELEVATION. THOUGH UA1LSC CALCULATES R, A

AND E, THE INTERNAL SUBROUTINE IS FASTER AND THUS MORE DESIRABLE. EQUATIONS USED ARE AS FOLLOWS -

A) λ' = INERTIAL LONGITUDE OF STATICN AT TIME t

$\lambda' = \lambda'_0 + .004375269\ t$

B) R = INERTIAL POSITICN OF STATION AT TIME t

$R_x = R_s \quad COS\phi' COS\ \lambda$

$R_y = R_s \quad COS\phi' SIN\ \lambda$

$R_z = R_s \quad SIN\ \phi'$

WHERE

ϕ = STATICN GEOCENTRIC LATITUDE

R_5 = RADIUS OF EARTH AT STATION

C) N = LOCAL POSITION OF STATION

N_x = CCS ϕ COS λ

N_y = CCS ϕ SIN λ

N_z = SIN ϕ

WHERE

ϕ = STATICN GEODETIC LATITUDE

D) $\rho = \bar{R} - R$

WHERE R = INERTIAL POSITION OF S/C

E) $\rho = |\rho|$

$$SIN\ E = \frac{N \cdot \rho}{|\rho|}$$

INTERNAL ERROR RETURNS ARE SUBSEQUENTLY ROUTED TO ONE COMMON LOCATION, RREA. A CHECK IS MADE TO SEE IF ANY VALID TIME POINTS HAVE BEEN RECORDED, IF SO, TMSTAD IS CONSTRUCTED USING SUCH TIMES. IF NOT, THE SUBROUTINE ERROR EXIT IS TAKEN. SOME OF THE MORE COMMCN ERROR CONDITIONS ARE -

A) LIBRARY SUBROUTINE ERROR - TRA RREA

B) REENTRY TABLE IMPACTS - USE LAST VECTOR IN TABLE

AS TIME OF SECOND
HORIZON CORSSING

C) INSUFFICIENT INTEGRATION TABLE - TRA RREA

D) NEGATIVE ELEVATION CCMPUTED (OTHER THAN FOR A MARGINAL PASS) IN CONSTRUCTING TMSTAD - SKIP TO NEXT TIME POINT

USAGE

A) CALLING SEQUENCE -

α TSX AQDATA,4

α + 1 TRA ERROR RETURN (NO DATA)

α + 2 NORMAL RETURN (DATA)

B) COMMUNICATION CELLS -

MCSUBR
MCACQ1
MCESAB
MCMANC
MCPHSE
MCNOVL

C) TABLES -

TCRKO1
TCRKO2
TRKO1H
TRKR1H
TMSTCH
TOBLK1
TOBLK2
TMANUV
TMSTAD
TNINT2
TMACQM

D) CONSTANTS -

KCH233 K0060.
K00001 K00060
K001.0 KM0001
K.SM2D KMUYDS
K00000 KRTDEG
K0.S1D KD0060
K004.0 K.RTER
K003.0 K00003
K000.5 K00030
K002.0 K00006
K00004

E) PARAMETERS -

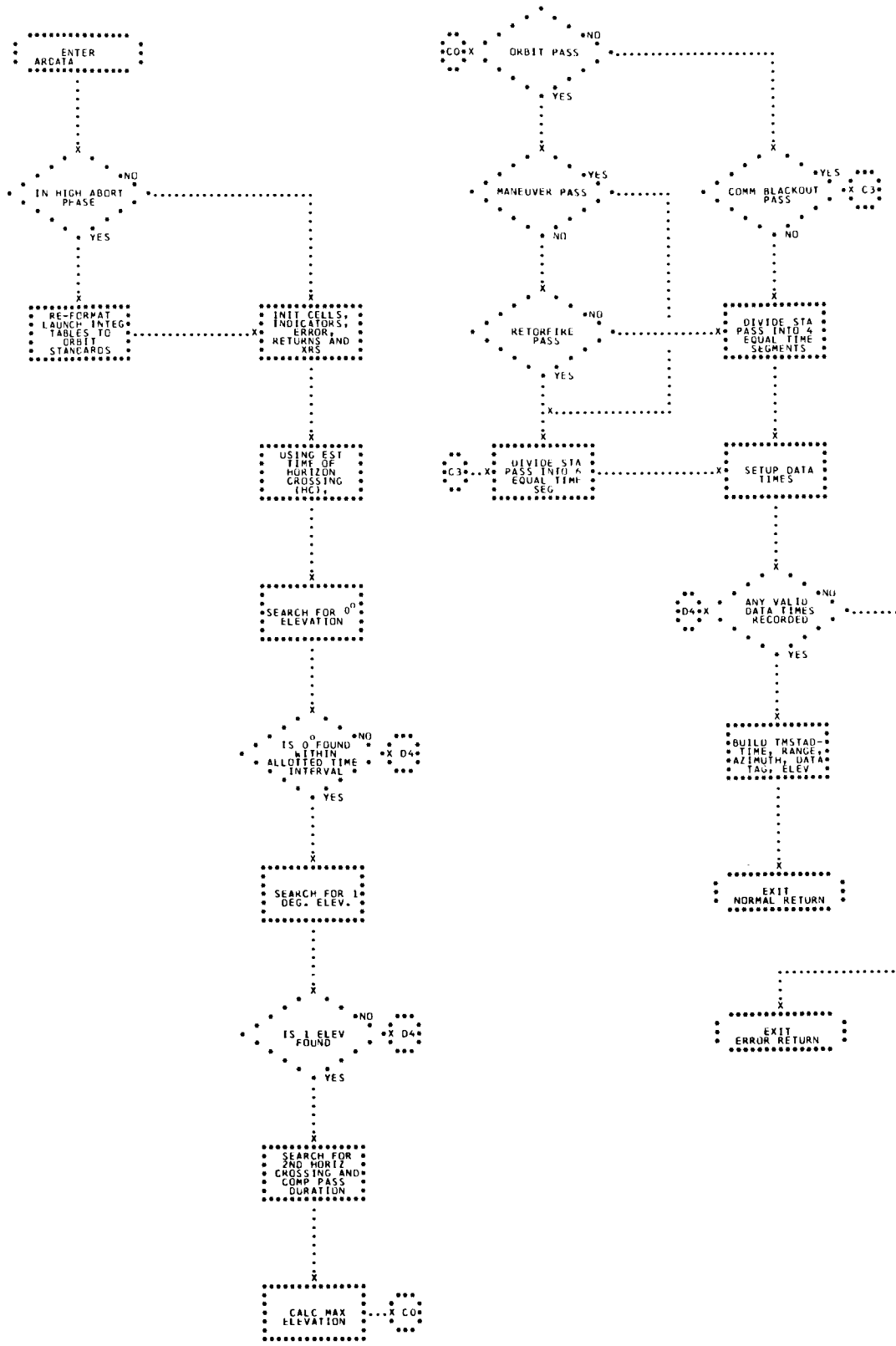
MNQANS
MNLINS

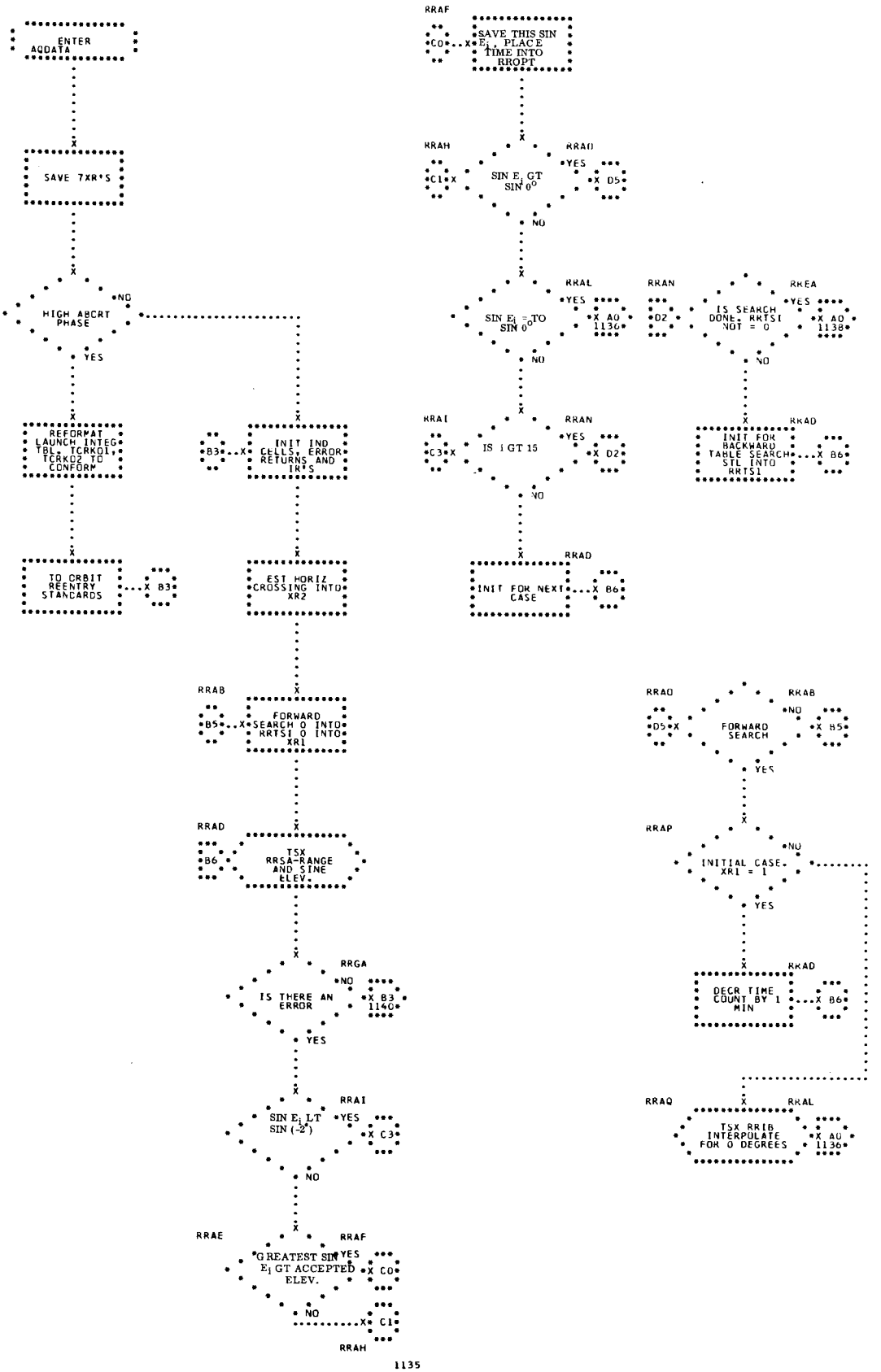
F) SUBROUTINES -

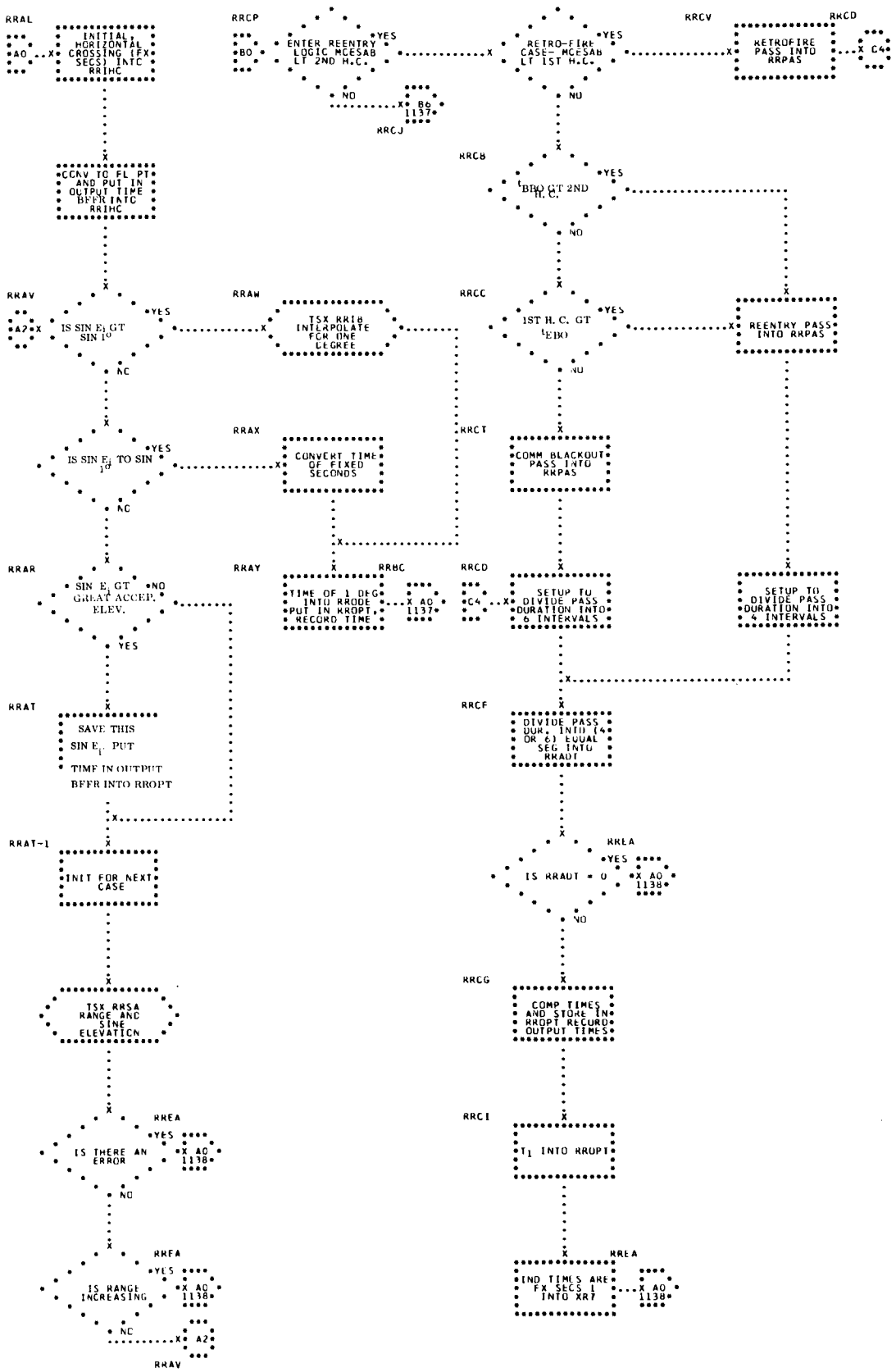
GINTAB
GLINTS
U7INTP
UA1LSC
ARSIN
SQRT
U3DOT
COS
SIN
VMAG

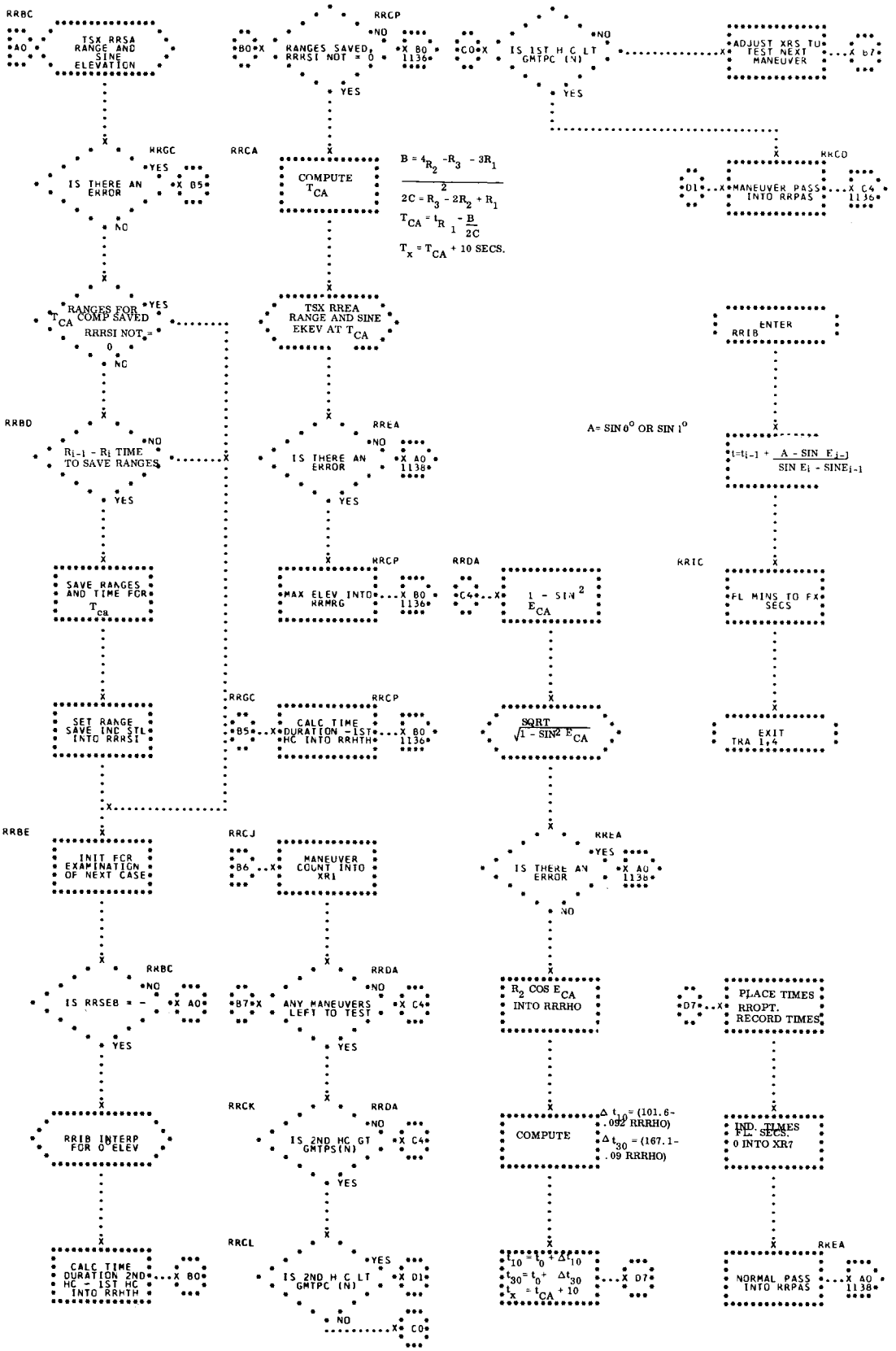
G) STORAGE REQUIRED -

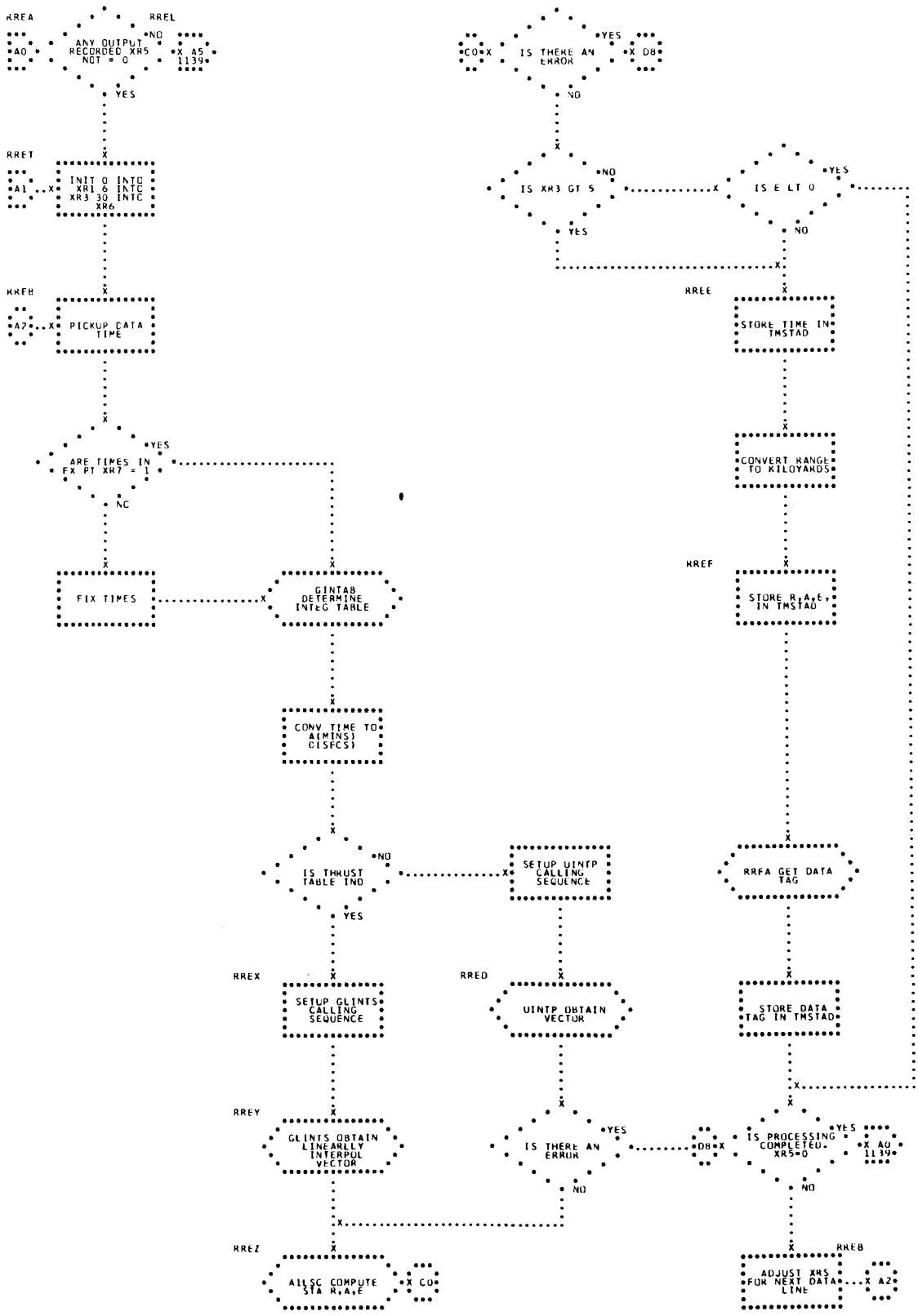
775 LOCATIONS

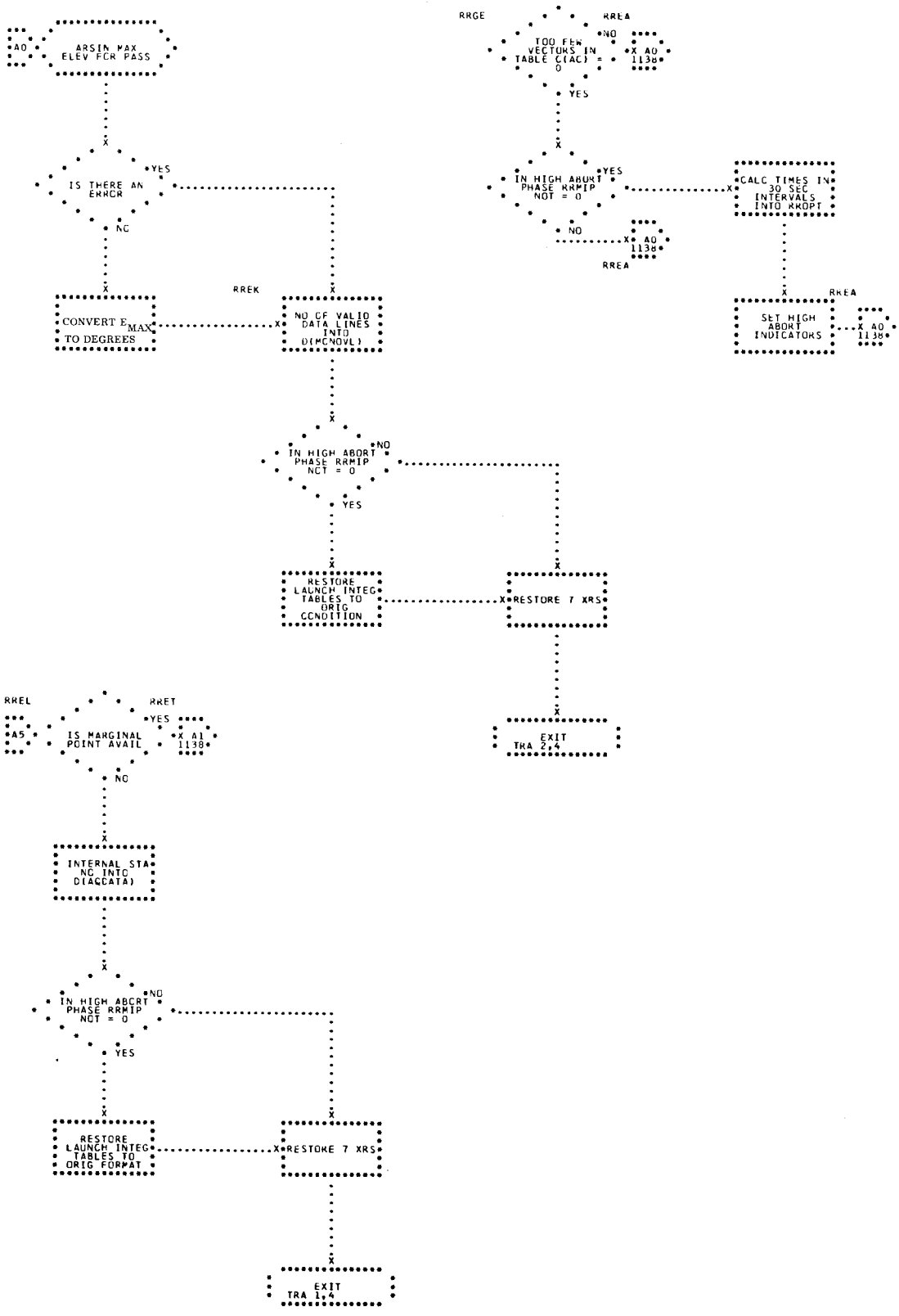

ENTER ARCATA
IN HIGH ABORT PHASE
NO
YES
RE-FORMAT LAUNCH INTEG TABLES TO ORBIT STANDARDS
INIT CELLS, INDICATORS, ERROR, RETURNS AND XRS
USING EST TIME OF HORIZON CROSSING (HC),
SEARCH FOR 0° ELEVATION
IS 0° FOUND WITHIN ALLOTTED TIME INTERVAL
NO
YES
D4
SEARCH FOR 1 DEG. ELEV.
IS 1 ELEV FOUND
NO
YES
D4
SEARCH FOR 2ND HORIZ CROSSING AND COMP PASS DURATION
CALC MAX ELEVATION
CO
ORBIT PASS
NO
YES
CO
MANEUVER PASS
YES
NO
RETOFIRE PASS
NO
YES
C3
DIVIDE STA PASS INTO 4 EQUAL TIME SEG
COMM BLACKOUT PASS
YES
NO
C3
DIVIDE STA PASS INTO 4 EQUAL TIME SEGMENTS
SETUP DATA TIMES
ANY VALID DATA TIMES RECORDED
NO
YES
D4
BUILD TMSTAD- TIME, RANGE, AZIMUTH, DATA TAG, ELEV
EXIT NORMAL RETURN
EXIT ERROR RETURN

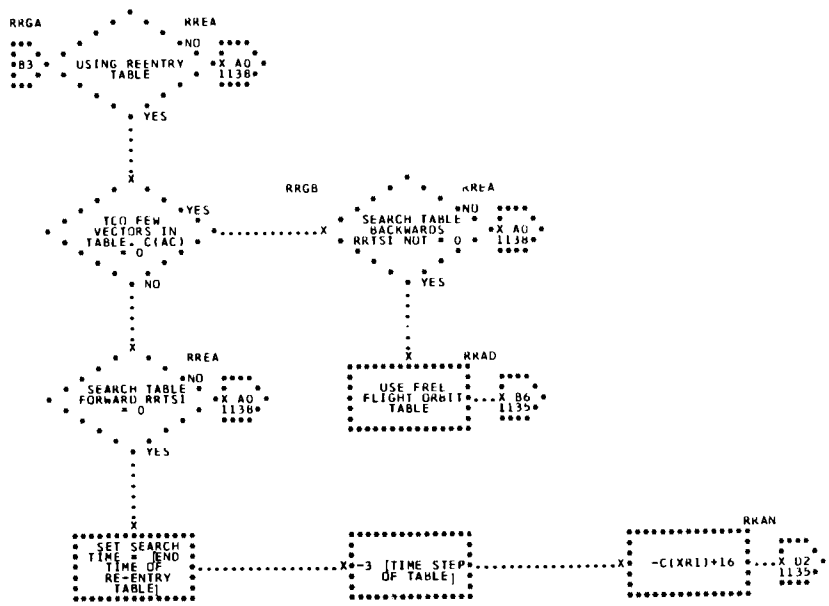

ENTER AQDATA
SAVE 7XR'S
HIGH ABCRT PHASE
NO
YES
REFORMAT LAUNCH INTEG TBL, TCRKOI, TCRKO2 TO CONFORM
TO ORBIT REENTRY STANDARDS
B3
INIT IND CELLS, ERROR RETURNS AND IR'S
EST HORIZ CROSSING INTO XR2
RRAB
FORWARD SEARCH 0 INTO RRTSI 0 INTO XR1
RRAD
TSX RRSA-RANGE AND SINE ELEV.
RRGA
IS THERE AN ERROR
B3 1140
RRAI
SIN Ei LT SIN (-2°)
C3
RRAE
GREATEST SIN Ei GT ACCEPTED ELEV.
RRAF
C0
C1
RRAH
SAVE THIS SIN Ei, PLACE TIME INTO RROPT
RRAO
SIN Ei GT SIN 0°
D5
RRAL
SIN Ei = TO SIN 0°
A0 1136
RRAN
IS i GT 15
D2
INIT FOR NEXT CASE
B6
RREA
IS SEARCH DONE, RRTSI NOT = 0
A0 1138
RRAD
INIT FOR BACKWARD TABLE SEARCH STL INTO RRTS1
RRAO
FORWARD SEARCH
B5
RRAP
INITIAL CASE, XR1 = 1
DECR TIME COUNT BY 1 MIN
RRAQ
TSX RRIB INTERPOLATE FOR 0 DEGREES
A0 1136

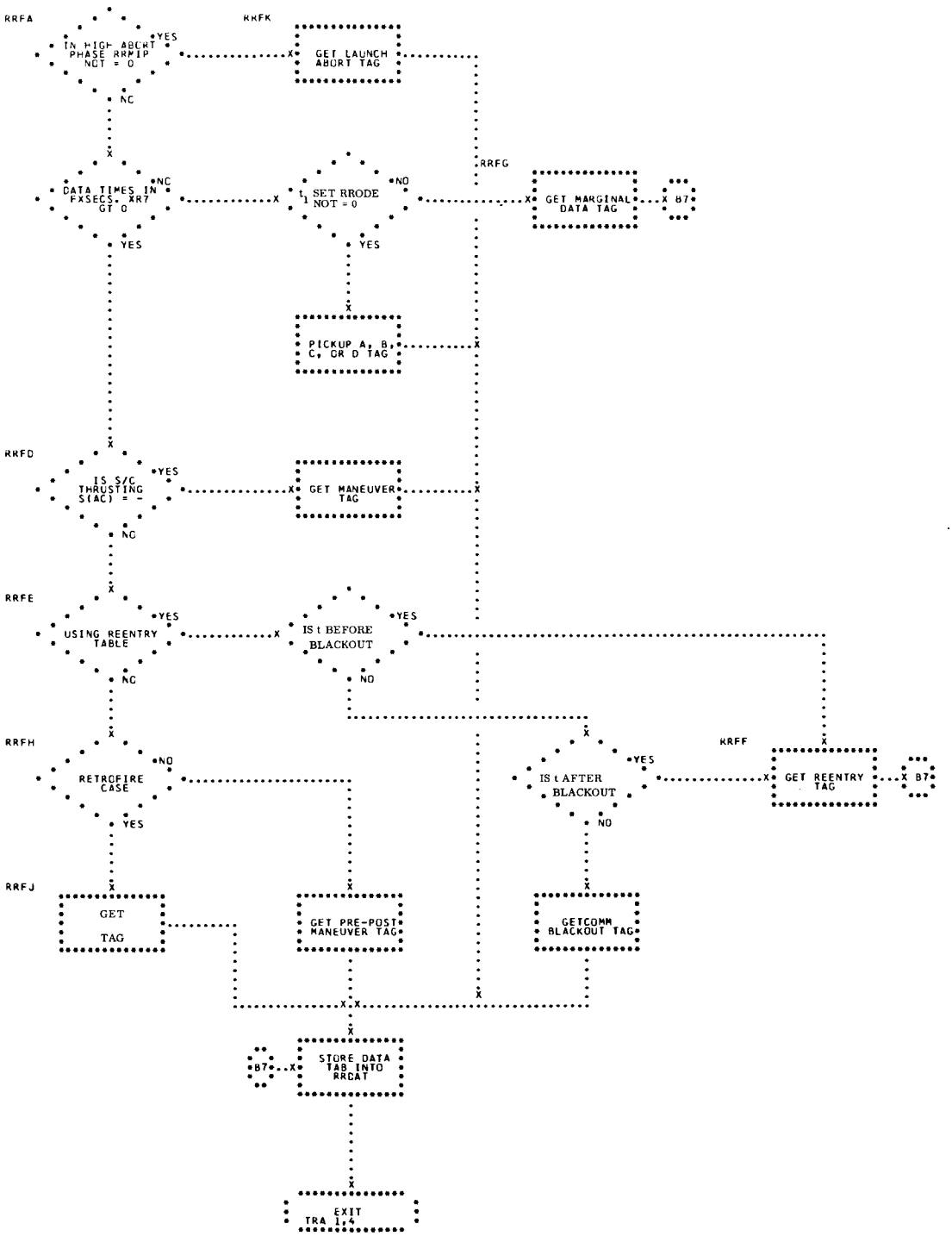

RRAL
INITIAL HORIZONTAL CROSSING (FX SECS) INTO RRIHC
A0
CONV TO FL PT AND PUT IN OUTPUT TIME BEFR INTO RRIHC
RRAV
IS SIN E1 GT SIN 1°
A2
YES
NO
RRAW
TSX RRIB INTERPOLATE FOR ONE DEGREE
RRAX
IS SIN E1 TO SIN 1°
CONVERT TIME OF FIXED SECONDS
RRAR
SIN E1 GT GREAT ACCEP. ELEV.
RRAY
TIME OF 1 DEG INTO RRODE PUT IN RROPT RECORD TIME
RRBC
B6 1137
RRAT
SAVE THIS SIN E1 PUT TIME IN OUTPUT BEFR INTO RROPT
RRAT-1
INIT FOR NEXT CASE
TSX RRSA RANGE AND SINE ELEVATION
RREA
IS THERE AN ERROR
IS RANGE INCREASING
A0 1138
RRFA
RRCP
ENTER REENTRY LOGIC MCESAB LT 2ND H.C.
B0
RRCJ
RETRO-FIRE CASE- MCESAB LT 1ST H.C.
RRCV
RETROFIRE PASS INTO RRPAS
RRCD
C4
RRCB
BEO GT 2ND H.C.
RRCC
1ST H.C. GT BEO
REENTRY PASS INTO RRPAS
RRCT
COMM BLACKOUT PASS INTO RRPAS
RRCD
SETUP TO DIVIDE PASS DURATION INTO 6 INTERVALS
SETUP TO DIVIDE PASS DURATION INTO 4 INTERVALS
RRCF
DIVIDE PASS DUR. INTO (4 OR 6) EQUAL SEG INTO RRADT
RREA
IS RRADT = 0
RRCG
COMP TIMES AND STORE IN RROPT RECORD OUTPUT TIMES
RRCI
T1 INTO RROPT
RREA
IND TIMES ARE FX SECS I INTO XR7

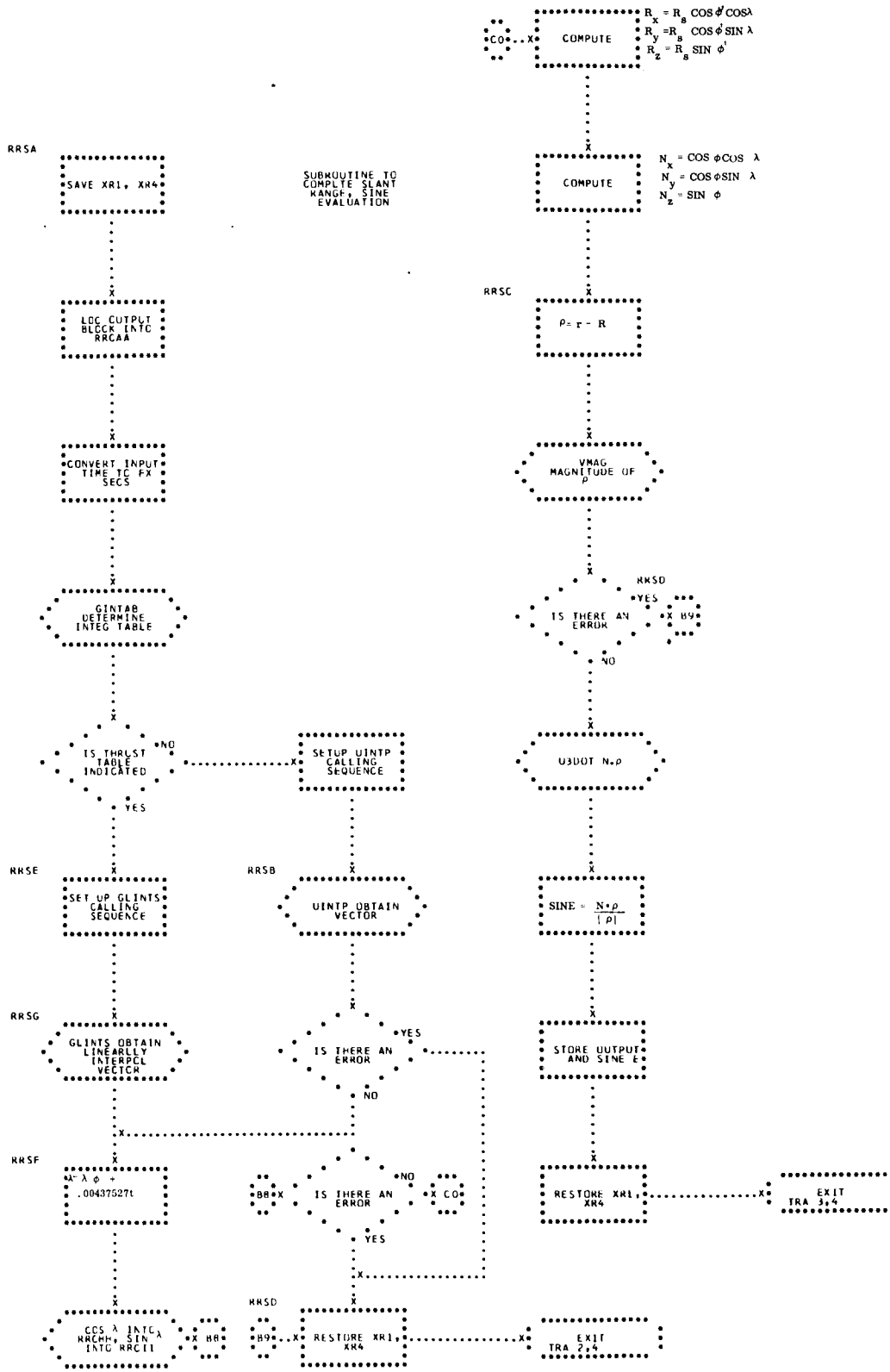

RRBC
TSX RRSA RANGE AND SINE ELEVATION
AO
RRGC
IS THERE AN ERROR
YES
X B5
NO
RANGES FOR T_CA COMP SAVED RRRSI NOT = 0
YES
NO
RRBD
R_i-1 - R_i TIME TO SAVE RANGES
NO
YES
SAVE RANGES AND TIME FOR T_ca
SET RANGE SAVE INC STL INTO RRRSI
RRBE
INIT FCR EXAMINATION OF NEXT CASE
RRBC
IS RRSEB = -
NO
X A0
YES
RRIB INTERP FOR 0° ELEV
CALC TIME DURATION 2ND HC - 1ST HC INTO RRHTH
X B0
RRCP
RANGES SAVED, RRRSI NOT = 0
NO
X B0 1136
YES
RRCA
COMPUTE T_CA
B = 4R_2 - R_3 - 3R_1 / 2
2C = R_3 - 2R_2 + R_1
T_CA = t_R1 - B/2C
T_x = T_CA + 10 SECS.
TSX RREA RANGE AND SINE EKEV AT T_CA
RREA
IS THERE AN ERROR
NO
X A0 1138
YES
RRCP
MAX ELEV INTO RRMRG
X B0 1136
RRGC
B5
RRCP
CALC TIME DURATION - 1ST HC INTO RRHTH
X B0 1136
RRCJ
B6
MANEUVER COUNT INTO XR1
RRDA
ANY MANEUVERS LEFT TO TEST
NO
X C4
YES
RRCK
RRDA
IS 2ND HC GT GMTPS(N)
NO
X C4
YES
RRCL
IS 2ND H C LT GMTPC (N)
YES
X D1
NO
X C0
RRDA
C4
1 - SIN^2 E_CA
SQRT / √1 - SIN^2 E_CA
RREA
IS THERE AN ERROR
YES
X A0 1138
NO
R_2 COS E_CA INTO RRRHO
COMPUTE
Δt_10 = (101.6 - .092 RRRHO)
Δt_30 = (167.1 - .09 RRRHO)
t_10 = t_0 + Δt_10
t_30 = t_0 + Δt_30
t_x = t_CA + 10
X D7
RRCO
D1
MANEUVER PASS INTO RRPAS
X C4 1136
ADJUST XRS TO TEST NEXT MANEUVER
X B7
IS 1ST H C LT GMTPC (N)
NO
YES
D7
PLACE TIMES RROPT. RECORD TIMES
IND. TIMES FL. SECS. 0 INTO XR7
RKEA
NORMAL PASS INTO RRPAS
X A0 1138
RRIB
ENTER
A = SIN 0° OR SIN 1°
t = t_i-1 + (A - SIN E_i-1) / (SIN E_i - SIN E_i-1)
RRIC
FL MINS TO FX SECS
EXIT TRA 1,4


RREA
RREL
A0
ANY OUTPUT RECORDED XR5 NOT = 0
NO
X A5 1139
YES
RRET
A1
INIT 0 INTC
XR1 6 INTC
XR3 30 INTC
XR6
RREB
A2
PICKUP DATA TIME
ARE TIMES IN FX PT XR7 = 1
YES
NO
FIX TIMES
GINTAB DETERMINE INTEG TABLE
CONV TIME TO A(MINS) D(SECS)
IS THRUST TABLE IND
NO
YES
SETUP GLINTS CALLING SEQUENCE
RREX
GLINTS OBTAIN LINEARLY INTERPOL VECTOR
RREY
AILSC COMPUTE STA R,A,E
X C0
SETUP UINTP CALLING SEQUENCE
RRED
UINTP OBTAIN VECTOR
IS THERE AN ERROR
YES
NO
C0
X
IS THERE AN ERROR
YES
X D8
NO
IS XR3 GT 5
NO
YES
IS E LT 0
YES
NO
RREE
STORE TIME IN TMSTAD
CONVERT RANGE TO KILOYARDS
RREF
STORE R,A,E, IN TMSTAD
RRFA GET DATA TAG
STORE DATA TAG IN TMSTAD
D8
X
IS PROCESSING COMPLETED- XR5=0
YES
X A0 1139
NO
RREB
ADJUST XRS FOR NEXT DATA LINE
X A2


AO
ARSIN MAX ELEV FOR PASS
IS THERE AN ERROR
YES
NO
CONVERT E_{MAX} TO DEGREES
RREK
NO OF VALID DATA LINES INTO D(MCNOVL)
IN HIGH ABORT PHASE RRMIP NOT = 0
YES
NO
RESTORE LAUNCH INTEG TABLES TO ORIG CONDITION
RESTORE 7 XRS
EXIT TRA 2,4
RRGE
RREA
TOO FEW VECTORS IN TABLE C(AC) = 0
X AO 1138
IN HIGH ABORT PHASE RRMIP NOT = 0
CALC TIMES IN 30 SEC INTERVALS INTO RKOPT
SET HIGH ABORT INDICATORS
RREL
RRET
A5
IS MARGINAL POINT AVAIL
X A1 1138
INTERNAL STATE VECTOR INTO D(ACDATA)
IN HIGH ABORT PHASE RRMIP NOT = 0
RESTORE LAUNCH INTEG TABLES TO ORIG FORMAT
RESTORE 7 XRS
EXIT TRA 1,4


RRGA
RREA
NO
USING REENTRY TABLE
X AQ 1138
B3
YES
RRGB
RREA
TOO FEW VECTORS IN TABLE. C(AC) = 0
YES
SEARCH TABLE BACKWARDS RRTSI NOT = 0
NO
NO
YES
RREA
RRAD
SEARCH TABLE FORWARD RRTSI = 0
USE FREE FLIGHT ORBIT TABLE
X B6 1135
YES
RRAN
SET SEARCH TIME = END TIME OF RE-ENTRY TABLE
-3 [TIME STEP OF TABLE]
-C(XR1)+16
X D2 1135


RRFA
IN HIGH ABORT PHASE RRMIP NOT = 0
YES
NO
RRFK
GET LAUNCH ABORT TAG
DATA TIMES IN FXSECS, XR7 GT 0
NO
YES
SET RRODE NOT = 0
NO
YES
RRFG
GET MARGINAL DATA TAG
B7
PICKUP A, B, C, OR D TAG
RRFD
IS S/C THRUSTING S(AC) = -
YES
NO
GET MANEUVER TAG
RRFE
USING REENTRY TABLE
YES
NO
IS t BEFORE BLACKOUT
YES
NO
IS t AFTER BLACKOUT
YES
NO
RRFF
GET REENTRY TAG
B7
RRFH
RETROFIRE CASE
NO
YES
RRFJ
GET TAG
GET PRE-POST MANEUVER TAG
GETCOMM BLACKOUT TAG
B7
STORE DATA TAB INTO RRCAT
EXIT TRA 1,4


RRSA
SAVE XR1, XR4
SUBROUTINE TO COMPUTE SLANT RANGE, SINE EVALUATION
LOC OUTPUT BLOCK INTO RRCAA
CONVERT INPUT TIME TO FX SECS
GINTAB DETERMINE INTEG TABLE
IS THRUST TABLE INDICATED
NO
YES
SETUP UINTP CALLING SEQUENCE
RRSE
SET UP GLINTS CALLING SEQUENCE
RRSB
UINTP OBTAIN VECTOR
RRSG
GLINTS OBTAIN LINEARLLY INTERPOL VECTOR
IS THERE AN ERROR
YES
NO
RRSF
.00437527t
IS THERE AN ERROR
NO
YES
CO
B8
CCS λ INTO RRCHH, SIN λ INTO RRCII
B8
RRSD
B9
RESTORE XR1, XR4
EXIT TRA 2,4
CO
COMPUTE
$R_x = R_s \cos\phi' \cos\lambda$
$R_y = R_s \cos\phi' \sin\lambda$
$R_z = R_s \sin\phi'$
COMPUTE
$N_x = \cos\phi \cos\lambda$
$N_y = \cos\phi \sin\lambda$
$N_z = \sin\phi$
RRSC
$\rho = r - R$
VMAG MAGNITUDE OF ρ
RRSD
IS THERE AN ERROR
YES
B9
NO
U3DOT N•ρ
SINE = $\frac{N \cdot \rho}{|\rho|}$
STORE OUTPUT AND SINE E
RESTORE XR1, XR4
EXIT TRA 3,4

63. MPORMC - MONITOR PREFIX TO O5ORMC

MPORMC SETS UP PROGRAM SUPPRESSION, CALCULATES PRESENT TIME AND DETERMINES MISSION PHASE. (SEE FLOWCHART.)

INPUT

A) MCHFSC PRESENT GMT IN FIXED HALF-SECONDS
B) MCNRRF REENTRY PHASE INDICATOR

OUTPUT

A) MCRTMS PRESENT GMT A (MINUTES),,D (SECONDS)
B) MCO5IP SET NON-ZERO TO INDICATE O5ORMC IN PROGRESS
C) C(AC) SET ZERO TO INDICATE ORBIT PHASE
SET NON-ZERO TO INDICATE REENTRY PHASE

USAGE

ENTRY TO MPORMC IS MADE FROM MOPRIO.

A) COMMUNICATION CELLS -
MCHFSC
MCRTMS
MCO5IP
MCNRRF
B) CONSTANTS -
K00060
K00000
K00001
C) MACROS -
TRNON
TRNOF

D) PARAMETERS -

A

B

C

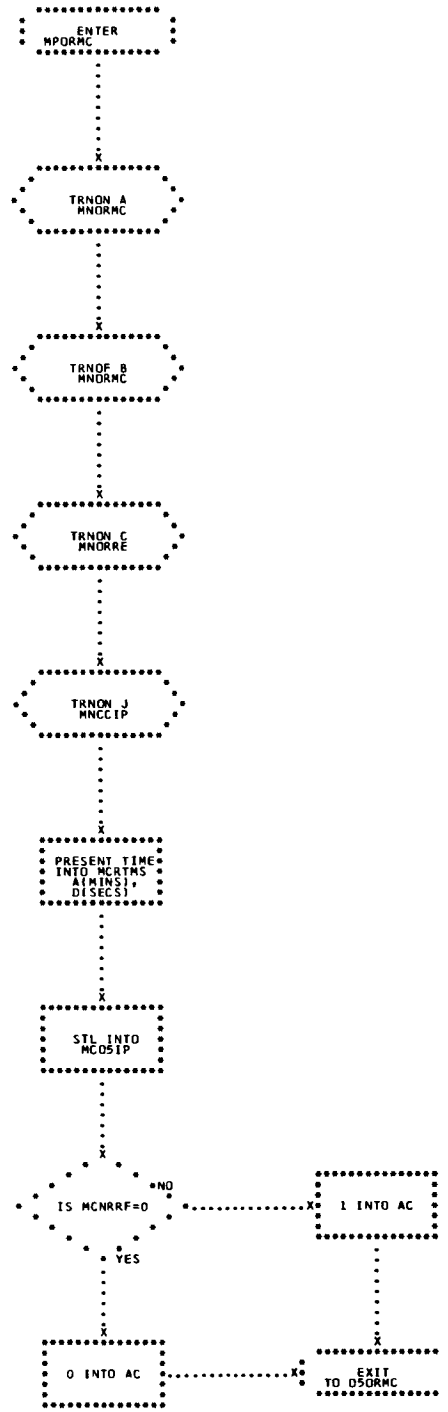
J

MNORMC

MNORRE

MNCCIP

E) STORAGE REQUIRED - 19 LOCATIONS


ENTER MPORMC
TRNON A MNORMC
TRNOF B MNORMC
TRNON C MNORRE
TRNON J MNCCIP
PRESENT TIME INTO MCRTMS A(MINS)' D(SECS)
STL INTO MCOSIP
IS MCNRRF=0
NO
YES
1 INTO AC
0 INTO AC
EXIT TO OSORMC

64. O5ORMC - ORBIT, REENTRY HIGH-SPEED OUTPUT COMPUTATION PROCESSOR

HIGH-SPEED CUTPUT DATA FOR ORBIT AND REENTRY IS COMPUTED BY THE PROCESSOR, O5ORMC. THIS DATA IS PRIMARILY TIME-RELATED, CHANGING AS A FUNCTION OF GMT AND/OR THE ORBIT IN SPACE. DIGITAL DISPLAYS, PLOTBOARDS, CONSOLES AND MAPS ARE CONTINUALLY UPDATED TO PROVIDE REAL-TIME TRAJECTORY PERFORMANCE. (SEE FOWCHART.)

INPUT

A) C(AC) Z ORBIT, NZ REENTRY

B) TABLES -

TMGTRS TABLE OF VALUES FOR RETROFIRE SETTING

TNINT2 REENTRY DISPLAY TABLE

TMORMC ORBIT, REENTRY COMPUTATIONS TABLE

C) COMMUNICATION CELLS -

MCRTMS PRESENT TIME IN GMT, A(MINS),,D(SECS)

MCGTLO GMT OF LIFT-OFF (FX. SECS.)

MCTOFS GMT OF RETROFIRE (FX. SECS.)

OUTPUT

TMORMC IS UPDATED TO REFLECT THE LATEST CHANGES IN THE TRAJECTORY. (SEE TMORMC FORMAT.)

METHOD

UPON ENTRY THE QUANTITIES COMMCN TO ORBIT AND REENTRY ARE COMPUTED -

A) LATITUDE OF PRESENT POSITICN

B) LONGITUDE OF PRESENT POSITION

C) HEIGHT OF PRESENT POSITION

D) TIME ELAPSED SINCE LIFT-OFF

E) INERTIAL VELOCITY OF S/C

F) $A - \bar{R}$

G) INCLINATION ANGLE

H) TIME TO GO TO GMTPI OR GMTRB

THE ACCUMULATOR SETTING IS CHECKED TO DETERMINE MISSION PHASE. IF IN ORBIT, THESE ADDITIONAL QUANTITIES ARE CALCULATED -

LOCATION	QUANTITY	NAME	UNITS	PHASE	WHERE DISPLAYED	STORED BY
TMORMC	ϕ_{PP}	LAT. OF PRESENT POSITION	FL.PT.RAD.	O/R	P.B. NO. 4; WALL MAP	O5ORMC
+1	λ_{PP}	LONG. OF PRESENT POSITION	FL.PT.RAD.	O/R	P.B. NO. 4; WALL MAP	O5ORMC
+2	ϕ_{770}/ϕ_{IP}	LAT. IN 770 SECONDS/ LAT. OF IMPACT	FL.PT.RAD	O/R	P.B. NO. 4; WALL MAP	O5ORMC/ MYREN2
+3	$\lambda_{770}/\lambda_{IP}$	LONG. IN 770 SECONDS/ LONG. OF IMPACT	FL.PT.RAD	O/R	P.B. NO. 3, P.B. NO. 2, FDO (GODDARD)	O5ORMC/ MYREN2
+4	H	HEIGHT ABOVE AN OBLATE EARTH	FL.PT.N.M.	O/R	P.B. NO. 3, P.B. NO. 2, FDO (GODDARD)	O5ORMC
+5	V	INERTIAL S/C VELOCITY	FL.PT.FPS	O/R	P.B. NO. 3, FDO	O5ORMC
+6	$A - \overline{R}$	A MINUS R BAR	FL.PT.N.M.	O	P.B. NO. 2	O5ORMC
+7	T_E	ELAPSED LINE SINCE LIFT-OFF	FX.PT. SECS.	O/R	P.B. NO. 1 P.B. NO. 2/ P.B. NO. 2	O5ORMC
+8	H_{AP}/γ	APOGEE HEIGHT ABOVE OBLATE EARTH/FLIGHT PATH ANGLE	FL.PT.N.M/ FL.PT.RAD.	O/R	P.B. NO. 1, FOD/FOD	MYAPE3/ O5ORMC
+9	λ_{PER}	LONG. OF PERIGEE	FL.PT.RAD.	O	P.B. NO. 1	MYAPE3
+10	ϕ_{IP-ZL}	LAT. OF ZERO LIFT IMPACT POINT	FL.PT.RAD.	R	P.B. NO. 1	MYREN2
+11	λ_{IP-ZL}	LONG. OF ZERO LIFT IMPACT POINT	FL.PT.RAD.	R	P.B. NO 1	MYREN2
+12	ϕ_{IP-ML}	LAT. OF MAXIMUM LIFT IMPACT POINT	FL.PT.RAD.	R	P.B. NO. 1	MYREN2
+13	λ_{IP-ML}	LONG. OF MAXIMUM LIFT IMPACT POINT	FL.PT.RAD.	R	P.B. NO. 1	MYREN2
+14	I	INCLINATION ANGLE	FL.PT.RAD.	O/R	FLIGHT DYNAMICS OFFICER'S CONSOLE	O5ORMC
+15	REV. CAP./ZERO	REVOLUTION CAPABILITY	FX. PT. (B - 35)	O/R	FDO	MYLIF8/ MYREST
+16	REV. NO.	REVOLUTION NUMBER	FX. PT. (B - 35)	O/R	FDO	O5ORMC

TMORMC Format (Sheet 1 of 2)

LOCATION	QUANTITY	NAME	UNITS	PHASE	WHERE DISPLAYED	STORED BY
+17	H_{PER}/ZERO	HEIGHT OF PERIGEE ABOVE OBLATE EARTH	FL.PT.N.M	O/R	FDO	MYAPE3/ MYREST
+18	$GMTRC_{SPA}$/GMTRF	GMT TO RETRO-FIRE COMPUTED/ GMT OF RETROFIRE	FX.PT.SECS.	O/R	RETRO CONSOLE	MYTTF4/ MYREN2
+19	$ESTRC_{SPA}$/GETRC	ELAPSED S/C TIME TO RETRO-FIRE COMPUTED/GET OF RETRO-FIRE	FX.PT.SECS.	O/R	RETRO CONSOLE	MYTTF4/ MYREN2
+20	$GMTRC_{SPA,MPA,MCA}$/ZERO	SAME AS ABOVE//SPA-STORED PRIMARY AREA MPA-MANUAL PRIMARY AREA	FX.PT.SECS.	O/R	RETRO CONSOLE	MYTTF4/ MYREST
+21	$ESTRC_{SPA,MPA,MCA}$/ZERO	SAME AS ABOVE//SCA-STORED CONTINGENCY AREA MCA-MANUAL CONTINGENCY AREA	FX.PT.SECS.	O/R	RETRO CONSOLE	MYTTF4/ MYREST
+22	$GMTLC_{SPA,MPA,MCA}$/ GMTLC	GMT OF LANDING $COMPUTED_{SCA,MPA,MCA}$/GMT OF LANDING	FX.PT.SECS.	O/R	RETRO CONSOLE	MYTTF4/ MYREN2
+23	$\phi IP_{SCA,MPA,MCA}/\phi IP$	LAT. OF $IMPACT_{SCA,MPA,MCA}$/ LAT. OF IMPACT	FL.PT.RAD.	O/R	RETRO CONSOLE	MYTTF4/ MYREN2
+24	$\lambda IP_{SCA,MPA,MCA}/\lambda IP$	LONG. OF $IMPACT_{SCA,MPA,MCA}$/ LONG. OF IMPACT	FL.PT.RAD.	O/R	RETRO CONSOLE	MYTTF4/ MYREN2
+25	$REC.\ AREA_{SCA,MPA,MCA}$	RECOVERY AREA	SPECIAL CODE	O/R	RETRO CONSOLE	MYTTF4/ MYREN2
+26	GMTRS/ETSRF	GMT OF RETROFIRE SETTING/ ET SINCE RETROFIRE	FX.PT.SECS.	O/R	RETRO CONSOLE	MFCCMN, MFMN06/ O5ORMC
+27	ETRSC/ETRLC	ET TO RETROFIRE SETTING/ ET TILL LANDING	FX.PT.SECS.	O/R	WALL DIGITAL	O5ORMC/ O5ORMC
+28	PITCH (B)/BANK ANGLE	S/C PITCH ANGLE/BANK ANGLE	FL.PT.RAD.	O/R	PROPULSION PANEL	MYMUV7/ MYREN2
+29	YAW (α)/ZERO	S/C YAW ANGLE/ZERO	FL.PT.RAD.	O/R	PROPULSION PANEL	MYMUV7/ MYREST
+30	ΔV/ZERO	VELOCITY INCREMENT/ZERO	FL.PT.FPS	O/R	PROPULSION PANEL	MYMUV7/ MYREST
+31	GMTPI/GMTRB	GMT OF PROPULSION INITIATION/ GMT TO REVERSE BANK ANGLE	FX.PT.SECS.	O/R	PROPULSION PANEL	MYMUV7/ MYREN2
+32	$\Delta T_{PI}/\Delta T_{RB}$	ET TO PROPULSION INITIATION/ ET TO REVERSE BANK ANGLE	FX.PT.SECS.	O/R	PROPULSION PANEL	O5ORMC/ O5ORMC

TMORMC Format (Sheet 2 of 2)

A) REVOLUTION NUMBER (O5REV)

B) LATITUDE IN 770 SECONDS

C) LONGITUDE IN 770 SECONDS

D) TIME TO GO TO GMTRS

FOR REENTRY, THESE ADDITIONAL PARAMETERS ARE REQUIRED -

A) FLIGHT PATH ANGLE

B) TIME ELAPSED SINCE RETRO-FIRE

C) TIME TO GO TO GMTLC

EQUATIONS USED BY O5ORMC -

1)
$$a = \frac{|\vec{R}|}{2 - |\vec{R}|\ |\vec{V}|^2}$$
$$\alpha = a - \bar{R}$$

WHERE A = SEMI-MAJOR AXIS

$\bar{R}$ = AN AVERAGE EARTH RADIUS

2)
$$i = \mathrm{Sin}^{-1}\left[\frac{\sqrt{h_x^2 + h_y^2}}{|\vec{h}|}\right]$$

WHERE

3)
$$h = \vec{R} \times \vec{V}$$
$$\gamma = \mathrm{Sin}^{-1}\ \frac{\vec{R} \cdot \vec{V}}{|\vec{R}|\ |\vec{V}|}$$

O5REV IS AN INTERNAL SUBROUTINE USED BY O5ORMC. IT IS ENTERED ONCE EVERY 30 SECONDS TO CHECK FOR THE INITIALIZATION OF A NEW REVOLUTION. THIS IS DONE AS A FUNCTION OF LONGITUDE - PRESENT, PREVIOUS AND CAPE. SHOULD A NEW

REVOLUTION BE ENTERED, AN ON-LINE MESSAGE IS QUEUED TO SUCH OCCURRANCE.

IN EVENT OF A SYSTEM RESTART, EITHER MAGNETIC OR PAPER TAPE, THE REVOLUTION NUMBER ENTERED IN THE COMPUTER CONSOLE KEYS BECOMES THE PRESENT REVOLUTION NUMBER.

A PROGRAM FIRST ENTRY WILL ALSO CAUSE THE PERFORMANCE OF THIS EVENT SEQUENCE -

1) COMPUTATION OF AN APPROXIMATE TIME OF REVOLUTION NUMBER UPDATE.

$$\eta = \frac{\Delta\lambda}{2\pi} \text{ (period)}$$

2) INITIALIZATION OF TTF TIME CELLS.
3) UNSUPPRESSION OF RADAR EDIT.

USAGE

050RMC IS ENTERED FROM MPORMC AND EXITS TO MFORMC
ALL PROGRAM ERROR RETURNS ARE MADE TO MFORMC

A) TABLES -
TMORMC
TMGTRS
TNINT2

B) CONSTANTS -
KD0060
KCGUNM
KCFTSC
K002.0
K.ERGU
K00010
K00770
K00060
K00006
K10800

K86400
KLCAPE
KKK2PI
K0060.
KCH233
K0C001
K0C005
K01020
K04200

C) COMMUNICATION CELLS -

MCRTMS
MCGTLO
MCTOFS
MCTSPA
MCMINS
MCYYYY
MCXXXX
MCORBN
MCNRRF
MCREST
MCREVN
MCHFSC
MCTRNO
MCPER.

D) PARAMETERS -

E
MNMESS
MNLED1

E) MACROS -

GENBZ
QUEUE
QENBA

TRNOF

F) SUBROUTINES -

GINTAB

GLINTS

U7INTP

A3MSCP

VMAG

U3XPR

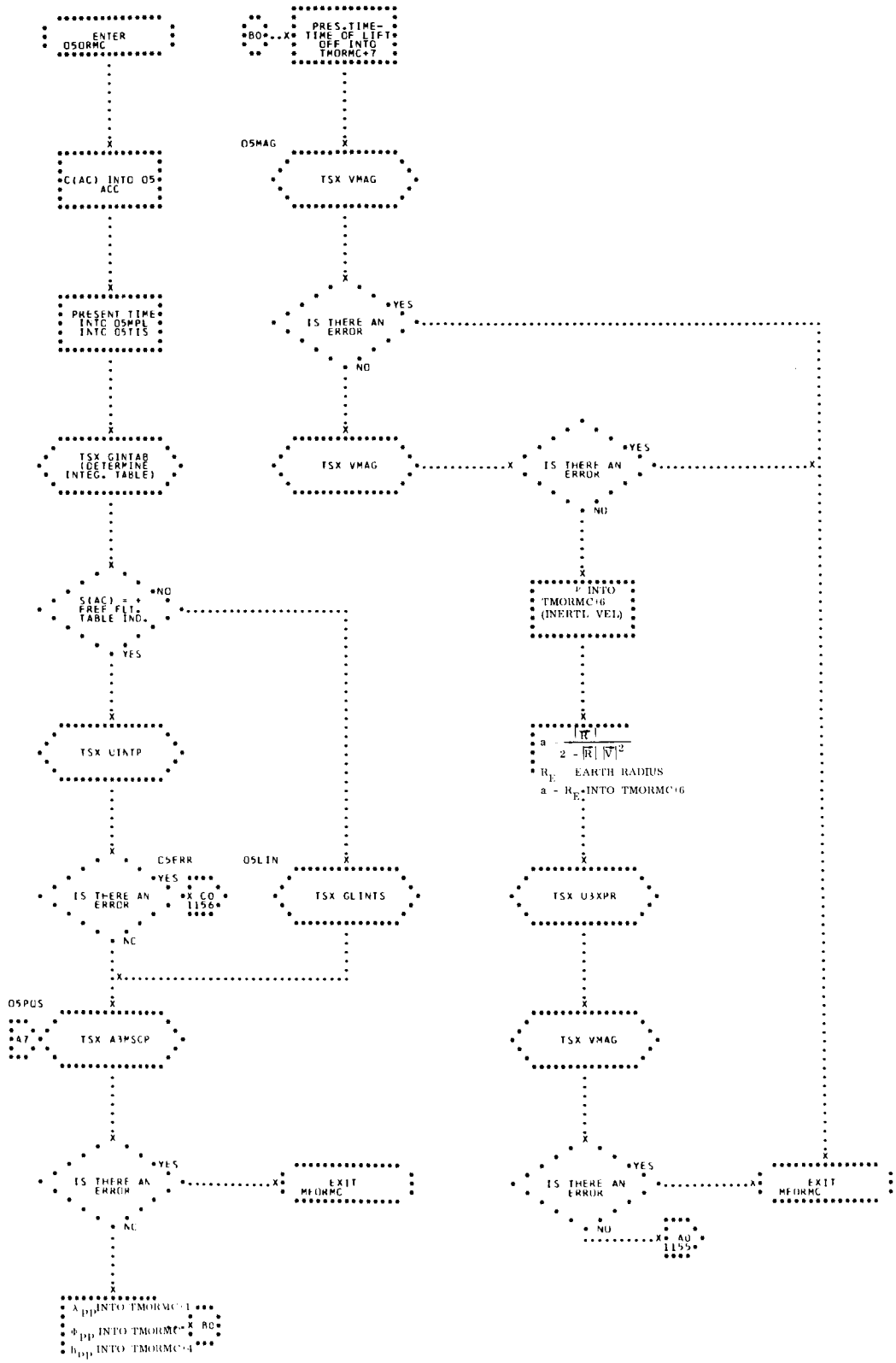
SQRT

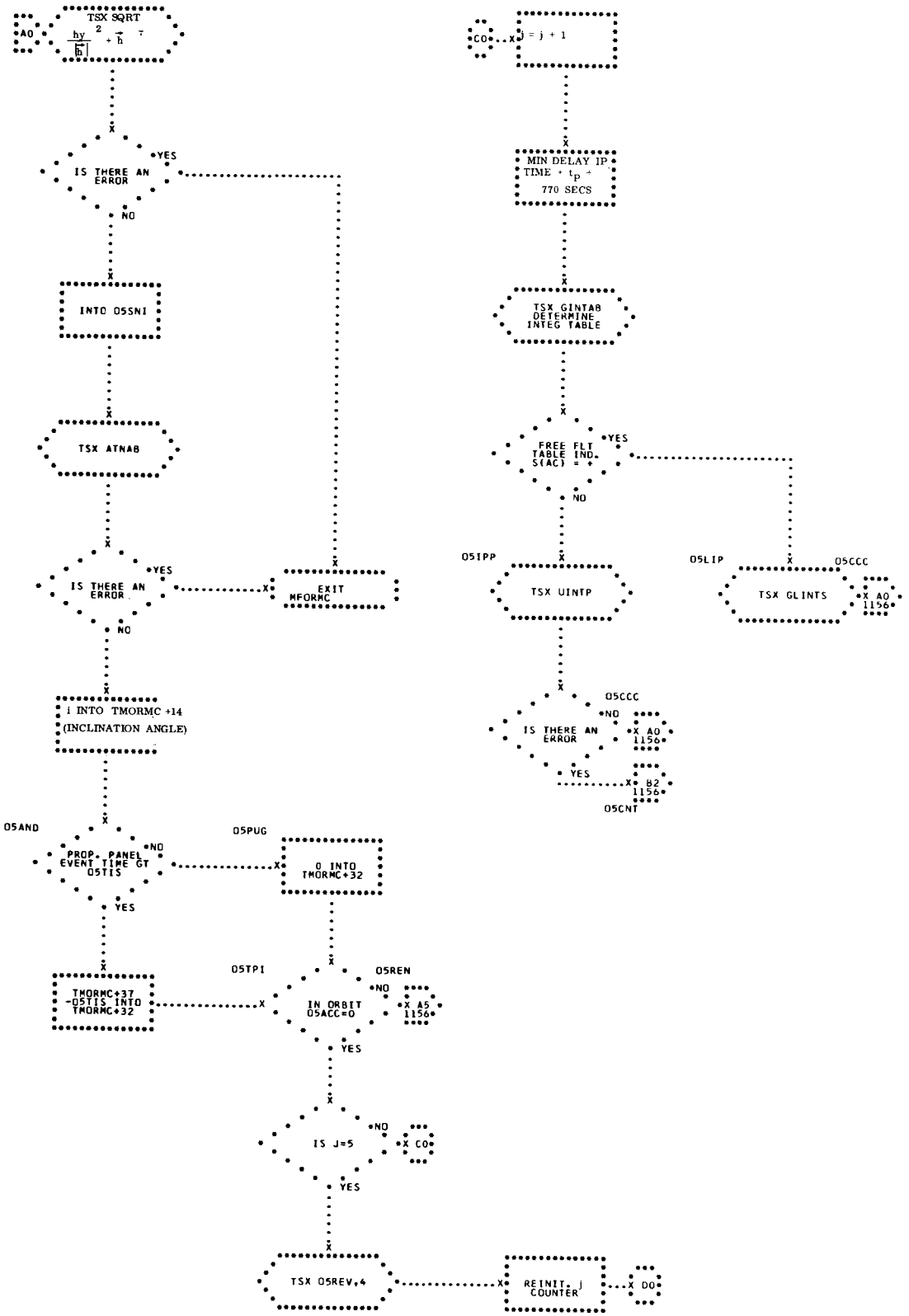
ATNAB

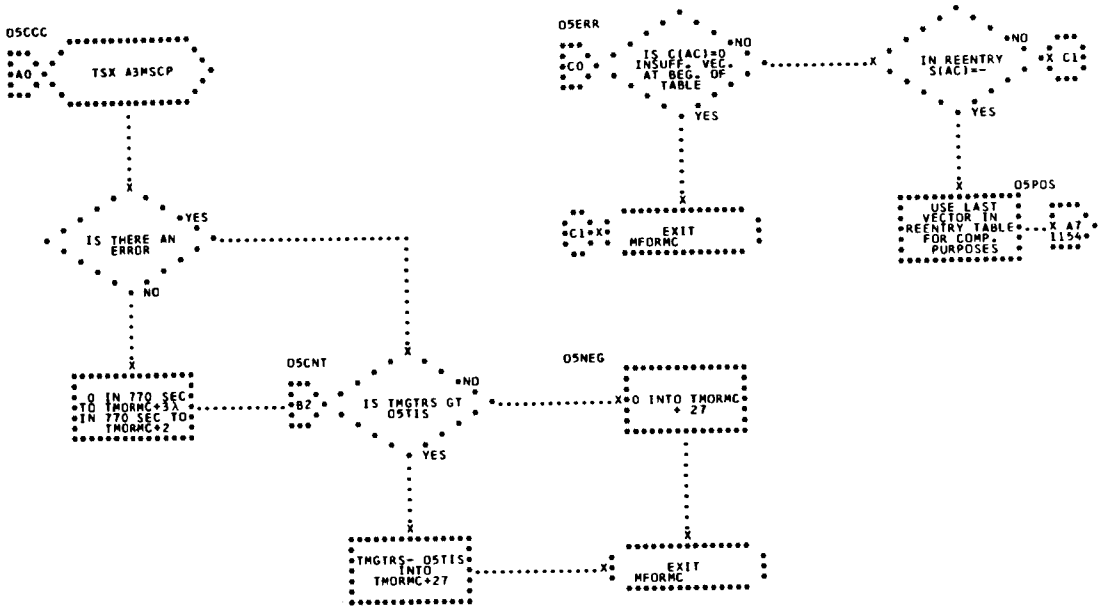
U3DOT

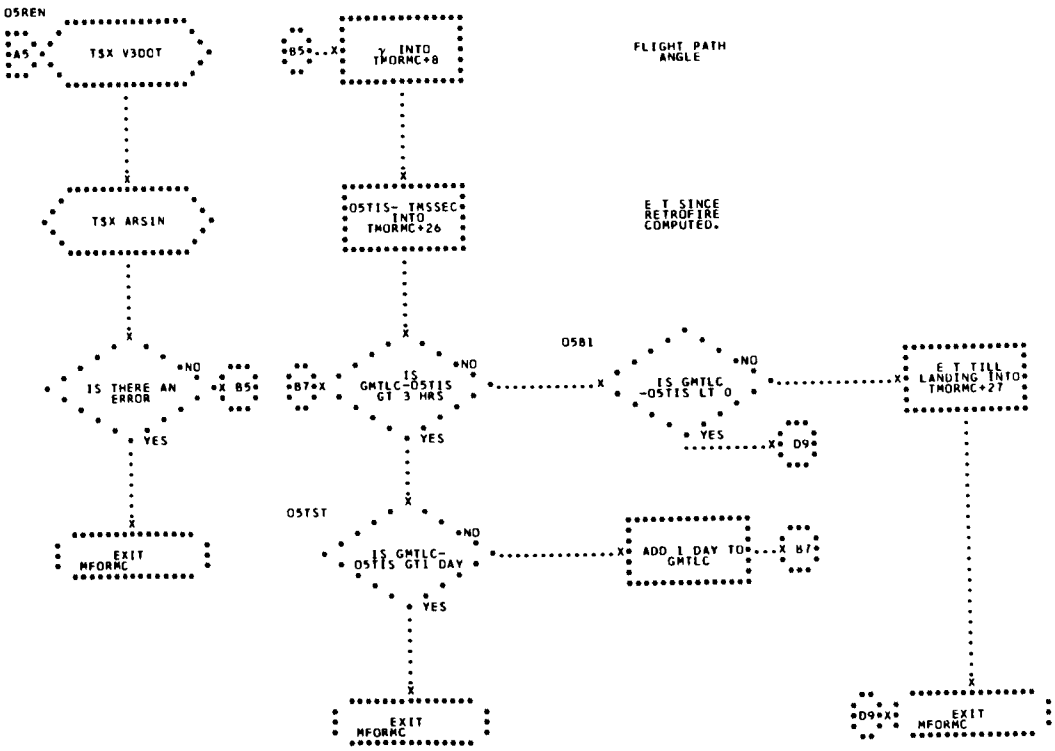
ARSIN

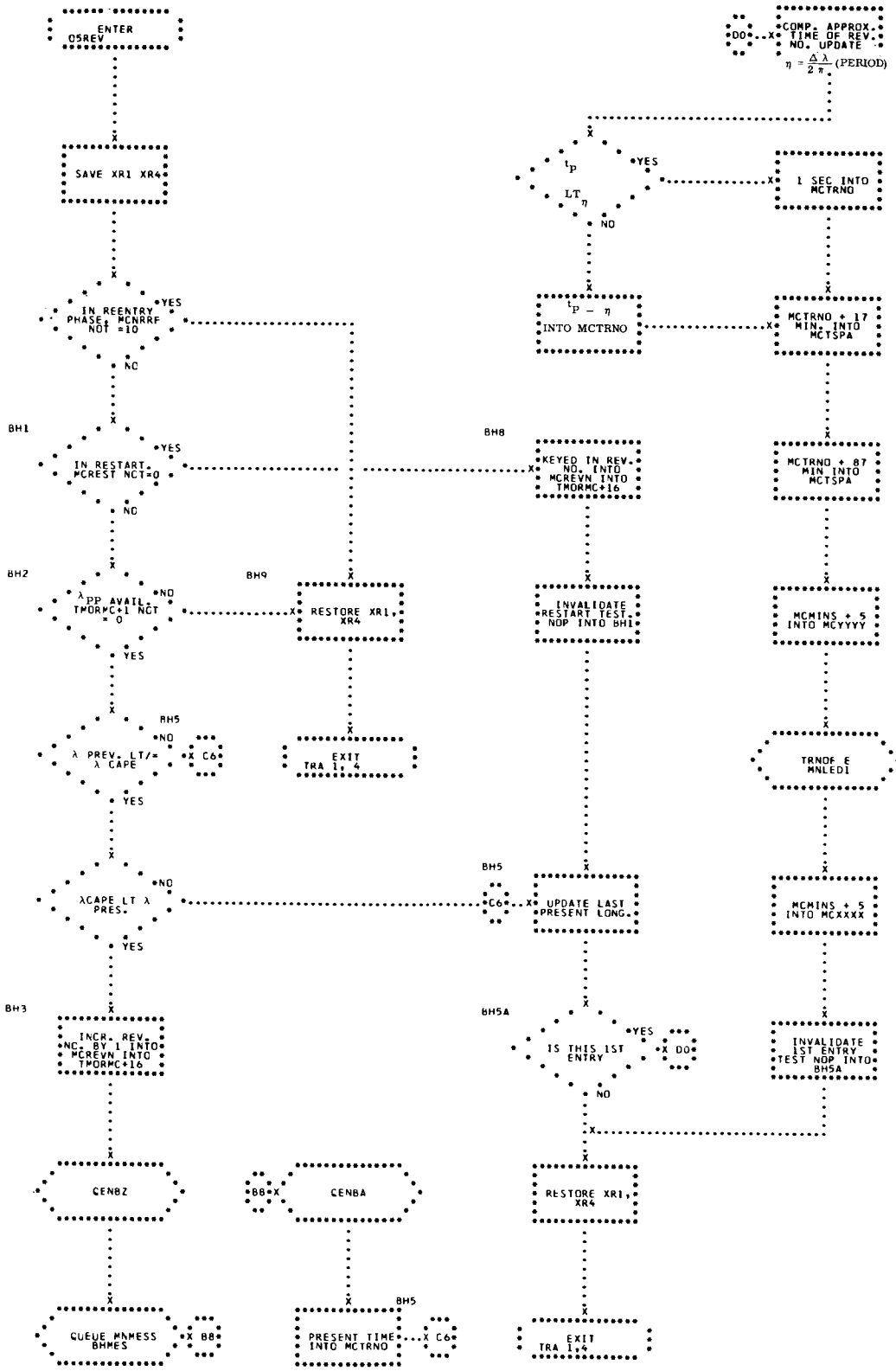
G) STORAGE REQUIRED - 320 LOCATIONS


ENTER O5ORMC
C(AC) INTO O5 ACC
PRESENT TIME INTO O5MPL INTO O5TIS
TSX GINTAB (DETERMINE INTEG. TABLE)
S(AC) = + FREF FLT. TABLE IND.
NO
YES
TSX UINTP
C5ERR
IS THERE AN ERROR
YES
X CO 1156
NO
O5PDS
47
TSX A3PSCP
IS THERE AN ERROR
YES
EXIT MFORMC
NO
λ_{pp} INTO TMORMC+1
Φ_{pp} INTO TMORMC+2
h_{pp} INTO TMORMC+4
X BO
BO
PRES. TIME-TIME OF LIFT OFF INTO TMORMC+7
O5MAG
TSX VMAG
IS THERE AN ERROR
YES
NO
TSX VMAG
IS THERE AN ERROR
YES
NO
$\dot{r}$ INTO TMORMC+6 (INERTL VEL)
$a = \frac{|\bar{R}|}{2 - |\bar{R}|\,|\bar{V}|^2}$
R_E EARTH RADIUS
$a = R_E$ INTO TMORMC+6
O5LIN
TSX GLINTS
TSX U3XPR
TSX VMAG
IS THERE AN ERROR
YES
EXIT MFORMC
NO
X AU 1155


TSX SQRT
AO
IS THERE AN ERROR
YES
NO
INTO O5SNI
TSX ATNAB
IS THERE AN ERROR
YES
NO
EXIT MFORMC
i INTO TMORMC +14
(INCLINATION ANGLE)
O5AND
PROP. PANEL EVENT TIME GT O5TIS
NO
YES
O5PUG
0 INTO TMORMC+32
TMORMC+37 -O5TIS INTO TMORMC+32
O5TPI
O5REN
IN ORBIT O5ACC=0
NO
YES
A5 1156
IS J=5
NO
YES
CO
TSX O5REV,4
REINIT. j COUNTER
DO
CO
j = j + 1
MIN DELAY IP TIME + tp + 770 SECS
TSX GINTAB DETERMINE INTEG TABLE
FREE FLT TABLE IND. S(AC) = +
YES
NO
O5IPP
TSX UINTP
O5LIP
O5CCC
TSX GLINTS
AO 1156
O5CCC
IS THERE AN ERROR
NO
YES
AO 1156
B2 1156
O5CNT


O5CCC
A0
TSX A3MSCP
IS THERE AN ERROR
YES
NO
0 IN 770 SEC TO TMORMC+3 & IN 770 SEC TO TMORMC+2
O5CNT
B2
IS TMGTRS GT O5TIS
NO
YES
TMGTRS- O5TIS INTO TMORMC+27
O5NEG
0 INTO TMORMC + 27
EXIT MFORMC
O5ERR
C0
IS C(AC)=0 INSUFF. VEC. AT BEG. OF TABLE
NO
YES
C1
EXIT MFORMC
IN REENTRY S(AC)=-
NO
X C1
YES
O5POS
USE LAST VECTOR IN REENTRY TABLE FOR COMP. PURPOSES
X A7 1154


O5REN
A5
TSX V3OOT
TSX ARSIN
IS THERE AN ERROR
NO
YES
X B5
EXIT MFORMC
B5
Y INTO TMORMC+8
FLIGHT PATH ANGLE
O5TIS- TMSSEC INTO TMORMC+26
E T SINCE RETROFIRE COMPUTED.
B7
IS GMTLC-O5TIS GT 3 HRS
NO
YES
O5B1
IS GMTLC -O5TIS LT 0
NO
YES
D9
E T TILL LANDING INTO TMORMC+27
O5TST
IS GMTLC- O5TIS GT1 DAY
NO
YES
ADD 1 DAY TO GMTLC
X B7
EXIT MFORMC
D9
EXIT MFORMC


ENTER O5REV
SAVE XR1 XR4
IN REENTRY PHASE, MCNRRF NOT =10
YES
NO
BH1
IN RESTART, MCREST NCT=0
YES
NO
BH2
λPP AVAIL, TMORMC+1 NOT = 0
NO
YES
BH9
RESTORE XR1, XR4
EXIT TRA 1, 4
BH5
λ PREV. LT/= λ CAPE
NO
C6
YES
λCAPE LT λ PRES.
NO
YES
BH3
INCR. REV. NO. BY 1 INTO MCREVN INTO TMORMC+16
CEN8Z
QUEUE MNMESS BHMES
B8
CEN8A
PRESENT TIME INTO MCTRNO
BH5
C6
DO
COMP. APPROX. TIME OF REV. NO. UPDATE
η = Δλ/2π (PERIOD)
tP LT η
YES
NO
1 SEC INTO MCTRNO
tP − η INTO MCTRNO
MCTRNO + 17 MIN. INTO MCTSPA
BH8
KEYED IN REV. NO. INTO MCREVN INTO TMORMC+16
MCTRNO + 87 MIN INTO MCTSPA
INVALIDATE RESTART TEST, NOP INTO BH1
MCMINS + 5 INTO MCYYYY
TRNOF E MNLEDI
BH5
C6
UPDATE LAST PRESENT LONG.
MCMINS + 5 INTO MCXXXX
BH5A
IS THIS 1ST ENTRY
YES
DO
NO
INVALIDATE 1ST ENTRY TEST NOP INTO BH5A
RESTORE XR1, XR4
EXIT TRA 1,4

65. MFORMC - MONITOR SUFFIX TO O5ORMC

MFORMC PREPARES O5ORMC EXIT CONDITIONS. (SEE FLOW CHART)

USAGE

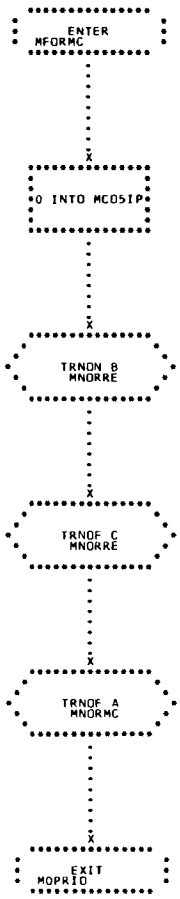
MFORMC IS ENTERED FROM O5ORMC.

A) COMMUNICATIONS CELL -
   MCC5IP

B) MACROS -
   TRNCN
   TRNOF

C) PARAMETERS -
   A
   B
   C
   MNCRMC
   MNCRRE
   MNCCIP

D) STORAGE REQUIRED - 8 LOCATIONS


ENTER
MFORMC
X
0 INTO MCOSIP
X
TRNON B
MNORRE
X
TRNOF C
MNORRE
X
TRNOF A
MNORMC
X
EXIT
MOPRIO

66. RVECHT - COMPUTE HEIGHT VECTOR

SUBROUTINE RVECHT IS USED BY MYREN2 AND R5RARF TO COMPUTE A VECTOR AT A SPECIFIED HEIGHT. THIS SUBROUTINE IS WRITTEN FOR B CORE WITH ALL TEMPORARY STORAGE AND CONSTANTS IN A CORE. (SEE FLOWCHART)

INPUT

A) PROCESSOR -

NOCPNI	NUMERICAL INTEGRATOR

B) SUBROUTINE -

A3MSCP	COMPUTES SUB-CAPSULE POSITION λ, Φ, H.

C) MACRO -

QTSX	SIMULATES A TRAP AND IS USED TO QUEUE INTEGRATOR (NOCPNI)

D) PARAMETERS -

MNBBNI	EQUALS THE SIZE AT THE INPUT TABLE TO THE INTEGRATOR.
MNCPNI	INTEGRATOR ROUTINE NUMBER.

E) COMMUNICATION CELL -

MCRWGT	CONTAINS THE REENTRY WEIGHT IN FLOATING POINT POUNDS.

F) TABLES -

L(HEIGHT)	A 1-CELL TABLE CONTAINING THE HEIGHT, IN GEMINI LENGTH UNITS, AT WHICH A VECTOR IS TO BE COMPUTED.

L(INPUT VECTORS)	A 14-LOCATION TABLE CONTAINING TWO VECTORS WHICH BRACKET THE SPECIFIED HEIGHT IN THE FORMAT t_1, $\bar{r}_1$, $\bar{v}_1$, t_2, $\bar{r}_2$, $\bar{v}_2$ THE TIMES IN FLOATING POINT SECONDS AND THE $\bar{r}$'s AND $\bar{v}$'s IN GEMINI UNITS.
L(N.I.TABLE)	A TABLE WITH MNBBNI NUMBER OF LOCATIONS WHICH IS USED TO SET UP INPUT TO THE INTEGRATOR.

OUTPUT

L(OUTPUT)	A 10-LOCATION TABLE IN THE INTEGRATOR OUTPUT FORMAT FOR A SINGLE VECTOR I.E., TIME UNIT INDICATOR, STORE INTERVAL, t_1, t_n, $\bar{r}_n$, $\bar{v}_n$, WHERE THE TIMES ARE IN FLOATING POINT SECONDS AND $\bar{r}_n$, $\bar{v}_n$ IN GEMINI UNITS.

METHOD

A) IF EITHER t_1 OR t_2 = 0, THEN AN ERROR EXIT IS TAKEN, LIKEWISE FOR ANY SUBSEQUENT ERRORS.

B) A HEIGHT IS COMPUTED FOR EACH INPUT VECTOR USING A3MSCP.

C) THE TIME AT THE SPECIFIED HEIGHT t_H IS LINEARLY INTERPOLATED.

$$t_H = t_1 + h, \quad \text{WHERE } h = (t_2 - t_1)(H - H_1)/(H_2 - H_1)$$

D) THE OUTPUT VECTOR IS COMPUTED BY INTEGRATING ONE STEP FORWARD FROM t_1 TO t_H. THE INTEGRATION IS BASED ON ZERO LIFT USING THE REENTRY WEIGHT.

E) t_H IS GOOD TO WITHIN 0.1 SECONDS ONLY IF $(t_2-t_1) \leq$ 25 SECONDS.

F) NOTES -

1) ALL INPUT MUST BE IN A-CORE. THE REASON FOR THIS IS TO MINIMIZE THE NUMBER OF SEB AND SEA INSTRUCTIONS.

2) A3MSCP IS AN A-CORE SUBROUTINE USED BY RVECHT THROUGH THE LINKAGE ROUTINE RVA3M.

USAGE

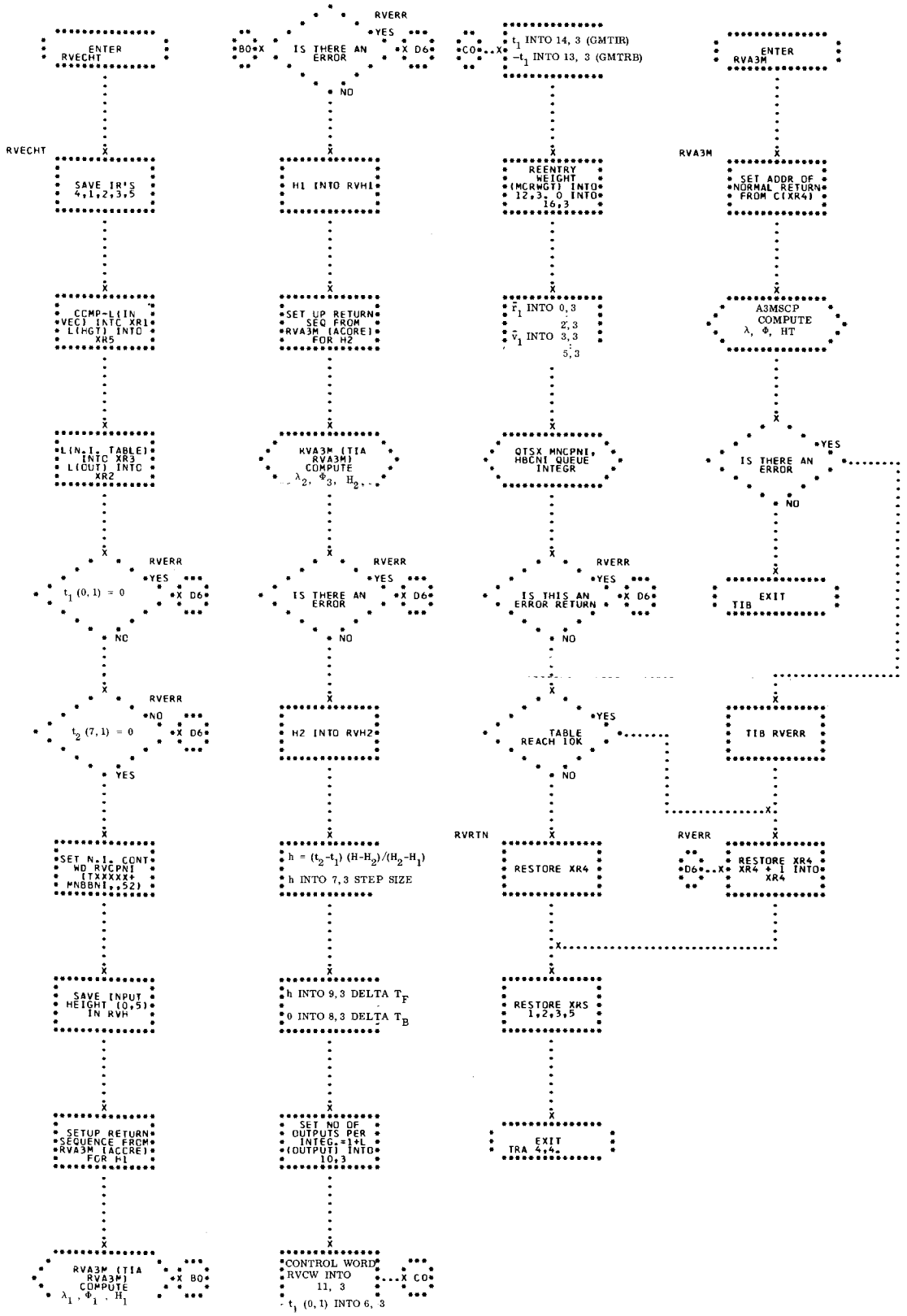
A) CALLING SEQUENCE -

```
TSX             RVECHT,4
PZE             L(INPUT VECTORS),,L(HEIGHT)
PZE             L(N.I. TABLE),,L(OUTPUT)
ERROR           RETURN
NORMAL          RETURN
```

B) STORAGE REQUIRED -

B-CORE	80 LOCATIONS (PROGRAM)
A-CORE	28 LOCATIONS (STORAGE, CONSTANTS, LINKAGE)

C) TIMING - 3.846 MILLISECONDS EXCLUDING THE ONE STEP INTEGRATION TIME.


ENTER RVECHT
RVECHT
SAVE IR'S 4,1,2,3,5
COMP-L(IN VEC) INTO XR1 L(HGT) INTO XR5
L(IN.I. TABLE) INTO XR3 L(OUT) INTO XR2
RVERR
t1 (0, 1) = 0
YES
NO
D6
RVERR
t2 (7, 1) = 0
NO
YES
D6
SET N.I. CONT WD RVCPNI (1XXXXX+ MNBBNI,,52)
SAVE INPUT HEIGHT (0,5) IN RVH
SETUP RETURN SEQUENCE FROM RVA3M (ACCRE) FOR H1
RVA3M (TIA RVA3M) COMPUTE λ1, Φ1, H1
B0
RVERR
IS THERE AN ERROR
YES
NO
D6
H1 INTO RVH1
SET UP RETURN SEQ FROM RVA3M (ACORE) FOR H2
RVA3M (TIA RVA3M) COMPUTE λ2, Φ2, H2
RVERR
IS THERE AN ERROR
YES
NO
D6
H2 INTO RVH2
h = (t2-t1) (H-H2)/(H2-H1)
h INTO 7,3 STEP SIZE
h INTO 9,3 DELTA TF
0 INTO 8,3 DELTA TB
SET NO OF OUTPUTS PER INTEG.=1+L (OUTPUT) INTO 10,3
CONTROL WORD RVCW INTO 11, 3
t1 (0, 1) INTO 6, 3
C0
t1 INTO 14, 3 (GMTIR)
-t1 INTO 13, 3 (GMTRB)
REENTRY WEIGHT (MCRWGT) INTO 12,3, 0 INTO 16,3
R1 INTO 0,3
2,3
V1 INTO 3,3
5,3
QTSX MNCPNI, HBCNI QUEUE INTEGR
RVERR
IS THIS AN ERROR RETURN
YES
NO
D6
TABLE REACH IOK
YES
NO
RVRTN
RESTORE XR4
RESTORE XR5 1,2,3,5
EXIT TRA 4,4
ENTER RVA3M
RVA3M
SET ADDR OF NORMAL RETURN FROM C(XR4)
A3MSCP COMPUTE λ, Φ, HT
YES
IS THERE AN ERROR
NO
EXIT TIB
TIB RVERR
RVERR
D6
RESTORE XR4 XR4 + 1 INTO XR4

67. R6ESTF - ESTIMATED TIME-TO-FIRE PROCESSOR

R6ESTF IS A CLOSED SUBROUTINE OF MYTTF4. IT COMPUTES A FIRST GUESS TO A RETORFIRE TIME, GIVEN AN ORBIT-DEFINING VECTOR AND A TARGET. (SEE FLOWCHART.)

INPUT

TMGTRC	FIRST 7 LOCATIONS CONTAIN x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$ AND T OF A CAPE VECTOR. R6ESTF USES THIS VECTOR, WHICH IS AT THE BEGINNING OF THE REVOLUTION OF THE TARGET, TO CALCULATE AN ESTIMATED TIME-TO-FIRE RETROROCKETS TO HIT THE TARGET.
TDRDWT	FIRST 6 LOCATIONS ARE CURVE-FIT COEFFICIENTS OF DOWNRANGE DISTANCE AS A FUNCTION OF ECCENTRIC ANOMALY. THE FINAL 6 LOCATIONS ARE THE CONVERGENCE CRITERION ON ECCENTRIC ANOMALY, CORRECTION FOR THE EARTH'S ROTATION, CIRCUMFERENCE OF THE EARTH IN NAUTICAL MILES, PRIMARY AREA DELAY, 80° IN RADIANS, AND A FACTOR TO CORRECT FOR NEW LIFT AND DRAG COEFFICIENTS, IN THAT ORDER.
TOSSCH	EIGHTH WORD CONTAINS TIME BETWEEN THE PRERETRO OAMS BURN AND INITIATION OF RETROFIRE.

B) COMMUNICATION CELLS -

MCTYPE — CONTAINS THE QUEUE WORD FOR MYTTF4. R6ESTF USES THE ADDRESS TO OBTAIN THE LONGITUDE AND LATITUDE OF THE TARGET AND THE Δv OF THE PRERETRO OAMS BURN.

C) CONSTANTS -

KKK2PI
KLCAPE
K00PI
KRTDEG
K0060.
KCH233

D) SYSTEM SUBROUTINE -

R7E0AM

E) LIBRARY SUBROUTINES -

BBSQRT
BATNAB
BBBCOS
BBBSIN
BARCOS

F) CONDITIONS ON ENTRY -

MERELY THAT THE INPUT CELLS MUST BE SET. NORMAL RETURN IS THE TSX LOCATION PLUS TWO, ERROR RETURN TSX LOCATION PLUS ONE.

OUTPUT

THE ESTIMATED TIME-TO-FIRE THE PRERETRO OAMS ROCKETS TO HIT THE TARGET IS LEFT IN THE AC, UPON NORMAL EXIT, IN FLOATING POINT SECONDS. IF THERE IS AN ERROR RETURN DUE TO AN ELEMENTARY FUNCTION ERROR RETURN, 0 WILL BE LEFT IN THE AC. IF AN ERROR RETURN IS DUE TO NON-CON-

VERGENCE, THE LAST VALUE OF t_E AFTER 60 ITERATIONS WILL BE LEFT IN THE AC.

METHOD

THE METHOD OF COMPUTING THE FIRST GUESS CONSISTS OF ITERATING UNTIL WE FIND THE ECCENTRIC ANOMALY SUCH THAT FIRING THE PRERETRO OAMS ROCKETS FOLLOWING 12 MINUTES LATER BY THE RETROROCKETS GIVES A DOWNRANGE DISTANCE (CONVERTED TO CENTRAL ANGLE) WITHIN 40 N.M. OF THE DISTANCE BETWEEN THE PRESENT POSITION LOCATION AND THE TARGET. USING THE MEAN MOTION, THIS ECCENTRIC ANOMALY CAN THEN BE CONVERTED TO $A\Delta t$, WHICH IS THE TIME FROM THE STARTING POINT, THE CAPE VECTOR. DETAILS ARE GIVEN IN THE FLOWCHART.

USAGE

ENTRY IS FROM MYTTF4 AND EXIT TO MYTTF4

A) CALLING SEQUENCE -

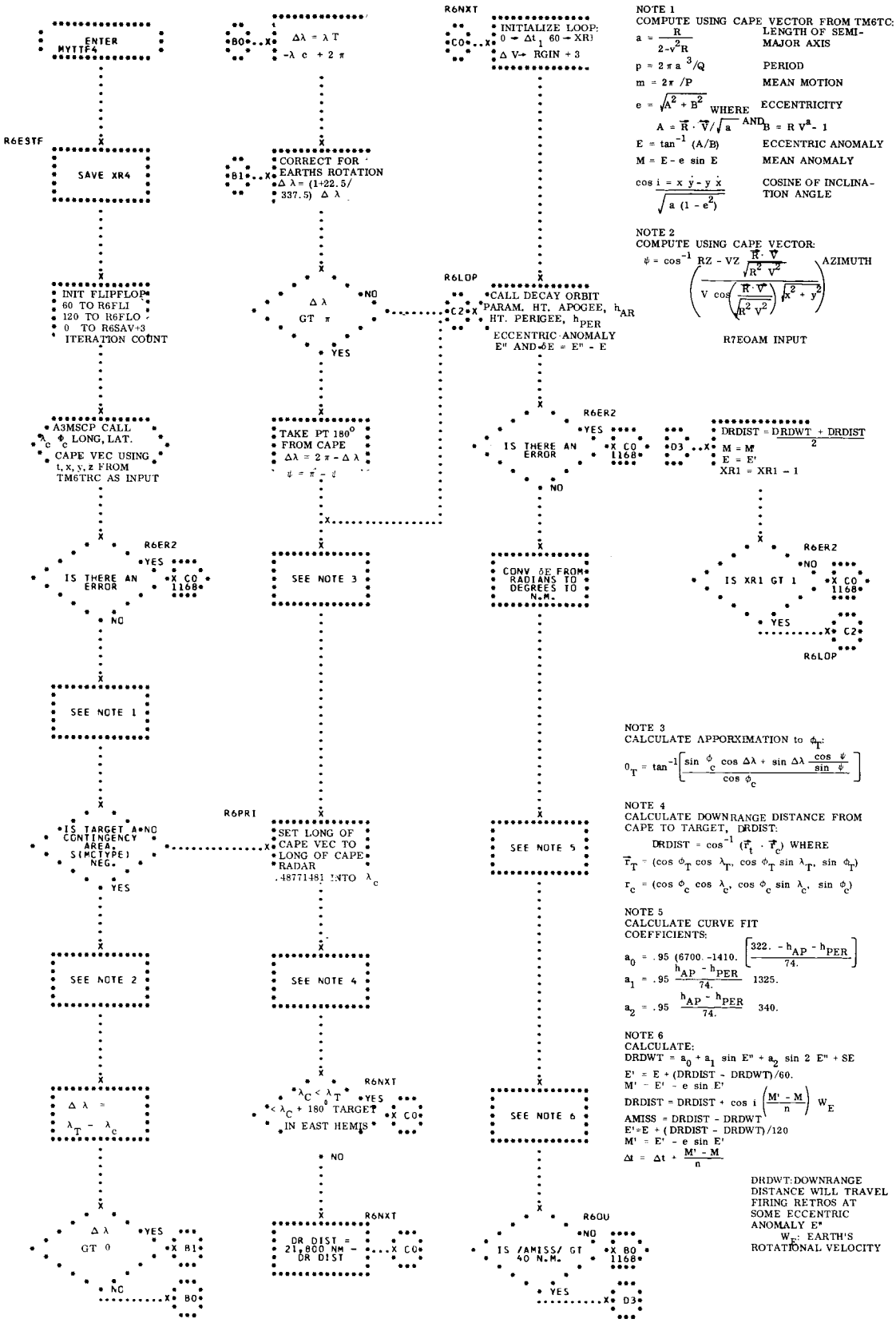
```
α     TSX R6ESTF,4
α +1      ERROR RETURN
α +2      NORMAL RETURN
```

B) STORAGE REQUIRED - 545 LOCATIONS

C) TIMING - MAXIMUM (CONTINGENCY AREA, 60 ITERATIONS)
63.205 MS + 60 • R7

WHERE

R7 IS THE TIME REQUIRED TO EXECUTE R7EOAM.



NOTE 1
COMPUTE USING CAPE VECTOR FROM TM6TC:

$a = \frac{R}{2 - v^2 R}$ LENGTH OF SEMI-MAJOR AXIS

$p = 2\pi a^{3}/Q$ PERIOD

$m = 2\pi / P$ MEAN MOTION

$e = \sqrt{A^2 + B^2}$ WHERE ECCENTRICITY

$A = \vec{R} \cdot \vec{V} / \sqrt{a}$ AND $B = R V^2 - 1$

$E = \tan^{-1}(A/B)$ ECCENTRIC ANOMALY

$M = E - e \sin E$ MEAN ANOMALY

$\cos i = \frac{x\dot{y} - y\dot{x}}{\sqrt{a(1 - e^2)}}$ COSINE OF INCLINATION ANGLE

NOTE 2
COMPUTE USING CAPE VECTOR:

$$\psi = \cos^{-1}\left(\frac{RZ - VZ\left(\frac{\vec{R}\cdot\vec{V}}{R^2 V^2}\right)}{V\cos\left(\frac{\vec{R}\cdot\vec{V}}{R^2 V^2}\right)\sqrt{x^2 + y^2}}\right)$$ AZIMUTH

R7EOAM INPUT

NOTE 3
CALCULATE APPORXIMATION to ϕ_T:

$$\phi_T = \tan^{-1}\left[\frac{\sin\phi_c \cos\Delta\lambda + \sin\Delta\lambda \frac{\cos\psi}{\sin\psi}}{\cos\phi_c}\right]$$

NOTE 4
CALCULATE DOWNRANGE DISTANCE FROM CAPE TO TARGET, DRDIST:

$DRDIST = \cos^{-1}(\vec{r}_t \cdot \vec{r}_c)$ WHERE

$\vec{r}_T = (\cos\phi_T \cos\lambda_T, \cos\phi_T \sin\lambda_T, \sin\phi_T)$

$r_c = (\cos\phi_c \cos\lambda_c, \cos\phi_c \sin\lambda_c, \sin\phi_c)$

NOTE 5
CALCULATE CURVE FIT COEFFICIENTS:

$a_0 = .95\ (6700. - 1410.\left[\frac{322. - h_{AP} - h_{PER}}{74.}\right]$

$a_1 = .95\ \frac{h_{AP} - h_{PER}}{74.}\ 1325.$

$a_2 = .95\ \frac{h_{AP} - h_{PER}}{74.}\ 340.$

NOTE 6
CALCULATE:

$DRDWT = a_0 + a_1 \sin E" + a_2 \sin 2\ E" + SE$

$E' = E + (DRDIST - DRDWT)/60.$

$M' = E' - e \sin E'$

$DRDIST = DRDIST + \cos i \left(\frac{M' - M}{n}\right) W_E$

$AMISS = DRDIST - DRDWT$

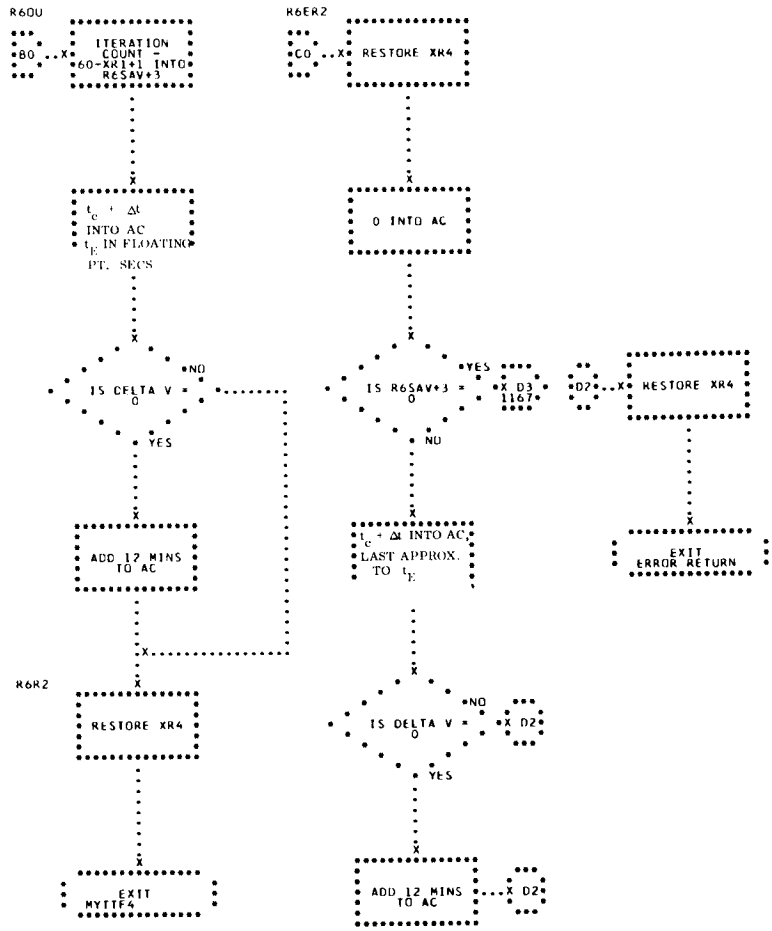
$E' = E + (DRDIST - DRDWT)/120$

$M' = E' - e \sin E'$

$\Delta t = \Delta t + \frac{M' - M}{n}$

DRDWT: DOWNRANGE DISTANCE WILL TRAVEL FIRING RETROS AT SOME ECCENTRIC ANOMALY E"

W_E: EARTH'S ROTATIONAL VELOCITY


R60U
80
ITERATION COUNT = 60-XR1+1 INTO R6SAV+3
$t_c + \Delta t$ INTO AC
t_E IN FLOATING PT. SECS
IS DELTA V = 0
NO
YES
ADD 12 MINS TO AC
R6R2
RESTORE XR4
EXIT MYITF4
R6ER2
CO
RESTORE XR4
0 INTO AC
IS R6SAV+3 = 0
YES
D3 1167
NO
$t_c + \Delta t$ INTO AC, LAST APPROX. TO t_E
IS DELTA V = 0
NO
D2
YES
ADD 12 MINS TO AC
D2
D2
RESTORE XR4
EXIT ERROR RETURN

68. R7ECAM - FIRST GUESS FOR TIME TO FIRE RETROS

PROGRAM R7ECAM UPDATES CERTAIN KEPLERIAN PARAMETERS TO REFLECT A CHANGE IN VELOCITY DUE TO AN IMPULSIVELY APPLIED OAMS BURN. IT ALSO COMPUTES THE ECCENTRIC ANOMALY AT RETROFIRE AND APPROXIMATES THE EARTH-RELATIVE CENTRAL ANGLE SUBTENDED BY THE SPACECRAFT FROM THE BEGINNING OF OAMS BURN TO THE BEGINNING OF RETROFIRE. (SEE FLOWCHART.)

INPUT

A) CONSTANTS -

KOO.SG — CONVERTS SECONDS TO GEMINI TIME UNITS

K.OMES — ROTATION RATE OF THE EARTH IN RADIANS PER SECOND

B) LIBRARY SUBROUTINES -

U1SQRT

U1SICO

U1ASCO

C) TABLES -

TOSCCH — TOSCCH+8 CONTAINS δt, THE TIME INTERVAL BETWEEN THE BEGINNING OF OAMS BURN AND THE BEGINNING OF RETROFIRE IN FLOATING POINT SECONDS

L(INPUT) — THE LOCATION OF THE FIRST ENTRY IN A FIVE-ITEM TABLE WHICH CONTAINS THE FOLLOWING -

L(INPUT) — a, THE LENGTH OF THE SEMI MAJOR AXIS OF THE ORBIT IN FLOATING POINT GEMINI LENGTH UNITS

+1 e, THE ECCENTRICITY OF THE ORBIT, A FLOATING POINT DIMENSIONLESS NUMBER

+2 E, THE ECCENTRIC ANOMALY OF THE SPACECRAFT POSITION, IN FLOATING POINT RADIANS

+3 Δv, THE CHANGE IN VELOCITY DUE TO OAMS BURN, IN FLOATING POINT FT/SEC

+4 COS i, THE COSINE OF THE ANGLE BETWEEN THE ORBITAL AND EQUATORIAL PLANES, A FLOATING POINT DIMENSIONLESS NUMBER.

OUTPUT

A) TABLE -

L(OUTPUT) THE LOCATION OF THE FIRST ENTRY IN A SIX-ITEM TABLE WHICH CONTAINS THE FOLLOWING -

L(OUTPUT) THE APOGEE HEIGHT AFTER OAMS BURN, IN FLOATING POINT NAUTICAL MILES ABOVE THE SURFACE OF A SPHERICAL EARTH OF RADIUS 1.0 GLU

+1 THE PERIGEE HEIGHT AFTER OAMS BURN, IN FLOATING POINT NAUTICAL MILES ABOVE THE SURFACE OF A SPHERICAL EARTH OF RADIUS 1.0 GLU

+2 a', THE LENGTH OF THE SEMI MAJOR AXIS OF THE ORBIT AF-

TER OAMS BURN, IN FLOATING POINT GEMINI LENGTH UNITS

+3 e', THE ECCENTRICITY OF THE ORBIT AFTER OAMS BURN, A FLOATING POINT DIMENSIONLESS NUMBER

+4 E'', THE ECCENTRIC ANOMALY AT THE BEGINNING OF RETROFIRE, IN FLOATING POINT RADIANS

+5 δE, THE EARTH RELATIVE CENTRAL ANGLE SUBTENEDED BY THE SPACECRAFT FROM THE BEGINNING OF OAMS BURN TO THE BEGINNING OF RETROFIRE, IN FLOATING POINT RADIANS

METHOD

E IS FIRST SCALED TO ASSURE THAT

$$0 \leq E < 2\pi$$

AFTER WHICH THE FOLLOWING KEPLERIAN PARAMETERS ARE COMPUTED -

$$r = a(1 - e\cos E)$$
$$h^2 = a(1 - e^2)$$
$$V^2 = (2/r) - (1/a)$$
$$V_T = h/r$$

THE CHANGE IN VELOCITY DUE TO OAMS BURN IS INTRODUCED. IT ACTS IN THE ORBITAL PLANE PERPENDICULAR TO THE POSITION VECTOR, AND OPPOSITE THE DIRECTION OF TRAVEL.

$$V_T' = V_T - \Delta V,$$
$$h' = r\,V_T',$$
$$(V')^2 = (V_T')^2 + V^2 - V_T^2,$$

WHERE THE LAST TWO TERMS REPRESENT THE SQUARE OF THE VELOCITY COMPONENT PARALLEL TO THE POSITION VECTOR.

$$a' = r/\left[2 - r\,(V')^2\right]$$
$$e' = \sqrt{1 - (h')^2/a'}$$

BECAUSE OF THE INHERENT INACCURACY OF THIS FORMULATION OF e' , A LOWER LIMIT OF $.3 \times 10^{-6}$ HAS BEEN ESTABLISHED FOR $(e')^2$. FOR VALUES BELOW THIS, THE PARAMETERS ARE SET AS FOLLOWS -

$$e' = 0$$
$$\text{apogee} = \text{perigee} = a' - 1$$
$$E = \pi$$

THE FLOW THEN CONTINUES WITH THE COMPUTATION OF THE MEAN ANOMALY M'.

FOR VALUES OF $(e')^2$ GREATER THAN THE ABOVE CRITERION, COMPUTATIONS PROCEED AS FOLLOWS -

$$\text{apogee height} = a'\,(1 + e') - 1$$
$$\text{perigee height} = a'\,(1 - e') - 1$$
$$\cos E' = \frac{1}{e'}\,(1 - r/a')$$

FOR E' NEAR PERIGEE OR APOGEE, WHEN $|1 - r / a'| \cong e'$, THE ERROR IN e' MAY DRIVE $|\cos E'|$ ABOVE 1.0. THUS IF

$$1.2 > |\cos E'| > 1.0,$$

$|\cos E'|$ IS SET TO 1.0.

E' IS COMPUTED USING U1ASCO, WHICH LEAVES IT IN THE RANGE 0 TO π . IT IS THEN SCALED AS FOLLOWS -

$$2\pi - E' \rightarrow E' \quad \text{for} \quad \pi \leq E < 2\pi$$
$$E' \rightarrow E' \quad \text{for} \quad 0 \leq E < \pi$$

THE MEAN ANOMALY AT RETROFIRE CAN NOW BE COMPUTED -

$$M' = E' - e' \sin E' + \delta t / a^{3/2}$$

KEPLER'S EQUATION,

$$M' = E'' - e' \sin E''$$

IS SOLVED FOR E'', USING

$$E_o'' = M' + e' \sin M'$$

AS AN INITIAL GUESS FOR E'', AND

$$M_o' = E_o'' - e' \sin E_o'',$$

$$\Delta M' = M' - M_o', \text{ and}$$

$$\Delta E_o'' = \Delta M' / (1 - e' \cos E_o'')$$

FOR ITERATION EQUATIONS.

THE APPROXIMATION OF THE CENTRAL ANGLE IS THEN COMPUTED

$$\delta E = E'' - E' - \omega_e \delta t \cos i$$

WHERE ω_e IS THE ROTATION RATE OF THE EARTH IN RADIANS/SEC.

USAGE

A) CALLING SEQUENCE -

```
TSX     R7EOAM,4
PZE     L(INPUT),,L(OUTPUT)
ERROR   RETURN
NORMAL  RETURN
```

B) STORAGE REQUIRED - 237 LOCATIONS

C) TIME REQUIRED -

7094 MAXIMUM	14.52 MILLISECONDS
7094 AVERAGE	3.69 MILLISECONDS

D) ERROR CONDITION - AN ERROR RETURN INDICATES ONE OF THE FOLLOWING -

1) AN ERROR RETURN FROM ONE OF THE SUBROUTINES
2) THE ABSOLUTE VALUE OF COS E' EQUALS OR EXCEEDS 1.2
3) THE ABSOLUTE VALUE OF E IN THE INPUT IS GREATER THAN 40π
4) KEPLER'S EQUATION DOES NOT CONVERGE IN 25 ITERATIONS

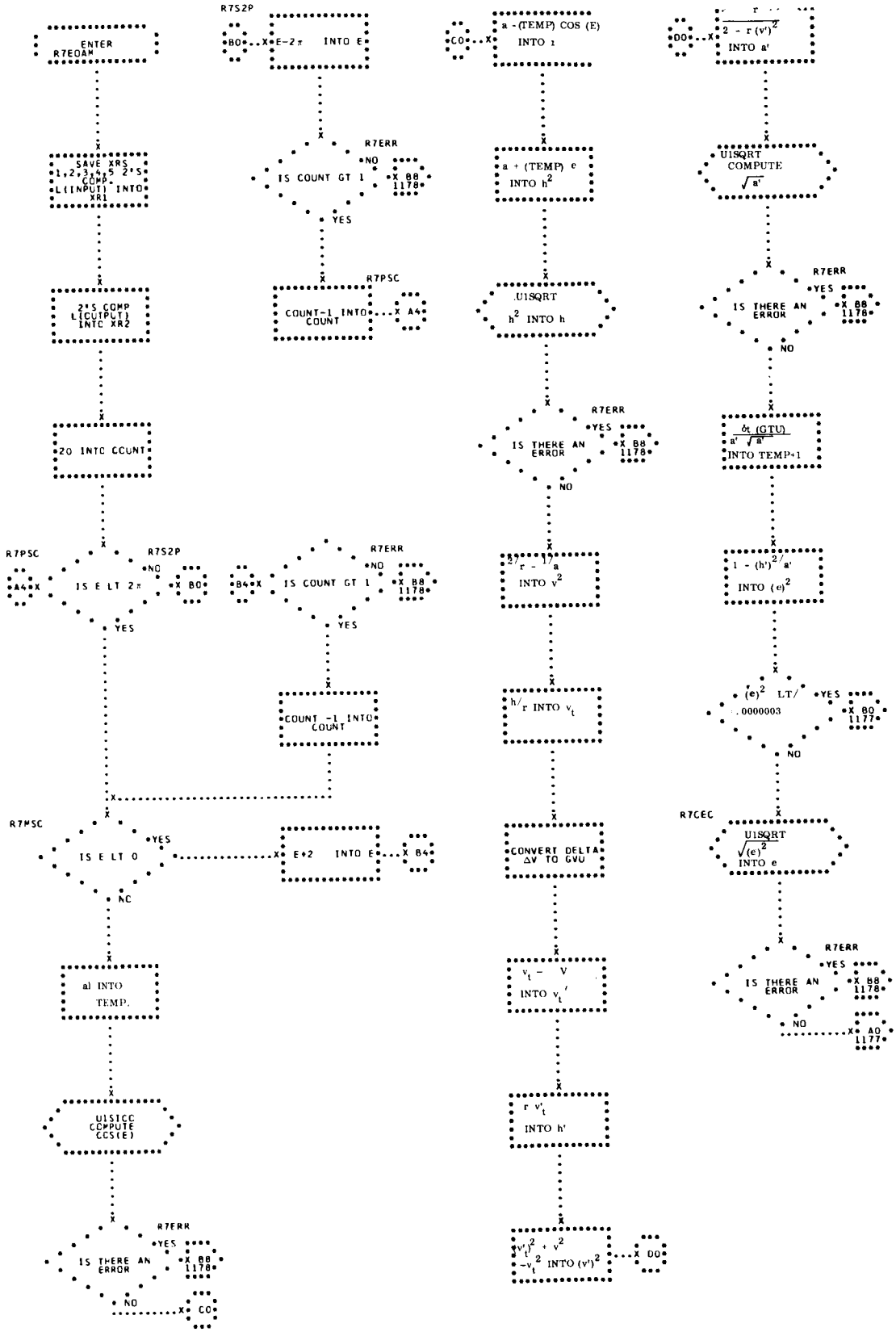
E) THIS PROGRAM HAS BEEN CODED FOR USAGE IN B CORE. IT IS ASSUMED THAT THE INPUT AND OUTPUT TABLES ARE ALSO LOCATED IN B CORE. THE ONLY A CORE CELLS USED ARE THE FOLLOWING -

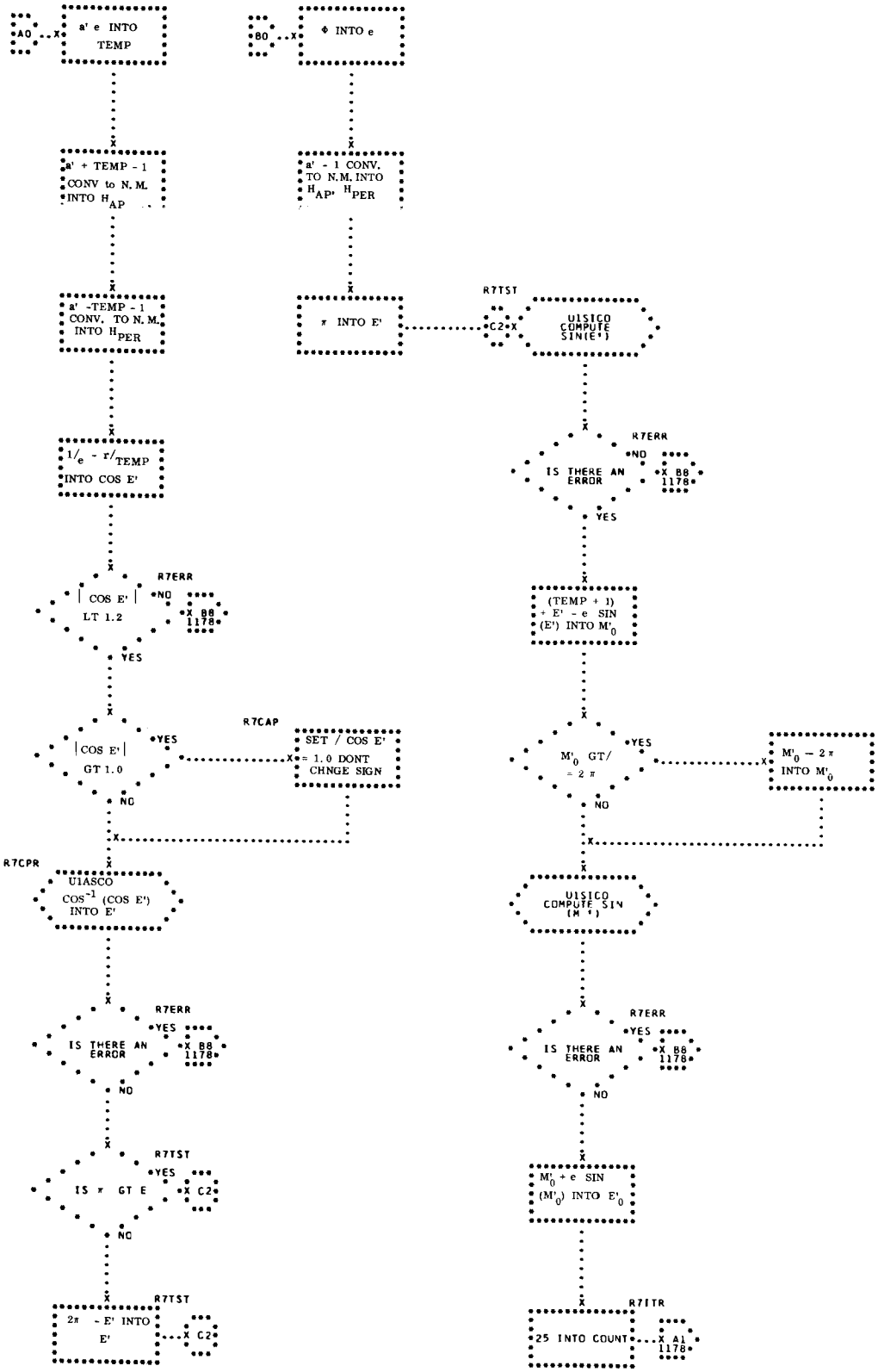
TOSCCH+8
KOO.SG

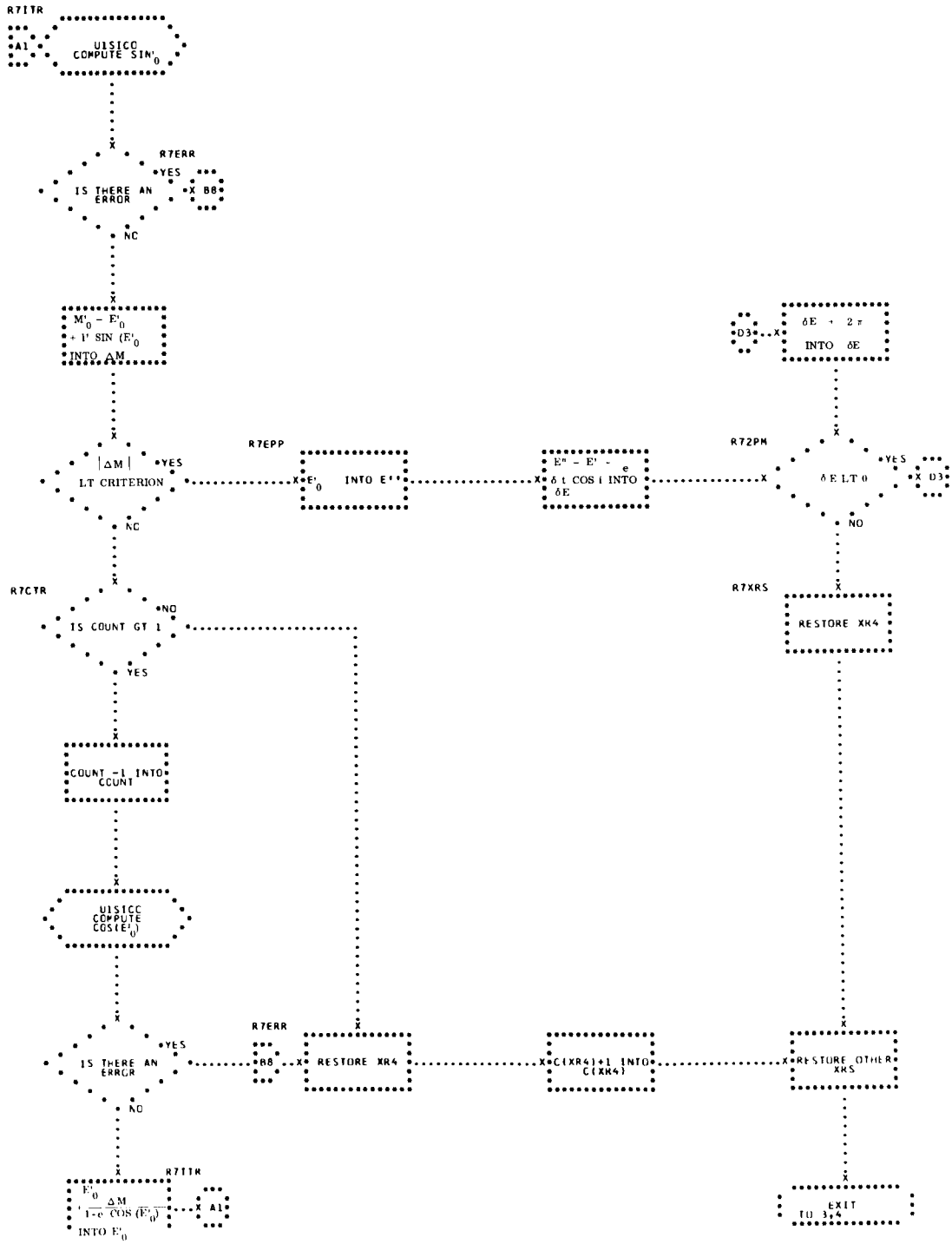
VOL III REAL TIME PROCESSORS PART II - ORBIT/REENTRY

K.OMES

NORMAL LINKAGE FOR A CORE LIBRARY SUBROUTINES


ENTER R7EOAM
SAVE XR5 1,2,3,4,5 2'S COMP. L(INPUT) INTO XR1
2'S COMP L(OUTPUT) INTO XR2
20 INTO COUNT
R7PSC
R7S2P
IS E LT 2π
NO
YES
X B0
A4 X
R7ERR
IS COUNT GT 1
NO
YES
X B8 1178
B4 X
COUNT -1 INTO COUNT
R7MSC
IS E LT 0
YES
NO
E+2 INTO E
X B4
a1 INTO TEMP.
U1SICC COMPUTE COS(E)
IS THERE AN ERROR
X C0
R7S2P
B0 X
E-2π INTO E
R7PSC
COUNT-1 INTO COUNT
X A4
a - (TEMP) COS (E) INTO r
C0 X
a + (TEMP) e INTO h²
U1SQRT h² INTO h
IS THERE AN ERROR
X D0
2 - r (v')² INTO a'
U1SQRT COMPUTE √a'
IS THERE AN ERROR
X B8 1178
δt (GTU) / a' √a' INTO TEMP+1
1 - (h')²/a' INTO (e)²
(e)² LT/ .0000003
X B0 1177
R7CEC
U1SQRT √(e)² INTO e
IS THERE AN ERROR
X A0 1177
2/r - 1/a INTO v²
h/r INTO v_t
CONVERT DELTA ΔV TO GVU
v_t - V INTO v_t'
r v'_t INTO h'
(v'_t)² + v² - v_t² INTO (v')²
X D0


A0
a' e INTO TEMP
a' + TEMP - 1 CONV to N.M. INTO H_{AP}
a' -TEMP - 1 CONV. TO N.M. INTO H_{PER}
$1/e - r/_{TEMP}$ INTO COS E'
| COS E' | LT 1.2
NO
YES
R7ERR
88 1178
| COS E' | GT 1.0
YES
NO
R7CAP
SET / COS E' = 1.0 DONT CHNGE SIGN
R7CPR
U1ASCO COS^{-1} (COS E') INTO E'
IS THERE AN ERROR
IS π GT E
C2
2π - E' INTO E'
B0
Φ INTO e
a' - 1 CONV. TO N.M. INTO H_{AP}, H_{PER}
π INTO E'
R7TST
U1SICO COMPUTE SIN(E')
(TEMP + 1) + E' - e SIN (E') INTO M'_0
M'_0 GT/ = 2π
$M'_0 - 2\pi$ INTO M'_0
U1SICO COMPUTE SIN (M')
M'_0 + e SIN (M'_0) INTO E'_0
25 INTO COUNT
R7ITR
A1


R7ITR
A1
UISICC
COMPUTE SIN'0
R7ERR
YES
IS THERE AN
ERROR
B8
NO
M'0 - E'0
+ l' SIN (E'0
INTO ΔM
|ΔM|
YES
LT CRITERION
R7EPP
E'0
INTO E''
E'' - E' - e
δ t COS i INTO
δE
R72PM
δE LT 0
YES
D3
NO
δE + 2π
INTO δE
D3
R7CTR
NO
IS COUNT GT 1
YES
COUNT -1 INTO
CCUNT
UISICC
COMPUTE
COS(E'0)
R7ERR
YES
IS THERE AN
ERROR
B8
NO
RESTORE XR4
C(XR4)+1 INTO
C(XR4)
R7XRS
RESTORE XR4
RESTORE OTHER
XRS
E'0 + ΔM
1-e COS (E'0)
INTO E'0
R7ITR
A1
EXIT
TO 3,4

69. MPBUDC - MONITOR PREFIX TO BUDCCP AND BUOBCP.

SEE FLOWCHART.

INPUT

MNBUDC IS QUEUED WITH AN UPPER CORE BLOCK NUMBER FOR PROCESSING.

OUTPUT

MPBUDC SETS UP A BLOCK OF RADAR DATA FOR BUDCCP OR BUOBCP.

METHOD

MPBUDC SUPPRESSES THE DIFFERENTIAL CORRECTION PROCESSORS. THE INPUT AND OUTPUT LOCATIONS FOR BUDCCP OR BUOBCP ARE INITIALIZED AND THE RADAR DATA IS MOVED INTO THE INPUT BUFFER. CONTROL IS THEN PASSED TO BUOBCP FOR ORBIT PROCESSING OR TO BUDCCP FOR REENTRY PROCESSING.

USAGE

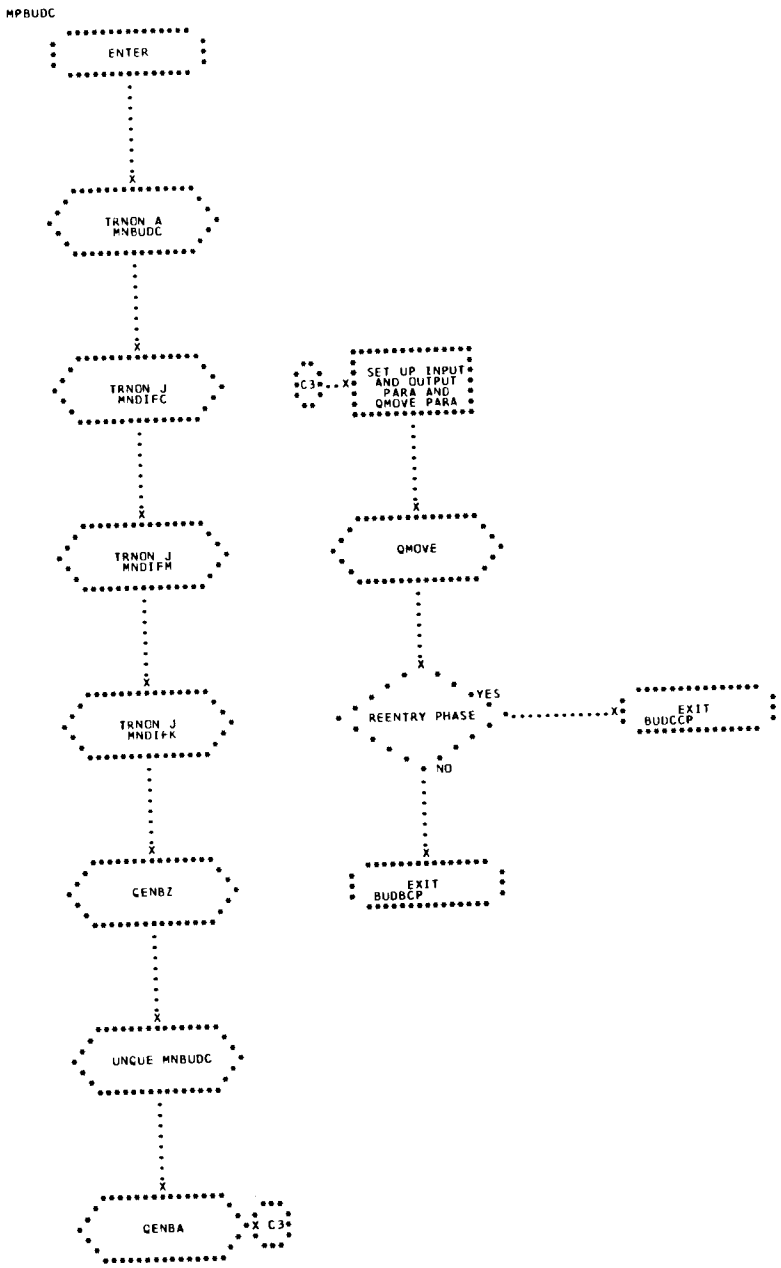
MPBUDC IS ENTERED FROM MOPRIO AND EXITS TO BUOBCP OR BUDCCP.

A) STORAGE REQUIRED -
   387 LOCATIONS INCLUDING INPUT AND OUTPUT BUFFERS
B) COMMUNICATION CELLS -
   MCDCCP,
   MCPHSE
C) TABLES - TMDATA
D) MACROS - TRNON, QENBZ, QENBA, QMOVE

E) PARAMETERS -
A, J, MNBUDC, MNDIFC, MNDIFM, MNDIFK, MNUCBL, AND MNBTOA

F) CONSTANTS - K00003

G) TIME REQUIRED -
.028 MILLISECONDS EXCLUSIVE OF TIME REQUIRED FOR UNQUE AND QMOVE MACROS.


MPBUDC
ENTER
TRNON A MNBUDC
TRNON J MNDIFC
TRNON J MNDIFM
TRNON J MNDIFK
GENBZ
UNQUE MNBUDC
GENBA
C3
C3
SET UP INPUT AND OUTPUT PARA AND QMOVE PARA
QMOVE
REENTRY PHASE
YES
EXIT BUDCCP
NO
EXIT BUDBCP

70. BUDCCP - REENTRY PHASE BACK UP TO DIFFERENTIAL CORRECTION PROGRAM.

BUDCCP OBTAINS SIX POSITION AND VELOCITY VECTORS OVER A GIVEN TIME SPAN UTILIZING LEAST SQUARES SMOOTHING TECHNIQUES ON LOW SPEED RADAR DATA FROM A SINGLE STATION. IT ALSO COMPUTES GEODETIC LATITUDE, LONGITUDE, GEODETIC HEIGHT, STATISTICAL INFORMATION AND LANDING FOOTPRINT DATA ASSOCIATED WITH THE VECTORS. (SEE FLOWCHART.)

INPUT

A) TABLE -

A 203-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L(INPUT TABLE)	STATION NUMBER j,,PASS NUMBER
+1	NUMBER OF RADAR MESSAGES IN THIS BLOCK
+2	NOT USED
+3	TIME T_1,FIXED POINT (MIN.,,SEC.)
+4	RANGE r_1 , FLOATING POINT GEMINI LENGTH UNITS
+5	AZIMUTH a_1 , FLOATING POINT RADIANS
+6	ELEVATION e_1 , FLOATING POINT RADIANS
+7	T_2 SECOND RADAR MESSAGE
+8	r_2 SECOND RADAR MESSAGE
+9	a_2 SECOND RADAR MESSAGE
+10	e_2 SECOND RADAR MESSAGE
...	ETC., UP TO 50 MESSAGES

B) CONSTANTS -

K005.0	FLOATING POINT FIVE
5006.0	FLOATING POINT SIX
K120.0	FLOATING POINT ONE HUNDRED AND TWENTY
KMNMK1	OCTAL MASK, -377777777777

C) COMMUNICATION CELL -

MCDCCP	CONTAINS L(INPUT TABLE) IN THE ADDRESS, L(OUTPUT TABLE) IN THE DECREMENT

D) SUBROUTINES -

SUBROUTINES BULTOI AND BUMAIN ARE USED.

OUTPUT

A) A 146-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L(OUTPUT TABLE)	STATION NUMBER j, FIXED POINT
+1	SLIDE, FLOATING POINT SECONDS

THE REMAINING 144 LOCATIONS CONSTITUTE 6 SUB-TABLES, EACH 24 LOCATIONS LONG. (REFER TO THE DOCUMENTATION OF SUBROUTINE BUMAIN FOR A DETAILED DESCRIPTION OF THESE 24-LOCATION OUTPUT TABLES.)

B) COMMUNICATION CELL -

MCBUER	NONZERO IF AN ERROR RETURN FROM SUBROUTINE BULTOI OCCURRED, OTHERWISE ZERO

METHOD

WHEN BUDCCP OBTAINS CONTROL FROM ITS PREFIX, MPBUDC, THE CONTENTS OF THE INDEX REGISTERS USED ARE SAVED.

NEXT, SUBROUTINE BULTOI CONVERTS THE RADAR MESSAGES

FROM LOCAL COORDINATES TO INERTIAL COORDINATES.
BUDCCP THEN COMPUTES THE SLIDE, USING

$$\text{SLIDE} = \frac{T_N - (T_1 + 120)}{5}$$

WHERE

T_1 = THE TIME OF THE FIRST RADAR MESSAGE
T_N = THE TIME OF THE LAST RADAR MESSAGE

THE SLIDE IS MADE TO BE AT LEAST 6 SECONDS. HAVING OBTAINED A SLIDE, THE TOTAL TIME SPAN IS DIVIDED INTO 6 INTERVALS, EACH INTERVAL BEING TWO MINUTES LONG, USING THE FOLLOWING FORMULAE -

LET T_{Fi} BE THE START OF THE ITH INTERVAL
T_{Li} BE THE END OF THE ITH INTERVAL.

$$T_{F1} = T_1$$
$$T_{L1} = T_{F1} + 120$$
$$T_{Fi} = T_{F(i-1)} + \text{SLIDE} \quad \text{WHEN } i = 2, 3, 4, 5, 6$$
$$T_{Li} = T_{L(i-1)} + \text{SLIDE} \quad \text{WHEN } i = 2, 3, 4, 5, 6$$

IF T_{Fi}AND/ORT_{Li}($i = 1, 2, \ldots, 6$) IS GREATER THAN T_N, THEN T_N REPLACES THOSE T_{Fi}AND/ORT_{Li}.

THE LOCATIONS OF T_{Fi} AND T_{Li} (i=1,2,...,6) ARE OBTAINED, AND A TEST FOR INTERVALS CONTAINING IDENTICAL MESSAGES IS PERFORMED. IF THERE ARE TWO OR MORE IDENTICAL INTERVALS, ONES ARE PLACED IN EACH BIT OF ALL 24 LOCATIONS OF THE OUTPUT TABLE(S) CORRESPONDING TO THE SECOND (AND LATER) OF THE IDENTICAL INTERVAL(S).

SUBROUTINE BUMAIN THEN OBTAINS THE POSITION AND VELOCITY VECTORS, GEODETIC LATITUDE, LONGITUDE, GEODETIC HEIGHT, ASSOCIATED STATISTICAL INFORMATION, AND A FOOTPRINT FOR EACH DISTINCT INTERVAL.

THE CONTENTS OF THE INDEX REGISTERS USED ARE RESTORED AND CONTROL IS TRANSFERRED TO THE SUFFIX, MFBUDC.

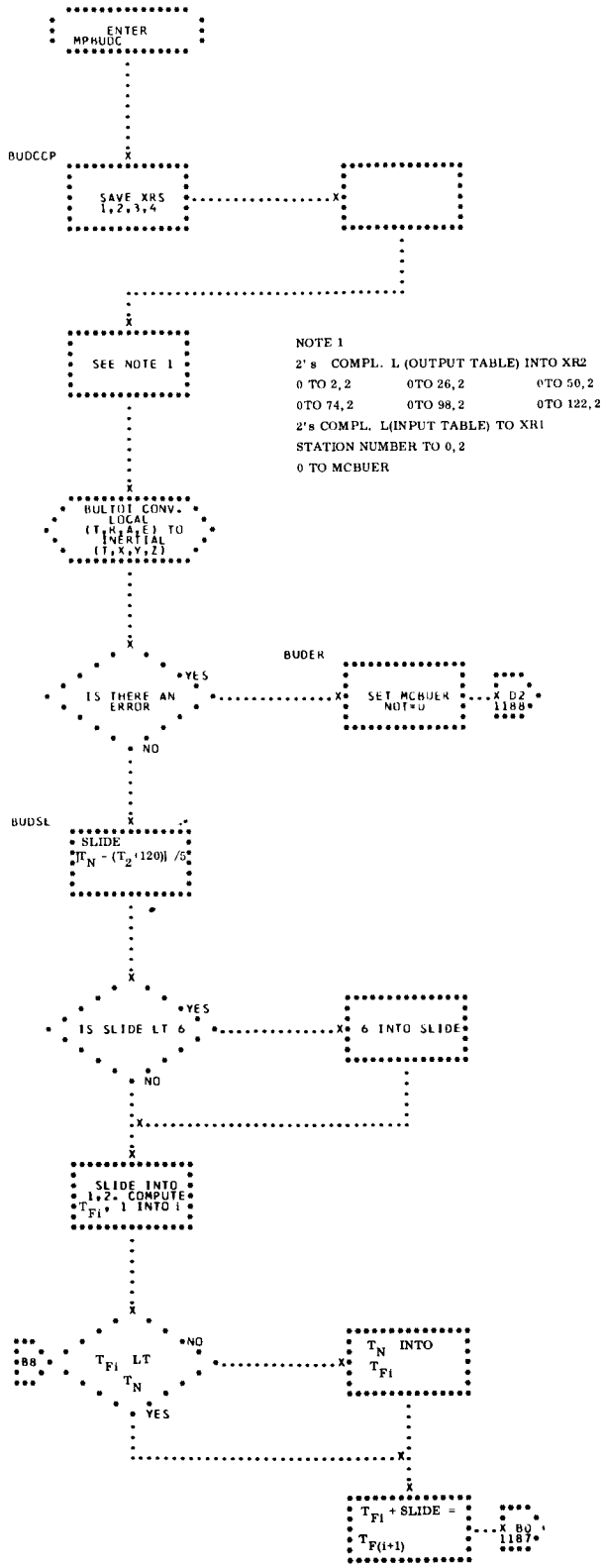
USAGE

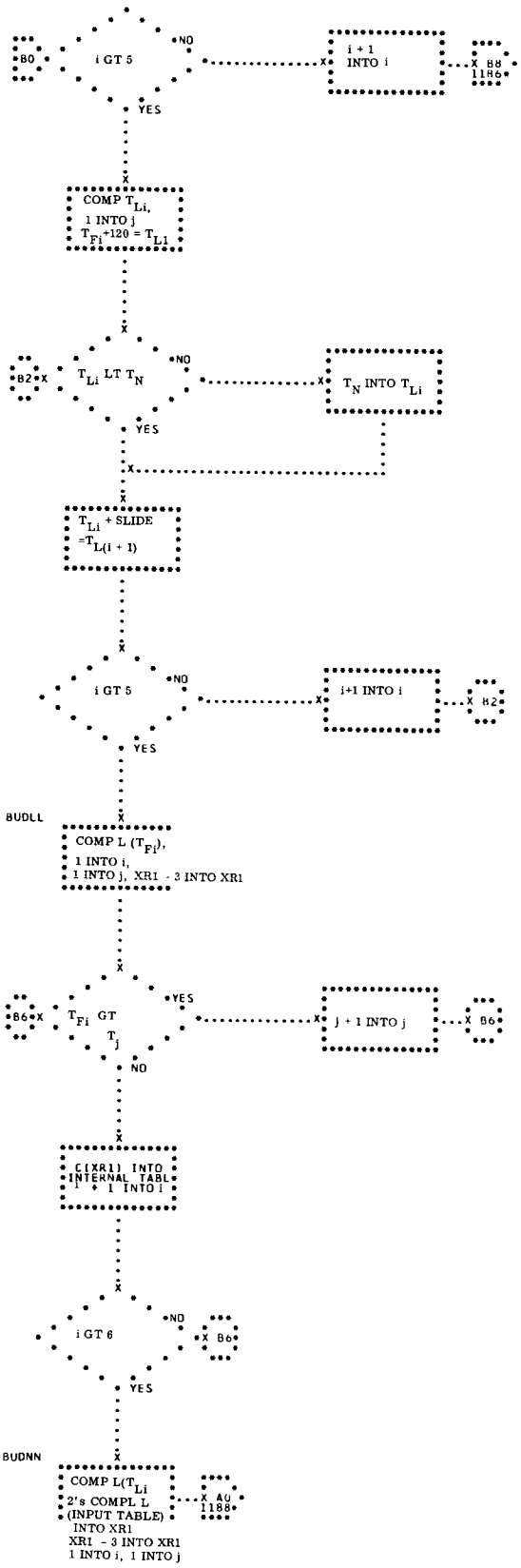
A) BUDCCP IS ENTERED FROM ITS PREFIX MPBUDC, IT TRANSFERS CONTROL TO ITS SUFFIX MFBUDC.

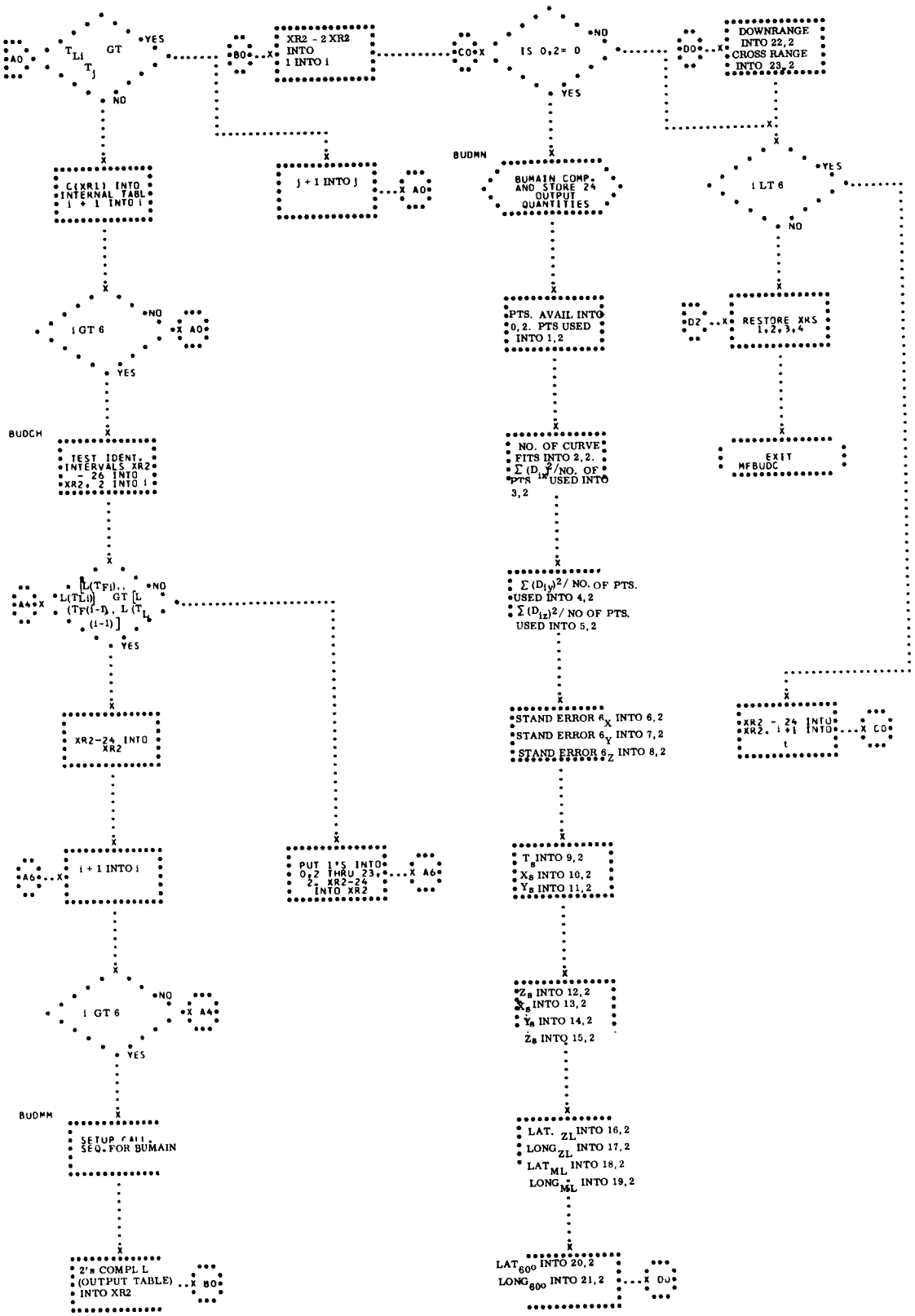
B) STORAGE REQUIRED - 117 LOCATIONS.

C) TIMING (7094) (EXCLUDING SUBROUTINE BUFPPR) - 305.658 MS (BASED UPON 50 RADAR MESSAGES, EACH ONE 6 SECONDS APART, 6 DIFFERENT INTERVALS CONTAINING 21 MESSAGES EACH, NO MESSAGES ELIMINATED BY BUEDIT).

D) ERROR CONDITIONS - COMMUNICATION CELL MCBUER WILL BE SET NONZERO IF THERE IS AN ERROR RETURN FROM SUBROUTINE BULTOI. IN THIS EVENT THERE IS NO VALID OUTPUT FROM THE PROGRAM.


ENTER MPBUDC
BUDCCP
SAVE XRS 1,2,3,4
SEE NOTE 1
NOTE 1
2's COMPL. L (OUTPUT TABLE) INTO XR2
0 TO 2,2
0TO 26,2
0TO 50,2
0TO 74,2
0TO 98,2
0TO 122,2
2's COMPL. L(INPUT TABLE) TO XR1
STATION NUMBER TO 0,2
0 TO MCBUER
BUILTOI CONV. LOCAL (T,R,A,E) TO INERTIAL (T,X,Y,Z)
BUDER
IS THERE AN ERROR
YES
NO
SET MCBUER NOT=0
X D2 1188
BUDSL
SLIDE [T_N - (T_2 '120)] /5
IS SLIDE LT 6
YES
NO
6 INTO SLIDE
SLIDE INTO 1,2. COMPUTE T_Fi, 1 INTO i
88
T_Fi LT T_N
NO
YES
T_N INTO T_Fi
T_Fi + SLIDE = T_F(i+1)
X B0 1187


B0
i GT 5
NO
i + 1 INTO i
X B0 1186
YES
COMP T_{Li}, 1 INTO j T_{Fi}+120 = T_{Li}
B2
T_{Li} LT T_N
NO
T_N INTO T_{Li}
YES
T_{Li} + SLIDE = $T_{L(i+1)}$
i GT 5
NO
i+1 INTO i
X B2
YES
BUDLL
COMP L (T_{Fi}), 1 INTO i, 1 INTO j, XR1 - 3 INTO XR1
B6
T_{Fi} GT T_j
YES
j + 1 INTO j
X B6
NO
C(XR1) INTO INTERNAL TABL i + 1 INTO i
i GT 6
NO
X B6
YES
BUDNN
COMP L(T_{Li} 2's COMPL L (INPUT TABLE) INTO XR1 XR1 - 3 INTO XR1 1 INTO i, 1 INTO j
X A0 1188


A0
T_{Li} GT T_j
YES
NO
B0
XR2 - 2 XR2 INTO 1 INTO i
C0
IS 0,2= 0
NO
YES
D0
DOWNRANGE INTO 22,2 CROSS RANGE INTO 23,2
C(XR1) INTO INTERNAL TABL i + 1 INTO i
j + 1 INTO j
BUDMN
BUMAIN COMP. AND STORE 24 OUTPUT QUANTITIES
i LT 6
i GT 6
PTS. AVAIL INTO 0,2. PTS USED INTO 1,2
D2
RESTORE XHS 1,2,3,4
BUDCH
TEST IDENT. INTERVALS XR2 - 26 INTO XR2, 2 INTO i
NO. OF CURVE FITS INTO 2,2. $\Sigma (D_{1X})^2$/NO. OF PTS USED INTO 3,2
EXIT MFBUDC
A4
L(TFi), L(TLi) GT [L (TF(i-1)), L (TL (i-1))]
$\Sigma (D_{1Y})^2$/ NO. OF PTS. USED INTO 4,2
$\Sigma (D_{1Z})^2$/ NO OF PTS. USED INTO 5,2
XR2-24 INTO XR2
STAND ERROR σ_X INTO 6,2
STAND ERROR σ_Y INTO 7,2
STAND ERROR σ_Z INTO 8,2
XR2 - 24 INTO XR2, i + 1 INTO i
A6
i + 1 INTO i
PUT 1'S INTO 0_{i2} THRU 23, 2_i XR2-24 INTO XR2
T_s INTO 9,2
X_s INTO 10,2
Y_s INTO 11,2
Z_s INTO 12,2
$\dot{X}_s$ INTO 13,2
$\dot{Y}_s$ INTO 14,2
$\dot{Z}_s$ INTO 15,2
BUDMM
SETUP CALL SEQ. FOR BUMAIN
LAT. $_{ZL}$ INTO 16,2
$LONG_{ZL}$ INTO 17,2
LAT_{ML} INTO 18,2
$LONG_{ML}$ INTO 19,2
2's COMPL (OUTPUT TABLE) INTO XR2
LAT_{600} INTO 20,2
$LONG_{600}$ INTO 21,2

71. BUCBCP - ORBIT PHASE BACK UP TO DIFFERENTIAL CORRECTION PROGRAM

BUCBCP CONVERTS UP TO FIFTY RADAR MESSAGES FROM LOCAL (t, r, a, e) TO INERTIAL (T, X, Y, Z), DETERMINES THE INTERVAL CONSISTING OF THE LAST TWO MINUTES OF DATA OR LESS, GENERATES A POSITION AND VELOCITY VECTOR, GEODETIC LATITUDE, LONGITUDE, AND GEODETIC HEIGHT, ASSOCIATED STATISTICAL INFORMATION, AND SIX CLASSICAL ELEMENTS FOR THE SOLUTION TIME OF THIS INTERVAL. (SEE FLOWCHART)

INPUT

A) TABLE -

A 203-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L(INPUT TABLE)	STATION NUMBER j,, PASS NUMBER
+1	NUMBER OF RADAR MESSAGES IN THIS BLOCK
+2	NOT USED
+3	TIME t_1 FIXED POINT (MIN.,,SEC)
+4	RANGE r_1 FLOATING POINT GEMINI LENGTH UNITS
+5	AZIMUTH a_1 FLOATING POINT RADIANS
+6	ELEVATION e_1 FLOATING POINT RADIANS
+7	t_2 SECOND RADAR MESSAGE
+8	r_2 SECOND RADAR MESSAGE
+9	a_2 SECOND RADAR MESSAGE
+10	e_2 SECOND RADAR MESSAGE
...	ETC., UP TO 50 MESSAGES

B) CONSTANTS -

K0C004	FIXED POINT FOUR
K120.0	FLOATING POINT ONE HUNDRED AND TWENTY
KCGUNM	K_1 CONVERTS GEMINI LENGTH UNITS INTO NAUTICAL MILES, $.34439336X10^4$
K.ERGU	R_s RADIUS OF THE SPHERICAL EARTH IN GEMINI LENGTH UNITS, .99924461

C) COMMUNICATION CELLS -

MCDCCP	CONTAINS L(INPUT TABLE) IN THE ADDRESS, L(OUTPUT TABLE) IN THE DECREMENT

D) SUBROUTINES-BULTOI, BUMAIN AND BUKEPE ARE USED.

OUTPUT

A) TABLE -

A 27-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L(OUTPUT TABLE)	j	STATION NUMBER, FIXED POINT
+1	T_F	TIME OF FIRST RADAR MESSAGE IN INTERVAL USED FLOATING POINT SECONDS
+2	T_N	TIME OF LAST RADAR MESSAGE IN INTERVAL USED, FLOATING POINT SECONDS
+3	P	RADAR MESSAGES AVAILABLE, FIXED POINT
+4	M	RADAR MESSAGES USED, FIXED POINT
+5	N	NO. OF CURVE FITS, FIXED POINT
+6	σ_x	X-STANDARD ERROR, FLOATING POINT YARDS

+7	σ_y	Y-STANDARD ERROR, FLOATING POINT YARDS
+8	σ_Z	Z-STANDARD ERROR, FLOATING POINT YARDS
+9	ϕ_s	GEODETIC LATITUDE AT SOLUTION TIME, FLOATING POINT RADIANS
+10	λ_s	LONGITUDE AT SOLUTION TIME, FLOATING POINT RADIANS
+11	H_s	GEODETIC HEIGHT AT SOLUTION TIME, FLOATING POINT FEET
+12	T_s	SOLUTION TIME, FLOATING POINT SECONDS
+13	X_s	X-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI LENGTH UNITS
+14	Y_s	Y-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI LENGTH UNITS
+15	Z_s	Z-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI LENGTH UNITS
+16	$\dot{X}_s$	X-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI VELOCITY UNITS
+17	$\dot{Y}_s$	Y-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI VELOCITY UNITS
+18	$\dot{Z}_s$	Z-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI VELOCITY UNITS
+19	a	SEMI-MAJOR AXIS, FLOATING POINT NAUTICAL MILES

+20	e	ECCENTRICITY, DIMENSIONLESS
+21	ν	TRUE ANOMALY, FLOATING POINT RADIANS
+22	i	INCLINATION ANGLE, FLOATING POINT RADIANS
+23	Ω	LONGITUDE OF THE ASCENDING NODE, FLOATING POINT RADIANS
+24	ω	ARGUMENT OF PERIGEE, FLOATING POINT RADIANS
+25	H_{ap}	HEIGHT AT APOGEE, FLOATING POINT NAUTICAL MILES
+26	H_{per}	HEIGHT AT PERIGEE, FLOATING POINT NAUTICAL MILES

B) COMMUNICATION CELL -

MCBUER NON-ZERO IS AN ERROR RETURN FROM SUBROUTINE BULTOI OCCURRED, OTHERWISE ZERO

METHOD

WHEN BUOBCP OBTAINS CONTROL FROM ITS PREFIX, MPBUDC, THE CONTENTS OF THE INDEX REGISTERS USED ARE SAVED.

NEXT, SUBROUTINE BULTOI CONVERTS THE RADAR MESSAGES FROM LOCAL COORDINATES TO INERTIAL COORDINATES.

BUCBCP THEN DETERMINES THE INTERVAL CONSISTING OF THE LAST TWO MINUTES OF DATA OR LESS, AND OBTAINS THE LOCATIONS OF THE TIME TAGS OF THE FIRST (T_F) AND LAST (T_N) RADAR MESSAGES IN THIS INTERVAL, USING

$$L(T_N) = L(\text{Input Table}) + 4n - 1$$

$$T_F = T_N - 120$$

$$T_1 \leq T_F \leq T_i$$

WHERE

n = NUMBER OF RADAR MESSAGES IN THE INPUT TABLE

i = 1,2,...,50

SUBROUTINE BUMAIN THEN OBTAINS THE POSITION AND VELOCITY VECTORS, GEODETIC LATITUDE, LONGITUDE, GEODETIC HEIGHT, AND ASSOCIATED STATISTICAL INFORMATION FOR THIS INTERVAL.

NEXT, SUBROUTINE BUKEPE OBTAINS SIX CLASSICAL ELEMENTS FOR THIS INTERVAL.

IF AND HAVE NOT BEEN COMPUTED SUCCESSFULLY CONTROL PASSES TO THE NEXT PARAGRAPH. IF AND HAVE BEEN OBTAINED, BUOBCP COMPUTES THE HEIGHTS AT APOGEE AND PERIGEE, USING

$$H_{ap.} = a(1 + e) - K_1 R_s$$
$$H_{per.} = a(1 - e) - K_1 R_s$$

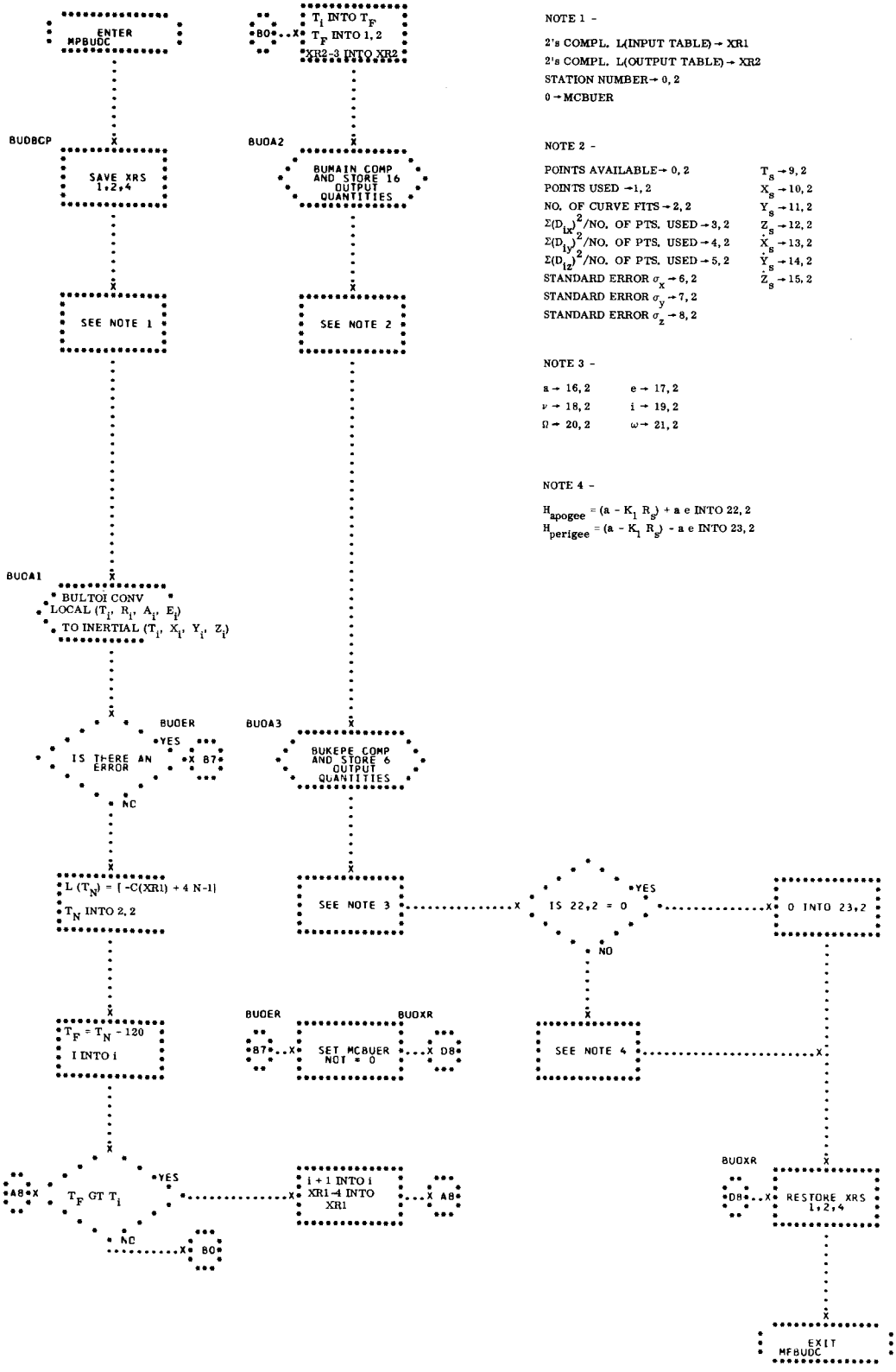
FINALLY, THE CONTENTS OF THE INDEX REGISTERS USED ARE RESTORED AND CONTROL IS TRANSFERRED TO THE SUFFIX, MFBUDC.

USAGE

A) STORAGE REQUIRED - 67 LOCATIONS.

B) TIMING (7094) (INCLUDING ALL SUBROUTINES) - 140.325 (BASED UPON 50 RADAR MESSAGES IN THE INPUT TABLE, AND NO MESSAGES ELIMINATED BY THE EDIT ROUTINE BUEDIT)

C) ERROR CONDITIONS - COMMUNICATION CELL MCBUER WILL BE SET NONZERO IF THERE IS AN ERROR RETURN FROM SUBROUTINE BULTOI. IN THIS EVENT THERE IS NO VALID OUTPUT FROM THE PROGRAM.



NOTE 1 -

2's COMPL. L(INPUT TABLE) → XR1
2's COMPL. L(OUTPUT TABLE) → XR2
STATION NUMBER → 0, 2
0 → MCBUER

NOTE 2 -

POINTS AVAILABLE → 0, 2	T_s → 9, 2
POINTS USED → 1, 2	X_s → 10, 2
NO. OF CURVE FITS → 2, 2	Y_s → 11, 2
$\Sigma(D_{ix})^2$/NO. OF PTS. USED → 3, 2	Z_s → 12, 2
$\Sigma(D_{iy})^2$/NO. OF PTS. USED → 4, 2	$\dot{X}_s$ → 13, 2
$\Sigma(D_{iz})^2$/NO. OF PTS. USED → 5, 2	$\dot{Y}_s$ → 14, 2
STANDARD ERROR σ_x → 6, 2	$\dot{Z}_s$ → 15, 2
STANDARD ERROR σ_y → 7, 2	
STANDARD ERROR σ_z → 8, 2	

NOTE 3 -

a → 16, 2	e → 17, 2
ν → 18, 2	i → 19, 2
Ω → 20, 2	ω → 21, 2

NOTE 4 -

$H_{apogee} = (a - K_1 R_s) + a\,e$ INTO 22, 2
$H_{perigee} = (a - K_1 R_s) - a\,e$ INTO 23, 2

72. BULTOI - LOCAL TO INERTIAL CONVERSION PROGRAM

BULTCI CONVERTS ONE OR MORE SETS OF LOCAL COORDINATES OF POSITION, r_i, a_i, e_i AT TIMES T_i, TO INERTIAL COORDINATES OF POSITION, X_i, Y_i, Z_i AT TIMES T_i. THIS PROGRAM IS A SUBROUTINE OF THE ORBIT PHASE AND REENTRY PHASE BACK UP TO DIFFERENTIAL CORRECTION PROGRAMS BUDCCP AND BUCBCP. (SEE FLOWCHART)

INPUT

A) TABLES -

A 203-LOCATION TABLE (THE SIZE OF THE INPUT TABLE NEED NOT BE RESTRICTED TO 203 LOCATIONS.) ARRANGED IN THE FOLLOWING FORMAT -

L(INPUT TABLE)	STATION NUMBER j ,,PASS NUMBER
+1	NUMBER OF RADAR MESSAGES IN THIS BLOCK
+2	NOT USED
+3	TIME T_1, FIXED POINT (MIN.,,SEC.)
+4	RANGE r_1, FLOATING POINT GEMINI LENGTH UNITS
+5	AZIMUTH a_1 , FLOATING POINT RADIANS
+6	ELEVATION e_1 , FLOATING POINT RADIANS
+7	T_2 SECOND RADAR MESSAGE
+8	r_2 SECOND RADAR MESSAGE
+9	a_2 SECOND RADAR MESSAGE
+10	e_2 SECOND RADAR MESSAGE
...	ETC., UP TO 50 MESSAGES

STATION CHARACTERISTICS TABLE - A COMPLETE DESCRIPTICN OF THE STATION CHARACTERISTICS TABLE IS NOT GIVEN HERE, BUT MAY BE FOUND IN OTHER DOCUMNETATION. ONLY THE LOCATIONS USED BY BULTOI ARE DESCRIBED BELOW -

L(STATICN CHARACTERISTICS TABLE)

•
•

+5 λ_{STA} GEODETIC LONGITUDE, FLOATING POINT RADIANS

•
•
•

+8 SIN ϕ, FLOATING POINT RADIANS

+9 COS ϕ, FLOATING POINT RADIANS

•
•
•

+27 R'SIN (ϕ-ϕ'), FLOATING POINT GEMINI LENGTH UNITS

+28 R'COS (ϕ-ϕ'), FLOATING POINT GEMINI LENGTH UNITS

WHERE ϕ IS THE GEODETIC LATITUDE OF THE RADAR, ϕ' IS THE GECCENTRIC LATITUDE OF THE PROJECTION OF THE RADAR ONTO THE EARTH, AND R' IS THE RADIUS OF THE EARTH MEASURED FROM THE CENTER OF THE EARTH TO THE PROJECTION OF THE RADAR.

TMSTCH - THE ADDRESS FIELD OF TMSTCH +j CONTAINS THE (FIRST LOCATION OF THE STATION CHARACTERISTICS BLCCK OF THE JTH STATION) MINUS 1.

B) SUBROUTINES -

LIBRARY SUBROUTINE U1SICO IS USED

C) CONSTANTS -

K00C00	ZERO
K00C01	FIXED POINT ONE
K00060	FIXED POINT SIXTY
KCF233	OCTAL MASK FOR FLOATING POINT CHARACTERISTIC, 233000000000
KLAMDO	INERTIAL LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH
K.OMES	ROTATIONAL VELOCITY OF THE EARTH IN FLOATING POINT RADIANS PER SECOND, .72921151 X 10

OUTPUT

A 203-LOCATION TABLE (THE SIZE OF THE OUTPUT TABLE NEED NOT BE RESTRICTED TO 203 LOCATIONS.) ARRANGED IN THE FOLLOWING FORMAT -

L(OUTPUT TABLE) STATION NUMBER j ,, PASS NUMBER

+1	NUMBER OF RADAR MESSAGES IN THIS BLOCK
+2	NOT USED
+3	TIME T_1, FLOATING POINT SECONDS
+4	X_1 FLOATING POINT GEMINI LENGTH UNITS
+5	Y_1 FLOATING POINT GEMINI LENGTH UNITS
+6	Z_1 FLOATING POINT GEMINI LENGTH UNITS
+7	T_2 SECOND RADAR MESSAGE
+8	X_2 SECOND RADAR MESSAGE
+9	Y_2 SECOND RADAR MESSAGE
+10	Z_2 SECOND RADAR MESSAGE
...	ETC., UP TO 50 MESSAGES

THE INPUT TABLE MAY BE USED AS OUTPUT TABLE, AND BUDCCP USES THIS OPTION.

METHOD

WHEN CONTROL IS TRANSFERRED TO BULTOI, THE CONTENTS OF THE INDEX REGISTERS USED ARE SAVED. IF THERE ARE ZERO MESSAGES IN THE INPUT TABLE, THE CONTENTS OF THE INDEX REGISTERS USED ARE RESTORED AND AN ERROR RETURN IS MADE FROM THE PROGRAM. OTHERWISE THE PROGRAM CONTINUES BY MOVING THE FIRST THREE WORDS OF THE INPUT TABLE TO THE FIRST THREE LOCATIONS OF THE OUTPUT TABLE.

THE POINTS IN THE INPUT TABLE ARE CONVERTED ONE AT A TIME UNTIL THE LAST POINT SPECIFIED HAS BEEN CONVERTED. THE FOLLOWING EQUATIONS ARE USED -

$$X = -X' \sin\lambda + \cos\lambda\ (-Y'\sin\phi + Z'\cos\phi)$$

$$Y = X'\cos\lambda + \sin\lambda\ (-Y'\sin\phi + Z'\cos\phi)$$

$$Z = Y'\cos\phi + Z'\sin\phi$$

FOR THE DERIVATION OF THESE EQUATIONS AND THE SYMBOLS USED REFER TO GRTS PROGRAMMER WORKBOOK, VOLUME III, BOOK A, PROGRAM CLSTOI.

FINALLY, THE CONTENTS OF THE INDEX REGISTERS USED ARE RESTORED AND CONTROL IS TRANSFERRED BACK TO BUDCCP.

USAGE

A) CALLING SEQUENCE -

```
TSX         BULTOI,4
PZE         L(INPUT TABLE),,L(OUTPUT TABLE)
ERRCR       RETURN
NORMAL      RETURN
```

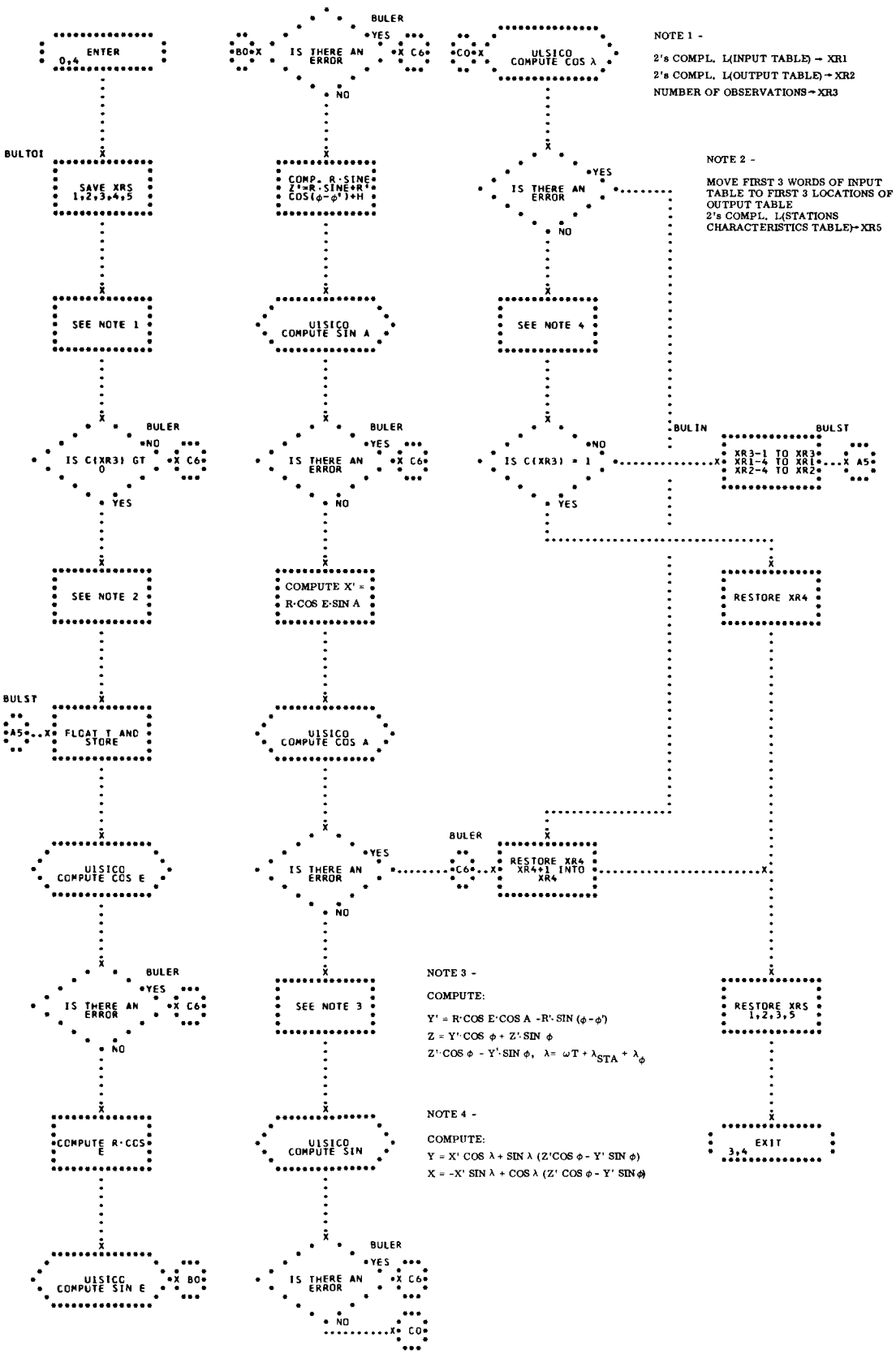
B) STORAGE REQUIRED - 116 LOCATIONS (EXCLUDING SUB-

ROUTINE U1SICO)

C) TIMING (7C94) - .059 + 2.0664P MS (INCLUDING TIME REQUIRED BY SUBROUTINE U1SICO), WHERE P = THE NUMBER OF POINTS TO BE CONVERTED PER ENTRY. MAXIMUM TIMING - 103.379 (INCLUDING TIME REQUIRED BY SUBROUTINE U1SICO, AND BASED UPON 50 POINTS PER ENTRY)

D) ERROR CONDITIONS - AN ERROR RETURN IS THE RESULT OF

1) 0 RADAR MESSAGES IN THE INPUT TABLE, OR
2) AN ERROR RETURN FROM LIBRARY SUBROUTINE U1SICO.


ENTER
0,4
BULTOI
SAVE XRS
1,2,3,4,5
SEE NOTE 1
BULER
IS C(XR3) GT
0
NO
YES
C6
SEE NOTE 2
BULST
A5
FLOAT I AND
STORE
UISICO
COMPUTE COS E
BULER
IS THERE AN
ERROR
YES
NO
C6
COMPUTE R·COS
E
UISICO
COMPUTE SIN E
B0
BULER
IS THERE AN
ERROR
YES
NO
C6
COMP. R·SINE
Z'=R·SINE·R'
COS(φ-φ')+H
UISICO
COMPUTE SIN A
BULER
IS THERE AN
ERROR
YES
NO
C6
COMPUTE X' =
R·COS E·SIN A
UISICO
COMPUTE COS A
BULER
IS THERE AN
ERROR
YES
NO
C6
SEE NOTE 3
UISICO
COMPUTE SIN
BULER
IS THERE AN
ERROR
YES
NO
C6
C0
BULER
ULSICO
COMPUTE COS λ
IS THERE AN
ERROR
YES
NO
C0
SEE NOTE 4
BULER
IS C(XR3) = 1
NO
YES
BULIN
BULST
XR3-1 TO XR3
XR1-4 TO XR1
XR2-4 TO XR2
A5
RESTORE XR4
BULER
RESTORE XR4
XR4+1 INTO
XR4
RESTORE XRS
1,2,3,5
EXIT
3,4
NOTE 1 -
2's COMPL. L(INPUT TABLE) → XR1
2's COMPL. L(OUTPUT TABLE) → XR2
NUMBER OF OBSERVATIONS → XR3
NOTE 2 -
MOVE FIRST 3 WORDS OF INPUT TABLE TO FIRST 3 LOCATIONS OF OUTPUT TABLE
2's COMPL. L(STATIONS CHARACTERISTICS TABLE) → XR5
NOTE 3 -
COMPUTE:
Y' = R·COS E·COS A -R'· SIN (φ-φ')
Z = Y'·COS φ + Z'·SIN φ
Z'·COS φ - Y'·SIN φ, λ= ωT + λSTA + λφ
NOTE 4 -
COMPUTE:
Y = X' COS λ + SIN λ (Z'COS φ - Y' SIN φ)
X = -X' SIN λ + COS λ (Z' COS φ - Y' SIN φ)

73. BUMAIN - COMPUTATION OF POSITION AND VELOCITY VECTORS AND ASSOCIATED INFORMATION

BUMAIN GENERATES AND EDITS CURVE FITS FOR X-, Y-, AND Z-COORDINATES, COMPUTES $a(t, \bar{r}, \bar{v})$, GEODETIC LATITUDE, LONGITUDE, AND GEODETIC HEIGHT AT SOLUTION TIME, AND SUPPLIES SUPPLEMENTARY STATISTICAL INFORMATION. THIS PROGRAM IS A SUBROUTINE OF THE ORBIT PHASE AND REENTRY PHASE BACK UP TO,DIFFERENTIAL CORRECTION PROGRAMS BUDCCP AND BUOBCP. (SEE FLOWCHART.)

INPUT

A) TABLE -

AN 84-LOCATION TABLE, (THE SIZE OF THE INPUT TABLE NEED NOT BE RESTRICTED TO 84 LOCATIONS.) ARRANGED IN THE FOLLOWING FORMAT -

$L(T_F)$	TIME T_1, FLOATING POINT SECONDS
+1	X_1, FLOATING POINT GEMINI LENGTH UNITS
+2	Y_1, FLOATING POINT GEMINI LENGTH UNITS
+3	Z_1, FLOATING POINT GEMINI LENGTH UNITS
+4	T_2, SECOND RADAR MESSAGE
+5	X_2 SECOND RADAR MESSAGE
+6	Y_2, SECOND RADAR MESSAGE
+7	Z_2, SECOND RADAR MESSAGE
...	ETC., UP TO 21 MESSAGES

$L(T_L)$ IS THE LOCATION OF THE TIME OF THE LAST RADAR MESSAGE TO BE USED.

B) SUBROUTINES -

SUBROUTINES BUCFIT, BUEDIT, BUGSCP, AND BUFPPR ARE USED.

C) CONSTANTS -

K00000	ZERO
K00001	FIXED POINT ONE
K00004	FIXED POINT FOUR
K002.0	FLOATING POINT TWO
K060.0	FLOATING POINT SIXTY
K120.0	FLOATING POINT ONE HUNDRED TWENTY
KCH233	OCTAL MASK FOR FLOATING POINT CHARACTERISTIC, 233000000000
K00.UT	CONVERSION FROM SECONDS TO GEMINI TIME UNITS, +.80681364E3
KGUYDS	CONVERSION FROM GEMINI LENGTH UNITS TO YARDS, +.6975246E7

OUTPUT

A 24-LOCATION TABLE, (THE OUTPUT TABLE WILL CONSIST OF THE FIRST 16 LOCATIONS ONLY IF SUBROUTINE BUFPPR IS NOT USED) ARRANGED IN THE FOLLOWING FORMAT -

L(OUTPUT TABLE)	P	POINTS AVAILABLE, FIXED POINT
+1	M	POINTS USED, FIXED POINT
+2	N	NO. OF CURVE FITS, FIXED POINT
+3	σ_x	X-STANDARD ERROR, FLOATING POINT YARDS
+4	σ_y	Y-STANDARD ERROR, FLOATING POINT YARDS
+5	σ_z	Z-STANDARD ERROR, FLOATING POINT YARDS
+6	ϕ_s	GEODETIC LATITUDE AT SOLUTION TIME, FLOATING POINT RADIANS
+7	λ_s	LONGITUDE AT SOLUTION TIME, FLOATING POINT RADIANS

+8	H_s	GEODETIC HEIGHT AT SOLUTION TIME, FLOATING POINT FEET
+9	T_s	SOLUTION TIME, FLOATING POINT SECONDS
+10	X_s	X-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI LENGTH UNITS
+11	Y_s	Y-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI LENGTH UNITS
+12	Z_s	Z-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI LENGTH UNITS
+13	$\dot{X}_s$	-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI VELOCITY UNITS
+14	$\dot{Y}_s$	-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI VELOCITY UNITS
+15	$\dot{Z}_s$	-COORDINATE AT SOLUTION TIME, FLOATING POINT GEMINI VELOCITY UNITS
+16	ϕ_{ZL}	LATITUDE OF MINIMUM LIFT IMPACT POINT, FLOATING POINT RADIANS
+17	λ_{ZL}	LONGITUDE OF MINIMUM LIFT IMPACT POINT, FLOATING POINT RADIANS
+18	ϕ_{ML}	LATITUDE OF MAXIMUM LIFT IMPACT POINT, FLOATING POINT RADIANS
+19	λ_{ML}	LONGITUDE OF MAXIMUM LIFT IMPACT POINT, FLOATING POINT RADIANS
+20	ϕ_{60^o}	LATITUDE OF 60 BANK ANGLE IMPACT POINT, FLOATING POINT RADIANS

+21	λ_{60^o}	LONGITUDE OF 60 BANK ANGLE IMPACT POINT, FLOATING POINT RADIANS
+22	D_{DR}	DOWN RANGE FOOTPRINT DIMENSION, FLOATING POINT NAUTICAL MILES
+23	D_{CR}	CROSS RANGE FOOTPRINT DIMENSION, FLOATING POINT NAUTICAL MILES

METHOD

A) THE CONTENTS OF THE INDEX REGISTERS USED BY BUMAIN ARE SAVED.

B) P, THE NUMBER OF INPUT MESSAGES AVAILABLE FOR USE, IS COMPUTED USING

$$P = \frac{L(T_L) - L(T_F)}{4} + 1$$

P IS STORED IN THE OUTPUT TABLE.

C) THE TIME OF EACH RADAR MESSAGE IS ADDITIVELY REDUCED MODULO T_F IN ORDER TO OBTAIN MAXIMUM EFFICIENCY IN THE CURVE FITTING ROUTINE, BUCFIT.

D) IF P IS LESS THAN 11 THE PROGRAM TRANSFERS CONTROL TO THE LOGIC OF PARAGRAPH O BELOW.

E) BUCFIT IS USED TO OBTAIN THE CURVE FIT COEFFICIENTS FOR THE X-COORDINATES. THE DATA USED BY BUCFIT IS THEN EDITED BY USE OF THE EDIT PROGRAM BUEDIT. IN THE EVENT OF AN ERROR RETURN FROM BUEDIT CONTROL PASSES TO PARAGRAPH O. IF BUEDIT REJECTS ANY RADAR MESSAGE THE FIT-EDIT CYCLE IS REPEATED ON THE REMAINING X-COORDINATES BY TRANSFERRING CONTROL BACK TO THE BEGINNING OF PARAGRAPH E.

F) BUCFIT IS USED TO OBTAIN THE CURVE FIT COEFFICIENTS FOR THE Y-COORDINATES. BUEDIT

EDITS THE DATA USED. IN THE EVENT OF A BUEDIT ERROR RETURN CONTROL PASSES TO PARAGRAPH O. IF BUEDIT REJECTS ONE OR MORE RADAR MESSAGES, A RESTART INDICATOR IS SET, AND THE FIT-EDIT CYCLE FOR THE Y-COORDINATES IS REPEATED ON THE REMAINING Y-COORDINATES BY TRANSFERRING CONTROL BACK TO THE BEGINNING OF PARAGRAPH F. IF BUEDIT DOES NOT REJECT ANY MESSAGE ON A GIVEN PASS THE RESTART INDICATOR IS CHECKED. IF IT HAS BEEN SET TO THE NULL SETTING AND CONTROL PASSES BACK TO PARAGRAPH E TO RESTART THE PROCESS ON THE REMAINING VALID MESSAGES.

G) THE PROCESS OUTLINED IN PARAGRAPH F FOR THE Y-COORDINATES IS THEN CARRIED OUT ON THE Z-COORDINATES. THUS, ANY FIT FINALLY USED AT THE CONCLUSION OF THE PROCESS HAS BEEN MADE FROM MESSAGES ALL OF WHOSE COORDINATES HAVE PASSED THE EDITING TESTS.

H) M, THE NUMBER OF MESSAGES USED IN THE FINAL FIT, AND N, THE TOTAL NUMBER OF CURVE FITS MADE, ARE STORED IN THE OUTPUT TABLE. N WOULD THUS BE THREE IF NO MESSAGES WERE REJECTED BY BUEDIT.

I) THE STANDARD ERRORS OF THE FINAL X, Y, AND Z FITS ARE STORED IN THE OUTPUT TABLE.

J) IF THE PROGRAM IS IN THE ORBIT PHASE, THE SOLUTION TIME T_s IS COMPUTED USING

$$T_{ss} = \left(\begin{array}{l}\text{integral} \\ \text{part of}\end{array} \left[\frac{\frac{T_F + T_L}{2}}{60} \right] \right) X\ 60 - T_F$$

$$T_s = T_{ss} + T_F$$

WHERE

T_F=THE (UNMODULIZED) TIME OF THE FIRST RADAR MESSAGE IN THE INPUT TABLE

T_L=THE (UNMODULIZED) TIME OF THE LAST RADAR MESSAGE IN THE INPUT TABLE

T_{ss} THE SOLUTION TIME MODULO

THE SOLUTION TIME T_s WILL BE THE INTEGRAL MINUTE NEAREST THE MIDPOINT OF THE INTERVAL. IF THE PROGRAM IS IN THE REENTRY PHASE, THE SOLUTION TIME T_s IS COMPUTED USING

$$T_{ss} = T_{+F} + K_3 (T_{+L} - T_{+F})$$

$$T_s = T_{ss} + T_F$$

WHERE

T_{+L}= THE TIME OF THE LAST RADAR MESSAGE USED IN THE FINAL FIT (MODULO T_F)

T_{+F}= THE TIME OF THE FIRST RADAR MESSAGE USED IN THE FINAL FIT (MODULO T_F)

K_3 = THE SOLUTION TIME FACTOR, CURRENTLY +.5

K) THE POSITION AND VELOCITY VECTORS AT SOLUTION TIME T_S ARE COMPUTED USING

$$C_s = A_{0C} + A_{1C}T_{ss} + A_{2C}T_{ss}^2$$

$$\dot{C}_s = (A_{1C} + 2A_{2C}T_{ss})K_2$$

WHERE

C = X,Y, OR Z

K_2 = THE NUMBER OF SECONDS IN A GEMINI TIME UNIT

L) BUGSCP IS USED TO OBTAIN THE GEODETIC LATITUDE, LONGITUDE, AND GEODETIC HEIGHT AT SOLUTION TIME T_s. IN THE EVENT OF AN ERROR RETURN FROM BUGSCP

CONTROL PASSES TO PARAGRAPH Q.

M) IF A FOOTPRINT IS REQUIRED, SUBROUTINE BUFPPR IS USED TO COMPUTE THE DATA FOR THE LAST EIGHT LOCATIONS OF THE 24-LOCATION OUTPUT TABLE. AN ERROR RETURN FROM BUFPPR CAUSES A TRANSFER TO PARAGRAPH P.

N) FINALLY, THE CONTENTS OF THE INPUT TABLE ARE RESTORED BY ADDITIVELY INCREASING THE MAGNITUDE OF THE TIME OF EACH RADAR MESSAGE MODULO T_F AND SETTING THE SIGNS OF THE TIME TAGS POSITIVE. THE CONTENTS OF THE INDEX REGISTERS USED ARE RESTORED AND BUMAIN EXITS VIA ITS NORMAL RETURN.

O) ZEROS ARE STORED IN THE 2ND THROUGH THE 16TH LOCATION OF THE OUTPUT TABLE. IF A FOOTPRINT HAS NOT BEEN REQUESTED CONTROL PASSES TO PARAGRAPH N.

P) ZEROS ARE PLACED IN THE 17TH THROUGH THE 24TH LOCATION OF THE OUTPUT TABLE AND CONTROL PASSES TO PARAGRAPH N.

Q) ZEROS ARE PLACED IN THE 7TH THROUGH 9TH LOCATION OF THE OUTPUT TABLE AND CONTROL PASSES TO PARAGRAPH M.

USAGE

A) CALLING SEQUENCE -

```
TSX      BUMAIN,4
G        L(T_F) ,, L(T_L)
PZE      **,,L(OUTPUT TABLE)
NORMAL   RETURN
```

WHERE

G = PZE IF A FOOTPRINT IS REQUIRED (24-LOCATION OUTPUT TABLE),

G = MZE IF NO FOOTPRINT IS REQUIRED (16-LOCATION OUTPUT TABLE).

B) STORAGE REQUIRED - 229 LOCATION

C) TIMING (7094) (EXCLUDING SUBROUTINE BUFPPR) -

MINIMUM (BASED UPON 21 MESSAGES, NO MESSAGE ELIMINATED BY THE EDIT ROUTINE) -

ORBIT PHASE - 33.583 MS

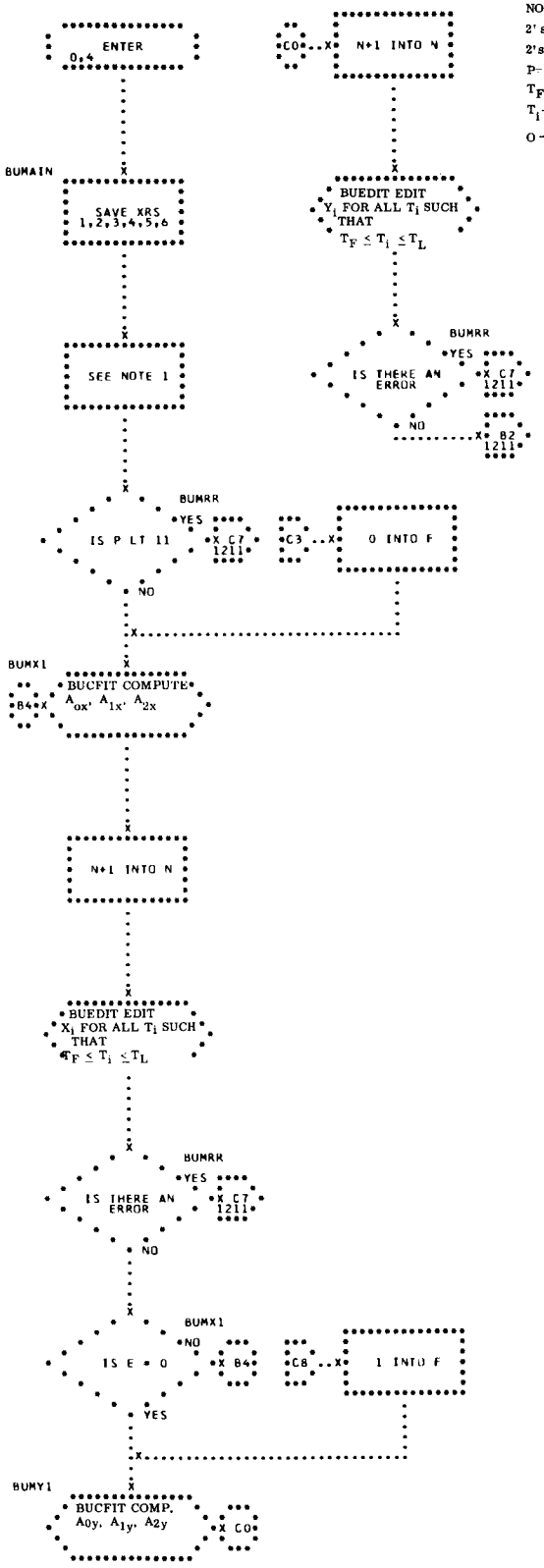
REENTRY PHASE - 33.522 MS

MAXIMUM (BASED UPON 21 MESSAGES, ONE MESSAGE ELIMINATED ON EACH EDIT PASS THROUGH THE Z-COMPONENTS, RESULTING IN A TOTAL OF 45 CURVE FITS AND 59 EDIT PASSES) -

ORBIT PHASE - 524.283 MS

REENTRY PHASE - 524.222 MS

D) ERROR CONDITIONS - THERE IS NO ERROR RETURN FROM THE PROGRAM. INTERNAL ERROR CONDITIONS ARE HANDLED AS STATED UNDER METHOD.



NOTE 1

2's COMPL. L (OUTPUT TABLE) → XR2

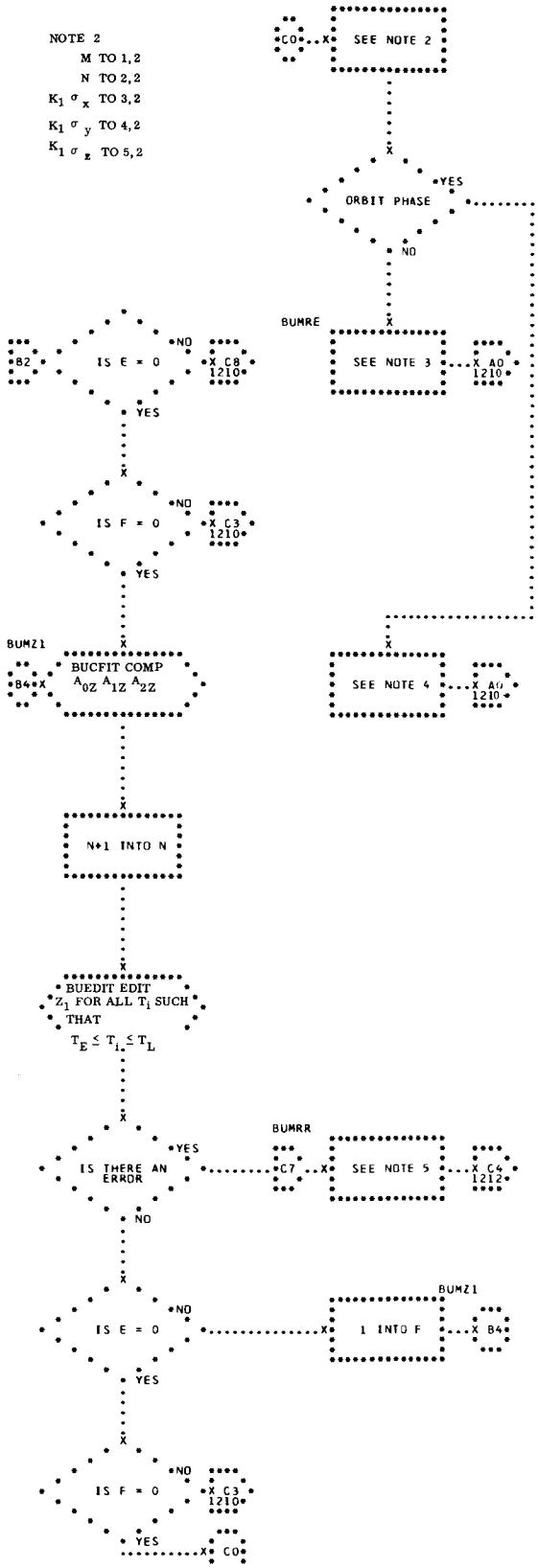
2's COMPL. $L(T_F)$ → XR1

P= [$L(T_L)$ - $L(T_F)$] /4 + 1 → 0,2

T_F → H

T_i-H → T_i WHERE $T_F \leq T_I \leq T_L$

0 → F 0 → N



NOTE 3

$T_{ss} = T_{+F} + K_3 (T_{+L} - T_{+F})$

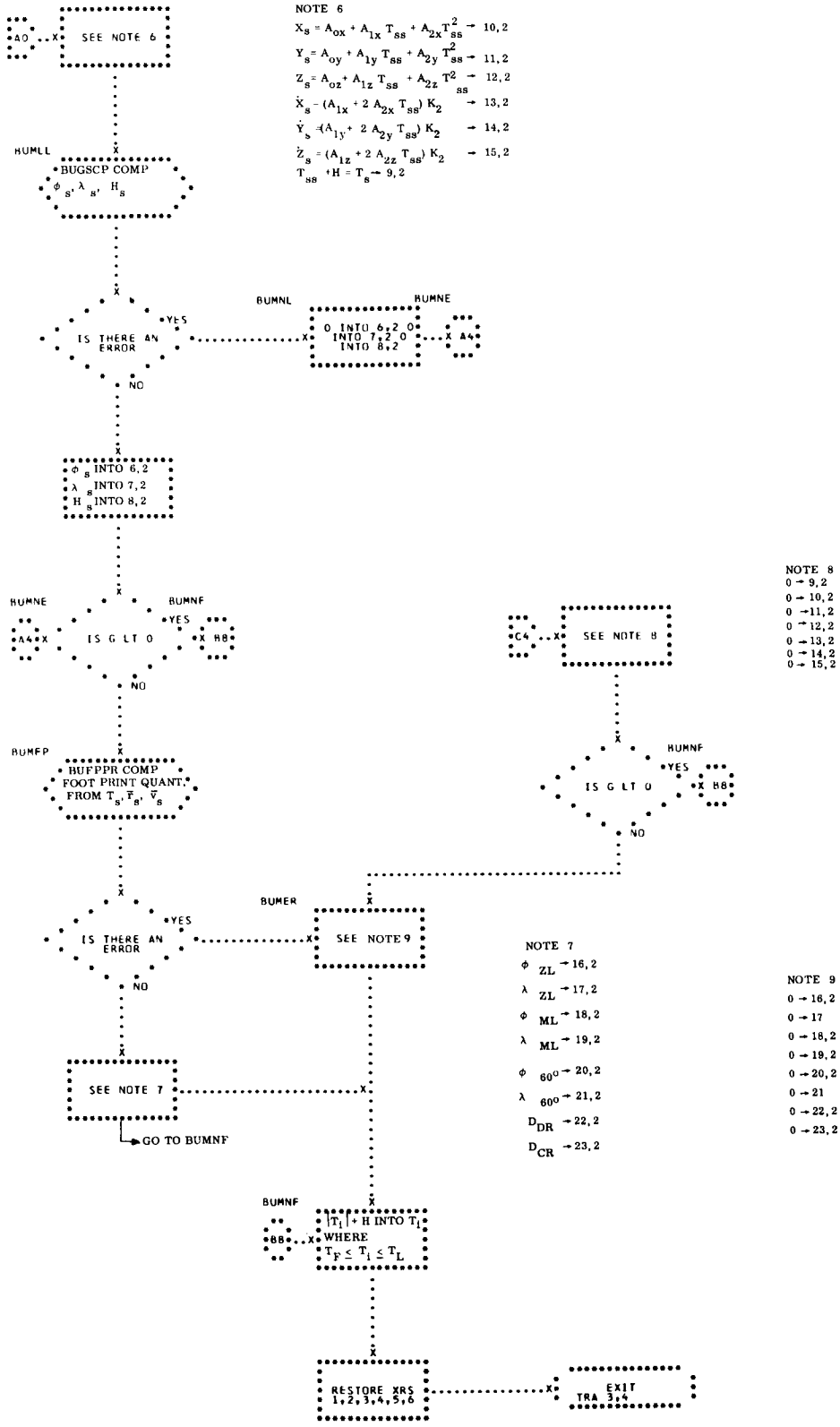
(T_{+F} = TIME OF FIRST MESSAGE USED IN FINAL FIT; T_{+L} = TIME OF LAST MESSAGE USED IN FINAL FIT)

NOTE 4

$$T_{ss} = \left(\text{INTEGRAL PART OF } \left[\frac{T_F + T_L + 2H}{120}\right] \times 60\right) - H$$

NOTE 5

0 TO 1,2
0 TO 2,2
0 TO 3,2
0 TO 4,2
0 TO 5,2
0 TO 6,2
0 TO 7,2
0 TO 8,2


A0
SEE NOTE 6
BUMLL
BUGSCP COMP
ϕ_s, λ_s, H_s
IS THERE AN ERROR
YES
NO
BUMNL
0 INTO 6,2 0 INTO 7,2 0 INTO 8,2
BUMNE
A4
ϕ_s INTO 6,2
λ_s INTO 7,2
H_s INTO 8,2
BUMNE
A4
IS G LT 0
BUMNF
YES
B8
NO
BUMFP
BUFPPR COMP FOOT PRINT QUANT. FROM $T_s, \bar{r}_s, \bar{v}_s$
IS THERE AN ERROR
YES
NO
BUMER
SEE NOTE 9
SEE NOTE 7
GO TO BUMNF
C4
SEE NOTE 8
BUMNF
IS G LT 0
YES
B8
NO
BUMNF
B8
$|T_i| + H$ INTO T_i WHERE $T_F \leq T_i \leq T_L$
RESTORE XRS 1,2,3,4,5,6
EXIT TRA 3,4
NOTE 6
$X_s = A_{ox} + A_{1x} T_{ss} + A_{2x} T_{ss}^2 \rightarrow 10,2$
$Y_s = A_{oy} + A_{1y} T_{ss} + A_{2y} T_{ss}^2 \rightarrow 11,2$
$Z_s = A_{oz} + A_{1z} T_{ss} + A_{2z} T_{ss}^2 \rightarrow 12,2$
$\dot{X}_s = (A_{1x} + 2 A_{2x} T_{ss}) K_2 \rightarrow 13,2$
$\dot{Y}_s = (A_{1y} + 2 A_{2y} T_{ss}) K_2 \rightarrow 14,2$
$\dot{Z}_s = (A_{1z} + 2 A_{2z} T_{ss}) K_2 \rightarrow 15,2$
$T_{ss} + H = T_s \rightarrow 9,2$
NOTE 8
$0 \rightarrow 9,2$
$0 \rightarrow 10,2$
$0 \rightarrow 11,2$
$0 \rightarrow 12,2$
$0 \rightarrow 13,2$
$0 \rightarrow 14,2$
$0 \rightarrow 15,2$
NOTE 7
$\phi_{ZL} \rightarrow 16,2$
$\lambda_{ZL} \rightarrow 17,2$
$\phi_{ML} \rightarrow 18,2$
$\lambda_{ML} \rightarrow 19,2$
$\phi_{60^\circ} \rightarrow 20,2$
$\lambda_{60^\circ} \rightarrow 21,2$
$D_{DR} \rightarrow 22,2$
$D_{CR} \rightarrow 23,2$
NOTE 9
$0 \rightarrow 16,2$
$0 \rightarrow 17$
$0 \rightarrow 18,2$
$0 \rightarrow 19,2$
$0 \rightarrow 20,2$
$0 \rightarrow 21$
$0 \rightarrow 22,2$
$0 \rightarrow 23,2$

74. BUCFIT - SECOND DEGREE CURVE FITTING PROGRAM FOR BUMAIN

BUCFIT FITS A SECOND DEGREE POLYNOMIAL

$$C = A_{0C} + A_{1C}\ T + A_{2C}\ T^2$$

TO THE INERTIAL C-COORDINATES AS A FUNCTION OF TIME, WHERE C = X,Y, OR Z. BUCFIT IS A SUBROUTINE FOR BUMAIN. (SEE FLOWCHART)

INPUT

A) TABLE -

AN 84-LOCATION TABLE, (THE SIZE OF THE INPUT TABLE NEED NOT BE RESTRICTED TO 84 LOCATIONS), ARRANGED IN THE FOLLOWING FORMAT -

$L(T_F)$	TIME T_1, FLOATING POINT SECONDS
+1	X_1, FLOATING POINT GEMINI LENGTH UNITS
+2	Y_1, FLOATING POINT GEMINI LENGTH UNITS
+3	Z_1, FLOATING POINT GEMINI LENGTH UNITS
+4	T_2, SECOND RADAR MESSAGE
+5	X_2, SECOND RADAR MESSAGE
+6	Y_2, SECOND RADAR MESSAGE
+7	Z_2, SECOND RADAR MESSAGE
...	ETC, UP TO 21 MESSAGES

$L(T_L)$ IS THE LOCATION OF THE TIME OF THE LAST RADAR MESSAGE TO BE USED.

$L(C_F)$ IS THE LOCATION OF C_1, WHERE C = X,Y, OR Z

B) CONSTANTS -

K001.0 FLOATING POINT ONE

OUTPUT

A 3-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L(COEFF.$_C$ TABLE)	A_{0C}, FLOATING POINT
+1	A_{1C}, FLOATING POINT
+2	A_{2C}, FLOATING POINT

METHOD

WHEN CONTROL IS TRANSFERRED TO BUCFIT, THE CONTENTS OF INDEX REGISTERS 1,2, AND 3 ARE SAVED.

NEXT A SECOND DEGREE POLYNOMIAL

$$C = A_{0C} + A_{1C} T + A_{2C} T^2$$

IS FITTED TO THE C-COORDINATE VALUES AS A FUNCTION OF TIME BY MEANS OF THE FOLLOWING THREE NORMAL EQUATIONS - (NUMERICAL MATHEMATICAL ANALYSIS, J.B. SCARBOROUGH, THE JOHNS HOPKINS PRESS, 1950, CHAPTER XVI, ART. 145.)

$$A_{0C} \sum_1^N 1 + A_{1C} \sum_1^N T_i + A_{2C} \sum_1^N T_i^2 = \sum_1^N C_i$$

$$A_{0C} \sum_1^N T_i + A_{1C} \sum_1^N T_i^2 + A_{2C} \sum_1^N T_i^3 = \sum_1^N C_i T_i$$

$$A_{0C} \sum_1^N T_i^2 + A_{1C} \sum_1^N T_i^3 + A_{2C} \sum_1^N T_i^4 = \sum_1^N C_i T_i^2$$

THE SUMS ARE COMPUTED, AND THE THREE NORMAL EQUATIONS ARE SOLVED FOR A_{0C}, A_{1C}, AND A_{2C}. ALL COMPUTATIONS ARE DONE IN SINGLE PRECISION ARITHMETIC.

THE CONTENTS OF INDEX REGISTERS 1,2, AND 3 ARE RESTORED AND CONTROL IS TRANSFERRED BACK TO BUMAIN.

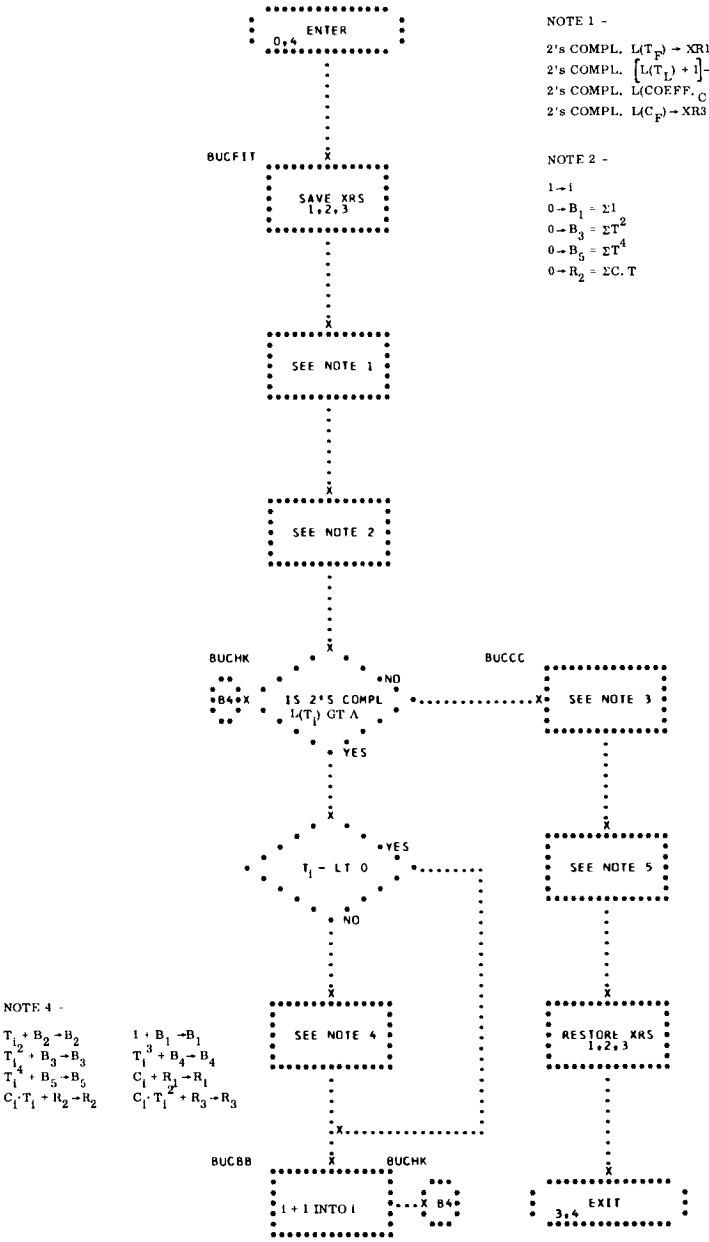
USAGE

A) CALLING SEQUENCE -

```
TSX      BUCFIT,4
PZE      L(T_F),,L(T_L)
PZE      L(COEFF._C TABLE),,L(C_F)
NORMAL   RETURN
```

B) STORAGE REQUIRED - 118 LOCATIONS

C) TIMING (7C94) - .623 + .291 P MS, WHERE P = THE NUMBER OF RADAR MESSAGES OVER WHICH THE CURVE IS TO BE FITTED PER ENTRY.
MAXIMUM TIMING - 6.730 MS (BASED UPON 21 MESSAGES PER ENTRY).



NOTE 1 -

2's COMPL. $L(T_F) \rightarrow XR1$
2's COMPL. $\left[L(T_L) + 1\right] \rightarrow A$
2's COMPL. $L(\text{COEFF.}_C \text{ TABLE}) \rightarrow XR2$
2's COMPL. $L(C_F) \rightarrow XR3$

NOTE 2 -

$1 \rightarrow i$

$0 \rightarrow B_1 = \Sigma 1$	$0 \rightarrow B_2 \quad \Sigma T$
$0 \rightarrow B_3 = \Sigma T^2$	$0 \rightarrow B_4 \quad \Sigma T^3$
$0 \rightarrow B_5 = \Sigma T^4$	$0 \rightarrow R_1 \quad \Sigma C$
$0 \rightarrow R_2 = \Sigma C \cdot T$	$0 \rightarrow R_3 \quad \Sigma C \cdot T^2$

NOTE 3 -

V_1	B_3/B_1	U_1	B_2/B_1
$-D_1$	$-B_3 + U_1 \cdot B_2$	$-D_2$	$-B_4 + U_1 \cdot B_3$
$-D_3$	$-R_2 + U_1 \cdot R_1$	$-F_1$	$-B_4 + V_1 \cdot B_2$
$-F_2$	$-B_5 + V_1 \cdot B_3$	$-F_3$	$-R_3 + V_1 \cdot R_1$
Q_1	$F_2 - F_1 \cdot D_2/D_1$		
Q_2	$F_3 - D_2 \cdot D_3/D_1$		

NOTE 4 -

$T_i + B_2 \rightarrow B_2$	$1 + B_1 \rightarrow B_1$
$T_i^2 + B_3 \rightarrow B_3$	$T_i^3 + B_4 \rightarrow B_4$
$T_i^4 + B_5 \rightarrow B_5$	$C_i + R_1 \rightarrow R_1$
$C_i \cdot T_i + R_2 \rightarrow R_2$	$C_i \cdot T_i^2 + R_3 \rightarrow R_3$

NOTE 5 -

$A_{2C} = Q_2/Q_1$ INTO 2, 2
$A_{1C} = (D_3 - F_1 \cdot A_{2C})/D_1$ INTO 1, 2
$A_{0C} = (R_1 - B_3 \cdot A_{2C} - B_2 \cdot A_{1C})/B_1$ INTO 0, 2

75. BUEDIT - EDITING PROGRAM FOR BUMAIN

BUEDIT EDITS CURVE-FITTED DATA ACCORDING TO THE FIT ON THE C-COORDINATES BY MEANS OF A 3-SIGMA TEST, WHERE C=X,Y, OR Z. BUEDIT IS A SUBROUTINE FOR BUMAIN. (SEE FLOWCHART).

INPUT

A) TABLES -

AN 84-LOCATION TABLE (THE SIZE OF THE INPUT TABLE NEED NOT BE FIXED. HOWEVER, THE MAXIMUM NUMBER OF RADAR MESSAGES TO BE EDITED CAN NOT EXCEED THE NUMBER OF LOCATIONS IN THE RESIDUALS TABLE. HENCE, IF THE LENGTH OF THE INPUT TABLE IS INCREASED, THE LENGTH OF THE RESIDUALS TABLE MUST ALSO BE INCREASED BY THE APPROPRIATE FACTOR.) ARRANGED IN THE FOLLOWING FORMAT -

$L(T_F)$	TIME T_1, FLOATING POINT SECONDS
+1	X_1, FLOATING POINT GEMINI LENGTH UNITS
+2	Y_1, FLOATING POINT GEMINI LENGTH UNITS
+3	Z_1, FLOATING POINT GEMINI LENGTH UNITS
+4	T_2 SECOND RADAR MESSAGE
+5	X_2 SECOND RADAR MESSAGE
+6	Y_2 SECOND RADAR MESSAGE
+7	Z_2 SECOND RADAR MESSAGE
...	ETC, UP TO 21 MESSAGES

$L(C_F)$ IS THE LOCATION OF C_1, WHERE C IS X, Y, OR Z.

L(COEFF.$_C$ TABLE) A_{0C}, FLOATING POINT
+1 A_{1C}, FLOATING POINT
+2 A_{2C}, FLOATING POINT

P=THE NUMBER OF RADAR MESSAGES

B) SUBROUTINES -

LIBRARY SUBROUTINE U1SQRT IS USED.

C) CONSTANTS -

K00C00 ZERO
K003.0 FLOATING POINT THREE
KCH233 OCTAL MASK FOR FLOATING POINT CHARACTERISTIC, 233000C0000

OUTPUT

A) A 3-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L(CUTPUT TABLE) σ = SIGMA, FLOATING POINT
+1 L = NUMBER OF MESSAGES ACCEPTED, FIXED POINT
+2 E = 0 IF NO MESSAGES HAVE BEEN EDITED OUT, FIXED POINT
E ≠ 0 IF ONE OR MORE MESSAGES HAVE BEEN EDITED OUT, FIXED POINT

B) THE PROGRAM ALSO SETS THE TIME TAG (CONTAINED IN THE INPUT TABLE) NEGATIVE FOR ANY OBSERVATION THAT IS REJECTED BY THE EDITING PROCESS.

METHOD

A) WHEN CONTROL IS TRANSFERRED TO BUEDIT, THE CONTENTS OF ALL INDEX REGISTERS USED ARE SAVED.

B) THE INPUT CURVE FIT COEFFICIENTS ARE USED TO COMPUTE THE MAGNITUDES OF THE RESIDUALS OF THE

SPECIFIED INPUT COORDINATES OF THE MESSAGES USING

$$D_i = \left| C_i - (A_{0C} + A_{1C} \cdot T_i + A_{2C} \cdot T_i^2) \right|$$

THESE RESIDUAL MAGNITUDES ARE STORED IN INTERNAL STORAGE. P IS USED TO DETERMINE THE MAXIMUM NUMBER OF RESIDUAL MAGNITUDES TO BE COMPUTED. IF THE TIME TAG, T , FOR ANY MESSAGE IS NEGATIVE THE RESIDUAL IS NOT COMPUTED AND THE SIGN OF THE APPROPRIATE INTERNAL STORAGE LOCATION IS SET NEGATIVE.

C) THE ONE SIGMA ERROR IS THEN COMPUTED USING THE RESIDUAL MAGNITUDES HAVING POSITIVE SIGNS AND

$$\sigma_C = \sqrt{\sum_1^M (D_i)^2/(M-1)}$$

WHERE M=NUMBER OF RESIDUAL MAGNITUDES USED.

D) EACH RESIDUAL MAGNITUDE IS THEN COMPARED WITH $3\sigma_C$. THE TIME TAGS ASSOCIATED WITH ANY RESIDUAL MAGNITUDE GREATER THAN $3\sigma_C$ ARE SET NEGATIVE TO REJECT ALL THREE COORDINATES OF THE MESSAGE FROM USE IN FURTHER CURVE FITS. THE SIGN OF THE REJECTED RESIDUAL MAGNITUDE IN THE INTERNAL STORAGE TABLE IS ALSO SET NEGATIVE.

E) IF LESS THAN SIX MESSAGES ARE LEFT UNREJECTED AFTER THIS COMPARISON THE CONTENTS OF ALL REGISTERS USED ARE RESTORED AND AN ERROR RETURN IS MADE FROM THE PROGRAM.

F) IF SIX OR MORE MESSAGES ARE LEFT AND ONE OR MORE WERE REJECTED USING THE CURRENT VALUE OF σ_C THE PROGRAM RETURNS TO THE LOGIC OF PARAGRAPH C)

ABOVE.

G) IF SIX OR MORE COORDINATES ARE LEFT AND NONE WAS REJECTED USING THE CURRENT VALUE OF σ_C THE CONTENTS OF ALL INDEX REGISTERS USED ARE RESTORED AND THE PROGRAM EXITS VIA ITS NORMAL RETURN.

USAGE

A) CALLING SEQUENCE -

```
TSX          BUEDIT,4
PZE          L(T_F),,L(OUTPUT TABLE)
PZE          L(COEFF._C TABLE),,L(C_F)
PZE          P
ERRCR        RETURN
NORMAL       RETURN
```

B) STORAGE REQUIRED - 122 LOCATIONS (EXCLUDING SUBROUTINE U1SQRT)

C) TIMING (7094) (INCLUDING TIME REQUIRED BY SUBROUTINE U1SQRT) -

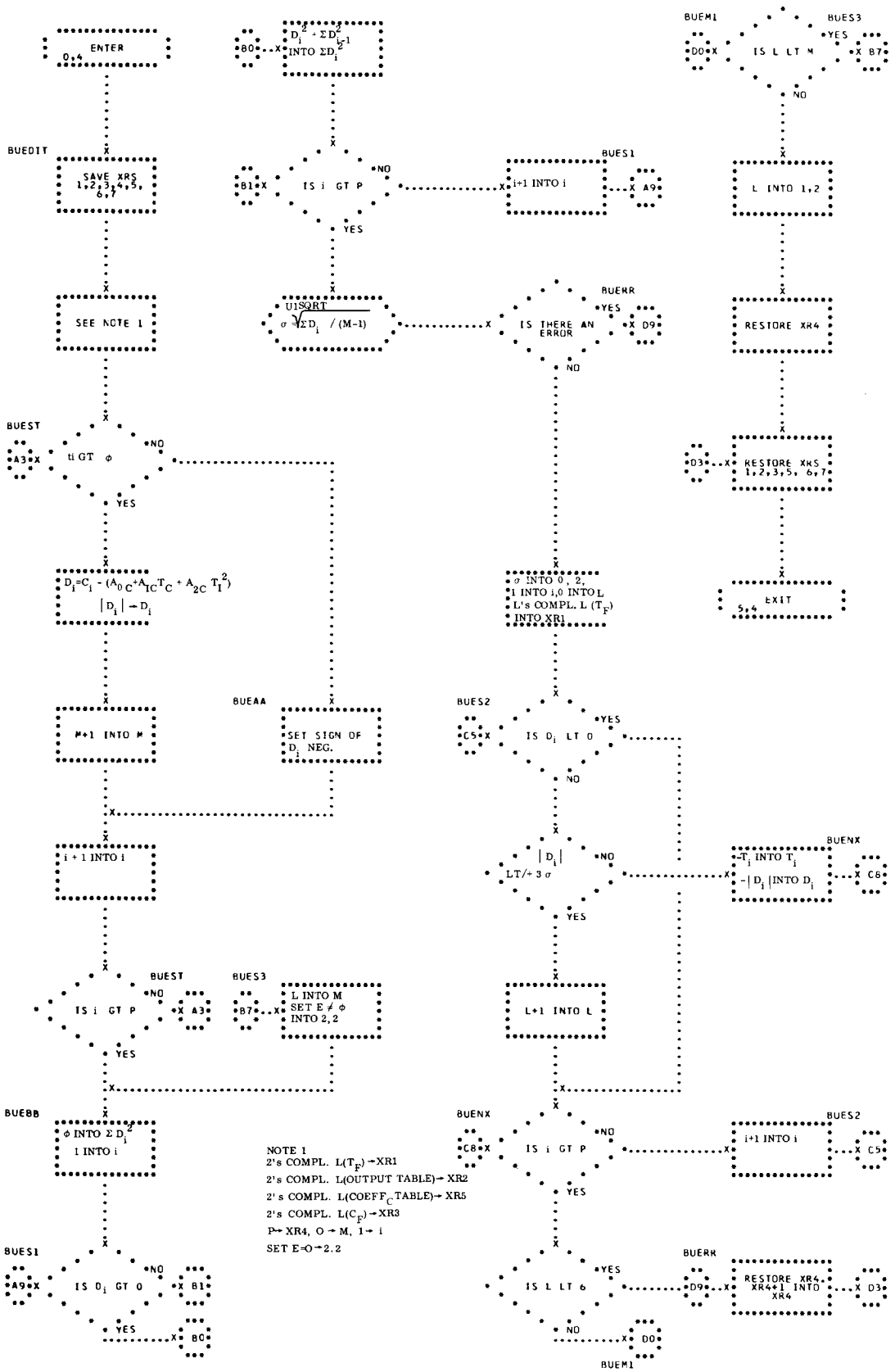
CASE 1-21 RADAR MESSAGES, NO MESSAGES ELIMINATED - 3.715 MS

CASE 2-21 RADAR MESSAGES, 1 MESSAGE ELIMINATED ON EACH SUCCESSIVE EDIT PASS, RESULTING IN 16 PASSES AND AN ERROR RETURN - 43.197 MS

D) ERROR CONDITIONS - AN ERROR RETURN IS THE RESULT OF

1) AN ERROR RETURN FROM LIBRARY SUBROUTINE U1SQRT, OR

2) LESS THAN 6 RADAR MESSAGES ARE REMAINING AFTER ONE OR MORE EDIT PASSES.


ENTER 0,4
BUEDIT
SAVE XRS 1,2,3,4,5, 6,7
SEE NOTE 1
BUEST
ti GT φ
NO
YES
$D_i=C_i-(A_0C+A_{1C}T_C+A_{2C}T_I^2)$
$|D_i|\rightarrow D_i$
M+1 INTO M
BUEAA
SET SIGN OF D_i NEG.
i + 1 INTO i
IS i GT P
BUEST
BUES3
L INTO M
SET E ≠ φ
INTO 2,2
BUE8B
φ INTO Σ D_i^2
1 INTO i
BUES1
IS D_i GT 0
B0
$D_i^2+\Sigma D_{i-1}^2$ INTO ΣD_i^2
IS i GT P
BUES1
i+1 INTO i
$\sigma=\sqrt{\Sigma D_i/(M-1)}$
BUERR
IS THERE AN ERROR
φ INTO 0, 2,
1 INTO i,0 INTO L
L's COMPL. L (T_F)
INTO XR1
BUES2
IS D_i LT 0
$|D_i|$ LT/+ 3σ
NO
YES
BUENX
$-T_i$ INTO T_i
$-|D_i|$ INTO D_i
L+1 INTO L
IS i GT P
BUENX
BUES2
i+1 INTO i
IS L LT 6
BUERR
RESTORE XR4+1 INTO XR4
BUEM1
IS L LT M
BUES3
L INTO 1,2
RESTORE XR4
RESTORE XRS 1,2,3,5, 6,7
EXIT 5,4
NOTE 1
2's COMPL. L(T_F)→XR1
2's COMPL. L(OUTPUT TABLE)→XR2
2's COMPL. L($COEFF_C$ TABLE)→XR5
2's COMPL. L(C_F)→XR3
P→XR4, 0→M, 1→i
SET E=0→2.2

76. BUGSCP - SPACECRAFT POSITION AND HEIGHT DETERMINATION PROGRAM FOR BUMAIN

BUGSCP COMPUTES THE GEODETIC LATITUDE, LONGITUDE, AND GEODETIC HEIGHT OF THE SPACECRAFT AT TIME T. BUGSCP IS A SUBROUTINE FOR BUMAIN. (SEE FLOWCHART).

INPUT

A) TABLE -

A 4-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L(INPUT TABLE)	T	TIME, FLOATING POINT SECONDS
+1	X	X-COORDINATE, FLOATING POINT GEMINI LENGTH UNITS
+2	Y	Y-COORDINATE, FLOATING POINT GEMINI LENGTH UNITS
+3	Z	Z-COORDINATE, FLOATING POINT GEMINI LENGTH UNITS

B) SUBROUTINES -

SUBROUTINES U1SQRT, U1ATAB, AND U1ATNA ARE USED.

C) CONSTANTS -

K001.0	FLOATING POINT ONE
KKK2PI	FLOATING POINT VALUE OF 2π, 6.2831853
K.INB2	INVERSE OF THE SQUARE OF THE SEMI-MINOR AXIS OF THE GEMINI SPHEROID IN FLOATING POINT GEMINI LENGTH UNITS, 1.C067383
K.B2GU	SQUARE OF THE SEMI-MINOR AXIS OF THE GEMINI SPHEROID IN FLOATING

POINT GEMINI LENGTH UNITS, .99330680

K.ECC2 — SQUARE OF THE ECCENTRICITY OF THE GEMINI SPHEROID IN FLOATING POINT GEMINI LENGTH UNITS, $.66931988 \times 10^{-2}$

K.OMES — ANGULAR ROTATIONAL VELOCITY OF THE EARTH IN FLOATING POINT RADIANS/SECONDS, $.72921151 \times 10^{-4}$

KLAMDO* — INERTIAL LONGITUDE OF GREENWICH AT MIDNIGHT PRECEDING LAUNCH IN FLOATING POINT RADIANS.

*KLAMDO IS NOT A TRUE CONSTANT. IT IS A PARAMETER WHOSE VALUE CHANGES WITH EACH DAY OF THE YEAR. FOR SIMPLICITY, IT IS CURRENTLY ASSIGNED THE VALUE ZERO IN THE GEMINI SYSTEM.

OUTPUT

A 3-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L(OUTPUT TABLE) — ϕ GEODETIC LATITUDE AT TIME T, FLOATING POINT RADIANS ($-\pi/2 \leq \phi \leq \pi/2$)

+1 — λ LONGITUDE AT TIME T, FLOATING POINT RADIANS ($\phi \leq \lambda \leq 2\pi$)

+2 — H GEODETIC HEIGHT AT TIME T, FLOATING POINT FEET

METHOD

WHEN CONTROL IS TRANSFERRED TO BUGSCP, THE CONTENTS OF THE INDEX REGISTERS USED ARE SAVED.

NEXT, ϕ λ AND H ARE COMPUTED. REFER TO GRTS PROGRAMMER WORKBOOK, VOLUME III, PROGRAM A3GSCP FOR THE DERIVATION OF EQUATIONS USED.

FINALLY, THE CONTENTS OF THE INDEX REGISTERS USED ARE RESTORED AND CONTROL IS TRANSFERRED BACK TO BUMAIN.

USAGE

A) CALLING SEQUENCE -

```
TSX       BUGSCP,4
PZE       L(INPUT TABLE),,L(OUTPUT TABLE)
ERROR     RETURN
NORMAL    RETURN
```

B) STORAGE REQUIRED -

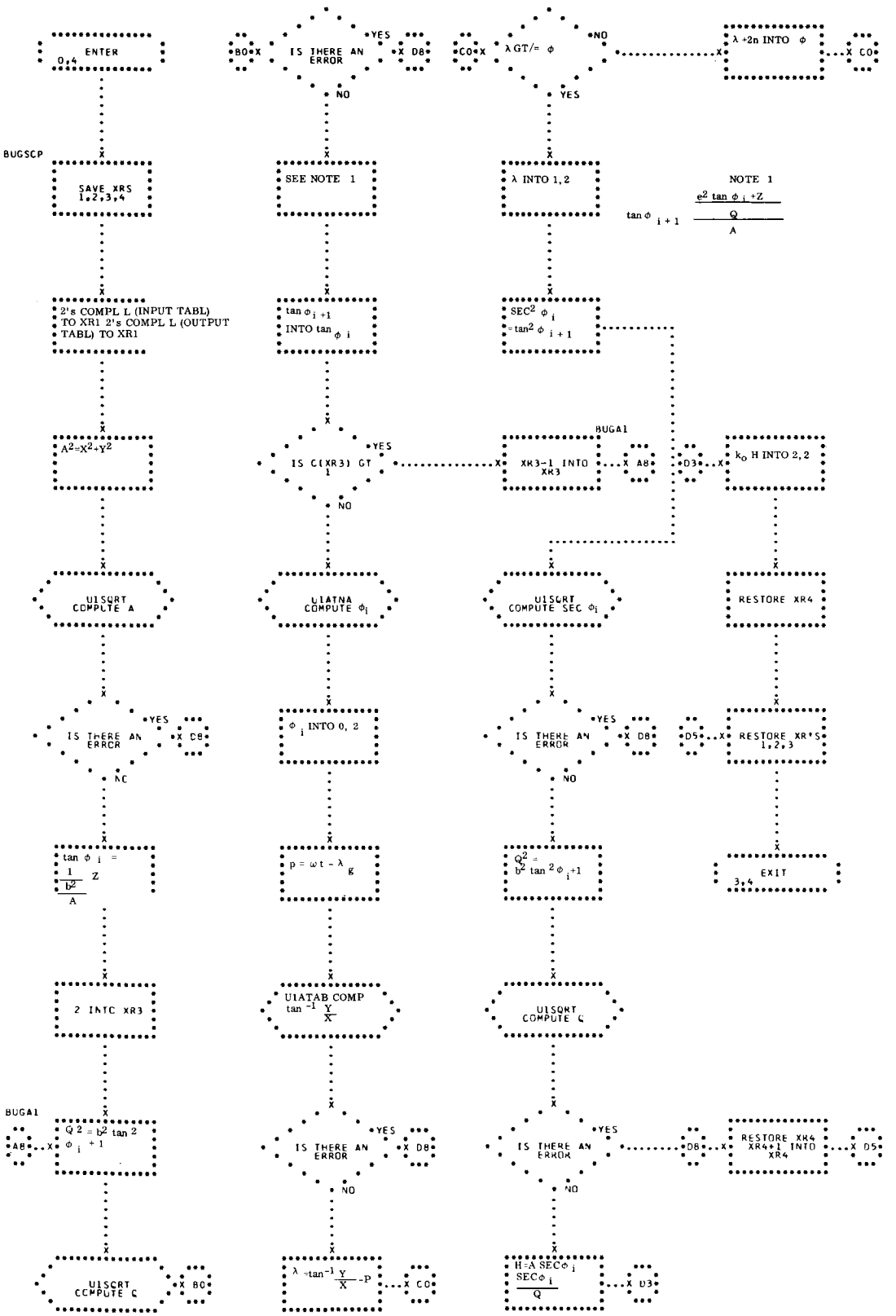
84 LOCATIONS (EXCLUDING SUBROUTINES U1SQRT, U1ATAB, AND U1ATNA)

C) TIMING (7094) -

1.757 MS (INCLUDING TIME REQUIRED BY U1SQRT, U1ATAB, AND U1ATNA)

D) ERROR CONDITION -

AN ERROR RETURN IS THE RESULT OF AN ERROR FROM ONE OF THE LIBRARY SUBROUTINES.


ENTER 0,4
BUGSCP
SAVE XRS 1,2,3,4
2's COMPL L (INPUT TABL) TO XR1 2's COMPL L (OUTPUT TABL) TO XR1
A2=X2+Y2
U1SQRT COMPUTE A
IS THERE AN ERROR
YES
NO
D8
tan φ i = 1/(b2/A) Z
2 INTO XR3
BUGA1
Q2 = b2 tan2 φ i + 1
U1SQRT COMPUTE Q
B0
IS THERE AN ERROR
YES
NO
D8
SEE NOTE 1
tan φ i +1 INTO tan φ i
IS C(XR3) GT 1
U1ATNA COMPUTE φi
φ i INTO 0, 2
p = ωt - λ g
U1ATAB COMP tan-1 Y/X
λ =tan-1 Y/X -P
C0
λ GT/= φ
λ +2n INTO φ
λ INTO 1,2
SEC2 φ i =tan2 φ i + 1
BUGA1
XR3-1 INTO XR3
A8
D3
ko H INTO 2,2
U1SQRT COMPUTE SEC φi
RESTORE XR4
D5
RESTORE XR'S 1,2,3
EXIT 3,4
Q2 = b2 tan2 φ i +1
U1SQRT COMPUTE Q
RESTORE XR4 XR4+1 INTO XR4
H=A SECφ i SECφ i/Q
NOTE 1
tan φ i + 1 = (e2 tan φ i +Z)/(Q/A)

77. BUFPPR - FOOTPRINT DESCRIPTION

GIVEN A t, $\bar{r}$, $\bar{v}$ ARRAY, BUFPPR OBTAINS THE IMPACT POSITICNS FOR ZERO LIFT, MAXIMUM LIFT, AND 60-DEGREE BANK ANGLE CONDITIONS, AND COMPUTES THE DOWNRANGE AND CROSS-RANGE DIMENSIONS OF THE ASSOCIATED FOOTPRINTS. IT IS A SUBROUTINE OF BUMAIN, THE MAIN COMPUTATIONAL PROCESSOR FOR THE DIFFERENTIAL CORRECTION BACKUP PROGRAM. (SEE FLOWCHART.)

INPUT

A) CONSTANT -
KMINTR - CONVERTS MINUTES TO RADIANS

B) COMMUNICATION CELL -
MCRWGT - REENTRY WEIGHT OF THE SPACECRAFT (FLOATING POINT POUNDS.)

C) TABLE -
L(INPUT) - THE LOCATION OF A 7-ITEM TABLE CONTAINING, RESPECTIVELY, TIME IN FLOATING POINT SECONDS, COMPONENTS OF THE POSITION VECTOR IN GEMINI LENGTH UNITS, AND COMPONENTS OF THE VELOCITY VECTOR IN GEMINI VELOCITY UNITS.

D) INTERMEDIATE TABLE -
TMBUFO - GEODETIC LATITUDE OF IMPACT IN FLOATING POINT RADIANS.
TMBUFO+1 - LONGITUDE OF IMPACT IN FLOATING POINT RADIANS.
THE ABOVE VALUES ARE SET WITHIN MYFPPR BEFORE IT RETURNS TO BUFPPR.

E) SUBROUTINES -

1) LIBRARY -

U1SICC

U3XPRC

U3UNTV

U3DCTP

U3ASCC

2) GEMINI SYSTEM -

MYFPPR, A MONITOR PROGRAM WHICH CALLS THE INTEGRATOR, NOCPNI.

OUTPUT

A) TABLE -

L(OUTPUT - THE LOCATION OF AN 8-ITEM TABLE WHICH CONTAINS THE FOLLOWING -

1) ZERO LIFT IMPACT LATITUDE (FLOATING POINT RADIANS)

2) ZERO LIFT IMPACT LONGITUDE (FLOATING POINT RADIANS)

3) MAXIMUM LIFT IMPACT LATITUDE (FLOATING POINT RADIANS)

4) MAXIMUM LIFT IMPACT LONGITUDE (FLOATING POINT RADIANS)

5) 60-DEGREE BANK ANGLE IMPACT LATITUDE (FLOATING POINT RADIANS)

6) 60-DEGREE BANK ANGLE IMPACT LONGITUDE (FLOATING POINT RADIANS)

7) DOWNRANGE DIMENSION OF FOOTPRINT (FLOATING POINT NAUTICAL MILES)

8) CROSSRANGE DIMENSION OF FOOTPRINT (FLOATING POINT NAUTICAL MILES)

B) INTERMEDIATE TABLE -

TBUFIN - A 17-LOCATION TABLE REQUIRED AS INPUT TO NOCPNI. THE FOLLOWING ARE SET WITHIN BUFPPR -

TBUFIN
X COMPONENT OF POSITION VECTOR IN FLOATING POINT GEMINI LENGTH UNITS.

+1 - Y COMPONENT OF POSITION VECTOR

+2 - Z COMPONENT OF POSITION VECTOR

+3 - $\dot{X}$ COMPONENT OF VELOCITY VECTOR IN FLOATING POINT GEMINI VELOCITY UNITS

+4 - $\dot{Y}$ COMPONENT OF VELOCITY VECTOR

+5 - $\dot{Z}$ COMPONENT OF VELOCITY VECTOR

+6 - TIME OF THE ABOVE VECTORS IN FLOATING POINT SECONDS

+12 - REENTRY WEIGHT OF SPACECRAFT IN FLOATING POINT POUNDS

+14 - GMT TO INITIATE ROLL IN FLOATING POINT SECONDS
= TBUFIN +6 FOR ZERO LIFT CASE
< 0 FOR MAXIMUM LIFT AND 60 DEGREE BANK ANGLE CASES, INDICATING NO ROLL

+15 - BANK ANGLE IN FLOATING POINT DEGREES
NOT USED FOR ZERO LIFT CASE
= 0 FOR MAXIMUM LIFT CASE
= 60. FOR 60-DEGREE BANK ANGLE CASE.

+16 = 0 (INDICATES DO NOT FILL IN TNREEP TABLE.)

METHOD

A t, $\overline{r}$, $\overline{v}$ ARRAY IS SPECIFIED BY THE INPUT. USING THIS FOR INITIAL CONDITIONS, THE TRAJECTORY IS INTEGRATED TO IMPACT THREE TIMES USING THE GEMINI INTEGRATOR, NOCPNI.

THIS IS DONE ONCE EACH FOR ZERO AND MAXIMUM LIFT CONDITIONS, AND ONCE USING A 60-DEGREE BANK ANGLE. AFTER EACH CASE, THE GEODETIC LATITUDE AND LONGITUDE OF IMPACT ARE STORED IN THE OUTPUT.

ONLY A SMALL ERROR OCCURS IF THE GEODETIC LATITUDE AND LONGITUDE ARE TREATED AS SPHERICAL COORDINATES ON A UNIT SPHERE. USING THIS APPROXIMATION, THE UNIT POSITION VECTORS OF AN IMPACT POINT IN AN EARTH-RELATIVE CARTESIAN FRAME MAY BE OBTAINED AS FOLLOWS -

$$X = \cos\lambda \cos\phi$$

$$Y = \sin\lambda \cos\phi$$

$$Z = \sin\phi$$

THE UNIT POSITION VECTORS ARE DESCRIBED USING THE ABOVE EQUATIONS FOR $\bar{r}_{ZL}$, THE ZERO LIFT CASE, $\bar{r}_{ML}$, THE MAXIMUM LIFT CASE, AND $\bar{r}_{60}$, THE 60-DEGREE BANK ANGLE CASE.

A VECTOR IS FORMED WHICH IS PERPENDICULAR TO THE PLANE OF $\bar{r}_{ML}$ AND $\bar{r}_{ZL}$ BY A CROSS PRODUCT -

$$\bar{r}_C = \bar{r}_{ML} \times \bar{r}_{ZL}$$

THE VECTOR r_C IS THEN UNITIZED -

$$\bar{r}_D = \bar{r}_C / |\bar{r}_C|$$

THE ANGLES SUBTENDED BY THE DOWNRANGE AND CROSSRANGE DIMENSIONS OF THE FOOTPRINT ARE FOUND, RESPECTIVELY, BY

$$\phi_L = \cos^{-1} \bar{r}_{ML} \cdot \bar{r}_{ZL}, \quad \text{(DOWNRANGE)} \quad \text{AND}$$

$$\phi_W = \text{SIN}^{-1} \bar{r}_{60} \cdot \bar{r}_D, \quad \text{(CROSSRANGE)} \quad \text{SEE SECTION 79}$$

SINCE ONE NAUTICAL MILE IS, BY DEFINITION, EQUAL TO ONE MINUTE OF ARC AT THE EQUATOR, THE DOWNRANGE AND CROSS-

RANGE DIMENSIONS OF THE FOOTPRINT ARE,

L (NAUT. MI.) $\cong \Theta_L$ (MIN), AND

W (NAUT. MI.) $\cong \Theta_W$ (MIN.).

L AND W ARE STORED IN THE SEVENTH AND EIGHTH OUTPUT LOCATICNS.

USAGE

A) CALLING SEQUENCE -

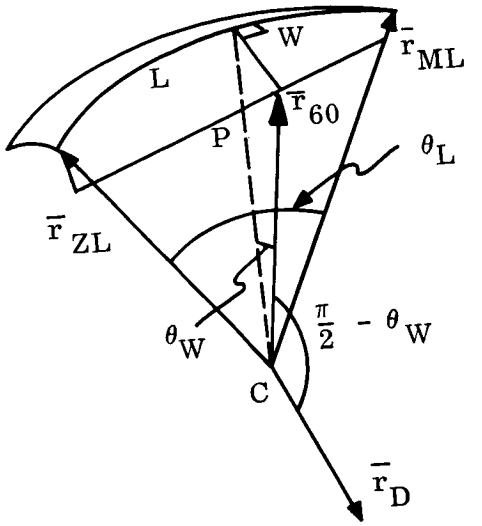
```
TSX     BUFPPR,4
PZE     L(INPUT),,L(OUTPUT)
ERROR   RETURN
NORMAL  RETURN
```

B) STORAGE REQUIREMENTS - 104 LOCATIONS, EXCLUDING THOSE REQUIRED BY EXTERNAL CONSTANTS, TABLES, AND SUBROUTINES.

C) TIME REQUIRED (MAXIMUM 7094) - 2.89 MILLI-SECONDS, EXCLUDING TIME REQUIRED BY THE INTE-GRATOR, NOCPNI.

D) ERROR CONDITIONS - AS OF 6 APRIL 1965, NO PATH EXISTS WITHIN BUFPPR TO EXIT TO THE ERROR RETURN PORTION OF THE CALLING SEQUENCE.
THE CALLING SEQUENCE HAS BEEN LEFT UNCHANGED SO THAT REPRCGRAMMING WILL NOT BE IMPOSED ON OTHER PORTIONS OF THE SYSTEM.
SHOULD AN ERROR RETURN BE ENCOUNTERED FROM ONE OF THE SUBROUTINES, ZEROS ARE STORED IN THE PORTION OF THE OUTPUT WHICH HAS NOT BEEN SET.
BUFPPR THEN EXITS TO THE CALLING PROGRAM VIA THE NORMAL RETURN.

DERIVATION OF FOOTPRINT EQUATIONS



L AND W ARE SEGMENTS OF GREAT CIRCLES ON THE SURFACE OF A UNIT SPHERE WITH CENTER AT C. $\bar{r}_D$ IS THE UNIT VECTOR OF $\bar{r}_{ML} \times r_{ZL}$.

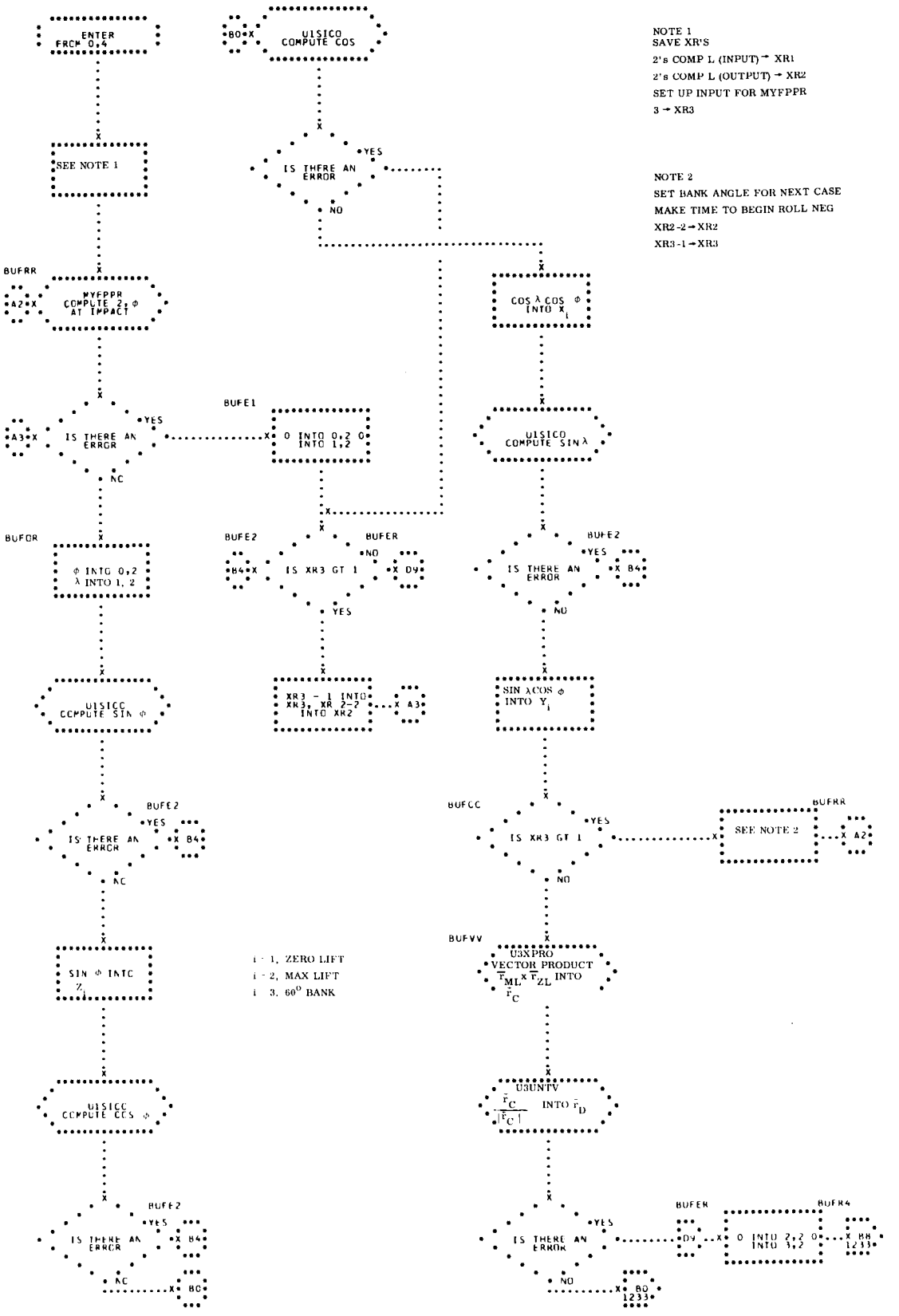
THE PLANE FORMED BY $\bar{r}_{ZL}$ AND $\bar{r}_{ML}$ IS PERPENDICULAR TO THE PLANE FORMED BY $\bar{r}_D$ AND $\bar{r}_{60}$, AND THE TWO PLANES INTERSECT ALONG LINE SEGMENT P.

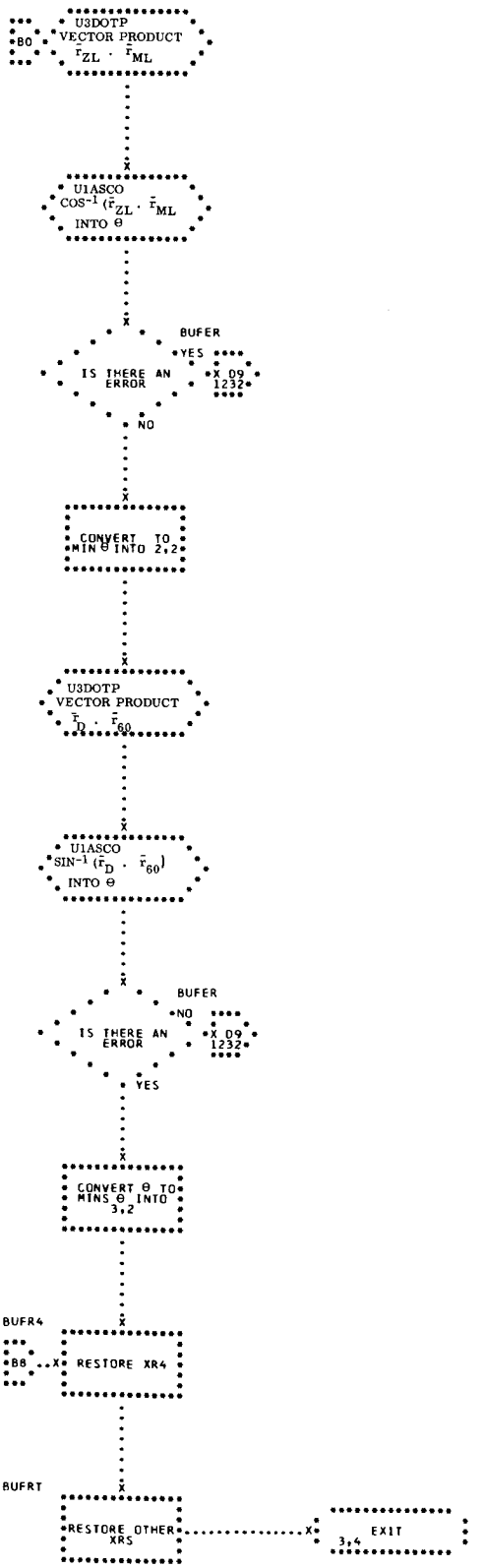
$$\bar{r}_{ZL} \cdot \bar{r}_{ML} = \left|\bar{r}_{ZL}\right| \left|\bar{r}_{ML}\right| \cos \theta_L = \cos \theta_L$$

$$\theta_L = \cos^{-1} (\bar{r}_{ZL} \cdot \bar{r}_{ML})$$

$$\bar{r}_{60} \cdot \bar{r}_D = \left|\bar{r}_{60}\right| \left|\bar{r}_D\right| \cos (\pi/2 - \theta_W) = \sin \theta_W$$

$$\theta_W = \sin^{-1} (\bar{r}_{60} \cdot \bar{r}_D)$$


ENTER FROM 0,4
SEE NOTE 1
BUFRR
A2
MYFPPR COMPUTE Z, Φ AT IMPACT
A3
IS THERE AN ERROR
YES
NO
BUFOR
Φ INTO 0,2
λ INTO 1, 2
ULSICC COMPUTE SIN Φ
BUFE2
B4
IS THERE AN ERROR
YES
NO
SIN Φ INTO Z
ULSICC COMPUTE COS Φ
BUFE2
B4
IS THERE AN ERROR
YES
NO
B0
B0
ULSICO COMPUTE COS
IS THERE AN ERROR
YES
NO
BUFE1
0 INTO 0,2 0 INTO 1,2
BUFER
B4
IS XR3 GT 1
D9
YES
NO
XR3 - 1 INTO XR3, XR 2-2 INTO XR2
A3
COS λ COS Φ INTO X
ULSICO COMPUTE SIN λ
BUFE2
B4
IS THERE AN ERROR
YES
NO
SIN λ COS Φ INTO Y
BUFCC
IS XR3 GT 1
YES
NO
SEE NOTE 2
BUFRR
A2
BUFVV
U3XPRO VECTOR PRODUCT $\bar{r}_{ML} \times \bar{r}_{ZL}$ INTO $\bar{r}_C$
U3UNTV $\frac{\bar{r}_C}{|\bar{r}_C|}$ INTO $\bar{r}_D$
BUFER
D9
IS THERE AN ERROR
YES
NO
B0
1233
BUFER4
0 INTO 2,2 0 INTO 3,2
BR
1233
NOTE 1
SAVE XR'S
2's COMP L (INPUT) → XR1
2's COMP L (OUTPUT) → XR2
SET UP INPUT FOR MYFPPR
3 → XR3
NOTE 2
SET BANK ANGLE FOR NEXT CASE
MAKE TIME TO BEGIN ROLL NEG
XR2 -2 → XR2
XR3 -1 → XR3
i = 1, ZERO LIFT
i = 2, MAX LIFT
i = 3, 60° BANK


U3DOTP
VECTOR PRODUCT
$\bar{r}_{ZL} \cdot \bar{r}_{ML}$
B0
U1ASCO
$\cos^{-1}(\bar{r}_{ZL} \cdot \bar{r}_{ML})$
INTO θ
BUFER
YES
IS THERE AN ERROR
X 09 1232
NO
CONVERT TO MIN θ INTO 2,2
U3DOTP
VECTOR PRODUCT
$\bar{r}_D \cdot \bar{r}_{60}$
U1ASCO
$\sin^{-1}(\bar{r}_D \cdot \bar{r}_{60})$
INTO θ
BUFER
NO
IS THERE AN ERROR
X 09 1232
YES
CONVERT θ TO MINS θ INTO 3,2
BUFR4
B8
RESTORE XR4
BUFRT
RESTORE OTHER XRS
EXIT 3,4

78. BUKEPE - PROGRAM TO COMPUTE SIX CLASSICAL ELEMENTS.

GIVEN A POSITION AND VELOCITY VECTOR IN GEMINI UNITS, THIS PROGRAM COMPUTES SIX CLASSICAL ELEMENTS. BUKEPE IS A SUBROUTINE OF THE ORBIT PHASE BACK UP TO DIFFERENTIAL CORRECTION PROGRAM BUOBCP. (SEE FLOWCHART.)

INPUT

A) TABLE -

A 6-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L (INPUT TABLE)	X	FLOATING POINT GEMINI LENGTH UNITS
+1	Y	FLOATING POINT GEMINI LENGTH UNITS
+2	Z	FLOATING POINT GEMINI LENGTH UNITS
+3	$\dot{X}$	FLOATING POINT GEMINI VELOCITY UNITS
+4	$\dot{Y}$	FLOATING POINT GEMINI VELOCITY UNITS
+5	$\dot{Z}$	FLOATING POINT GEMINI VELOCITY UNITS

B) CONSTANTS -

K001.0	FLOATING POINT ONE
KCGUNM	K_{ϕ}, CONVERTS GEMINI LENGTH UNITS INTO NAUTICAL MILES, $.34439336X10^{4}$
KKK2PI	FLOATING POINT 2π, 6.2831853

C) SUBROUTINES - U1SQRT, U3VMAG, U3DOTP, U3XPRO, U1ASCO, AND U1ATAB ARE USED.

OUTPUT

A) TABLE -

A 7-LOCATION TABLE, ARRANGED IN THE FOLLOWING FORMAT -

L(OUTPUT TABLE)	a	SEMI-MAJOR AXIS, FLOATING POINT NAUTICAL MILES
+1	e	ECCENTRICITY, DIMENSIONLESS
+2	ν	TRUE ANOMALY, FLOATING POINT RADIANS
+3	i	INCLINATION ANGLE FLOATING POINT RADIANS
+4	Ω	LONGITUDE OF THE ASCENDING NODE, FLOATING POINT RADIANS
+5	ω	ARGUMENT OF PERIGEE, FLOATING POINT RADIANS
+6	$\neq 0$	IF a AND e HAVE BEEN COMPUTED SUCCESSFULLY
	$= 0$	IF a AND e HAVE NOT BEEN COMPUTED SUCCESSFULLY

METHOD

(FUNDAMENTALS OF CELESTIAL MECHANICS, J.M.A. DANBY, THE MACMILLAN COMPANY, 1962, CHAPTER 6, PP 115.)

A) THE CONTENTS OF THE INDEX REGISTERS USED ARE SAVED AND THE 7TH LOCATION OF THE OUTPUT TABLE IS SET TO ZERO.

B) THE SEMI-MAJOR AXIS a IS COMPUTED USING

$$e \cos E = rv^2 - 1$$

$$a = \frac{r}{1 - e \cos E}$$

a IS CONVERTED TO NAUTICAL MILES, USING THE CONSTANT K_ϕ. IF r EQUALS ZERO OR UPON AN ERROR RETURN FROM A LIBRARY SUBROUTINE, THE PROGRAM TRANSFERS CONTROL TO THE LOGIC OF PARAGRAPH H) BELOW.

C) THE ECCENTRICITY IS COMPUTED USING

$$e \sin E = \frac{\bar{r} \cdot \bar{v}}{\sqrt{a}}$$

$$e = \sqrt{(e \sin E)^2 + (e \cos E)^2}$$

UPON AN ERROR RETURN FROM A LIBRARY SUBROUTINE THE PROGRAM TRANSFERS CONTROL TO THE LOGIC OF PARAGRAPH H) BELOW. OTHERWISE, HAVING COMPUTED a AND e SUCCESSFULLY, THE 7TH LOCATION OF THE OUTPUT TABLE IS SET NONZERO.

D) THE TRUE ANOMALY ν IS COMPUTED USING

$$\nu = \arctan\left(\frac{e \sin E \sqrt{1 - e^2}}{e \cos E - e^2}\right)$$

IF ν IS NEGATIVE, 2π IS ADDED TO ν
UPON AN ERROR RETURN FROM A LIBRARY SUBROUTINE THE PROGRAM TRANSFERS CONTROL TO THE LOGIC OF PARAGRAPH H) BELOW.

E) THE INCLINATION ANGLE i IS COMPUTED USING

$$\bar{h} = \bar{r} \times \bar{v}$$

$$i = \text{arc sin} \left(\frac{\sqrt{h_x^2 + h_y^2}}{h}\right)$$

UPON AN ERROR RETURN FROM A LIBRARY SUBROUTINE THE PROGRAM TRANSFERS CONTROL TO THE LOGIC OF PARAGRAPH H) BELOW.

F) THE LONGITUDE OF THE ASCENDING NODE Ω IS COMPUTED USING

$$\Omega = \arctan\left(\frac{h_x}{-h_y}\right)$$

IF Ω IS NEGATIVE, 2π IS ADDED TO Ω
UPON AN ERROR RETURN FROM A LIBRARY SUBROUTINE THE PROGRAM TRANSFERS CONTROL TO THE LOGIC OF PARAGRAPH H) BELOW.

G) THE ARGUMENT OF PERIGEE ω IS COMPUTED USING

$$u = \arctan\left(\frac{Zh}{Yh_x - Xh_y}\right)$$

$$\omega = u - \nu$$

IF u IS NEGATIVE, 2π IS ADDED TO u. IF ω IS NEGATIVE, 2π IS ADDED TO ω IF THERE IS NO ERROR RETURN FROM A LIBRARY SUBROUTINE THE PROGRAM TRANSFERS CONTROL TO THE LOGIC OF PARAGRAPH I) BELOW.

H) ZEROS ARE STORED IN THE LOCATION OF THE QUANTITY CURRENTLY COMPUTED THROUGH THE 6TH LOCATION OF THE OUTPUT TABLE.

I) THE CONTENTS OF THE INDEX REGISTERS USED ARE RESTORED AND BUKEPE EXITS VIA ITS NORMAL RETURN.

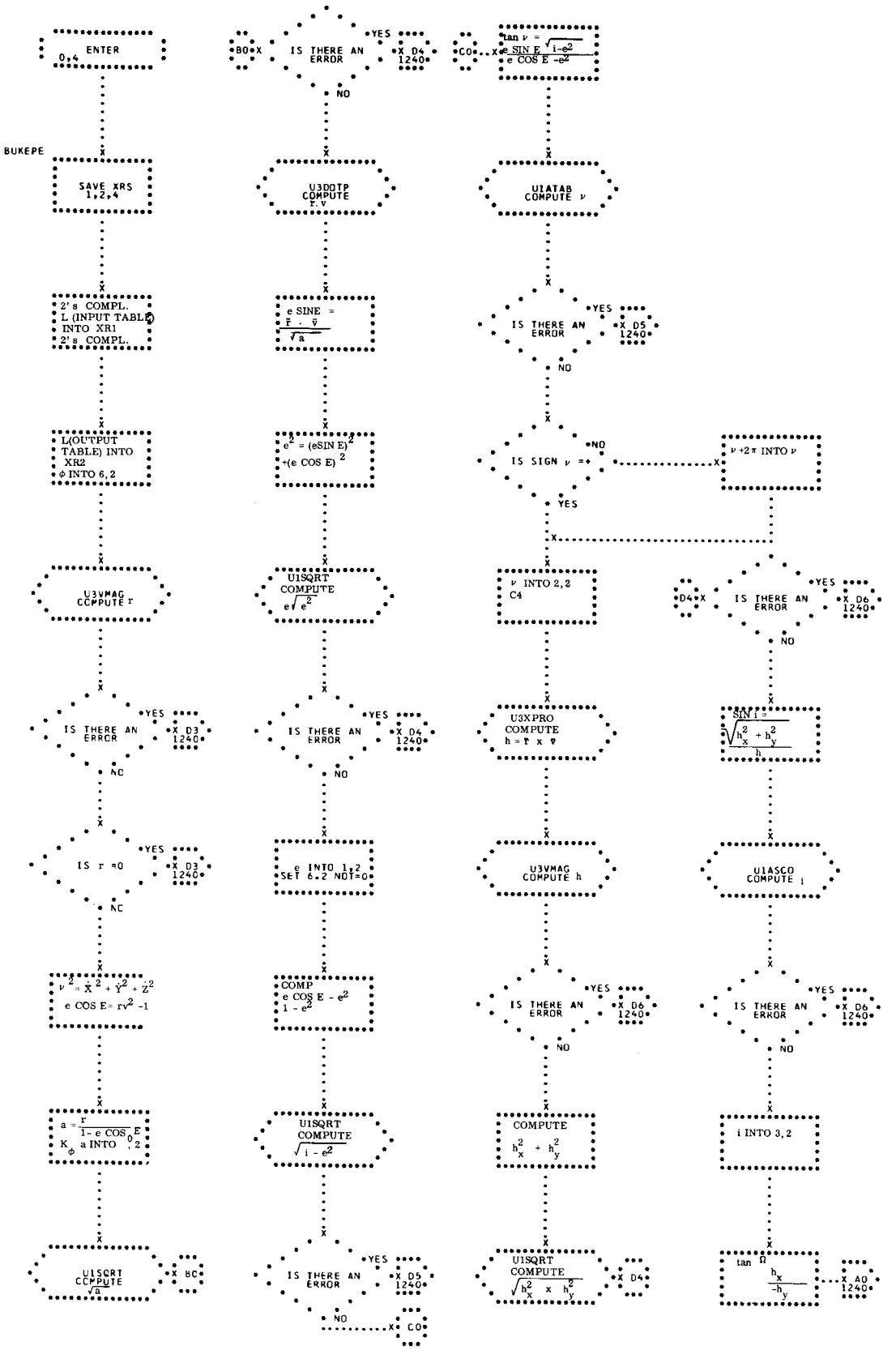
USAGE

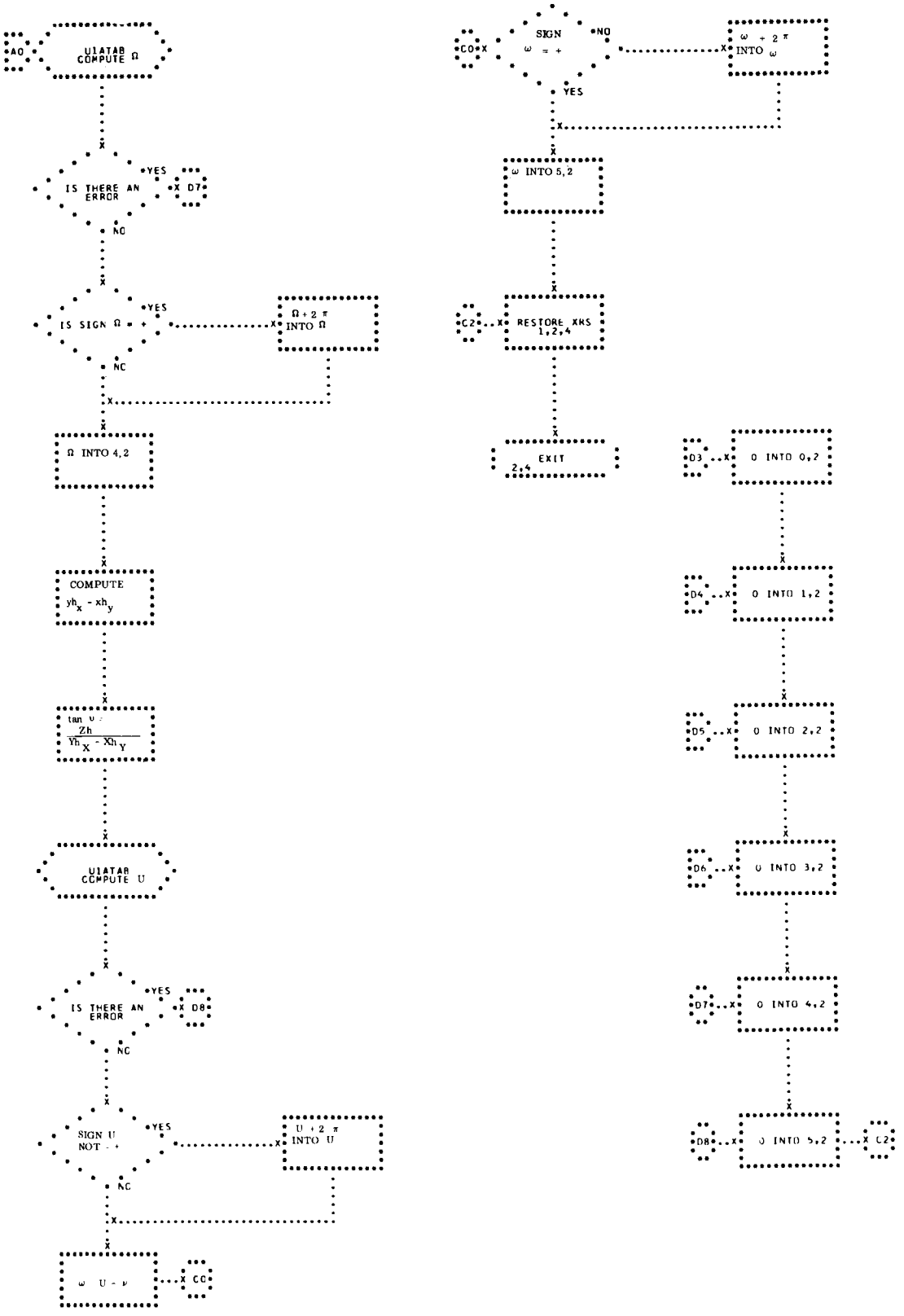
A) CALLING SEQUENCE -

```
TSX     BUKEPE,4
PZE     L(INPUT TABLE,,L(OUTPUT TABLE)
NORMAL  RETURN
```

B) STORAGE REQUIRED - 130 LOCATIONS

C) TIMING (7094) (INCLUDING ALL LIBRARY SUBROUTINES) - 3.068 MS.


ENTER 0,4
BUKEPE
SAVE XRS 1,2,4
2's COMPL. L (INPUT TABLE) INTO XR1 2's COMPL.
L(OUTPUT TABLE) INTO XR2 φ INTO 6,2
U3VMAG COMPUTE r
IS THERE AN ERROR
YES
X D3 1240
NO
IS r =0
YES
X D3 1240
NO
ν² = Ẋ² + Ẏ² + Ż²
e COS E = rv² -1
a = r / 1 - e COS E
K_φ a INTO 0,2
U1SQRT COMPUTE √a
X BC
BO
IS THERE AN ERROR
YES
X D4 1240
NO
U3DOTP COMPUTE r.v
e SINE = r · v / √a
e² = (eSIN E)² +(e COS E)²
U1SQRT COMPUTE e√e²
IS THERE AN ERROR
YES
X D4 1240
NO
e INTO 1,2 SET 6,2 NOT=0
COMP e COS E - e² / 1 - e²
U1SQRT COMPUTE √(1 - e²)
IS THERE AN ERROR
YES
X D5 1240
NO
X CO
CO
tan ν = e SIN E √(1-e²) / e COS E - e²
U1ATAB COMPUTE ν
IS THERE AN ERROR
YES
X D5 1240
NO
IS SIGN ν =+
NO
ν +2π INTO ν
YES
ν INTO 2,2 C4
U3XPRO COMPUTE h = r x v
U3VMAG COMPUTE h
IS THERE AN ERROR
YES
X D6 1240
NO
COMPUTE h_x² + h_y²
U1SQRT COMPUTE √(h_x² x h_y²)
X D4
D4
IS THERE AN ERROR
YES
X D6 1240
NO
SIN i = √(h_x² + h_y²) / h
U1ASCO COMPUTE i
IS THERE AN ERROR
YES
X D6 1240
NO
i INTO 3,2
tan Ω h_x / -h_y
X A0 1240


A0
U1ATAB COMPUTE Ω
IS THERE AN ERROR
YES
D7
NO
IS SIGN Ω = +
YES
Ω + 2 π INTO Ω
NO
Ω INTO 4,2
COMPUTE yh_x - xh_y
tan υ = Zh / (Yh_X - Xh_Y)
U1ATAB COMPUTE U
IS THERE AN ERROR
YES
D8
NO
SIGN U NOT = +
YES
U + 2 π INTO U
NO
ω U - ν
C0
SIGN ω = +
NO
ω + 2 π INTO ω
YES
ω INTO 5,2
C2
RESTORE XRS 1,2,4
EXIT 2,4
D3
0 INTO 0,2
D4
0 INTO 1,2
D5
0 INTO 2,2
D6
0 INTO 3,2
D7
0 INTO 4,2
D8
0 INTO 5,2
C2

79. MFBUDC - MONITOR SUFFIX TO BUDCCP AND BUOBCP.

SEE FLOWCHART.

INPUT

MCBUER - ZERO IF NO ERROR OCCURED IN BUDCCP.
- NON-ZERO IF ERROR OCCURED IN BUDCCP.
- OUTPUT TABLE FROM BUDCCP OR BUOBDP

OUTPUT

TMBUPR - TABLE OF QUANTITIES TO BE PRINTED ON-LINE.

METHOD

A) IF THIS PASS IS NOT IN REENTRY, CONTROL IS PASSED TO PARAGRAPH (C). IF AN ERROR OCCURED ON THIS PASS, CONTROL IS PASSED TO PARAGRAPH (D). THE REENTRY OUTPUT IS MOVED TO THE PRINT BUFFER, THE LATITUDE AND LONGITUDE QUANTITIES ARE CONVERTED FROM RADIANS TO DEGREES, AND THE TIME IS CONVERTED TO FIXED POINT. THE REENTRY QUANTITIES ARE THEN PRINTED ON-LINE.

B) THE DIFFERENTIAL CORRECTION PROCESSORS ARE UNSUPPRESSED AND CONTROL IS PASSED TO MOPRIO.

C) THE LATITUDE AND LONGITUDE QUANTITIES ARE CONVERTED FROM RADIANS TO DEGREES AND THE ORBIT OUTPUT QUANTITIES ARE MOVED TO THE PRINT BUFFER. THE ORBIT QUANTITIES ARE THEN PRINTED ON-LINE AND CONTROL IS PASSED TO PARAGRAPH (B).

D) THE ERROR FLAG IS RESET AND THE UPPER CORE BLOCK NUMBER IS SAVED FOR PRINTING. AN ERROR MESSAGE IS PRINTED ON-LINE AND CONTROL IS PASSED TO

PARAGRAPH (B).

USAGE

MFBUDC IS ENTERED FROM BUDCCP AND BUOBCP AND EXITS TO MOPRIO.

A) STORAGE REQUIRED - 86 LOCATIONS

B) COMMUNICATION CELLS - MCPHSE, MCBUER

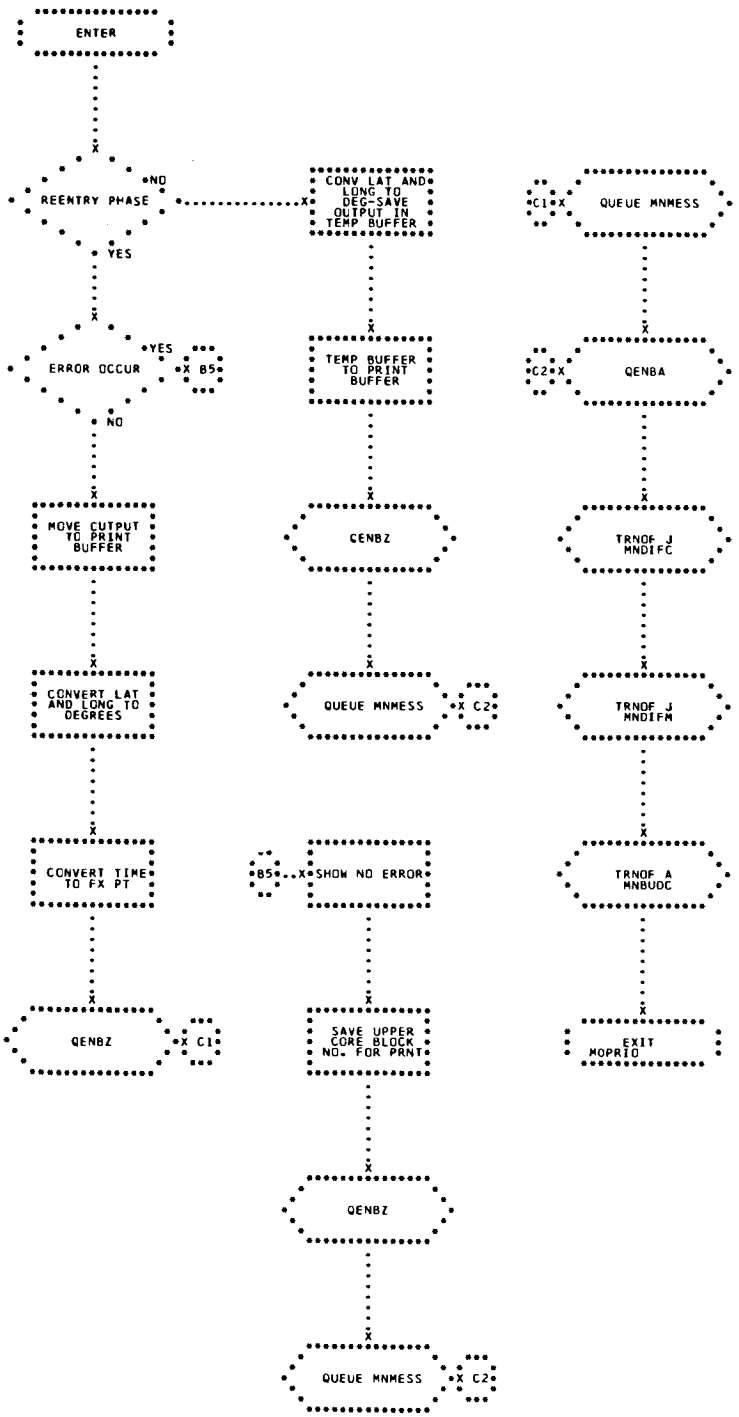
C) TABLES - TMBUPR

D) CONSTANTS - K00.PI, KKK2PI,KRTDEG,KCH233,KMNMSK

E) MACROS - QENBZ, QENBA, QUEUE, TRNOF

F) PARAMETERS -
A,J, MNMESS, MNDIFC, MNDIFM, MNDIFK, MNBUDC

G) TIME REQUIRED -
MINIMUM - .024 MILLISECONDS PLUS TIME REQUIRED FOR EXECUTION OF QUEUE
MAXIMUM - 1.62 MILLISECONDS PLUS TIME REQUIRED FOR EXECUTION OF QUEUE


ENTER
REENTRY PHASE
NO
YES
CONV LAT AND LONG TO DEG-SAVE OUTPUT IN TEMP BUFFER
C1
QUEUE MNMESS
ERROR OCCUR
YES
B5
NO
TEMP BUFFER TO PRINT BUFFER
C2
QENBA
MOVE OUTPUT TO PRINT BUFFER
QENBZ
TRNOF J MNDIFC
CONVERT LAT AND LONG TO DEGREES
QUEUE MNMESS
C2
TRNOF J MNDIFM
CONVERT TIME TO FX PT
B5
SHOW NO ERROR
TRNOF A MNBUDC
QENBZ
C1
SAVE UPPER CORE BLOCK NO. FOR PRNT
EXIT MOPRIO
QENBZ
QUEUE MNMESS
C2

80. MYFPPR - MONITOR FOOTPRINT PREDICTOR.

SEE FLOWCHART

INPUT

MYFPPR RECEIVES ITS INPUT VIA QTSX. THIS IS THEN USED AS INPUT TO NOCPNI.

OUTPUT

OUTPUT FROM MYFPPR CONSISTS OF THE OUTPUT FROM NOCPNI.

METHOD.

MYFPPR SUPPRESSES ALL USERS OF NOCPNI. MYFPPR USES ITS INPUT AS INPUT TO NOCPNI AND USES THE OUTPUT FROM NOCPNI AS ITS OUTPUT. THE PROPER RETURN IS SET UP, ALL USERS OF NOCPNI ARE UNSUPPRESSED AND RETURN IS MADE TO MOPRIO.

USAGE

MYFPPR IS ENTERED FROM AND EXITS TO MOPRIO.

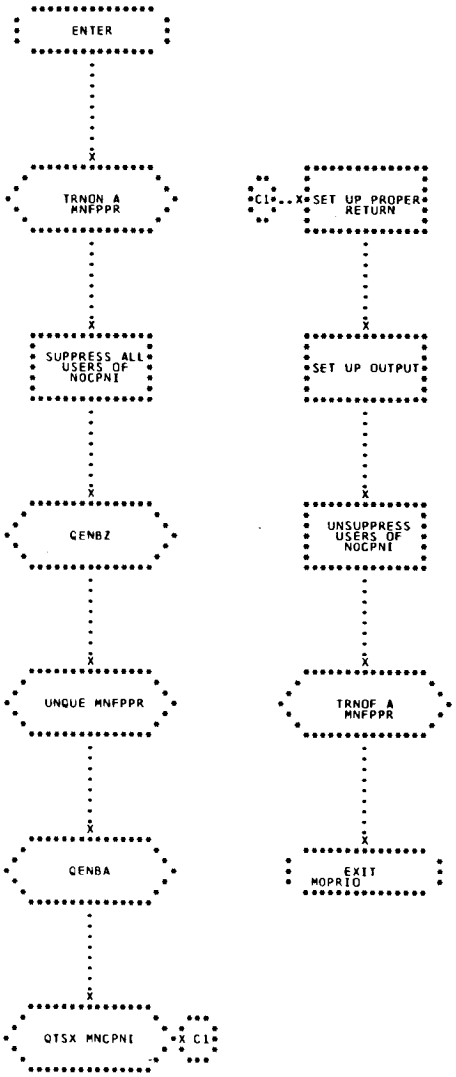
A) STORAGE REQUIRED - 54 LOCATIONS

B) TABLES - TMSAVE, TMBBNI, TMIMPP

C) MACROS - TRNON, TRNOF, QENBZ, QENBA, UNQUE, QTSX

D) PARAMETERS - A,H,S, MNFPPR, MNCOTO, MNOUR1, MNREN2, MNAPE3, MNTTF4, MNNAV5, MNGMT6, MNMUV7, MNLIF8, MNDIC9, MNCPNI

E) TIME REQUIRED - .06 MILLISECONDS PLUS TIME REQUIRED BY UNQUE, QTSX, AND NOCPNI.


ENTER
TRNON A MNFPPR
SUPPRESS ALL USERS OF NOCPNI
QENBZ
UNQUE MNFPPR
QENBA
QTSX MNCPNI
C1
C1
SET UP PROPER RETURN
SET UP OUTPUT
UNSUPPRESS USERS OF NOCPNI
TRNOF A MNFPPR
EXIT MOPRIO

81. RHBRVG - PROGRAM TO COMPUTE THE RELATIVE VELOCITY VECTOR MAGNITUDE AND RELATIVE FLIGHT PATH ANGLE.

GIVEN A (t, $\bar{r}$, $\bar{v}$ DEFINING THE TRAJECTORY OF A SPACECRAFT, RHBRVG COMPUTES THE MAGNITUDE OF THE ASSOCIATED RELATIVE VELOCITY VECTOR AND THE RELATIVE FLIGHT PATH ANGLE. IT IS A SUBROUTINE OF THE REENTRY PROGRAM TO CONTROL THE GENERATION OF BACKUP GUIDANCE PARAMETERS, MYREN2. (SEE FLOWCHART).

INPUT

A) THE INPUT (t, $\bar{r}$, $\bar{v}$) IS IN FLOATING POINT IN A SEVEN-LOCATION TABLE WHICH IS FORMATTED AS FOLLOWS -

L(INPUT)		
	t	TIME IN GMT SECONDS
+1	x	$\bar{r}$, POSITION VECTOR IN GEMINI LENGTH UNITS.
+2	y	
+3	z	
+4	$\dot{x}$	$\bar{v}$, VELOCITY VECTOR IN GEMINI VELOCITY UNITS.
+5	$\dot{y}$	
+6	$\dot{z}$	

B) CONSTANTS -

K.OMEG	DEC	ROTATIONAL VELOCITY OF THE EARTH IN RADIANS PER GEMINI TIME UNIT
KRTDEG	DEC	FACTOR TO CONVERT RADIANS TO DEGREES
KFTSCG	DEC	FLOATING POINT GEMINI VELOCITY UNIT IN FEET/SECOND

C) SUBROUTINES (THESE ARE ACTUALLY B-CORE LINKAGES TO A-CORE LIBRARY ROUTINES UISQRT AND UIASCO, RESPECTIVELY.) -

BBSQRT	B-CORE FLOATING POINT SQUARE ROOT LIBRARY ROUTINE
BARSIN	B-CORE FLOATING POINT ARCSIN LIBRARY ROUTINE

OUTPUT

THE OUTPUT IS IN FLOATING POINT IN A 2-LOCATION TABLE AS FOLLOWS -

L(OUTPUT)	$\lvert \bar{V}_e \rvert$	RELATIVE VELOCITY VECTOR MAGNITUDE IN FEET/SECOND
+1	γ_R	RELATIVE DIGIT PATH ANGLE IN DEGREES

METHOD

WHEN CONTROL IS RECEIVED BY RHBRVG, THE ECC IS SET TO A CORE AND THE ICC IS SET TO B CORE. RHBRVG SETS THE ECC TO B CORE, SAVES THE CONTENTS OF INDEX REGISTERS 1,2,3, AND 4 AND PICKS UP THE LOCATIONS OF THE INPUT AND OUTPUT TABLES. IT THEN RESTORES THE ECC TO A CORE AND SAVES THE LOCATIONS OF THE INPUT AND OUTPUT TABLES. THE RELATIVE VELOCITY VECTOR IS THEN COMPUTED USING -

$$\bar{V}_R = \begin{pmatrix} \dot{x} + y\,\omega \\ \dot{y} + x\,\omega \\ \dot{z} \end{pmatrix}$$

WHERE ω IS THE ROTATIONAL VELOCITY OF THE EARTH IN RADIANS PER GEMINI TIME UNIT (SEE C4DRFT FOR A DERIVATION OF THIS FORMULA).

THE MAGNITUDE OF THE RELATIVE VELOCITY VECTOR IS THEN COMPUTED. THE RELATIVE FLIGHT PATH ANGLE IS COMPUTED USING -

$$\gamma_R = \text{SIN}^{-1}\left(\frac{\bar{r} \cdot \bar{v}_R}{|\bar{r}\|\bar{v}_R|}\right)$$

(SEE C4DRFT FOR A DERIVATION OF THIS FORMULA.)

THE RELATIVE FLIGHT PATH ANGLE IS CONVERTED TO FLOATING POINT DEGREES AND THE RELATIVE VELOCITY VECTOR MAGNITUDE IS CONVERTED TO FEET PER SECOND. EACH OF THE ABOVE QUANTITIES IS STORED IN THE OUTPUT TABLE, INDEX REGISTERS ARE RESTORED AND AN EXIT IS MADE TO THE CALLING PROGRAM. AN ERROR RETURN IS THE RESULT OF AN ERROR CONDITION UPON RETURN FROM BBSQRT OR BARSIN.

'JSAGE

A) THE CALLING SEQUENCE IS AS FOLLOWS -

```
TSX     RHBRVG,4
PZE     L(INPUT),,L(OUTPUT)
ERROR   RETURN
NORMAL  RETURN
```

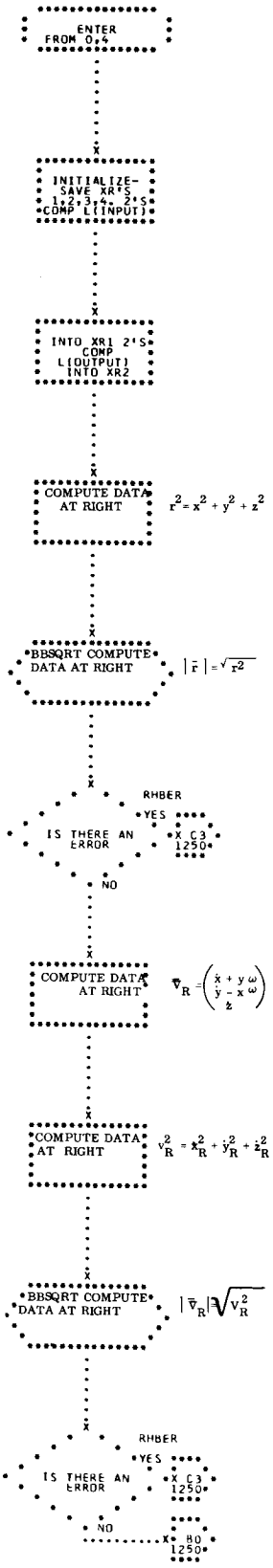
B) STORAGE REQUIRED -

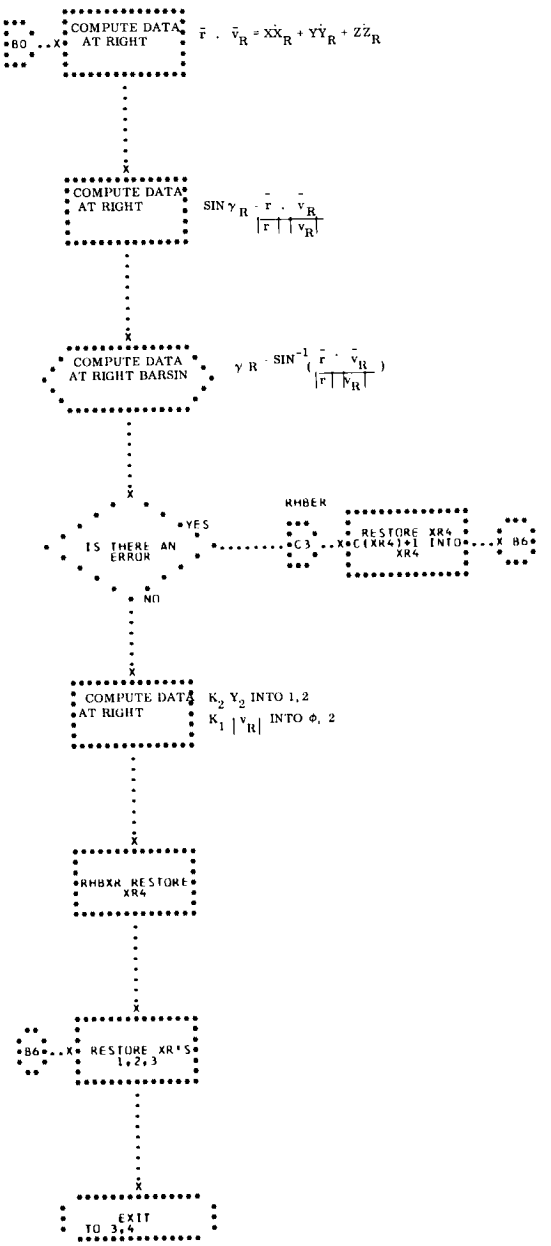
A-CORE 6 LOCATIONS

B-CORE 72 LOCATIONS

C) TIMING -

1.135 MILLISECONDS INCLUDES SQUARE ROOT AND ARCSINE LIBRARY ROUTINES.


ENTER FROM 0,4
INITIALIZE- SAVE XR'S 1,2,3,4, 2'S COMP L[INPUT]
INTO XR1 2'S COMP L[OUTPUT] INTO XR2
COMPUTE DATA AT RIGHT
$r^2 = x^2 + y^2 + z^2$
BBSQRT COMPUTE DATA AT RIGHT
$|\bar{r}| = \sqrt{r^2}$
IS THERE AN ERROR
RHBER
YES
X C3 1250
NO
COMPUTE DATA AT RIGHT
$\bar{v}_R = \begin{pmatrix} \dot{x} + y\omega \\ \dot{y} - x\omega \\ \dot{z} \end{pmatrix}$
COMPUTE DATA AT RIGHT
$v_R^2 = \dot{x}_R^2 + \dot{y}_R^2 + \dot{z}_R^2$
BBSQRT COMPUTE DATA AT RIGHT
$|\bar{v}_R| = \sqrt{v_R^2}$
IS THERE AN ERROR
RHBER
YES
X C3 1250
NO
X B0 1250


B0
COMPUTE DATA AT RIGHT
$\bar{r} \cdot \bar{v}_R = X\dot{X}_R + Y\dot{Y}_R + Z\dot{Z}_R$
COMPUTE DATA AT RIGHT
$\sin\gamma_R = \frac{\bar{r} \cdot \bar{v}_R}{|r| |v_R|}$
COMPUTE DATA AT RIGHT BARSIN
$\gamma_R = \sin^{-1}\left(\frac{\bar{r} \cdot \bar{v}_R}{|r| |v_R|}\right)$
IS THERE AN ERROR
YES
RHBER
C3
RESTORE XR4 C(XR4)+1 INTO XR4
B6
NO
COMPUTE DATA AT RIGHT
K_2 Y_2 INTO 1, 2
$K_1 |v_R|$ INTO 0, 2
RHBXR RESTORE XR4
B6
RESTORE XR'S 1,2,3
EXIT TO 3,4

82. GINTAB - SUBROUTINE TO DETERMINE INTEGRATION TABLE

WITH THE REQUIREMENT TO PROVIDE AN ORBIT EPHEMERIS REFLECTING THE S/C'S MANEUVER CAPABILITY, A METHOD IS NEEDED TO DETERMINE INTEGRATION TABLE AS A FUNCTION OF TIME. THE EPHEMERIS IS A SERIES OF TABLES, FREE FLIGHT, THRUST, FREE FLIGHT, ETC. THROUGH INTERROGATION OF THE MANEUVER TABLE IN CONJUNCTION WITH THE INPUT GMT, IT IS POSSIBLE TO DETERMINE THE PROPER INTEGRATION TABLE FOR THE ASSCCIATED INPUT TIME. (SEE FLOWCHART).

INPUT

A) THE AC CONTAINS THE INPUT TIME -
   1. S(AC) = +, TIME IS IN THIS FORMAT -
      A(FIXED GMT MINUTES),,D(FIXED GMT SECONDS)
   2. S(AC) = -, TIME IS IN FIXED GMT SECONDS.

B) TMANUV - S/C MANEUVER TABLE

C) TPCIAC - TABLE OF INTEGRATION TABLE ADDRESSES

D) MCMANC - NUMBER OF MANEUVERS IN TMANUV

E) MCESAB - CELL CONTAINING FIRST EVEN SECOND AFTER RETROFIRE BURNOUT.

F) MCPHSE - CELL INDICATING MISSION PHASE

G) MCC5IP - NZ INDICATES 050RMC IS IN PROCESS

CUTPUT

THE ADDRESS OF THE AC CONTAINS THE ADDRESS OF THE DESIRED INTEGRATION TABLE -

1. S(AC) = +, FREE FLIGHT TABLE INDICATED
2. S(AC) = -, THRUST TABLE INDICATED

METHOD

UPON PROGRAM ENTRY, THE INPUT TIME IS CONVERTED TO FIXED GMT SECONDS IF NOT ALREADY SO. MCESAB IS EXAMINED TO SEE IF A REENTRY TABLE IS DESIRED. IF SO, TNINT2 (REENTRY) OR TCRKO2 (HIGH ABORT) IS INDICATED. OTHERWISE, THE TIMES IN THE MANEUVER TABLE ARE COMPARED TO THE INPUT TIME UNTIL A SUITABLE TABLE IS FOUND. AT THIS TIME THE BASE TABLE ADDRESS FROM TPDIAC IS PLACED IN THE AC WITH THE REQUIRED SIGN.

USAGE

A) CALLING SEQUENCE -

```
ALPHA   TSX     GINTAB,4
ALPHA+1 TRA     NORMAL RETURN
```

B) ERROR CONDITIONS - NO ERROR RETURN IS REQUIRED.

C) CELLS -

COMMON

+1

+2

D) COMMUNICATION CELLS -

MCESAB

MCPHSE

MCOSIP

E) TABLES -

TMANUV

TPDIAC

TCRKO2

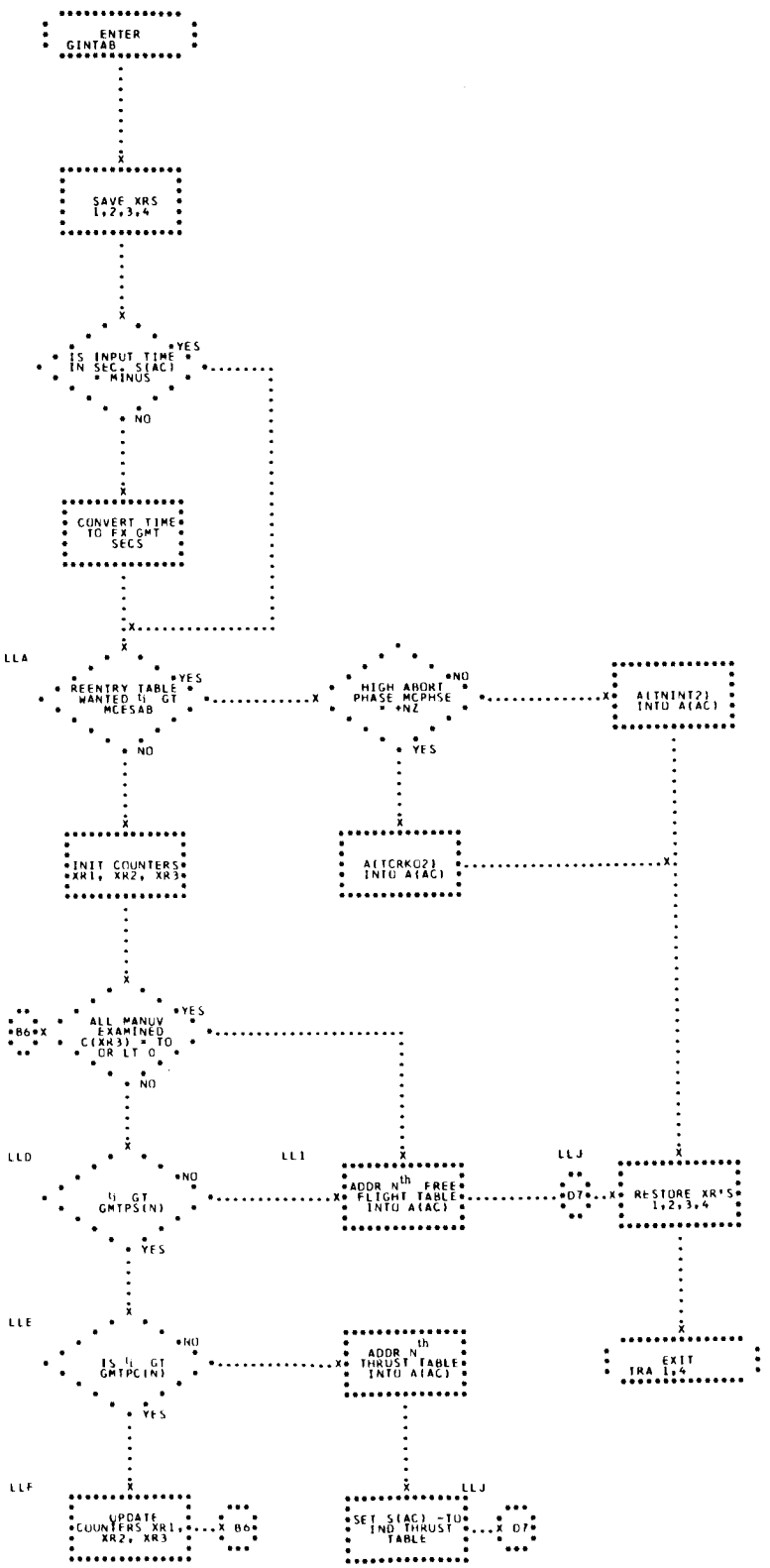
TNINT2

F) CONSTANTS -

KD0060

G) STORAGE REQUIRED -

45 LOCATIONS


ENTER GINTAB
SAVE XRS 1,2,3,4
IS INPUT TIME IN SEC. S(AC) = MINUS
YES
NO
CONVERT TIME TO FX GMT SECS
LLA
REENTRY TABLE WANTED t_i GT MCESAB
YES
NO
HIGH ABORT PHASE MCPHSE = +NZ
NO
YES
ALTNINT2) INTO A(AC)
INIT COUNTERS XR1, XR2, XR3
A(TCRKO2) INTO A(AC)
B6
ALL MANUV EXAMINED C(XR3) = TO OR LT 0
YES
NO
LLD
LLI
LLJ
t_i GT GMTPS(N)
NO
YES
ADDR N^{th} FREE FLIGHT TABLE INTO A(AC)
D7
RESTORE XR'S 1,2,3,4
LLE
IS t_i GT GMTPC(N)
NO
YES
ADDR N^{th} THRUST TABLE INTO A(AC)
EXIT TRA 1,4
LLF
UPDATE COUNTERS XR1, XR2, XR3
B6
SET S(AC) -TO IND THRUST TABLE
LLJ
D7

83. GEMWGT - ORBIT WEIGHT SUBROUTINE

TO PROVIDE THE S/C ORBIT WEIGHT FOR ANY GIVEN GMT, SUBROUTINE GEMWGT IS USED. THIS WEIGHT IS COMPUTED AS A FUNCTION OF TIME AND THE PLANNED S/C MANEUVERS. AS SUCH, INPUT TIMES ASSUMING NOMINAL MANEUVER COMPLETION WILL YIELD THE S/C WEIGHT REFLECTING WEIGHT LOSS DUE TO THRUSTER FIRING. (SEE FLOWCHART.)

INPUT

A) THE AC SHOULD CONTAIN THE GMT, IN FIXED POINT SECONDS, FOR WHICH THE S/C WEIGHT IS DESIRED.
B) TMANUV - S/C MANEUVER TABLE.
C) TOSCCH - S/C CHARACTERISTICS TABLE.
D) MCMANC - CONTAINS NUMBER OF MANEUVERS IN TMANUV
E) MCOWGT - CONTAINS PRESENT S/C ORBIT WEIGHT

OUTPUT

THE AC CONTAINS THE REQUESTED S/C ORBIT WEIGHT IN FLOATING POINT POUNDS.

METHOD

AFTER SAVING THE INPUT TIME, A SEARCH OF THE MANEUVER TABLE (TMANUV) IS BEGUN. FOR EVERY MANEUVER COMPLETED, BASED ON THE INPUT GMT, THE WEIGHT LOSS DUE TO THRUST IS COMPUTED. THE WEIGHT LOSSES ARE THEN SUMMED AND SUBTRACTED FROM THE PRESENT S/C ORBIT WEIGHT, YIELDING THE DESIRED WEIGHT.

AN INPUT TIME COINCIDENT WITH AN ORBITAL MANEUVER WILL YIELD THE PRE-MANEUVER S/C WEIGHT.

USAGE

A) CALLING SEQUENCE -

```
ALPHA   TSX     GEMWGT,4
ALPHA+1 TRA     NORMAL RETURN
```

B) ERROR CONDITIONS - NO ERROR RETURN IS REQUIRED.

C) CELLS -

COMMON
+1
+2
+3

D) COMMUNICATION CELLS -

MCMANC
MCOWGT
MCSUBR

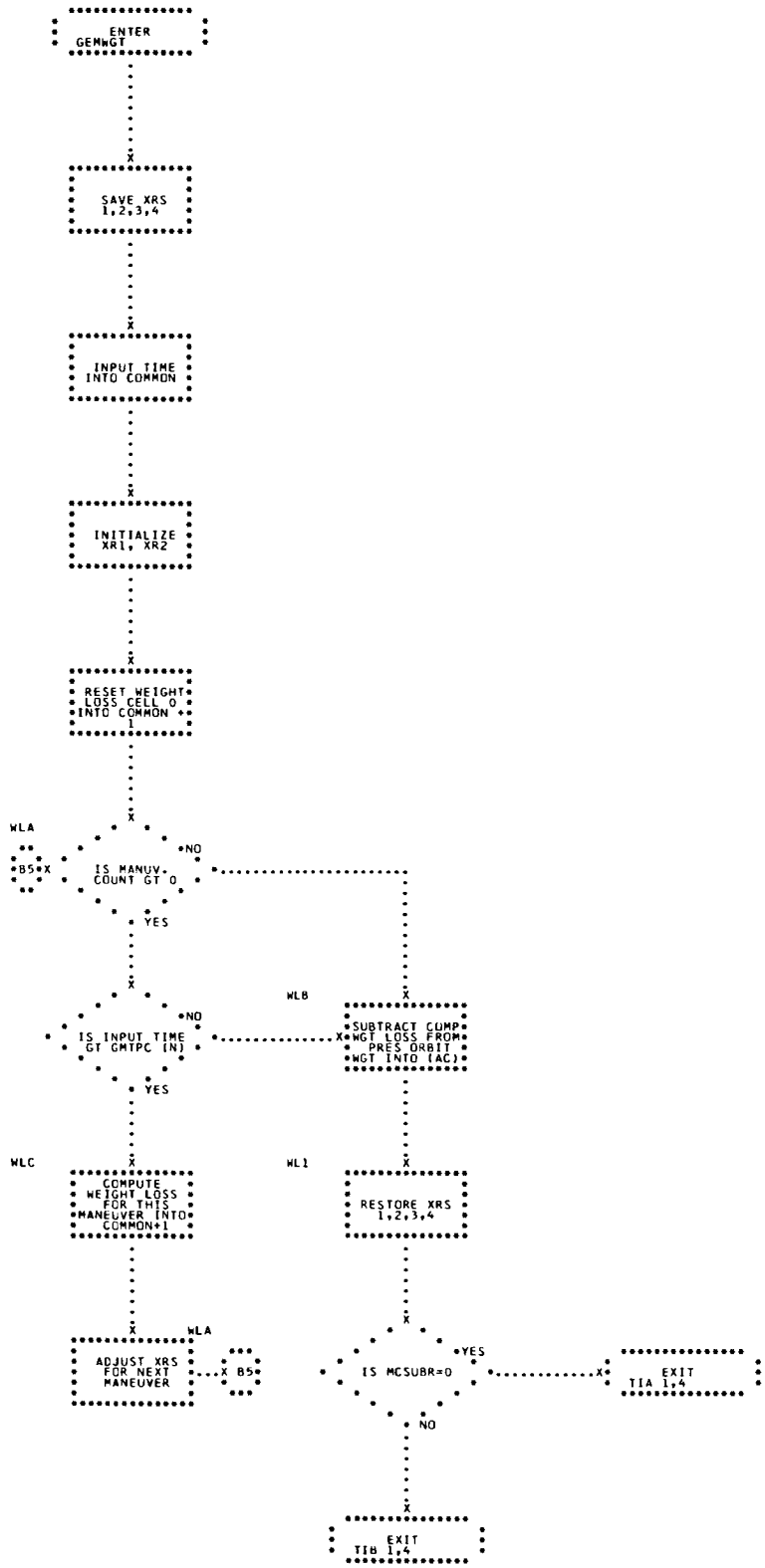
E) TABLES -

TMANUV
TOSCCH

F) CONSTANTS -

K77777
KCH233
K002.0

G) STORAGE REQUIRED - 32 LOCATIONS.


ENTER GEMWGT
SAVE XRS 1,2,3,4
INPUT TIME INTO COMMON
INITIALIZE XR1, XR2
RESET WEIGHT LOSS CELL 0 INTO COMMON +1
WLA
B5
IS MANUV COUNT GT 0
NO
YES
IS INPUT TIME GT GMTPC (N)
NO
YES
WLB
SUBTRACT COMP WGT LOSS FROM PRES ORBIT WGT INTO (AC)
WLC
COMPUTE WEIGHT LOSS FOR THIS MANEUVER INTO COMMON+1
WL1
RESTORE XRS 1,2,3,4
WLA
ADJUST XRS FOR NEXT MANEUVER
B5
IS MCSUBR=0
YES
NO
EXIT TIA 1,4
EXIT TIB 1,4

84. GLINTS - VECTOR LINEAR INTERPOLATION SUBROUTINE

GLINTS USES A LINEAR INTERPOLATION TO OBTAIN A VECTOR AT THE REQUESTED INPUT TIME. (SEE FLOWCHART.)

INPUT

A) THE SECOND WORD OF THE CALLING SEQUENCE CONTAINS THE FOLLOWING INFORMATION -
   1. PREFIX - (+) BOTH $\vec{R}$ AND $\vec{V}$ ARE WANTED.
      - (-) ONLY $\vec{R}$ IS WANTED.
   2. DECREMENT - ADDRESS OF 3- OR 6-LOCATION OUTPUT BLOCK.
   3. ADDRESS - LOCATION OF INTEGRATION TABLE.

B) THE INPUT TIME, IN FIXED GMT SECONDS, OCCUPIES THE THIRD WORD OF THE CALLING SEQUENCE.

OUTPUT

EITHER AN $\vec{R}$ OR AN $\vec{R}$ AND $\vec{V}$ IS THE ROUTINE OUTPUT.

OUTPUT		
	$\vec{R}_x$	
+1	$\vec{R}_y$	INTERPOLATED COMPONENTS OF $\vec{R}$
+2	$\vec{R}_z$	
+3	$\vec{V}_x$	
+4	$\vec{V}_y$	INTERPOLATED COMPONENTS OF $\vec{V}$, IF REQUIRED.
+5	$\vec{V}_z$	

METHOD

A LINEAR INTERPOLATION TECHNIQUE IS USED -

1) $t_\alpha \leq t_i \leq t_\omega$

WHERE

t_α = VECTOR TIME (1)

t_i = REQUEST TIME

t_ω = VECTOR TIME (2)

2) $\Delta i = \dfrac{t_\omega - t_\alpha}{h}$

3) $C_j = C_{\alpha_j} + \Delta i \, (C_{\omega_j} - C_{\alpha_j}) \quad j = 1, 6$

WHERE

C_j = INTERPOLATED VECTOR COMPONENTS

4) h IS PRESETNTLY EQUAL TO 8 SECONDS.

5) THE INTEGRATION TABLE IS ASSUMED TO CONTAIN A SERIES OF (R AND V)'S.

USAGE

A) CALLING SEQUENCE -

```
ALPHA       TSX GLINTS,4
ALPHA + 1   PZE A(OUTPUT),,D(INTEG. TAB.)
ALPHA + 2   PZE (INPUT TIME)
ALPHA + 3   TRA NORMAL RETURN
```

B) ERROR CONDITIONS - SINCE IT IS ASSUMED THAT THE INPUT TIME LIES WITHIN THE LIMITS OF THE SPECIFIED INTEGRATION TABLE, NO ERROR RETURN IS INDICATED.

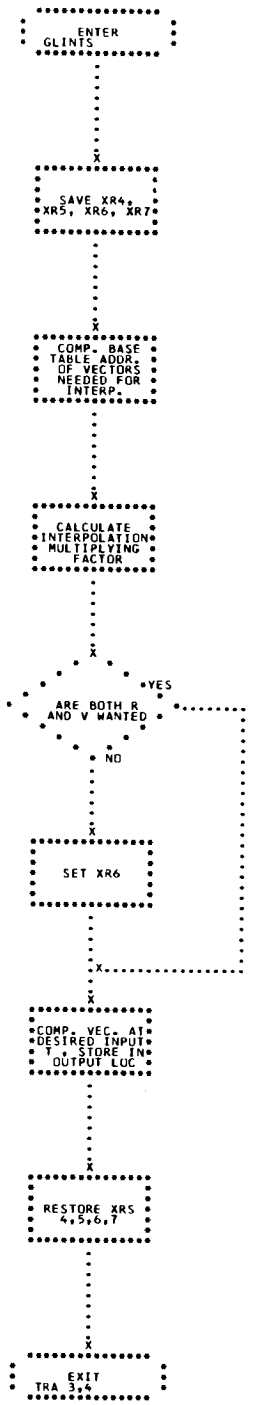
C) CELLS -

COMMON
+1
+2
+3

D) CONSTANTS -

K00008
K00004
KCH233
K0.125
K00006

E) STORAGE REQUIRED - 44 LOCATIONS


ENTER GLINTS
SAVE XR4, XR5, XR6, XR7
COMP. BASE TABLE ADDR. OF VECTORS NEEDED FOR INTERP.
CALCULATE INTERPOLATION MULTIPLYING FACTOR
ARE BOTH R AND V WANTED
YES
NO
SET XR6
COMP. VEC. AT DESIRED INPUT T, STORE IN OUTPUT LOC
RESTORE XRS 4,5,6,7
EXIT
TRA 3,4

85. MYTABU - MANEUVER TABLE CONTROL PROCESSOR

ORBITAL S/C MANEUVERS ARE ENTERED INTO OR DELETED FROM THE MANEUVER TABLE (TMANUV) UNDER THE PROGRAM CONTROL OF MYTABU. AS INDICATED BY THE UNQUEUE WORD, A MANEUVER WILL BE CHRONOLOGICALLY ORDERED IN THE TABLE, COMPLETELY REMOVED FROM IT OR REPLACED BY A NEW SET OF DEFINING PARAMETERS. THESE PARAMETERS ARE PRESENTLY SIX IN NUMBER -

1) TIME OF PROPULSION SETTING (GMTPS)
2) TIME OF PROPULSION COMPLETION (GMTPC)
3) DELTA TIME OF THRUST
4) THRUSTER INDICATOR
5) S/C PITCH
6) S/C YAW

WITH THE COMPLETION OF A PROPOSED MANEUVER, SPACECRAFT WEIGHT IS RECOMPUTED AS A FUNCTION OF MASS LOSS DUE TO FUEL EXPENDITURE. IN ADDITION, MANEUVER TIMES, FOR USE IN DATA EDITING, ARE ADJUSTED TO REFLECT THE LATEST CHANGE.

INPUT

MYTABU QUEUED BY MYMINS MYOUR1 AND MFMN14. THE QUEUE WORD INDICATES THE UPDATE TYPE -

1) Z = AUTOMATIC DELETION OF A MANEUVER
2) -NZ = MANUAL DELETION OF A MANEUVER
3) +NZ = MANUAL INSERTION OF A MANEUVER
4) -Z = AUTOMATIC CATCH-UP DELETION OF A MANEUVER

MANUAL TYPE UPDATES CONTAIN THE MANEUVER GMTPS IN THE QUEUE WORD.

A) TABLES

TMZCOT IOMANI TABLE CONTAINING MANEUVER QUANTITIES

TMANUV CONTAINS S/C MANEUVER DATA

TMREFL CONTAINS MANEUVER REFLECTING VECTORS

B) COMMUNICATION CELLS -

MCMANC CONTAINS CURRENT MANEUVER COUNT

MCCMIP /NZ = MANEUVER IN PROGRESS, Z = FREE FLIGHT

MCHFSC CONTAINS CURRENT GMT HALF-SECOND COUNT

OUTPUT

A) TABLES -

TMANUV UPDATED MANEUVER TABLE

TMORTN TABLE OF MANEUVER TIMES FOR DATA EDITING

TMCODP PARAMETERS OF LAST 2 EXECUTED S/C MANEUVERS

TMCODW WEIGHT ASSOCIATED WITH LAST 2 EXECUTED S/C MANEUVERS

TMMATU INPUT VECTOR TO INITIATE BASIC ORBIT CYCLE

TMREST TABLE OF PARAMETERS FOR RESTART PROGRAM

TPTABU PRINT TABLE USED FOR ERROR MESSAGES

B) COMMUNICATION CELLS -

MCMANC UPDATED MANEUVER COUNT

MCCWGT RECOMPUTED S/C WEIGHT

MCMTPQ MANEUVER TABLE PRINT-OUT INFORMATION

IN ADDITION MYTABU PROVIDES QUEUES FOR MYROUR1 AND MYMESS.

METHOD

THE INITIAL STEP IS TO INTERPOLATE FOR A FREE FLIGHT VECTOR AT PRESENT TIME USING U7INTP. A CHECK IS THEN UNDERTAKEN TO DETERMINE IF THE INTERPOLATED VECTOR HAS BEEN OBTAINED FROM A REVOLUTION OTHER THAN THAT ASSOCIATED WITH THE PRESENT ORBIT DEFINING VECTOR. THE SUBCAPSULE LONGITUDES OF THE VECTORS ARE COMPARED AND, IF NECESSARY, THE REVOLUTION NUMBER IS UPDATED. THIS VECTOR INFORMATION IS REQUIRED AS INPUT TO MYOUR1.

TO ASCERTAIN THE TYPE OF TABLE UPDATE, THE UNQUEUE WORD IS EXAMINED, AND ONE OF THE FOLLOWING FOUR CHOICES IS POSSIBLE -

1) REQUESTED MANEUVER INSERTION - THE DEFINING PARAMETERS ARE PLACED IN TMANUV IN CHRONOLOGICAL ORDER, AND IF NECESSARY, THE OTHER MANEUVERS ARE REPOSITIONED. AN INSERTED GMTPS IDENTICAL TO A TMANUV ENTRY WILL CAUSE THE NEW PARAMETERS TO BE OVERLAID OVER THE OLD ONES.
2) REQUESTED MANEUVER DELETION - THE MANEUVER CORRESPONDING TO THE INSERTED GMTPS WILL BE REMOVED FROM TMANUV. IF NEEDED, REPOSITIONING IS EXECUTED AND THE MANEUVER COUNT IS DECREMENTED.
3) TIME GENERATED DELETION - MANEUVER 1 OF TMANUV IS DELETED AS PRESENT TIME EXCEEDS THE GMTPC. DUE TO FUEL EXPULSION, S/C WEIGHT LOSS IS DEDUCTED FROM PRESENT S/C WEIGHT TO PRODUCE A NEW ORBITAL WEIGHT. AN ON-LINE MESSAGE REPORTS THIS ACTION.

4) CATCH-UP DELETION - THIS OPTION IS EXERCISED IN CONNECTION WITH SYSTEM RESTARTS. MANEUVERS LESS THAN THE RESTART GMT ARE EXECUTED (REFLECTED INTO THE ORBIT) AND THEN DROPPED FROM THE TABLE.

TABLES TMCODP, TMCODW AND TMORTN ARE UPDATED IN EVENT OF A TIME-GENERATED OR CATCH-UP DELETION.

THE CONTENTS OF TMANUV ARE PRINTED ON LINE TO PROVIDE VISUAL VERIFICATION OF EXECUTED UPDATES. AND FINALLY, MYOUR1 IS QUEUED TO INITIATE THE BASIC ORBIT CYCLE, WHEREUPON AN EXIT TO MOPRIO IS TAKEN.

MESSAGE ERROR CODES FOR MYTABU ARE AS FOLLOWS -

A) 10 A3MSCP ERROR

B) 60 U7INIP ERROR

C) 70 TMANUV SET-UP ERROR

TMANUV FORMAT

TMANUV		GMTPS (1)	FIXED GMT SECONDS
	+1	GMTPC (1)	FIXED GMT SECONDS
	+2	A (DELTA TIME),,	D (THRUSTER IND.)
	+3	S/C PITCH	FLOATING PT. RADIANS
	+4	S/C YAW	FLOATING PT. RADIANS
	+5	GMTPS (2)	
	•		
	•		
	•		
	+49		

PRESENTLY TMANUV IS A 50-LOCATION TABLE, PERMITTING A MAXIMUM OF 10 MANEUVERS TO OCCUPY THE TABLE AT ANY ONE TIME.

TMORTN FORMAT

TMORTN		GMTPC OF LAST EXECUTED MANEUVER
	+1	GMTPS OF NEXT EXPECTED MANEUVER
	+2	GMTPC OF NEXT EXPECTED MANEUVER

USAGE

MYTABU ENTERS FROM AND EXITS TO MOPRIO.

A) TABLES -

- TMANUV
- TMZOOT
- TMORTN
- TMREFL
- TMREST
- TPTABU
- TMMATU
- TMCCDP
- TMCODW

B) CONSTANTS -

- K00001
- K0C005
- KLCAPE
- KMDISK

C) COMMUNICATION CELLS -

- MCMANC
- MCHFSC
- MCOWGT
- MCMTPC
- MCOMIP

D) SUBROUTINES -

U7INTP

A3MSCP

GEMWGT

GINTAB

E) PARAMETERS -

A

K

MNTABU

MNCUR1

MNMESS

F) MACROS -

TRNCN

TRNOF

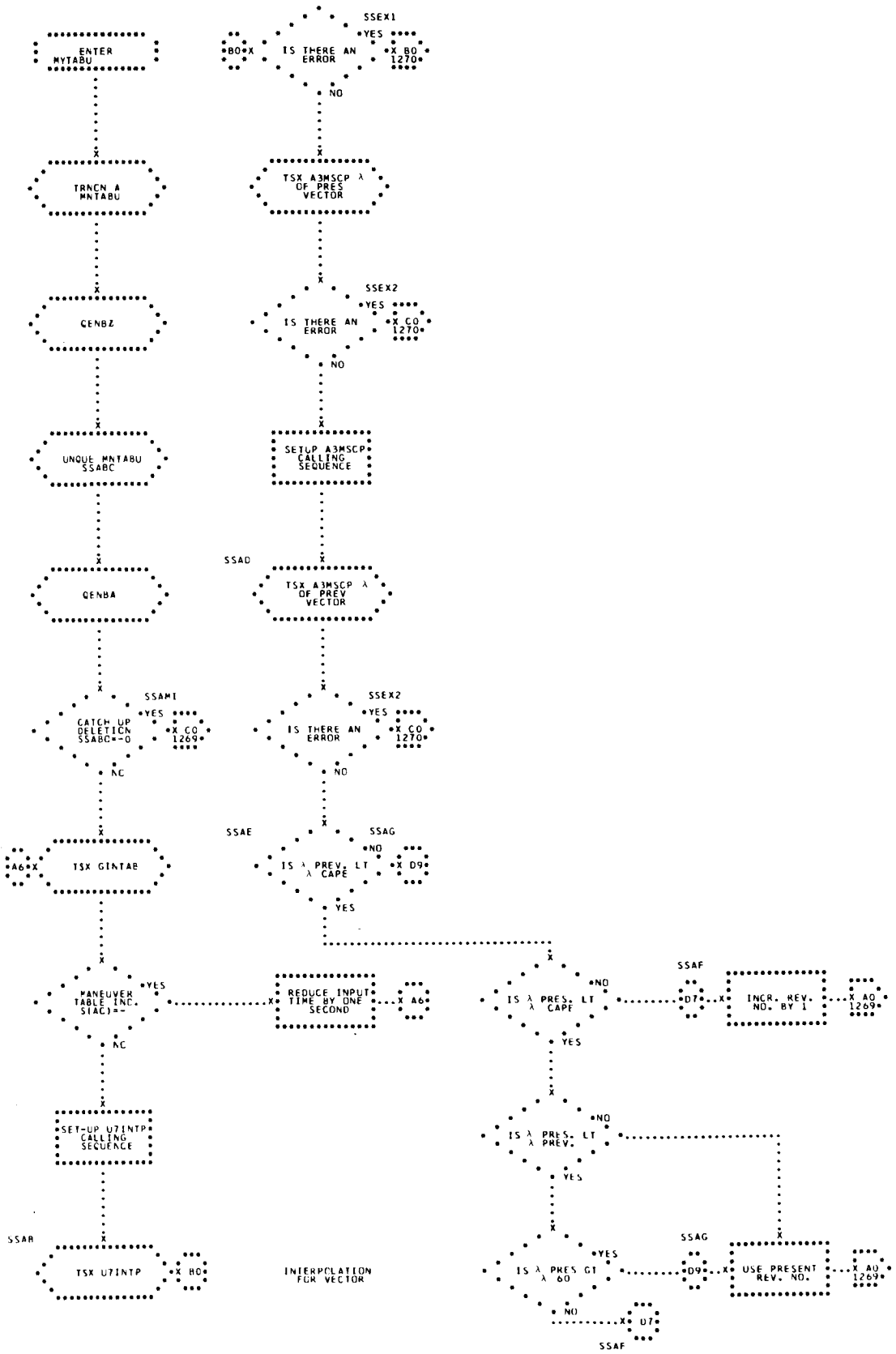
GENBZ

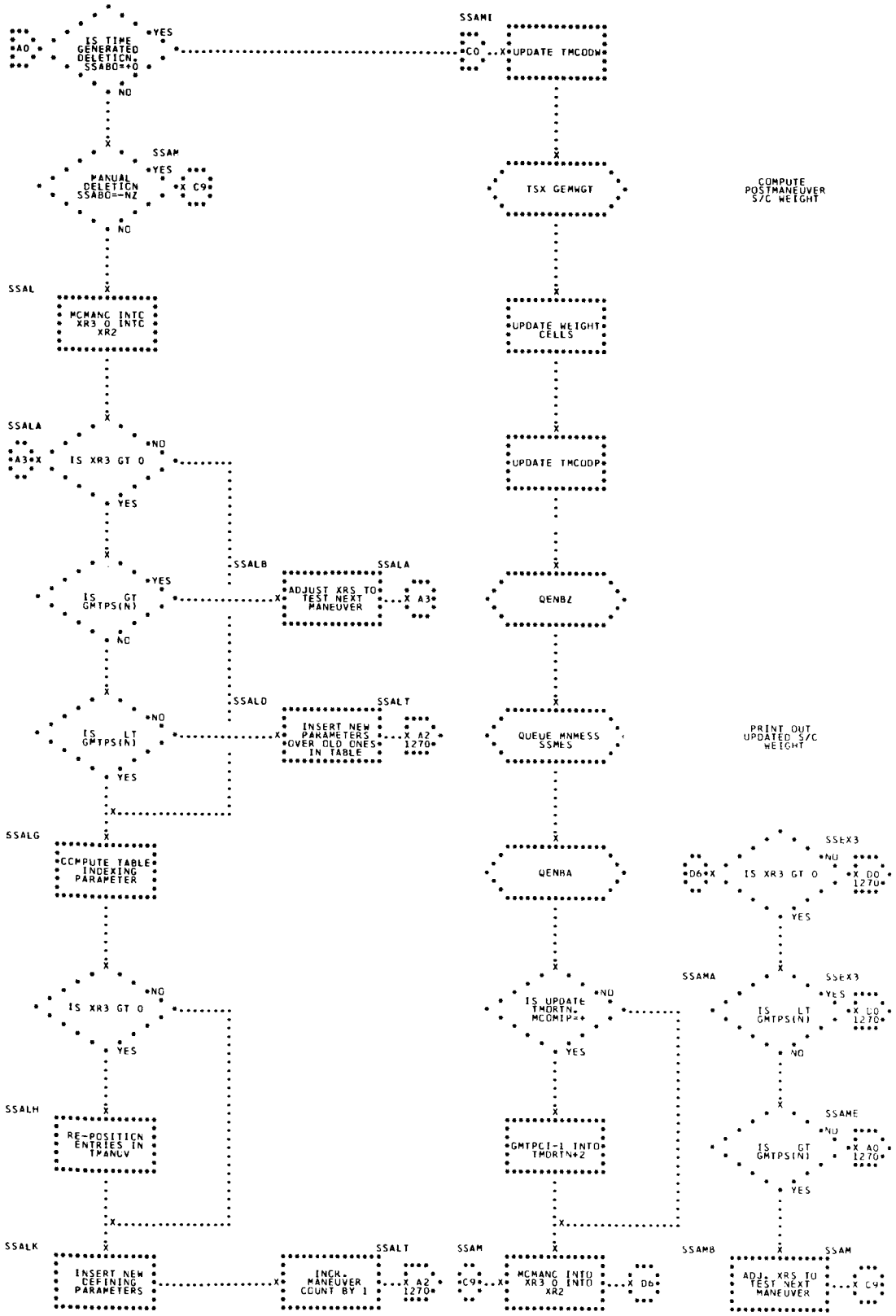
GENBA

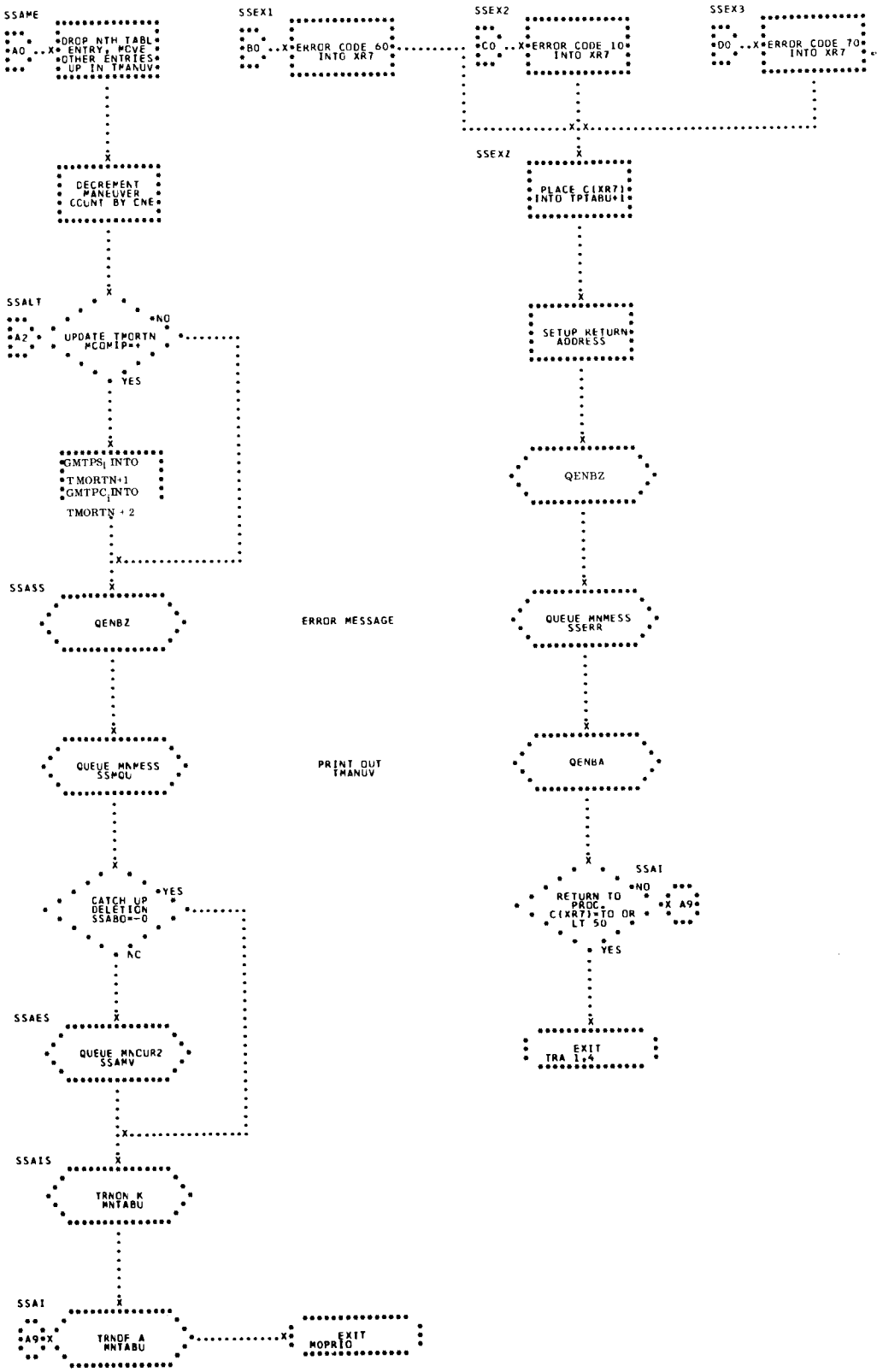
UNQUE

QUEUE

G) STORAGE REQUIRED - 220 LOCATIONS


ENTER MYTABU
TRNCN A MNTABU
GENBZ
UNQUE MNTABU SSABC
GENBA
CATCH UP DELETION SSABC=-0
SSAMI
YES
X CO 1269
NO
TSX GINTAB
A6
MANEUVER TABLE IND. SIAC)=-
YES
REDUCE INPUT TIME BY ONE SECOND
X A6
NO
SET-UP U7INTP CALLING SEQUENCE
SSAB
TSX U7INTP
X B0
INTERPOLATION FOR VECTOR
B0
IS THERE AN ERROR
SSEX1
YES
X B0 1270
NO
TSX A3MSCP λ OF PRES VECTOR
SSEX2
IS THERE AN ERROR
X CO 1270
SETUP A3MSCP CALLING SEQUENCE
SSAD
TSX A3MSCP λ OF PREV VECTOR
SSEX2
IS THERE AN ERROR
X CO 1270
SSAE
SSAG
IS λ PREV. LT λ CAPE
NO
X D9
YES
IS λ PRES. LT λ CAPE
NO
SSAF
D7
INCR. REV. NO. BY 1
X A0 1269
YES
IS λ PRES. LT λ PREV.
NO
YES
SSAG
IS λ PRES GT λ 60
YES
D9
USE PRESENT REV. NO.
X A0 1269
NO
D7
SSAF


IS TIME GENERATED DELETION, SSABO=+0
YES
NO
AO
SSAM
MANUAL DELETION SSABO=-NZ
YES
X C9
NO
SSAL
MCMANC INTC XR3 0 INTC XR2
SSALA
IS XR3 GT 0
NO
A3
YES
SSALB
IS GT GMTPS(N)
YES
NO
SSALA
ADJUST XRS TO TEST NEXT MANEUVER
X A3
SSALD
IS LT GMTPS(N)
NO
YES
SSALT
INSERT NEW PARAMETERS OVER OLD ONES IN TABLE
X A2 1270
SSALG
COMPUTE TABLE INDEXING PARAMETER
IS XR3 GT 0
NO
YES
SSALH
RE-POSITION ENTRIES IN TMANCV
SSALK
INSERT NEW DEFINING PARAMETERS
SSALT
INCR. MANEUVER COUNT BY 1
X A2 1270
SSAMI
CO
UPDATE TMCODW
TSX GEMWGT
UPDATE WEIGHT CELLS
UPDATE TMCODP
QENBZ
QUEUE MNMESS SSMES
QENBA
IS UPDATE TMORTN, MCOMIP=+
NO
YES
GMTPCI-1 INTO TMORTN+2
SSAM
C9
MCMANC INTO XR3 0 INTO XR2
X D6
COMPUTE POSTMANEUVER S/C WEIGHT
PRINT OUT UPDATED S/C WEIGHT
SSEX3
D6
IS XR3 GT 0
NO
X D0 1270
YES
SSAMA
IS LT GMTPS(N)
YES
NO
SSAME
IS GT GMTPS(N)
NO
YES
SSAMB
ADJ. XRS TO TEST NEXT MANEUVER
X C9


SSAME
A0
DROP NTH TABL ENTRY, MOVE OTHER ENTRIES UP IN TMANUV
DECREMENT MANEUVER COUNT BY ONE
SSALT
A2
UPDATE TMORTN MCOMIP=+
NO
YES
GMTPS1 INTO TMORTN+1
GMTPC1 INTO TMORTN + 2
SSASS
QENBZ
QUEUE MNMESS SSMOU
CATCH UP DELETION SSABO=-0
YES
NO
SSAES
QUEUE MNCUR2 SSAMV
SSAIS
TRNON K MNTABU
SSAI
A9
TRNOF A MNTABU
EXIT MOPRIO
ERROR MESSAGE
PRINT OUT TMANUV
SSEX1
B0
ERROR CODE 60 INTO XR7
SSEX2
C0
ERROR CODE 10 INTO XR7
SSEX3
D0
ERROR CODE 70 INTO XR7
SSEXZ
PLACE C(XR7) INTO TPTABU+1
SETUP RETURN ADDRESS
QENBZ
QUEUE MNMESS SSERR
QENBA
SSAI
RETURN TO PROC. C(XR7)=TO OR LT 50
NO
X A9
YES
EXIT TRA 1,4

86. VOMANV - PROPULSIONS PANEL COMPUTATIONS

VOMANV COMPUTES THE PARAMETERS NECESSARY TO PERFORM PRESCRIBED ORBITAL SPACECRAFT MANEUVERS. GIVEN THE TIME OF THE MANEUVER AND THE TYPE OF MANEUVER, VOMANV COMPUTES THE PITCH, YAW, DURATION OF BURN AND VELOCITY IMPULSE NECESSARY TO PERFORM A MANEUVER. AT THE PRESENT TIME VOMANV CAN PERFORM ANY ONE OF THE FOLLOWING MANEUVERS -

A) A CHANGE IN THE ORBITAL PLANE

B) RAISING AND/OR LOWERING EITHER APOGEE OR PERIGEE

C) CIRCULARIZING THE ORBIT OF THE SPACE CAPSULE.

(SEE FLOWCHART.)

INPUT

A) SUBROUTINES -

U3VMAG	U1SICO
U3VPRO	U1ASCO
U1SQRT	U3DOTP
U1ATAB	

B) PROGRAMS -

MYMUV7

GEMWGT

MPCPNI

C) TABLE -

A 4-LOCATION TABLE (TIPROP) CONTAINING IN THIS ORDER, GMTM,ΔH (OR S-THE WEDGE ANGLE), THRUSTER INDICATOR, AND THE TYPE OF MANUEVER REQUESTED (0-APOGEE KICK, 1-CIRCULARIZATION, 2-PLANE CHANGE). A 6-LOCATION TABLE CONTAINING THE POSITION VECTOR R_0 AND THE VELOCITY VECTOR V AT

GMTM

D) CONSTANTS -

TOSCCH	TOSCCH+6
TOSCCH+1	TOSCCH+7
TOSCCH+2	K00060
TOSCCH+3	K00001
K002.0	K00.PI
K001.0	KCFTSC
KCC003	K004.0
KCC002	KACCDG
KCC004	KCH233

OUTPUT

A TABLE CONTAINING THE FOLLOWING VALUES -

VELOCITY IMPULSE (ΔV) - NECESSARY TO PERFORM THE MANEUVER (FT/SEC)

PITCH - FLOATING POINT RADIANS

YAW - FLOATING POINT RADIANS

GMTPI - GREENWICH TIME TO INITIATE PROPULSION (FIXED SECONDS)

TOTAL LENGTH OF THRUST PERIOD (ΔT) - (FIXED SECONDS)

METHOD

EACH OF THE POSSIBLE TYPES OF MANEUVERS ARE DISCUSSED SEPARATELY.

A) ADJUSTMENT OF PERIGEE OR APOGEE

TO ADJUST PERIGEE OR APOGEE HEIGHT (MEMORANDUM FOR RECORD, SERIAL #90, FILE 14.0/12.7, ''ADJUSTING PERIGEE HEIGHT'') THE DIRECTION OF THE IMPULSE IS SPECIFIED AND THE SOLUTION USED BE FOR MAGNITUDE ONLY. ASSUME THAT IT IS REQUESTED

TO RAISE PERIGEE BY ΔH. CONSIDER THE INPUTTED POSITION VECTOR R_o AND THE VELOCITY VECTOR V_o. COMPUTE

$$|R_0| = (X_0^2 + Y_0^2 + Z_0^2)^{1/2}$$

$$|V_0| = (\dot{X}_0^2 + \dot{Y}_0^2 + \dot{Z}_0^2)^{1/2}$$

$$A_0 = 1/(2/|R_0| - |V_o|^2)$$

A

WHERE

A = SEMIMAJOR AXIS

$$V_K = \left(\frac{2}{|R_0|} - \frac{1}{A_0 + \Delta H/2}\right)^{1/2}$$

WHERE

V_K = FIRST GUESS OF THE VELOCITY REQUIRED TO RAISE PERIGEE BY ΔH.

$$\Delta V_K = |V_K| - |V_o|$$

THE VELOCITY IMPULSE ΔV_K IS ADDED TANGENTIALLY TO V_0 -

$$\dot{X}_0' = \dot{X}_0 (1 + \Delta V_K / |V_0|)$$

$$\dot{Y}_0' = \dot{Y}_0 (1 + \Delta V_K / |V_0|)$$

$$\dot{Z}_0' = \dot{Z}_0 (1 + \Delta V_K / |V_0|)$$

USING THE INPUTTED R_0 AND V_0, INTEGRATE TO PERIGEE USING PHASE ANGLE LOGIC AND COMPUTE

$$|R_P| = (X_P^2 + Y_P^2 + Z_P^2)^{1/2}$$

NEXT, INTEGRATE TO PERIGEE USING R_0 AND V_0^1 AS THE VECTOR. COMPUTE

$$|R'_P| = (X'^2_P + Y'^2_P + Z'^2_P)^{1/2}$$

$$\Delta R_P = |R'_P| - |R_P|$$

TEST TO SEE WHETHER CONVERGENCE HAS BEEN OBTAINED BY COMPARING ΔR_P WITH ΔH. IF CONVERGENCE IS NOT OBTAINED COMPUTE A NEW ESTIMATE OF VELOCITY ΔV_0.

C

WHERE $C_s = \Delta R_P / \Delta V_K$

C_S = SENSITIVITY COEFFICIENT.

$$\Delta V_0 = \frac{\Delta H}{C_s}$$

APPLY ΔV_0 TO THE ORIGINAL APOGEE AND VELOCITY VECTOR AS BEFORE. REPEAT THE ABOVE STEPS UNTIL CONVERGENCE HAS BEEN OBTAINED (ITERATION LIMIT OF 4).

B) CIRCULARIZING THE ORBIT OF THE CAPSULE

TO CIRCULARIZE (MEMORANDUM FOR RECORD, SERIAL 97, FILE 14.0/12.7, "CIRCULARIZATION") ANYWHERE THE ORBIT OF THE SPACE CAPSULE CONSIDER THE POSITION VECTOR R_A AND A VELOCITY VECTOR V_A AT THAT POINT. AS A FIRST GUESS OF THE REQUIRED VELOCITY, A KEPLERIAN SOLUTION IS CHOSEN. COMPUTE A NEW COORDINATE SYSTEM -

$$H = R_A \times V_A$$

$$G^* = \frac{H \times R_A}{|H \times R_A|}$$

WHERE

G^* = THE DIRECTION COSINES

$V_{cir} = (1/|R_A|)^{1/2}$

$V_0 = (V_{cir})(G^*)$

$|V_0| = (X_0^2 + Y_0^2 + Z_0^2)^{1/2}$

USING THE INPUTTED POSITION VECTOR R_A AND THE COMPUTED VELOCITY VECTOR V_0, INTEGRATE FOR ONE HALF A PERIOD (180 DEGREES) USING PHASE ANGLE LOGIC AND COMPUTE

$|R_P| = (X_P^2 + Y_P^2 + Z_P^2)^{1/2}$

$\Delta R_P = |R_A| - |R_P|$

FOR A CIRCULAR ORBIT, $|R_P|$ WILL EQUAL $|R_A|$. HOWEVER, DUE TO DRAG AND OBLATENESS OF THE EARTH THE FIRST GUESS OF THE COMPUTED VELOCITY VECTOR WILL NOT BE ACCURATE ENOUGH. THE FOLLOWING EXPRESSION GIVES A GOOD APPROXIMATION OF THE VELOCITY VECTOR REQUIRED TO CIRCULARIZE -

$$|V_K| = \left(\frac{2}{|R_A|} - \frac{1}{|R_A| + \frac{\Delta R_P}{2}}\right)^{1/2}$$

$\Delta V_K = |V_K| - |V_0|$

AS WITH RAISING PERIGEE AT APOGEE, THE VELOCITY IMPULSE ΔV_K IS ADDED TANGENTIALLY TO V_0. THE REMAINDER OF THE COMPUTATIONS ARE IDENTICAL WITH THOSE USED IN ADJUSTING PERIGEE OR APOGEE HEIGHT.

C) CHANGE IN THE ORBITAL PLANE

A CHANGE IN ORBITAL PLANE (MEMORANDUM FOR RECORD, SERIAL 89, FILE 14.0/12.7, ''PLANE CHANGE'') IS PERFORMED WHILE MAINTAINING THE SAME VELOCITY MAGNITUDE AND FLIGHT PATH ANGLE. CONSIDER A POSITION VECTOR R_0 AND A VELOCITY VECTOR V_0 IT IS DESIRED TO CHANGE THE ORBITAL PLANE BY σ RADIANS (THE WEDGE ANGLE). COMPUTE

$$|R_0| = (X_0^2 + Y_0^2 + Z_0^2)^{1/2}$$

$$|V_0| = (\dot{X}_0^2 + \dot{Y}_0^2 + \dot{Z}_0^2)^{1/2}$$

$$\gamma = \text{SIN}^{-1} \frac{R_0 \cdot V_0}{|R_0|\,|V_0|}$$

WHERE

γ = flight path angle

$\sigma = 1/2\,(180^{\circ} - \delta)$

$\Delta V_1 = 2\,|V_0|\,\text{SIN}\,(\delta/2)$

REDEFINE THE SPACE CAPSULE COORDINATE SYSTEM. LET -

$F = R_0$

$H = R_0 \times V_0$

$G = H \times R_0$

LET THE DIRECTION COSINES OF -

F = FA, FB, FC

H = HA, HB, HC

G = GA, GB, GC

THE VELOCITY IMPULSE IN THE F, G, H COORDINATE SYSTEM IS-

$$\Delta V_F = 0$$

$$\Delta V_G = \Delta V_1 \frac{\cos\alpha}{\cos\gamma}$$

$$\Delta V_H = (\Delta V_1^2 - \Delta V_G^2)^{1/2}$$

IN THE EARTH CENTERED INERTIAL CORDINATE SYSTEM -

$$\Delta\dot{X} = (\Delta V_G)(GA) + (\Delta V_H)(HA)$$

$$\Delta\dot{Y} = (\Delta V_G)(GB) + (\Delta V_H)(HB)$$

$$\Delta\dot{Z} = (\Delta V_G)(GC) + (\Delta V_H)(HC),$$

$$\Delta V = (\Delta\dot{X}^2 + \Delta\dot{Y}^2 + \Delta\dot{Z}^2)^{1/2}$$

TO CONVERT THE VELOCITY IMPULSE (ΔV) NECESSARY TO PERFORM A MANEUVER INTO PITCH, YAW, GMTPI AND TOTAL LENGTH OF THRUST PERIOD (ΔT) THE FOLLOWING COMPUTATIONS ARE MADE (MEMORANDUM FOR RECORD, SERIAL 112, FILE 7.3.3 ''DIGITAL MANEUVER UPDATES FOR GT-3.'') -

$$|\Delta V| = (\Delta\dot{X}^2 + \Delta\dot{Y}^2 + \Delta\dot{Z}^2)$$

$$\Delta t = 2M \cdot \Delta V/(2F \cdot \cos\xi + \dot{M} \cdot \Delta V)$$

WHERE

M = MASS OF CAPSULE AT GMTM (SLUGS)

F = THRUST LEVEL (POUNDS)

$\dot{M}$ = MASS EXPULSION RATE (SLUGS/SEC)

ξ = CAN'T ANGLE (RADIANS)

$$V_i = Vi/|V_0| \qquad i = 1, 2, 3$$

$$R_i = R_i/|R_0| \qquad i = 1, 2, 3$$

$$U = V_0 \times R_0$$

$$W = R_o \times U$$

$$|U| = (U_1^2 + U_2^2 + U_3^2)^{1/2}$$

$$|W| = (W_1^2 + W_2^2 + W_3^2)^{1/2}$$

$$U_i = U_i/|U| \qquad i = 1, 2, 3,$$

$$W_i = W_i/|W| \qquad i = 1, 2, 3$$

$$\Delta V - R_0 = \Delta\dot{X} \cdot R_1 + \Delta\dot{Y} \cdot R_2 + \Delta\dot{Z} \cdot R_3$$

$$(\Delta V)^2 = (\Delta\dot{X})^2 + (\Delta\dot{Y})^2 + (\Delta\dot{Z})^2$$

$$\Delta V \cdot U = \Delta\dot{X} \cdot U_1 + \Delta\dot{Y} \cdot U_2 + \Delta\dot{Z} \cdot U_3$$

$$\Delta V \cdot W = \Delta\dot{X} \cdot W_1 + \Delta\dot{Y} \cdot W_2 + \Delta\dot{Z} \cdot W_3$$

$$\text{Pitch } (\alpha) = \tan^{-1} \frac{\Delta V \cdot R}{(|\Delta V \cdot U|^2 + |\Delta V \cdot W|^2)^{1/2}}$$

$$\text{Yaw } (\beta) = \tan^{-1} \frac{\Delta V \cdot U}{\Delta V \cdot W}$$

$$Y_P = -R_0/|R_0|$$

$$Z_P = -U/|U|$$

$$X_P = Y_P \times Z_P/|Y_P \times Z_P|$$

$$\Delta\dot{X}_P = \Delta V \cdot X_P$$

$$\Delta\dot{Y}_P = \Delta V \cdot Y_P$$

$$\Delta\dot{Z}_P = \Delta V \cdot Z_P$$

$$\Delta V_P = (\Delta\dot{X}_P^2 + \Delta\dot{Y}_P^2 + \Delta\dot{Z}_P)^{1/2}$$

$$\Delta V = \left((\Delta\dot{X})^2 + (\Delta\dot{Y})^2 + (\Delta\dot{Z})^2\right)^{1/2} \quad 25936.272 \text{ (ft/sec)}$$

$$\Delta t = 2M \cdot \Delta V/(2F\cos\xi + \dot{M} \cdot \Delta V)$$

WHERE

Δt = DURATION OF BURN

M = MASS OF S/C

F = THRUST LEVEL

$\dot{M}$ = MASS EXPULSION RATE

GMTB = GMTM - $\Delta T/2$

WHERE

GMTB = GREENWICH MERIDIAN TIME TO BURN

GMTM = GREENWICH MERIDIAN TIME TO MANEUVER.

USAGE

A) CALLING SEQUENCE -

```
TSX     VOMANV,4
+1      ERROR RETURN
+2      NORMAL RETURN
```

B) ERROR CONDITIONS -

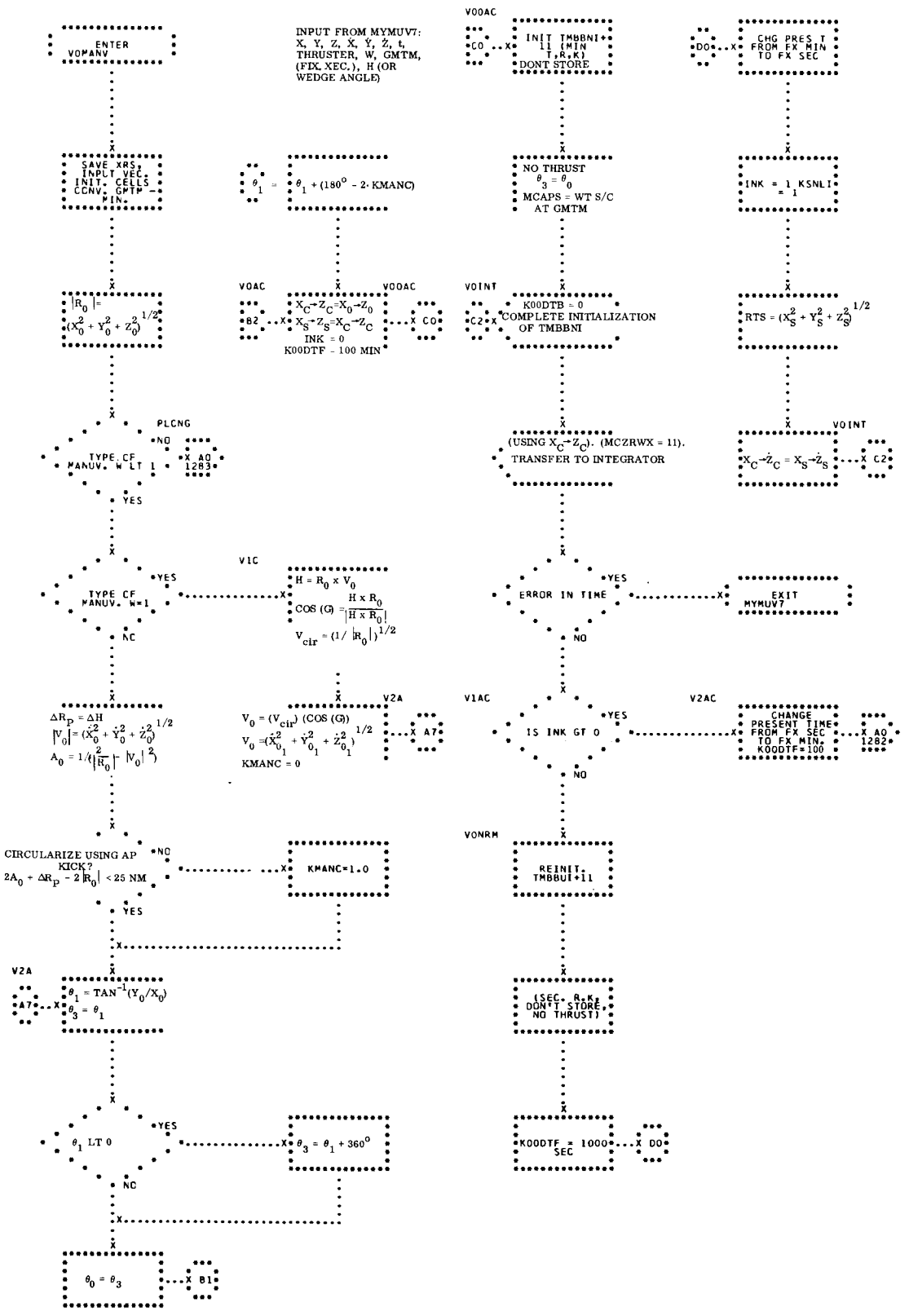
TIME ERROR

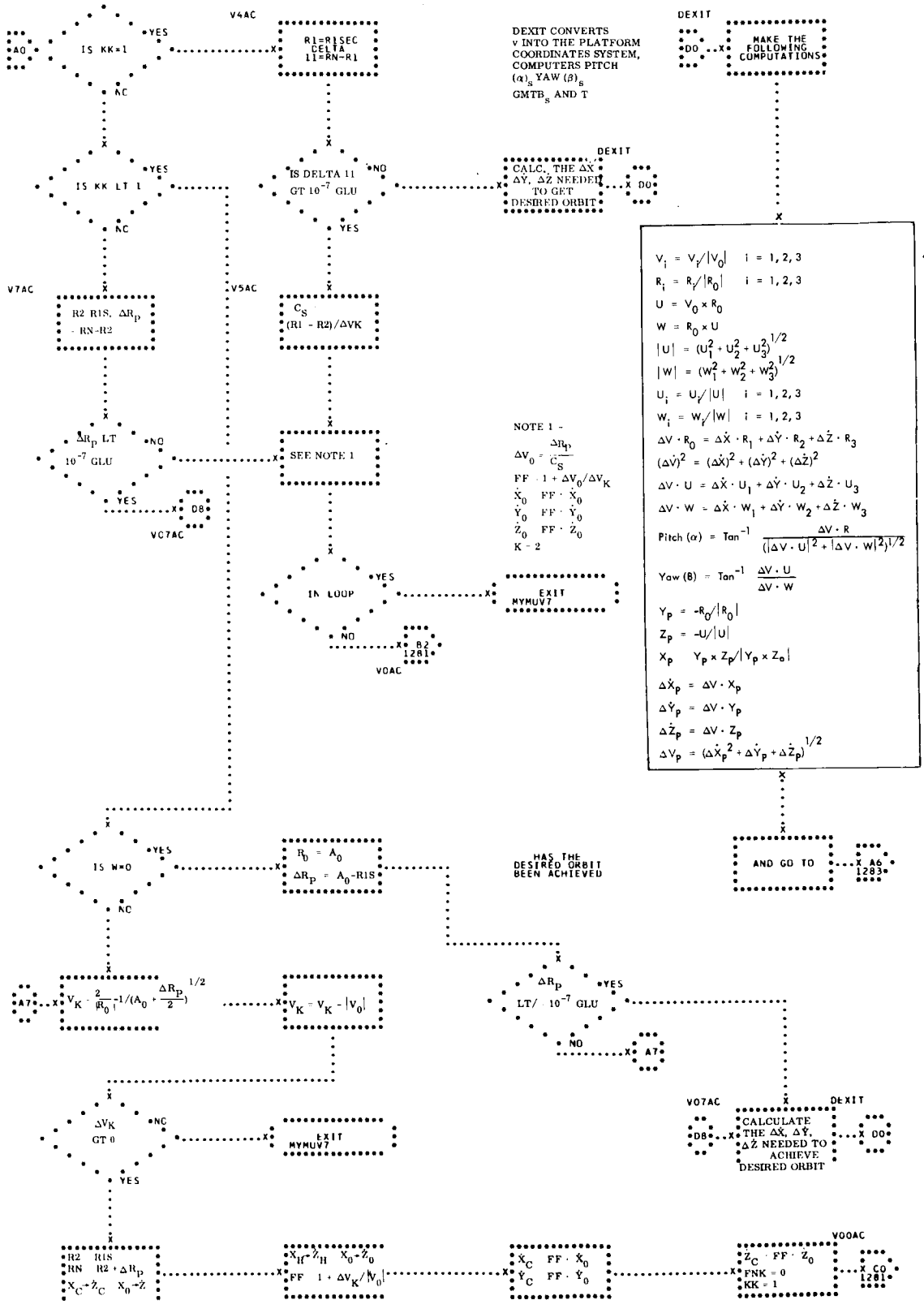
LOOPING ERROR - NON-CONVERGENCE

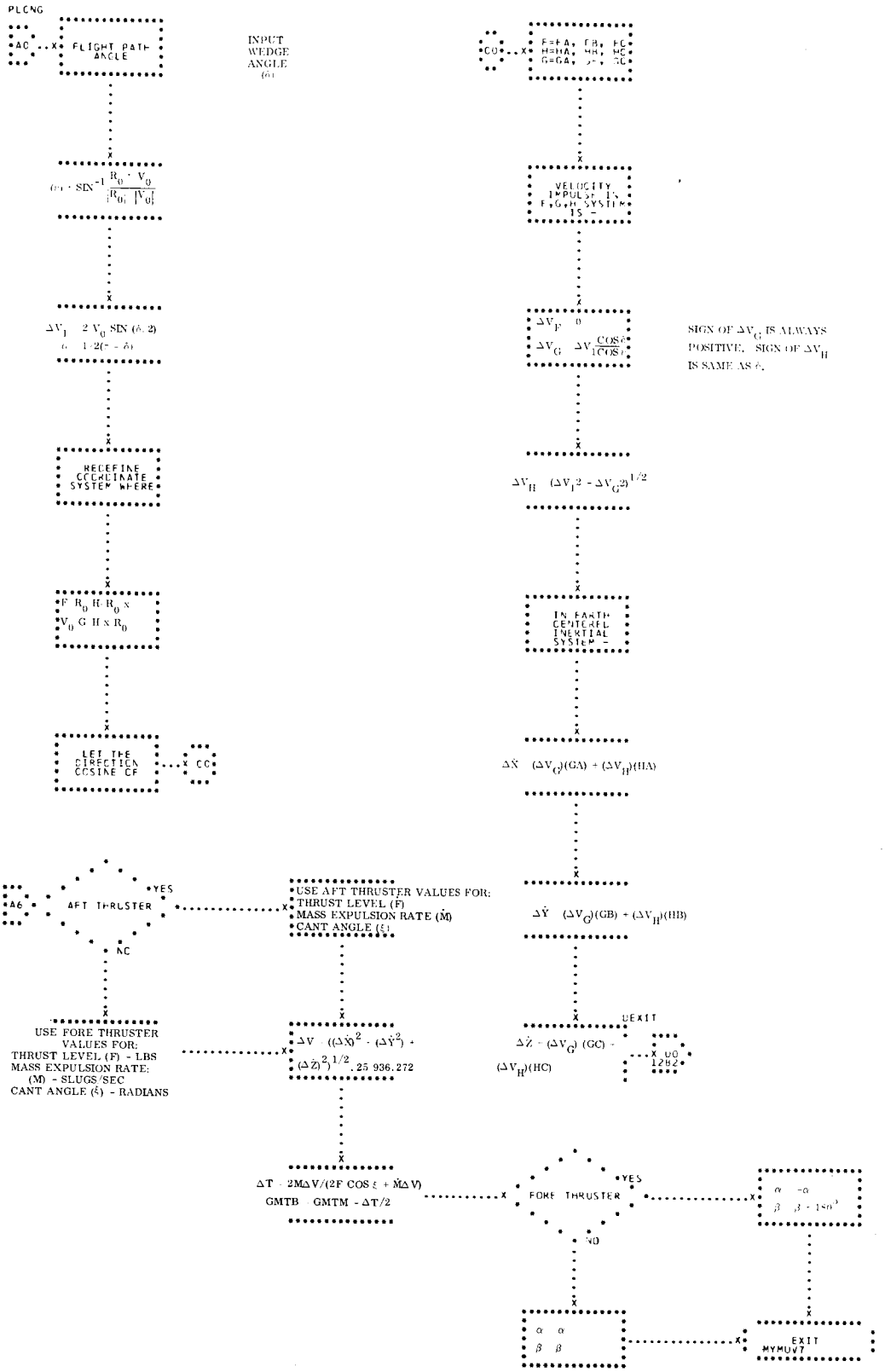
VELOCITY IMPULSE - $(\Delta V_K) = \phi$

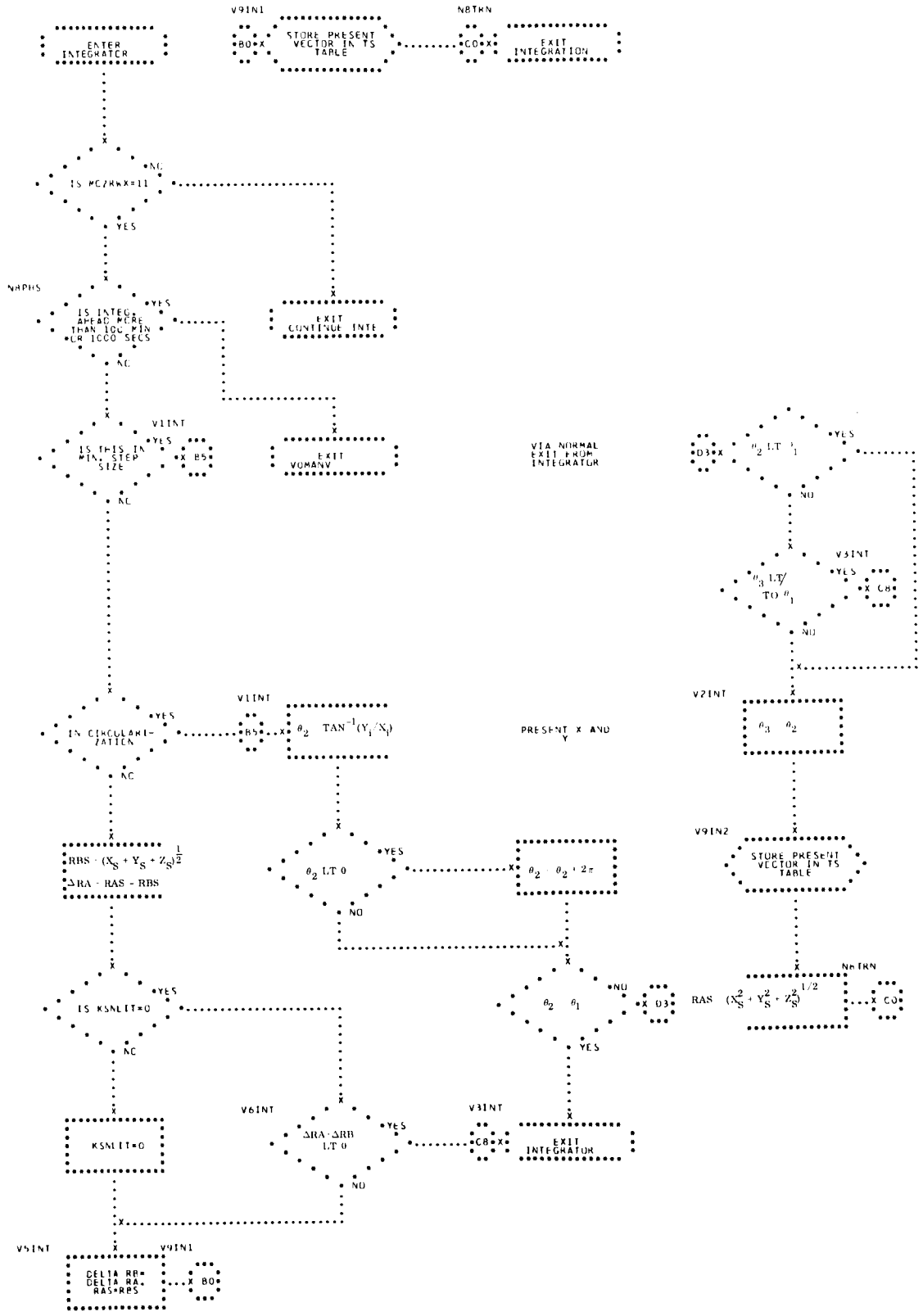
C) STORAGE LOCATION -

775


ENTER VOMANV
INPUT FROM MYMUV7: X, Y, Z, Ẋ, Ẏ, Ż, t, THRUSTER, W, GMTM, (FIX, XEC.), H (OR WEDGE ANGLE)
V00AC
INIT TMBBNI+11 (MIN T,R,KI DONT STORE
CO
CHG PRES T FROM FX MIN TO FX SEC
DO
SAVE XRS, INPUT VEC, INIT. CELLS CONV. GMTM = MIN.
θ1 = θ1 + (180° - 2·KMANC)
NO THRUST θ3 = θ0 MCAPS = WT S/C AT GMTM
INK = 1 KSNLI = 1
|R0| = (X0² + Y0² + Z0²)^1/2
V0AC
B2
XC→ZC=X0→Z0 XS→ZS=XC→ZC INK = 0 K00DTF - 100 MIN
V00AC
CO
V0INT
C2
K00DTB = 0 COMPLETE INITIALIZATION OF TMBBNI
RTS = (XS² + YS² + ZS²)^1/2
PLCNG
TYPE OF MANUV. W LT 1
NO
X AQ 1283
YES
(USING XC→ZC). (MCZRWX = 11). TRANSFER TO INTEGRATOR
V0INT
XC→ŻC = XS→ŻS
X C2
V1C
TYPE OF MANUV. W=1
YES
NO
H = R0 × V0
COS (G) = H × R0 / |H × R0|
Vcir = (1/|R0|)^1/2
ERROR IN TIME
YES
NO
EXIT MYMUV7
ΔRP = ΔH
|V0| = (Ẋ0² + Ẏ0² + Ż0²)^1/2
A0 = 1/|R0| - |V0|²
V0 = (Vcir) (COS (G))
V0 = (Ẋ0₁² + Ẏ0₁² + Ż0₁²)^1/2
KMANC = 0
V2A
A7
V1AC
IS INK GT 0
YES
NO
V2AC
CHANGE PRESENT TIME FROM FX SEC TO FX MIN. K00DTF=100
X AQ 1282
CIRCULARIZE USING AP KICK?
2A0 + ΔRP - 2|R0| < 25 NM
NO
YES
KMANC=1.0
VONRM
REINIT. TMBBUI+11
V2A
A7
θ1 = TAN⁻¹(Y0/X0)
θ3 = θ1
(SEC. R,K, DON'T STORE, NO THRUST)
θ1 LT 0
YES
NO
θ3 = θ1 + 360°
K00DTF = 1000 SEC
X DO
θ0 = θ3
X B1


V4AC
A0
IS KK=1
YES
NO
R1=R1SEC
DELTA
11=RN-R1
IS KK LT 1
IS DELTA 11
GT 10^{-7} GLU
DEXIT
CALC. THE ΔX, ΔY, ΔŻ NEEDED TO GET DESIRED ORBIT
DO
DEXIT CONVERTS v INTO THE PLATFORM COORDINATES SYSTEM, COMPUTERS PITCH $(\alpha)_S$ YAW $(\beta)_S$ $GMTB_S$ AND T
MAKE THE FOLLOWING COMPUTATIONS
V7AC
R2 R1S, ΔR_P
= RN-R2
V5AC
C_S
(R1 - R2)/ΔVK
$V_i = V_i/|V_0|$ i = 1, 2, 3
$R_i = R_i/|R_0|$ i = 1, 2, 3
$U = V_0 \times R_0$
$W = R_0 \times U$
$|U| = (U_1^2 + U_2^2 + U_3^2)^{1/2}$
$|W| = (W_1^2 + W_2^2 + W_3^2)^{1/2}$
$U_i = U_i/|U|$ i = 1, 2, 3
$W_i = W_i/|W|$ i = 1, 2, 3
$\Delta V \cdot R_0 = \Delta\dot{X} \cdot R_1 + \Delta\dot{Y} \cdot R_2 + \Delta\dot{Z} \cdot R_3$
$(\Delta V)^2 = (\Delta\dot{X})^2 + (\Delta\dot{Y})^2 + (\Delta\dot{Z})^2$
$\Delta V \cdot U = \Delta\dot{X} \cdot U_1 + \Delta\dot{Y} \cdot U_2 + \Delta\dot{Z} \cdot U_3$
$\Delta V \cdot W = \Delta\dot{X} \cdot W_1 + \Delta\dot{Y} \cdot W_2 + \Delta\dot{Z} \cdot W_3$
$\text{Pitch } (\alpha) = \text{Tan}^{-1} \dfrac{\Delta V \cdot R}{(|\Delta V \cdot U|^2 + |\Delta V \cdot W|^2)^{1/2}}$
$\text{Yaw } (B) = \text{Tan}^{-1} \dfrac{\Delta V \cdot U}{\Delta V \cdot W}$
$Y_P = -R_0/|R_0|$
$Z_P = -U/|U|$
$X_P \quad Y_P \times Z_P/|Y_P \times Z_0|$
$\Delta\dot{X}_P = \Delta V \cdot X_P$
$\Delta\dot{Y}_P = \Delta V \cdot Y_P$
$\Delta\dot{Z}_P = \Delta V \cdot Z_P$
$\Delta V_P = (\Delta\dot{X}_P^2 + \Delta\dot{Y}_P + \Delta\dot{Z}_P)^{1/2}$
ΔR_P LT 10^{-7} GLU
SEE NOTE 1
NOTE 1 -
$\Delta V_0 = \dfrac{\Delta R_P}{C_S}$
FF · 1 + $\Delta V_0/\Delta V_K$
$\dot{X}_0$ FF · $\dot{X}_0$
$\dot{Y}_0$ FF · $\dot{Y}_0$
$\dot{Z}_0$ FF · $\dot{Z}_0$
K - 2
D8
VC7AC
IN LOOP
EXIT MYMUV7
R2 1281
VOAC
AND GO TO
X A6 1283
IS W=0
$R_0 = A_0$
$\Delta R_P = A_0$-R1S
HAS THE DESIRED ORBIT BEEN ACHIEVED
A7
$V_K = \dfrac{2}{|R_0|} - 1/(A_0 + \dfrac{\Delta R_P}{2})^{1/2}$
$V_K = V_K - |V_0|$
ΔR_P LT/ 10^{-7} GLU
ΔVK GT 0
EXIT MYMUV7
VO7AC
CALCULATE THE ΔẊ, ΔẎ, ΔŻ NEEDED TO ACHIEVE DESIRED ORBIT
R2 R1S
RN R2 + ΔR_P
X_C - $\dot{Z}_C$ $\dot{X}_0$ - $\dot{Z}$
X_H - $\dot{Z}_H$ $\dot{X}_0$ - $\dot{Z}_0$
FF 1 + $\Delta V_K/|V_0|$
$\dot{X}_C$ FF · $\dot{X}_0$
$\dot{Y}_C$ FF · $\dot{Y}_0$
$\dot{Z}_C$ FF · $\dot{Z}_0$
FNK = 0
KK = 1
VOOAC
X C0 1281


PLONG
AC
FLIGHT PATH ANGLE
INPUT WEDGE ANGLE (δ)
CO
F=FA, FB, FC
H=HA, HB, HC
G=GA, GB, GC
VELOCITY IMPULSE IN F,G,H SYSTEM IS -
ΔV_F = 0
ΔV_G = ΔV_1 COS δ / COS ε
SIGN OF ΔV_G IS ALWAYS POSITIVE. SIGN OF ΔV_H IS SAME AS δ.
ΔV_H = (ΔV_1^2 - ΔV_G^2)^1/2
IN EARTH CENTERED INERTIAL SYSTEM -
ΔẊ = (ΔV_G)(GA) + (ΔV_H)(HA)
ΔẎ = (ΔV_G)(GB) + (ΔV_H)(HB)
DEXIT
ΔŻ = (ΔV_G)(GC) + (ΔV_H)(HC)
X DO 1282
REDEFINE COORDINATE SYSTEM WHERE
F = R_0 H = R_0 x V_0 G = H x R_0
LET THE DIRECTION COSINE OF
CO
A6
AFT THRUSTER
YES
NO
USE AFT THRUSTER VALUES FOR:
THRUST LEVEL (F)
MASS EXPULSION RATE (Ṁ)
CANT ANGLE (ξ)
USE FORE THRUSTER VALUES FOR:
THRUST LEVEL (F) - LBS
MASS EXPULSION RATE: (M) - SLUGS/SEC
CANT ANGLE (ξ) - RADIANS
ΔV = ((ΔẊ)^2 + (ΔẎ)^2) + (ΔŻ)^2)^1/2 . 25 936.272
ΔT = 2MΔV/(2F COS ξ + ṀΔV)
GMTB = GMTM - ΔT/2
FORE THRUSTER
YES
NO
α = -α
β = β + 180°
α = α
β = β
EXIT
MYMUV7


ENTER INTEGRATOR
V9IN1
B0
STORE PRESENT VECTOR IN TS TABLE
N8TRN
C0
EXIT INTEGRATION
IS MC/RWX=11
NO
YES
N4PHS
IS INTEG. AHEAD HERE THAN 10C MIN OR 1CCC SECS
YES
NO
EXIT CONTINUE INTE
V1INT
IS THIS IN MIN. STEP SIZE
YES
B5
NO
EXIT VOMANV
VIA NORMAL EXIT FROM INTEGRATOR
D3
θ_2 LT θ_1
YES
NO
V3INT
θ_3 LT θ_1 TO θ_1
YES
C8
NO
V2INT
θ_3 θ_2
V1INT
IN CIRCULARI-ZATION
YES
B5
NO
θ_2 $\tan^{-1}(Y_1/X_1)$
PRESENT X AND Y
V9IN2
STORE PRESENT VECTOR IN TS TABLE
RBS $(X_S^2 + Y_S^2 + Z_S^2)^{\frac{1}{2}}$
ΔRA = RAS - RBS
θ_2 LT 0
YES
NO
θ_2 $\theta_2 + 2\pi$
N8TRN
RAS $(X_S^2 + Y_S^2 + Z_S^2)^{1/2}$
C0
θ_2 θ_1
NO
YES
D3
IS KSNLIT=0
YES
NO
KSNLIT=0
V6INT
ΔRA-ΔRB LT 0
YES
NO
V3INT
C8
EXIT INTEGRATOR
V5INT
V9IN1
DELTA RR= DELTA RA. RAS=RBS
B0

87. A3MSCP - SUBROUTINE TO COMPUTE LATITUDE AND LONGITUDE

GIVEN AN INERTIAL POSITION VECTOR, $\vec{R}$, OF THE SPACECRAFT AT A PARTICULAR TIME, A3MSCP COMPUTES THE GEODETIC LATITUDE (ϕ) AND LONGITUDE (λ) OF THE SPACECRAFT POSITION. IF DESIRED, THE S/C ALTITUDE (H) WILL ALSO BE COMPUTED. (SEE FLOWCHART)

INPUT

A) A 4-LOCATION TABLE CONTAINING TIME, X, Y, AND Z.
   1. TIME IS EITHER FLOATING POINT SECONDS OR FIXED POINT MINUTES AND SECONDS. (SEE USAGE)
   2. X, Y, AND Z (COMPONENTS OF POSITION VECTOR) ARE IN FLOATING POINT GEMINI LENGTH UNITS.

B) A 7-LOCATION WORKING BUFFER IS ALSO SPECIFIED IN THE CALLING SEQUENCE.

UTILIZATION OF SUCH A BUFFER PERMITS A3MSCP TO FUNCTION AS A TRUE SUBROUTINE.

C)

K.INB2 ($1/b^2$)	WHERE b = SEMI-MINOR AXIS OF GEMINI SPHEROID
K.B2GU (b^2)	
K.ECC2 (e^2)	WHERE e = ECCENTRICITY OF GEMINI SPHERIOD ROTATION OF EARTH IN
K.OMES	RAD./SEC.
KLAMDO	GREENWICH HOUR ANGLE (λ_0)

OUTPUT

A 3-LOCATION TABLE CONTAINING λ AND ϕ IN FLOATING POINT RADIANS, AND IF REQUESTED H IN FLOATING POINT

GEMINI LENGTH UNITS. λ RANGES FROM 0 TO 2π RADIANS AND ϕ FROM $-\pi/2$ TO $+\pi/2$ RADIANS.

METHOD

THE FOLLOWING EQUATIONS ARE USED BY THE A3MSCP PROGRAM -

$$\tan\varphi = \frac{e^2 N \sin\varphi + Z}{\sqrt{X^2 + Y^2}} \tag{1}$$

$$H = (\sec\varphi)\left(\sqrt{X^2 + Y^2}\right) - N \tag{2}$$

$$\lambda = \tan^{-1}\frac{Y}{X} - \omega t - \lambda_g \tag{3}$$

$$N = \frac{a}{\sqrt{1 - e^2\sin^2\varphi}} = \frac{1}{\sqrt{1 - e^2\sin^2\varphi}} \tag{4}$$

$$\tan\varphi'' = \frac{b^2}{a^2}\tan\varphi = b^2\tan\varphi \tag{5}$$

$$\tan\varphi''' = \frac{a^2}{b^2}\tan\varphi' = \frac{1}{b^2}\tan\varphi' \tag{6}$$

THE DERIVATION OF EQUATIONS (1) AND (2) IS SHOWN IN PARAGRAPH 2.21.5. EQUATION (3) IS A STANDARD EXPRESSION OF IN THE GEMINI SYSTEM. EQUATIONS (4), (5), AND (6) ARE STANDARD GOEDESY TEXTBOOK FORMULAS.

IF A3MSCP IS ENTERED WITH TIME IN FIXED-POINT, IT IS IMMEDIATELY CONVERTED TO FLOATING-POINT SECONDS FOR LATER USE IN SOLVING EQUATION (3).

AS A FIRST ESTIMATE OF ϕ, THE ANGLE ϕ''' (GEODETIC LATITUDE OF THE INTERSECTION OF THE GIVEN RADIUS VECTOR WITH THE SURFACE OF THE EARTH) IS USED AND IT IS DESIGNATED ϕ_i. THUS, BY EQUATION (6) $\tan\phi_i = (1/b^2)\tan\phi' = (1/b^2)\, Z/\sqrt{X^2 + Y^2}$.

THE SEC ϕ_i, SIN ϕ_i, AND $1/N_i$ VALUES ARE COMPUTED

USING TAN ϕ_i . TAN ϕ_{i+1} MAY THEN BE COMPUTED UTILIZING THE NEW SIN ϕ_i AND $1/N_i$ VALUES IN EQUATION (1).

THE COMPUTED VALUE OF TAN ϕ_{i+1} IS USED AS THE NEW ESTIMATE OF TAN ϕ AND A SECOND ITERATION OF THE COMPUTATION IS MADE. THE NEW VALUE OF TAN ϕ_{i+1} IS THE TRUE VALUE OF TAN ϕ CORRECT TO EIGHT DECIMAL PLACES.

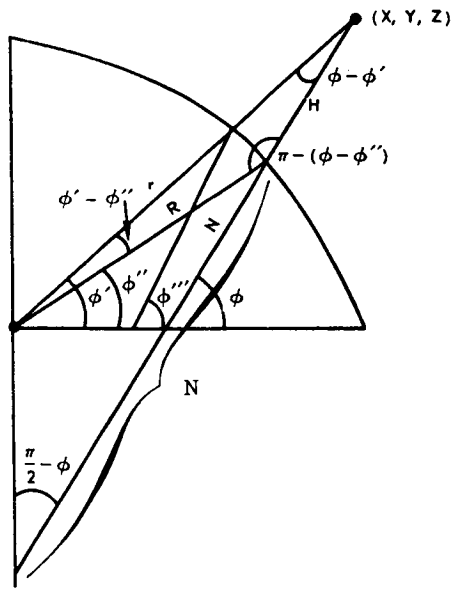
USING THIS VALUE AND THE ARCTANGENT SUBROUTINE, ϕ IS DETERMINED. THE QUANTITY ϕ IS BETWEEN $-\pi/2$ AND $+\pi/2$ WHEN STORED.

THE VALUE OF THE LONGITUDE, λ, IS FOUND USING EQUATION (3). λ IS BETWEEN 0 AND 2π WHEN STORED.

THE PROGRAM NOW TESTS WHETHER TO COMPUTE H BY DETERMINING IF THE SIGN OF 1, 4 IS NEGATIVE. IF THE SIGN IS POSITIVE THE PROGRAM RETURNS TO 4,4. IF NEGATIVE THE PROGRAM COMPUTES ϕ_i USING THE TRUE VALUE OF TAN ϕ. USING THE LATEST VALUE OF ϕ, THE PREVIOUSLY COMPUTED VALUE FOR 1/N AND EQUATION (2), H IS DETERMINED AND STORED AND THE PROGRAM RETURNS TO 4,4.

TRUE GEODETIC LATITUDE AND ALTITUDE OF SPACECRAFT

FORMULATION



BY LAW OF SINES-

$$(1)\quad R = \frac{r \sin(\varphi - \varphi')}{\sin(\varphi - \varphi'')} = \frac{N \cos\varphi}{\cos\varphi''}$$

$$(2)\quad \frac{N + H}{r} = \frac{\cos\varphi'}{\cos\varphi}$$

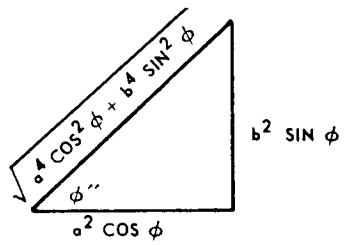
$$(3)\quad \text{where } N = \frac{a}{\sqrt{1 - e^2 \sin^2\varphi}}$$

$$(4)\quad \text{and } \tan\varphi'' = \frac{b^2}{a^2} \tan\varphi$$

$$(5)\quad \text{and } \tan\varphi''' = \frac{a^2}{b^2} \tan\varphi'$$

} from any book on Geodesy

THE FOLLOWING FIGURE DEPICTS THE TRIGONOMETRIC RELATIONSHIPS FOR THE ANGLE ϕ'' -



$$\sin(\varphi - \varphi'') = \sin\varphi \cos\varphi'' - \cos\varphi \sin\varphi''$$

$$= \sin\phi \frac{a^2 \cos\phi}{\sqrt{a^4 \cos^2\phi + b^4 \sin^2\phi}} - \frac{\cos\phi\, b^2 \sin\phi}{\sqrt{a^4 \cos^2\phi + b^4 \sin^2\phi}}$$

$$= \frac{(a^2 - b^2) \sin\phi \cos\phi}{\sqrt{a^4 \cos^2\phi + b^4 \sin^2\phi}}$$

$$r \sin(\varphi - \varphi') = r \sin\varphi \cos\varphi' - r \cos\varphi \sin\varphi' = \sqrt{x^2 + y^2} \sin\varphi - Z \cos\varphi.$$

HENCE, EQUATION (1) GIVES -

$$\frac{\sqrt{x^2 + y^2}\,\sin\phi - Z\cos\phi}{(a^2 - b^2)\sin\phi\cos\phi} = \frac{N}{a^2}$$

$$\sqrt{x^2 + y^2}\,\tan\varphi - Z = \left(\frac{a^2 - b^2}{a^2}\right) N\sin\varphi = e^2 N\sin\varphi$$

$$\boxed{\tan\varphi = \frac{e^2 N\sin\varphi + Z}{\sqrt{x^2 + y^2}}}$$

FROM THE DRAWING -

$$\cos\varphi = \frac{\sqrt{x^2 + y^2}}{N + H}$$

$$H = \frac{\sqrt{x^2 + y^2}}{\cos\varphi} - N$$

$$\boxed{H = (\sec\varphi)\,(\sqrt{x^2 + y^2}) - N}$$

USAGE

A) CALLING SEQUENCE -

```
ALPHA        TSX     A3MSCP,4
ALPHA + 1    PZE     (INPUT),,(OUTPUT)
ALPHA + 2    PZE     (BUFFER),,0
ALPHA + 3    ERROR RETURN
ALPHA + 4    NORMAL RETURN
```

INPUT = 4-LOCATION TABLE CONTAINING TIME, X,Y AND Z.

OUTPUT = 3-LOCATION TABLE CONTAINING λ , ϕ AND H.

BUFFER = 7-LOCATION PROGRAM WORKING TABLE.

IF THE 2ND WORD OF THE CALLING SEQUENCE IS -

1) PZE- λ, ϕ AND H ARE COMPUTED
2) MZE - HEIGHT COMPUTATION IS DELETED

IF THE 3RD WORD OF THE CALLING SEQUENCE IS -

1) PZE- INPUT TIME IS A(MINS.),,D(SECS.)
2) MZE - INPUT TIME IS FLOATING POINT SECONDS

NOTE - XR5 IS NOT SAVED BY THIS SUBROUTINE

B) CONSTANTS

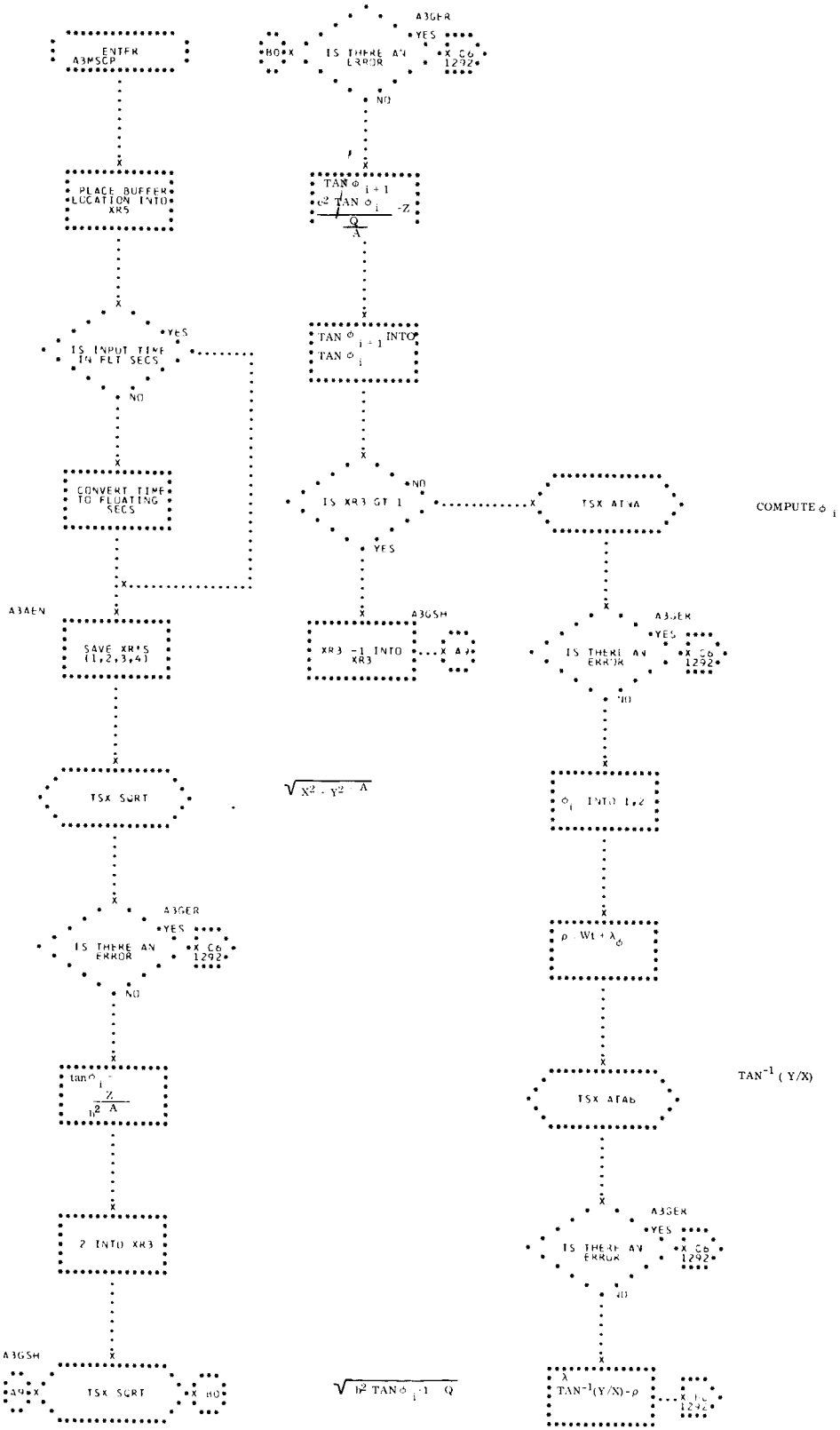
```
K00060  KKK2PI
K00000  K.INB2
K001.0  K.B2GU
KCH233  K.ECC2
KLAMDO  K.OMES
```

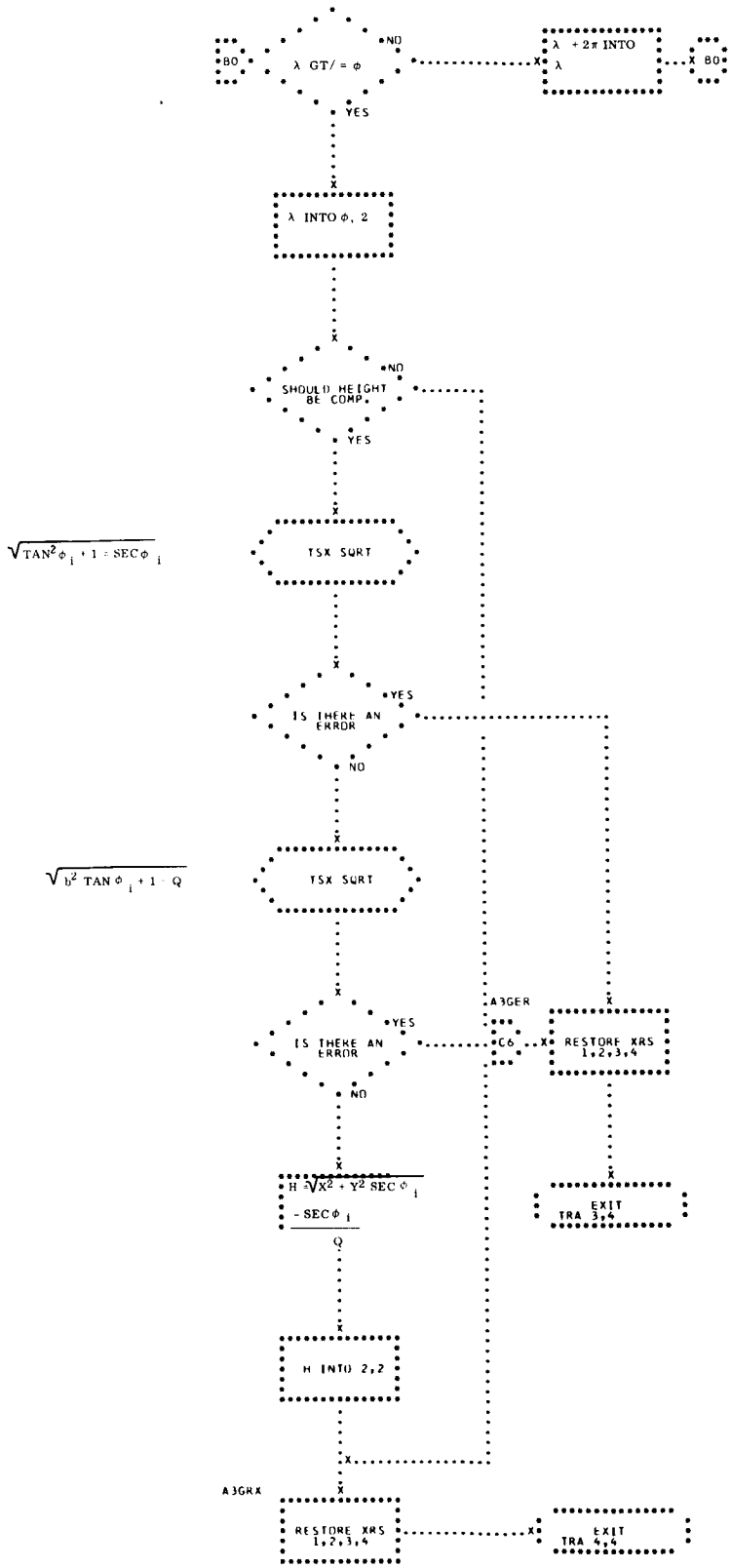
C) SUBROUTINES -

```
SQRT
ATNA
ATNAB
```

D) STORAGE REQUIRED - 105 LOCATIONS

E) TIMING - MAXIMUM = 1.847 MS


ENTER A3MSCP
PLACE BUFFER LOCATION INTO XR5
IS INPUT TIME IN FLT SECS
YES
NO
CONVERT TIME TO FLOATING SECS
A3AEN
SAVE XR'S (1,2,3,4)
TSX SQRT
A3GER
IS THERE AN ERROR
YES
NO
X C6 1292
2 INTO XR3
A3GSH
A9
TSX SQRT
B0
A3GER
IS THERE AN ERROR
YES
NO
X C6 1292
TAN φ i+1 INTO TAN φ i
IS XR3 GT 1
NO
YES
A3GSH
XR3 -1 INTO XR3
A9
TSX ATNA
COMPUTE φ i
A3GER
IS THERE AN ERROR
YES
NO
X C6 1292
φ i INTO 1,2
TSX ATAB
TAN-1 (Y/X)
A3GER
IS THERE AN ERROR
YES
NO
X C6 1292
TAN-1(Y/X)-ρ
1292


λ GT/= Φ
NO
YES
BO
λ + 2π INTO λ
X BO
λ INTO Φ, 2
SHOULD HEIGHT BE COMP.
NO
YES
$\sqrt{TAN^2 \Phi_i + 1} = SEC \Phi_i$
TSX SQRT
IS THERE AN ERROR
YES
NO
$\sqrt{b^2 TAN \Phi_i + 1} = Q$
TSX SQRT
A3GER
IS THERE AN ERROR
YES
NO
C6
RESTORE XRS 1,2,3,4
$H = \sqrt{X^2 + Y^2} SEC \Phi_i - SEC \Phi_i / Q$
EXIT TRA 3,4
H INTO 2,2
A3GRX
RESTORE XRS 1,2,3,4
EXIT TRA 4,4

88. N80TST - NOCPNI END TEST

N80TST TERMINATES INTEGRATION AS A FUNCTION OF TIME, PHASE ANGLE AND LONGITUDE. IT STORES THE LAST VECTOR IN THE DESIGNATED OUTPUT TABLE WHEN KDONTS IS SET. (SEE FLOWCHART.)

INPUT

A)	MCZRWX	MONITOR CONTROL WORD WHOSE DOCUMENT INDICATES TYPE OF INTEGRATION BEING PERFORMED.
B)	KOOOET	ENDING TIME OF OUTPUT TABLE.
C)	KOOOST	STARTING TIME OF OUTPUT TABLE.
D)	KOOOTI	TIME, t_i IN FIXED OR FLOATING POINT MINUTES OR SECONDS.
E)	KOOOTL	CONTAINS ADDRESS OF THE START OF THE OUTPUT TABLE.
F)	KOODTB	LENGTH OF TIME IN INTEGRATE BACKWARD.
G)	KOODTF	LENGTH OF TIME TO INTEGRATE FORWARD.
H)	KCAVEC	$\neq$0, REVOLUTION NUMBER IS BEING UPDATED. =0, ANOTHER TYPE OF INTEGRATION IS TAKING PLACE.
I)	KDONTS	EQUAL ZERO, TABLE IS STORED. EQUAL NON-ZERO, STORE LAST VECTOR.
J)	KFLOFX	EQUAL ZERO, FIXED POINT INTEGRATION. EQUAL NON-ZERO, FLOATING POINT INTEGRATION. IF NEGATIVE, HAD FLOATING POINT INPUT TO INTEGRATOR.
K)	KGMTIR	TIME TO INITIATE ROLL.
L)	KGMTRB	TIME OF BANK ANGLE REVERSAL.
M)	KRCRDP	= 0, USING CORWELL'S MIXED PRECISION

≠ 0, USING RUNGE-KUTTA SINGLE PRECISION.

N) KTTH.2 H/2 IN FLOATING POINT MINUTES OR SECONDS.

O) TEMNO COMMON LOCATION
+146 = 0, INDICATES FORWARD INTEGRATION
≠ 0, INDICATES BACKWARD INTEGRATION.

P) TNETIM TIME OF THE LAST VECTOR STORED IN TABLE.

Q) TNFUNK +22 X
R) TNFUNK + 48 Y $\vec{r}$ VECTOR
S) TNFUNK + 74 Z
T) TNFUNK + 24 XD
U) TNFUNK + 50 YD $\vec{V}$ VECTOR
V) TNFUNK + 76 ZD
W) TRPEAT = 0, TABLE STORED
≠ 0, STORE LAST VECTOR.

OUTPUT

THE LAST POINT IN THE TABLE IS STORED IN THE FOLLOWING FORMAT -

A) INDICATION OF MINUTES OR SECONDS. ZERO, FOR MINUTES, NON-ZERO, FOR SECONDS.
B) INTEGRATION OUTPUT INTERVAL
C) STARTING TIME
D) ENDING TIME
E) X
F) Y $\vec{r}$ VECTOR IN GEMINI UNITS
G) Z
H) XD
I) YD $\vec{v}$ VECTOR IN GEMINI UNITS.
J) ZD

METHOD

NON-APPLICABLE.

USAGE

A) CALLING SEQUENCE -

α	TSX	N8OTST
$\alpha + 1$		CONTINUE INTEGRATION
$\alpha + 2$		END INTEGRATION

B) STORAGE REQUIRED -

240 LOCATIONS

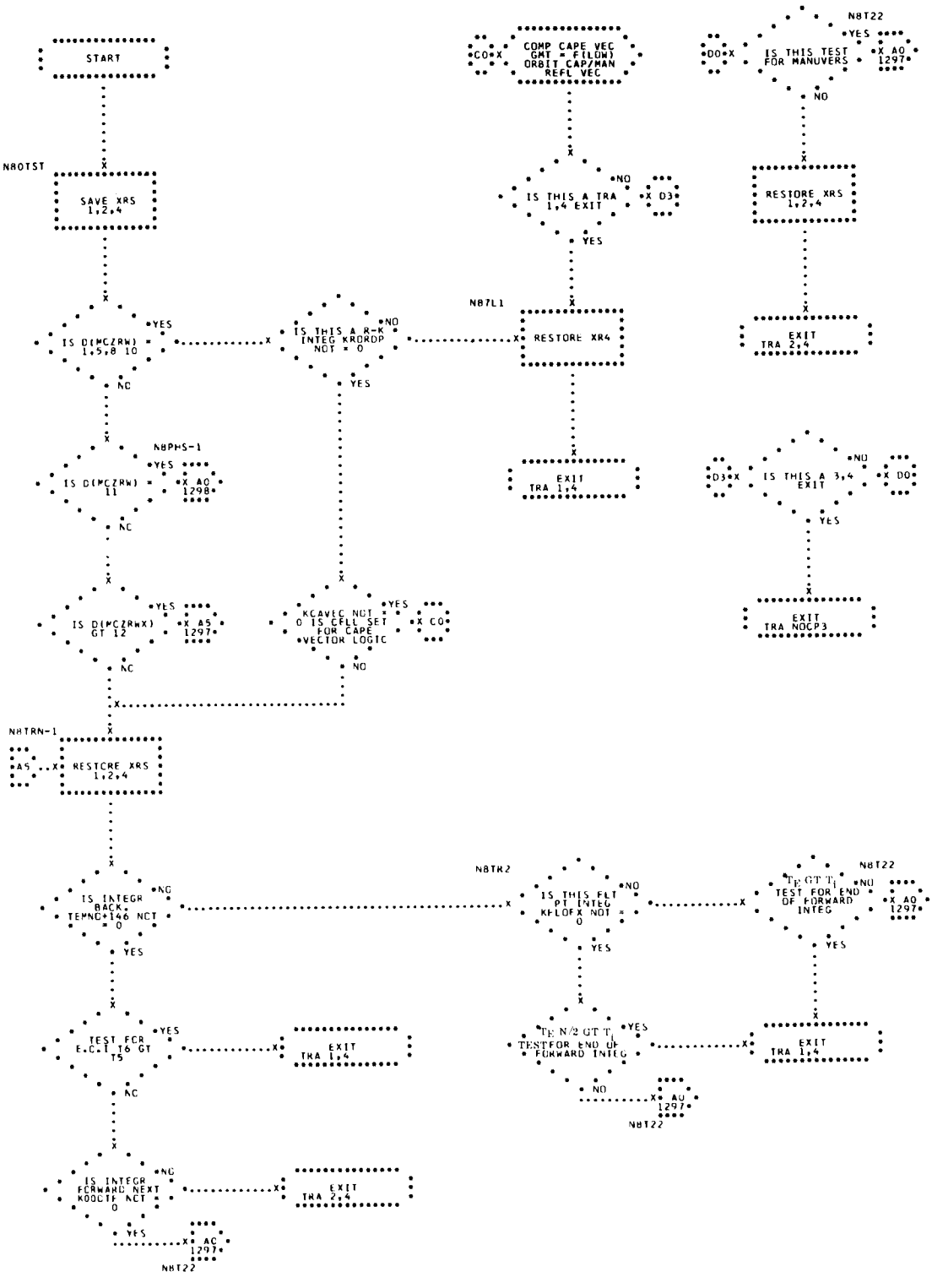
C) CONSTANTS -

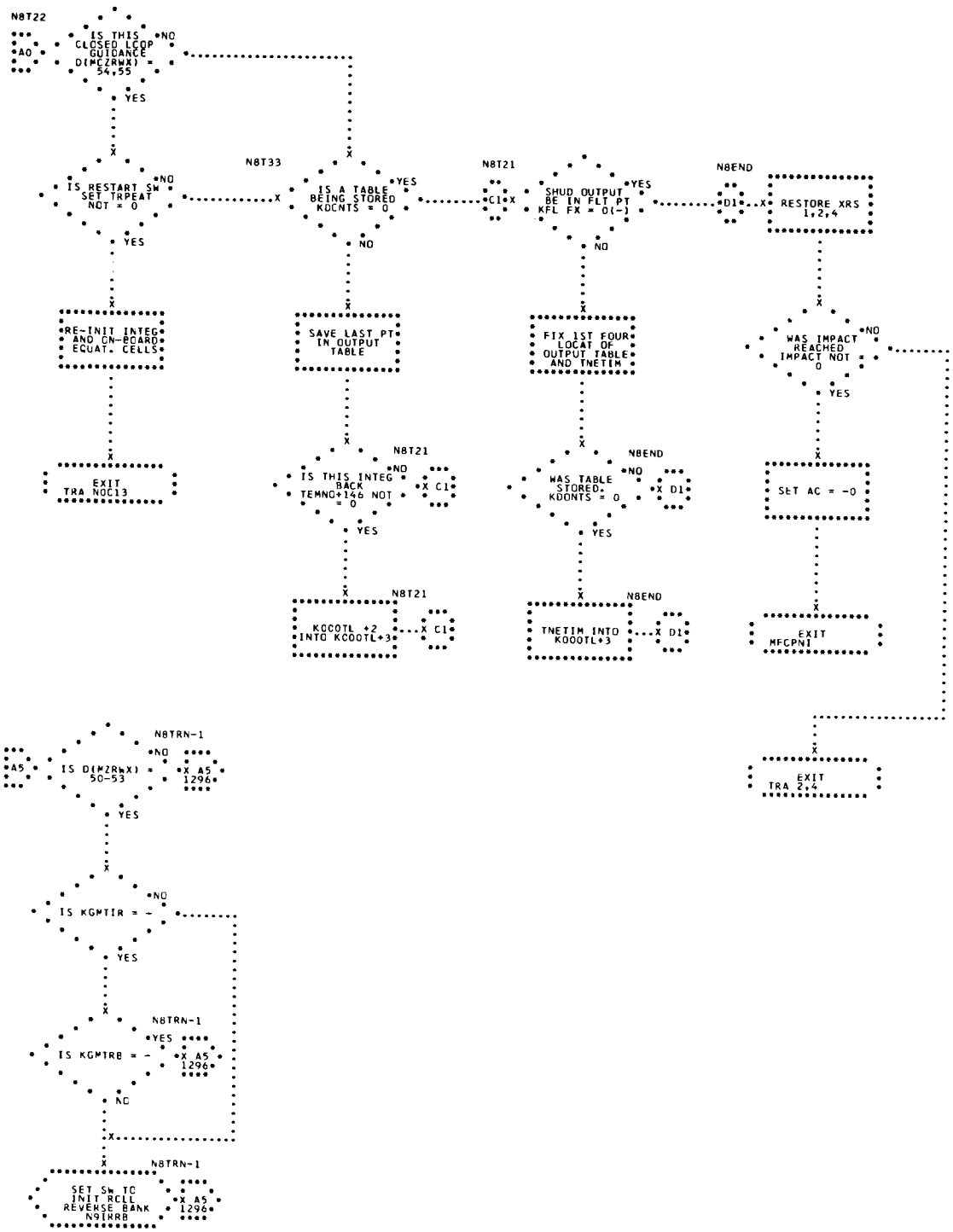
KMOOOO	DEC	-0
KOOOO1	DEC	1
KOOOO2	DEC	2
KOOOO4	DEC	4
KOOOO7	DEC	7
KOOOO8	DEC	8
KOOO10	DEC	10
KOOO12	DEC	12
KOOO50	DEC	50
KOOO53	DEC	53

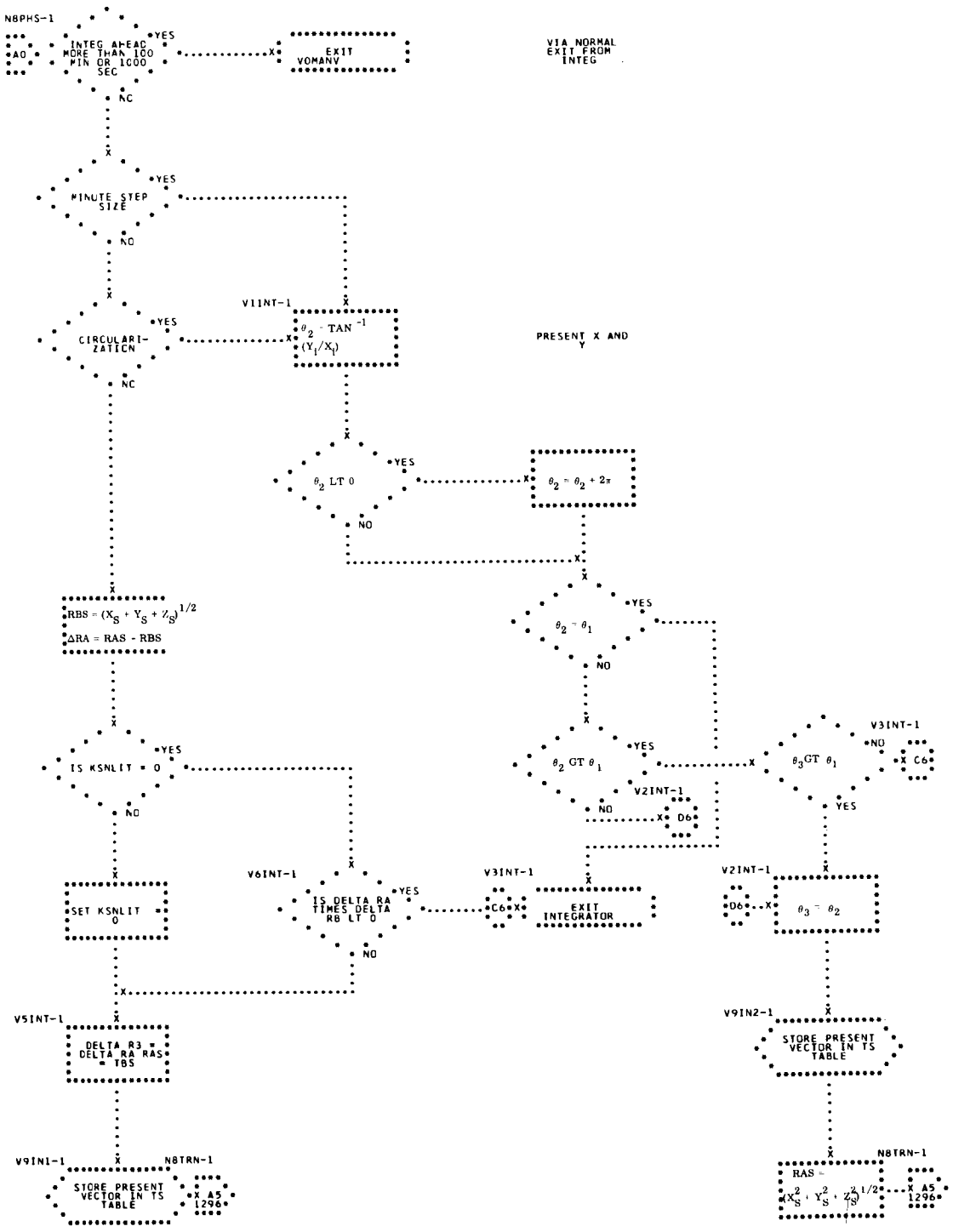
D) PROGRAMS -

N5CAPS

N9IRRB


START
N80TST
SAVE XRS 1,2,4
IS D(MCZRW) = 1,5,8 10
YES
NO
IS THIS A R-K INTEG KRORDP NOT = 0
NO
YES
N88PHS-1
IS D(MCZRW) = 11
YES
X AQ 1298
NO
IS D(MCZRW) GT 12
YES
X A5 1297
NO
KCAVEC NOT = 0 IS CELL SET FOR CAPE VECTOR LOGIC
YES
X CO
NO
N88TRN-1
A5
RESTORE XRS 1,2,4
IS INTEGR BACK. TEMP0+146 NOT = 0
NO
YES
TEST FOR E.C.I T6 GT T5
YES
EXIT TRA 1,4
NO
IS INTEGR FORWARD NEXT KOOCTF NOT = 0
NO
EXIT TRA 2,4
YES
X AQ 1297
N88T22
COMP CAPE VEC GMT = F(LOW) ORBIT CAP/MAN REFL VEC
CO
IS THIS A TRA 1,4 EXIT
NO
X D3
YES
N87L1
RESTORE XR4
EXIT TRA 1,4
N88T22
YES
IS THIS TEST FOR MANUVERS
DO
X AQ 1297
NO
RESTORE XRS 1,2,4
EXIT TRA 2,4
D3
IS THIS A 3,4 EXIT
NO
X DO
YES
EXIT TRA NOCP3
N88TR2
IS THIS FLT PT INTEG KFLOFX NOT = 0
NO
YES
TE N/2 GT Ti
YES
TESTFOR END OF FORWARD INTEG
NO
X AQ 1297
N88T22
N88T22
TE GT Ti
NO
TEST FOR END OF FORWARD INTEG
YES
X AQ 1297
EXIT TRA 1,4


N8T22
IS THIS CLOSED LOOP GUIDANCE D(MCZRWX) = 54,55
NO
YES
A0
IS RESTART SW SET TRPEAT NOT = 0
NO
YES
RE-INIT INTEG AND ON-BOARD EQUAT. CELLS
EXIT TRA NOC13
N8T33
IS A TABLE BEING STORED KOCNTS = 0
YES
NO
SAVE LAST PT IN OUTPUT TABLE
N8T21
IS THIS INTEG BACK TEMNO+146 NOT = 0
NO
YES
C1
KOCOTL +2 INTO KCOOTL+3
N8T21
C1
N8T21
SHUD OUTPUT BE IN FLT PT KFL FX = 0(-)
YES
NO
C1
FIX 1ST FOUR LOCAT OF OUTPUT TABLE AND TNETIM
N8END
WAS TABLE STORED. KOONTS = 0
NO
YES
D1
TNETIM INTO KOOOTL+3
N8END
D1
N8END
RESTORE XRS 1,2,4
WAS IMPACT REACHED IMPACT NOT = 0
NO
YES
SET AC = -0
EXIT MFCPNI
EXIT TRA 2,4
N8TRN-1
IS D(MZRWX) = 50-53
NO
YES
A5
1296
IS KGMTIR = -
NO
YES
N8TRN-1
IS KGMTRB = -
YES
NO
A5
1296
N8TRN-1
SET SW TO INIT RCLL REVERSE BANK N91RRB
A5
1296


N8PHS-1
A0
INTEG AHEAD MORE THAN 100 MIN OR 1000 SEC
YES
NO
EXIT VOMANV
VIA NORMAL EXIT FROM INTEG
MINUTE STEP SIZE
YES
NO
V1INT-1
CIRCULARI-ZATION
YES
NO
$\theta_2 = TAN^{-1}(Y_1/X_1)$
PRESENT X AND Y
θ_2 LT 0
YES
NO
$\theta_2 = \theta_2 + 2\pi$
$RBS = (X_S + Y_S + Z_S)^{1/2}$
$\Delta RA = RAS - RBS$
$\theta_2 = \theta_1$
YES
NO
IS KSNLIT = 0
YES
NO
θ_2 GT θ_1
YES
NO
V2INT-1
D6
θ_3GT θ_1
V3INT-1
NO
C6
YES
SET KSNLIT = 0
V6INT-1
IS DELTA RA TIMES DELTA RB LT 0
YES
NO
V3INT-1
C6
EXIT INTEGRATOR
V2INT-1
D6
$\theta_3 = \theta_2$
V5INT-1
DELTA R3 = DELTA RA RAS = TBS
V9IN2-1
STORE PRESENT VECTOR IN TS TABLE
V9IN1-1
STORE PRESENT VECTOR IN TS TABLE
N8TRN-1
A5
1296
N8TRN-1
RAS = $(X_S^2 + Y_S^2 + Z_S^2)^{1/2}$
A5
1296

89. N9IRRB - INITIATE ROLL, REVERSE BANK TESTS

PROGRAM N9IRRB TESTS ON TIME TO SET SWITCHES TO BEGIN REVERSING BANK OR INITIATING ROLL. IT COMPUTES CHANGES IN THE INTEGRATION STEP SIZE NEEDED FOR THESE PROCESSES. (SEE FLOWCHART)

INPUT

A) KGMTIR TIME TO INITIATE ROLL, IF SET MINUS, BYPASS TESTS.
B) KGMTRB TIME TO REVERSE BANK, IF SET MINUS, BYPASS TESTS.
C) KOOOTI PRESENT TIME.
D) KOOOOH PRESENT INTEGRATION STEP SIZE.
E) KINTVN STORE INTERVAL OF OUTPUT TABLE.
F) KBANK PRESENT BANK ANGLE
G) KORATE ROLL RATE
H) TEMNO DEFINED IN NOCPNI

OUTPUT

A) RBNOW SET NON-ZERO WHEN IT IS TIME TO REVERSE BANK ANGLE.
B) KRATE ROLL RATE FOR N3TRB.
C) FBANK FINAL BANK ANGLE REQUIRED.
D) TIN3RB SET NON-ZERO WHEN TIME TO INITIATE ROLL.

METHOD

KOOOTI AND KGMTIR ARE COMPARED. IF KGMTIR IS PAST KOOOTI, ROLL IS INITIATED IMMEDIATELY AND CONTINUES TO IMPACT. IF THIS IS NOT THE CASE, THE COMPARISONS CONTINUE UNTIL THEY ARE WITHIN AN INTEGRATION STEP SIZE. THE NEXT

INTEGRATION STEP SIZE IS COMPUTED AND TIN3RB IS SET NON-ZERO TO BEGIN THE INITIATION OF ROLL.

FOR BANK ANGLE REVERSAL, IF KGMTRB IS PAST KOOOTI, NO BANK ANGLE REVERSING OCCURS. WHEN THEY ARE WITHIN AN INTEGRATION STEP, THE NEXT TWO INTEGRATION STEPS ARE COMPUTED AND RBNOW IS SET NON-ZERO TO BEGIN BANK ANGLE REVERSAL IN N3TRB.

USAGE

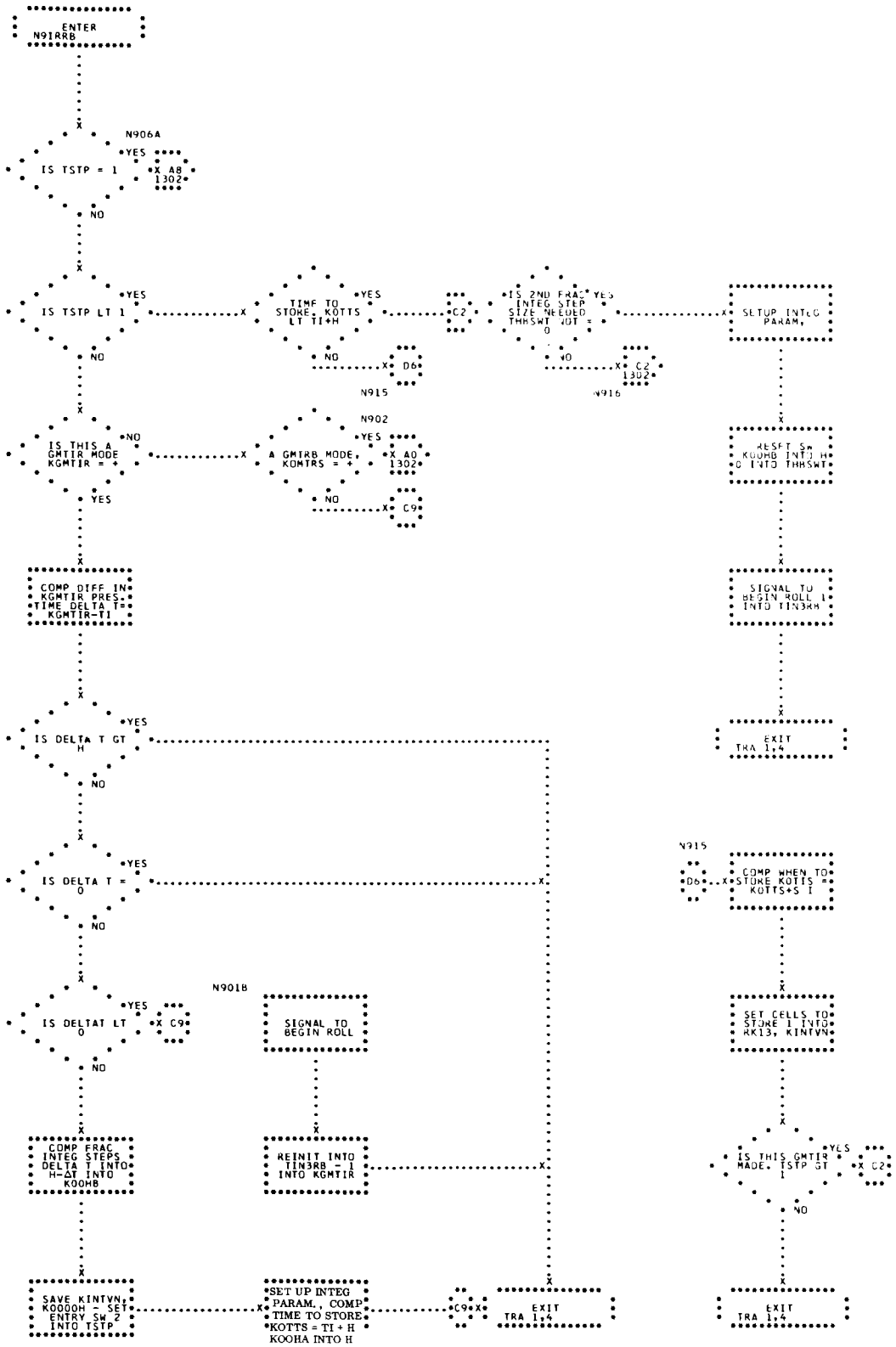
A) CALLING SEQUENCE -

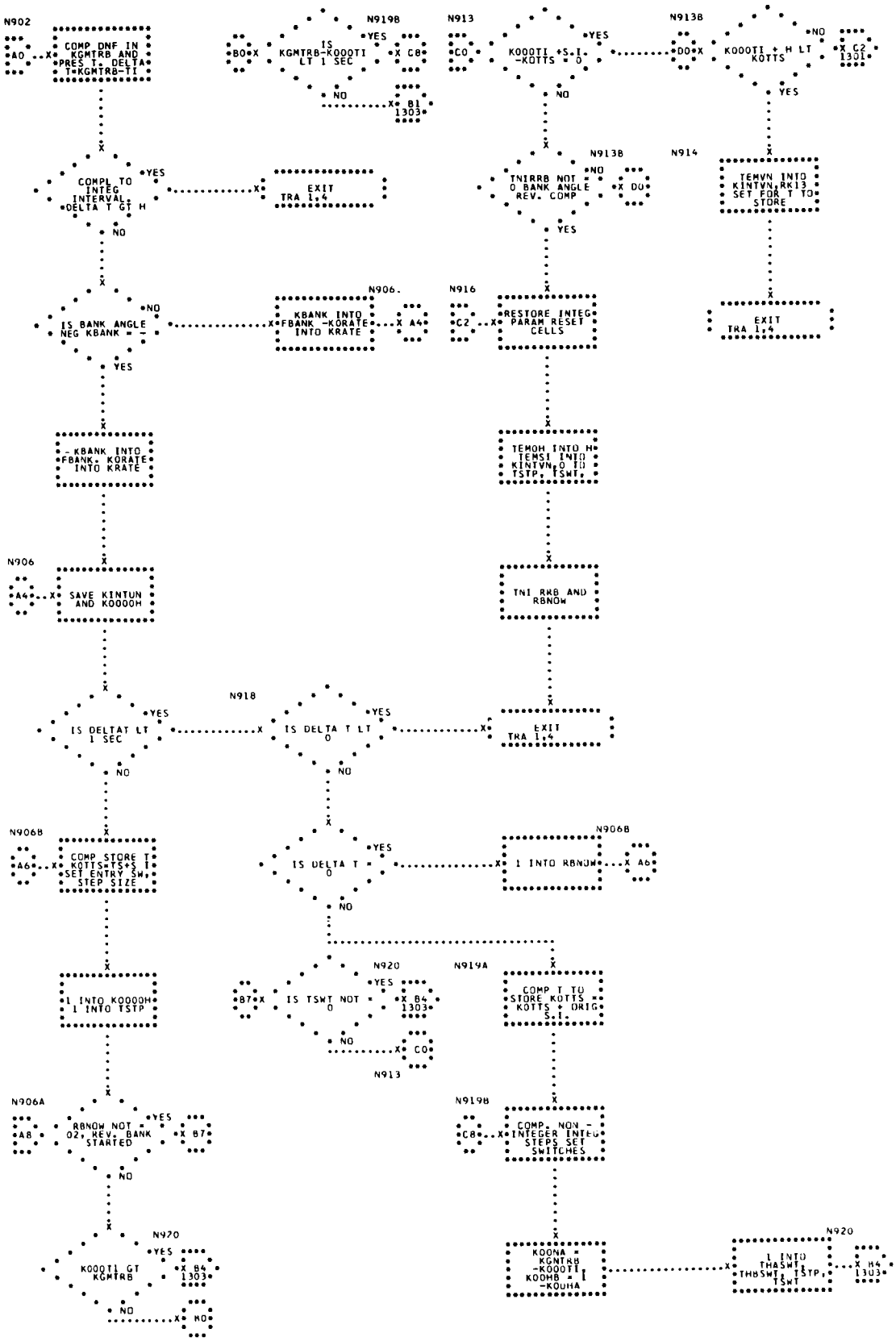
```
    TSX          N9IRRB,4
+1               NORMAL RETURN
```

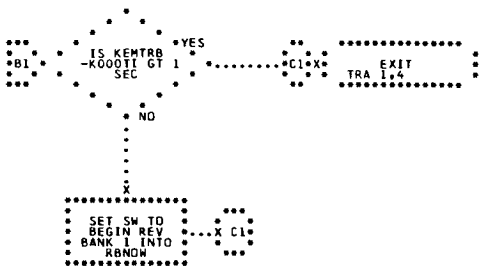
B) STORAGE REQUIRED - 249 LOCATIONS

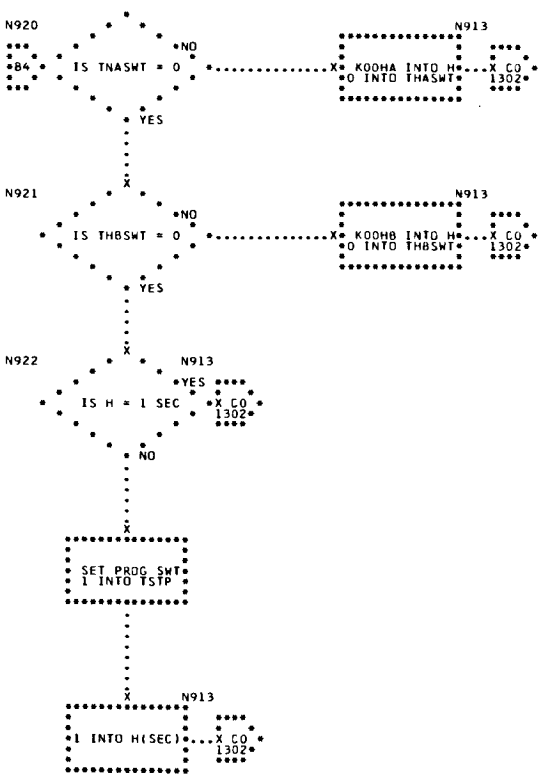
C) CONSTANTS -

```
KOO1.0  DEC 1.0
KOO2.0  DEC 2.0
KOO.25  DEC .25
```


ENTER N91RRB
N906A
IS TSTP = 1
YES
X A8 1302
NO
IS TSTP LT 1
YES
TIME TO STORE KOTTS LT TI+H
YES
C2
IS 2ND FRAC INTEG STEP SIZE NEEDED THHSWT NOT = 0
YES
SETUP INTEG PARAM,
NO
X D6
N915
NO
X C2 1302
N916
IS THIS A GMTIR MODE KGMTIR = +
NO
N902
A GMTRB MODE, KGMTRS = +
YES
X A0 1302
NO
X C9
YES
RESET S, KOOHB INTO H, 0 INTO THHSWT
COMP DIFF IN KGMTIR PRES, TIME DELTA T= KGMTIR-TI
SIGNAL TO BEGIN ROLL 1 INTO TIN3RB
IS DELTA T GT H
YES
EXIT TRA 1,4
NO
N915
D6
COMP WHEN TO STORE KOTTS = KOTTS+S I
IS DELTA T = 0
YES
NO
N9018
IS DELTAT LT 0
YES
X C9
SIGNAL TO BEGIN ROLL
SET CELLS TO STORE 1 INTO RK13, KINTVN
NO
COMP FRAC INTEG STEPS, DELTA T INTO H-ΔT INTO KOOHB
REINIT INTO TIN3RB - 1 INTO KGMTIR
IS THIS GMTIR MADE, TSTP GT 1
YES
X C2
NO
SAVE KINTVN, KOOOOH - SET ENTRY SW 2 INTO TSTP
SET UP INTEG PARAM., COMP TIME TO STORE KOTTS = TI + H
KOOHA INTO H
C9
EXIT TRA 1,4
EXIT TRA 1,4


N902
COMP DNF IN KGMTRB AND PRES T, DELTA T=KGMTRB-TI
COMPL TO INTEG INTERVAL, DELTA T GT H
YES
NO
EXIT TRA 1,4
IS BANK ANGLE NEG KBANK = -
NO
YES
KBANK INTO FBANK, KORATE INTO KRATE
N906
SAVE KINTUN AND KOOOOH
IS DELTAT LT 1 SEC
YES
NO
N906B
COMP STORE T KOTTS=TS+S.I SET ENTRY SW, STEP SIZE
1 INTO KOOOOH 1 INTO TSTP
N906A
RBNOW NOT = 02, REV. BANK STARTED
YES
NO
N920
KOOOTI GT KGMTRB
YES
NO
N906.
KBANK INTO FBANK -KORATE INTO KRATE
N9198
IS KGMTRB-KOOOTI LT 1 SEC
YES
NO
N918
IS DELTA T LT 0
YES
NO
IS DELTA T = 0
YES
NO
N920
IS TSWT NOT = 0
YES
NO
N913
N913
KOOOTI +S.I. -KOTTS = 0
YES
NO
N9138
TNIRRB NOT = 0 BANK ANGLE REV. COMP
NO
YES
N916
RESTORE INTEG PARAM RESET CELLS
TEMOH INTO H TEMSI INTO KINTVN, 0 TO TSTP, TSWT,
TNI RKB AND RBNOW
N9138
KOOOTI + H LT KOTTS
NO
YES
N914
TEMVN INTO KINTVN, RK13 SET FOR T TO STORE
EXIT TRA 1,4
N9068
1 INTO RBNOW
N919A
COMP T TO STORE KOTTS = KOTTS + ORIG S.I.
N919B
COMP. NON - INTEGER INTEG STEPS SET SWITCHES
KOONA = KGNTRB -KOOOTI KOOHB = I -KOUHA
N920
1 INTO THASWT, THBSWT, TSTP, TSWT


B1
IS KEMTRB -KOOOTI GT 1 SEC
YES
C1
EXIT TRA 1,4
NO
SET SW TO BEGIN REV BANK 1 INTO RBNOW
C1


N920
B4
IS TNASWT = 0
NO
N913
KOOHA INTO H
0 INTO THASWT
CO 1302
YES
N921
IS THBSWT = 0
NO
N913
KOOHB INTO H
0 INTO THBSWT
CO 1302
YES
N922
N913
IS H = 1 SEC
YES
CO 1302
NO
SET PROG SWT 1 INTO TSTP
N913
1 INTO H(SEC)
CO 1302

90. A3TSUP - A3MSCP INPUT TIME SET UP FOR NOCPNI

A3TSUP SETS UP THE INPUT TIME FOR A3MSCP IN THE PROPER UNITS. (SEE FLOWCHART.)

INPUT

THE TIME OF THE VECTOR TO BE USED FOR A3MSCP.

K000TU	=0,	TIME IS IN MINUTES
	≠0,	TIME IS IN SECONDS
KFLOFX	=0,	TIME IS IN FIXED POINT UNITS
	=0,	TIME IS IN FLOATING POINT UNITS
K060.0	DEC	60.0
K00060	DEC	60

OUTPUT

TIME IN THE PORPER UNITS FOR A3MSCP.

METHOD

NOT APPLICABLE.

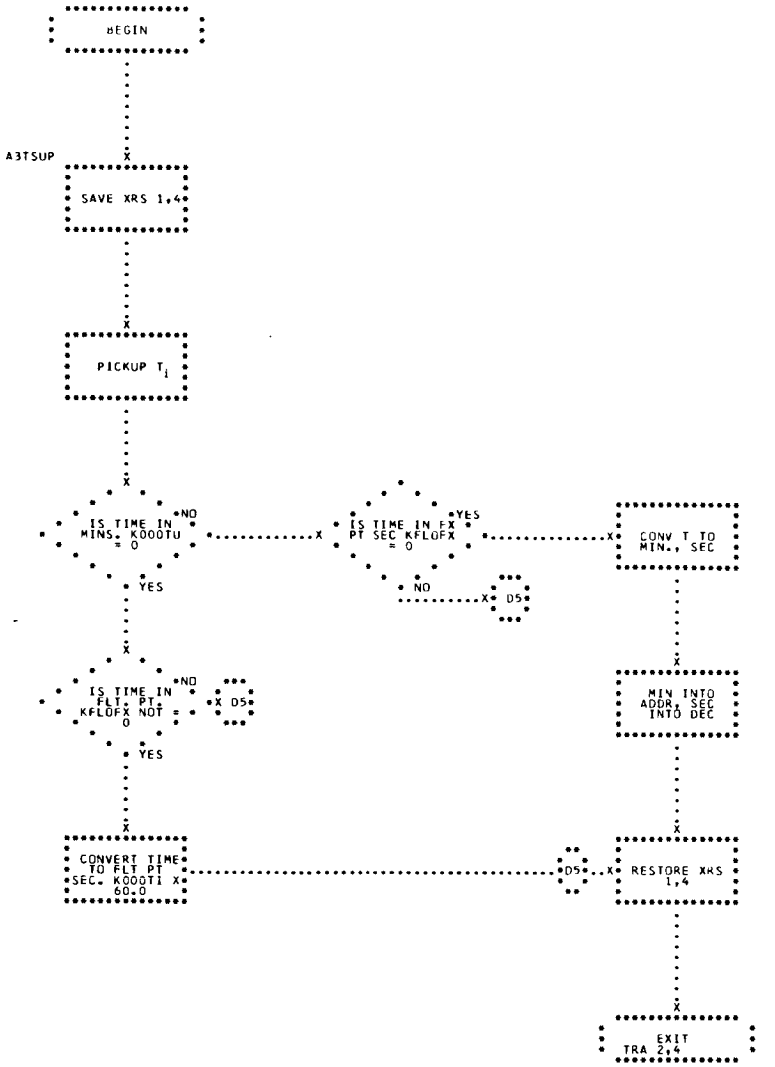
USAGE

A) CALLING SEQUENCE -

```
TSX     A3TSUP,4
PZE     INPUT TIME,,OUTPUT TIME
        NORMAL RETURN
```

B) STORAGE REQUIRED -

31 LOCATIONS


BEGIN
A3TSUP
SAVE XRS 1,4
PICKUP T_i
IS TIME IN MINS. K000TU = 0
NO
YES
IS TIME IN FX PT SEC KFLOFX = 0
YES
NO
D5
CONV T TO MIN., SEC
IS TIME IN FLT. PT. KFLOFX NOT = 0
NO
D5
YES
MIN INTO ADDR, SEC INTO DEC
CONVERT TIME TO FLT PT SEC. K000TI X 60.0
D5
RESTORE XRS 1,4
EXIT TRA 2,4

91. MYCOTO - CONFIRMATION OF MANEUVER CONTROL PROCESSOR

CONTROL LOGIC REQUIRED WHEN THE ORBITAL TRAJECTORY UNDERGOES REDEFINITION DUE TO SOME MANEUVER - INDUCED PERTURBATION IS PROVIDED BY MYCOTO. THROUGH USE OF THE INPUT MANEUVER CONDITIONS AND A VECTOR-GENERATING SUBROUTINE, AN ORBIT REFLECTING THE PROGRAM INPUT WILL BE ESTABLISHED. (SEE FLOWCHART.)

PROCESSOR OPTIONS AVAIABLE ARE -

A) THE PRESENT ORBIT-DEFINING VECTOR IS PERTURBED TO REFLECT THE INPUT MANEUVER PARAMETERS.

B) A NOMINAL MANEUVER IS REDEFINED TO COINCIDE WITH THE ACTUAL THRUST CONDITIONS.

C) A PREVIOUS EXECUTED MANEUVER IS ERASED AND THE ORBIT IS RESTORED TO PREMANEUVER CONDITIONS.

INPUT

A) QUEUES ARE RECEIVED FROM MFMN15, MFMN26

1. QUEUE WORD = (+) MANEUVER PARAMETER ENTRY
2. QUEUE WORD = (-) VELOCITY COMPONENT ENTRY

B) TABLES -

TICOTH CONTAINS MANEUVER PARAMETERS
TMCODV CONTAINS VECTOR HISTORY
TMCODW CONTAINS S/C WEIGHT HISTORY
TOSCCH S/C CHARACTERISTICS TABLE

OUTPUT

A) TABLES

TMCOTO CONTAINS MANEUVER - REFLECTING VECTOR
TMCODW S/C WEIGHT HISTORY IS UPDATED
TPCOTO CONTAINS ERROR PRINT INFORMATION

B) COMMUNICATION CELL -
MCNIAC ERROR CODE FROM INTEGRATOR

C) QUEUES PROVIDED FOR THE FOLLOWING -
MYROUR1 TO BEGIN BASIC CYCLE
MYMESS TO PRINT OUTPUT VECTOR
MYMESS TO PRINT ERROR MESSAGE

METHOD

AFTER UNQUEUING, MYCOTO TESTS FOR THE TYPE OF ENTRY. AN INCREMENT VELOCITY COMPONENT ENTRY IS INVALID, PROGRAM LOGIC HAS NOT AS YET BEEN DEFINED FOR THIS TYPE OF MANEUVER CONFIRMATION. SUCH AN ENTRY WILL CAUSE THE PRINTING OF AN ERROR MESSAGE.

HOWEVER UPON INDICATION OF VALID ENTRY, EXAMINATION OF OPTION DESIRABILITY IS BEGUN.

OPTION (A) - USING THE PRESENT ORBIT-DEFINING VECTOR AND INSERTED MANEUVER CONDITIONS AS INPUT TO GVMMRF (MANEUVER-REFLECTING VECTOR SUBROUTINE), THE REQUIRED VECTOR IS GENERATED.

OPTION (B) - USING A VECTOR PRIOR TO THE TIME OF THE NOMINAL S/C MANEUVER AND THE ACTUAL S/C THRUST CONDITIONS, GVMMRF WILL COMPUTE A NEW DEFINING VECTOR.

OPTION (C) - A VECTOR PRIOR TO THE EXECUTED MANEUVER IS OBTAINED FROM TMCODV AND IS USED AS THE OUTPUT VECTOR. GVMMRF LOGIC IS BYPASSED.

FOR OPTIONS (A) AND (B), A S/C POST-MANEUVER WEIGHT IS COMPUTED AS A FUNCTION OF THRUST DURATION. MYOUR1 IS QUEUED TO INITIATE THE BASIC CYCLE AND MYMESS IS QUEUED

TO PRINT ON-LINE THE OUTPUT VECTOR. EXIT IS THEN MADE TO MOPRIO.

USAGE

A) ENTRY FROM AND EXIT TO MOPRIO

B) COMMUNICATION CELLS -

MCNIAC

MCVPCT

C) TABLES -

TMCODV

TMCODW

TICOTH

TMCOTO

TOSCCH

TPCOTO

D) CONSTANTS -

K77777

KCH233

K00001

C) PARAMETERS -

A

MNCOTO

MNMESS

F) MACROS -

TRNON

TRNOF

QENBZ

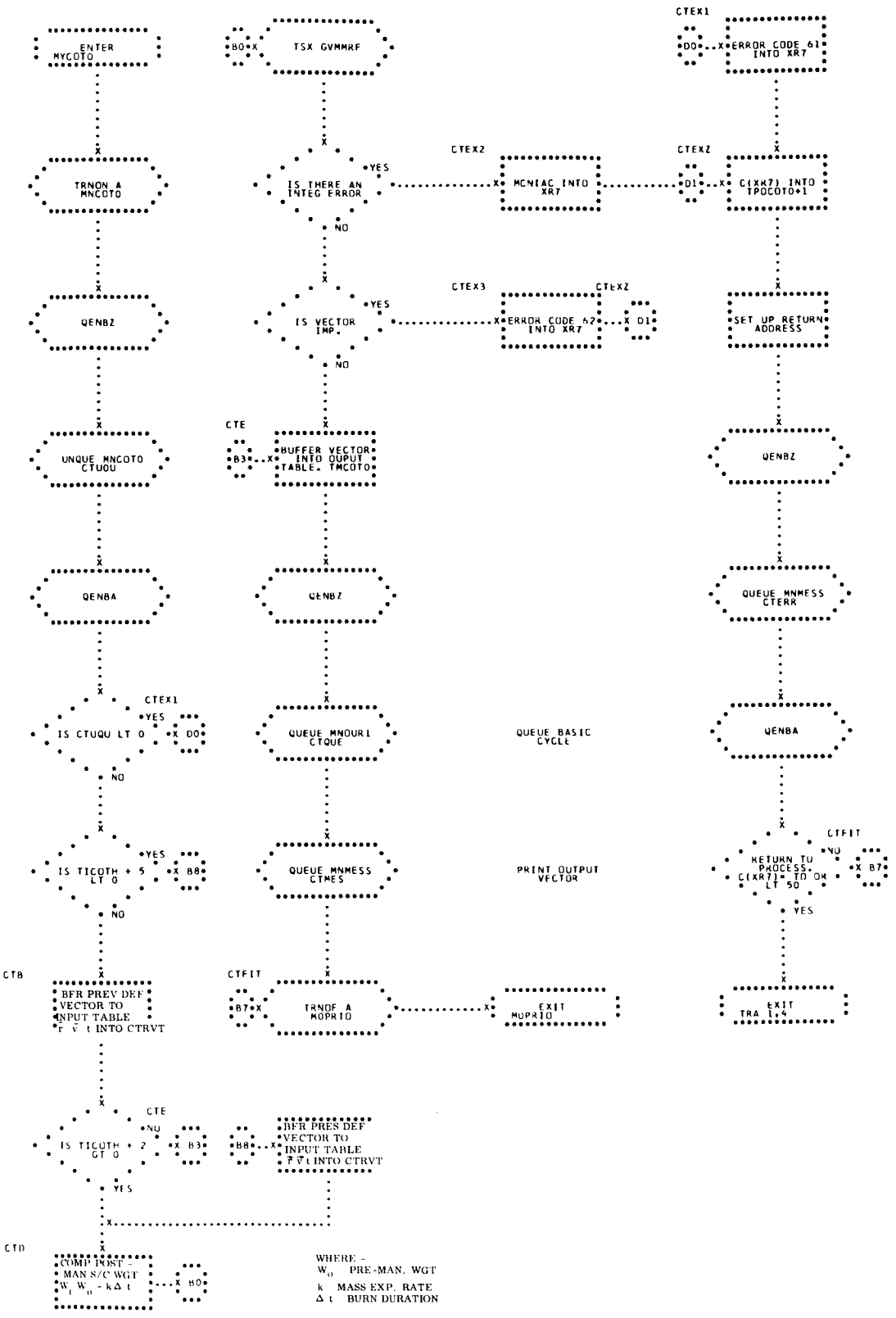
QENBA

UNQUE

QUEUE

G) SUBROUTINES -
GVMMRF

H) STORAGE REQUIRED - 85 LOCATIONS


ENTER MYCOTO
TRNON A MNCOTO
QENB2
UNQUE MNCOTO CTUOU
QENBA
IS CTUQU LT 0
YES
NO
CTEX1
D0
IS TICOTH + 5 LT 0
YES
NO
B8
CTB
BFR PREV DEF VECTOR TO INPUT TABLE
r v t INTO CTRVT
IS TICOTH + 2 GT 0
NO
YES
B3
CTE
B8
BFR PRES DEF VECTOR TO INPUT TABLE
r v t INTO CTRVT
CTD
COMP POST - MAN S/C WGT
W W0 - kΔ t
B0
B0 TSX GVMMRF
IS THERE AN INTEG ERROR
YES
NO
CTEX2
MCNIAC INTO XR7
IS VECTOR IMP.
YES
NO
CTEX3
ERROR CODE 62 INTO XR7
CTEXZ
D1
CTE
B3
BUFFER VECTOR INTO OUPUT TABLE. TMCOTO
QENBZ
QUEUE MNOUR1 CTQUE
QUEUE BASIC CYCLE
QUEUE MNMESS CTMES
PRINT OUTPUT VECTOR
CTFIT
B7
TRNOF A MOPRIO
EXIT MOPRIO
WHERE -
W0 PRE-MAN. WGT
k MASS EXP. RATE
Δ t BURN DURATION
CTEX1
D0
ERROR CODE 61 INTO XR7
CTEXZ
D1
C(XR7) INTO TPOCOTO+1
SET UP RETURN ADDRESS
QENBZ
QUEUE MNMESS CTERR
QENBA
CTFIT
NO
RETURN TO PROCESS. C(XR7) = TO OR LT 50
B7
YES
EXIT TRA 1,4

INDEX OF PROGRAMS

SYMBOL	TITLE	PAGE
A3GSCP	COMPUTATION OF LATITUDE, LONGITUDE AND ALTITUDE	433
A3MSCP	SUBROUTINE TO COMPUTE LATITUDE AND LONGITUDE	1285
A3TSUP	A3MSCP INPUT TIME SET UP FOR NOCPNI	1304
AQDATA	SUBROUTINE TO GENERATE ACQUISITION DATA	1125
BOSCER	EARTH RADIUS DETERMINATION	444
BUCFIT	SECOND DEGREE CURVE FITTING PROGRAM FOR BUMAIN	1213
BUDBCP	ORBIT PHASE BACK UP TO DIFFERENTIAL CORRECTION PROGRAM	1189
BUDCCP	REENTRY PHASE BACK UP TO DIFFERENTIAL CORRECTION PROGRAM	1182
BUEDIT	EDITING PROGRAM FOR BUMAIN	1217
BUFPPR	FOOTPRINT DESCRIPTION	1226
BUGSCP	SPACECRAFT POSITION AND HEIGHT DETERMINATION PROGRAM FOR BUMAIN	1222
BUKEPE	PROGRAM TO COMPUTE SIX CLASSICAL ELEMENTS	1234
BULTOI	LOCAL TO INERTIAL CONVERSION PROGRAM	1196
BUMAIN	COMPUTATION OF POSITION AND VELOCITY VECTORS AND ASSOCIATED INFORMATION	1202
CORKNI	RUNGE-KUTTA NUMERICAL INTEGRATION	467
C1DVGS	COMPUTE $-\Delta V_G$, AND $-\Delta V$	254
C2TAFH	COMPUTATION OF TIME AS A FUNCTION OF HEIGHT	607

INDEX OF PROGRAMS

SYMBOL	TITLE	PAGE
C3GSCP	COMPUTATION OF LATITUDE AND LONGITUDE FOR NUMERICAL INTEGRATION	520
C4COEF	COMPUTATION OF COEFFICIENTS OF DRAG AND LIFT	515
C4DRFT	COMPUTATION OF ACCELERATION DUE TO DRAG, LIFT, AND THRUST	498
C4PDEN	COMPUTATION OF ATMOSPHERIC DENSITY	510
C6HPEP	COMPUTATION OF GEOCENTRIC HEIGHT AND ACCELERATION DUE TO THE POTENTIAL OF THE EARTH	493
C6PTLI	LAUNCH PROGRAM FOR SIX POINT INTERPOLATION OF POSITION AND VELOCITY VECTORS	404
C6PTLU	LAUNCH PROGRAM FOR FOUR POINT LAGRANGIAN INTERPOLATION OF TIME	397
C6RETC	LAUNCH RETROFIRE PROGRAM	329
C6TDRF	RETROFIRE TIME CORRECTION	418
CABDAT	LAUNCH ABORT PHASE PROGRAM FOR UPDATING A GIVEN	612
CALTOI	LOCAL TO INERTIAL COORDINATE SUBROUTINE	116
CAMPSO	APPROXIMATE MIDPOINT SOLUTION SUBROUTINE	111
CANORM	SUBROUTINE TO FORM NORMAL EQUATIONS	121
CASARC	SHORT ARC PROGRAM	83
CASOLN	SUBROUTINE TO SOLVE NORMAL EQUATIONS	126
CBIPRE	IMPACT PREDICTOR FOR AVERAGE VECTOR COMPUTATION	322

INDEX OF PROGRAMS

SYMBOL	TITLE	PAGE
CCMAIN -	LAUNCH MAIN CALCULATIONS PROGRAM	220
CCSTRP -	LAUNCH STRIP CHARTS PROCESSOR	184
CDELTV -	LAUNCH PROGRAM TO INCREMENT THE VELOCITY VECTOR BY FIRING RETROROCKETS AS POSIGRADES	315
CEAPRV -	COMPUTE THE VECTORS AT APOGEE OR PERIGEE	272
CELEVP -	ELEVATE PERIGEE	260
CFCFAV -	LAUNCH COMPUTATIONS FOR HOLD PHASE	238
CGFTCI -	COMPUTATION OF IMPACT, GIVEN THE TIME TO FIRE RETROROCKETS	423
CHABRT -	LAUNCH PROCESSOR FOR HIGH ABORT	583
CIMPRK -	IMPACT PREDICTOR FOR NORMAL LAUNCH	528
CLABRT -	LOW ABORT PROCESSOR	572
CLSTOI -	LOCAL TO INERTIAL COORDINATE CONVERSION	130
COCOO1 -	LAUNCH COORDINATE CONVERSICN OF B-GE DATA	15
CPRVCM -	LAUNCH PRESENT VECTORS COMPUTATIONS	283
CRADES -	LAUNCH PROGRAM TO SELECT RECOVERY AREA	412
CSOGLS -	GENERAL LEAST SQUARES SMOOTHING	142
CS1GLS -	INITIALIZATION SUBROUTINE FOR CSOGLS	165
CS2GLS -	RECURSIVE SUBROUTINE FOR CSOGLS	169
CS3GLS -	SUBROUTINE TO GENERATE SOLUTIONS FOR CSOGLS	173
CSCOO2 -	LAUNCH COORDINATE CONVERSICN OF DATA	27

INDEX OF PROGRAMS

SYMBOL	TITLE	PAGE
CTSFT3	COMPUTATION OF TIME STEP FOR TYPE 3 INTEGRATION	392
CUPDAT	LAUNCH PROGRAM FOR UPDATING A GIVEN	538
CVAFPA	COMPUTATION OF RELATIVE VELOCITY AND RELATIVE FLIGHT PATH ANGLE AT A SPECIFIED HEIGHT	427
DODIFC	DIFFERENTIAL CORRECTION CONTROL PROGRAM	986
D2CSFE	CALCULATE STANDARD ERROR FIT	1026
D2SLNE	SOLVE NORMAL EQUATIONS	1029
D2SSMT	STATION SUM OF SQUARES OVER M	1034
D2SUNE	SET UP NORMAL EQUATIONS	1037
D2WEQT	WEIGHTING THE EQUATIONS OF CONDITION	1045
D3CDRM	CHOOSE DESIRED RADAR MESSAGE	1052
D3PTWT	WEIGHT TABLE LOOKUP	1059
D4CFRV	CALCULATE FUNCTIONS OF $\overline{r}$ AND $\overline{v}$	1062
EOBDIT	BERMUDA HIGH-SPEED EDIT	48
EOLED1	EDIT	800
GEMWGT	ORBIT WEIGHT SUBROUTINE	1255
GENERAL	CONTROL DURING LAUNCH/ORBIT	1
GINTAB	SUBROUTINE TO DETERMINE INTEGRATION TABLE	1251
GLINTS	VECTOR LINEAR INTERPOLATION SUBROUTINE	1258
GVMMRF	MANEUVER REFLECTING VECTORS	670
LIMPRK	CONTROL PROGRAM FOR CIMPRK	525
MABORT	MONITOR CONTROL FOR ABORT PHASE	542
MFBDAS	MONITOR SUFFIX TO CASARC AND CSOGLS	177

INDEX OF PROGRAMS

SYMBOL	TITLE	PAGE
MFBUDC	MONITOR SUFFIX TO BUDCCP AND BUOBCP	1241
MFCCGB	MONITOR SUFFIX TO COCO01	21
MFCCIP	MONITOR SUFFIX TO CSCO02	34
MFCCMN	SUFFIX TO CCMAIN	448
MFCPNI	MONITOR SUFFIX TO NUMERICAL INTEGRATION DERIVATION	875
MFDIFC	MONITOR SUFFIX TO DODIFC	1011
MFEOBD	MONITOR SUFFIX TO EOBDIT	69
MFEBAD	MONITOR BAD POINTS SUFFIX TO EOBDIT	65
MFLED1	MONITOR SUFFIX TO EOLED1	806
MFMPRK	MONITOR SUFFIX TO LIMPRK	535
MFORMC	MONITOR SUFFIX TO O5ORMC	1158
MFSTRP	MONITOR SUFFIX TO CCSTRP	194
MPBDAS	MONITOR PREFIX TO CSOGLS AND CASARC	73
MPBUDC	MONITOR PREFIX TO BUDCCP AND BUOBCP	1179
MPCCGB	MONITOR PREFIX TO COCO01	12
MPCCIP	MONITOR PREFIX TO CSCO02	24
MPCCMN	MONITOR PREFIX TO CCMAIN	198
MPCPNI	MONITOR PREFIX TO CONTROL PROGRAM NUMERICAL INTEGRATION	817
MPDIFC	MONITOR PREFIX TO DODIFC	965
MPDIFK	MONITOR PREFIX TO DODIFC	973
MPDIFM	MONITOR PREFIX TO DODIFC	976
MPEOBD	MONITOR PREFIX TO EOBDIT	37
MPLED1	MONITOR PREFIX TO EOLED1	793
MPLED2	MONITOR PREFIX TO EOLED1	797
MPMPRK	MONITOR PREFIX TO LIMPRK	523
MPORMC	MONITOR PREFIX TO O5ORMC	1143

INDEX OF PROGRAMS

SYMBOL	TITLE	PAGE
MPSTRP	MONITOR PREFIX TO CCSTRP	182
MSDELT	MONITOR SUBROUTINE TO DELETE AN ENTRY IN THE TMSTMS TABLE	644
MSINSR	SUBROUTINE TO INSERT RADAR OBSERVATIONS INTO DIFFERENTIAL CORRECTION	654
MSLCUC	MONITOR SUBROUTINE TO MOVE A RADAR MESSAGE BLOCK TO THE 32K CORE BUFFER MODULE	635
MSTINT	SUBROUTINE TO REMOVE RADAR OBSERVATIONS FROM DIFFERENTIAL CORRECTION	648
MSVTCA	GENERATION OF TMVTCA TABLE	631
MSWTMS	MONITOR SUBROUTINE TO COMPUTE A TMDATA BLOCK ADDRESS	641
MYAPE3	APOGEE-PERIGEE CONTROL PROCESSOR	725
MYCOTO	CONFIRMATION OF MANEUVER CONTROL PROCESSOR	1306
MYDIC9	DIFFERENTIAL CORRECTION MONITOR CONTROLLER	781
MYFPPR	MONITOR FOOTPRINT PREDICTOR	1244
MYGMT6	CALCULATE GMT	767
MYLIF8	REVOLUTION CAPABILITY MONITOR CONTROL PROCESSOR	777
MYMUV7	MANEUVER PARAMETER COMPUTATION CONTROL PROCESSOR	772
MYNAV5	ORBIT NAVIGATION VECTOR FOR DCS	762
MYOUR1	ORBIT BASIC CYCLE CONTROL PROCESSOR	657
MYQNEA	MONITOR QUEUE NEXT EMERGENCY AREA PROCESSOR	789

INDEX OF PROGRAMS

SYMBOL	TITLE	PAGE
MYQPRA	MONITOR QUEUE NEXT PRIMARY AREA PROCESSOR	784
MYREN2	REENTRY CONTROL PROGRAM	676
MYTABU	MANEUVER TABLE CONTROL PROCESSOR	1262
MYTTF4	TIME-TO-FIRE CONTROL PROCESSOR	744
N1SUFT	SET UP FUNCTION TABLE	879
N2EXCR	EXTRAPOLATION AND CORRECTION EQUATION	884
N3SDCL	SECOND DERIVATIVE CALCULATION CONTROL	891
N3TRB	COMPUTATION OF INSTANTANEOUS BANK ANGLE FOR TIME TO REVERSE BANK	895
N4COEF	COEFFICIENT OF DRAG CALCULATION	919
N4DRAG	CAPSULE DRAG FORMULATION	898
N4PDEN	COMPUTATION OF ATMOSPHERIC DENSITY	912
N5CAPS	NUMERICAL INTEGRATION CAPE VECTOR STORING	939
N5OTBI	NUMERICAL INTEGRATION OUTPUT TABLE INITIALIZATION	925
N5OTBL	NUMERICAL INTEGRATION OUTPUT TABLE STORING	929
N6OPEP	EARTH'S EFFECT ON SPACECRAFT CALCULATION	946
N7PCAL	ON-BOARD COMPUTER EQUATIONS	957
N7VARS	VARIABLE STEP INTEGRATION EQUATIONS	950
N8OTST	NOCPNI END TEST	1293
N9IRRB	INITIATE ROLL, REVERSE BANK TESTS	1299
NKCAPI	NUMERICAL INTEGRATION CAPE VECTOR INITIALIZATION	935

INDEX OF PROGRAMS

SYMBOL	TITLE	PAGE
O5ORMC -	ORBIT, REENTRY HIGH-SPEED OUTPUT COMPUTATION PROCESSOR	1146
R1NDCP -	NUMERICAL DIFFERENTIAL CORRECTION	1066
R2AINT -	NUMERICAL INTEGRATION INPUT TABLES TO BE USED FOR GENERATION OF PERTURBED TABLES FOR NUMERICAL PARTIAL DERIVATIVES	1073
R5RARF -	TIME TO FIRE PROGRAM	1081
R6ESTF -	ESTIMATED TIME-TO-FIRE PROCESSOR	1164
R7EOAM -	FIRST GUESS FOR TIME TO FIRE RETROS	1169
RHBRVG -	PROGRAM TO COMPUTE THE RELATIVE VELOCITY VECTOR MAGNITUDE AND RELATIVE FLIGHT PATH ANGLE	1246
ROCKNR -	THRUST PROGRAM	860
RUCIVI -	PLATFORM TO INERTIAL COORDINATES CONVERSION	719
RUPDAT -	FOR UPDATING THE POST RETROFIRE TRAJECTORY	712
RVECHT -	COMPUTE HEIGHT VECTOR	1160
VOMANV -	PROPULSIONS PANEL COMPUTATIONS	1271