

N71-34178

R-120U

LIFTING ROBOT-SPACE SHUTTLE
PRELIMINARY MISSION PROFILE
OPTIMIZATION PROGRAM

PART II - PROGRAM LISTINGS

May 29, 1971

Prepared for

Marshall Space Flight Center
National Aeronautics and Space Administration
Huntsville, Alabama

Generated under
Contract NAS8-25713

Prepared by

Robert G. Gottlieb
Project Engineer

Approved by

Allan Dushman
Manager, Systems Analysis
and Design Department

DYNAMICS RESEARCH CORPORATION
Systems Division
60 Concord Street
Wilmington, Massachusetts 01887

SUMMARY

This volume contains a listing for LIFTING ROBOT plus a tabulation of the program routines with a brief description of each routine. The subroutines occur in alphabetical order in both the listing and the table, with the exceptions that MAIN and AINIT, which are listed first, and the subroutines in the integration package - DESOLV, DPIR, RTMRK - which are listed last.

TABLE OF SUBROUTINE DESCRIPTIONS

NAME	DESCRIPTION
MAIN	Controls overall program execution and presets input values
AINIT	Reads case data, performs initialization
ACNTRØ	Calculates χ_p and χ_y out of the atmosphere
ACSTØP	Controls library functions for payoff and terminal constraints
ADDR	Specifies adjoint equations for APPG package
ADER1	Gets time dependent portion of derivatives in atmosphere
ADER	Gets total derivatives in atmosphere
ADER11	Gets exo-atmospheric time dependent portion of derivatives
ADERI	Gets total exo-atmospheric derivatives
AEØSR	End of step routine; saves 4 state variables and time for use in backward solution
AFØRND	Completes forward trajectory; evaluates terminal conditions and initializes lambdas
AFØRUN	Controls entire logic per forward trajectory
AGEØ	Calculates oblate gravitational properties
AKALT	Control routine for 10 km and 14 km altitudes
AMULG	Linear Lagrangian interpolation routine
ANEWCH	Calculates changes in control parameters and control variables and controls convergence
APPG	Computes radii of apogee and perigee and associated partial derivatives
APRTN	Performs output editing, prints data, and writes record tape
AQMAX	Trigger routine for maximum dynamic pressure
ASIMP	Control logic for impact trajectory integration
ATCST	Controls forward integration through coasting period
ATHREV	Thrust event control routine
ATILT	Controls atmospheric χ_p and χ_y logic
ATTRAC	Thrust, weight, and propellant flowrate tables
AWDEV	Weight drop event control routine; also controls end of chi-freeze
AXPRT	Normal print control routine
BADLX	Gets partials with respect to lift-off weight (WZERØ) and CHIDØT and AZ
BAKRN	Initialization and overall control of adjoint differential equation solution

TABLE OF SUBROUTINE DESCRIPTIONS (CONT'D)

NAME	DESCRIPTION
BDR1I	Computes time dependent portion of adjoint derivatives
BDRI	Computes total adjoint derivatives
BKEND	Terminates adjoint run and adjusts influence coefficients
BMULG7	Linear Lagrangian interpolation routine for four variables; used in adjoint run
BØPTBL	Edits and prints 11 tables of output for purposes of publication
BSAVEG	Saves impulse response functions and performs Simpson integration on products of impulse response functions
BTCST	Controls backward integration through coasting period
BTHREV	Backward thrust event control routine
BWDEV	Backward weight drop event control routine
CACN	Aerodynamic data
DERIMP	Gets derivatives for impact trajectory
DRJUMP	Integration interrupt routine
FØRKM	Fourth order Runge-Kutta differential equation solver for APPG package
GETAM	Gets jacobian for impact adjoints
GETLAM	Gets lambdas for coasting equations
GETS	Gets S terms for coasting equations
GRØUND	Trigger routine for impact point
IMPPT	Instantaneous impact penalty routine
MATINV	Performs matrix inversion
PRA63	Patrick reference atmosphere - 1963 version
PRB63	Same as PRA63 but computes partials of density speed of sound and pressure for the backward integration
SPLINE	Spline fit routine
SYSDER	Specifies orbital equations of motion for APPG package
TARGET	Gets target position and velocity components in rendezvous option
DESØLV }	Integration package
DPIR	
RTMRK }	

MAIN

LIFTING-ROBOT LISTINGS

OT*TPF\$.MAIN

C

MAIN

COMMON/DELX/DELX(10),XLAMBX(3,10)

COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),

1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,

2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM

COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL

COMMON/ALONGO/ALONGO

COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,

1 TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),

2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),

3 HNOM(15),HMAX(15)

COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,

1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GK,GTH,

2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,

3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,

4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QUOT,CAX,CNP,CN,

5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,

6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,

7 BHMAX(15),BHMN,BSTEP(15)

COMMON/COVL/COVL(10)

COMMON/AZMTH/STH1,CTH1,XASM,YASM,ZASM,XJX(3),XJV(3),NALTW

COMMON/RESTX/RESTX(8)

COMMON/PREDP/PREDP

COMMON/ASYMPT/ASM(3)

COMMON/COAST/NCOST1,NCOST2,CX(6,2)

COMMON/FSCY/NFSCY,FSCYD,TFSCY

COMMON/CYT/NCYT,ICY1,TCY2,TCY3,TCY4,CYTM

COMMON/IMPACT/XIMP(10),IMP

COMMON/CONVK/CONVK(17,3)

COMMON/CHIROL/TCHIR,FAZ,CHRDOT

COMMON/BLAMB/YLBI(15,10),YLBW(15,10),GAPHI(10),PROD(2,10),WAH

COMMON/BAKE/WISS(10,10),YLF(7,10)

COMMON/IMP/GAPT(200,10),GSPT(200,10),AHTB(200),SITB(200),IS(4),

1 DELTS(4),KJTB

COMMON/DRHO/DRHO(9),E1,WIPPB,WGNU

COMMON/TABLK/MCAN,MCAF,MCON1,MCON2,MCON3,MCON4,MBS4,MBS7,MTT1,MTT2

1,MTT3,MTT4

COMMON/CADATA/PNM(44),CAN(44),CNN(44),PFM(25),CAF(25),CNF(25)

COMMON/CSTOP/CSTVAR(8)

COMMON/TABS/TBL(151),XBL(151),YBL(151),ZBL(151),XMBL(151)

1,WBL(151),UBL(151),VBL(151)

COMMON/NTABLE/NTABLE

COMMON/INPUT/AZ,LAT,DRAG,CHIDT,W01,WPM,CYT,FSCY,RTASC,DECL,DELVO,

1 DELVP,VELTGT,GAMTGT,RTGT,INCTGT,LATTGT,BTATGT,IFAD,IAA,IPHIT,

2 PSIRST(6),HEADER(60),FLBS(15),WDOT(15),CWDOT(15),WDLBS(15),

3 WTJET(15),LAST

COMMON/QY/QY,CFR(15)

COMMON/CONTRO/TTBL(201),CPTBL(201),CYTBL(201)

COMMON/YLI/YLINT(7,10),PHITE(10),PSIREQ(10),KCDPHI(10),PSTR(10)

COMMON/ARDC/PAT(15)

COMMON/GRAV/GT0,G11,G22,G23,G33

COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,

1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,

2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT

COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)

COMMON/AUTO/KDB(30),SAVCP(30),XLAMB(30,10),DP2,KDT(30)

MAIN

LIFTING-ROBOT LISTINGS

```

57      COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,wlBT(17)
58      COMMON/PERF/ILAS1,IPR,PRK1,PRK2,PRK3,PRK4,PRTOT,FGR,FPR,WPMX
59      COMMON/JTHR/JTHR(15),TBLTM(4)
60      COMMON/COUNT/ICOUNT,ICONJ,LONGPR
61      COMMON/LPRINT/LPRINT
62      C
63      C      *****
64      C      THE ORIGINAL ROBOT PROGRAM WAS FORMULATED AND PROGRAMMED
65      C      DURING THE FALL OF 1967 BY ROBERT G. GOTTLIEB
66      C      OF APPLIED ANALYSIS, INCORPORATED
67      C      3322 SO. MEM. PKWY., HUNTSVILLE,ALA.
68      C      *****
69      C
70      C      *****
71      C      ATMOSPHERIC OPTIMAZATION WITH THROTTLING WAS FORMULATED AND
72      C      PROGRAMMED INTO THE ROBOT DECK DURING 1970-1971 BY
73      C      ROBERT G. GOTTLIEB OF DYNAMICS RESEARCH CORP
74      C      60 CONCORD ST., WILMINGTON,MASS. 01887
75      C      *****
76      C
77      JORB=0
78      JUMP=0
79      IWDCHI=0
80      KIND=0
81      KWTA=0
82      NMAX=0
83      IPCNST=0
84      LPRINT=0
85      NTABLE=0
86      NVRST=0
87      IPR=0
88      LAST=0
89      KRDER=0
90      KINDB=0
91      KRDERB=0
92      IMP=0
93      NCYT=0
94      NFSCY=0
95      NCOST1=0
96      NCOST2=0
97      IFAB=0
98      IAA=0
99      IPHIT=0
100     DO 1 I=1,15
101     KDB(I)=0
102     KDT(I)=0
103     KDB(I+15)=0
104     KDT(I+15)=0
105     TAUT(I)=0.
106     TAUW(I)=0.
107     NOWD(I)=0
108     STEP(I)=8.
109     BSTEP(I)=16.
110     PRINT(I)=10.
111     HMAX(I)=100000.
112     BHMAX(I)=200000.
113     AE(I)=0.

```

MAIN

LIFTING-ROBOT LISTINGS

```
114      S(I)=0.
115      WIBT(I)=1.
116      WDOT(I)=0
117      CWDOT(I)=0
118      WDLBS(I)=0
119      WTJET(I)=0
120      FLBS(I)=0
121      JTHR(I)=0
122      DO 1 J=1,4
123      TNE(J,1)=0.
124      1 CONTINUE
125      DO 12 I=1,4
126      12 NP(I)=0
127      DO 13 I=1,10
128      13 KCDPHI(I)=0
129      DO 14 I=1,5
130      KCDRES(I)=0
131      14 NOEVNT(I)=0
132      KCDRES(6)=0
133      STEP(1)=1.
134      PI=3.14159265
135      RAD=180./PI
136      JUMP=1
137      IWDCHI=1
138      KIND=3
139      KWT=2
140      KRDER=3
141      KINDB=3
142      KRDERB=3
143      AZ=90.
144      LAT=28.531855
145      DRAG=1 697.2
146      XJEXT=1.
147      CASE=1.
148      UP2=.5
149      QY=.8
150      CHIDI=.1
151      DELVG=0.
152      DELVP=0.
153      WPM=0.
154      ALONG0=80.5649528
155      EU=1.E-05
156      BEU=2.E-05
157      AYL=2.E-03
158      BYL=4.E-05
159      HMN=.25
160      BHMN=.50
161      CMUE=3.986032E14
162      OMEGA=7.2921158E-05
163      CJ=1.62345E-03
164      H=-5.75E-06
165      DJ=7.875E-06
166      FLAT=1./298.3
167      RE=6378165.
168      GZERO=9.80665
169      V1V(7)=0.
170      NOWD(1)=3
```

MAIN

LIFTING-ROBOT LISTINGS

```

171      PRK1=0.
172      PRK2=0.
173      PRK3=0.
174      PRK4=0.
175      PRTOT=0.
176      FGR=0.
177      FPR=0.
178      2 CALL AINIT
179      3 CALL AFORUN
180      CALL AFORND
181      IF(NMAX.LE.0)GO TO 4
182      15 CALL BAKRN
183      WRITE(6,31)
184      31 FORMAT(/1X,11HCOV LAMBDA$)
185      WRITE(6,30)T,(COVL(I),I=1,7)
186      30 FORMAT(/,3X,8E14.6)
187      CALL BADLX
188      IF(KDB(18).EQ.0)GO TO 35
189      C DOING FORWARD RUNS FOR AZIMUTH PARTIALS
190      C BECAUSE OF OVERLAY STRUCTURE PROBLEM
191      CALL AFORUN
192      CALL AFORND
193      CALL BADLX
194      35 CALL ANEWCH
195      NMAX=NMAX-1
196      GO TO 3
197      4 ICOUNT=ICOUNT+1
198      NMAX=1
199      IF(ICOUNT.LT.ICONT) GO TO 15
200      IF(IMP.NE.0)CALL ASIMP
201      DATA ENDPRT/6HENDPRT/
202      AT=ENDPRT
203      WRITE( 9)AT,AZ,LAT,DRAG,XJEXT,CASE,DP2,QY,CHIDT,W01,DELVG,DELVP,
204      1 WPM,TCHIR,CHRDOT,FAZ,ALONGO, TCY1,TCY2,TCY3,TCY4,CYT,FSCYD,
205      2 EU,BEU,TFSCY,AYL,BYL,HMN,BHMN,CMUE,OMEGA,CJ,H,DJ,FLAT,RE,GZERO,
206      3 RTASC,DECL,VELTGT,GAMTGT,RTGT,INCTGT,LATTGT,BTATGT
207      IF(NTABLE.EQ.1)CALL BOPTBL
208      CHIDT=CHIDOT*RAD
209      W01=WZERO/.45359237
210      WIBT(17)=1.
211      DO 5 I=1,NOPAR
212      5 WIBT(17)=AMAX1(W1BT(I),WIBT(17))
213      DO 6 I=1,NOPAR
214      6 WIBT(I)=WIBT(I)/WIBT(17)
215      IF(LAST.EQ.1)GO TO 2
216      STOP
217      END

```

5 .L.O.

6

AINIT

LIFTING-ROBOT LISTINGS

I,S TPF\$.AINIT
PUR 0018-05/17-11:50

AINIT

LIFTING-ROBOT LISTINGS

OT*TPF\$.AINIT

```

1  SUBROUTINE AINIT
2  COMMON/XCON/UMF2,OMEGA2,FUMF1,UMFC
3  COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
4  COMMON/FSCY/NFSCY,FSCYD,TFSCY
5  COMMON/FAB/FAB,IFAB,MFAB,FABAL(44),FARTB(44)
6  COMMON/TARPO/TARPO(6),TAR(6),TAULN(2)
7  COMMON/COAST/NCOST1,NCOST2,CX(6,2)
8  COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
9  COMMON/AUTO/KDB(30),SAVCP(30),XLAMB(30,10),DP2,KDT(30)
10 COMMON/AZMTH/STH1,CTH1,XASM,YASM,ZASM,XJX(3),XJV(3),NALTW
11 COMMON/CYT/NCYT,TCY1,TCY2,TCY3,TCY4,CYTM
12 COMMON/CADATA/PNM(44),CAN(44),CNN(44),PFM(25),CAF(25),CNF(25)
13 COMMON/CHIROL/TCHIR,FAZ,CHRDOT
14 COMMON/ALONG0/ALONG0
15 COMMON/IMPACT/XIMP(10),IMP
16 COMMON/NTABLE/NTABLE
17 COMMON/PERF/ILAS1,IPR,PRK1,PRK2,PRK3,PRK4,PRTOT,FGR,FPR,WPMX
18 COMMON/YLI/YLINT(7,10),PHITE(10),PSIREQ(10),KCDPHI(10),PSTR(10)
19 COMMON/ARDC/PAT(15)
20 COMMON/CONST/RAD,P1,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
21 COMMON/GRAV/GTO,G11,G22,G23,G33
22 COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
23 1 TCHFRZ,DTZ,TQ,IL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
24 2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
25 3 HNOM(15),HMAX(15)
26 COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
27 1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPH12,GR,GTH,
28 2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
29 3 VIV(8),DVARS(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
30 4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
31 5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
32 6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
33 7 BHMAX(15),BHMN,BSTEP(15)
34 COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
35 1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
36 2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
37 COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
38 COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAK(15),
39 1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
40 2NTRG6,TV6,NTRG7,IV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
41 COMMON/CONTRO/TTBL(201),CPTBL(201),CYTBL(201)
42 COMMON/QY/QY,CFR(15)
43 COMMON/INPUT/AZ,LAT,DRAG,CHIDT,W01,WPM,CYT,FSCY,RTASC,DECL,DELVG,
44 1 DELVP,VELTGT,GAMTGT,RTGT,INCTGT,LATTGT,BTATGT,IFAD,IAA,IPH1T,
45 2 PSIRST(6),HEADER(60),FLBS(15),WDOT(15),CWDOT(15),WULBS(15),
46 3 WTJET(15),LAST
47 COMMON/JTHR/JTHR(15),TBLTM(4)
48 COMMON/COUNT/ICOUNT,ICONT,LONGPR
49 COMMON/LPRINT/LPRINT
50 COMMON/AEDK/IAEDK
51 COMMON/PENLTY/NCONPT,DXS(6)
52 COMMON/IMDATA/NO1,LLA(10),LLO(10),S2LO(10),S2LA(10),RH12(10)
53 DIMENSION BCD(60)
54 DIMENSION LATWTH(10),LONWTH(10),ROTA(10),TH1(10),PH1(10)
55 DIMENSION GLIMG(15),FUELG(15)
56 REAL LATWTH,LONWTH

```

AINIT

LIFTING-ROBOT LISTINGS

```

57      REAL LLA,LLO
58      DATA BLANK/6H      /
59      C * REWIND BOPTBL TAPE
60      REWIND 9
61      ICOUNT=0
62      DO 10 I=1,10
63      10 HEAD(I)=BLANK
64      NUM1=0
65      LSB=0
66      NALTW=0
67      KPRT=1
68      KPAGE=0
69      MERR=0
70      MERR=10
71      ICOUNT=0
72      ICONT=1
73      LONGPR=1
74      IAEDK=0
75      C READ INPUT DATA
76      C THE FOLLOWING TWO CARDS REINITIALUZE CASE BY TRUNCATING IT
77      ICASE=CASE
78      CASE=ICASE
79      NMAX=INMAX
80      IAEDK=JAEDK
81      LONGPR=JONGPR
82      ICONT=JCONT
83      IF (LONGPR.EQ.0) LONGPR=1
84      IF (ICONT.EQ.0) ICONT=1
85      REAL INCTGT, LATIGT,LAT
86      NAMELIST/INPUT/HEAD,TZERO,TDRAG,TLIFT,TTILT,TCHFRZ,DIZ,FLBS,
87      1 TNE,WDOT,CWDOT,WDLBS,WTJET,AE,S,TAUT,TAUW,NOWD,NOEVNT,PRINT,
88      2 STEP,BSTEP,AZ,LAT,DRAG,XJEXT,CASE,DP2,QY,CHIDT,W01,DELVG,DELVP,
89      3 WPM,TCHIR,CHRDOT,FAZ,ALONGO,          TCY1,TCY2,TCY3,TCY4,CYT,FSCY,
90      4 TFSCY,
91      4 EU,BEU,AYL,BYL,HMN,BHMN,CMUE,OMEGA,CJ,H,DJ,FLAT,RE,GZERO,JORB,
92      5 JUMP,IWDCHI,KIND,KWTA,NMAX,NTABLE,NVRST,IPR,LAST,KRDER,
93      6 KINDB,KRDERB,IMP,NCYT,NFSCY,JTHR,KCDPHI,PSIREQ,KCDRES,
94      7 PSIRST,KDB,KDT,WIBT,NBGCT,NENDCT,NP,TTBL,CPTBL,CYTB,L,VIV,RTASC,
95      8 DECL,VELTGT,GAMTGT,RTGT,INCTGT,LATTGT,BTATGT,NCOST1,NCOST2,
96      9 IFAB,IAA,IPHIT,LPRINT,          ICONT,IAEDK,LONGPR,TH1,PH1,ROTA,
97      * LATWTH,LONWTH,NOI,IPCNST,MSWCH,GLIMG,FUELG
98      N=0
99      READ(5,INPUT)
100     7 IF(N.EQ.0) GO TO 2
101     WRITE(6,3)
102     3 FORMAT(1X,12HMAVRIK ERROR,/,1X,29H**WARNING**WARNING**WARNING**,
103     A/1X,30H**YOUR DATA DOES NOT COMPUTE**)
104     2 AA=AZ/RAD
105     INMAX=NMAX
106     JCONT=ICONT
107     JONGPR=LONGPR
108     JAEDK=IAEDK
109     THET1=(90.-LAT)/RAD
110     DRAG1=DRAG*4.44822162
111     CHIDOT=CHIDT/RAD
112     WZERO=W01*.45359237
113     WPMX=WPM*.45359237

```

AINIT

LIFTING-ROBOT LISTINGS

```

114      CYTM=CYT/RAD
115      FSCYD=FSCY/RAD
116      RA=RTASC/RAD
117      DC=DECL/RAD
118      C
119      TEMP=GAMTGT/RAD
120      TEMP1=INCTGT/RAD
121      TARP0(1)=0.
122      TARP0(2)=0.
123      TARP0(3)=RTGT
124      TARP0(4)=VELTGT*COS(TEMP)*COS(TEMP1)
125      TARP0(5)=VELTGT*COS(TEMP)*SIN(TEMP1)
126      TARP0(6)=VELTGT*SIN(TEMP)
127      IF(IAA.EQ.1)
128      1AA=ASIN(COS(TEMP1)/SIN(THET1))
129      TAULN(1)=SIN(LATTGT/RAD)
130      IF(IPHIT.EQ.1)
131      1TAULN(1)=COS(TEMP1)*COS(THET1)/(SIN(TEMP1)*SIN(THET1))
132      TAULN(2)=BTATGT*RTGT/(RAD*VELTGT)
133      TAULN(1)=ASIN(TAULN(1))/OMEGA
134      TEMP1=TAULN(2)
135      TAULN(2)=TAULN(2)-TAULN(1)
136      TEMP=TAULN(1)*OMEGA*RAD
137      TEMP2=AA*RAD
138      WRITE(6,53)TAULN(1),TAULN(2),TEMP,TEMP1,TEMP2
139      53 FORMAT(4X,7E15.8)
140      DO 8 I=1,15
141      F(I)=FLBS(I)*4.44822162
142      XMD(I)=WDOT(I)*.45359237
143      FUEL(I)=FUELG(I)*.45359237
144      GLIM(I)=GLIMG(I)*GZERO
145      CFR(I)=CWDOT(I)*.45359237
146      WD(I)=WDLBS(I)*.45359237
147      8 WJET(I)=WTJET(I)*.45359237
148      EL=EU/2.**((KRDER+2)
149      BEL=BEU/2.**((KRDERB+2)
150      C*****CHECK THAT NP(I) IS ODD
151      DO 9 I=1,4
152      IF(NP(I).EQ.0)GO TO 9
153      IF(NP(I).EQ.1)NP(I)=3
154      IF(NP(I).GE.50)NP(I)=49
155      J=(NP(I)/2)*2
156      IF(J.EQ.NP(I))NP(I)=NP(I)+1
157      9 CONTINUE
158      DO 11 I=1,15
159      IF(F(I).EQ.0.)AE(I)=0.
160      11 CONTINUE
161      C*****
162      C      CALCULATE NUMBER OF PARAMETERS
163      IF(JUMP.EQ.1)GO TO 111
164      J=JUMP-1
165      DO 110 I=1,J
166      KDT(I)=0
167      110 KDB(I)=0
168      111 NOPAR=0
169      DO 12 I=1,20
170      IF(KDB(I).EQ.0)GO TO 12

```

AINIT

LIFTING-ROBOT LISTINGS

```

171      NOPAR=NOPAR+1
172  C***  CHECK FOR ERROR IN KDB AND KDT SPECIFICATIONS
173      J=KDT(I)+I
174      IF(J.EQ.1.OR.KDB(J).EQ.0)GO TO 12
175      WRITE(6,3)
176      NMAX=0
177      12 CONTINUE
178  C*****
179      IF(KDB(19).EQ.1)NFSCY=1
180  C      CALCULATE NUMBER OF REGULAR SETS
181      NCONPT=0
182      NOSS=0
183      DO 13 I=1,7
184          IF(KCDPHI(I).EQ.26)NCONPT=I
185          IF(KCDPHI(I).EQ.0)GO TO 13
186          NOSS=NOSS+1
187      13 CONTINUE
188  C*****
189  C      CALCULATE NUMBER OF RESTART CONSTRAINTS
190      TR=1.E19
191      NCNRS=0
192      IF(NVRST.EQ.0)GO TO 15
193      DO 14 I=1,6
194          IF(KCDRES(I).EQ.0)GO TO 14
195          NCNRS=NCNRS+1
196      14 CONTINUE
197      15 CONTINUE
198          IF((NOSS+NCNRS).LE.10)GO TO 17
199          WRITE(6,16)
200      16  FORMAT(1X,23H    100 MANY CONSTRAINTS)
201          STOP
202      17 CONTINUE
203  C***  CALCULATE NO OF THRUST EVENTS
204      NVNT=0
205      DO 18 I=1,5
206          IF(NOEVRT(I).EQ.0)GO TO 18
207          NVNT=NVNT+NOEVRT(I)
208          JA=I
209      18 CONTINUE
210      ILAST=NVNT-NOEVRT(JA)+1
211  C***  CALCULATE NO OF WEIGHT EVENTS
212      NNVNT=0
213      DO 19 I=1,15
214          IF(NOWD(I).EQ.0)GO TO 19
215          NNVNT=NNVNT+1
216      19 CONTINUE
217  C***  CALCULATE IMPACT DATA
218      DO 22 I=1,NOI
219          LLA(I)=TH1(I)/RAD
220          LLO(I)=PH1(I)/RAD
221          TEMP=LATWTH(I)/LONWTH(I)
222          TEMP=TEMP*TEMP-1.
223          SINA=SIN(ROTA(I)/RAD)
224          TEMP1=RAD/LATWTH(I)
225          TEMP1=TEMP1*TEMP1
226          S2LO(I)=TEMP1*(1.+TEMP*(1.-SINA*SINA))
227          S2LA(I)=TEMP1*(1.+TEMP*SINA*SINA)

```

AINIT

LIFTING-ROBOT LISTINGS

```

228      22 RH12(I)=TEMP1*TEMP*SIN(2.*ROTA/RAD)
229 C*** CALCULATE PERFORMANCE RESERVE CONSTANTS
230      IF(IPR.EQ.0)GO TO 20
231      PRK1=1.-EXP(-DELVG*XMD(IPR)/F(IPR))
232      PRK2=1.-EXP(-DELVP*XMD(IPR)/F(IPR))
233      PRK3=(1.-PRK1)*PRK2
234      PRK4=PRK1+PRK3
235 C*** CHECK KDT SETUP FOR IPR
236      DO 191 J=ILAST,NVNT
237      IF(KDB(J).EQ.0)GO TO 191
238      KDT(J)=IPR-J
239      IF(KDT(J).GT.0)GO TO 191
240      KDT(J)=0
241      KDB(J)=0
242      NOPAR=NOPAR-1
243      191 CONTINUE
244      20 CONTINUE
245 C
246      PAT(1)=0.
247      CALL PRA63(PAT(1),IER)
248      PSL=PAT(2)*10000.
249      STH1=SIN(THET1)
250      CTH1=COS(THET1)
251      STHL=STH1
252      CTHL=CTH1
253      IF(JORB.EQ.0)GO TO 21
254 C / SPHERICAL EARTH
255      R=RE
256      UMF=1.
257      STH=STH1
258      CTH=CTH1
259      GO TO 30
260 C . OBLATE EARTH
261      21 UMF=1.-FLAT
262      UMFC=FLAT*(2.-FLAT)/(RE*RE*UMF*UMF)
263      TEMP=P1/2.-ATAN2(UMF*UMF*CTH1,STH1)
264      STH=SIN(TEMP)
265      CTH=COS(TEMP)
266      R =UMF*RE/SQRT((UMF*STH)**2+CTH**2)
267      30 WRZ=R*STH*OMEGA
268      DPHIZ=OMEGA*DTZ
269      SDPHI=SIN(DPHIZ)
270      CDPHI=COS(DPHIZ)
271      IF(JUMP.NE.1)GO TO 40
272      B0(1,1)=CDPHI
273      B0(1,2)=SDPHI*STH
274      B0(2,1)=0.
275      B0(2,2)=CTH
276      B0(3,1)=-SDPHI
277      B0(3,2)=CDPHI*STH
278      XJV(1)=WRZ*B0(1,1)
279      XJV(2)=0.
280      XJV(3)=WRZ*B0(3,1)
281      XJX(1)=R*B0(1,2)
282      XJX(2)=R*B0(2,2)
283      XJX(3)=R*B0(3,2)
284      GO TO 50

```

AINIT

LIFTING-ROBOT LISTINGS

```

285      40 IF (VIV(7).LT.1.)GO TO 50
286 C*** GET INITIAL STATE AS FUNCTIONS OF (V,GAM,R,AZ,LAT,NODE)
287      AZZ=(180.-VIV(4))/RAD
288      SAZZ=SIN(AZZ)
289      CAZZ=COS(AZZ)
290      THZZ=(90.-VIV(5))/RAD
291      STHZZ=SIN(THZZ)
292      CTHZZ=COS(THZZ)
293      WBAR=ATAN2(CTHZZ*SAAZ,CAZZ)
294      IF (WBAR.LT.0.)WBAR=2.*PI-WBAR
295      PHIZ=VIV(6)/RAD-WBAR
296      SPHIZ=SIN(PHIZ)
297      CPHIZ=COS(PHIZ)
298      VU=SIN(VIV(2)/RAD)*VIV(1)
299      VH=COS(VIV(2)/RAD)*VIV(1)
300      XUX(1)=VIV(3)*SPHIZ*STHZZ
301      XUX(2)=VIV(3)*CTHZZ
302      XUX(3)=VIV(3)*CPHIZ*STHZZ
303      XUV(1)=VH*(SAAZ*CPHIZ+CAZZ*CTHZZ*SPHIZ)+VU*SPHIZ*STHZZ
304      XUV(2)=VU*CTHZZ-VH*CAZZ*STHZZ
305      XUV(3)=-VH*(SAAZ*SPHIZ-CAZZ*CPHIZ*CTHZZ)+VU*CPHIZ*STHZZ
306      50 ALT1=10000.020
307      ALT2=14000.028
308      ALT=0.
309 C      SET ASYMPTOTE
310      XASM=COS(DC)*SIN(RA)
311      YASM=SIN(DC)
312      ZASM=COS(DC)*COS(RA)
313 C*****SET UP HEADERS FOR CONSTRAINTS
314      DATA(BCD(I),I=1,52)/6H      MA,6HSS      ,6H      VE,6HL      ,
315      1 6H      GA,6HM      ,6H      R,6H      ,6H      C,6H3      ,
316      2 6H      C,6H1      ,6H      LO,6HNG      ,6H      BE,6HTA      ,
317      3 6H      CO,6H-LAT ,6H      IN,6HCL      ,6H      NO,6HDES      ,
318      4 6H      SEM,6HL R ,6H      EC,6HC      ,6H      R O,6HF P      ,
319      5 6H      R O,6HF A ,6H      TR,6H AN      ,6H      ARG,6H OF P      ,
320      6 6H      A,6HSYM      ,6H      AS,6HYM PL ,6H      RE,6HND w ,6H      RE,
321      7 6HND U ,6H      RE,6HND V ,6H      RE,6HND X ,6H      RE,6HND Y ,
322      8 6H      RE,6HND Z ,6H      IMP,6HACT      /
323      KV=1
324      DO 51 I=1,10
325      IF (KCDPHI(I).EQ.0)GO TO 51
326      J=2*KCDPHI(I)-1
327      HEADER(KV)=BCD(J)
328      HEADER(KV+1)=BCD(J+1)
329      KV=KV+2
330      51 CONTINUE
331      WRITE(6,INPUT)
332      IF (NVRST.EQ.0)RETURN
333      DO 52 I=1,6
334      IF (KCDRES(I).EQ.0)GO TO 52
335      J=2*KCDRES(I)-1
336      HEADER(KV)=BCD(J)
337      HEADER(KV+1)=BCD(J+1)
338      KV=KV+2
339      52 CONTINUE
340      RETURN
341      END

```

AINIT

LIFTING-ROBOT LISTINGS

3

.L.O.

3

ACNTRO

LIFTING-ROBOT LISTINGS

T,S TPF\$.ACNTRO
PUR 0018-05/17-11:50

ACNTRO

LIFTING-ROBOT LISTINGS

DT*TPF\$.ACNTRO

```

1      SUBROUTINE ACNTRO
2      COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
3      COMMON/TABLK/MCAN,MCAF,MCON1,MCON2,MCON3,MCON4,MBS4,MBS7,MTT1,MTT2
4      1,MTT3,MTT4
5      COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
6      1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
7      2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
8      3 VIV(8),DVAR(7),DELXD(15,3),DELXDR(7),DELXD(15,7),EXIT,
9      4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
10     5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
11     6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
12     7 BHMAX(15),BHMN,BSTEP(15)
13     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
14     1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
15     2 NOEVT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
16     COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
17     COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
18     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
19     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
20     COMMON/CONTRO/TTBL(201),CPTBL(201),CYTBL(201)
21     C GET CHIP AND CHIY AS FUNCTIONS OF TIME
22     IF(KCST.EQ.1)RETURN
23     I=KWTA-1
24     NN=NP(KCYTAB)
25     GO TO (1,2,3,4),KCST
26     1 CALL AMULG(I,MCON1,NN,T,TTBL(1),CHIP,CPTBL(1),CHIY,CYTBL(1))
27     GO TO(5,6),I
28     2 CALL AMULG(I,MCON2,NN,T,TTBL(51),CHIP,CPTBL(51),CHIY,CYTBL(51))
29     GO TO(5,6),I
30     3 CALL AMULG(I,MCON3,NN,T,TTBL(101),CHIP,CPTBL(101),CHIY,CYTBL(101))
31     GO TO(5,6),I
32     4 CALL AMULG(I,MCON4,NN,T,TTBL(151),CHIP,CPTBL(151),CHIY,CYTBL(151))
33     GO TO(5,6),I
34     C COME HERE IF CHIY=0
35     5 SCHIY=0.
36     CCHIY=1.
37     GO TO 7
38     C COME HERE IF CHIY.NE.0
39     6 SCHIY=SIN(CHIY)
40     CCHIY=COS(CHIY)
41     7 SCHIP=SIN(CHIP)
42     CCHIP=COS(CHIP)
43     RETURN
44     END

```

3 .L.O.



3

ACSTOP

LIFTING-ROBOT LISTINGS

T,S TPF\$.ACSTOP
PUR 0018-05/17-11:50

ACSTOP

LIFTING-ROBOT LISTINGS

J1*TPF\$.ACSTOP

```

1  SUBROUTINE ACSTOP(M,VALE,YL,I)
2  COMMON/TARPO/TARPO(6),TAR(6),TAULN(2)
3  COMMON/ASYMPT/ASM(3)
4  COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
5  COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHY,
6  1 SCHY,CCHY,THZ,STH,CTH,THGT,THLTL,STHL,CTHL,DPHIZ,GR,OTH,
7  2 RTH,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
8  3 VIV(8),DVAR(7),DLLXDW(15,3),DELDX(7),DELDX(15,7),EXIT,
9  4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
10 5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
11 6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
12 7 BHMAX(15),BHMN,BSTEP(15)
13 COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCSI,
14 1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWIA,LINES,NBGCT(4),NENDCT(4),NMAX,
15 2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
16 COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
17 COMMON/CSTOP/W,U,V,X,Y,Z,XM,XM,T,PEN
18 C**** N. B. --FORINT COMMON CANNOT BE IN THIS ROUTINE ****
19 DIMENSION YL(7,10)
20 DIMENSION AL(3,3),PLTARX(3),PLTARV(3)
21 DIMENSION XTMCN(6),PSPG(6),PSAP(6)
22 REAL N(6,6),ND(3)
23 IF(M.EQ.0)GO TO 1000
24 IF(M.GT.19)GO TO 20
25 GO TO (1,2,3,4,5,6,7,8,9,8,8,12,12,14,15,16,16, 6,19),M
26 C
27 C PAYLOAD
28 1 VALE=XXXM
29 YL(7,I)=1.
30 RETURN
31 C
32 C INERTIAL VELOCITY
33 2 VALE=VI
34 YL (1,I)=W/VALE
35 YL (2,I)=U/VALE
36 YL (3,I)=V/VALE
37 RETURN
38 C
39 C INERTIAL FLIGHT PATH ANGLE
40 3 SING=US/VI
41 TEMP=RAD/(VI*SQR1(AMAX1(0.,1.-SING**2)))
42 VALE=ASIN(SING)*RAD
43 IF(VALE.GT.180.)VALE=VALE-360.
44 YL(1,I)=(N(2,1)-SING*W/VI)*TEMP
45 YL(2,I)=(N(2,2)-SING*U/VI)*TEMP
46 YL(3,I)=(N(2,3)-SING*V/VI)*TEMP
47 DO 103 J=4,6
48 103 YL(J,I)=N(2,J)*TEMP
49 RETURN
50 C
51 C RADIUS
52 4 VALE=R
53 YL (4,I)=X/VALE
54 YL (5,I)=Y/VALE
55 YL (6,I)=Z/VALE
56 RETURN

```

ACSIOP

LIFTING-ROBOT LISTINGS

```

57 C
58 C ENERGY (C3)
59 5 VALE=C3
60 YL(1,I)=2.*W
61 YL(2,I)=2.*U
62 YL(3,I)=2.*V
63 YL(4,I)=2.*CMUE*X/(R2*R)
64 YL(5,I)=2.*CMUE*Y/(R2*R)
65 YL(6,I)=2.*CMUE*Z/(R2*R)
66 RETURN
67 C
68 C ANGULAR MOMENTUM (C1)
69 6 E=Y*V-U*Z
70 B=Z*W-V*X
71 C=X*U-Y*W
72 VALE=SQRT(E*E+B*B+C*C)
73 YL(1,I)=(Z*B-Y*C)/VALE
74 YL(2,I)=(X*C-Z*E)/VALE
75 YL(3,I)=(Y*E-X*B)/VALE
76 YL(4,I)=(U*C-V*B)/VALE
77 YL(5,I)=(V*E-W*C)/VALE
78 YL(6,I)=(W*B-U*E)/VALE
79 IF(M.EQ.18)GO TO 18
80 C
81 C
82 RETURN
83 RETURN
84 C
85 C INERTIAL LONGITUDE
86 7 TEMP1=A(1,1)*D(1,2)+A(2,1)*D(2,2)+A(3,1)*D(3,2)
87 TEMP2=A(1,3)*D(1,2)+A(2,3)*D(2,2)+A(3,3)*D(3,2)
88 VALE=ATAN2(TEMP1,TEMP2)*RAD
89 DO 107 J=1,6
90 107 YL(J,I)=N(4,J)*RAD
91 RETURN
92 C
93 C INERTIAL HEADING ANGLE
94 8 VALE=PI/2.-ATAN2(VS,WS)
95 TEMP1=WS/(VS*VS+WS*WS)
96 TEMP=VS/(VS*VS+WS*WS)
97 DO 108 J=1,6
98 108 YL(J,I)=(TEMP*N(1,J)-TEMP1*N(3,J))*RAD
99 IF(M.EQ.10)GO TO 10
100 IF(M.EQ.11)GO TO 11
101 VALE=VALE*RAD
102 RETURN
103 C
104 C COLATITUDE
105 9 VALE=ACOS(CTH)*RAD
106 DO 109 J=1,6
107 109 YL(J,I)=N(6,J)*RAD
108 RETURN
109 C
110 C INCLINATION
111 10 SINB=SIN(VALE)
112 COSB=COS(VALE)
113 VALE=ACOS(STH*SINB)

```

ACSTOP

LIFTING-ROBOT LISTINGS

```

114      TEMP1=SINB*CTH/SIN(VALE)*RAD
115      TEMP2=COSB*STH/SIN(VALE)
116      DO 110 J=1,6
117      110 YL(J,I)=- (TEMP1*N(6,J)+TEMP2*YL(J,I))
118      VALE=VALE*RAD
119      RETURN
120  C
121  C      LINE OF NODES
122      11 TEMP1=A(1,1)*D(1,2)+A(2,1)*D(2,2)+A(3,1)*D(3,2)
123      TEMP2=A(1,3)*D(1,2)+A(2,3)*D(2,2)+A(3,3)*D(3,2)
124      PHI=ATAN2(TEMP1,TEMP2)
125      VALE=PHI+ATAN2(CTH*WS,VS)
126      TA=WS*VS*STH/(VS**2+(WS*CTH)**2)
127      TB=(VS**2+WS**2)*CTH/(VS**2+(WS*CTH)**2)
128      DO 111 J=1,6
129      111 YL(J,I)=(N(4,J)+TB*YL(J,1)/RAD-TA*N(6,J))*RAD
130      VALE=VALE*RAD
131      IF(VALE.LT.0.)VALE=VALE+360.
132      RETURN
133  C
134  C      SEMI-LATUS RECTUM
135      12 WSVS=WS*WS+VS*VS
136      VALE=R2*WSVS/CMUE
137      TEMP=2.*R2/CMUE
138      YL(1,I)=TEMP*(W      -US*N(2,1))
139      YL(2,I)=TEMP*(U      -US*N(2,2))
140      YL(3,I)=TEMP*(V      -US*N(2,3))
141      YL(4,I)=TEMP*(WSVS*X/R2-US*N(2,4))
142      YL(5,I)=TEMP*(WSVS*Y/R2-US*N(2,5))
143      YL(6,I)=TEMP*(WSVS*Z/R2-US*N(2,6))
144      IF(M.EQ.13)GO TO 13
145      IF(M.EQ.14 .OR. M.EQ.15)GO TO 14
146      RETURN
147  C
148  C      ECCENTRICITY
149      13 PP=VALE
150      VALE=SQRT(1.+PP*C3/CMUE)
151      TEMP=C3/(2.*VALE*CMUE)
152      TEMP1=PP/(VALE*CMUE)
153      TEMP2=TEMP1*CMUE/(R*R*R)
154      YL(1,I)=TEMP*YL(1,I)+TEMP1*W
155      YL(2,I)=TEMP*YL(2,I)+TEMP1*U
156      YL(3,I)=TEMP*YL(3,I)+TEMP1*V
157      YL(4,I)=TEMP*YL(4,I)+TEMP2*X
158      YL(5,I)=TEMP*YL(5,I)+TEMP2*Y
159      YL(6,I)=TEMP*YL(6,I)+TEMP2*Z
160      RETURN
161  C
162  C      RADIUS OF PERIGEE---RADIUS OF APOGEE
163      14 XTCN(1) = W
164      XTCN(2) = U
165      XTCN(3) = V
166      XTCN(4) = X
167      XTCN(5) = Y
168      XTCN(6) = Z
169      CALL APPG(XTCN,RA,VALE,PSPG,PSAP)
170      DO 142 IJI = 1,6

```

ACSTOP

LIFTING-ROBOT LISTINGS

```

11 142 YL(IJI,I) = PSAP(IJI)
12 RETURN
13 15 VALE = RA
14 DO 152 IJI = 1,6
15 152 YL(IJI,I) = PSPG(IJI)
16 RETURN
C
C TRUE ANOMALY
16 WSVS=WS*WS+VS*VS
   QQ=WSVS*R/CMUE
   ECC=SQRT(1.+QQ*R*      C3      /CMUE)
   ETA=ATAN2(QQ*US/SQRT(WSVS),QQ-1.)
   B=VI2*QQ*(QQ-1.+2.*US*US/VI2)/(ECC*WSVS)
   TEMP=B/(ECC*SQRT(WSVS))
   TEMP1=-SIN(ETA)/(ECC*R2)
   TEMP2=-(QQ+1.)*SIN(ETA)/(ECC*WSVS)
   VALE=ETA*RAD
   IF(VALE.GT.180.)VALE=360.-VALE
   YL(1,I)=(TEMP2*W+TEMP*N(2,1))*RAD
   YL(2,I)=(TEMP2*U+TEMP*N(2,2))*RAD
   YL(3,I)=(TEMP2*V+TEMP*N(2,3))*RAD
   YL(4,I)=(TEMP1*X+TEMP*N(2,4))*RAD
   YL(6,I)=(TEMP1*Z+TEMP*N(2,6))*RAD
   YL(5,I)=(TEMP1*Y+TEMP*N(2,5))*RAD
   IF(M.EQ.17)GO TO 17
C
C RETURN
C ARGUMENT OF PERIGEE
   VALE1=ATAN2(CTH,STH*COS(BETA))*RAD
17 BETA=ATAN2(WS,VS)
   IF(VALE1.GT.180.)VALE1=360.-VALE1
   VALE=VALE+VALE1
   COSI=STH*SIN(BETA)
   SIN2I=1.-COSI*COSI
   TEMP=-COS(BETA)/SIN2I
   TEMP1=CTH*COSI*VS/(WSVS*SIN2I)
   TEMP2=-CTH*COSI*WS/(WSVS*SIN2I)
   DO 171 J=1,6
171 YL(J,I)=(TEMP*N(6,J)+TEMP1*N(1,J)+TEMP2*N(3,J))*RAD+YL(J,I)
   RETURN
C
C ASYMPTOTE CONSTRAINT
18 VNF=0.
   IF(C3.GT.0.)VNF=SQRT(C3)
   C1=VALE
   TMFA=X*ASM(1)+Y*ASM(2)+Z*ASM(3)
   TMFB=SQRT(R2-TMFA*TMFA)
   VALE=C1*(C1-VNF*TMFB)/CMUE-R+TMFA
   TMFC=(2.*C1-VNF*TMFB)/CMUE
   TMFD=-C1*TMFB/(VNF*CMUE)
   TMFE=TMFD/(R2*R)
   TMFF=C1*VNF*R/(TMFB*CMUE)+1.
   TMFG=C1*VNF*TMFA/(TMFB*CMUE)+1.
   YL(1,I)= TMFC*YL(1,I)+TMFD*W
   YL(2,I)= TMFC*YL(2,I)+TMFD*U
   YL(3,I)= TMFC*YL(3,I)+TMFD*V
   YL(4,I)= TMFC*YL(4,I)+TMFE*X-TMFF*D(1,2)+TMFG*ASM(1)

```

ACSTOP

LIFTING-ROBOT LISTINGS

```

228      YL(5,I)= TMFC*YL(5,I)+TMFE*Y-TMFF*D(2,2)+TMFG*ASM(2)
229      C
230      YL(6,I)= TMFC*YL(6,I)+TMFE*Z-TMFF*D(3,2)+TMFG*ASM(3)
231      RETURN
232      C
233      19 VALE=ASM(1)*(Y*V-U*Z)+ASM(2)*(Z*W-X*V)+ASM(3)*(X*U-Y*W)
234      YL(1,I)=Z*ASM(2)-Y*ASM(3)
235      YL(2,I)=X*ASM(3)-Z*ASM(1)
236      YL(3,I)=Y*ASM(1)-X*ASM(2)
237      YL(4,I)=U*ASM(3)-V*ASM(2)
238      YL(5,I)=V*ASM(1)-W*ASM(3)
239      YL(6,I)=W*ASM(2)-U*ASM(1)
240      RETURN
241      C
242      C
243      20 J=M-19
244      GO TO(21,22,23,24,25,26,27),J
245      21 TAUC=TAULN(1)+TAULN(2)+T
246      IF(LSB.EQ.0)CALL TARGET(TAUC)
247      SPT=SIN(OMEGA*TAULN)
248      CPT=COS(OMEGA*TAULN)
249      AL(1,1)= A(1,1)*CPT+A(1,3)*SPT
250      AL(1,3)=-(A(1,1)*SPT-A(1,3)*CPT)
251      AL(2,1)=A(2,3)*SPT
252      AL(2,3)=A(2,3)*CPT
253      AL(3,1)= A(3,1)*CPT+A(3,3)*SPT
254      AL(3,3)=-(A(3,1)*SPT-A(3,3)*CPT)
255      PLTARX(1)=AL(1,1)*TAR(1)+A(1,2)*TAR(2)+AL(1,3)*TAR(3)
256      PLTARX(2)=AL(2,1)*TAR(1)+A(2,2)*TAR(2)+AL(2,3)*TAR(3)
257      PLTARX(3)=AL(3,1)*TAR(1)+A(3,2)*TAR(2)+AL(3,3)*TAR(3)
258      PLTARV(1)=AL(1,1)*TAR(4)+A(1,2)*TAR(5)+AL(1,3)*TAR(6)
259      PLTARV(2)=AL(2,1)*TAR(4)+A(2,2)*TAR(5)+AL(2,3)*TAR(6)
260      PLTARV(3)=AL(3,1)*TAR(4)+A(3,2)*TAR(5)+AL(3,3)*TAR(6)
261      RTAR2=TAR(1)*TAR(1)+TAR(2)*TAR(2)+TAR(3)*TAR(3)
262      RTAR=SQRT(RTAR2)
263      TARA=-CMUE/(RTAR2*RTAR)
264      VALE=W-PLTARV(1)
265      YL(1,I)=1.
266      DELXD(ITHR,1)=DELXD(ITHR,1)-TARA*PLTARX(1)
267      RETURN
268      22 VALE=U-PLTARV(2)
269      YL(2,I)=1.
270      DELXD(ITHR,2)=DELXD(ITHR,2)-TARA*PLTARX(2)
271      RETURN
272      23 VALE=V-PLTARV(3)
273      YL(3,I)=1.
274      DELXD(ITHR,3)=DELXD(ITHR,3)-TARA*PLTARX(3)
275      RETURN
276      24 VALE=X-PLTARX(1)
277      YL(4,I)=1.
278      DELXD(ITHR,4)=DELXD(ITHR,4)-PLTARV(1)
279      RETURN
280      25 VALE=Y-PLTARX(2)
281      YL(5,I)=1.
282      DELXD(ITHR,5)=DELXD(ITHR,5)-PLTARV(2)
283      RETURN
284      26 VALE=Z-PLTARX(3)

```

ACSTOP

LIFTING-ROBOT LISTINGS

```

2,      YL(6,I)=1.
2,      DELXD(ITHR,6)=DELXD(ITHR,6)-PLTARV(3)
27      RETURN
23      27 VALE=PEN
29      RETURN
20      C*** INITIALIZE PARTIAL MATRICES ***
21      C CALCULATE D MATRIX
22      1000 R2=X*X+Y*Y+Z*Z
23      R=SQRT(R2)
24      VI2=W*W+U*U+V*V
25      VI=SQRT(VI2)
296     C3=VI2-2.*CMUE/R
297     D(1,2)=X/R
298     D(2,2)=Y/R
299     D(3,2)=Z/R
300     CTH=A(1,2)*D(1,2)+A(2,2)*D(2,2)+A(3,2)*D(3,2)
301     STH=SQRT(AMAX1(0.,1.-CTH*CTH))
302     RSTH=R*STH
303     D(1,1)=(-Y*A(3,2)+Z*A(2,2))/RSTH
304     D(2,1)=(X*A(3,2)-Z*A(1,2))/RSTH
305     D(3,1)=(-X*A(2,2)+Y*A(1,2))/RSTH
306     D(1,3)=(D(1,2)*CTH-A(1,2))/STH
307     D(2,3)=(D(2,2)*CTH-A(2,2))/STH
308     D(3,3)=(D(3,2)*CTH-A(3,2))/STH
309     WS=D(1,1)*W+D(2,1)*U+D(3,1)*V
310     US=D(1,2)*W+D(2,2)*U+D(3,2)*V
311     VS=D(1,3)*W+D(2,3)*U+D(3,3)*V
312     C CALCULATE N MATRIX
313     DO 1001 I=1,3
314     ND(I)=D(I,3)*CTH+D(I,2)*STH
315     N(4,I+3)=D(I,1)/RSTH
316     N(5,I+3)=D(I,2)
317     N(6,I+3)=D(I,3)/R
318     DO 1001 J=1,3
319     N(I,J)=D(J,I)
320     1001 N(I+3,J)=0.
321     TEMP=WS*CTH/STH
322     N(1,4)=(A(3,2)*U-A(2,2)*V-WS*ND(1))/RSTH
323     N(1,5)=(A(1,2)*V-A(3,2)*W-WS*ND(2))/RSTH
324     N(1,6)=(A(2,2)*W-A(1,2)*U-WS*ND(3))/RSTH
325     N(2,4)=(W-D(1,2)*US)/R
326     N(2,5)=(U-D(2,2)*US)/R
327     N(2,6)=(V-D(3,2)*US)/R
328     N(3,4)=(D(1,1)*TEMP-D(1,3)*US)/R
329     N(3,5)=(D(2,1)*TEMP-D(2,3)*US)/R
330     N(3,6)=(D(3,1)*TEMP-D(3,3)*US)/R
331     RETURN
332     END

```

5 .L,0.



3

ADDR

LIFTING-ROBOT LISTINGS

T,S TPF\$.ADDR
PUR 0018-05/17-11:50

ADDR

LIFTING-ROBOT LISTINGS

OT*TPF\$.ADDR

```

1      SUBROUTINE ADDR(NCH,YS)
2      COMMON/DFEQS/N,XN,X,XH,II,YN(6),Y(6),Y1(6),DY(6)
3      COMMON/GRAV/GT0,G11,G22,G23,G33
4      COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CH1Y,
5      1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
6      2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
7      3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
8      4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
9      5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
10     6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
11     7 BHMAX(15),BHMN,BSTEP(15)
12     DIMENSION YS(3,3)
13
14     C      THIS SUBROUTINE SPECIFIES THE ADJOINT DIFFERENTIAL EQUATIONS
15     C
16     C      IF NCH = 1 INTEGRATE BACKWARDS, OTHERWISE INTEGRATE FORWARD.
17     C
18     1 II = 1
19     2 CALL FORKM
20     IF(II.NE.6) GO TO 3
21     RETURN
22
23     C      RETRIEVE STORED SOLUTION
24     C
25     3 IF(II.GT.4) GO TO 5
26     IF(II.GT.2) GO TO 4
27     AX = YS(1,1)
28     AY = YS(2,1)
29     AZ = YS(3,1)
30     GO TO 6
31     4 AX = YS(1,2)
32     AY = YS(2,2)
33     AZ = YS(3,2)
34     GO TO 6
35     5 AX = YS(1,3)
36     AY = YS(2,3)
37     AZ = YS(3,3)
38     6 R2 = AX*AX + AY*AY + AZ*AZ
39     R = SQRT(R2)
40     CTH = (A(1,2)*AX + A(2,2)*AY + A(3,2)*AZ)/R
41     CT2 = CTH*CTH
42     CALL AGE0(2)
43
44     C      NEXT CONSTRUCT SECOND VARIATION GRAVITY TERMS
45     C
46     TEMP1 = G22*AX + G23*A(1,2)
47     TEMP2 = G22*AY + G23*A(2,2)
48     TEMP3 = G22*AZ + G23*A(3,2)
49     TEMP4 = G33*A(1,2) + G23*AX
50     TEMP5 = G33*A(2,2) + G23*AY
51     TEMP6 = G33*A(3,2) + G23*AZ
52     GXX = G11 + TEMP1*AX + TEMP4*A(1,2)
53     GYX = TEMP1*AY + TEMP4*A(2,2)
54     GZX = TEMP1*AZ + TEMP4*A(3,2)
55     GYY = G11 + TEMP2*AY + TEMP5*A(2,2)
56     GZY = TEMP2*AZ + TEMP5*A(3,2)

```

ADDR

LIFTING-ROBOT LISTINGS

```
57      GZZ = G11 + TEMP5*AZ + TEMP6*A(3,2)
58      DY(1) = Y(4)
59      DY(2) = Y(5)
60      DY(3) = Y(6)
61      DY(4) = GXX*Y(1) + GYX*Y(2) + GZX*Y(3)
62      DY(5) = GYX*Y(1) + GYY*Y(2) + GZY*Y(3)
63      DY(6) = GZX*Y(1) + GZY*Y(2) + GZZ*Y(3)
64      IF(NCH.EQ.1) GO TO 2
65      DO 7 I = 1,6
66 7      DY(I) = -1.*DY(I)
67      GO TO 2
68      END
```

3 .L,0.

3

ADER1

LIFTING-ROBOT LISTINGS

T, S TPF3, ADER1
PUR 0018-05/17-11:50

ADER1

LIFTING-ROBOT LISTINGS

DT*TPF\$,ADER1

```

1      SUBROUTINE ADER1
2      COMMON/CONVEC/XCV,YCV,ZCV
3      COMMON/TABLK/MCAN,MCAF,MCON1,MCON2,MCON3,MCON4,MBS4,MBS7,MTT1,MTT2
4      1 ,MTT3,MTT4
5      COMMON/FSCY/NFSCY,FSCYD,TFSCY
6      COMMON/CYT/NCYT,TCY1,TCY2,TCY3,TCY4,CYTM
7      COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
8      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
9      1  TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
10     2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
11     3  HNOM(15),HMAX(15)
12     COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
13     1  SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
14     2  RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
15     3  VIV(8), DVAR(7), DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
16     4  DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
17     5  FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
18     6  CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
19     7  BHMAX(15),BHMN,BSTEP(15)
20     COMMON/AGEN3/IORK,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
21     1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
22     2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
23     COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
24     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
25     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
26     COMMON/JTHR/JTHR(15),TBLTM(4)
27     COMMON/XMIAD/XMIAD,DELMA
28     1 IF(JTHR(ITHR).EQ.0)GO TO 2
29     MMM=IABS(JTHR(ITHR))
30     ACT=T-TBLTM(MMM)
31     CALL ATTRAC(ACT,FTRAC,XMDRAC,MMM,AE(ITHR),TNE(2,1),TNE(4,1))
32     40 THR=ANE(1)*FTRAC+PSL*EXIT
33     10 IF(JTHR(ITHR))11,2,12
34     12 XMDOT=ANE(2)*XMDRAC
35     GO TO 2
36     11 XMIAD=XMIAD-XMDRAC+DELMA
37     DELMA=XMDRAC
38     2 DVAR(7)=-XMDOT
39     IF(JTHR(ITHR).GE.0)XMIAD=XMI
40     GO TO(3,4,6,7),KS1
41 C**  GET HERE DURING INITIAL LIFT OFF
42     3 PHIL=DPHIZ+OMEGA*(T-TZERO)
43     STSP=STHL*SIN(PHIL)
44     CPST=STHL*COS(PHIL)
45     XCV =A(1,1)*STSP+A(1,2)*CTHL+A(1,3)*CPST
46     YCV =          A(2,2)*CTHL+A(2,3)*CPST
47     ZCV =A(3,1)*STSP+A(3,2)*CTHL+A(3,3)*CPST
48     RETURN
49 C    TILTOVER
50     4 CHIP=CHIDOT*(T-TLIFT)
51     5 XCV=SIN(CHIP)
52     YCV=COS(CHIP)
53     ZCV=0.
54 C    GRAV TURN
55     6 RETURN
56 C    OPT CON

```

ADER1

LIFTING-ROBOT LISTINGS

```
57      7 CALL ACNTRO
58      IF(KWTA.EQ.2)GO TO 5
59      CCHIY=COS(CHIY)
60      XCV=CCHIY*SIN(CHIP)
61      YCV=CCHIY*COS(CHIP)
62      ZCV=SIN(CHIY)
63      RETURN
64      END
```

3 .L,0.

3

ADER

LIFTING-ROBOT LISTINGS

T,S TPF\$.ADER
PUR 0018-05/17-11:50

ADER

LIFTING-ROBOT LISTINGS

OT*TPF\$.ADER

```

1      SUBROUTINE ADER
2      COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
3      COMMON/ARODTA/ACF,BCF,CCF,DADM,DBDM,DCDM
4      COMMON/CONVEC/XCV,YCV,ZCV
5      COMMON/FAB/FAB,IFAB,MFAB,FABAL(44),FABTB(44)
6      COMMON/FSCY/NFSCY,FSCYD,TFSCY
7      COMMON/TABLK/MCAN,MCAF,MCON1,MCON2,MCON3,MCON4,MBS4,MBS7,MTT1,MTT2
8      1,MTT3,MTT4
9      COMMON/CADATA/PNM(44),CAN(44),CNN(44),PFM(25),CAF(25),CNF(25)
10     COMMON/ARDC/PAT(15)
11     COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
12     COMMON/GRAV/GT0,G11,G22,G23,G33
13     COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
14     COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
15     1 TCHFRZ,DTZ,TQ,IL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
16     2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
17     3 HNOM(15),HMAX(15)
18     COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
19     1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPH1Z,GR,GTH,
20     2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
21     3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
22     4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
23     5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
24     6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
25     7 BHMAX(15),BHMN,BSTEP(15)
26     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
27     1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
28     2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
29     COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
30     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
31     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
32     COMMON/XMIAD/XMIAD,DELMA
33     R2=X*X+Y*Y+Z*Z
34     R=SQRT(R2)
35     IF(JORB.EQ.0)GO TO 1
36     G11=-CMUE/(R*R2)
37     GX=X*G11
38     GY=Y*G11
39     GZ=Z*G11
40     ALT=R-RE
41     GO TO 2
42     1 CTH=(A(1,2)*X+A(2,2)*Y+A(3,2)*Z)/R
43     CT2=CTH*CTH
44     CALL AGE0(1)
45     GX=X*G11-A(1,2)*GT0
46     GY=Y*G11-A(2,2)*GT0
47     GZ=Z*G11-A(3,2)*GT0
48     RTHE=UMF*RE/SQRT(UMF*UMF*(1.-CT2)+CT2)
49     ALT=R-RTHE
50     2 SW=W-(Z*A22W-Y*A32W)
51     SU=U-(X*A32W-Z*A12W)
52     SV=V-(Y*A12W-X*A22W)
53     VR2=SW*SW+SU*SU+SV*SV
54     VR=SQRT(VR2)
55     C
56     C*****

```

ADER

LIFTING-ROBOT LISTINGS

```

57      PAT(1)=ALT
58      CALL PRA63(PAT(1),IER)
59      PAM=PAT(2)*10000.
60      RHO=PAT(6)
61      XMACH=VR/PAT(9)
62      CALL CACN(XMACH,1)
63      4 QSOV2=.5*RHO*S(ITHR)
64      QSOV=QSOV2*VR
65      QS=QSOV*VR
66      IF(KS1.NE.3)GO TO 7
67      C GRAV TURN
68      5 TEMP=SQRT(SW*SW+SU*SU)
69      XCV=SW/TEMP
70      YCV=SU/TEMP
71      ZCV=0.
72      7 CONTINUE
73      VDOTC=SW*XCV+SU*YCV+SV*ZCV
74      THROT=1.
75      XM=XMIAD+XMAUG
76      IF (THR.GT.GLIM(ITHR)*XM) THROT=XM*GLIM(ITHR)/THR
77      QSBPC=QS*(BCF+CCF)
78      THMFA=THR*THROT-EXIT*PAM-QSBPC
79      THMFB=QSOV2*BCF*VDOTC-QSOV*ACF
80      FX=THMFA*XCV+THMFB*SW
81      FY=THMFA*YCV+THMFB*SU
82      FZ=THMFA*ZCV+THMFB*SV
83      9 DVAR(1)=FX/XM+GX
84      DVAR(2)=FY/XM+GY
85      DVAR(3)=FZ/XM+GZ
86      DVAR(4)=W
87      DVAR(5)=U
88      DVAR(6)=V
89      IF (THROT.LT.1.)DVAR(7)=-XMDOT*THROT
90      IF (NF8-8)16,14,15
91      14 CALL IMPPT(0)
92      GO TO 16
93      15 CALL IMPPT(0)
94      VMAG=SQRT(W*W+U*U+V*V)
95      COSAL=(W*XCV+U*YCV+V*ZCV)/VMAG
96      COSV=(W*SW+U*SU+V*SV)/VMAG
97      VDOTG=(W*GX+U*GY+V*GZ)/VMAG
98      DVAR(9)=(THR*THROT-EXIT*PAM)/XM
99      DVAR(10)=DVAR(9)*(1.-COSAL)
100     DVAR(11)=-VDOTG
101     DVAR(12)=Q*VR/(PI/2.-ALF)
102     DVAR(13)=- (THMFB*COSV-QSBPC*COSAL)/XM
103     16 CONTINUE
104     IF(NTRG6.LT.0) RETURN
105     C
106     C CALCULATE Q DOT
107     C CALCULATE P(RHO)WRT ALT
108     Q=.5*RHO*VR2
109     PAT(1)=ALT+20.
110     CALL PRA63(PAT(1),IER)
111     RHO2=PAT(6)
112     PAT(1)=ALT-20.
113     CALL PRA63(PAT(1),IER)

```

ADER

LIFTING-ROBOT LISTINGS

```

114      DRDR=.025*RHO*ALOG(RHO2/PAT(6))
115      C      CALCULATE DQDX
116      TEMP=Q*DRDR/(RHO*R)
117      DQDX(3)=RHO*SV
118      DQDX(1)=RHO*SW
119      DQDX(2)=RHO*SU
120      DQDX(4)=TEMP*X-DQDX(2)*A32W+DQDX(3)*A22W
121      DQDX(5)=TEMP*Y+DQDX(1)*A32W-DQDX(3)*A12W
122      DQDX(6)=TEMP*Z-DQDX(1)*A22W+DQDX(2)*A12W
123      IF(JORB.EQ.1)GO TO 10
124      C      CALCULATE D(R)WRTD(THETA)
125      AUX3=RTHE/RE
126      DRTOS=AUX3*AUX3*RTHE*(2.-FLAT)/(UMF*UMF)*CTH*FLAT
127      C
128      TEMP=TEMP*DRTOS
129      DQDX(4)=DQDX(4)-TEMP*(X/R*CTH-A(1,2))
130      DQDX(5)=DQDX(5)-TEMP*(Y/R*CTH-A(2,2))
131      DQDX(6)=DQDX(6)-TEMP*(Z/R*CTH-A(3,2))
132      10 QDOT=0.
133      DO 11 I=1,6
134      11 QDOT=QDOT+DQDX(I)*DVAR(I)
135      RETURN
136      END

```

5 .L,0.

3

ADER11

LIFTING-ROBOT LISTINGS

T,S TPF\$.ADER11
PUR 0018-05/17-11:50

ADER11

LIFTING-ROBOT LISTINGS

BT*TPF\$.ADER11

```

1      SUBROUTINE ADER11
2      COMMON/AGEN1/ TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
3      1TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
4      2WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
5      3HNOM(15),HMAX(15)
6      COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHY,
7      1 SCHY,CCHY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
8      2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
9      3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
10     4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
11     5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
12     6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
13     7 BHMAX(15),BHMN,BSTEP(15)
14     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
15     1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
16     2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
17     COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
18     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
19     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
20     COMMON/JTHR/JTHR(15),TBLTM(4)
21     COMMON/TABLK/MCAN,MCAF,MCON1,MCON2,MCON3,MCON4,MBS4,MBS7,MTT1,MTT2
22     1,MTT3,MTT4
23     COMMON/XMIAD/XMIAD,DELMA
24     IF(JTHR(ITHR).EQ.0)GO TO 1
25     MMM=IABS(JTHR(ITHR))
26     ACT=T-TBLTM(MMM)
27     CALL ATTRAC(ACT,FTRAC,XMDRAC,MMM,0.,0.,0.)
28     THR=ANE(1)*FTRAC
29     IF(JTHR(ITHR).LT.0)GO TO 5
30     6 XMDOT=ANE(2)*XMDRAC
31     GO TO 1
32     5 XMIAD=XMIAD-XMDRAC+DELMA
33     DELMA=XMDRAC
34     1 CALL ACNTR0
35     IF(JTHR(ITHR).GE.0)XMIAD=XMI
36     IF(KWTA.EQ.3)GO TO 2
37     FX=THR*SCHIP
38     FY=THR*CCHIP
39     FZ=0.
40     GO TO 3
41     2 FY=THR*CCHY
42     FX=FY*SCHIP
43     FY=FY*CCHIP
44     FZ=THR*SCHY
45     3 DVAR(7)=-XMDOT
46     RETURN
47     END

```

3 .L,0.

3

ADERI

LIFTING-ROBOT LISTINGS

T,S TPF\$.ADERI
PUR 0018-05/17-11:50

ADERI

LIFTING-ROBOT LISTINGS

JT*TPF\$.ADERI

```

1      SUBROUTINE ADERI
2      COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
3      COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
4      COMMON/GRAV/GTO,G11,G22,G23,G33
5      COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
6      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
7      1 TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
8      2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
9      3 HNOM(15),HMAX(15)
10     COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
11     1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
12     2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
13     3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
14     4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
15     5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
16     6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
17     7 BHMAX(15),BHMN,BSTEP(15)
18     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCSI,
19     1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
20     2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
21     COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
22     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
23     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
24     COMMON/XMIAD/XMIAD,DELMA
25     R2=X*X+Y*Y+Z*Z
26     R=SQRT(R2)
27     IF(JORB.EQ.0)GO TO 1
28     G11=-CMUE/(R2*R)
29     GX=X*G11
30     GY=Y*G11
31     GZ=Z*G11
32     GO TO 2
33     1 CTH=(A(1,2)*X+A(2,2)*Y+A(3,2)*Z)/R
34     CT2=CTH*CTH
35     CALL AGE0(1)
36     GX=X*G11-A(1,2)*GTO
37     GY=Y*G11-A(2,2)*GTO
38     GZ=Z*G11-A(3,2)*GTO
39     2 XM=XMIAD+XMAUG
40     THROT=XM*GLIM(ITHR)/THR
41     IF(THROT.GT.1.)THROT=1.
42     TEMP=THROT/XM
43     DVAR(1)=FX*TEMP+GX
44     DVAR(2)=FY*TEMP+GY
45     DVAR(3)=FZ*TEMP+GZ
46     DVAR(4)=W
47     DVAR(5)=U
48     DVAR(6)=V
49     IF(THROT.LT.1.)DVAR(7)=-XMDOT*THROT
50     IF(NF8-8)16,14,15
51     14 CALL IMPPT(0)
52     GO TO 16
53     15 CALL IMPPT(0)
54     VMAG=SQRT(W*W+U*U+V*V)
55     COSAL=((W*SCHIP+U*CCHIP)*CCHIY+V*SCHIY)/VMAG
56     SINAL=SQRT(1.-COSAL*COSAL)

```

ADERI

LIFTING-ROBOT LISTINGS

```
57      VDOTG=(W*GX+U*GY+V*GZ)/VMAG
58      DVAR(9)=THR*TEMP
59      DVAR(10)=DVAR(9)*(1.-COSAL)
60      DVAR(11)=-VDOTG
61  16  CONTINUE
62      RETURN
63      END
```

3 .L.O.

9

AEOSR

LIFTING-ROBOT LISTINGS

T, S TPF\$.AEOSR
PUR 0018-05/17-11:50

AEOSR

LIFTING-ROBOT LISTINGS

DT*TPF\$.AEOSR

```

1      SUBROUTINE AEOSR
2      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
3      1  TCHFRZ,DTZ,TQ,IL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
4      2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
5      3  HNOM(15),HMAX(15)
6      COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
7      1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
8      2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
9      COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
10     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
11     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
12     COMMON/TABS/TBL(151),XBL(151),YBL(151),ZBL(151),XMBL(151)
13     1 ,WBL(151),UBL(151),VBL(151)
14     COMMON/XMIAD/XMIAD,DELMA
15     COMMON/PENLTY/NCUNPT,DXS(6)
16
17     C
18     C      SAVES STATE VARIABLE PTS. FROM FORWARD INTEGRATION
19     IF(TL.GT.T)RETURN
20     IXR=IXR+1
21     C**
22     SET 4 STATES AND TIME IN TABLES
23     TBL(IXR)=T
24     XBL(IXR)=X
25     YBL(IXR)=Y
26     ZBL(IXR)=Z
27     XMBL(IXR)=XMIAD
28     WBL(IXR)=W
29     UBL(IXR)=U
30     VBL(IXR)=V
31     1 TL=T+STEP(ITHR)-.2
32     IF(IXR.LT.150)RETURN
33     C
34     WRITE OUT SAVED VALUES (4 STATES)
35     WRITE( 3)(TBL(I),XBL(I),YBL(I),ZBL(I),XMBL(I),
36     * WBL(I),UBL(I),VBL(I),I=1,150)
37     IXR=0
38     RETURN
39     END

```

3 .L,0.

AFORND

LIFTING-ROBOT LISTINGS

T.S TPF\$.AFORND
PUR 0018-05/17-11:50

AFORND

LIFTING-ROBOT LISTINGS

BT*TFB\$.AFORND

```

1  SUBROUTINE AFORND
2  COMMON/COUNT/ICOUNT,ICONT, LONGPR
3  COMMON/LPRINT/LPRINT
4  COMMON/PREDP/PREDP
5  COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
6  COMMON/PERF/ILAST,IPR,PRK1,PRK2,PRK3,PRK4,PRTOT,FGR,FPR,AL
7  COMMON/DRHO/DRHO(9),E1,WIPPB,WGNU
8  COMMON/RESTX/RESTX(8)
9  COMMON/CSTOP/CSTVAR(9)
10 COMMON/AUTO/KDB(30),SAVCP(30),XLAMB(30,10),DP2,KDT(30)
11 COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2
12 COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,IL
13 1 TCHFRZ,DTZ,TQ,IL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
14 2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
15 3 HNOM(15),HMAX(15)
16 COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIY,CHLY,
17 1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPH12,GR,GTH,
18 2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,X2,Y2,ZZ,WRZ,
19 3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
20 4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CB,
21 5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
22 6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
23 7 BHM(15),BHMN,BSTEP(15)
24 COMMON/AGEN3/IORK,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
25 1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
26 2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
27 COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
28 1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
29 2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
30 COMMON/YLI/YLINT(7,10),PHITE(10),PSIREQ(10),KCDPHI(10),PSTR(10)
31 COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
32 COMMON/INPUT/AZ,LAT,DRAG,CHIUT,W01,WPM,CYT,FSCY,RTASC,DECL,DELVG,
33 1 DELVP,VELTGT,GAMTGT,RTGT,INCTGT,LATTGT,BTATGT,IFAD,IAA,IPHIT,
34 2 PSIRST(6),HEADER(60),FLBS(15),WDOT(15),CWDOT(15),WDLBS(15),
35 3 WTJET(15),LAST
36 COMMON/XMIAD/XMIAD,DELMA
37 COMMON/AFPRN/VI,GAM,C1,C3,XINC,XNOD,XMLB,THETG,THETA,PHI,
38 ARNGO,RNGAN,AZ1
39 COMMON/FSCY/NFSCY,FSCYD,TFSCY
40 DIMENSION GOT(10),ASK(10),PERC(10)
41 C** SET LAST STATE POINT IN TABLES
42 C EVALUATE TERMINAL CONDITIONS AND INITIALIZE LAMBDAS
43 C*****NOSS AT THIS POINT IS EQUAL TO REGULAR SETS-SET BY ANEWCH
44 CSTVAR(1)=W
45 CSTVAR(2)=U
46 CSTVAR(3)=V
47 CSTVAR(4)=X
48 CSTVAR(5)=Y
49 CSTVAR(6)=Z
50 CSTVAR(7)=XM
51 CSTVAR(8)=T
52 CSTVAR(9)=ZAP(1)
53 CALL ACSTOP(0,VALE,YLINT,I)
54 DO 4 I=1,NOSS
55 DO 2 J=1,7
56 2 YLINT(J,I)=0.

```

```

57      KL=KCDPHI(I)
58      CALL ACSTOP(KL,VALE,YLINT,I)
59      IF(I.EQ.1)GO TO 3
60      PHITE(I)=VALE-PSIREQ(I-1)
61      GO TO 4
62      3 PHITE(I)=VALE
63      4 CONTINUE
64      IF(NVRST.EQ.0)GO TO 15
C**** IF NVRST.NE.0 EVALUATE RESTART TERMINAL COND AND INIT LAMS
65      DO 16 I=1,8
66      16 CSTVAR(I)=RESTX(I)
67      CALL ACSTOP(0,VALE,YLINT,I)
68      DO 18 I=1,NCNRS
69      JA=NOSS+I
70      DO 17 J=1,7
71      17 YLINT(J,JA)=0.
72      KL=KCDRES(I)
73      CALL ACSTOP(KL,VALE,YLINT,JA)
74      18 PHITE(JA)=VALE-PSIRST(I)
75      C** WRITE OUT TEST MATRIX
76      15 IF(LSB.EQ.1)RETURN
77      K=2*(NOSS+NCNRS)
78      WRITE(6,56)CASE,(HEAD(I),I=1,10)
79      56 FORMAT(1H1,6H ROBOT,2X,6HCASE =,F11.4,10X,10A6///)
80      57 WRITE(6,20)PRTOT,FGR,FPR
81      20 FORMAT(1X,4HFPR=,E15.8,7H GPR=,E15.8,7H PPR=,E15.8)
82      WRITE(6,5)(HEADER(I),I=1,K)
83      5 FORMAT(//,1X,6H ,20A6)
84      ASK(1)=E1*XJEXT*(WIPPB-WGNU)
85      JA=NOSS+NCNRS-1
86      ASK(1)=ASK(1)-PREDP
87      DO 6 I=1,JA
88      6 ASK(I+1)=DRHO(I)
89      JA=JA+1
90      DO 7 I=1,JA
91      GOT(I)=PHITE(I)-PSTR(I)
92      7 PERC(I)=GOT(I)/ASK(I)*100.
93      WRITE(6,8)(PSTR(I),I=1,JA)
94      WRITE(6,9)(PHITE(I),I=1,JA)
95      WRITE(6,10)(ASK(I),I=1,JA)
96      WRITE(6,11)(GOT(I),I=1,JA)
97      WRITE(6,12)(PERC(I),I=1,JA)
98      8 FORMAT(1X,6HOLD ,10E12.5)
99      9 FORMAT(1X,6HNEW ,10E12.5)
100     10 FORMAT(1X,6HASK ,10E12.5)
101     11 FORMAT(1X,6HGOT ,10E12.5)
102     12 FORMAT(1X,6HPCNT ,10E12.5)
103     DO 13 I=1,JA
104     13 PSTR(I)=PHITE(I)
105     DO 14 I=1,15
106     14 SAVCP(I)=TAUT(I)
107     SAVCP(16)=WZERO / .45359237
108     SAVCP(17)=CHIDOT*RAD
109     SAVCP(18)=AA*RAD
110     SAVCP(19)=FSCYD*RAD
111     WRITE(6,30)
112     30 FORMAT(/,1X,10HPARAMETERS)

```

A FORND

LIFTING-ROBOT LISTINGS

```

114      WRITE(6,25)(SAVCP(I),I=1,15)
115      WRITE(6,25)(SAVCP(I),I=16,20)
116      IF(LPRINT.EQ.2.AND.NMAX.NE.0) GO TO 39
117      IF(LPRINT.EQ.2.AND.ICONT-LONGPR.GT.ICOUNT)GO TO 39
118      GAMX=GAM/RAD
119      H1S=R*VI*COS(GAMX)
120      P=H1S**2/CMUE
121      CJD=CMUE/ABS(C3)
122      E=SQRT(ABS(1.-(P/CJD)))
123      IF(E.GT..00005)GO TO 34
124      E=0.
125      34 RA=CJD*(1.+E)
126      RP=CJD*(1.-E)
127      HA=(RA-RE)/1852.
128      HP=(RP-RE)/1852.
129      WRITE(6,37)
130      37 FORMAT(1H1)
131      WRITE(6,38)(HEAD(I),I=1,10)
132      38 FORMAT(29X,10A6,/)
133      WRITE(6,36)T,RNGO,RNGAN,VI,R,GAM,XINC,XNOD,SAVCP(18),THETG,THETA,
134      APhi,AZI,C1,XMLB,C3
135      36 FORMAT(5X4HTIMEE14.8,1X3HSEC12X5HRANGE14.8,1X1HM10X11HRANGE ANGLE
136      AE14.8,1X3HDEG ///1X8HINERTIALE14.8,1X5HM/SEC9X6HRADIUSE14.8,1X1H
137      BM11X6HFLIGHT4XE14.8,1X3HDEG /1X8HVELOCITY62X10HPATH ANGLE///
138      EE14.8,1X3HDEG15X4HLONGE14.8,1X3HDEG ///1X8HINERTIALE14.8,1X3HDEG
139      C5X4HINCLE14.8,1X3HDEG13X4HDES.E14.8,1X3HDEG12X6HFLIGHT1XE14.8,1X3H
140      DDEG /40X4HNODE30X7HAZIMUTH ///4X5HGDLE14.8,1X3HDEG12X5HGCLAT
141      F14X2HC11XE14.8,1X8HM**2/SEC /2X7HAZIMUTH///
142      G3X6HWEIGHTE14.8,1X3HLBS14X2HC31XE14.8,1X11HM**2/SEC**2/// )
143      WRITE(6,40)E,RA,RP,HA,HP
144      40 FORMAT(1X28HKEPLERIAN ORBITAL PARAMETERS // 1X8HECNTRICYE14.8,15X
145      A6HAPOGEE14.8,1X1HM14X7HPERIGEE14.8,1X1HM /38X6HRADIUS31X6HRADIUS
146      B///3X6HAPOGEE14.8,1X2HNM11X7HPERIGEE14.8,1X2HNM /3X6HHEIGHT29X
147      C6HHEIGHT)
148      WRITE(6,37)
149      39 CONTINUE
150      25 FORMAT( 1X,8E15.8,/,1X,8E15.8)
151      SAVCP(16)=SAVCP(16)*.45359237
152      SAVCP(17)=SAVCP(17)/RAD
153      SAVCP(18)=SAVCP(18)/RAD
154      SAVCP(19)=SAVCP(19)/RAD
155      RETURN
156      END

```

5

.L.O.

3
AFORUN

LIFTING-ROBOT LISTINGS

T,S TPF\$.AFORUN
PUR 0018-05/17-11:50

AFORUN

LIFTING-ROBOT LISTINGS

DT*TPF\$.AFORUN

```

1  SUBROUTINE AFORUN
2  COMMON/KLIFT/KLIFT
3  COMMON/JTHR/JTHR(15),TBLTM(4)
4  COMMON/FAB/FAB,IFAB,MFAB,FABAL(44),FABTB(44)
5  COMMON/ASYMPT/ASM(3)
6  COMMON/AZMTH/STH1,CTH1,XASM,YASM,ZASM,XJX(3),XJV(3),NALTW
7  COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
8  COMMON/TABLK/MCAN,MCAF,MCON1,MCON2,MCON3,MCON4,MBS4,MBS7,MTT1,MTT2
9  1,MTT3,MTT4
10 COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
11 1 TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
12 2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
13 3 HNOM(15),HMAX(15)
14 COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
15 1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
16 2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
17 3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
18 4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
19 5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
20 6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
21 7 BHMAX(15),BHMN,BSTEP(15)
22 COMMON/AGEN3/IORK,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
23 1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
24 2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
25 COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
26 COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
27 1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
28 2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
29 COMMON/XMIAD/XMIAD,DELMA
30 COMMON/PENLTY/NCONPT,DXS(6)
31 COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
32 COMMON/DELX/DELX(10),XLAMBX(3,10)
33 C
34 C FORWARD INTEGRATION
35 C CALCULATE A MATRIX
36 SA=SIN(AA)
37 CA=COS(AA)
38 A(1,1)=SA
39 A(2,1)=0.
40 A(3,1)=CA
41 A(1,2)=CA*STH1
42 A(2,2)=CTH1
43 A(3,2)=-SA*STH1
44 A(1,3)=-CA*CTH1
45 A(2,3)=STH1
46 A(3,3)=SA*CTH1
47 C
48 A12W=A(1,2)*OMEGA
49 A22W=A(2,2)*OMEGA
50 A32W=A(3,2)*OMEGA
51 ASM(1)=A(1,1)*XASM+A(1,2)*YASM+A(1,3)*ZASM
52 ASM(2)=A(2,1)*XASM+A(2,2)*YASM+A(2,3)*ZASM
53 ASM(3)=A(3,1)*XASM+A(3,2)*YASM+A(3,3)*ZASM
54 IF(JUMP,NE.1 .AND. VIV(7).LT.1.)GO TO 41
55 XZ=A(1,1)*XJX(1)+A(1,2)*XJX(2)+A(1,3)*XJX(3)
56 YZ= A(2,2)*XJX(2)+A(2,3)*XJX(3)

```

AFORUN

LIFTING-ROBOT LISTINGS

```

57      ZZ=A(3,1)*XJX(1)+A(3,2)*XJX(2)+A(3,3)*XJX(3)
58      WZ=A(1,1)*XJV(1)+A(1,2)*XJV(2)+A(1,3)*XJV(3)
59      UZ=      A(2,2)*XJV(2)+A(2,3)*XJV(3)
60      VZ=A(3,1)*XJV(1)+A(3,2)*XJV(2)+A(3,3)*XJV(3)
61      GO TO 99
62      41 WZ=VIV(1)
63      UZ=VIV(2)
64      VZ=VIV(3)
65      XZ=VIV(4)
66      YZ=VIV(5)
67      ZZ=VIV(6)
68      99 CONTINUE
69      LLDSOL=5
70      IF(KIND.NE.2)LLDSOL=3*(KRDER+2)+4
71      C ** KRDER CAN NOT BE GREATER THAN 5
72      C REWIND SAVE STATE TAPE
73      REWIND 3
74      C INITIALIZE TABLE LOOK-UP INDICES
75      100 MCAN=1
76      MCAF=1
77      MCON1=1
78      MCON2=1
79      MCON3=1
80      MCON4=1
81      MTT1=1
82      MTT2=1
83      MTT3=1
84      MTT4=1
85      MFAB=1
86      KLIFT=0
87      KS1=1
88      LINES=0
89      IXR=0
90      TL=0.
91      QDOT=.1
92      KUIR=0
93      IWD=0
94      KCYTAB=1
95      KCST=0
96      NF8=7
97      IF(NCONPT.GT.0)NF8=8
98      IF(JUMP.NE.1)TQ=IZERO
99      W=WZ
100     U=UZ
101     V=VZ
102     X=XZ
103     Y=YZ
104     Z=ZZ
105     DO 101 I=1,8
106     101 ZAP(I)=0.
107     C
108     C EXTERNAL AXPRT,AWDEV,ATILT,AKALT,AEOSR,AQMAX
109     C
110     C TAUT IS DURATION TIME OF THRUST EVENTS
111     C TAUW IS TIME INTERVAL BETWEEN A THRUST EVENT AND A WT DROP
112     C
113     XMAUG=WJET(JUMP)

```

AFORUN

LIFTING-ROBOT LISTINGS

```

114      TIME(1,JUMP)=TZERO
115      DO 1 I=JUMP,NVNT
116      TIME(1,I+1)=TIME(1,I)+TAUT(I)
117      1 XMAUG=XMAUG+WJET(I)
118      IWDJ=0
119      I=1
120      2 J=NOWD(I)
121      IF(J.GE.JUMP)GO TO 3
122      IF(I.GE.NWVNT)GO TO 5
123      I=I+1
124      GO TO 2
125      3 IWDJ=I
126      DO 4 I=IWDJ,NWVNT
127      J=NOWD(I)
128      TIME(2,I)=TIME(1,J)+TAUW(I)
129      4 XMAUG=XMAUG+WD(I)
130      5 XM=WZERO
131      XMI=XM-XMAUG
132      XMIAD=XMI
133      DELMA=0.
134      T=TZERO
135      C*** CHECK FOR THRUST TABLES
136      60 IF(NOM1.NE.0)GO TO 56
137      NJJ=NVNT+1
138      DO 55 I=1,NVNT
139      J=NJJ-1
140      MMM=IABS(JTHR(J))
141      IF(MMM.LE.0)GO TO 55
142      TBLTM(MMM)=TIME(1,J)
143      55 CONTINUE
144      56 CONTINUE
145      C
146      C      TRIGGERS      1-PRINT,2-THRUST,3-WT DROP,4-TILT,5-ALT,6-QMX
147      NOTRG=5
148      IF(NMAX.EQ.0)NOTRG=7
149      C
150      C *1* PRINT TRIGGER
151      NTRG1=-1
152      IF(NMAX.EQ.0)NTRG1=1
153      TV1=AIN(T)+PRINI(JUMP)
154      C
155      C *2* THRUST EVENT TRIGGER
156      NTRG2=2
157      NTRG7=-2
158      TV2=TIME(1,JUMP+1)
159      C
160      C *3* WEIGHT DROP EVENT TRIGGER
161      C      ALSO CONTROLS END OF CHI-FREEZE
162      NTRG3=-2
163      IF(IWDJ.LE.0)GO TO 6
164      NTRG3=2
165      TV3=TIME(2,IWDJ)
166      C
167      C *4* TILT TRIGGER
168      6 NTRG4=2
169      TV4=TLIFT
170      C

```

AFORUN

LIFTING-ROBOT LISTINGS

```

171 C *5* ALTITUDE IGER
172     NTRG5=1
173     TV5=ALT1
174     IF(NMAX.NE.0)NTRG5=-1
175 C
176 C *6* MAX Q TRIGGER
177     NTRG6=-1
178     TV6=0.
179 C
180 C
181 C     INITIALIZE THRUST,STEP,MDOT,ETC.
182     ITHR=JUMP-1
183     CALL ATHREV
184     IF(JUMP.EQ.1)GO TO 7
185     IF(JUMP.GT.NOEVNT(1))GO TO 8
186     IF(NMAX.EQ.0)NTRG6=1
187     NTRG4=-2
188     KS1=4
189     KLIFT=1
190     TQ=TZERO
191 C
192 C     RELATIVE INTEGRATION
193 7 IORR=1
194     IF(NMAX.EQ.0)NF8=13
195 C
196     CALL ADER1
197     CALL ADER
198     CALL APRTN(2)
199 C
200     IF(LSB.EQ.1)GO TO 110
201 C
202     NE=6
203     IF(NBGCT(1).LE.NOEVNT(1))NE=2
204     IF(NMAX.EQ.0)NE=6
205     CALL DESOLV(15,NF8,ADER1,ADER,AEOSR,NE,KERR,HNM,HBANK,FSAVE,
206 1 KIND,LLDSOL,EU,EL,HMX,HMN,AYL,NOTRG,
207 2 NTRG1,AXPRT,T,TV1, NTRG2,ATHREV, T,TV2,
208 * NTRG7,ATHREV,XM1,TV7,
209 3 NTRG3,AWDEV,T,TV3, NTRG4, ATILT,T,TV4,
210 4 NTRG5,AKALT,ALT,TV5,NTRG6,AQMAX,QDOT,TV6,
211 A 0,0,0,0, 0,0,0,0, 0,0,0,0, 0,0,0,0)
212     IF(KLIFT.EQ.1)GO TO 111
213     DELX(1)=W
214     DELX(2)=U
215     DELX(3)=V
216     DELX(4)=X
217     DELX(5)=Y
218     DELX(6)=Z
219     DELX(7)=XM1
220     DELX(8)=ZAP(1)
221     GO TO 111
222 C
223 110 CALL DESOLV(15,NF8,ADER1,ADER,0,6,KERR,HNM,HBANK,FSAVE,
224 1 KIND,LLDSOL,EU,EL,HMX,HMN,AYL,4,
225 2 NTRG2,ATHREV,T,TV2, NTRG3,AWDEV,T,TV3,
226 * NTRG7,ATHREV,XM1,TV7,
227 3 NTRG4,ATILT,T,TV4, 0,0,0,0, 0,0,0,0, 0,0,0,0,

```

AFORUN LIFTING-ROBOT LISTINGS

```

228      4 0,0,0,0, 0,0,0,0, 0,0,0,0, 0,0,0,0)
229  C
230 111 IF(KERR.NE.0)GO TO 11
231  C
232  C      INERTIAL INTEGRATION
233      8 IF(LSB.EQ.1 .AND. NALTW.EQ.0)RETURN
234      IORR=0
235      Q=0.
236      FAA=0.
237      FAN=0.
238      XMACH=0.
239      IF(NMAX.EQ.0) NF8=11
240      IF(JUMP.LE.NOEVNT(1))GO TO 10
241  C
242      DO 9 I=1,4
243      IF(NP(I).EQ.0)GO TO 9
244      IF(JUMP.GE.NENDCT(I))KCYTAB=I+1
245      IF(KCYTAB.GT.4)KCYTAB=4
246      9 CONTINUE
247      CALL ADER1I
248      CALL ADERI
249      CALL APRTN(3)
250  C
251 10 IF(LSB.EQ.1)GO TO 112
252  C
253      CALL DESOLV( 15, NF8,ADER1I,ADERI,AEOSR,2,KERR,HNM,HBANK,FSAVE,
254      1 KIND,LLDSOL,EU,EL,HMX,HMN,AYL,4,
255      2 NTRG1,AXPRT,T,TV1,NTRG2,ATHREV, T,TV2,
256      * NTRG7,ATHREV,XM1,TV7,
257      3 NTRG3,AWDEV,T,TV3,
258      A 0,0,0,0, 0,0,0,0, 0,0,0,0, 0,0,0,0,
259      A 0,0,0,0, 0,0,0,0, 0,0,0,0)
260      GO TO 113
261  C
262 112 CALL DESOLV(15,NF8,ADER1I,ADERI,0,6,KERR,HNM,HBANK,FSAVE,
263      1 KIND,LLDSOL,EU,EL,HMX,HMN,AYL,3,
264      2 NTRG2,ATHREV,T,TV2, NTRG3,AWDEV,T,TV3,
265      * NTRG7,ATHREV,XM1,TV7,
266      3 NTRG4,ATILT,T,TV5, 0,0,0,0, 0,0,0,0, 0,0,0,0,
267      4 0,0,0,0, 0,0,0,0, 0,0,0,0, 0,0,0,0)
268  C
269 113 IF(KERR.EQ.0)RETURN
270  C
271      11 WRITE(6,12)
272      12 FORMAT(/1X,24HINTEGRATION ERROR.DUMPED)
273      STOP
274      END

```

.L,0.

AGEO

LIFTING-ROBOT LISTINGS

T, S TPF\$.AGEO
PUR 0018-05/17-11:50

AGEO

LIFTING-ROBOT LISTINGS

DT*TPFS,AGEO

```

1  SUBROUTINE AGE0(N)
2  COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
3  COMMON/GRAV/GT0,G11,G22,G23,G33
4  COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CH1Y,
5  1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPH1Z,GR,GTH,
6  2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
7  3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
8  4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
9  5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),B0(3,3),D(3,3),DQDX(6),CASE,
10 6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
11 7 BHMAX(15),BHMN,BSTEP(15)
12 C NEEDS CTH,CT2, RE,R,CMUE,J,H,D,R2
13 C
14 C GET OBLATE EARTH GRAVITY TERMS
15 RER=RE/R
16 RER2=RER*RER
17 CJR=CJ*RER2
18 HR=H*RER2*RER
19 DJR=DJ*RER2*RER2
20 CMT=CMUE/(R2*R)
21 G11=-CMT*(1.+CJR*(1.-5.*CT2)+HR*(3.-7.*CT2)*CTH
22 1 +DJR*(.42857143-(6.-9.*CT2)*CT2))
23 IF(N.NE.1)GO TO 1
24 GT0=R*CMT*(2.*CJR*CTH-HR*(.6-3.*CT2)+DJR*(1.7142857-4.*CT2)*CTH)
25 RETURN
26 1 G22=-(3./R2)*(G11-CMT*(CJR*(.66666667-6.66666667*CT2)
27 1 +HR*(4.-14.*CT2)*CTH+DJR*(.57142857-(12.-24.*CT2)*CT2))
28 G23=(CMT/R)*(10.*CJR*CTH-HR*(3.-21.*CT2)+DJR*(12.-36.*CT2)*CTH)
29 G33=-CMT*(2.*CJR+6.*HR*CTH +DJR*(1.7142857-12.*CT2))
30 RETURN
31 END

```

3 .L,0.

AKALT

LIFTING-ROBOT LISTINGS

T,S TPF\$.AKALT
POR 0018-05/17-11:50

AKALT

LIFTING-ROBOT LISTINGS

OT*TPFS.AKALT¹

```

1  SUBROUTINE AKALT
2  COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
3  COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
4  1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
5  2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
6  3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
7  4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
8  5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
9  6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
10 7 BHMAX(15),BHMN,BSTEP(15)
11 COMMON/AGEN3/IORK,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
12 1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
13 2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
14 COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
15 1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
16 2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
17 IF(ALT.GE.(ALT1+10.))GO TO 1
18 C
19 C 10KILOMETER ALTITUDE
20 TV5=ALT2
21 CALL APRTN(12)
22 RETURN
23 C
24 C 14 KILOMETER ALTITUDE
25 1 NTRG5=-1
26 C
27 CALL APRTN(13)
28 RETURN
29 END

```

.L.O.

AMULG

LIFTING-ROBOT LISTINGS

T.S TPF\$.AMULG
PUR 0018-05/17-11:50

AMULG

LIFTING-ROBOT LISTINGS

DT*TPF\$.AMULG

```
1  SUBROUTINE AMULG(L,M,N,X,XT,Y1,Y1T,Y2,Y2T)
2  DIMENSION XT(1),Y1T(1),Y2T(1)
3  DIFF=X-XT(M)
4  IF(DIFF)1,3,2
5      C  GET HERE IF X.LT.XT(M)
6      1 IF(M.EQ.1)GO TO 4
7      M=M-1
8      IF(X-XT(M))1,3,4
9      C  GET HERE IF X.GT.XT(M)
10     2 IF(M.EQ.N)GO TO 5
11     MP1=M+1
12     IF(XT(MP1).LT.XT(M))GO TO 5
13     M=MP1
14     IF(X-XT(M))5,3,2
15     3 Y1=Y1T(M)
16     IF(L.NE.1)Y2=Y2T(M)
17     RETURN
18     4 M=M+1
19     5 MM1=M-1
20     DEL=(X-XT(MM1))/(XT(M)-XT(MM1))
21     Y1=Y1T(MM1)+DEL*(Y1T(M)-Y1T(MM1))
22     IF(L.NE.1)Y2=Y2T(MM1)+DEL*(Y2T(M)-Y2T(MM1))
23     RETURN
24     END
```

3 .L,0.

ANEWCH

LIFTING-ROBOT LISTINGS

1,5 TPF\$.ANEWCH
PUR 0018-05/17-11:50

ANEWCH

LIFTING-ROBOT LISTINGS

01*TPF\$.ANEWCH

```

1  SUBROUTINE ANEWCH
2  COMMON/FLORTE/FLORTE(15),DLFLRT(15)
3  COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
4  COMMON/FSCY/NFSCY,FSCYD,TFSCY
5  COMMON/PREDP/PREDP
6  COMMON/PERF/ILAST,IPR,PRK1,PRK2,PRK3,PRK4,PRTOT,FGR,FPR,WPMX
7  COMMON/BLAMB/YLBT(15,10),YLBW(15,10),GAPHI(10),PROD(2,10),WAH
8  COMMON/YLI/YLINT(7,10),PHITE(10),PSIREQ(10),KCDPHI(10),PSTR(10)
9  COMMON/BAKE/WISS(10,10),YL(7,10)
10 COMMON/FORINT/HBANK(2),T,TT,VAR(15),DVAR(15),
11 1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
12 2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
13 COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
14 COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
15 1 TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
16 2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
17 3 HNOM(15),HMAX(15)
18 COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
19 1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
20 2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
21 3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
22 4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
23 5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),B0(3,3),D(3,3),DQDX(6),CASE,
24 6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
25 7 BHMAX(15),BHMN,BSTEP(15)
26 COMMON/AGEN3/IORK,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCSI,
27 1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
28 2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
29 COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
30 COMMON/AUTO/KDB(30),SAVCP(30),XLAMB(30,10),DP2,KDT(30)
31 COMMON/CONTO/TTBL(201),CPTBL(201),CYTBL(201)
32 COMMON/IMP/GAPT(200,10),GSPT(200,10),AHTB(200),S1TB(200),TS(4),
33 1 DELTS(4),KJTB
34 COMMON/QY/QY,CFR(15)
35 COMMON/DRHO/DRHO(9),E1,WIPPB,WGNU
36 COMMON/CONVK/CONVK(17,3)
37 COMMON/LPRINT/LPRINT
38 DIMENSION SWIBT(17),DBETA(17),WAPS(9),WASS(10,10),WDS(9)
39 DIMENSION HUMAX(2),WIPSB(9),WISSB(9,9),XLBGNU(9),XLBWDS(9)
40 LOGICAL BETCON
41 C*****
42 C*****
43 C*****
44 C BEGIN MAIN NEWCH
45 C*****
46 88 XKAY=1.
47 CONV=1.-.5*E1
48 CASE=CASE+.0001
49 BETCON=.TRUE.
50 LSB=0
51 C*****
52 C ADJUST LAMBDA BETA MATRIX
53 C ** FLX LAUNCH WT COEF IF UPPER STAGES STAGE ON FUEL
54 DO 90 I=1,NVNT
55 IF(MSWCH(I).GE.0)GO TO 90
56 DO 89 J=1,NOSS

```

ANewCH

LIFTING-ROBOT LISTINGS

```

57      89 XLAMB(16,J)=XLAMB(16,J)+XLAMB(I,J)*DLFLRT(I)
58      90 CONTINUE
59      C
60      C ** FIX STAGING TIME COEF IF UPPER STAGES STAGE ON FUEL
61          NVNT1=NVNT-1
62          DO 93 I=JUMP,NVNT1
63              IF(KDB(I).EQ.0)GO TO 93
64              JI=I+1
65              DO 92 J=JI,NVNT
66                  IF(MSWCH(J).GT.0)GO TO 92
67                  DO 91 JJ=1,NOSS
68                      91 XLAMB(I,JJ)=XLAMB(I,JJ)+XLAMB(J,JJ)*FLORTE(I)*DLFLRT(J)
69                      92 CONTINUE
70                      93 CONTINUE
71      C*****
72          NV=1
73          DO 12 I=1,20
74              IF(KDB(I).NE.1)GO TO 12
75              DO 11 J=1,NOSS
76                  11 XLAMB(NV,J)=XLAMB(I,J)
77                  NV=NV+1
78                  12 CONTINUE
79                  WRITE(6,97)
80                  97 FORMAT(/,1X,22HINFLUENCE COEFFICIENTS)
81                  DO 81 I=1,NOPAR
82                      81 WRITE(6,9)(XLAMB(I,J),J=1,NOSS)
83                      9 FORMAT(1X,7E15.8)
84      C*****
85      C*****
86      C CALCULATE I-SY-SY MATRIX
87          71 KK=NOSS-1
88              WIPP=WISS(1,1)
89              DO 16 I=1,9
90                  16 WAPS(I)=WISS(I+1,1)
91                  DO 17 I=2,9
92                      DO 17 J=2,I
93                          IM=I-1
94                          JM=J-1
95                          WASS(IM,JM)=WISS(I,J)
96                  17 WASS(JM,IM)=WASS(IM,JM)
97                  WRITE(6,18)
98                  18 FORMAT(1X,12H    1 PSI PSI)
99                  WRITE(6,21)WIPP
100                 DO 20 I=1,KK
101                     20 WRITE(6,21)WAPS(I),(WASS(I,J),J=1,KK)
102                     21 FORMAT(1X,8E15.8)
103      C*****GET WIBT
104      C*****
105          DO 14 I=1,NOPAR
106              SWIBT(I)=0.
107              DO 13 J=2,NOSS
108                  TEMP=XLAMB(I,J)*XLAMB(I,J)/WASS(J-1,J-1)
109                  13 SWIBT(I)=AMAX1(SWIBT(I),TEMP)
110                  14 SWIBT(I)=WIBT(I)/SWIBT(I)
111      C*****
112          IF(NOPAR.EQ.0)SWIBT(1)=0.
113      C*****GOT WIBT

```

ANEWCH

LIFTING-ROBOT LISTINGS

```

114      WRITE(6,98)
115      98 FORMAT(/,1X,5HPSUBI)
116      15 WRITE(6,21)(SWIBT(I),I=1,NOPAR)
117      WRITE(6,99)
118      99 FORMAT(/,1X,4HWIBT)
119      WRITE(6,21)(WIBT(I),I=1,NOPAR)
120      WIPPB=WIPP
121      DO 22 I=1,NOPAR
122      22 WIPPB=WIPPB+XLAMB(I,1)*SWIBT(I)*XLAMB(I,1)
123      DO 23 I=1,KK
124      WIPSB(I)=WAPS(I)
125      DO 23 J=1,NOPAR
126      23 WIPSB(I)=WIPSB(I)+XLAMB(J,1)*SWIBT(J)*XLAMB(J,I+1)
127      DO 24 I=1,KK
128      DO 24 J=1,KK
129      WISSB(I,J)=WASS(I,J)
130      DO 24 K=1,NOPAR
131      24 WISSB(I,J)=WISSB(I,J)+XLAMB(K,I+1)*SWIBT(K)*XLAMB(K,J+1)
132      C PRINT TOTAL I MATRIX
133      WRITE(6,25)
134      25 FORMAT(1X,23H TOTAL I SY SY MATRIX)
135      WRITE(6,21)WIPPB
136      DO 251 I=1,KK
137      251 WRITE(6,21) WIPSB(I),(WISSB(I,J),J=1,KK)
138      IF(KK.GT.1)GO TO 26
139      WISSB(1,1)=1./WISSB(1,1)
140      GO TO 271
141      26 CALL MATINV(WISSB(1,1),KK)
142      DO 2005 I=1,KK
143      DO 2005 J=1,I
144      WISSB(I,J)=(WISSB(I,J)+WISSB(J,I))/2.
145      2005 WISSB(J,I)=WISSB(I,J)
146      C*****
147      C CALCULATE MATRIX PRODUCTS
148      C*****
149      271 DO 27 I=1,KK
150      27 DRHO(I)=-PSTR(I+1)
151      DO 28 I=1,KK
152      WDS(I)=0.
153      DO 28 J=1,KK
154      28 WDS(I)=WDS(I)+WISSB(I,J)*DRHO(J)
155      DO 30 I=1,KK
156      GNU(I)=0.
157      DO 30 J=1,KK
158      30 GNU(I)=GNU(I)-WISSB(I,J)*WIPSB(J)
159      WGNU=0.
160      DO 31 I=1,KK
161      31 WGNU=WGNU-GNU(I)*WIPSB(I)
162      WRITE(6,1000)WGNU
163      1000 FORMAT(1X,4HWGNU,E15.8)
164      C*****
165      C BEGIN E1 LOGIC
166      C*****
167      NOM2=NOM1+1
168      IF(NOM2.GE.5)GO TO 325
169      GO TO(321,322,323,324),NOM2
170      321 E1=0.

```

ANewCH

LIFTING-ROBOT LISTINGS

```

171      XKAY=AMAX1(DP2,.25)
172      IF(XKAY.LT..4999)GO TO 34
173      IF(XKAY.GT..9999)GO TO 32
174      NOM1=1
175      XKAY=.5
176      GO TO 34
177      32 NOM1=2
178      GO TO 36
179      322 E1=0.
180      XKAY=.5
181      GO TO 34
182      323 E1=0.
183      GO TO 36
184      324 E1=QY/2.
185      GO TO 36
186      325 E1=QY
187      GO TO 36
188      34 DO 35 I=1,KK
189      DRHO(I)=DRHO(I)*XKAY
190      35 WDS(I)=XKAY*WDS(I)
191 C*****
192 C      CALCULATE DBETA
193 C*****
194      36 IF(NOPAR.EQ.0)GO TO 50
195      DO 37 I=1,NOPAR
196      CONVK(I,3)=ABS(XLAMB(I,1))
197      XLBGNU(I)=0.
198      XLBWDS(I)=0.
199      DO 37 J=1,KK
200      TEMP=XLAMB(I,J+1)*GNU(J)
201      XLBGNU(I)=XLBGNU(I)+TEMP
202      XLBWDS(I)=XLBWDS(I)+XLAMB(I,J+1)*WDS(J)
203      TEMP=TEMP+XJEXT*XLBWDS(I)
204      37 CONVK(I,3)=AMAX1(CONVK(I,3),ABS(TEMP))
205      WRITE(6,1001)XKAY,E1
206 1001 FORMAT( 1X,4HXKAY,E15.8,3H E1,E15.8)
207      WRITE(6,450)
208      450 FORMAT( 1X,12HDKHO,WDS,GNU)
209      DO 470 I=1,KK
210      470 WRITE(6,21)DRHO(I),WDS(I),GNU(I)
211      DO 40 I=1,NOPAR
212      TEMP=XLAMB(I,1)+XLBGNU(I)
213      CONVK(I,2)=TEMP/CONVK(I,3)
214      40 DBETA(I)=SWIBT(I)*(      TEMP      *E1*XJEXT+XLBWDS(I))
215      WRITE(6,96)
216      96 FORMAT(/,7X,4HDPAR,9X,8H OLD PCON,7X,8HNEW PCON,8X,7HP SCALE)
217      DO 1002 I=1,NOPAR
218      IF(ABS(CONVK(I,2)).GT..005)BETCON=.FALSE.
219 1002 WRITE(6,21)DBETA(I),(CONVK(I,J),J=1,3)
220 C*****
221 C      ADD DBETA TO BETA
222 C*****
223      NV=1
224      DO 45 I=1,20
225      IF(KDB(I).EQ.0)GO TO 45
226      IF(I.GT.15)GO TO 43
227      41 TAUT(I )=TAUT(I )+DBETA(NV)

```

ANewCH

LIFTING-ROBOT LISTINGS

```

228      FUEL(I)=FUEL(I)-FLORTE(I)*DBETA(NV)
229      C
230      C      IF(I.LT.NVNT)TAUT(I+1)=TAUT(I+1)-DBETA(NV)
231      C
232      IF(KDT(I).EQ.0)GO TO 42
233      J=I +KDT(I)
234      TEMP=(TNE(1,I )+TNE(3,I ))*CFR(I )/(CFR(J)*(TNE(1,J)+TNE(3,J)))
235      TAUT(J)=TAUT(J)-TEMP*DBETA(NV)
236      IF(TAUT(J).LE..01)TAUT(J)=.01
237      42 IF(ABS(DBETA(NV)).GT..5)BETCON=.FALSE.
238      IF(TAUT(I ).LE..01)TAUT(I )=.01
239      GO TO 44
240      43 I2=I-15
241      GO TO(141,142,143,144,145),I2
242      141 WZERO=WZERO+DBETA(NV)
243      IF(ABS(DBETA(NV)).GT.100.)BETCON=.FALSE.
244      GO TO 44
245      142 CHIDOT=CHIDOT+DBETA(NV)
246      IF(ABS(DBETA(NV)).GT..00002)BETCON=.FALSE.
247      GO TO 44
248      143 AA=AA+DBETA(NV)
249      IF(ABS(DBETA(NV)).GT..00085)BETCON=.FALSE.
250      144 FSCYD=FSCYD+DBETA(NV)
251      IF(ABS(DBETA(NV)).GT..001)BETCON=.FALSE.
252      145 GO TO 44
253      44 NV=NV+1
254      45 CONTINUE
255      C*****
256      C      CALCULATE CHANGE IN CHIP AND CHIY
257      C*****
258      50 DELCY=0.
259      HUMAX(1)=0.
260      HUMAX(2)=0.
261      KV=1
262      WRITE(6,95)
263      95 FORMAT(/,7X,4HTIME,9X,6HNEW CP,9X,6HOLD CP,8X,6HNEW CY,8X,
264      1 6HOLD CY,10X,3HDCP,10X,3HDCY)
265      2020 DO 60 II=1,4
266      IF(NP(KV).EQ.0)GO TO 59
267      IF(NBGCT(II).EQ.1)GO TO 59
268      KJTB=(KV-1)*50
269      T=TS(KV)
270      DTB4=DELTS(KV)
271      JTB=NP(KV)
272      JA=NOSS
273      IF(T+1. .GT.TR)JA=NOSS-M CNRS
274      DO 46 I=1,JTB
275      JJ=KJTB+I
276      TEMP=0.
277      TEMP1=0.
278      DO 47 J=2,JA
279      TEMP=TEMP+GAPTb(JJ,J)*GNU(J-1)
280      47 TEMP1=TEMP1+GAPTb(JJ,J)*WDS(J-1)
281      DELCP=(GAPTb(JJ,1)+TEMP)*XJEXT
282      HUMAX(1)=AMAX1(ABS(DELCP+TEMP1),HUMAX(1))
283      DELCP=DELCP+E1+TEMP1
284      C***      END OF CHIP

```

ANewCH

LIFTING-ROBOT LISTINGS

```

285      IF(KWTA.EQ.2)GO TO 57
286      TEMP=0.
287      TEMP1=0.
288      DO 48 J=2,JA
289      TEMP=TEMP+GSPTB(JJ,J)*GNU(J-1)
290      48 TEMP1=TEMP1+GSPTB(JJ,J)*WDS(J-1)
291      DELCY=(GSPTB(JJ,1)+TEMP)*XJEXT
292      HUMAX(2)=AMAX1(ABS(DELGY+TEMP1),HUMAX(2))
293      DELCY=DELCY*E1+TEMP1
294      C*** END OF CHIY
295      C UPDATE TTBL,CPTBL,CYTBL
296      57 TTBL(JJ)=T
297      CPTBL(JJ)=AHTB(JJ)+DELCY
298      CYTBL(JJ)=SITB(JJ)+DELCY
299      IF(CYTBL(JJ).GT.1.5)CYTBL(JJ)=(1.5+SITB(JJ))/2.
300      IF(CYTBL(JJ).LT.-1.5)CYTBL(JJ)=-((1.5-SITB(JJ))/2.
301      IF(LPRINT.EQ.0) GO TO 75
302      IF(I.EQ.1.OR.I.EQ.JTB) GO TO 75
303      GO TO 76
304      75 WRITE(6,49)T,CPTBL(JJ),AHTB(JJ),CYTBL(JJ),SITB(JJ),DELCY,
305      1 DELCY,JJ
306      49 FORMAT(1X,7E14.7,I3)
307      76 CONTINUE
308      46 T=T+DTB4
309      59 KV=KV+1
310      60 CONTINUE
311      WRITE(6,61)HUMAX(1),HUMAX(2),BETCON,KAT
312      61 FORMAT(1X,13HDEL CHIP MAX=,E11.4,16H DEL CHIY MAX=,E11.4,
313      1 10H BETCON=,L3,6H KAT=,I3)
314      C*****RESTORE NOSS FOR FORWARD RUN
315      NOSS=NOSS-NCNRS
316      NOM1=NOM1+1
317      IF(NOM1.LT.5)BETCON=.FALSE.
318      IF(HUMAX(1).LT..005.AND.HUMAX(2).LT..005.AND.BETCON)NMAX=1
319      C*****UPDATE CONVERGENCE SCALE FACTORS*****
320      IF(NOM1.LE.5)GO TO 64
321      DO 63 I=1,NOPAR
322      IF(ABS(CONVK(I,2)).LT..005)GO TO 63
323      IF(ABS(CONVK(I,2)).LT.CONVP*ABS(CONVK(I,1)))GO TO 63
324      IF(ABS(CONVK(I,2)-CONVK(I,1)).LT.ABS(CONVK(I,2)))GO TO 62
325      WIBT(I)=WIBT(I)/2.
326      NOM1=4
327      GO TO 63
328      62 WIBT(I)=WIBT(I)*2.
329      NOM1=4
330      63 CONTINUE
331      64 DO 65 I=1,NOPAR
332      65 CONVK(I,1)=CONVK(I,2)
333      PREDP=0.
334      DO 66 I=1,KK
335      PREDP=PREDP+GNU(I)*DRHO(I)
336      IF(NOM1.GT.2)GNU(I)=GNU(I)+XJEXT*WDS(I)
337      66 CONTINUE
338      RETURN
339      END

```

ANEWCH

LIFTING-ROBOT LISTINGS

3 .L.O.

9 APPG

LIFTING-ROBOT LISTINGS

T.S TPF\$.APPG
PUR 0018-05/17-11:50

APPG

LIFTING-ROBOT LISTINGS

DT*TPFS,APPG

```

1      SUBROUTINE APPG(XF,RA,RP,PSAP,PSPG)
2      COMMON/DFEQS/N,XN,X,XH,II,YN(6),Y(6),Y1(6),DY(6)
3      DIMENSION PSAP(6),PSPG(6),XF(6),ST(3,401),      YL(6),YS(3,3)
4      C
5      C      THIS SUBROUTINE TREATS THE APOGEE AND PERIGE CONSTRAINTS. THE
6      C
7      C      INPUTS ARE THE STATE VECTOR AT TF (W,U,V,X,Y,Z) , THE DESIRED
8      C
9      C      APOGEE RA, AND THE DESIRED PERIGEE RP. THE OUTPUTS ARE SA AND SP -
10     C
11     C      THE AMOUNTS BY WHICH THE COMPUTED APOGEE AND PERIGEE, RESPECTIVELY
12     C
13     C      , EXCEED THE DESIRED VALUES - AND THE ASSOCIATED BOUNDARY
14     C
15     C      CONDITIONS PSAP AND PSPG ON THE ADJOINT VARIABLES.
16     C
17     C      THIS SUBROUTINE MAKES USE OF THE FOLLOWING SUBROUTINES -
18     C
19     C      SYSDER,ADDR,FORKM,AGEO.
20     C      SYSDER SPECIFIES THE SYSTEM EQUATIONS. ADDR SPECIFIES THE ADJOINT
21     C
22     C      EQUATIONS, AND FORKM IS A FOURTH ORDER RUNGE KUTTA DIFFERENTIAL
23     C
24     C      EQUATION SOLVER
25     C
26     N = 6
27     DT = 20.
28     TEST = .01
29     ST(1,1) = XF(4)
30     ST(2,1) = XF(5)
31     ST(3,1) = XF(6)
32     C
33     C      FIRST COMPUTE FIRST EXTREMUM OF ORBIT BY FORWARD INTEGRATION
34     C
35     1 DO 2 I = 1,6
36       YL(I) = XF(I)
37     2 YN(I) = XF(I)
38     XN = 0.
39     C
40     C      STOPPING CODITION IS WHEN OMG = 0., OR R AND RDOT ARE ORTHOGONAL.
41     C
42     OMG = XF(1)*XF(4) + XF(2)*XF(5) + XF(3)*XF(6)
43     IF(ABS(OMG).GT.TEST) GO TO 3
44     RCT1 = SQRT(XF(4)*XF(4) + XF(5)*XF(5) + XF(6)*XF(6))
45     ICK = 0
46     PSAP(1) = 0.
47     PSAP(2) = 0.
48     PSAP(3) = 0.
49     PSAP(4) = XF(4)/RCT1
50     PSAP(5) = XF(5)/RCT1
51     PSAP(6) = XF(6)/RCT1
52     GO TO 30
53     3 I = 1
54     XH = DT
55     5 CALL SYSDER
56     C

```

APP6

LIFTING-ROBOT LISTINGS

```

57 C CHECK TO SEE IF STOPPING CONDITION IS SATISFIED. IF NOT KEEP GOING
58 C
59 6 OMN = YN(1)*YN(4)+YN(2)*YN(5)+YN(3)*YN(6)
60 IF (OMG*OMN.LT.0.) GO TO 8
61 I = I + 1
62 ST(1,I) = YN(4)
63 ST(2,I) = YN(5)
64 ST(3,I) = YN(6)
65 OMG = OMN
66 DO 4 J = 1,6
67 4 YL(J) = YN(J)
68 GO TO 5
69 C
70 C ZERO IN ON EXTEMUM
71 8 IF (ABS(OMN).GT.TEST) GO TO 9
72 DH = DT
73 I = I + 1
74 DO 800 J = 1,3
75 K = J + 3
76 800 ST(J,I) = YN(K)
77 GO TO 12
78 9 XB = XN - DT
79 TS = (XB*OMN - XN*OMG)/(OMN - OMG)
80 DH = .5*(TS - XB)
81 XH = DH
82 DO 900 K = 1,6
83 900 YN(K) = YL(K)
84 XN = XB
85 DO 902 L = 1,2
86 I = I + 1
87 10 CALL SYSDER
88 DO 902 LL = 1,3
89 KK = LL + 3
90 902 ST(LL,I) = YN(KK)
91 11 OMG = YN(1)*YN(4)+YN(2)*YN(5)+YN(3)*YN(6)
92 12 ICK = 1
93 RCT1 = SQRT(YN(4)*YN(4) + YN(5)*YN(5) + YN(6)*YN(6))
94 C
95 C NOW BEGIN BACKWARD INTEGRATION OF THE ADJOINT EQUATION
96 C
97 C
98 C FIRST CHECK TO SEE IF THE STORED SOLUTION IS COMPLETE.
99 C
100 C THE INTEGRATION OF THE ADJOINT EQUATIONS USES AN INTEGRATION STEP
101 C
102 C SIZE TWICE THAT USED TO INTEGRATE THE SYSTEM EQUATIONS. HENCE I
103 C
104 C THE TOTAL NUMBER OF SOLUTIONS STORED MUST BE AN ODD NUMBER.
105 C
106 C IF NOT ADD ANOTHER SOLUTION AT T = H/2.
107 C
108 IF (I.LT.400) GO TO 129
109 WRITE(6,100) DT
110 100 FORMAT(///50X,'ERROR IN APP6',//20X,'INTEGRATION STEP SIZE ',
111 1F10.5,' IS TOO SMALL')
112 DT = 2.*DT
113 GO TO 1

```

APP6

LIFTING-ROBOT LISTINGS

```

114 129 I1 = I/2
115      I2 = 2*I1
116      IF(I2.EQ.I) GO TO 130
117      HL = DT
118      GO TO 13
119 130 HL =DT*.5
120      XH = HL
121      DO 131 L = 1,6
122 131 YN(L) = XF(L)
123      XN = 0.
124 132 CALL SYSDER
125 133 DO 134 J = 2,I
126      K1 = I - J + 2
127      K2 = K1 + 1
128      DO 134 L = 1,3
129 134 ST(L,K2) = ST(L,K1)
130      DO 135 L = 1,3
131      K = L + 3
132 135 ST(L,2) = YN(K)
133      I = I + 1
134 C
135 C      PERFORM FIRST STEP OF BACKWARD INTEGRATION OF ADJOINT EQUATIONS
136 C
137 13 YN(1) = 0.
138      YN(2) = 0.
139      YN(3) = 0.
140      YN(4) = ST(1,I)/RCT1
141      YN(5) = ST(2,I)/RCT1
142      YN(6) = ST(3,I)/RCT1
143      XH = 2.*DH
144      XN = 0.
145      DO 14 JJ = 1,3
146      KK = I - JJ +1
147      DO 14 LL = 1,3
148 14 YS(LL,JJ) = ST(LL,KK)
149 15 CALL ADDR(1,YS)
150 C
151 C      SO MUCH FOR THE FIRST STEP. NOW LETS INTEGRATE BACK TO TF - 2HL
152 C
153      IF(I.LT.4) GO TO 285
154      IF(I.EQ.5) GO TO 275
155 16 TB = TS - DT
156      XH = 2.*DT
157      NT = KK/2 - 1
158      DO 27 IJ = 1,NT
159      N1 = I - 2*IJ
160      N2 = N1 - 1
161      NL = N2 - 1
162 17 DO 18 IK = 1,3
163      YS(IK,1) = ST(IK,N1)
164      YS(IK,2) = ST(IK,N2)
165 18 YS(IK,3) = ST(IK,NL)
166 19 CALL ADDR(1,YS)
167      T = TS - XN
168 27 CONTINUE
169 C
170 C      NOW INTEGRATE OVER THE LAST INTERVAL

```

```

171 C
172 275 XH = 2.*HL
173 28 DO 26 IJ = 1,3
174 JJ = 4 - IJ
175 DO 26 IK = 1,3
176 26 YS(IK,IJ) = ST(IK,JJ)
177 25 CALL ADDR(1,YS)
178 285 T = TS - XN
179 DO 29 I = 1,6
180 29 PSAP(I) = YN(I)
181 C
182 C NEXT INTEGRATE BACKWARDS FROM TF TO FIND THE OTHER EXTREMUM
183 C
184 C FIRST STEP
185 C
186 30 DO 31 I = 1,3
187 J = I + 3
188 YN(I) = XF(I)*(-1.)
189 31 YN(J) = XF(J)
190 OMG = YN(1)*YN(4) + YN(2)*YN(5) + YN(3)*YN(6)
191 XN = 0.
192 XH = DT
193 32 CALL SYSDER
194 33 OMN = YN(1)*YN(4) + YN(2)*YN(5) + YN(3)*YN(6)
195 IF(OMN*OMG.GT.0.) GO TO 37
196 IF(ICK.NE.0) GO TO 36
197 RA = RCT1
198 DO 34 I = 1,6
199 34 PSPG(I) = PSAP(I)
200 RP = RCT1
201 RETURN
202 36 I = 1
203 DO 360 J = 1,3
204 K = J + 3
205 YL(J) = -XF(J)
206 360 YL(K) = XF(K)
207 GO TO 43
208 37 OMG = OMN
209 DO 370 J = 1,6
210 370 YL(J) = YN(J)
211 DO 38 J = 1,3
212 K = J + 3
213 38 ST(J,2) = YN(K)
214 I = 2
215 39 CALL SYSDER
216 40 OMN = YN(1)*YN(4) + YN(2)*YN(5) + YN(3)*YN(6)
217 C
218 C TEST TO SEE IF STOPPING CONDITION HAS BEEN PASSED.
219 C
220 IF(OMN*OMG.LT.0.) GO TO 42
221 I = I + 1
222 OMG = OMN
223 DO 401 J = 1,6
224 401 YL(J) = YN(J)
225 DO 41 J = 1,3
226 K = J + 3
227 41 ST(J,I) = YN(K)

```

APPG

LIFTING-ROBOT LISTINGS

```

228      GO TO 39
229  C
230  C
231  C      IF STOPPING CONDITION HAS BEEN PASSED, ZERO IN ON FINAL TIME
232  42  IF (ABS(OMN).GT.TEST) GO TO 43
233      DH = DT
234      RCT2 = SQRT(YN(4)*YN(4) + YN(5)*YN(5) + YN(6)*YN(6))
235      GO TO 606
236  43  XB = XN - DT
237      TS = (XB*OMN - XN*OMG)/(OMN - OMG)
238      DH = (TS - XB)*.5
239      XH = DH
240      DO 430 J = 1,6
241  430  YN(J) = YL(J)
242      XN = XB
243      DO 451 M = 1,2
244      I = I + 1
245  44  CALL SYSDER
246  45  OMG = YN(1)*YN(4) + YN(2)*YN(5) + YN(3)*YN(6)
247      DO 451 L = 1,3
248      K = L + 3
249  451  ST(L,I) = YN(K)
250  452  RCT2 = SQRT(YN(4)*YN(4) + YN(5)*YN(5) + YN(6)*YN(6))
251  C
252  C      NOW BEGIN FORWARD INTEGRATION OF THE ADJOINT EQUATION
253  C
254  C
255  C      CHECK TO SEE IF SOLUTION IS COMPLETE.
256  C
257      IF(I.LT.400) GO TO 459
258      WRITE(6,100) DT
259      DT = 2.*DT
260      GO TO 30
261  459  I1 = I/2
262      I2 = 2*I1
263      IF(I2.EQ.I) GO TO 460
264      HL = DT
265      GO TO 49
266  460  HL =DT*.5
267      DO 461 L = 1,3
268      K = L + 3
269      YN(L) = -XF(L)
270  461  YN(K) = XF(K)
271      XN = 0.
272      XH = HL
273  46  CALL SYSDER
274  463  DO 464 J = 2,1
275      K1 = I - J + 2
276      K2 = K1 + 1
277      DO 464 L = 1,3
278  464  ST(L,K2) = ST(L,K1)
279      DO 465 J = 1,3
280      K = J + 3
281  465  ST(J,2) = YN(K)
282      I = I + 1
283  C
284  C      NEXT DEFINE INITIAL CONDITIONS FOR INTEGRATION OF THE ADJOINT EQNS

```

APP6

LIFTING-ROBOT LISTINGS

```

285 C
286 49 YN(1) = 0.
287 YN(2) = 0.
288 YN(3) = 0.
289 YN(4) = ST(1,1)/RCT2
290 YN(5) = ST(2,1)/RCT2
291 YN(6) = ST(3,1)/RCT2
292 XH = 2.*DH
293 XN = 0.
294 C
295 C FIRST STEP
296 C
297 50 DO 51 J = 1,3
298 JJ = I - J + 1
299 DO 51 K = 1,3
300 51 YS(K,J) = ST(K,JJ)
301 52 CALL ADDR(2,YS)
302 54 TB = XN - TS
303 C
304 C SO MUCH FOR THE FIRST STEP NOW INTEGRATE UP TO TF - 2HL
305 C
306 IF(I.EQ.3) GO TO 606
307 IF(I.EQ.5) GO TO 575
308 NT = JJ/2 - 1
309 XH = 2.*DT
310 N3 = I
311 DO 60 I = 1,NT
312 N1 = N3 - 2*I
313 N2 = N1 - 1
314 NL = N2 - 1
315 55 DO 56 J = 1,3
316 YS(J,1) = ST(J,N1)
317 YS(J,2) = ST(J,N2)
318 56 YS(J,3) = ST(J,NL)
319 57 CALL ADDR(2,YS)
320 T = XN - TS
321 60 CONTINUE
322 C
323 C NOW INTEGRATE OVER THE LAST INTERVAL
324 C
325 575 XH = 2.*HL
326 58 DO 59 J = 1,3
327 K = 4 - J
328 DO 59 L = 1,3
329 59 YS(L,J) = ST(L,K)
330 61 CALL ADDR(2,YS)
331 607 T = XN - TS
332 C
333 C DETERMINE WHICH OF THE EXTREMA IS THE APOGEE AND WHICH THE PERIGEE
334 C
335 606 IF(RCT1.LT.RCT2) GO TO 63
336 DO 610 J = 1,6
337 610 PSPG(J) = YN(J)
338 RA = RCT1
339 RP = RCT2
340 RETURN
341 63 DO 64 J = 1,6

```

APP6

LIFTING-ROBOT LISTINGS

342
343
344
345
346
347

64 PSPG(J) = PSAP(J)
PSAP(J) = YN(J)
RA = RCT2
RP = RCT1
RETURN
END

5 .L.O.

5

APRTN

LIFTING-ROBOT LISTINGS

T.S IPF\$.APRTN
PUR 0018-05/17-11:50

APRTN

LIFTING-ROBOT LISTINGS

DT*TPF\$.APRTN

```

1      SUBROUTINE APRTN(NN)
2      COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
3      COMMON/ARODTA/ACF,BCF,CCF,DADM,DBDM,DCDM
4      COMMON/CONVEC/XCV,YCV,ZCV
5      COMMON/AFPRN/VI,GAM,C1,C3,XINC,XNOD,XMLB,THETG,THETA,PHI,
6      ARNGO,RNGAN,AZI
7      COMMON/ALONGO/ALONGO
8      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
9      1  TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
10     2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
11     3  HNOM(15),HMAX(15)
12     COMMON/CHIROL/TCHIR,FAZ,CHRDOT
13     COMMON/NTABLE/NTABLE
14     COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
15     COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
16     COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CH1Y,
17     1  SCH1Y,CCH1Y,TH2,STH,CTH,THGT,THETL,STHL,CTHL,DPH1Z,GR,GTH,
18     2  RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
19     3  VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
20     4  DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
21     5  FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),B0(3,3),D(3,3),DQDX(6),CASE,
22     6  CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
23     7  BHMAX(15),BHMN,BSTEP(15)
24     COMMON/AGEN3/IORK,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCS1,
25     1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
26     2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
27     COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
28     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
29     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
30     COMMON/XMIAD/XMIAD,DELMA
31     COMMON/LPRINT/LPRINT
32     COMMON/COUNT/ICOUNT,ICONT,LANGPR
33     DIMENSION RID(40)
34     DATA(RID(I),I=3,40)/6HLIFT=0,2HFF,6HIGNITI,2HON,6HBOOST,6HIMPACT,
35     16HTHRUST,6H EVENT,6HENG CU,4HTOFF,6HBEGIN,4HTILT,4HEND,4HTILT,
36     2  6HSEPARA,4HTION,6HSTART,5HCOAST,6HEND CO,3HAST,6H10 KMS,1H.,
37     26H14 KMS,1H.,
38     36HQ MAXI,3HMUM,6HINJECT,5HION,6HBEGIN,6HCHIFRZ,6HWEIGHT,
39     36H DROP,
40     46HBEGIN,5HMIN-H,6HEND AT,3HMOS,6HREIGNI,4HTION/
41     DATA(RID(I),I=1,2)/6H,6H/
42     IF(LSB.EQ.1)RETURN
43     IF(LPRINT.EQ.2.AND.NMAX.GT.0) RETURN
44     AHI=ZAP(5)
45     DAZ=ABS(FAZ-AA*RAD)
46     CHIR=DAZ
47     IF(T.GT.TCHIR)CHIR=DAZ-ABS(CHRDOT)*(T-TCHIR)
48     IF(CHIR.LT.0.)CHIR=0.
49     CHIR=CHIR*SIGN(1.,DAZ)
50     XM=XMIAD+XMAUG
51     D(1,2)=X/R
52     D(2,2)=Y/R
53     D(3,2)=Z/R
54     CTH=A(1,2)*D(1,2)+A(2,2)*D(2,2)+A(3,2)*D(3,2)
55     SPHI=A(1,1)*D(1,2)+A(2,1)*D(2,2)+A(3,1)*D(3,2)
56     CPHI=A(1,3)*D(1,2)+A(2,3)*D(2,2)+A(3,3)*D(3,2)

```

APRTN

LIFTING-ROBOT LISTINGS

```

57      PHI=ATAN2(SPHI,CPHI)
58      IF(T.LT..1)PHI=DTZ*OMEGA
59      XNOD=PHI
60      IF(PHI.LT.0.)PHI =PHI+2.*PI
61      PHI=(PHI-OMEGA*(T+DTZ))*RAD-ALONGO
62      VI=(W*W+U*U+V*V)**.5
63      GAM=(W*X+U*Y+V*Z)/(R*VI)
64      IF(GAM.LE.1..AND.GAM.GE.-1.)GO TO 11
65      IF(GAM.LE.1.)GO TO 12
66      GAM=PI/2.
67      GO TO 13
68      12 GAM=1.5*PI
69      GO TO 13
70      11 GAM=ASIN(GAM)
71      13 IF(GAM.GT.PI)GAM=GAM-2.*PI
72      C1=VI*R*COS(GAM)
73      GAM=GAM*RAD
74      C3=VI*VI-2.*CMUE/R
75      THE=THR*THROT
76      IF(IORR.EQ.0)GO TO 20
77      THE=THE-EXIT*PAM
78      CHIP=ATAN2(XCV,YCV)
79      ALFP=ATAN2(SW,SU)-CHIP
80      SALFY=SV/VR
81      VR2=VR*VR
82      QSOV2=.5*RHO*S(ITHR)
83      QSOV=QSOV2*VR
84      QS=QSOV*VR
85      Q=.5*RHO*VR2
86      VDOTC=SW*XCV+SU*YCV+SV*ZCV
87      TEMP=VR2-VDOTC*VDOTC
88      IF(TEMP.LT.0.)TEMP=0.
89      SINVR=SQRT(TEMP)
90      FAA=ACF*VDOTC*QSOV+BCF*TEMP*QSOV2+CCF*QS
91      FAN=ACF*SINVR*QSOV-BCF*SINVR*VDOTC*QSOV2
92      CALF=(SW*XCV+SU*YCV+SV*ZCV)/VR
93      ALF=ACOS(CALF)
94      IF(SALFY.LE.1..AND.SALFY.GE.-1.)GO TO 111
95      IF(SALFY.LE.1.)GO TO 112
96      ALFY=PI/2.
97      GO TO 113
98      112 ALFY=1.5*PI
99      GO TO 113
100     111 ALFY=ASIN(SALFY)
101     113 CHIY=ASIN(ZCV)
102     IF(CHIY.GT.PI)CHIY=CHIY-2.*PI
103     IF(SSIG.LE.1..AND.SSIG.GE.-1.)GO TO 141
104     IF(SSIG.LE.1.)GO TO 145
105     SIG=PI/2.
106     GO TO 143
107     145 SIG=1.5*PI
108     GO TO 143
109     141 SIG=ASIN(SSIG)
110     143 SDEG=SIG*RAD
111     20 CONTINUE
112     XLACC=(THE-FAA)/XM
113     CPDG=CHIP*RAD

```

```

114      CYDG=CHIY*RAD
115      IF(IORR.EQ.1)GO TO 30
116      SW=W-(Z*A22W-Y*A32W)
117      SU=U-(X*A32W-Z*A12W)
118      SV=V-(Y*A12W-X*A22W)
119      VR=(SW*SW+SU*SU+SV*SV)**.5
120      TEMP=(W*CCHIY*CHIP+U*CCHIY*CHIP+V*SCHIY)/VI
121      IF(TEMP.LE.1..AND.TEMP.GE.-1.)GO TO 121
122      IF(TEMP.LE.1.)GO TO 122
123      ALF=PI
124      GO TO 123
125      122 ALF=0.
126      GO TO 123
127      121 ALF=ACOS(TEMP)
128      123 ALFP=ATAN2(W,U)-CHIP
129      SALFY=(V/VI)
130      IF(SALFY.LE.1..AND.SALFY.GE.-1.)GO TO 131
131      IF(SALFY.LE.1.)GO TO 132
132      ALFY=PI/2.
133      GO TO 133
134      132 ALFY=1.5*PI
135      GO TO 133
136      131 ALFY=ASIN(SALFY)
137      133 TEMP=RE
138      IF(JORB.EQ.0)TEMP=UMF*RE/(UMF*UMF*(1.-CT2)+CT2)**.5
139      ALT=R-TEMP
140      30 ALTP=ALT
141      ADEG=ALF*RAD
142      ALFP=ALFP*RAD
143      ALFY=ALFY*RAD
144      THEP=THE/4.44822162
145      XMLB=XM/.45359237
146      TISP=F(ITHR)/(GZERO*XMD(ITHR))
147      STH=(1.-CTH*CTH)**.5
148      RSTH=R*STH
149      D(1,3)=(D(1,2)*CTH-A(1,2))/STH
150      D(2,3)=(D(2,2)*CTH-A(2,2))/STH
151      D(3,3)=(D(3,2)*CTH-A(3,2))/STH
152      D(1,1)=(-Y*A(3,2)+Z*A(2,2))/RSTH
153      D(2,1)=(X*A(3,2)-Z*A(1,2))/RSTH
154      D(3,1)=(-X*A(2,2)+Y*A(1,2))/RSTH
155      VTH=D(1,3)*W+D(2,3)*U+D(3,3)*V
156      VPH=D(1,1)*W+D(2,1)*U+D(3,1)*V
157      XI=-VTH/SQRT(VTH*VTH+VPH*VPH)
158      AZI=ACOS(XI)
159      THET=ACOS(CTH)
160      THETA=90.-THET*RAD
161      THETG=ATAN2(SIN(THETA/RAD),UMF*UMF*COS(THETA/RAD))*RAD
162      XINC=ACOS(SIN(AZI)*SIN(THET))
163      XNOD=(XNOD+ATAN2(CTH*VPH,VTH))*RAD
164      IF(XNOD.LT.0.)XNOD=XNOD+360.
165      XINC=XINC*RAD
166      A1=ABS(CMUE/C3)
167      P=C1*C1/CMUE
168      ECC=1.+P*C3/CMUE
169      IF(ECC.LT.0.)GO TO 31
170      ECC=SQRT(ECC)

```

APRTN

LIFTING-ROBOT LISTINGS

```

171      IF (RE-P/(1.+ECC).LT.RE/100.)GO TO 31
172      CETANW=(P-R)/(R*ECC)
173      ETANW=ABS(CETANW)
174      IF (ETANW.LE.1.)GO TO 140
175      ETANW=0.
176      GO TO 142
177 140  ETANW=ACOS(ETANW)
178 142  IF (CETANW.LT.0.)ETANW=PI-ETANW
179      IF (GAM.LT.0.)ETANW=2.*PI-ETANW
180      CETATN=(P-RE)/(RE*ECC)
181      ETATN=ABS(CETATN)
182      IF (ETATN.LE.1.)GO TO 150
183      ETATN=0.
184      GO TO 151
185 150  ETATN=ACOS(ETATN)
186 151  IF (CETATN.LT.0.)ETATN=PI-ETATN
187      DELETA=2.*PI-ETATN-ETANW
188      CBGENW=(A1-R)/(A1*ECC)
189      BGENOW=ABS(CBGENW)
190      IF (BGENOW.LE.1.)GO TO 152
191      BGENOW=0.
192      GO TO 153
193 152  BGENOW=ACOS(BGENOW)
194 153  IF (CBGENW.LT.0.)BGENOW=PI-BGENOW
195      IF (GAM.LT.0.)BGENOW=2.*PI-BGENOW
196      CBGETN=(A1-RE)/(A1*ECC)
197      BGETHN=ABS(CBGETN)
198      IF (BGETHN.LE.1.)GO TO 154
199      BGETHN=0.
200      GO TO 155
201 154  BGETHN=ACOS(BGETHN)
202 155  IF (CBGETN.LT.0.)BGETHN=PI-BGETHN
203      BGETHN=2.*PI-BGETHN
204      DELBGE=BGETHN-BGENOW
205      DTI=(DELBGE-ECC*(SIN(BGETHN)-SIN(BGENOW)))/SQRT(CMUE/(A1*A1*A1))
206      THT1=ACOS(CTH*COS(DELETA)+STH*SIN(DELETA)*COS(AZ1))
207      DLPHI=ASIN(SIN(DELETA)*SIN(AZ1)/SIN(THT1))
208      PHIMP=PHI+(DLPHI-OMEGA*DTI)*RAD
209      THIMP=90.-THT1*RAD
210 31  AZI=AZ1*RAD
211      IF (NN.NE.2.OR.NN.NE.3) GO TO 90
212      PRINT 75
213 75  FORMAT(1H1)
214      LINES=45
215 90  FFF=1.-FLAT
216      PHIPP=PI/2.-ATAN2(FFF*FFI*CTHL,STHL)
217      SITH=SIN(PHIPP)
218      CITH=COS(PHIPP)
219      RPAD=FFF*RE/SQRT(FFF*FFF*SITH*SITH+CITH*CITH)
220      DLTL=T+DTZ
221      DO 79 I=1,2
222      W1R=SIN(OMEGA*DL1L)*SITH
223      W2R=CITH
224      W3R=COS(OMEGA*DLTL)*SITH
225      XLR=A(1,1)*W1R+A(1,2)*W2R+A(1,3)*W3R
226      YLR=A(2,1)*W1R+A(2,2)*W2R+A(2,3)*W3R
227      ZLR=A(3,1)*W1R+A(3,2)*W2R+A(3,3)*W3R

```

APRTN

LIFTING-ROBOT LISTINGS

```

228      CPDX=Y*ZLR-Z*YLR
229      CPDY=Z*XLR-X*ZLR
230      CPDZ=X*YLR-Y*XLR
231      RRR=SQRT(CPDX**2+CPDY**2+CPDZ**2)
232      CPDXX=(YLR*CPDZ-ZLR*CPDY)/RRR
233      CPDYY=(ZLR*CPDX-XLR*CPDZ)/RRR
234      CPDZZ=(XLR*CPDY-YLR*CPDX)/RRR
235      RNGAN=ATAN2(X*CPDXX+Y*CPDYY+Z*CPDZZ,X*XLR+Y*YLR+Z*ZLR)
236      IF(I.NE.1) GO TO 79
237      RNGO=RNGAN*(RPAD+R-ALTP)/2.
238      ULTL=DTZ
239      79 CONTINUE
240      RNGAN=RNGAN*RAD
241      N=2*NN-1
242      50 IF(LINES)55,55,60
243      C WRITE HEADER
244      55 IF(LPRINT.EQ.2.AND.NMAX.NE.0) GO TO 41
245      IF(LPRINT.EQ.2.AND.ICONT-LONGPR.GT.ICOUNT)GO TO 41
246      KPAGE=KPAGE+1
247      WRITE( 6,56)CASE,(HEAD(I),I=1,10),KPAGE
248      56 FORMAT(1H1,6H ROBOT,2X,6HCASE =,F11.4,10X,10A6,10X,4HPAGE,I3)
249      41 LINES=45
250      60 IF(LPRINT.EQ.2.AND.NMAX.NE.0) GO TO 42
251      IF(LPRINT.EQ.2.AND.ICONT-LONGPR.GT.ICOUNT)GO TO 42
252      WRITE(6,61)RID(N),RID(N+1),T,VI,VR,THE,XLACC,X,Y,Z,w,U,v,ALIP,
253      1XMACH,FAA,ADEG,SDEG,CPDG,CYDG,R,XM,Q,TISP,FAN,THIMP,PHIMP,THETG,
254      2XINC,XNOD,SW,SU,SV,AHI,GAM,THEP,XMLB,C1,C3,THETA,PHI,AZ1
255      3,RNGO,RNGAN,DVAR(8),(ZAP(I),I=1,4),ZAP(6)
256      61 FORMAT( /1X,2A6,6H TIME,E15.8,6H VSUBI,E15.8,6H VSUBR,E15.8,
257      16H THRST,E15.8,6H LACC,E15.8,/13X,6H Z13-X,E15.8,6H X13-Y,E15.8,
258      26H Y13-Z,E15.8,6H ZD13W,E15.8,6H XD13U,E15.8,/13X,6H YD13V,E15.8,
259      36H ALT,E15.8,6H MACH,E15.8,6H FAA,E15.8,6H ALPHA,E15.8,/13X,
260      46H SIGMA,E15.8,6H CHIP,E15.8,6H CHIY,E15.8,6H R,E15.8,
261      56H MASS,E15.8,/13X,6H Q,E15.8,6H ISP,E15.8,6H FAN,E15.8,
262      66H LTIMP,E15.8,6H LNGMP,E15.8,/13X,6H GDLAT,E15.8,6H INCL,E15.8,
263      76H NODS,E15.8,6H SW,E15.8,6H SU,E15.8,/13X,6H SV,E15.8,
264      86H AHI,E15.8,6H GAM,E15.8,6H THEP,E15.8,6H XMLB,E15.8,/13X,
265      96H C1,E15.8,6H C3,E15.8,6H LAT,E15.8,6H LONG,E15.8,
266      96H AZ,S,E15.8/13X,6H RANGE,E15.8,6H RNGAN,E15.8,6H PDOT,E15.8,
267      *6H PEN,E15.8,6H CH VL,E15.8,/13X,6H TN L,E15.8,6H GV L,E15.8,
268      *6H DR L,E15.8)
269      42 LINES=LINES-10
270      IF(NTABLE.EQ.0)RETURN
271      IF(NMAX.EQ.0.AND.ICOUNT.EQ.ICONT-1)GO TO 101
272      RETURN
273      C
274      C*** WRITE OUTPUT FOR REPORT
275      101 GAMR=(SW*X+SU*Y+SV*Z)/(R*VR)
276      IF(GAMR.LE.1..AND.GAMR.GE.-1.)GO TO 160
277      IF(GAMR.LE.1.)GO TO 161
278      GAMR=PI/2.
279      GO TO 162
280      161 GAMR=1.5*PI
281      GO TO 162
282      160 GAMR=ASIN(GAMR)
283      162 IF(GAMR.GT.PI)GAMR=GAMR-2.*PI
284      GAMR=GAMR*RAD

```

APRTN

LIFTING-ROBOT LISTINGS

C

285
286
287
288
289
290
291
292
293

```
      ALFQ=ALF*Q  
      CPDG=-CPDG  
      WRITE( 9)RID(N),RID(N+1),T,VI,VR,GAM,GAMR,XM,XMLB,THE,THEP,  
A  XLACC,ALTP,R,XMACH,Q,ALFP,FAA,AZI,CPDG,CYDG,CHIR,AHI,X,Y,Z,  
B  W,U,V,ALFQ,THETA,PHI,(DVAR(I),I=1,3),XNOD,RNGO,RNGAN,ADEG,THIMP,  
C  PHIMP  
100 RETURN  
      END
```

5

.L.O.

3 AQMAX

LIFTING-ROBOT LISTINGS

1,5 TPF\$ AQMAX
PUR 0018-05/17-11:50

AQMAX

LIFTING-ROBOT LISTINGS

DT*TPF\$.AQMAX

```
1      SUBROUTINE AQMAX
2      COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
3      1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBCT(4),NENDCT(4),NMAX,
4      2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
5      COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
6      1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
7      2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
8      CALL APRTN(14)
9      NTRG6=-1
10     RETURN
11     END
```

5 .L,U.

ASIMP

LIFTING-ROBOT LISTINGS

S TPF\$.ASIMP
PUR 0018-05/17-11:50

ASIMP

LIFTING-ROBOT LISTINGS

OT*TPF\$.ASIMP

```

1      SUBROUTINE ASIMP
2      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
3      1 TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
4      2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
5      3 HNOM(15),HMAX(15)
6      COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
7      1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
8      2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
9      3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
10     4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
11     5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
12     6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
13     7 BHMAX(15),BHMN,BSTEP(15)
14     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
15     1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
16     2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
17     COMMON/FORINT/HBANK(2),T,TT,VAR(6),XMI,ZAP(8),DVAR(15),
18     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
19     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
20     COMMON/IMPACT/XIMP(10),IMP
21     EXTERNAL DERIMP,AXPRT,GROUND
22     LLDSOL=3*(KRDER+2)+4
23     ITHR=IMP
24     DO 1 I=1,6
25     1 VAR(I)=XIMP(I)
26     XMI=XIMP(7)
27     T=XIMP(8)
28     XM=XMI
29     XMAUG=0.
30     TT=0.
31     THR=0.
32     NTRG1=1
33     TV1=T
34     NTRG2=1
35     ALT=100.
36     HNM=STEP(ITHR)
37     CALL DESOLV(15,0,0,DERIMP,0,5,KERR,HNM,HBANK,FSAVE,
38     1 1,LLDSOL,EU,EL,HMX,HMN,AYL,2,
39     2 NTRG1,AXPRT,T,TV1,NTRG2,GROUND,ALT,0.,
40     3 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
41     4 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)
42     RETURN
43     END

```

.L,0.

ATCST

LIFTING-ROBOT LISTINGS

1,5 TPF\$.ATCST
PUR 0018-05/17-11:50

ATCST

LIFTING-ROBOT LISTINGS

DT*TPF\$.ATCST

```

1      SUBROUTINE ATCST(Z,Y,XM,TD)
2      DOUBLE PRECISION X(6),Y(6),S(6),XM,R,RZ,SZ,P,A,TD,TG,F,G,FD,GD,DT
3      DIMENSION Z(6)
4      DO 5 I=1,6
5          X(I)=Z(I)
6      RZ=SQRT(Z(1)*Z(1)+Z(2)*Z(2)+Z(3)*Z(3))
7      A=X(4)*X(4)+X(5)*X(5)+X(6)*X(6)-2.*XM/RZ
8      SZ=X(1)*X(4)+X(2)*X(5)+X(3)*X(6)
9      P=DABS(XM/A)
10     P=TD/DMAX1(RZ,P)
11     DO 10 J=1,25
12     CALL GETS(A,P,S)
13     R=RZ*S(1)+SZ*S(2)+XM*S(3)
14     TG=RZ*S(2)+SZ*S(3)+XM*S(4)
15     DT=TG-TD
16     IF(DABS(DT /TD ).LT.5.D-12)GO TO 20
17     10 P=P-DT/R
18     WRITE(6,40)TD,TG,P,R
19     20 F=1.D0-XM *S(3)/RZ
20     G=RZ*S(2)+SZ *S(3)
21     FD=-XM *S(2)/(R *RZ)
22     GD=1.D0-XM *S(3)/R
23     DO 30 I=1,3
24     Y(I)=F*X(I)+G*X(I+3)
25     30 Y(I+3)=FD*X(I)+GD*X(I+3)
26     40 FORMAT(17H  CANNOT GET PSI,4D21.14)
27     RETURN
28     END

```

.L.O.

ATHREV

LIFTING-ROBOT LISTINGS

TPF\$.ATHREV
0018-05/17-11:50

ATHREV

LIFTING-ROBOT LISTINGS

JT*TPF\$.ATHREV

```

1  SUBROUTINE ATHREV
2  COMMON/FLORTE/FLORTE(15),DLFLRT(15)
3  COMMON/COAST/NCOST1,NCOST2,CX(6,2)
4  COMMON/IMPACT/XIMP(10),IMP
5  COMMON/QY/QY,CFR(15)
6  COMMON/PERF/ILAST,IPR,PRK1,PRK2,PRK3,PRK4,PRTOT,FGR,FPR,WPMX
7  COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
8  COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
9  COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
10 1  TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
11 2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
12 3  HNOM(15),HMAX(15)
13  COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
14 1  SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPH1Z,GR,GTH,
15 2  RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
16 3  VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
17 4  DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
18 5  FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
19 6  CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
20 7  BHMAX(15),BHMN,BSTEP(15)
21  COMMON/AGEN3/IORK,IHR,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
22 1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGT(4),NENDCT(4),NMAX,
23 2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
24  COMMON/FORINT/HBANK(2),T,TT,VAR(15),DVAR(15),
25 1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
26 2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
27  COMMON/RESTX/RESIX(8)
28  COMMON/JTHR/JTHR(15),TBLTM(4)
29  COMMON/XMIAD/XMIAD,DELMA
30  COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
31  DOUBLE PRECISION CY(6),XMUE,TAUD
32  C***** THIS ROUTINE TAKES SPECIAL FORINT COMMON*****
33 10 JT=IHR
34  IF(JT.GE.JUMP)TAUT(JT)=T-TBGIN
35  IF(JT.GE.JUMP)FLORTE(JT)=DVAR(7)
36  DLFLRT(JT)=0.
37  IF(JT.GE.JUMP.AND.XMDOT.GT.0.)DLFLRT(JT)=1./XMDOT+1./DVAR(7)
38  TBGIN=T
39  IF(JT.EQ.NVNT)GO TO 9
40  IF(JT.NE.JUMP-1)CALL APRTN(1)
41  IHR=IHR+1
42  TIME(1,IHR)=T
43  TV7=VAR(7)-FUEL(IHR)
44  NTRG7=-2
45  NTRG2=2
46  IF(MSWCH(IHR).GT.0)GO TO 200
47  NTRG2=-2
48  NTRG7=2
49 200 CONTINUE
50  HNM=STEP(IHR)
51  HMX=PRINT(IHR)
52  ANE(1)=TNE(1,IHR)*COS(TNE(2,IHR)/RAD) +TNE(3,IHR)*
53 1 COS(TNE(4,IHR)/RAD)
54  ANOLD=ANE(2)*THROT
55  ANE(2)=TNE(1,IHR)+TNE(3,IHR)
56  EXIT=ANE(1)*AE(IHR)

```

ATHREV

LIFTING-ROBOT LISTINGS

```

57      XMDOT=ANE(2)*XMD(ITHR)
58      IF(JTHR(ITHR).LT.0)XMDOT=0.
59      THR=ANE(1)*F(ITHR)
60      KCST=0
61      IF(THR.LT.1.E-6 .AND.IORR.EQ.0)KCST=1
62      IF(ITHR.GT.NOEVNT(1))GO TO 1
63      THR=THR+EXIT*PSL
64      C
65      C*** CHECK FOR FIRST POINT
66      1 IF(JT.EQ.JUMP-1)GO TO 8
67      IF(JTHR(JT).NE.JTHR(ITHR))DELMA=0.
68      IF(JTHR(JT).LT.0)VAR(7)=XMIAD
69      IF(ITHR.EQ.NENDCT(KCYTAB))KCYTAB=KCYTAB+1
70      XMAUG=XMAUG-WJET(JT)
71      IF(LSB.EQ.1)GO TO 8
72      C
73      C*** CHECK FOR IMPACT
74      IF(IMP.NE.JT)GO TO 99
75      DO 98 I=1,6
76      98 XIMP(I)=VAR(I)
77      XIMP(7)=WJET(JT)
78      XIMP(8)=T
79      99 CONTINUE
80      C
81      C*** IMPACT CHECK OVER
82      C
83      C*** CALCULATE PERF RES AND TAUT(IPR)
84      IF(ITHR.NE.IPR)GO TO 103
85      XM=XMIAD+XMAUG
86      TEMP=WPMX-PRK4*XM*CFR(IPR)/XMD(IPR)
87      JA=ILAST
88      100 IF(JA.EQ.IPR)GO TO 101
89      TEMP=TEMP-(TNE(1,JA)+TNE(3,JA))*CFR(JA)*TAUT(JA)
90      101 IF(JA.GE.NVNT)GO TO 102
91      JA=JA+1
92      GO TO 100
93      102 TEMP1=TEMP/(1.-PRK4)
94      TAUT(IPR)=TEMP1/(ANE(2)*CFR(IPR))
95      TIME(1,IPR+1)=TIME(1,IPR)+TAUT(IPR)
96      PRTOT=PRK4*(XM-ANE(2)*XMD(IPR)*TAUT(IPR))
97      FGR=PRTOT*PRK1/PRK4
98      FPR=PRTOT*PRK3/PRK4
99      C
100     C*** PERF RESERVES AND TAUT(IPR) CALCULATED
101     103 CONTINUE
102     C
103     C*** CHECK FOR RESTART
104     DO 2 I=1,7
105     2 DVAR(I)=DVAR(I)
106     IF(JT.NE.NVRST)GO TO 4
107     C
108     DO 3 I=1,7
109     RESTX(I)=VAR(I)
110     3 DELXDR(I)=DVAR(I)
111     RESTX(7)=XM
112     RESTX(8)=T
113     CALL APRTN(15)

```

ATHREV

LIFTING-ROBOT LISTINGS

```

114      4 IF(IORR.EQ.1)GO TO 5
115      CALL ADER1I
116      CALL ADERI
117      GO TO 6
118      5 CALL ADER1
119      CALL ADER
120      6 DO 7 I=1,7
121      7 DELXD(JT,I)=DVAR(I)-DVAR(I)
122 C      DELXD(JT,7)=XMDOT-ANOLD*XMD(JT)
123      CALL APRTN(5)
124      8 TV2=T+TAUT(ITHR)
125 C**** CHECK FOR ANALYTIC COAST ***
126      IF(KCST.EQ.0)RETURN
127      IF(ITHR.NE.NCOST1 .AND. ITHR.NE.NCOST2)RETURN
128      IL=0.
129      CALL AEOSR
130      IF(ITHR.EQ.JUMP)CALL ADERI
131      CALL APRTN(10)
132      L=1
133      IF(ITHR.EQ.NCOST2)L=2
134      DO 96 I=1,3
135      CX(I,L)=VAR(I+3)
136      96 CX(I+3,L)=VAR(I)
137      XMUE=CMUE
138      TAUD=TAUT(ITHR)
139      CALL ATCST(CX(1,L),CY,XMUE,TAUD)
140      T=T+TAUD
141      DO 95 I=1,3
142      VAR(I)=CY(I+3)
143      95 VAR(I+3)=CY(I)
144      TL=0.
145      CALL AEOSR
146      CALL ADER1I
147      CALL ADERI
148      CALL APRTN(11)
149      TV1=T+PRINT(ITHR+1)
150 C**** COAST COMPLETED ***
151      GO TO 10
152      9 CONTINUE
153      IF(LSB.EQ.1)GO TO 111
154      TL=0.
155      CALL AEOSR
156      CALL APRTN(15)
157      DO 50 I=1,7
158      50 DELXD(ITHR,I)=DVAR(I)
159 C      DELXD(ITHR,7)=-ANE(2)*XM (ITHR)*THROT
160      TIME(1,NVNT+1)=T
161      FLORTE(ITHR)=DELXD(ITHR,7)
162      WRITE(6,9943)(DLFLRT(I),I=1,NVNT)
163      WRITE(6,9943)(FLORTE(I),I=1,NVNT)
164      9943 FORMAT(6HBXNDX=7E16.8)
165      111 THR=0.
166      XMDOT=0.
167      XMAUG=XMAUG-WJET(ITHR) -PRTOT
168      XM=XMIAD+XMAUG
169      CALL ADER1I
170      CALL ADERI

```

ATHREV

LIFTING-ROBOT LISTINGS

171 CALL APRTN(1)
172 CALL RTMRK
173 RETURN
174 END

3 .L.O.

ATILT

LIFTING-ROBOT LISTINGS

T,S TPF\$.ATILT
PUR 0018-05/17-11:50

ATILT

LIFTING-ROBOT LISTINGS

DT*TPF\$,ATILT

```

1      SUBROUTINE ATILT
2      COMMON/AZMTH/STH1,CTH1,XASM,YASM,ZASM,XJX(3),XJV(3),NALTW
3      COMMON/DELX/DELX(10),XLAMBX(3,10)
4      COMMON/KLIFT/KLIFT
5      COMMON/FSCY/NFSCY,FSCYD,TFSCY
6      COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
7      COMMON/CYT/NCYT,TCY1,TCY2,TCY3,TCY4,CYTM
8      COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
9      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
10     1 TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
11     2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
12     3 HNOM(15),HMAX(15)
13     COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
14     1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
15     2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
16     3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
17     4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
18     5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
19     6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
20     7 BHMAX(15),BHMN,BSTEP(15)
21     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
22     1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
23     2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
24     COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
25     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
26     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
27
28     C      KS1=      1      2      3      4
29     C      LFTOFF      TILTOVR      GRVTRN      OPTCON
30     C
31     GO TO(1,2,3),KS1
32     1 KS1=2
33     TV4=TTILT
34     SCHIY=0.
35     CCHIY=0.
36     CCHIY=1.
37     CALL APRTN(7)
38     RETURN
39     2 KS1=3
40     IF(NMAX.EQ.0)NTRG6=1
41     IF(TCHFRZ.GT.TTILT)CALL APRTN(8)
42     TV4=TCHFRZ
43     RETURN
44     3 KS1=4
45     TQ=T
46     NTRG4=-2
47     IF(LSB.EQ.0.OR.NALTW.EQ.1)GO TO 4
48     CALL RTMRK
49     RETURN
50     4 CONTINUE
51     KLIFT=1
52     CALL ADER1
53     CALL ADER
54     CALL APRTN(18)
55     DELX(1)=W
56     DELX(2)=U

```

ATILT

LIFTING-ROBOT LISTINGS

```
57      DELX(3)=V
58      DELX(4)=X
59      DELX(5)=Y
60      DELX(6)=Z
61      DELX(7)=XMI
62      DELX(8)=ZAP(1)
63      RETURN
64      END
```

5 .L+0.



5

ATTRAC

LIFTING-ROBOT LISTINGS

T,S TPF\$.ATTRAC
PUR 0018-05/17-11:50

ATTRAC

LIFTING-ROBOT LISTINGS

DI*TPF\$.ATTRAC

```

1      SUBROUTINE ATTRAC(T,F,XMD,N,AREX,CANIN,CANOUT)
2      COMMON/AEDK/IAEDK
3      COMMON/TABLK/MMM(8),MTT(4)
4      COMMON/AGEN2/DUMM1(213),EXIT,DUMM2(113)
5      DIMENSION TT1(218,4),WDT(218,4),THRT(218,4),K(4),ANIE(58,2),
6      IANOE(58,2)
7      DATA(TT1 (1,1),I=1, 90)/          .000,          .125,          .250,
8      B      .375,          .500,          .625,          .750,          .875,
9      C      1.000,          2.000,          3.000,          4.000,          5.000,
10     D      6.000,          7.000,          8.000,          9.000,          10.000,
11     E      11.000,          12.000,          13.000,          14.000,          15.000,
12     F      16.000,          17.000,          18.000,          19.000,          20.000,
13     G      21.000,          22.000,          23.000,          24.000,          25.000,
14     H      26.000,          27.000,          28.000,          29.000,          30.000,
15     I      31.000,          32.000,          33.000,          34.000,          35.000,
16     J      36.000,          37.000,          38.000,          39.000,          40.000,
17     K      41.000,          42.000,          43.000,          44.000,          45.000,
18     L      46.000,          47.000,          48.000,          49.000,          50.000,
19     M      51.000,          52.000,          53.000,          54.000,          55.000,
20     N      56.000,          57.000,          58.000,          59.000,          60.000,
21     O      61.000,          62.000,          63.000,          64.000,          65.000,
22     P      66.000,          67.000,          68.000,          69.000,          70.000,
23     Q      71.000,          72.000,          73.000,          74.000,          75.000,
24     R      76.000,          77.000,          78.000,          79.000,          80.000,
25     S      81.000,          82.000/
26     DATA(TT1 (1,1),I= 91,148)/          83.000,          84.000,          85.000,
27     B      86.000,          87.000,          88.000,          89.000,          90.000,
28     C      91.000,          92.000,          93.000,          94.000,          95.000,
29     D      96.000,          97.000,          98.000,          99.000,          100.000,
30     E      101.000,          102.000,          103.000,          104.000,          105.000,
31     F      106.000,          107.000,          108.000,          109.000,          110.000,
32     G      111.000,          112.000,          113.000,          114.000,          115.000,
33     H      116.000,          117.000,          118.000,          119.000,          120.000,
34     I      121.000,          122.000,          123.000,          124.000,          125.000,
35     J      126.000,          127.000,          128.000,          129.000,          130.000,
36     K      131.000,          132.000,          133.000,          134.000,          135.000,
37     L      136.000,          137.000,          138.000,          138.875,          139.000/
38     DATA(THRT(I,1),I=1, 90)/ 1528225.766, 1539220.812, 1559303.760,
39     A 1539772.234, 1541362.031, 1544997.312, 1545365.609, 1545236.672,
40     B 1545193.391, 1544920.266, 1544227.156, 1543524.266, 1542786.500,
41     C 1542060.437, 1541998.312, 1542540.156, 1543094.812, 1543662.234,
42     D 1544129.781, 1544623.000, 1545125.484, 1545430.781, 1545744.766,
43     E 1545945.750, 1546153.953, 1546367.812, 1546590.141, 1546821.047,
44     F 1547007.281, 1547249.500, 1547500.406, 1547760.141, 1548028.984,
45     G 1548306.937, 1548593.561, 1548889.203, 1549193.891, 1549507.609,
46     H 1549813.719, 1550128.297, 1550451.672, 1550783.812, 1551124.687,
47     I 1551475.328, 1551835.062, 1552203.250, 1552579.766, 1552964.359,
48     J 1553365.312, 1553773.969, 1554185.141, 1554603.031, 1555027.281,
49     K 1555457.781, 1555894.516, 1556336.859, 1556784.516, 1557237.187,
50     L 1557627.875, 1558031.531, 1558440.250, 1558783.562, 1559097.250,
51     M 1559404.125, 1559703.812, 1560018.562, 1560301.875, 1560563.062,
52     N 1560791.391, 1561033.094, 1561364.562, 1561685.141, 1562054.750,
53     O 1562629.891, 1563213.625, 1563839.500, 1564510.937, 1565169.750,
54     P 1565824.969, 1566494.125, 1567181.766, 1567886.500, 1568636.062,
55     Q 1569332.391, 1570043.875, 1570770.891, 1571497.469, 1572224.900,
56     R 1572913.344, 1573612.562/

```

ATTRAC

LIFTING-ROBOT LISTINGS

```

57      DATA(THRT(I,1),I= 91,148)/ 1574321.078, 1575037.531, 1575749.391,
58      A 1576451.953, 1577156.812, 1577873.750, 1578573.734, 1579270.000,
59      B 1579903.750, 1580525.172, 1581143.891, 1581760.422, 1582358.141,
60      C 1582920.750, 1583482.187, 1584044.187, 1584607.422, 1585173.000,
61      D 1585680.453, 1586191.547, 1586707.219, 1587223.703, 1587744.359,
62      E 1588290.781, 1588845.547, 1589409.437, 1589983.484, 1590574.906,
63      F 1590980.750, 1591398.844, 1591833.547, 1592293.375, 1592768.500,
64      G 1593272.531, 1593797.469, 1594342.875, 1594910.000, 1595502.375,
65      H 1596286.203, 1597094.187, 1597927.766, 1598792.641, 1599686.625,
66      I 1600604.312, 1601553.031, 1602536.141, 1603555.875, 1604611.766,
67      J 1605795.000, 1607018.031, 1608251.437, 1609525.437, 1610847.875,
68      K 1612221.625, 1613630.406, 1615082.219, 1616388.937, 1616577.219/
69      DATA(WDT (I,1),I=1, 90)/ .000, 3630.312, 7260.625,
70      A 10890.937, 14521.250, 18151.562, 21781.875, 25412.187,
71      B 29042.500, 58085.000, 87137.500, 116190.000, 145219.500,
72      C 174249.000, 203268.000, 232287.000, 261321.500, 290356.000,
73      D 319407.000, 348458.000, 377522.500, 406587.000, 435661.000,
74      E 464735.000, 493814.000, 522893.000, 551978.500, 581064.000,
75      F 610154.000, 639244.000, 668342.000, 697440.000, 726543.000,
76      G 755646.000, 784757.500, 813869.000, 842987.000, 872105.000,
77      H 901231.500, 930358.000, 959491.500, 988625.000, 1017767.500,
78      I 1046910.000, 1076061.500, 1105213.000, 1134373.500, 1163534.000,
79      J 1192704.000, 1221874.000, 1251054.500, 1280235.000, 1309427.000,
80      K 1338619.000, 1367820.500, 1397022.000, 1426235.500, 1455449.000,
81      L 1484672.500, 1513896.000, 1543129.000, 1572362.000, 1601602.000,
82      M 1630842.000, 1660089.000, 1689336.000, 1718589.000, 1747842.000,
83      N 1777099.000, 1806356.000, 1835619.000, 1864882.000, 1894223.500,
84      O 1923565.000, 1952920.500, 1982276.000, 2011649.000, 2041022.000,
85      P 2070413.500, 2099805.000, 2129215.750, 2158626.500, 2188057.500,
86      Q 2217488.500, 2246940.250, 2276392.000, 2305864.500, 2335337.000,
87      R 2364761.000, 2394185.000/
88      DATA(WDT (I,1),I= 91,148)/ 2423629.750, 2453074.500, 2482540.750,
89      A 2512007.000, 2541494.500, 2570982.000, 2600491.250, 2630000.500,
90      B 2659529.000, 2689057.500, 2718605.250, 2748153.000, 2777719.250,
91      C 2807285.500, 2836869.750, 2866454.000, 2896055.250, 2925656.500,
92      D 2955274.000, 2984891.500, 3014525.250, 3044159.000, 3073808.750,
93      E 3103458.500, 3133126.000, 3162793.500, 3192479.500, 3222165.500,
94      F 3251864.250, 3281563.000, 3311275.250, 3340987.500, 3370715.000,
95      G 3400442.500, 3430186.250, 3459930.000, 3489691.750, 3519453.500,
96      H 3549239.500, 3579025.500, 3608838.000, 3638650.500, 3668491.750,
97      I 3698333.000, 3728204.000, 3758075.000, 3787979.000, 3817883.000,
98      J 3847824.000, 3877765.000, 3907746.000, 3937727.000, 3967750.500,
99      K 3997774.000, 4027842.500, 4057911.000, 4084277.812, 4088044.500/
100     DATA(TT1 (I,2),I=1, 58)/ .000, .125, .250,
101     A .375, .500, .625, .750, .875,
102     B 1.000, 1.12, 1.250, 1.375, 1.500,
103     C 1.625, 1.750, 1.875, 2.000, 2.125,
104     D 2.250, 2.375, 2.500, 2.625, 2.750,
105     E 2.875, 3.000, 3.125, 3.250, 3.375,
106     F 3.500, 3.625, 3.750, 3.875, 4.000,
107     G 4.125, 4.250, 5.000, 6.000, 7.000,
108     H 8.000, 9.000, 10.000, 11.000, 12.000,
109     I 13.000, 14.000, 15.000, 16.000, 17.000,
110     J 18.000, 19.000, 20.000, 20.033, 20.125,
111     K 20.250, 20.375, 20.5, 20.625, 21.733/
112     DATA(THRT(I,2),I=1, 58)/ 2020721.523, 1976767.891, 1827748.625,
113     A 1729488.156, 1675133.062, 1599701.062, 1581751.859, 1585083.344,

```

ATTRAC

LIFTING-ROBOT LISTINGS

114	B	1586842.719,	1586723.859,	1586399.047,	1586149.422,	1585932.125,
115	C	1585715.672,	1585497.391,	1585279.156,	1585061.234,	1584843.750,
116	D	1584626.578,	1584409.766,	1584193.266,	1583977.156,	1583761.375,
117	E	1583545.953,	1583330.875,	1583116.187,	1582901.797,	1582687.750,
118	F	1582474.062,	1582260.719,	1582047.781,	1581835.172,	1581622.937,
119	G	1581711.844,	1581855.719,	1582581.500,	1583557.250,	1584653.359,
120	H	1585762.812,	1586883.234,	1588009.906,	1589138.484,	1590280.375,
121	I	1591405.375,	1592500.906,	1593552.219,	1594514.953,	1595339.391,
122	J	1595894.953,	1592428.031,	1547664.281,	1537433.141,	1508908.844,
123	K	949532.156,	585998.008,	375854.219,	90221.077,	35080.993/
124		DATA(WDT(I,2),I=1, 58)/		.000,	3199.625,	6399.250,
125	A	9598.844,	12798.437,	15998.031,	19197.625,	22397.219,
126	B	25596.812,	28573.937,	31551.062,	34528.187,	37505.312,
127	C	40482.437,	43459.562,	46436.687,	49413.812,	52390.937,
128	D	55368.062,	58345.187,	61322.312,	64299.437,	67276.562,
129	E	70253.687,	73230.812,	76184.062,	79137.312,	82090.562,
130	F	85043.750,	87996.937,	90950.125,	93903.312,	96856.500,
131	G	99809.687,	102762.875,	120482.000,	144133.000,	167784.000,
132	H	191464.000,	215144.000,	238853.000,	262562.000,	286300.750,
133	I	310039.500,	333807.000,	357574.500,	381368.000,	405161.500,
134	J	428897.750,	452634.000,	476427.687,	477212.812,	478111.875,
135	K	479333.437,	480555.000,	481776.500,	482998.000,	484053.312/
136		DATA(ANIE(I,2),I=1, 58)/		1.000,	.891,	.540,
137	A	.352,	.227,	.043,	.019,	.019,
138	B	.018,	.017,	.016,	.016,	.015,
139	C	.014,	.013,	.013,	.012,	.011,
140	D	.010,	.010,	.009,	.008,	.007,
141	E	.007,	.006,	.005,	.004,	.004,
142	F	.003,	.002,	.001,	.001,	.000,
143	G	.000,	.000,	.000,	.000,	.000,
144	H	.000,	.000,	.000,	.000,	.000,
145	I	.000,	.000,	.000,	.000,	.000,
146	J	.000,	.000,	.000,	.000,	.000,
147	K	.000,	.000,	.000,	.000,	.000/
148		DATA(ANOE(I,2),I=1, 58)/		4.000,	4.000,	4.000,
149	A	4.000,	4.000,	4.000,	4.000,	4.000,
150	B	4.000,	4.000,	4.000,	4.000,	4.000,
151	C	4.000,	4.000,	4.000,	4.000,	4.000,
152	D	4.000,	4.000,	4.000,	4.000,	4.000,
153	E	4.000,	4.000,	4.000,	4.000,	4.000,
154	F	4.000,	4.000,	4.000,	4.000,	4.000,
155	G	4.000,	4.000,	4.000,	4.000,	4.000,
156	H	4.000,	4.000,	4.000,	4.000,	4.000,
157	I	4.000,	4.000,	4.000,	4.000,	4.000,
158	J	4.000,	4.000,	4.000,	4.000,	4.000,
159	K	2.518,	1.557,	.999,	.24,	.06/
160		DATA(TT1(I,3),I=1, 90)/		-0.000,	0.125,	0.250,
161	A	0.375,	0.500,	0.625,	0.750,	0.875,
162	B	1.000,	1.125,	1.250,	1.375,	1.500,
163	C	1.625,	1.750,	1.875,	2.000,	2.125,
164	D	2.250,	2.375,	2.500,	2.625,	2.750,
165	E	2.875,	3.000,	4.000,	5.000,	6.000,
166	F	7.000,	8.000,	9.000,	10.000,	11.000,
167	G	12.000,	13.000,	15.000,	17.000,	19.000,
168	H	21.000,	23.000,	25.000,	27.000,	29.000,
169	I	31.000,	33.000,	35.000,	37.000,	39.000,
170	J	41.000,	43.000,	45.000,	47.000,	49.000,

ATTRAC

LIFTING-ROBOT LISTINGS

171	K	51.000,	53.000,	55.000,	57.000,	59.000,
172	L	61.000,	63.000,	65.000,	67.000,	69.000,
173	M	71.000,	73.000,	75.000,	77.000,	79.000,
174	N	81.000,	83.000,	85.000,	87.000,	89.000,
175	O	91.000,	93.000,	95.000,	97.000,	99.000,
176	P	101.000,	103.000,	105.000,	107.000,	109.000,
177	Q	111.000,	113.000,	115.000,	117.000,	119.000,
178	R	121.000,	123.000/			
179		DATA(TT1(I,3),I= 91,180)/		125.000,	127.000,	129.000,
180	A	131.000,	133.000,	135.000,	136.000,	137.000,
181	B	138.000,	138.625,	138.750,	138.875,	139.000,
182	C	139.125,	139.250,	139.375,	139.500,	139.625,
183	D	139.750,	139.875,	140.000,	140.125,	140.250,
184	E	140.375,	141.000,	143.000,	145.000,	147.000,
185	F	149.000,	151.000,	153.000,	155.000,	157.000,
186	G	159.000,	161.000,	163.000,	165.000,	167.000,
187	H	169.000,	171.000,	173.000,	175.000,	177.000,
188	I	179.000,	181.000,	183.000,	185.000,	187.000,
189	J	189.000,	191.000,	193.000,	195.000,	197.000,
190	K	199.000,	201.000,	203.000,	205.000,	207.000,
191	L	209.000,	211.000,	213.000,	215.000,	217.000,
192	M	219.000,	221.000,	223.000,	225.000,	227.000,
193	N	229.000,	231.000,	233.000,	235.000,	237.000,
194	O	239.000,	241.000,	243.000,	245.000,	247.000,
195	P	249.000,	251.000,	253.000,	255.000,	257.000,
196	Q	259.000,	261.000,	263.000,	265.000,	267.000,
197	R	269.000,	271.000/			
198		DATA(TT1(I,3),I=181,218)/		273.000,	275.000,	277.000,
199	A	279.000,	281.000,	283.000,	285.000,	287.000,
200	B	289.000,	291.000,	293.000,	295.000,	297.000,
201	C	299.000,	301.000,	303.000,	305.000,	307.000,
202	D	309.000,	311.000,	313.000,	315.000,	317.000,
203	E	319.000,	321.000,	323.000,	325.000,	327.000,
204	F	329.000,	331.000,	333.000,	335.000,	337.000,
205	G	339.000,	341.000,	342.000,	343.000,	343.875/
206		DATA(THRT(I,3),I=1, 90)/		0.000,	0.000,	0.000,
207	A	0.000,	0.000,	0.000,	0.000,	0.000,
208	B	0.000,	0.000,	0.000,	7179.359,	31115.708,
209	C	44042.923,	51385.045,	86197.294,	222111.580,	281996.961,
210	D	361113.289,	452220.441,	539537.773,	622820.961,	698230.430,
211	E	762529.117,	815232.109,	946068.227,	984699.930,	1046296.500,
212	F	1122707.703,	1131998.203,	1141278.500,	1143010.203,	1144750.906,
213	G	1146490.203,	1148216.000,	1151664.000,	1155098.406,	1158013.906,
214	H	1160425.703,	1162838.703,	1164257.906,	1165678.297,	1166738.594,
215	I	1167481.703,	1168170.797,	1168415.000,	1168361.000,	1168077.500,
216	J	1167873.203,	1168000.203,	1168121.406,	1167491.297,	1166828.594,
217	K	1167566.406,	1167383.500,	1167366.000,	1167346.203,	1167327.797,
218	L	1167307.797,	1167297.203,	1167047.094,	1166787.906,	1166715.500,
219	M	1166818.703,	1166928.297,	1167033.594,	1167145.906,	1167251.594,
220	N	1167354.594,	1167458.500,	1167568.703,	1167685.703,	1167795.406,
221	O	1167905.000,	1168011.703,	1168126.297,	1168231.203,	1168349.797,
222	P	1168480.094,	1168609.797,	1168617.297,	1168618.500,	1168621.594,
223	Q	1168623.703,	1168627.906,	1168621.797,	1168631.203,	1168627.703,
224	R	1168637.094,	1168642.203/			
225		DATA(THRT(I,3),I= 91,180)/		1168641.797,	1168649.594,	1168652.297,
226	A	1168660.094,	1168665.906,	1168672.703,	1168675.406,	1168675.500,
227	B	1168674.000,	1168635.000,	1160490.000,	958239.844,	945359.844,

ATTRAC

LIFTING-ROBOT LISTINGS

228	C	933742.383,	933791.156,	933778.023,	933767.344,	933763.094,
229	D	933752.406,	933754.523,	933749.422,	933741.961,	933739.953,
230	E	933733.742,	933726.938,	933689.656,	933622.672,	933536.609,
231	F	933503.906,	933531.172,	933552.273,	933574.828,	933593.650,
232	G	933620.477,	933639.383,	933659.117,	933681.906,	933705.297,
233	H	933733.594,	933752.867,	933775.289,	933796.906,	933819.563,
234	I	933822.867,	933808.859,	933787.156,	933763.977,	933748.883,
235	J	933731.094,	933709.938,	933689.406,	933668.594,	933647.281,
236	K	933627.422,	933608.641,	933588.492,	933604.188,	933617.219,
237	L	933629.180,	933644.313,	933662.578,	933676.938,	933690.063,
238	M	933704.359,	933718.289,	933731.617,	933746.789,	933765.594,
239	N	933769.117,	933754.703,	933742.000,	933727.297,	933714.633,
240	O	933700.820,	933696.719,	933682.273,	933669.523,	933656.023,
241	P	933643.727,	933629.844,	933614.047,	933598.797,	933580.150,
242	Q	933574.344,	933555.430,	933524.070,	933508.133,	933483.650,
243	R	933469.328,	933450.531/			
244		DATA(THRT(I,3),I=181,218)/		933431.563,	933413.672,	933394.617,
245	A	933378.289,	933360.547,	933338.852,	933321.953,	933314.344,
246	B	933279.297,	933226.891,	933169.820,	933114.727,	933065.398,
247	C	931308.156,	931264.273,	931229.273,	931188.711,	931153.641,
248	D	931119.047,	931083.148,	931044.359,	931001.961,	930955.727,
249	E	930907.453,	930858.094,	930800.703,	930745.867,	930613.977,
250	F	930412.438,	930205.844,	929997.344,	929788.898,	929578.531,
251	G	929365.000,	929150.461,	929043.477,	928937.023,	928841.539/
252		DATA(WDT(I,3),I=1, 90)/		0.000,	0.037,	0.074,
253	A	0.111,	0.148,	0.186,	0.223,	0.260,
254	B	0.297,	89.828,	179.359,	268.891,	358.422,
255	C	447.953,	537.484,	627.016,	716.547,	806.078,
256	D	895.609,	985.141,	1074.672,	1164.203,	1253.734,
257	E	1343.266,	1432.797,	3814.638,	6196.478,	8578.319,
258	F	10960.159,	13342.000,	16026.000,	18710.000,	21394.000,
259	G	24078.000,	26762.000,	32180.919,	37599.837,	43038.438,
260	H	48496.719,	53955.000,	59438.800,	64922.600,	70411.651,
261	I	75908.162,	81408.297,	86910.500,	92412.698,	97913.598,
262	J	103413.200,	108912.800,	114410.800,	119908.800,	125404.740,
263	K	130898.620,	136392.500,	141886.301,	147380.100,	152873.461,
264	L	158366.381,	163859.301,	169349.900,	174840.500,	180330.340,
265	M	185819.420,	191308.500,	196798.461,	202288.420,	207778.820,
266	N	213269.660,	218760.500,	224252.301,	229744.100,	235236.301,
267	O	240728.900,	246221.500,	251715.260,	257209.020,	262703.199,
268	P	268197.801,	273692.398,	279187.559,	284682.719,	290177.699,
269	Q	295672.500,	301167.301,	306661.578,	312155.859,	317650.059,
270	R	323144.180,	328638.301/			
271		DATA(WDT(I,3),I= 91,180)/		334131.941,	339625.578,	345119.180,
272	A	350612.738,	356106.301,	361599.379,	364345.922,	367092.461,
273	B	369839.000,	371265.238,	371550.488,	371835.738,	372120.988,
274	C	372406.238,	372691.488,	372976.738,	373261.988,	373547.238,
275	D	373832.488,	374117.738,	374402.988,	374688.238,	374973.480,
276	E	375258.730,	376684.961,	381248.898,	385650.340,	390051.781,
277	F	394445.199,	398830.602,	403216.000,	407601.559,	411987.121,
278	G	416372.762,	420758.480,	425144.199,	429529.922,	433915.641,
279	H	438301.480,	442687.441,	447073.398,	451459.398,	455845.398,
280	I	460231.379,	464617.340,	469003.301,	473388.859,	477774.422,
281	J	482159.820,	486545.059,	490930.301,	495315.180,	499700.059,
282	K	504084.781,	508469.340,	512853.898,	517238.219,	521622.539,
283	L	526006.953,	530391.430,	534775.898,	539160.430,	543544.953,
284	M	547929.508,	552314.109,	556698.703,	561083.391,	565468.063,

ATTRAC

LIFTING-ROBOT LISTINGS

285	N	569852.773,	574237.477,	578622.203,	583006.609,	587391.008,
286	O	591775.297,	596159.508,	600543.703,	604927.594,	609311.461,
287	P	613695.273,	618078.977,	622462.703,	626846.109,	631229.508,
288	Q	635612.797,	639996.008,	644379.203,	648762.047,	653144.883,
289	R	657527.578,	661910.141/			
290		DATA(WDT(I,3),I=181,218)/		666292.703,	670674.906,	675057.109,
291	A	679439.203,	683821.203,	688203.203,	692584.891,	696960.563,
292	B	701348.000,	705729.203,	710110.398,	714490.797,	718871.203,
293	C	723249.547,	727625.852,	732001.703,	736376.656,	740751.617,
294	D	745126.281,	749500.641,	753875.000,	758251.078,	762627.150,
295	E	767004.078,	771381.844,	775759.602,	780137.570,	784515.523,
296	F	788892.641,	793268.922,	797645.203,	802019.398,	806393.602,
297	G	810766.602,	815138.398,	817324.297,	819510.203,	821192.844/
298		DATA(TT1(I,4),I=1, 82)/		-0.000,	0.125,	0.250,
299	A	0.375,	0.500,	0.625,	0.750,	0.875,
300	B	1.000,	1.125,	1.250,	1.375,	1.500,
301	C	1.625,	1.750,	1.875,	2.000,	2.125,
302	D	3.125,	4.125,	5.125,	6.125,	7.125,
303	E	8.125,	9.125,	10.125,	11.125,	12.125,
304	F	13.125,	14.125,	15.125,	17.125,	19.125,
305	G	21.125,	23.125,	25.125,	27.125,	29.125,
306	H	31.125,	33.125,	35.125,	37.125,	39.125,
307	I	41.125,	43.125,	45.125,	47.125,	49.125,
308	J	51.125,	53.125,	55.125,	57.125,	59.125,
309	K	61.125,	63.125,	65.125,	67.125,	69.125,
310	L	71.125,	73.125,	75.125,	77.125,	79.125,
311	M	81.125,	83.125,	85.125,	87.125,	88.854,
312	N	89.000,	89.125,	89.250,	89.375,	89.500,
313	O	89.625,	89.750,	89.875,	90.000,	90.125,
314	P	90.250,	90.375,	90.500,	90.625/	
315		DATA(THRT(I,4),I=1, 82)/		928841.539,	928795.328,	928657.539,
316	A	928425.094,	928103.797,	927713.313,	927174.422,	925732.406,
317	B	920730.500,	887475.859,	850123.891,	823549.859,	804978.250,
318	C	789907.539,	775202.609,	758039.867,	738288.977,	733304.023,
319	D	733325.906,	733201.758,	733076.539,	732949.820,	732821.289,
320	E	732692.211,	732557.148,	732428.828,	732292.648,	732153.680,
321	F	732009.844,	731874.063,	731713.820,	731409.117,	731122.313,
322	G	730831.578,	730574.203,	730311.133,	730054.797,	729790.039,
323	H	729542.781,	729292.148,	729032.477,	728778.758,	728512.008,
324	I	728251.039,	727985.609,	727728.977,	727465.648,	727204.953,
325	J	726934.953,	726673.977,	726409.180,	726129.203,	725850.344,
326	K	725580.906,	725309.359,	725022.852,	724732.820,	724445.250,
327	L	724153.797,	723856.65,	723544.406,	723224.922,	722895.563,
328	M	722546.750,	722172.813,	721780.281,	721354.758,	720863.867,
329	N	700193.977,	75749.873,	39246.130,	0.000,	0.000,
330	O	0.000,	0.000,	0.000,	0.000,	0.000,
331	P	0.000,	0.000,	0.000,	0.000/	
332		DATA(WDT(I,4),I=1, 82)/		0.000,	240.383,	480.760,
333	A	721.148,	961.523,	1201.914,	1442.297,	1682.680,
334	B	1923.063,	2163.445,	2403.828,	2644.203,	2884.578,
335	C	3124.953,	3365.328,	3605.703,	3846.078,	4086.453,
336	D	6009.453,	7932.453,	9635.078,	11337.703,	13040.313,
337	E	14742.938,	16445.555,	18146.695,	19847.836,	21548.977,
338	F	23250.117,	24951.258,	26650.719,	30049.633,	33448.555,
339	G	36844.078,	40239.594,	43633.594,	47026.078,	50418.555,
340	H	53808.164,	57197.758,	60585.813,	63972.336,	67358.859,
341	I	70742.383,	74125.898,	77507.938,	80888.500,	84269.055,

ATTRAC

LIFTING-ROBOT LISTINGS

```
342      J  87646.500,   91023.938,   94399.797,   97774.078,  101148.359,
343      K  104519.477,  107890.609,  111260.109,  114627.977,  117995.859,
344      L  121360.383,  124724.898,  128087.563,  131448.359,  134809.156,
345      M  138166.125,  141523.078,  144878.109,  148231.203,  151129.953,
346      N  151159.047,  151183.945,  151208.844,  151233.750,  151258.633,
347      O  151283.539,  151308.438,  151333.336,  151358.234,  151383.125,
348      P  151408.02,   151432.91,   151457.80,   151482.69 /
349      DATA(K(1),I=1,4)/ 148,58,218,82/
350      IF(N.NE.IAEDK) GO TO 1
351      CALL AMULG(2,MTT(N),K(N),T,TT1(1,N),AIE,ANIE(1,N),AOE,ANOE(1,N))
352      EXIT=AREX*(AIE*COS(CANIN/57.29578)+AOE*COS(CANOUT/57.29578))
353      1  CALL AMULG(2,MTT(N),K(N),T,TT1(1,N),F,THRT(1,N),XMD,WDI(1,N))
354      F=F*4.44822162
355      XMD=XMD*.45359237
356      RETURN
357      END
```

3 .L,0.

AWDEV

LIFTING-ROBOT LISTINGS

T,S TPF\$.AWDEV
PUR 0018-05/17-11:50

AWDEV

LIFTING-ROBOT LISTINGS

JT*TPF\$.AWDEV

```

1      SUBROUTINE AWDEV
2      COMMON/KLIFT/KLIFT
3      COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
4      COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
5      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRA,TLIFT,TTILT,
6      1  TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
7      2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
8      3  HNOM(15),HMAX(15)
9      COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
10     1  SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPH12,GR,GTH,
11     2  RTHE,RTH2,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
12     3  VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
13     4  DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
14     5  FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
15     6  CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
16     7  BHMAX(15),BHMN,BSTEP(15)
17     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
18     1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
19     2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
20     COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
21     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
22     2NTRG6,TV6,NTRG7,IV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
23     IWD=IWD+1
24     IF(LSB.NE.1.AND.WD(IWD).GT.0.)CALL APRTN(1)
25     XMAUG=XMAUG-WD(IWD)
26     IF(LSB.EQ.1)GO TO 4
27     DO 1 I=1,3
28     1 DVAR(I)=DVAR(I)
29     IF(IORR.EQ.0)GO TO 7
30     CALL ADER1
31     CALL ADER
32     GO TO 3
33     7 CALL ADER1I
34     CALL ADER1
35     3 DO 2 I=1,3
36     2 DELXDW(IWD,I)=DVAR(I)-DVAR(I)
37     IF(WD(IWD).GT.0.)CALL APRTN(17)
38     4 IF(IWD.EQ.NWVNT)GO TO 5
39     TV3=TIME(2,IWD+1)
40     GO TO 6
41     5 NTRG3=-2
42     6 IF(IWD.NE.IWDCHI)RETURN
43     IORR=0
44     CALL RTMRK
45     IF(KLIFT.EQ.1)RETURN
46     CALL ADER1I
47     CALL ADER1
48     CALL APRTN(18)
49     TQ=TIME(2,IWD)
50     RETURN
51     END

```

.L,0.



3

AXPRT

LIFTING-ROBOT LISTINGS

I,S TPF\$.AXPRT
PUR 0018-05/17-11:50

AXPRT

LIFTING-ROBOT LISTINGS

DT*TPF\$.AXPRT

```

1      SUBROUTINE AXPRT
2      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
3      1  TCHFRZ,DTZ,TQ,IL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
4      2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
5      3  HNOM(15),HMAX(15)
6      COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
7      1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
8      2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
9      COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
10     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
11     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
12     CALL APRTN(1)
13     TV1=AIN(T+.01)+PRINT(ITHR)
14     RETURN
15     END

```

5 .L,0.

BADLX

LIFTING-ROBOT LISTINGS

T,S TPF\$.BADLX
PUR 0018-05/17-11:50

BADLX

LIFTING-ROBOT LISTINGS

OT*TPF\$.BADLX

```

1  SUBROUTINE BADLX
2  COMMON/FSCY/NFSCY,FSCYD,TFSCY
3  COMMON/AZMTH/STH1,CTH1,XASM,YASM,ZASM,XJX(3),XJV(3),NALTW
4  COMMON/PERF/ILAST,IPR,PRK1,PRK2,PRK3,PRK4,PRTOT,FGR,FPR,WPMX
5  COMMON/BAKE/WISS(10,10),YL(7,10)
6  COMMON/BLAMB/YLBT(15,10),YLBW(15,10),GAPHI(10),PROD(2,10),WAH
7  COMMON/AUTO/KDB(30),SAVCP(30),XLAMB(30,10),DP2,KDT(30)
8  COMMON/YLI/YLINT(7,10),PHITE(10),PSIREQ(10),KCDPHI(10),PSTR(10)
9  COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
10 COMMON/TABLK/MCAN,MCAF,MCON1,MCON2,MCON3,MCON4,MBS4,MBS7,MTT1,MTT2
11 1,MTT3,MTT4
12 COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,ITILT,
13 1 TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
14 2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
15 3 HNOM(15),HMAX(15)
16 COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
17 1 SCHIY,CCHIY,TH2,STH,CTH,THGT,THETL,STHL,CTHL,DPHI2,GR,GTH,
18 2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WKZ,
19 3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
20 4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
21 5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
22 6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
23 7 BHMAX(15),BHMN,BSTEP(15)
24 COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCS1,
25 1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
26 2 NOEVRT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
27 COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
28 COMMON/FORINT/HBANK(2),T,TT,VAR(15),DVAR(15),
29 1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
30 2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
31 COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
32 COMMON/PENLTY/NCONPT,DXS(6)
33 COMMON/COUNT/ICOUNT,ICONT,LONGPR
34 COMMON/DELX/DELX(10),XLAMBX(3,10)
35 DIMENSION DDELX(8)
36
37 C
38 LSB=1
39 DELWZ=.005*WZERO
40 DCHID=.01*CHIDOT
41 DAA=.125/RAD
42 DCHYYD=.01/RAD
43 IF(NALTW.EQ.1)GO TO 95
44
45 C
46 C
47 GET LAMBDA BETA 16 FOR LIFTOFF WEIGHT
48
49 IF(KDB(16).EQ.0)GO TO 4
50 IF(NMAX.EQ.1.AND.ICOUNT.GT.0)GO TO 21
51 IF(JUMP.EQ.1)GO TO 101
52 DO 100 I=1,NOSS
53 XLAMBX(1,I)=YL(7,I)
54 100 XLAMB(16,I)=YL(7,I)
55 GO TO 23
56 101 CONTINUE
57 WZERO=WZERO+DELWZ
58 CALL AFORUN
59 DO 1 I=1,8

```

BADLX

LIFTING-ROBOT LISTINGS

```

57      1 DDELX(I)=VAR(I)
58      WZERO=WZERO-2.*DELWZ
59      CALL AFORUN
60      DO 2 I=1,8
61      2 DDELX(I)=(DDELX(I)-VAR(I))/(2.*DELWZ)
62      WZERO=WZERO+DELWZ
63      DO 3 I=1,NOSS
64      XLAMB(16,I)=0.
65      DO 3 J=1,7
66      XLAMB(16,I)=XLAMB(16,I)+DDELX(J)*YL(J,I)
67      3 XLAMBX(1,I)=XLAMB(16,I)
68      GO TO 23
69      21 DO 22 I=1,NOSS
70      22 XLAMB(16,I)=XLAMBX(1,I)
71      23 IF(NCONPT.GT.0)XLAMB(16,NCONPT)=XLAMB(16,NCONPT)+DDELX(8)
72      C
73      C      GET LAMBDA BETA 17 FOR CHIDOT
74      C
75      4 IF(KDB(17).EQ.0)GO TO 8
76      IF(NMAX.EQ.1.AND.ICOUNT.GT.0)GO TO 72
77      CHIDOT=CHIDOT+DCHID
78      CALL AFORUN
79      DO 5 I=1,8
80      5 DDELX(I)=VAR(I)
81      CHIDOT=CHIDOT-2.*DCHID
82      CALL AFORUN
83      DO 6 I=1,8
84      6 DDELX(I)=(DDELX(I)-VAR(I))/(2.*DCHID)
85      CHIDOT=CHIDOT+DCHID
86      DO 7 I=1,NOSS
87      XLAMB(17,I)=0.
88      DO 7 J=1,7
89      XLAMB(17,I)=XLAMB(17,I)+DDELX(J)*YL(J,I)
90      7 XLAMBX(2,I)=XLAMB(17,I)
91      GO TO 75
92      72 DO 73 I=1,NOSS
93      73 XLAMB(17,I)=XLAMBX(2,I)
94      75 IF(NCONPT.GT.0)XLAMB(17,NCONPT)=XLAMB(17,NCONPT)+DDELX(8)
95      C
96      C      GET LAMBDA BETA 18 FOR AZIMUTH
97      C
98      8 IF(KDB(18).EQ.0)GO TO 11
99      IF(NMAX.EQ.1.AND.ICOUNT.GT.0)GO TO 105
100     NALTW=1
101     AA=AA+DAA
102     DO 9 I=1,10
103     9 DELX(I)=PHITE(I)
104     RETURN
105     95 DO 10 I=1,NOSS
106     XLAMB(18,I)=(DELX(I)-PHITE(I))/-DAA
107     10 XLAMBX(3,I)=XLAMB(18,I)
108     AA=AA-DAA
109     NALTW=0
110     GO TO 11
111     105 DO 106 I=1,NOSS
112     106 XLAMB(18,I)=XLAMBX(3,I)
113     C

```

BADLX

LIFTING-ROBOT LISTINGS

```
114 C      GET LAMBDA BETA 19 FOR ATMOS CHIDOT YAW RATE
115 C
116     11 IF(KDB(19).EQ.0)GO TO 15
117     15 IF(IPR.EQ.0)GO TO 17
118 C***  ADJUST LAMB1 FOR PERF RES
119     TEMP=PRK4/((1.-PRK4)*XMD(IPR)*(TNE(1,IPR)+TNE(3,IPR)))
120     DO 16 I=1,NOSS
121     IF(KCDPHI(I).EQ.1)YLBT(IPR,I)=0.
122     16 XLAMB(16,I)=XLAMB(16,I)-TEMP*YLBT(IPR,I)
123 C*****
124 C      END OF LAMB1 AND LAMB2
125     17 CONTINUE
126     RETURN
127     END
```

.L,0.

BAKRN

LIFTING-ROBOT LISTINGS

TPF\$.BAKRN
PUR 0018-05/17-11:50

BAKRN

LIFTING-ROBOT LISTINGS

DT*TPF5.BAKRN

```

1  SUBROUTINE BAKRN
2  COMMON/THROR/THROR
3  COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
4  COMMON/PERF/ILAST,IPR,PRK1,PRK2,PRK3,PRK4,PRTOT,FGR,FPR,WPMX
5  COMMON/REST/JSTR,NCNRS,KCDRES(6),TR,LSB,NF8,NOPAR,WIBT(17)
6  COMMON/BLAMB/YLBT(15,10),YLBW(15,10),GAPHI(10),PROD(2,10),WAH
7  COMMON/TABLK/MCAN,MCAF,MCON1,MCON2,MCON3,MCON4,MBS4,MBS7,MTT1,MIT2
8  1,MTT3,MTT4
9  COMMON/YLI/YLINT(7,10),PHITE(10),PSIREQ(10),KCDPHI(10),PSTR(10)
10 COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCSI,
11 1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
12 2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
13 COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
14 1 TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
15 2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
16 3 HNOM(15),HMAX(15)
17 COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
18 1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
19 2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
20 3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
21 4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
22 5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
23 6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
24 7 BHMAX(15),BHMN,BSTEP(15)
25 COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
26 COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
27 1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
28 2NTRG6,TV6,NTRG7,IV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
29 COMMON/NOMOR/NOMOR
30 COMMON/BAKINT/BBANK(2),TBK,TBKD,YL(7,10),YLD(7,10),BSAVE(1350),
31 1 NBTRG1,TBV1,NBTRG2,TBV2,NBTRG3,TBV3,NBTRG4,TBV4,NBTRG5,TBV5,
32 2 TEND,DTB4,DU3,D23,D43,WISSD(10,10)
33 COMMON/BAKE/WISS(10,10),YLF(7,10)
34 C INITIALIZE BACKWARD RUN
35 EXTERNAL BDR11,BUR1,BWDEV,BSAVE6,BKEND,DRJUMP
36 MTT3=1
37 MTT4=1
38 LLDSOL=5
39 IF(KINDB.NE.2)LLDSOL=3*(KRDERB+2)+4
40 TEND=T
41 TBK=0.
42 TBKD=0.
43 NOMOR=0
44 KAT=0
45 KCST=0
46 KCYTAB=KCYTAB+1
47 JTB=1
48 ITHR=NVNT+1
49 IWD=NWVNT+1
50 NENT=IXR
51 IF(IXR.EQ.0)NENT=150
52 MBS4=NENT
53 XMAUG=XMAUG+PRTOT
54 C ZERO OUT LAMBDA S
55 DO 1 J=1,7
56 DO 1 I=1,NOSS

```

BAKKN

LIFTING-ROBOT LISTINGS

```

57      1 YL(J,I)=YLINT(J,I)
58      DO 2 I=1,10
59      DO 2 J=1,I
60      2 WISS(I,J)=0.
61      K=NOSS+NCNRS
62      DO 7 I=1,NOPAR
63      DO 7 J=1,K
64      7 YLBT(I,J)=0.
65      C
66      C      STOP TRIGGER
67      NBTRG1=1
68      TBV1=TEND-TQ
69      C
70      C*2* THRUST EVENT TRIGGER (BTHREV)
71      NBTRG2=2
72      TBV2=0.
73      C
74      C*3* WEIGHT DROP EVENT TRIGGER (BWDEV)
75      NBTRG3=2
76      TBV3=TEND-TIME(2,NWVNT)
77      C
78      C*4* SAVE IMPULSE RESPONSE FUNCTIONS (BSAVEG)
79      NBTRG4=1
80      TBV4=0.
81      C
82      C      THROT TRIGGER
83      NBTRG5=-2
84      C      INITIALIZE CHI SPACING
85      C      INITIALIZE THRUST,STEP,HNOM,HMAX,MDOT,ETC.
86      CALL BTHREV
87      C      INERTIAL INTEGRATION
88      3 NACT=7*NOSS
89      CALL DESOLV(70,NACT,BDR11,BDRI,0,6,KERR,HNM,BBANK,BSAVE,
90      1  KINDB,LLDSOL,BEU,BEL,HMX,BHMN,BYL,5,
91      3      NBTRG4,BSAVEG,TBK,TBV4,
92      2NBTRG3,BWDEV,TBK,TBV3 , NBTRG2,BTHREV,TBK,TBV2,
93      4NBTRG5,DRJUMP,THKOR,1., NBTRG1,BKEND,TBK,TBV1, 0,0,0,0, 0,0,0,0,
94      5  0,0,0,0, 0,0,0,0, 0,0,0,0,0,0,0,0)
95      TPRINT=TEND-TBK
96      WRITE(6,9340)TPRINT
97      WRITE(6,9340)((YL(I,J),J=1,8),I=1,7)
98      9340 FORMAT(8(1X,E15.8))
99      IF(KERR.NE.0)GO TO 4
100     C**** NOMOR IS SET TO 1 IN BTHREV BEFORE CALL TO RTMRK IF RESTART
101     IF(NOMOR.EQ.0)RETURN
102     TR=TEND-TBK
103     NOMOR=0
104     K=NOSS+1
105     NOSS=NOSS+NCNRS
106     DO 5 J=1,7
107     DO 5 I=K,NOSS
108     5 YL(J,I)=YLINT(J,I)
109     DO 6 I=K,NOSS
110     YLBT(ITHR,I)=0.
111     DO 6 J=1,7
112     6 YLBT(ITHR,I)=YLBT(ITHR,I)+DELXDR(J)*YL(J,I)
113     GO TO 3

```

BAKRN

LIFTING-ROBOT LISTINGS

```
114      4 WRITE(6,9310)  
115 9310 FORMAT(//1X,24HINTEGRATION ERROR.DUMPED)  
116      STOP  
117      END
```

3 .L,0.



3

BDR11

LIFTING-ROBOT LISTINGS

I,S TPF\$.BDR11
PUR 0018-05/17-11:50

BDR11

LIFTING-ROBOT LISTINGS

BT*TPF\$.BDR11

```

1      SUBROUTINE BDR11
2      COMMON/THROR/THROR
3      COMMON/GRAV/GT0,G11,G22,G23,G33
4      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
5      1  TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
6      2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
7      3  HNOM(15),HMAX(15)
8      COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
9      1  SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPH1Z,GR,GTH,
10     2  RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
11     3  VIV(8),DVAR(7),DELXD(15,3),DELXDR(7),DELXD(15,7),EXIT,
12     4  DRAG1,CHIDOT,WZERO,GZERO,TLA,OHR,MAP,SS,XMACH,Q,QDOT,CAX,CNP,CN,
13     5  FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
14     6  CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
15     7  BHMAX(15),BHMN,BSTEP(15)
16     COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
17     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
18     2NTRG6,TV6,NTRG7,IV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
19     COMMON/BAKINT/BBANK(2),TBK,TBKD,YL(7,10),YLD(7,10),BSAVE(1350),
20     1  NBTRG1,TBV1,NBTRG2,TBV2,NBTRG3,TBV3,NBTRG4,TBV4,NBTRG5,TBV5,
21     2  TEND,DTB4,DU3,D23,D43,WISSD(10,10)
22     COMMON/BAKE/WISS(10,10),YLF(7,10)
23     COMMON/CONST/RAD,P1,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
24     COMMON/CALIF/TOM,ACC,XAC,YAC,ZAC,XACY,YACY,GXX,GYX,GZX,GYY,GZY,GZZ
25     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
26     1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
27     2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
28     COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
29     COMMON/TABLK/MCAN,MCAF,MCON1,MCON2,MCON3,MCON4,MBS4,MBS7,MT11,MT12
30     1,MTT3,MTT4
31     COMMON/TABS/TBL(151),XBL(151),YBL(151),ZBL(151),XMBL(151)
32     1 ,WBL(151),UBL(151),VBL(151)
33     COMMON/JTHR/JTHR(15),TBLTM(4)
34     COMMON/XMIAD/XMIAD,DELMA
35     COMMON/PENLTY/NCONPT,DXS(6)
36     COMMON/ARDC/ALT,PAM,DPDR,DRDDR,DSSDR,RHO,RHO1,RHO2,SPDS,RHO3(6)
37     COMMON/ARODTA/ACF,BCF,CCF,DADM,DBDM,DCDM
38     COMMON/XCON/UMF2,OMEGA2,FUMF1,UMFC
39     COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
40     COMMON/ATDER/DALT(6),DVR(6),DCR(6),DGR(6),DHDH1,VQSB,USW,USU,USV,
41     1  DHDVR1,DHDVR2,VDH1,VDH2
42     REAL MAP
43     T=TEND-TRK
44     C*****
45     C      GETS STATE VARIABLES FROM FORWARD RUN
46     IF(T.GE.TBL(1))GO TO 1
47     IF(T.LT.TQ+1.)GO TO 1
48     NENT=151
49     C      EQUATE TABLES(151) TO TABLES(1)
50     TBL(151)=TBL(1)
51     XBL(151)=XBL(1)
52     YBL(151)=YBL(1)
53     ZBL(151)=ZBL(1)
54     XMBL(151)=XMBL(1)
55     WBL(151)=WBL(1)
56     UBL(151)=UBL(1)

```

BDR11

LIFTING-ROBOT LISTINGS

```

57      VBL(151)=VBL(1)
58      BACKSPACE 3
59      READ( 3)(TBL(I),XBL(I),YBL(I),ZBL(I),XMBL(I),
60      * WBL(I),UBL(I),VBL(I),I=1,150)
61      BACKSPACE 3
62      IF(TBL(150).GT.TBL(151))NENT=150
63      1 IF(NCONPT.EQ.0.AND.IORR.EQ.0)GO TO 12
64      CALL BMULG7(7,MBS4,NENT,T,TBL,X,XBL,Y,YBL,Z,ZBL,XMI,XMBL,
65      * W,WBL,U,UBL,V,VBL)
66      GO TO 13
67      12 CALL BMULG7(4,MBS4,NENT,T,TBL,X,XBL,Y,YBL,Z,ZBL,XMI,XMBL,
68      * W,WBL,U,UBL,V,VBL)
69      13 CONTINUE
70      C*****
71      XMIAD=XMI
72      XM=XMIAD+XMAUG
73      R2=X*X+Y*Y+Z*Z
74      R=SQRT(R2)
75      IF(NCONPT.GT.0)CALL IMPPT(1)
76      C
77      IF(JORB.EQ.0)GO TO 2
78      G11=-CMUE/(R2*R)
79      G22=-3.*G11/R2
80      TEMP1=G22*X
81      TEMP2=G22*Y
82      TEMP3=G22*Z
83      C
84      GXX=G11+TEMP1*X
85      GYX=    TEMP1*Y
86      GZX=    TEMP1*Z
87      GYY=G11+TEMP2*Y
88      GZY=    TEMP2*Z
89      GZZ=G11+TEMP3*Z
90      IF(IORR.EQ.0)GO TO 3
91      ALT=R-RE
92      DALT(1)=X/R
93      DALT(2)=Y/R
94      DALT(3)=Z/R
95      GO TO 3
96      C
97      2 CTH=(A(1,2)*X+A(2,2)*Y+A(3,2)*Z)/R
98      CT2=CTH*CTH
99      CALL AGE0(2)
100     C
101     TEMP1=G22*X+G23*A(1,2)
102     TEMP2=G22*Y+G23*A(2,2)
103     TEMP3=G22*Z+G23*A(3,2)
104     TEMP4=G33*A(1,2)+G23*X
105     TEMP5=G33*A(2,2)+G23*Y
106     TEMP6=G33*A(3,2)+G23*Z
107     C
108     GXX=G11+TEMP1*X      +TEMP4*A(1,2)
109     GYX=    TEMP1*Y      +TEMP4*A(2,2)
110     GZX=    TEMP1*Z      +TEMP4*A(3,2)
111     GYY=G11+TEMP2*Y      +TEMP5*A(2,2)
112     GZY=    TEMP2*Z      +TEMP5*A(3,2)
113     GZZ=G11+TEMP3*Z      +TEMP6*A(3,2)

```

```

114      IF (IORR.EQ.0) GO TO 3
115      RTHE=UMF*RE/SQRT(UMF*UMF*(1.-CT2)+CT2)
116      ALT=R-RTHE
117      DRTHE=RTHE**3*UMF*CTH/R2
118      DALT(1)=X/R-(X*CTH-R*A(1,2))*DRTHE
119      DALT(2)=Y/R-(Y*CTH-R*A(2,2))*DRTHE
120      DALT(3)=Z/R-(Z*CTH-R*A(3,2))*DRTHE
121  C
122      3 CALL ACNTRO
123      IF (JTHR(ITHR).EQ.0) GO TO 4
124      MMM=IABS(JTHR(ITHR))
125      ACT=T-TBLTM(MMM)
126      CALL ATTRAC(ACT,FTRAC,XMDRAC,MMM,0.,0.,0.)
127      THR=ANE(1)*FTRAC
128  C      NO NEED FOR XMDOT
129      4 CONTINUE
130      THMFAG=GLIM(ITHR)*XM
131      THROT=1.
132      IF (THR.GT.0.) THROT=THMFAG/THR
133      THROT=THROT
134      IF (THROT.GT.1.) THROT=1.
135      ACC=THR*THROT/(XM*XM)
136      IF (KWTA.EQ.3) GO TO 5
137      XAC=CHIP
138      YAC=CCHIP
139      ZAC=0.
140      GO TO 6
141      5 XAC=CHIP*CCHIY
142      YAC=CCHIP*CCHIY
143      ZAC=CHIPY
144      6 IF (IORR.EQ.0) RETURN
145      SW=W-Z*A22W+Y*A32W
146      SU=U-X*A32W+Z*A12W
147      SV=V-Y*A12W+X*A22W
148      VR2=SW*SW+SU*SU+SV*SV
149      VR=SQRT(VR2)
150      USW=SW/VR
151      USU=SU/VR
152      USV=SV/VR
153      CALL PRB63
154  C
155      OHR=RHO
156      TLA=ALT
157      MAP=PAM
158  C
159      QSOV2=.5*RHO*S(ITHR)/XM
160      QSOV=QSOV2*VR
161      QS=QSOV*VR
162      XMACH=VR/SPDS
163      CALL CACN(XMACH,2)
164      EXM=EXIT/XM
165      BPLUSC=BCF+CCF
166      TFOR=THR*THROT/XM-QS*BPLUSC-EXM*PAM
167      VQSB=QSOV*BCF
168      ACC=TFOR/XM
169      COSA=XAC*USW+YAC*USU+ZAC*USV
170      BCAMA=BCF*COSA-ACF

```

BDR11

LIFTING-ROBOT LISTINGS

```
171      DBCAMA=DBDM*CO5A-DADM
172      DBPC=DBDM+DCDM
173      C      DHDVR AND DHDH HAVE SIGNS REVERSED
174      DHDVR1=(2.*BPLUSC+XMACH*DBPC)*QSOV2
175      DHDVR2=(ACF-XMACH*DBCAMA)*QSOV2
176      VDHDH1=EXM*DPDR+QS*(BPLUSC*DR0DR-XMACH*DBPC*DSSDR)
177      VDHDH2=QS*(BCAMA*DR0DR-XMACH*DBCAMA*DSSDR)
178      DHDH1=QSOV*BCAMA
179      DVR(4)=SV*A22W-SU*A32W
180      DVR(5)=SW*A32W-SV*A12W
181      DVR(6)=SU*A12W-SW*A22W
182      DCR(4)=ZAC*A22W-YAC*A32W
183      DCR(5)=XAC*A32W-ZAC*A12W
184      DCR(6)=YAC*A12W-XAC*A22W
185      IF(NBTRG5.GT.0)TFOR=-QS*BPLUSC-EXM*PAM
186      FX=TFOR*XAC+DHDH1*SW
187      FY=TFOR*YAC+DHDH1*SU
188      FZ=TFOR*ZAC+DHDH1*SV
189      RETURN
190      END
```

.L.O.

G

BDRI

LIFTING-ROBOT LISTINGS

T,S TPF\$.BDRI
PUR 0018-05/17-11:50

BDRI

LIFTING-ROBOT LISTINGS

OT*TPF\$.BDRI

```

1      SUBROUTINE BDRI
2      COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHLY,
3      1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPH1Z,GR,GTH,
4      2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
5      3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
6      4 DRAG1,CHIDOT,WZER0,GZER0,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
7      5 FAA,FAN,FX,FY,FZ,A(3,3),RL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
8      6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
9      7 BHMAX(15),DHMN,BSTEP(15)
10     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCS1,
11     1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
12     2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
13     COMMON/BLAMP/YLBT(15,10),YLBW(15,10),GAPHI(10),PROD(2,10),WAH
14     COMMON/BAKINT/BBANK(2),TBK,TBKD,YL(7,10),YLD(7,10),BSAVE(1350),
15     1 NBTRG1,TBV1,NBTRG2,TBV2,NBTRG3,TBV3,NBTRG4,TBV4,NBTRG5,TBV5,
16     2 TEND,DTB4,DU3,D23,D43,WISSD(10,10)
17     COMMON/BAKE/WISS(10,10),YLF(7,10)
18     COMMON/CALIF/TOM,ACC,XAC,YAC,ZAC,XACY,YACY,GXX,GYX,GZX,GYY,GZY,GZZ
19     COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
20     COMMON/PENLTY/NCONPT,UXS(6)
21     COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
22     COMMON/ATDER/DALT(6),DVR(6),DCR(6),DGR(6),DHDH1,VQSB,USW,USU,USV,
23     1 DHDVR1,DHDVR2,VDH1,VDH2
24     C LAMBDA DOTS
25     DO 2 I=1,NOSS
26     YLD(1,I)=YL(4,I)
27     YLD(2,I)=YL(5,I)
28     YLD(3,I)=YL(6,I)
29     YLD(4,I)= GXX*YL(1,I)+GYX*YL(2,I)+GZX*YL(3,I)
30     YLD(5,I)= GYX*YL(1,I)+GYY*YL(2,I)+GZY*YL(3,I)
31     YLD(6,I)= GZX*YL(1,I)+GZY*YL(2,I)+GZZ*YL(3,I)
32     IF(IORR.EQ.1)GO TO 3
33     YLD(7,I)=- (XAC*YL(1,I)+YAC*YL(2,I)+ZAC*YL(3,I))*ACC
34     IF(NBTRG5.GT.0)YLD(7,I)=-YL(7,I)*XMDOT*GLIM(ITHR)/THR
35     GO TO 4
36     3 CONTINUE
37     CVDL=XAC*YL(1,I)+YAC*YL(2,I)+ZAC*YL(3,I)
38     VDOTL=USW*YL(1,I)+USU*YL(2,I)+USV*YL(3,I)
39     AC1=CVDL*VDH1-VDOTL*VDH2
40     AC2=CVDL*DHDVR1+VDOTL*DHDVR2
41     AC3=DHDH1
42     AC4=VQSB*VDOTL
43     DGR(4)=YL(3,I)*A22W-YL(2,I)*A32W
44     DGR(5)=YL(1,I)*A32W-YL(3,I)*A12W
45     DGR(6)=YL(2,I)*A12W-YL(1,I)*A22W
46     C
47     C NOTE THAT SIGNS HAVE BEEN REVERSED ON ADDITIONS BECAUSE OF
48     C BACKWARDS INTEGRATION
49     C
50     YLD(1,I)=YLD(1,I)-(AC2*SW-AC3*YL(1,I)-AC4*XAC)
51     YLD(2,I)=YLD(2,I)-(AC2*SU-AC3*YL(2,I)-AC4*YAC)
52     YLD(3,I)=YLD(3,I)-(AC2*SV-AC3*YL(3,I)-AC4*ZAC)
53     YLD(4,I)=YLD(4,I)-(AC1*DALT(1)+AC2*DVR(4)-AC3*DGR(4)-AC4*DCR(4) )
54     YLD(5,I)=YLD(5,I)-(AC1*DALT(2)+AC2*DVR(5)-AC3*DGR(5)-AC4*DCR(5) )
55     YLD(6,I)=YLD(6,I)-(AC1*DALT(3)+AC2*DVR(6)-AC3*DGR(6)-AC4*DCR(6) )
56     YLD(7,I)=- (FX*YL(1,I)+FY*YL(2,I)+FZ*YL(3,I))/XM

```

BDRI

LIFTING-ROBOT LISTINGS

```
57      IF (NBTRG5.GT.0) YLD(7,I)=YLD(7,I)-YL(7,I)*XMDOT*GLIM(ITHR)/THR
58      4 CONTINUE
59      2 CONTINUE
60      IF (NCONPT.EQ.0) RETURN
61      I=NCONPT
62      YLD(1,I)=YLD(1,I)+DXS(4)
63      YLD(2,I)=YLD(2,I)+DXS(5)
64      YLD(3,I)=YLD(3,I)+DXS(6)
65      YLD(4,I)=YLD(4,I)+DXS(1)
66      YLD(5,I)=YLD(5,I)+DXS(2)
67      YLD(6,I)=YLD(6,I)+DXS(3)
68      RETURN
69      END
```

.L,0.

BKEND

LIFTING-ROBOT LISTINGS

F,S TPF\$.BKEND
PUR 0018-05/17-11:50

BKEND

LIFTING-ROBOT LISTINGS

JT*TPF\$.BKEND

```

1      SUBROUTINE BKEND
2      COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
3      COMMON/FLORTE/FLORTE(15),DLFLRT(15)
4      COMMON/YLI/YLINT(7,10),PHITE(10),PSIREQ(10),KCDPHI(10),PSTR(10)
5      COMMON/QY/QY,CFR(15)
6      COMMON/PERF/ILAST,IPR,PRK1,PRK2,PRK3,PRK4,PRTOT,FGR,FPR,WPMX
7      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRA6,TLIFT,TTILT,
8      1  TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
9      2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
10     3  HNOM(15),HMAX(15)
11     COMMON/BLAMB/YLBT(15,10),YLBW(15,10),GAPHI(10),PROD(2,10),WAH
12     COMMON/BGEN3/IRIIE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
13     COMMON/BAKINT/BBANK(2),TBK,TBKD,YL(7,10),YLD(7,10),BSAVE(1350),
14     1  NBTRG1,TBV1,NBTRG2,TBV2,NBTRG3,TBV3,NBTRG4,TBV4,NBTRG5,TBV5,
15     2  TEND,DTB4,DU3,D23,D43,WISSD(10,10)
16     COMMON/BAKE/WISS(10,10),YLF(7,10)
17     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
18     1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
19     2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
20     COMMON/AUTO/KDB(30),SAVCP(30),XLAMB(30,10),DP2,KDT(30)
21     IF(JTB.NE.1)CALL BSAVEG
22     C      ADD WD LAMBDA BETA TO THRUST LAMBDA BETA
23     DO 1 I=1,NWVNT
24     K=NOWD(I)
25     DO 1 J=1,NOSS
26     C      ADDS WD LAMBDA BETAS TO APPROPRIATE THRUST LAMBDA BETAS
27     1  YLBT(K,J)=YLBT(K,J)+YLBW(I,J)
28     C      *****
29     C      ADD UP ALL LAMBS DOWNSTREAM
30     KV=NVNT-1
31     3  TEMP=1.
32     DO 99 I=1,NVNT
33     99  WRITE(6,9943)(YLBT(I,J),J=1,NOSS)
34     9943  FORMAT(5(1X,E15.8))
35     DO 4 K=1,NOSS
36     4  YLBT(KV,K)=YLBT(KV,K)+YLBT(KV+1,K)*TEMP
37     KV=KV-1
38     IF(KV.GT.0)GO TO 3
39     C****  TINKER LAMBDA BETAS IF CONNECTED
40     DO 5 I=1,NVNT
41     IF(KDB(I).EQ.0)GO TO 5
42     IF(KDT(I).EQ.0)GO TO 11
43     J=I+KDT(I)
44     TEMP=CFR(I)/CFR(J)*(TNE(1,I)+TNE(3,I))/(TNE(1,J)+TNE(3,J))
45     IF(I.LT.ILAST.OR.IPR.EQ.0)GO TO 9
46     TEMPR=(1.-CFR(J)*XMD(I)/CFR(I)*PRK4/XMD(J))/(1.-PRK4)
47     TEMP=TEMP*TEMPR
48     9  DO 10 K=1,NOSS
49     TEMP1=TEMP*YLBT(J,K)
50     IF(KCDPHI(K).NE.1)GO TO 10
51     IF(I.GE.ILAST.AND.IPR.NE.0)TEMP1=TEMP1/TEMPR
52     10  YLBT(I,K)=YLBT(I,K)-TEMP1
53     GO TO 5
54     C      ADJUST UNBOUNDED LAMBDA BETAS FOR PR
55     11  J=IPR
56     IF(J.EQ.0)GO TO 5

```

BKEND

LIFTING-ROBOT LISTINGS

```
57      TEMP=XMD(I)/XMD(J)*(TNE(1,I)+TNE(3,I))/(TNE(1,J)+TNE(3,J))
58      DO 12 K=1,NOSS
59      IF(KCDPHI(K).EQ.1)GO TO 12
60      YLBT(I,K)=YLBT(I,K)+TEMP*PRK4*YLBT(J,K)/(1.-PRK4)
61      12 CONTINUE
62      5 CONTINUE
63  C *****
64  C*** PUT ACTIVE YLBTS INTO XLAMB
65      DO 7 I=1,NVNT
66      DO 6 J=1,NOSS
67      6 XLAMB(I ,J)=YLBT(I,J)
68      7 CONTINUE
69      DO 8 I=1,7
70      DO 8 J=1,NOSS
71      8 YLF(I,J)=YL(I,J)
72      CALL RTMRK
73      RETURN
74      END
```

5 .L,0.



5

BMUL67

LIFTING-ROBOT LISTINGS



T,S TPF\$.BMUL67
PUR 0018-05/17-11:51



BMULG7

LIFTING-ROBOT LISTINGS

DT*TPF\$,BMULG7

```

1      SUBROUTINE BMULG7(L,M,N,X,XT,Y1,Y1T,Y2,Y2T,Y3,Y3T,Y4,Y4T,Y5,Y5T,
2      * Y6,Y6T,Y7,Y7T)
3      DIMENSION XT(1),Y1T(1),Y2T(1),Y3T(1),Y4T(1),Y5T(1),Y6T(1),Y7T(1)
4      IF(M.LT.1.OR.M.GT.N)M=1
5      DIFF=X-XT(M)
6      IF(DIFF)1,3,2
7      C   GET HERE IF X.LT.XT(M)
8      1   IF(M.EQ.1)GO TO 3
9          M=M-1
10         IF(X-XT(M))1,3,4
11      C   GET HERE IF X.GT.XT(M)
12      2   IF(M.EQ.N)GO TO 3
13         MP1=M+1
14         IF(XT(MP1).LT.XT(M))GO TO 3
15         M=MP1
16         IF(X-XT(M))5,3,2
17      3   Y1=Y1T(M)
18         Y2=Y2T(M)
19         Y3=Y3T(M)
20         Y4=Y4T(M)
21         IF(L.EQ.4)RETURN
22         Y5=Y5T(M)
23         Y6=Y6T(M)
24         Y7=Y7T(M)
25         RETURN
26      4   M=M+1
27      5   MM1=M-1
28         DEL=(X-XT(MM1))/(XT(M)-XT(MM1))
29         Y1=Y1T(MM1)+DEL*(Y1T(M)-Y1T(MM1))
30         Y2=Y2T(MM1)+DEL*(Y2T(M)-Y2T(MM1))
31         Y3=Y3T(MM1)+DEL*(Y3T(M)-Y3T(MM1))
32         Y4=Y4T(MM1)+DEL*(Y4T(M)-Y4T(MM1))
33         IF(L.EQ.4)RETURN
34         Y5=Y5T(MM1)+DEL*(Y5T(M)-Y5T(MM1))
35         Y6=Y6T(MM1)+DEL*(Y6T(M)-Y6T(MM1))
36         Y7=Y7T(MM1)+DEL*(Y7T(M)-Y7T(MM1))
37         RETURN
38         END

```

6 .L,0.

BOPTBL

LIFTING-ROBOT LISTINGS

T,S TPF\$.BOPTBL
PUR 0018-05/17-11:51

BOPTBL

LIFTING-ROBOT LISTINGS

OT*TPF\$.BOPTBL

```

1      SUBROUTINE BOPTBL
2      DIMENSION OFFICE(2),TITLE(8),DATE(2)
3      DIMENSION DVARF(3),DVAR(3),SRID(60)
4      ENDPRT=6HENDPRT
5      BOOST=5HBOOST
6      THRUST=6HTHRUST
7      WEIGHT=6HWEIGHT
8      BLANK=6H
9      DO 3000 I=1,60
10     3000 SRID(I)=BLANK
11     DO 1100 I=1,2
12     OFFICE(I)=BLANK
13     1100 DATE(I)=BLANK
14     DO 1101 I=1,8
15     1101 TITLE(I)=BLANK
16     RAD=57.29578
17     NERR=10
18     NAMELIST/INPUT2/NCASE,OFFICE,TITLE,DATE,SRID
19     READ(5,INPUT2)
20     WRITE(6,INPUT2)
21     NUMB=NCASE
22     NOT=0
23     NTAB=1
24     1500 NOT=NOT+1
25     11 REWIND 9
26     C NEW PAGE
27     LINES=1
28     KT=0
29     KW=0
30     102 READ(9)AT,B,T,VI,VR,GAM,GAMR,XM,XMLB,THE,THEP,
31     A XLACC,ALTP,R,XMACH,Q,ALFP,FAA,AZI,CPDG,CYDG,CHIR,AH1,X,Y,Z,
32     B W,U,V,ALFQ,THETA,PHI,(DVAR(I),I=1,3),XNOD,RNGO,RNGAN,ADEG,THIMP,
33     C PHIMP
34     LINES=LINES-1
35     IF(AT.EQ.ENDPRT)GO TO 1500
36     IF(AT.EQ.BOOST)WRITE(6,70)
37     3004 IF(AT.NE.THRUST)GO TO 3005
38     KT=KT+1
39     AT=SRID(KT*2-1)
40     B=SRID(KT*2)
41     GO TO 3003
42     3005 IF(AT.NE.WEIGHT)GO TO 3003
43     KW=KW+1
44     AT=SRID(KW*2+29)
45     B=SRID(KW*2+30)
46     3003 GO TO (100,200,300,400,500,4000,600,700,800,900,1000),NOT
47     C** HEADER FOR TABLE 1
48     100 IF(LINES.NE.0)GO TO 111
49     110 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
50     WRITE(6,101)
51     LINES=34
52     101 FORMAT(/17X,4HTIME,6X,17HINERTIAL VELOCITY,11X,
53     A 17HRELATIVE VELOCITY,4X,7HGAMMA S,3X,7HGAMMA R,18X,3HSEC,6X,
54     B 5HM/SEC,6X,6HFT/SEC,11X,5HM/SEC,6X,6HFT/SEC,6X,3HDEG,
55     C /X,3HDEG,/)
56     111 VIF=VI*3.2808399

```

```

57      VRF=VR*3.2808399
58      WRITE(6,103)AT,B,T,VI,VIF,VR,VRF,GAM,GAMR
59      GO TO 102
60  C*** HEADER FOR TABLE 2
61      200 IF(LINES.NE.0)GO TO 211
62      210 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
63          WRITE(6,201)
64          LINES=34
65      201 FORMAT(/,17X,4HTIME,8X,4HMASS,8X,6HWEIGHT,11X,6HTHRUST,13X,
66          A 20HLONGIT. ACCELERATION,/,18X,3HSEC,8X,3HKG,10X,3HLBS,11X,1HN,
67          B 9X,3HLBS,7X,21HM/(SEC)SQ FT/(SEC)SQ,/)
68      211 XLACCF=XLACC*3.2808399
69          WRITE(6,203) AT,B,T,XM,XMLB,THE,THEP,XLACC,XLACCF
70          GO TO 102
71  C*** HEADER FOR TABLE 3
72      300 IF(LINES.NE.0)GO TO 311
73      310 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
74          WRITE(6,301)
75          LINES=34
76      301 FORMAT(/,17X,4HTIME,13X,8HALTITUDE,21X,6HRADIUS,16X,4HINCL,/,18X,
77          A 3HSEC,11X,1HM,10X,2HFT,13X,1HM,12X,2HFT,12X,3HDEG,/)
78      311 ALTPF=ALTP*3.2808399
79          RF=R*3.2808399
80          TEMP=THETA/RAD
81          TEMP1=AZI/RAD
82          XINC=(ACOS(COS(TEMP)*SIN(TEMP1)))*RAD
83          WRITE(6,303)AT,B,T,ALTP,ALTPF,R,RF,XINC
84          GO TO 102
85      400 IF(LINES.NE.0)GO TO 411
86  C*** HEADER FOR TABLE 4
87      410 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
88          WRITE(6,401)
89          LINES=34
90      401 FORMAT(/,17X,4HTIME,7X,4HMACH,7X,13HDYN. PRESSURE, 9X,4HALFP,12X,
91          B 4HDRA,/,18X,3HSEC,15X,20HN/(M)SQ LB/(FT)SQ,6X,3HDEG,10X,1HN,
92          C 11X,3HLBS,/)
93      411 QLB=Q/47.8802589
94          FAALB=FAA*.22480894
95          WRITE(6,403)AT,B,T,XMACH,Q,QLB,ALFP,FAA,FAALB
96          GO TO 102
97  C*** HEADER FOR TABLE 5
98      500 IF(LINES.NE.0)GO TO 511
99      510 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
100          WRITE(6,501)
101          LINES=34
102      501 FORMAT(/,17X,4HTIME,7X,4HAZ.S,6X,8HLATITUDE,3X,9HLONGITUDE,
103          A 4X,4HCHIP,7X,4HCHIY,7X,4HCHIR,/,18X,3HSEC,8X,3HDEG,8X,3HDEG,
104          B 9X,3HDEG,8X,3HDEG,8X,3HDEG,8X,3HDEG,/)
105      511 WRITE(6,503)AT,B,T,AZI,THETA,PHI,CPDG,CYDG,CHIR
106          GO TO 102
107  C*** HEADER FOR TABLE 6
108      4000 IF(LINES.NE.0)GO TO 4011
109      4010 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
110          WRITE(6,4001)
111          LINES=34
112      4001 FORMAT(/,16X,4HTIME,9X,5HRANGE,6X,10HRANGE ANG.,5X,11HTOTAL ALPHA,
113          A 4X,9HIMP. LAT.,4X,10HIMP. LONG.,/,17X,3HSEC,10X,2HKM,10X,3HDEG,12X,

```

BOPTBL

LIFTING-ROBOT LISTINGS

```

114      B3HDEG,11X,3HDEG,10X,3HDEG,/)
115 4011 RNGOF=RNGO/1000.
116      WRITE(6,4003)AT,B,T,RNGOF,RNGAN,ADEG,THIMP,PHIMP
117      GO TO 102
118 C***  HEADER FOR TABLE 7
119      600 IF(LINES.NE.0)GO TO 611
120      610 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
121          WRITE(6,601)
122          LINES=34
123      601 FORMAT(/17X,4HTIME,11X,19HAERO HEATING INDIC.,15X,7HALPHA Q,11X,9
124          AHDES. NODE,/18X,3HSEC,8X,25HN-M/(M)SQ      LB-FT/(FT)SQ,6X,19H N*RAD
125          B      LB*DEG,8X,3HDEG,/)
126      611 AHIF=AH1*.3048/4.44822162
127          ALFQP=ALFQ*.22480894*RAD
128          WRITE(6,603)AT,B,T,AH1,AHIF,ALFQ,ALFQP,XNOD
129          GO TO 102
130 C***  HEADER FOR TABLE 8
131      700 IF(LINES.NE.0)GO TO 711
132      710 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
133          WRITE(6,701)
134          LINES=34
135      701 FORMAT(/39X,36HAPOLLO 13 POSITION COORDINATES,
136          A/17X,4HTIME,14X,3H X,21X,3H Y,21X,3H Z,/18X,3HSEC,
137          B10X,1HM,10X,2HFT,11X,1HM,10X,2HFT,11X,1HM,10X,2HFT,/)
138      711 XF=X*3.2808399
139          YF=Y*3.2808399
140          ZF=Z*3.2808399
141          WRITE(6,703)AT,B,T,Y,YF,Z,ZF,X,XF
142          GO TO 102
143 C***  HEADER FOR TABLE 9
144      800 IF(LINES.NE.0)GO TO 811
145      810 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
146          WRITE(6,801)
147          LINES=34
148      801 FORMAT(/39X,36HAPOLLO 13 VELOCITY COMPONENTS ,
149          A/17X,4HTIME,14X,4HXDOT,20X,4HYDOT,19X,4HZDOT,/
150          B18X,3HSEC,7X,5HM/SEC,7X,6HFT/SEC,7X,5HM/SEC,6X,6HFT/SEC,6X,
151          C5HM/SEC,5X,6HFT/SEC,/)
152      811 WF=W*3.2808399
153          UF=U*3.2808399
154          VF=V*3.2808399
155          WRITE(6,803)AT,B,T,U,UF,V,VF,W,WF
156          GO TO 102
157 C***  HEADER FOR TABLE 10
158      900 IF(LINES.NE.0)GO TO 911
159      910 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
160          WRITE(6,901)
161          LINES=34
162      901 FORMAT(/39X,34HAPOLLO 13 ACCELERATION COMPONENTS ,
163          A/17X,4HTIME,15X,5HXDDOT,20X,5HYDDOT,18X,5HZDDOT,/18X,
164          B3HSEC,7X,8HM/SEC**2,4X,9HFT/SEC**2,4X,8HM/SEC**2,3X,9HFT/SEC**2,
165          C3X,8HM/SEC**2,3X,9HFT/SEC**2,/)
166      911 DO 65 I=1,3
167          65 DVARF(I)=DVAR(I)*3.2808399
168          WRITE(6,903)AT,B,T,DVAR(2),DVARF(2),DVAR(3),DVARF(3),
169          1DVAR(1),DVARF(1)
170          GO TO 102

```

BOPTBL

LIFTING-ROBOT LISTINGS

```

171 1000 WRITE(6,80)OFFICE,DATE,TITLE,NUMB,NOT
172     GO TO (1001,1002,1003),NTAB
173 1001 WRITE(6,2000)
174 2000 FORMAT(/,21X45HDEFINITIONS AND SYMBOLS FOR TRAJECTORY TABLES//1X6
175     1HSYMBOL7X5HTABLE2X5HUNITS7X10HDEFINITION//1X4HTIME9X3HALL4X7HSECON
176     2US5X31HINSTANTANEOUS TIME FROM LIFTOFF//1X8HINERTIAL5X1H16X5HM/SEC
177     37X53HVEHICLE VELOCITY IN THE SPACE FIXED COORDINATE SYSTEM/2X8HVEL
178     4OCITY11X6HFT/SEC//1X8HRELATIVE5X1H16X5HM/SEC7X53HVEHICLE VELOCITY
179     5IN THE EARTH FIXED COORDINATE SYSTEM/2X8HVELOCITY11X6HFT/SEC//1X7H
180     6GAMMA 56X1H16X7HDEGREES5X51HFLIGHT PATH ANGLE,MEASURED FROM LOCAL
181     7HORIZONTAL TO/33X28HTHE INERTIAL VELOCITY VECTOR//1X7HGAMMA R6X1H1
182     86X7HDEGREES5X51HFLIGHT PATH ANGLE,MEASURED FROM LOCAL HORIZONTAL T
183     90/33X28HTHE RELATIVE VELOCITY VECTOR)
184     WRITE(6,2001)
185 2001 FORMAT(/1X4HMASS9X1H26X9HKILOGRAMS3X22HVEHICLE MASS AT TIME T//1X6
186     1HWEIGHT7X1H26X6HPOUNDS6X24HVEHICLE WEIGHT AT TIME T//1X6H1HKRUST7X1
187     2H26X7HNEWTONS5X24HVEHICLE THRUST AT TIME T/21X6HPOUNDS//1X7HLONGIT
188     3.6X1H26X8HM/SEC**24X48HACCELERATION ALONG THE LONGITUDINAL VEHICL
189     4E AXIS/2X12HACCELERATION7X9HFT/SEC**2//1X8HALTITUDE5X1H36X6HMETERS
190     56X39HDISTANCE FROM CENTER OF EARTH AT TIME T/21X4HFEET//1X4HINCL,
191     69X1H36X7HDEGREES5X25HINSTANTANEOUS INCLINATION//5H MACH9X1H46X4H--
192     7--8X38HMACH NUMBER,BASED ON RELATIVE VELOCITY//1X3HDYN10X1H46X6HN/
193     6M**2,6X41HDYNAMIC PRESSURE DUE TO RELATIVE VELOCITY/2X8HPRESSURE,
194     911X8HLB/FT**2)
195     NTAB=2
196     GO TO 1000
197 1002 WRITE(6,2002)
198 2002 FORMAT(/21X17HTABLE 11 (CONT'D)//1X6HSYMBOL,7X5HTABLE2X5HUNITS7X10
199     1HDEFINITION//1X4HALFP9X1H46X7HDEGREES5X64HRELATIVE PITCH ATTITUDE
200     2ANGLE MEASURED FROM RELATIVE VELOCITY IN/33X65HATMOSPHERE.INERTIAL
201     3 PITCH ATTITUDE ANGLE FOR OUT OF ATMOS. FLIGHT//1X,4HDRA6,9X,1H46X
202     41HN11X26HAXIAL FORCE OF THE VEHICLE /21X3HLBS //1X4HAZ.59X1H5
203     46X7HDEGREES5X55HAZIMUTH ANGLE,MEASURED IN SPACE FIXED SYSTEM FROM
204     5NORTH/33X18HTO SOUTH OVER EAST//1X8HLATITUDE5X1H56X7HDEGREES5X49HG
205     6EOCENTRIC LATITUDE OF VEHICLE POSITION AT TIME T//1X9HLONGITUDE,
206     74X1H56X7HDEGREES5X42HEARTH RELATIVE LONGITUDE AT TIME T (+EAST)//
207     81X4HCHI1P9X1H56X7HDEGREES5X32HAPOLLO 13 PITCH ATTITUDE COMMAND//
208     91X4HCHI1Y9X1H56X7HDEGREES5X30HAPOLLO 13 YAW ATTITUDE COMMAND//1X5HR
209     1ANGE8X1H66X2HKM10X60HSURFACE RANGE FROM THE INSTANTANEOUS LAUNCH S
210     2ITE POSITION TO/33X20HTHE SUBVEHICLE POINT//1X10HRANGE ANG.3X1H66X
211     33HDEG9X61HINERTIAL RANGE ANGLE MEASURED FROM THE INERTIAL RADIUS V
212     4ECTOR/ 33X60HTHROUGH THE LAUNCH SITE AT LIFTOFF TO THE VEHICLE GE
213     5OCENTRIC/33X13HRADIUS VECTOR//1X11HTOTAL ALPHA2X1H66X3HDEG9X59HOUR
214     6ING ATMOSPHERIC FLIGHT THE ANGLE BETWEEN THE AERODYNAMIC/33X59HVEL
215     7OCITY VECTOR AND THE THRUST VECTOR. DURING VACUUM FLIGHT/33X54HTHE
216     8 ANGLE BETWEEN THE INERTIAL VELOCITY VECTOR AND THE/33X13HTHRUST V
217     9ECTOR//1X9HIMP. LAT.4X1H66X3HDEG9X28HLATITUDE OF INSTANTANEOUS VA)
218     WRITE(6,4500)
219 4500 FORMAT(1H+,60X17HCUUM IMPACT POINT //
220     1 1X10HIMP. LONG.3X1H66X3HDEG9X55HEARTH RELATIVE LO
221     2NGITUDE OF INSTANTANEOUS VACUUM IMPACT/33X13HPOINT (+EAST))
222     NTAB=3
223     GO TO 1000
224 1003 WRITE(6,2004)
225 2004 FORMAT(/21X17HTABLE 11 (CONT'D)//1X6HSYMBOL7X5HTABLE2X5HUNITS,
226     17X10HDEFINITION// 1X4HA 1X4HA
227     3ER016X8HN=M/M**2/2X7HHEATING5X1H718X44HINTEGRAL [(Q*VR/(PI/2-ALPHA

```

BOPTBL

LIFTING-ROBOT LISTINGS

```

228      4) )DTJ FROM 0 TO T/3X9HINDICATOR9X11HLB-FT/FT**2, //
229      51X7HALPHA*Q6X1H76X8HN-RADIAN4X47HPRODUCT OF ANGLE OF ATTACK AND DY
230      6NAMIC PRESSURE/21X9HLB-DEGREE//
231      61X9HDES. NODE4X1H76X3HDEG9X56HANGLE TO THE DESCENDING NODE OF THE
232      6INSTANTANEOUS FLIGHT/33X58HPLANE MEASURED FROM THE LAUNCH MERIDIAN
233      6 AT LIFTOFF (+EAST) //1X1HX12X1H86X1HM11X30HAPOLLO 13 POS
234      7ITION COORDINATES/21X2HFT//1X1HY12X1H86X1HM11X30HAPOLLO 13 POSITIO
235      8N COORDINATES/21X2HFT//
236      9 1X1HZ12X1H86X1HM11X30HAPOLLO 13 POSITION COORDINA
237      1TES/21X2HFT//1X4HXDOT9X1H96X5HM/SEC7X29HAPOLLO 13 VELOCITYCOMPONEN
238      2TS /21X6HFT/SEC//1X4HYDOT9X1H96X5HM/SEC7X29HAPOLLO 13 VELOCITY
239      3 COMPONENTS/21X6HFT/SEC//1X4HZDOT9X1H96X5HM/SEC7X29HAPOLLO 13 VELO
240      4CITY COMPONENTS/21X6HFT/SEC//1X5HXDDOT8X2H106X8HM/SEC**24X33HAPOLL
241      50 13 ACCELERATION COMPONENTS/21X9HFT/SEC**2//1X5HYDDOT8X2H106X8HM/
242      6SEC**24X33HAPOLLO 13 ACCELERATION COMPONENTS/21X9HFT/SEC**2//1X5HZ
243      7DDOT8X2H106X8HM/SEC**24X33HAPOLLO 13 ACCELERATION COMPONENTS/21X9H
244      8FT/SEC**2)
245      WRITE(6,2005)
246      70 FORMAT(//)
247      80 FORMAT(1H1, //39X, 2A6, 25X, 2A6, //21X, 8A6, 15X, 6H CASE, 13, //61X,
248      19HTABLE NO., I3)
249      103 FORMAT(1X, 2A6, F10.3, F11.3, F12.3, F15.3, F12.3, F10.3, F9.3)
250      203 FORMAT(1X, 2A6, F10.3, F12.1, F13.1, F12.1, F13.1, F11.2, F11.2)
251      303 FORMAT(1X, 2A6, F10.3, F12.2, F13.1, F16.2, F14.1, F12.4)
252      FAALB=FAA/4.44799965
253      403 FORMAT(1X, 2A6, F10.3, F10.4, F11.3, F11.3, F11.3, F13.3, F13.3)
254      503 FORMAT(1X, 2A6, F10.3, F11.3, F12.3, 4F11.3)
255      4003 FORMAT(1X, 2A6, F9.3, F13.3, F15.3, F14.3, F14.3, F13.3)
256      603 FORMAT(1X, 2A6, F10.3, F16.1, F13.1, F14.3, F13.3, F13.3)
257      703 FORMAT(1X, 2A6, F10.3, F12.1, F13.1, F12.1, F12.1, 2F11.1)
258      803 FORMAT(1X, 2A6, F10.3, F11.3, F13.3, F12.3, F12.3, 2F11.3)
259      903 FORMAT(1X, 2A6, F10.3, F11.3, F13.3, F12.3, F12.3, 2F11.3)
260      2005 FORMAT(1H1)
261      RETURN
262      END

```

3 .L.O.

BSAVEG

LIFTING-ROBOT LISTINGS

T,S TPF\$.BSAVEG
PUR 0018-05/17-11:51

BSAVEG

LIFTING-ROBOT LISTINGS

I*TPF\$.BSAVEG

```

1      SUBROUTINE BSAVEG
2      COMMON/CONVEC/XCV,YCV,ZCV
3      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,ILIFT,ITILT,
4      1  TCHFRZ,DTZ,TQ,IL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
5      2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
6      3  HNOM(15),HMAX(15)
7      COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
8      COMMON/BLAMB/YLBT(15,10),YLBW(15,10),GAPHI(10),PROD(2,10),WAH
9      COMMON/BGEN3/IRIE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
10     COMMON/BAKINT/BBANK(2),TBK,TBKD,ZL(7,10),YLD(7,10),BSAVE(1350),
11     1  NBTRG1,TBV1,NBTRG2,TBV2,NBTRG3,TBV3,NBTRG4,TBV4,NBTRG5,TBV5,
12     2  TEND,DTB4,DU3,D23,D43,WISSD(10,10)
13     COMMON/BAKE/WISS(10,10),YLF(7,10)
14     COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIY,CHLY,
15     1  SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
16     2  RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
17     3  VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
18     4  DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
19     5  FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
20     6  CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDR,KRDEKB,BEU,BYL,
21     7  BHMN(15),BHMN,BSTEP(15)
22     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
23     1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
24     2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
25     COMMON/FORINT/HBANK(2),T,TT,w,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
26     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
27     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
28     COMMON/CALIF/TOM,ACC,XAC,YAC,ZAC,XACY,YACY,GXX,GYX,GZX,GYY,GZY,GZZ
29     COMMON/IMP/GAPT(200,10),GSPT(200,10),AHTB(200),S1TB(200),IS(4),
30     1  DELTS(4),KJTB
31     COMMON/COVL/COVL(10)
32     COMMON/ATDER/DAL(6),DVR(6),DCR(6),DGR(6),DHDH1,VQSB,USW,USU,USV,
33     1  DHDVR1,DHDVR2,VDHDH1,VDHDH2
34     DIMENSION GSPHI(10)
35     DIMENSION YL(7,10)
36
37     C      XCV=XAC
38     C      YCV=YAC
39     C      ZCV=ZAC
40
41     C      DO 22 I=1,7
42     C      DO 22 J=1,NOSS
43     22  YL(I,J)=ZL(I,J)
44     C      TOM=ACC*XM
45     C      IF(IORR.EQ.0)GO TO 21
46
47     C      CONSTRUCT EQUIV YL VECTOR
48     C
49     C      DO 23 J=1,NOSS
50     C      CVDL=ZL(1,J)*SW+ZL(2,J)*SU+ZL(3,J)*SV
51     C      CVDL=CVDL*VQSB/TOM
52     C      YL(1,J)=YL(1,J)+CVDL*USW
53     C      YL(2,J)=YL(2,J)+CVDL*USU
54     23  YL(3,J)=YL(3,J)+CVDL*USV
55
56     C      21 CONTINUE

```

```

57 C 30 FORMAT(3X,8E14.6)
58   TBV4=TBV4+DTB4
59   JTB=JTB-1
60   JJ=KJTB+JTB
61 C   IMPULSE RESPONSE FUNCTIONS AND 1-SY-SY DOTS
62   IF(NOM1.NE.0)GO TO 4
63   IF(KWTA.EQ.2)GO TO 1
64 C   GET HERE ONLY ON NOMINAL WITH KWTA=3
65   HV11=1.
66   HV12=0.
67   HV22=1.
68   GO TO 8
69   1 WAH=1.
70   2 TMW=TOM*WAH
71   DO 3 I=1,NOSS
72     GAPHI(I)=YL(1,I)*CCHIP-YL(2,I)*SCHIP
73     DO 3 J=1,I
74   3 WISSD(1,J)=TMW*GAPHI(I)*GAPHI(J)
75   GO TO 11
76 C   GET COV LAMBDAS
77   4 NOSSM1=NOSS-1
78   DO 5 I=1,7
79     COVL(I)=YL(I,1)
80     DO 5 J=1,NOSSM1
81   5 COVL(I)=COVL(I)+GNU(J)*YL(I,J+1)
82 C   DHDA1=TOM*(COVL(1)*SCHIP*CCHIIY+COVL(2)*CCHIP*CCHIIY+COVL(3)*SCHIIY)
83 C   DHDA2=-COVL(7)*(TNE(1,ITHR)+TNE(3,ITHR))*XMD(ITHR)
84 C   DHDA3=DHDA1+DHDA2
85 C   WRITE(6,30)T,DHDA1,DHDA2,DHDA3
86 C   GOT COV LAMBDAS
87   IF(KWTA.EQ.3)GO TO 6
88   WAH=1./(XJEXT*(COVL(1)*SCHIP+COVL(2)*CCHIP))
89   TEMP=1./SQRT(COVL(1)**2+COVL(2)**2)
90   IF(TEMP/WAH.GT..25)GO TO 2
91   KAT=1
92   WAH=TEMP
93   GO TO 2
94 C   3-D MIN-H
95 C   CHECK HUU FOR POSITIVE DEFINITE
96   6 A11=(COVL(1)*XAC+COVL(2)*YAC)*XJEXT
97     A12=(COVL(1)*CCHIP-COVL(2)*SCHIP)*XJEXT*SCHIIY
98     A22=A11+COVL(3)*SCHIIY*XJEXT
99     DET=A11*A22-A12*A12
100    TDET=COVL(1)*COVL(1)+COVL(2)*COVL(2)
101    IF(A11.GT.0..AND.A22.GT.0..AND.DET.GT.TDET/2.)GO TO 7
102    DET=TDET
103    A22=(DET+COVL(3)*COVL(3))**.5
104    A11=DET/A22
105    A12=0.
106    KAT=1
107 C   HUU INVERSE
108   7 HV11= A22/DET
109     HV12=-A12/DET
110     HV22= A11/DET
111 C
112   8 DO 9 I=1,NOSS
113     GAPHI(I)= YL(1,I)*YAC -YL(2,I)*XAC

```

BSAVEG

LIFTING-ROBOT LISTINGS

```

114      GSPHI(I)=- (YL(1,I)*SCHIP+YL(2,I)*CCHIP)*SCHIY+YL(3,I)*CCHIY
115      PROD(1,I)=HV11*GAPHI(I)+HV12*GSPHI(I)
116      PROD(2,I)=HV12*GAPHI(I)+HV22*GSPHI(I)
117      DO 9 J=1,I
118      9 WISSD(I,J)=TOM*(GAPHI(J)*PROD(1,I)+GSPHI(J)*PROD(2,I))
119      DO 10 I=1,NOSS
120      GAPTB(JJ,I)=PROD(1,I)
121      10 GSPTB(JJ,I)=PROD(2,I)
122      SITB(JJ)=CHIY
123      GO TO 13
124      11 DO 12 I=1,NOSS
125      12 GAPTB(JJ,I)=WAH*GAPHI(I)
126      13 AHTB(JJ)=CHIP
C      ** INTEGRATE ISYSY BY SIMPSON'S RULE--
C      **      A=H/3*(Y1+4*Y2+2*Y3+...+4*YN-1 +YN)
129      IF(JTB.LT.NP(KCYTAB))GO TO 15
C      GET HERE AT FIRST POINT (MULT BY 1/3)
131      LL=2
132      DO 14 I=1,NOSS
133      DO 14 J=1,I
134      14 WISS(I,J)=WISS(I,J)+ DU3 *WISSD(I,J)
135      RETURN
136      15 IF(LL.EQ.1)GO TO 17
C      GET HERE AT AN EVEN POINT (MULT BY 4/3)
138      LL=1
139      DO 16 I=1,NOSS
140      DO 16 J=1,I
141      16 WISS(I,J)=WISS(I,J)+ D43 *WISSD(I,J)
142      RETURN
143      17 IF(JTB.EQ.1)GO TO 19
C      GET HERE AT AN ODD POINT (MULT BY 2/3)
145      LL=2
146      DO 18 I=1,NOSS
147      DO 18 J=1,I
148      18 WISS(I,J)=WISS(I,J)+ D23 *WISSD(I,J)
149      RETURN
150      C      GET HERE AT LAST POINT (JTB=1) (MULT BY 1/3)
151      19 DO 20 I=1,NOSS
152      DO 20 J=1,I
153      20 WISS(I,J)=WISS(I,J)+ DU3 *WISSD(I,J)
154      NBTRG4=-1
155      TS(KCYTAB)=T
156      DELTS(KCYTAB)=DTb4
157      RETURN
158      END

```

.L,0.

BTCST

LIFTING-ROBOT LISTINGS

I,S TPF\$.BTCST
PUR 0018-05/17-11:51

BTCST

LIFTING-ROBOT LISTINGS

DT*TPF\$.BTCST

```

1      SUBROUTINE BTCST(Z,XM, TD,DYDX)
2      DOUBLE PRECISION X(6), S(6),XM,R,RZ,SZ,P,A,TD,TG,F,G,FD,GD,
3      A DT,DYDX(6,6)
4      DIMENSION Z(6)
5      DO 5 I=1,6
6      5 X(I)=Z(I)
7      RZ=SQRT(Z(1)*Z(1)+Z(2)*Z(2)+Z(3)*Z(3))
8      A=X(4)*X(4)+X(5)*X(5)+X(6)*X(6)-2.*XM/RZ
9      SZ=X(1)*X(4)+X(2)*X(5)+X(3)*X(6)
10     P=DABS(XM/A)
11     P=TD/DMAX1(RZ,P)
12     DO 10 J=1,25
13     CALL GETS(A,P,S)
14     R=RZ*S(1)+SZ*S(2)+XM*S(3)
15     TG=RZ*S(2)+SZ*S(3)+XM*S(4)
16     DT=TG-TD
17     IF (DABS(DT /TD ).LT.5.D-12)GO TO 20
18     10 P=P-DT/R
19     WRITE(6,40)TD,TG,P,R
20     F=1.D0-XM *S(3)/RZ
21     G=RZ*S(2)+SZ *S(3)
22     FD=-XM *S(2)/(R *RZ)
23     GD=1.D0-XM *S(3)/R
24     CALL GETLAM(TD,X,XM, TG,A,S,P,RZ,SZ,F,FD,G,GD,R,DYDX)
25     40 FORMAT(17H CANNOT GET PSI,4D21.14)
26     RETURN
27     END

```

.L,0.

BTHREV

LIFTING-ROBOT LISTINGS

T,S TPF\$.BTHREV
PUR 0018-05/17-11:51

BTHREV

LIFTING-ROBOT LISTINGS

DT*TPF\$.BTHREV

```

1      SUBROUTINE BTHREV
2      COMMON/THROR/THROR
3      COMMON/ABEL/FUEL(15),MSWCH(15),GLIM(15),THROT
4      COMMON/COAST/NCOST1,NCOST2,CX(6,2)
5      COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
6      1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
7      2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
8      3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
9      4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
10     5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
11     6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
12     7 BHMAX(15),BHMN,BSTEP(15)
13     COMMON/BAKINT/BBANK(2),TBK,TBKD,YL(7,10),YLD(7,10),BSAVE(1350),
14     1 NBTRG1,TBV1,NBTRG2,TBV2,NBTRG3,TBV3,NBTRG4,TBV4,NBTRG5,TBV5,
15     2 TEND,DTB4,DU3,D23,D43,WISSD(10,10)
16     COMMON/BAKE/WISS(10,10),YLF(7,10)
17     COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
18     COMMON/BLAMB/YLBT(15,10),YLBW(15,10),GAPHI(10),PROD(2,10),WAH
19     COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
20     1 TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
21     2 WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
22     3 HNOM(15),HMAX(15)
23     COMMON/AGEN3/IORR,IHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
24     1 KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
25     2 NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
26     COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
27     COMMON/IMP/GAPTB(200,10),GSPTB(200,10),AHTB(200),SITB(200),TS(4),
28     1 DELTS(4),KJTB
29     COMMON/NOMOR/NOMOR
30     DOUBLE PRECISION TAUD,XMUE,DYDX(6,6)
31     DIMENSION DYDXR(6,6)
32 C     BACKWARDS THRUST EVENT ROUTINE
33     CALL APRTN(1)
34     WRITE(6,9340)((YL(I,J),J=1,8),I=1,7)
35     9340 FORMAT(8(1X,E15.8))
36     101 IF(IHR.EQ.JUMP)GO TO 6
37 C*** CHECK FOR END(BEGINNING) OF CHI TABLE BEFORE UPDATING THRUST
38 C     CHECK ON CHI TABLES
39     IF(KCYTAB.EQ.1)GO TO 1
40     IF(NENDCT(KCYTAB-1).NE.IHR )GO TO 1
41 C * HAVE ARRIVED AT THE END OF A CHI TABLE
42     IF(JTB.NE.1)CALL BSAVEG
43     KCYTAB=KCYTAB-1
44     NBTRG4=1
45     TBV4=TBK
46     JTB=NP(KCYTAB)+1
47     KJTB=(KCYTAB-1)*50
48     SP=NP(KCYTAB)-1
49     NN=NBGCT(KCYTAB)
50     N=NENDCT(KCYTAB)
51     DTB4=(TIME(1,N)-AMAX1(TQ,TIME(1,NN)))/SP
52     DU3=DTB4/3.
53     D23=DU3*2.
54     D43=DU3*4.
55     1 J=NBGCT(KCYTAB)
56     IF(J.EQ.IHR.AND.JTB.NE.1)CALL BSAVEG

```

```

57 C      END OF CHI TABLE CHECK
58      ITHR=ITHR-1
59      HNM=BSSTEP(ITHR)
60      HMX=2.*PRINT(ITHR)
61      ANE(1)=TNE(1,ITHR)*COS(TNE(2,ITHR)/RAD) +TNE(3,ITHR)*
62      1 COS(TNE(4,ITHR)/RAD)
63      XMAUG=XMAUG+WJET(ITHR)
64      THR=ANE(1)*F(ITHR)
65      EXIT=ANE(1)*AE(ITHR)
66      XMDOT=(TNE(1,ITHR)+TNE(3,ITHR))*XMD(ITHR)
67      IF(ITHR.LE.NOEVNT(1))THR=THR+EXIT*PSL
68      KCST=0
69      IF(THR.EQ.0..AND.IORR.EQ.0)KCST=1
70      CALL BDR1I
71      NBTRG5=-2
72      IF(THROR.LT.1.)NBTRG5=2
73 C      GET LAMBDA BETAS
74      DO 2 I=1,NOSS
75      YLBT(ITHR,I)=0.
76      DO 2 J=1,7
77
78      2 YLBT(ITHR,I)=YLBI(ITHR,I)+DELXD(ITHR,J)*YL(J,I)
79 C      GOT LAMBDA BETAS
80      TBV2=TEND-TIME(1,ITHR)
81      IF(ITHR.NE.NVRST)GO TO 5
82      NOMOR=1
83      CALL RTMRK
84      5 IF(KCST.EQ.0)RETURN
85      IF(ITHR.NE.NCOST1 .AND. ITHR.NE.NCOST2)RETURN
86 C
87 C *** TRANSFORM LAMBDA'S ACROSS COAST ***
88      L=1
89      IF(ITHR.EQ.NCOST2)L=2
90      XMUE=CMUE
91      TAUD=TAUT(ITHR)
92      CALL BTCST(CX(1,L),XMUE,TAUD,DYDX)
93      DO 95 I=1,3
94      DO 95 J=1,3
95      DYDXR( I,J)=DYDX(I+3,J+3)
96      DYDXR( I+3,J)=DYDX(I,J+3)
97      DYDXR( I,J+3)=DYDX(I+3,J)
98      95 DYDXR( I+3,J+3)=DYDX(I,J)
99      DO 8 I=1,6
100     DO 8 J=1,NOSS
101     YLF(I,J)=0.
102     DO 8 K=1,6
103     8 YLF(I,J)=YLF(I,J)+DYDXR( K,I)*YL(K,J)
104     DO 9 I=1,6
105     DO 9 J=1,6
106     9 YL(I,J)=YLF(I,J)
107     TBK=TBK+TAUT(ITHR)
108     GO TO 101
109 C
110     6 CALL BKEND
111     RETURN
112     END

```

BTHREV

LIFTING-ROBOT LISTINGS

3

.L.O.

G

BWDEV

LIFTING-ROBOT LISTINGS

T,S TPF\$.BWDEV
PUR 0018-05/17-11:51

BWDEV

LIFTING-ROBOT LISTINGS

BT*TPF\$.BWDEV

```

1      SUBROUTINE BWDEV
2      COMMON/KLIFT/KLIFT
3      COMMON/BLAMB/YLBT(15,10),YLBW(15,10),GAPHI(10),PROD(2,10),WAH
4      COMMON/BAKINT/BBANK(2),TBK,TBKD,YL(7,10),YLD(7,10),BSAVE(1350),
5      1  NBTRG1,TBV1,NBTRG2,TBV2,NBTRG3,TBV3,NBTRG4,TBV4,NBTRG5,TBV5,
6      2  TEND,DTB4,DU3,D23,D43,WISSD(10,10)
7      COMMON/BAKE/WISS(10,10),YLF(7,10)
8      COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,CHIP,CHIP,CHIP,
9      1  SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
10     2  RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
11     3  VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
12     4  DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
13     5  FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),B0(3,3),D(3,3),DQDX(6),CASE,
14     6  CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
15     7  BHMAX(15),BHMN,BSTEP(15)
16     COMMON/BGEN3/IRITE,LSTIR,NOSS,NOSET,KK,JTB,NENT,NACT,NOM1,NP(4)
17     COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
18     1  TCHFRZ,DTZ,TQ,IL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
19     2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
20     3  HNOM(15),HMAX(15)
21     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
22     1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
23     2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
24     IWD=IWD-1
25     XMAUG=XMAUG+WD(IWD)
26     C   GET LAMBDA BETAS
27     DO 1 I=1,NOSS
28     YLBW(IWD,I)=0.
29     DO 1 J=1,3
30     1  YLBW(IWD,I)=YLBW(IWD,I)+DELXDW(IWD,J)*YL(J,I)
31     IF(IWD.EQ.1)GO TO 2
32     TBV3=TEND-TIME(2,IWD-1)
33     GO TO 3
34     2  NBTRG3=-2
35     3  IF(IWD.NE.IWDCHI)RETURN
36     C   HAVE REACHED CHI RELEASE. END ADJOINT INTEGRATION.
37     IF(KLIFT.EQ.1)GO TO 4
38     NBTRG5=-1
39     XMAUG=XMAUG-WD(IWD)
40     IF(JTB.NE.1)CALL BSAVE6
41     CALL BKEND
42     RETURN
43     4  IORR=1
44     RETURN
45     END

```

G .L,0.

CACN

LIFTING-ROBOT LISTINGS

T,S TPF\$,CACN
PUR 0018-05/17-11:51

CACN

LIFTING-ROBOT LISTINGS

OT*TPF\$.CACN

```

1      SUBROUTINE CACN(M,L)
2          C
3          C      AERO DATA -- A(M),B(M) AND C(M) -- CONF 19 BOOSTER HCR ORBITER
4          C
5          REAL M,MT
6          COMMON/ARODTA/ACF,BCF,CCF,DADM,DBDM,DCDM
7          COMMON/TABLK/K,MMM(11)
8          DIMENSION MT(20),ATB(20),BTB(20),CTB(20)
9          DATA N/12/
10         DATA( MT(I),I=1,12)/0.,.5,.85,1.,1.5,2.,3.,4.,5.,6.,7.,20./
11         DATA(ATB(I),I=1,12)/2.896,4.852,8.422,7.917,4.781,4.853,
12         1 4.386,3.330,2.809,2.651,2.643,2.643/
13         DATA(BTB(I),I=1,12)/.2754,1.692,4.598,4.043,1.948,2.563,
14         1 2.761,2.069,1.734,1.644,1.644,1.644/
15         DATA(CTB(I),I=1,12)/-2.796,-4.786,-8.322,-7.767,-4.641,-4.740,
16         1 -4.308,-3.2720,-2.765,-2.619,-2.619,-2.619/
17         CALL SPLINE(L,K,N, M , MT,ACF,ATB,BCF,BTB,CCF,CTB,DADM,DBDM,DCDM)
18         RETURN
19         END

```

.L,0.



3

DERIMP

LIFTING-ROBOT LISTINGS



T,S TPF\$.DERIMP
PUR 0018-05/17-11:51



DERIMP

LIFTING-ROBOT LISTINGS

OT*TPF\$.DERIMP

```

1      SUBROUTINE DERIMP
2      COMMON/ARDC/PAT(15)
3      COMMON/GRAV/GTO,G11,G22,G23,G33
4      COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
5      COMMON/AGEN1/TIME(2,16),TAUT(15),TAUW(15),TZERO,TDRAG,TLIFT,TTILT,
6      1  TCHFRZ,DTZ,TQ,TL,XMAUG,ANE(2),TNE(4,15),AE(15),S(15),WD(15),
7      2  WJET(15),XMD(15),F(15),CHIFRZ,HEAD(15),PRINT(15),STEP(15),
8      3  HNOM(15),HMAX(15)
9      COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
10     1  SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
11     2  RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
12     3  VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
13     4  DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
14     5  FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
15     6  CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
16     7  BHMAX(15),BHMN,BSTEP(15)
17     COMMON/AGEN3/IORR,ITHR,IWD,IWDCHI,IXR,JORB,JUMP,KAT,KCST,
18     1  KCYTAB,KDIR,KPAGE,KPRT,KS1,KWTA,LINES,NBGCT(4),NENDCT(4),NMAX,
19     2  NOEVNT(5),IPCNST,NOWD(15),NVNT,NVRST,NWVNT
20     COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
21     1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
22     2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
23     K2=X*X+Y*Y+Z*Z
24     R=R2**.5
25     IF(JORB.EQ.0)GO TO 1
26     G11=-CMUE/(R*R2)
27     GX=X*G11
28     GY=Y*G11
29     GZ=Z*G11
30     ALT=R-RE
31     GO TO 2
32     1 CTH=(A(1,2)*X+A(2,2)*Y+A(3,2)*Z)/R
33     CT2=CTH*CTH
34     CALL AGE0(1)
35     GX=X*G11-A(1,2)*GTO
36     GY=Y*G11-A(2,2)*GTO
37     GZ=Z*G11-A(3,2)*GTO
38     RTHE=UMF*RE/(UMF*UMF*(1.-CT2)+CT2)**.5
39     ALT=R-RTHE
40     2 SW=W-(Z*A22W-Y*A32W)
41     SU=U-(X*A32W-Z*A12W)
42     SV=V-(Y*A12W-X*A22W)
43     VR=(SW*SW+SU*SU+SV*SV)**.5
44
45     C*****
46     PAT(1)=ALT
47     RHO=0.
48     XMACH=0.
49     IF(ALT.GT.690000.)GO TO 3
50     CALL PRA63(PAT(1),IER)
51     RHO=PAT(6)
52     XMACH=VR/PAT(9)
53     3 Q=.5*RHO*VR*VR
54     FAA=2.*Q*S(ITHR)
55     VRXM=VR*XM
56     DVAR(1)=GX-FAA*SW/VRXM

```

DERIMP

LIFTING-ROBOT LISTINGS

57 DVAR(2)=GY-FAA*SU/VRXM
58 DVAR(3)=GZ-FAA*SV/VRXM
59 DVAR(4)=W
60 DVAR(5)=U
61 DVAR(6)=V
62 RETURN
63 END

3 .L,0.

DRJUMP

LIFTING-ROBOT LISTINGS

T,S TPF\$.DRJUMP
POR 0018-05/17-11:51

DRJUMP

LIFTING-ROBOT LISTINGS

DT*TPF\$.DRJUMP

```

1  SUBROUTINE DRJUMP
2  COMMON/BAKINT/BBANK(2),TBK,TBKD,YL(7,10),YLD(7,10),BSAVE(1350),
3  1  NBTRG1,TBV1,NBTRG2,TBV2,NBTRG3,TBV3,NBTRG4,TBV4,NBTRG5,TBV5,
4  2  TEND,DTB4,DU3,D23,D43,WISSD(10,10)
5  NBTRG5=-2
6  CALL APRTN(1)
7  RETURN
8  END

```

5 .L,0.

3 FORKM

LIFTING-ROBOT LISTINGS

I,S TPF\$.FORKM
PUR 0018-05/17-11:51

FORKM

LIFTING-ROBOT LISTINGS

DT*TPF\$.FORKM

```

1      SUBROUTINE FORKM
2      COMMON/DFEQS/      N,XN,X,H,II,YN(6),Y(6),Y1(6),DY(6)
3      DIMENSION YNP1(6),B1(6),B2(6),B3(6),B4(6)
4      C
5      C      THIS SUBROUTINE IS A FOURTH ORDER RUNGE KUTTA DIFFERENTIAL
6      C
7      C      EQUATION SOLVER
8      C
9      Y1(1) =Y1(1)
10     GO TO (650,652,655,668,671),II
11     C      PRINT COMPUTED DATA
12     511  XN=XN+H
13         DO 10 J=1,N
14         10  YN(J)= YNP1(J)
15             II=6
16             RETURN
17     C      RUNGE-KUTTA EQUATIONS
18     650  X=XN
19         DT=H
20         DO 651 J=1,N
21         651  Y(J)=YN(J)
22             II=2
23             RETURN
24         652  DO 653 J=1,N
25         653  B1(J)=DT*DY(J)
26             X=XN + DT/ 2.0
27             DO 654 J=1,N
28         654  Y(J)=YN(J)+ B1(J)/2.0
29             II=3
30             RETURN
31         655  DO 666 J=1,N
32         666  B2(J)=DT*DY(J)
33             DO 667 J=1,N
34         667  Y(J)=YN(J) + B2(J)/2.0
35             II=4
36             RETURN
37         668  DO 669 J=1,N
38         669  B3(J)=DT*DY(J)
39             X=XN+DT
40             DO 670 J=1,N
41         670  Y(J)=YN(J) + B3(J)
42             II=5
43             RETURN
44         671  DO 672 J=1,N
45         672  B4(J)=DT*DY(J)
46             DO 673 J=1,N
47         673  YNP1(J)=YN(J)+(B1(J) + 2.0*B2(J)+2.0*B3(J)+B4(J))/6.0
48             GO TO 511
49             END

```

235D

236B

236B

236

236

346

665

.L,0.

GETAM

LIFTING-ROBOT LISTINGS

I, S TPF\$.GETAM
PUR 0018-05/17-11:51

GETAM

LIFTING-ROBOT LISTINGS

OT*TPF\$.GETAM

```

1  SUBROUTINE GET AM(X,S,PSI,SIGZ,F,FD,G,GD,RR,RZ,XMUE,DYDX)
2  DIMENSION DYDC(2,3),DCDX(3,6),XC(3,3),X(6),S(6),DYDX(6,6)
3  DS1DA=(PSI*S(3)-S(4))*0.5
4  DS2DA=(PSI*S(4)-2.*S(5))*0.5
5  DS3DA=(PSI*S(5)-3.*S(6))*0.5
6  DTAUDA=RZ*DS1DA+SIGZ*DS2DA+XMUE*DS3DA
7  GTEMP=1.-GD
8  DYDC(1,1)=(1.-F)/RZ-FD*S(2)
9  DYDC(1,2)=-FD*S(3)
10 DYDC(1,3)=-(FD*DTAUDA+XMUE*DS2DA/RZ)
11 DYDC(2,1)=GTEMP*S(2)
12 DYDC(2,2)=GTEMP*S(3)
13 DYDC(2,3)=GTEMP*DTAUDA-XMUE*DS3DA
14 TEMPT=2.*XMUE/(RZ*RZ*RZ)
15 DO 3 I=1,3
16   DCDX(1,I)=X(I+3)/RZ
17   DCDX(1,I+3)=0.
18   DCDX(2,I)=X(I)
19   DCDX(2,I+3)=X(I+3)
20   DCDX(3,I)=TEMPT*X(I+3)
21 3 DCDX(3,I+3)=2.*X(I)
22 DO 4 I=1,3
23   DO 4 J=1,6
24 4 DYDX(I,J)=0.
25   DO 5 I=1,3
26   DYDX(I,I)=F
27   DYDX(I,I+3)=G
28   DO 5 J=1,3
29 5 XC(I,J)=X(I+3)*DYDC(1,J)+X(I)*DYDC(2,J)
30   DO 6 I=1,3
31   DO 6 J=1,6
32   DO 6 K=1,3
33 6 DYDX(I,J)=DYDX(I,J)+XC(I,K)*DCDX(K,J)
34 RETURN
35 END

```

3 .L.O.



3

GETLAM

LIFTING-ROBOT LISTINGS



1,5 TPF\$.GETLAM
PUR 0018-05/17-11:51



GETLAM

LIFTING-ROBOT LISTINGS

JT*TPF\$.GETLAM

```

1  SUBROUTINE GETLAM(TAUD,X,XMUE, TAUG,ALPH,S,PSI,RZ,SIGZ,F,FD,G,
2  1 GD,RR,DYDX)
3  DOUBLE PRECISION PSI, RZ,SIGZ,XMUE,F,FD,G,GD,RR, ALPH
4  DOUBLE PRECISION DS1DA,DS2DA,DS3DA,DTAUDA,DYDC(2,3),GTEMP,DRDA,
5  1 DRDSY,DYDDC(2,3),TERM,TEMPT,DCDX(3,6) ,XC(3,3),XCD(3,3),
6  2 TAUD,X(6), S(6),DYDX(6,6)
7  DS1DA=(PSI*S(3)-S(4))/2.D0
8  DS2DA=(PSI*S(4)-2.D0*S(5))/2.D0
9  DS3DA=(PSI*S(5)-3.D0*S(6))/2.D0
10 DTAUDA=RZ*DS1DA+SIGZ*DS2DA+XMUE*DS3DA
11 GTEMP=1.D0-GD
12 DRDA=RZ*S(3)+SIGZ*DS1DA+(XMUE+RZ*ALPH)*DS2DA
13 DRDSY=SIGZ*S(1)+(XMUE+RZ*ALPH)*S(2)
14 TERM=(GTEMP*DRDSY-XMUE*S(2))/RR
15 DYDC(1,1)=(1.D0-F)/RZ-FD*S(2)
16 DYDC(1,2)=-FD*S(3)
17 DYDC(1,3)=-(FD*DTAUDA+XMUE*DS2DA/RZ)
18 DYDC(2,1)=GTEMP*S(2)
19 DYDC(2,2)=GTEMP*S(3)
20 DYDC(2,3)=GTEMP*DTAUDA-XMUE*DS3DA
21 DYDDC(1,1)=FD*(DRDSY*S(2)/RR**2-1.D0/RZ-2.D0*S(1)/RR)
22 DYDDC(1,2)=((1.D0-F)*S(1)/RR+FD*S(3)*DRDSY/RR-FD*S(2))/RR
23 DYDDC(1,3)=((XMUE*S(1)/RZ+FD*DRDSY)*DTAUDA/RR-XMUE*DS1DA/RZ
24 1 -FD*DRDA)/RR
25 DYDDC(2,1)=(GTEMP*S(1)-S(2)*TERM )/RR
26 DYDDC(2,2)=(GTEMP*S(2)-S(3)*TERM )/RR
27 DYDDC(2,3)=(GTEMP*DRDA-XMUE*DS2DA-TERM*DTAUDA )/RR
28 TEMPT=2.D0*XMUE/(RZ*RZ*RZ)
29 DO 3 I=1,3
30 DCDX(1,I)=X(I)/RZ
31 DCDX(1,I+3)=0.D0
32 DCDX(2,I)=X(I+3)
33 DCDX(2,I+3)=X(I)
34 DCDX(3,I)=TEMPT*X(I)
35 3 DCDX(3,I+3)=2.D0*X(I+3)
36 DO 4 I=1,6
37 DO 4 J=1,6
38 4 DYDX(I,J)=0.D0
39 DO 5 I=1,3
40 DYDX(I,I)=F
41 DYDX(I,I+3)=G
42 DYDX(I+3,I)=FD
43 DYDX(I+3,I+3)=GD
44 DO 5 J=1,3
45 XC(I,J)=X(I)*DYDC(1,J)+X(I+3)*DYDC(2,J)
46 5 XCD(I,J)=X(I)*DYDDC(1,J)+X(I+3)*DYDDC(2,J)
47 DO 6 I=1,3
48 DO 6 J=1,6
49 DO 6 K=1,3
50 DYDX(I,J)=DYDX(I,J)+XC(I,K)*DCDX(K,J)
51 6 DYDX(I+3,J)=DYDX(I+3,J)+XCD(I,K)*DCDX(K,J)
52 RETURN
53 END

```

GETLAM

LIFTING-ROBOT LISTINGS

.L.O.

GETS

LIFTING-ROBOT LISTINGS

T,S TPF\$.GETS
PUR 0018-05/17-11:51

GETS

LIFTING-ROBOT LISTINGS

OT*TPF\$.GETS

```

1  SUBROUTINE GETS(ALPH,PSI,S)
2  DOUBLE PRECISION A,U,AP,C0,C1,C2,C3,C4,C5,ALPH,PSI,PSI2,S(6)
3  U=1.D0
4  PSI2=PSI*PSI
5  M=0
6  A=ALPH*PSI2
7  IF(DABS(A).LT.U)GO TO 120
8  AP=A
9  110 M=M+1
10  A=A/4.D0
11  IF(DABS(A).GT.U)GO TO 110
12  120 C5=(U+(U+(U+(U+(U+(U+A/342.D0)*A/272.D0)*A/210.D0)*A/156.D0)
13  1 *A/110.D0)*A/72.D0)*A/42.D0)/120.D0
14  C4=(U+(U+(U+(U+(U+(U+A/306.D0)*A/240.D0)*A/182.D0)*A/132.D0)
15  1 *A/90.D0)*A/56.D0)*A/30.D0)/24.D0
16  C3=U/6.D0+A*C5
17  C2=U/2.D0+A*C4
18  C1=U+A*C3
19  C0=U+A*C2
20  IF(M.LE.0)GO TO 140
21  130 C1=C1*C0
22  C0=2.D0*C0*C0-U
23  M=M-1
24  IF(M.GT.0)GO TO 130
25  C2=(C0-U)/AP
26  C3=(C1-U)/AP
27  C4=(C2-U/2.D0)/AP
28  C5=(C3-U/6.D0)/AP
29  140 S(1)=C0
30  S(2)=C1*PSI
31  S(3)=C2*PSI2
32  S(4)=C3*PSI*PSI2
33  S(5)=C4*PSI2*PSI2
34  S(6)=C5*PSI2*PSI2*PSI
35  RETURN
36  END

```

.L,0.

GROUND

LIFTING-ROBOT LISTINGS

T,S TPF\$.GROUND
PUR 0018-05/17-11:51

GROUND

LIFTING-ROBOT LISTINGS

DT*TPF\$.GROUND

```
1  SUBROUTINE GROUND
2  CALL APRTN(4)
3  CALL RTMRK
4  RETURN
5  END
```

3 .L,0.

IMPPT

LIFTING-ROBOT LISTINGS

T,S TPF\$.IMPPT
PUR 0018-05/17-11:51

IMPPT

LIFTING-ROBOT LISTINGS

DT*TPF\$.IMPPT

```

1      SUBROUTINE IMPPT(INDC)
2      COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
3      COMMON/FORINT/HBANK(2),T,TT,W,U,V,X,Y,Z,XMI,ZAP(8),DVAR(15),
4      1FSAVE(375),NTRG1,TV1,NTRG2,TV2,NTRG3,TV3,NTRG4,TV4,NTRG5,TV5,
5      2NTRG6,TV6,NTRG7,TV7,KIND,KRDER,EU,EL,AYL,HMX,HMN,HNM
6      COMMON/ALONG0/ALONG0
7      COMMON/AGEN1/DUM1(67),DTZ,DUM2(231)
8      COMMON/AGEN2/DUM3(25),R,DUM4(207),A(3,3),DUM5(85)
9      COMMON/AGEN3/DUM6(30),IPCNST,DUM7(18)
10     COMMON/PENLTY/NCONPT,DXS(6)
11     COMMON/IMDATA/NOI,LLA(10),LLO(10),S2LO(10),S2LA(10),RH12(10)
12     DIMENSIONS(6),DELO(10),DELA(10),ZETA(10),CNF(10),DXP(10),DYDX(6,6)
13     DIMENSION EK2(10),ELPS(10)
14     REAL LLO,LLA
15     DATA NOI/1/
16     DATA (LLA(I),I=1,10)/10*.57/
17     DATA (LLO(I),I=1,10)/10*-1.3/
18     DATA (S2LO(I),I=1,10)/1.,2.,3.,4.,5.,6.,7.,8.,9.,10./
19     DATA (S2LA(I),I=1,10)/2.,3.,4.,5.,6.,7.,8.,9.,10.,11./
20     DATA (RH12(I),I=1,10)/10*.1/
21     DATA (CNF(I),I=1,10)/10*1./
22     DATA (EK2(I),I=1,10)/10*2./
23     TWOMU=2.*CMUE
24     DO 10 I=1,6
25         10 DXS(I)=0.
26         DVAR(8)=0.
27         IF(IPCNST.EQ.0) RETURN
28         IF(R.LT.RE+10.)RETURN
29         RDV=X*W+Y*U+Z*V
30         V2=W*W+U*U+V*V
31         RV2=R*V2
32         AL=R/(TWOMU-RV2)
33         AI=CMUE*AL
34         P=(R*RV2-RDV*RDV)/CMUE
35         ECC=1.-P/AI
36         IF(ECC.GT.1.E-6)GO TO 100
37         RETURN
38     100 ECC=SQRT(ECC)
39         AE=AI*ECC
40         IF(AI-AE.GT.RE-1000.)RETURN
41         CDT=(AI-RE)/AE
42         CDN=(AI-R)/AE
43         CDNSQ=CDN*CDN
44         IF(CDNSQ.LT.1.)GO TO 20
45         IF(CDN.LT.0.)GO TO 17
46         CDN=1.
47         SDN=0.
48         GO TO 21
49     17  CDN=-1.
50         SDN=0.
51         GO TO 21
52     20  SDN=SQRT(1.-CDNSQ)
53     21  IF(CDT.LT.1.)GO TO 30
54         CDT=1.
55         SDT=0.
56         GO TO 31

```

IMPPT

LIFTING-ROBOT LISTINGS

```

57 30 SDT=SQRT(1.-CDT*CDT)
58 31 S(1)=CDT*CDN-SDT*SDN
59 SDE=-(SDT*CDN+SDN*CDT)
60 SQAL=SQRT(AL)
61 S(2)=SDE*SQAL
62 S(3)=AL*(1.-S(1))
63 DELE=ATAN2(SDE,S(1))
64 IF(DELE.LT.0.)DELE=DELE+2.*PI
65 S(4)=(DELE-SDE)*AL*SQAL
66 F=1.-CMUE*S(3)/R
67 G=R*S(2)+RDV*S(3)
68 XP=F*X+G*W
69 YP=F*Y+G*U
70 ZP=F*Z+G*V
71 UTI=R*S(2)+RDV*S(3)+CMUE*S(4)
72 STH=(XP*A(1,2)+YP*A(2,2)+ZP*A(3,2))/RE
73 XX=XP*A(1,1)+YP*A(2,1)+ZP*A(3,1)
74 ZZ=XP*A(1,3)+YP*A(2,3)+ZP*A(3,3)
75 THIMP=ASIN(STH)
76 PHIMP=ATAN2(XX,ZZ)
77 IF(PHIMP.LT.0.)PHIMP=PHIMP+2.*PI
78 PHIMP=PHIMP-OMEGA*(DTI+DTZ+T)-ALONGO/RAD
79 DO 1 I=1,NOI
80 DELO(I)=PHIMP-LL0(I)
81 DELA(I)=THIMP-LLA(I)
82 ELPS(I) = S2L0(I)*DELO(I)*DELO(I) + RH12(I)*DELO(I)*DELA(I)
83 1 + S2LA(I)*DELA(I)*DELA(I)
84 IF(ELPS(I).GT.1.) GO TO 11
85 ZETA(I) = EXP(-ELPS(I))*CNF(I)
86 GO TO 1
87 11 ZETA(I) = CNF(I)*EXP(EK2(I) - 1. - EK2(I)*ELPS(I))
88 1 DVAR(8)=DVAR(8)+ZETA(I)
89 IF(INDC.EQ.0)RETURN
90 C
91 C GET PARTIALS
92 C
93 PP1=0.
94 PP2=0.
95 DO 2 I=1,NOI
96 PPA = ZETA(I)*(2.*DELA(I)*S2LA(I) + RH12(I)*DELO(I)) ,
97 PPO = ZETA(I)*(2.*DELO(I)*S2L0(I) + RH12(I)*DELA(I))
98 IF(ELPS(I).LT.1.) GO TO 19
99 PPA = PPA*EK2(I)
100 PPO = PPO*EK2(I)
101 19 PP1 = PP1 - PPA
102 2 PP2 = PP2 - PPO
103 DEN=XX*XX+ZZ*ZZ
104 TA=PP2/DEN
105 TB=PP1/SQRT(DEN)
106 PRX=(ZZ*A(1,1)-XX*A(1,3))*TA+TB*A(1,2)
107 PRY=(ZZ*A(2,1)-XX*A(2,3))*TA+TB*A(2,2)
108 PRZ=(ZZ*A(3,1)-XX*A(3,3))*TA+TB*A(3,2)
109 FD=-CMUE*S(2)/(RE*R)
110 GD=1.-CMUE*S(3)/RE
111 WP=FD*X+GD*W
112 UP=FD*Y+GD*U
113 VP=FD*Z+GD*V

```

IMPPT

LIFTING-ROBOT LISTINGS

```
114      TEMP=PRX*WP+PRY*UP+PRZ*VP
115      TEMP=(PP2*OMEGA-TEMP)/(XP*WP+YP*UP+ZP*VP)
116      DXP(1)=PRX+TEMP*XP
117      DXP(2)=PRY+TEMP*YP
118      DXP(3)=PRZ+TEMP*ZP
119      PSI=SQAL*DELE
120      PSI2=PSI*PSI
121      S(5)=AL*(PSI2/2.-S(3))
122      S(6)=AL*(PSI2*PSI/6.-S(4))
123      C
124      CALL GETAM(W,S,PSI,RDV ,F,FD,G,GD,RE,R,CMUE,DYDX)
125      C
126      DO 3 I=1,6
127      DO 3 J=1,3
128      3 DXS(I)=DXS(I)+DXP(J)*DYDX(J,I)
129      RETURN
130      END
```

, .L,0.

MATINV

LIFTING-ROBOT LISTINGS

3

T,S TPF\$.MATINV
PUR 0018-05/17-11:51

MATINV

LIFTING-ROBOT LISTINGS

DT*TPF\$.MATINV

```

1      SUBROUTINE MATINV(A,N)
2      C      MATRIX INVERSION
3      DIMENSION IPIVOT(9),A(9,9),INDEX(9,2),PIVOT(9)
4      EQUIVALENCE (IROW,JROW), (ICOLUMN,JCOLUMN), (AMAX, T, SWAP)
5      C
6      C      INITIALIZATION
7      C
8      DO 20 J=1,N
9      20 IPIVOT(J)=0
10     30 DO 550 I=1,N
11     C
12     C      SEARCH FOR PIVOT ELEMENT
13     C
14     40 AMAX=0.0
15     45 DO 105 J=1,N
16     50 IF (IPIVOT(J)-1) 60, 105, 60
17     60 DO 100 K=1,N
18     70 IF (IPIVOT(K)-1) 80, 100, 730
19     80 IF( ABS(AMAX)- ABS(A(J,K))) 85,100,100
20     85 IROW=J
21     90 ICOLUMN=K
22     95 AMAX=A(J,K)
23     100 CONTINUE
24     105 CONTINUE
25     110 IPIVOT(ICOLUMN)=IPIVOT(ICOLUMN)+1
26     C
27     C      INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
28     C
29     130 IF (IROW-ICOLUMN) 140, 260, 140
30     140 DO 200 L=1,N
31     160 SWAP=A(IROW,L)
32     170 A(IROW,L)=A(ICOLUMN,L)
33     200 A(ICOLUMN,L)=SWAP
34     260 INDEX(I,1)=IROW
35     270 INDEX(I,2)=ICOLUMN
36     310 PIVOT(I)=A(ICOLUMN,ICOLUMN)
37     C
38     C      DIVIDE PIVOT ROW BY PIVOT ELEMENT
39     C
40     330 A(ICOLUMN,ICOLUMN)=1.0
41     340 DO 350 L=1,N
42     350 A(ICOLUMN,L)=A(ICOLUMN,L)/PIVOT(I)
43     C
44     C      REDUCE NON-PIVOT ROWS
45     C
46     380 DO 550 L1=1,N
47     390 IF(L1-ICOLUMN) 400, 550, 400
48     400 T=A(L1,ICOLUMN)
49     420 A(L1,ICOLUMN)=0.0
50     430 DO 450 L=1,N
51     450 A(L1,L)=A(L1,L)-A(ICOLUMN,L)*T
52     550 CONTINUE
53     C
54     C      INTERCHANGE COLUMNS
55     C
56     600 DO 710 I=1,N

```

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

F402

MATINV

LIFTING-ROBOT LISTINGS

```
57 610 L=N+1-I
58 620 IF (INDEX(L,1)-INDEX(L,2)) 630, 710, 630
59 630 JROW=INDEX(L,1)
60 640 JCOLUM=INDEX(L,2)
61 650 DO 705 K=1,N
62 660 SWAP=A(K,JROW)
63 670 A(K,JROW)=A(K,JCOLUM)
64 700 A(K,JCOLUM)=SWAP
65 705 CONTINUE
66 710 CONTINUE
67 730 CONTINUE
68 RETURN
69 END
```

F402
F402
F402
F402
F402
F402
F402
F402
F402
F402

3 .L,0.

PRA63

LIFTING-ROBOT LISTINGS

1,5 TPF\$.PRA63
PUR 0018-05/17-11:51

OT*TPF\$.PRA63

```

1      SUBROUTINE PRA63(PR,ERROR)                                IX40
2      DIMENSION PR(15),PB(14),ZI(5),PK(6,5),RHOK(6,3),TK(6,5),VTK(6,3), IX40
3      1ZB(14),TMB(14),LMB(14)
4      REAL LMB
5      C ENTER WITH PR(1)=CURRENT ALTITUDE                        IX40
6      C CALCULATED TABLE AT RETURN                             IX40
7      C PR(1)=CURRENT ALTITUDE      Z                           IX40
8      C PR(2)=PRESSURE               PRES                       IX40
9      C PR(6)=DENSITY                DENS                      IX40
10     C PR(9)=SPEED OF SOUND         SPDSO                     IX40
11     1 Z=PR(1)                                                  IX40
12     IF( Z -700000.) 15,15,10                                  IX40
13     10 ERROR=1.                                                IX40
14     GO TO 500                                                  IX40
15     15 IF(Z,LE.0.)Z=0.
16     20 N=1                                                      IX40
17     IF( Z -83004.) 40,30,30                                    IX40
18     40 IF( Z -ZI(N))60,50,50                                    IX40
19     50 N=N+1                                                    IX40
20     GO TO 40                                                    IX40
21     60 Z2 =Z*Z                                                  IX40
22     Z3 =Z2*Z                                                    IX40
23     Z4 =Z2*Z2                                                  IX40
24     Z5 =Z2*Z3                                                  IX40
25     GO TO 100                                                  IX40
26     30 IF(Z-90000.) 300,70,70                                  IX40
27     70 IF(Z-ZB(N))85,400,80                                    IX40
28     80 N=N+1                                                    IX40
29     GO TO 70                                                    IX40
30     85 N=N-1                                                    IX40
31     GO TO 400                                                  IX40
32     C***** KINETIC TEMPERATURE FOR (0 TO 83004)              IX40
33     100 IF(Z,LE.12000.)GO TO 120
34     TEMPK=TK(1,N)+TK(2,N)*Z+TK(3,N)*Z2 +TK(4,N)*Z3 +TK(5,N)*Z4 +TK(IX40
35     16,N)*Z5                                                    IX40
36     IF(Z-28000.) 120,140,140                                    IX40
37     C***** PRESSURE FOR (0 TO 28000)                          IX40
38     120 PRES= 10.0000000*EXP(PK(1,N)+PK(2,N)*Z+PK(3,N)*Z2+PK(4,N)*Z3+PK(5 IX40
39     1,N)*Z4+PK(6,N)*Z5)                                         IX40
40     C***** DENSITY FOR (0 TO 28000)                           IX40
41     DENS=(1.16790729)*EXP(RHOK(1,N)+RHOK(2,N)*Z+RHOK(3,N)*Z2 +RHOK(4, IX40
42     1N)*Z3+RHOK(5,N)*Z4+RHOK(6,N)*Z5)                          IX40
43     IF(Z-12000.) 130,130,160                                    IX40
44     C***** VIRTUAL TEMPERATURE FOR (0 TO 12000)              IX40
45     130 TEMPV=(VTK(1,N)+VTK(2,N)*Z+VTK(3,N)*Z2 +VTK(4,N)*Z3 +VTK(5,N)*Z4 IX40
46     1 +VTK(6,N)*Z5)                                             IX40
47     GO TO 170                                                  IX40
48     C***** PRESSURE FOR (28000 TO 83004)                      IX40
49     140 PRES=.000980665* EXP(PK(1,N)+PK(2,N)*Z+PK(3,N)*Z2 + PK(4,N)*Z3+ IX40
50     1 PK(5,N)*Z4 +PK(6,N)*Z5)                                    IX40
51     C***** DENSITY FOR (28000 TO 90000)                      IX40
52     150 DENS=34.83676*(PRES/TEMPK)                             IX40
53     160 TEMPV=TEMPK                                             IX40
54     C***** SPEED OF SOUND (0 TO 90000)                       IX40
55     170 SPDSO=20.0468*SQRT(TEMPV)
56     C***CALCULATIONS COMPLETE , RETURN TO MAIN PROGRAM      IX40

```

PRA63

LIFTING-ROBOT LISTINGS

```

57      GO TO 500                                     IX40
58      300 TEMPK=180.65                               IX40
59      C***** PRESSURE FOR (83004 TO 90000)          IX40
60      PRES=PBASE*EXP((-1.373301523E12*( Z -83004.))/(180.65*(6344860.+Z IX40
61      1) *(6344860.+83004.)))                       IX40
62      GO TO 150                                       IX40
63      C***** MOLECULAR TEMPERATURE FOR (90000 TO 700000) IX40
64      400 TEMPM=TMB(N)+LMB(N)*(Z-ZB(N))
65      C***** KINETIC TEMPERATURE FOR (90000 TO 700000) IX40
66      C      THIS EQ IS EQUIVALENT TO PREVIOUS FORM
67      U=TMB(N)/(TMB(N)+LMB(N)*(Z-ZB(N)))
68      V=1.373301523E12/(LMB(N)*(Z+6344860.)*(6344860.+ZB(N)))
69      PRES=PB(N)*U**V
70      DENS= 34.83676*PRES/TEMPM
71      SPDSO=20.0468*SQRT(TEMPM)                     IX40
72      500 PR(1)=Z                                     IX40
73      PR(2)=PRES                                     IX40
74      PR(6)=DENS                                     IX40
75      PR(9)=SPDSO                                    IX40
76      RETURN                                         IX40
77      DATA PBASE/6.23101759E-5/                   IX40
78      DATA (ZI(I),I=1,5)/10832.1,17853.3,28000.,49000.,83004./ IX40
79      DATA ((PK(I,J),I=1,6),J=1,5)/1.6871582E-2,-1.1425176E-4,-1.3612327 IX40
80      XE-9,7.3624145E-14,-1.0800315E-17,3.3046432E-22,-7.9910777E-2,-8.10 IX40
81      X46438E-5,-5.5522383E-9,3.1116969E-13,-1.6687827E-17,3.8319351E-22, IX40
82      X9.8414277E-1,-2.6976917E-4,8.5227541E-9,-3.9620263E-13,1.0146471E- IX40
83      X17,-1.0264318E-22,                               IX40
84      X1.14118495E1,-4.11497477E-4,1.33664855E-8,-3.59518975E-13,       IX40
85      X5.10097254E-18,-2.89055894E-23,                   IX40
86      X9.99324461,-2.58298177E-4,3.76139346E-9,-4.20887236E-14,       IX40
87      X1.60182148E-19,-1.92508927E-25/                   IX40
88      DATA ((RHOK(I,J),I=1,6),J=1,3)/1.3302117E-2,-8.8502064E-5,-4.21430 IX40
89      X56E-9,5.9517557E-13,-3.9744789E-17,7.8771273E-22,1.2667122E-1,   IX40
90      X-1.3373147E-4,2.0667371E-9,2.3396109E-13,-3.2562503E-17,7.9035209E IX40
91      X-22,9.2751266E-1,-1.4349679E-4,-2.8271736E-9,4.7480092E-14,     IX40
92      X1.8863246E-18,-4.2702411E-23/                   IX40
93      DATA ((TK(I,J),I=1,6),J=1,5)/2.9667877E2,-6.7731001E-3,8.4619805E- IX40
94      X7,-1.7004049E-10,1.1451454E-14,-2.4898788E-19,           IX40
95      X2.6892151E2,4.3075352E-3,-8.9159672E-7,-2.8929791E-11,5.0724856E-1 IX40
96      X5,-1.1490372E-19,                               IX40
97      X3.7064557E2,-3.2858965E-2,2.0645636E-6,-4.3283944E-11,-5.7507242E- IX40
98      X17,8.2924583E-21,                               IX40
99      X2.044798E1,2.07698384E-2,-8.63038789E-7,1.66392417E-11,       IX40
100     X-9.30076185E-17,-4.09005108E-22,                   IX40
101     X-4.98865953E2,3.92137281E-2,-4.95180601E-7,-3.26219854E-12,       IX40
102     X 9.66650364E-17,-4.78844279E-22/                   IX40
103     DATA ((VTK(I,J),I=1,6),J=1,3)/2.9937265E2,-7.717628E-3,9.4867202E- IX40
104     X7,-1.7136592E-10,1.1074297E-14,-2.3294094E-19,           IX40
105     X2.6892151E2,4.3075352E-3,-8.9159672E-7,-2.8929791E-11,5.0724856E-1 IX40
106     X5,-1.1490372E-19,                               IX40
107     X3.7064557E2,-3.2858965E-2,2.0645636E-6,-4.3283944E-11,-5.7507242E- IX40
108     X17,8.2924583E-21/                                   IX40
109     DATA (ZB(I),I=1,14)/                               IX40
110     X9.E4,1.E5,1.1E5,1.2E5,1.5E5,1.6E5,1.7E5,1.9E5,2.3E5,3.E5,4.E5,5.E5 IX40
111     X,6.E5,7.E5/                                         IX40
112     DATA (TMB(I),I=1,14)/ 180.65,210.65,260.65,360.65,960.65, IX40
113     X1110.65,1210.65,1350.65,1550.65,1830.65,2160.65,2420.65,2590.65, IX40

```

PRA63

LIFTING-ROBOT LISTINGS

```
114      X2700.65/                                IX40
115      DATA (LMB(I),I=1,14)/      3.E-3,5.E-3,10.E-3,20.E-3,15.E-3,10.E-3, IX40
116      X7.E-3,5.E-3,4.E-3,3.3E-3,2.6E-3,1.7E-3,1.1E-3,1.1E-3/      IX40
117      DATA (PB(I),I=1,14)/.172244361E-4,.315971712E-5,.774389807E-6,      IX40
118      X.265977111E-6,.535849383E-7,.391284945E-7,.295911117E-7,      IX40
119      X.178715656E-7,.739258171E-8,.200573116E-8,.430456606E-9,      IX40
120      X.117315480E-9,.370198961E-10,.128115330E-10/      IX40
121      END
```

5 .L.O.

PRB63

LIFTING-ROBOT LISTINGS

1,5 TPF\$.PRB63
2UR 0018-05/17-11:51

DT*TPF\$.PRB63

```

1      SUBROUTINE PRB63
2      COMMON/ARDC/PR(15)
3      DIMENSION PB(14),ZI(5),PK(6,5),RHOK(6,3),TK(6,5),VTK(6,3), IX40
4      1ZB(14),TMB(14),LMB(14)
5      REAL LMB
6      C ENTER WITH PR(1)=CURRENT ALTITUDE IX40
7      C CALCULATED TABLE AT RETURN IX40
8      C PR(1)=CURRENT ALTITUDE Z IX40
9      C PR(2)=PRESSURE PRES IX40
10     C PR(6)=DENSITY DENS IX40
11     C PR(9)=SPEED OF SOUND SPDSO IX40
12     C DD=DRODH/RHO
13     C DSO=DSSDH/SS
14     C UPR=DPDH
15     1 Z=PR(1) IX40
16     IF( Z -700000.) 15,15,10 IX40
17     10 ERROR=1. IX40
18     GO TO 500 IX40
19     15 IF(Z.LE.0.)Z=0.
20     20 N=1 IX40
21     IF( Z -83004.) 40,30,30 IX40
22     40 IF( Z -ZI(N))60,50,50 IX40
23     50 N=N+1 IX40
24     GO TO 40 IX40
25     60 Z2 =Z*Z IX40
26     Z3 =Z2*Z IX40
27     Z4 =Z2*Z2 IX40
28     Z5 =Z2*Z3 IX40
29     DZ2=2.*Z
30     DZ3=3.*Z2
31     DZ4=4.*Z3
32     DZ5=5.*Z4
33     GO TO 100 IX40
34     30 IF(Z-90000.) 300,70,70 IX40
35     70 IF(Z-ZB(N))85,400,80 IX40
36     80 N=N+1 IX40
37     GO TO 70 IX40
38     85 N=N-1 IX40
39     GO TO 400 IX40
40     C***** KINETIC TEMPERATURE FOR (0 TO 83004) IX40
41     100 IF(Z.LE.12000.)GO TO 120
42     TEMPK=TK(1,N)+TK(2,N)*Z+TK(3,N)*Z2 +TK(4,N)*Z3 +TK(5,N)*Z4 +TK(IX40
43     16,N)*Z5 IX40
44     DTK=TK(2,N)+TK(3,N)*DZ2+TK(4,N)*DZ3+TK(5,N)*DZ4+TK(6,N)*DZ5
45     IF(Z-28000.) 120,140,140 IX40
46     C***** PRESSURE FOR (0 TO 28000) IX40
47     120 PRES= 10.0000000*EXP(PK(1,N)+PK(2,N)*Z+PK(3,N)*Z2+PK(4,N)*Z3+PK(5 IX40
48     1,N)*Z4+PK(6,N)*Z5) IX40
49     DPR=PRES*(PK(2,N)+PK(3,N)*DZ2+PK(4,N)*DZ3+PK(5,N)*DZ4+PK(6,N)*DZ5)
50     C***** DENSITY FOR (0 TO 28000) IX40
51     DENS=(1.16790729)*EXP(RHOK(1,N)+RHOK(2,N)*Z+RHOK(3,N)*Z2 +RHOK(4, IX40
52     1N)*Z3+RHOK(5,N)*Z4+RHOK(6,N)*Z5) IX40
53     DD= (RHOK(2,N)+RHOK(3,N)*DZ2+RHOK(4,N)*DZ3+RHOK(5,N)*DZ4
54     1 +RHOK(6,N)*DZ5)
55     IF(Z-12000.) 130,130,160 IX40
56     C***** VIRTUAL TEMPERATURE FOR (0 TO 12000) IX40

```

```

57      130 TEMPV=(VTK(1,N)+VTK(2,N)*Z+VTK(3,N)*Z2 +VTK(4,N)*Z3 +VTK(5,N)*Z4IX40
58      1 +VTK(6,N)*Z5)IX40
59      DTV=VTK(2,N)+VTK(3,N)*DZ2 +VTK(4,N)*DZ3+VTK(5,N)*DZ4+VTK(6,N)*DZ5
60      GO TO 170IX40
61      C***** PRESSURE FOR (28000 TO 83004)IX40
62      140 PRES=.000980665* EXP(PK(1,N)+PK(2,N)*Z+PK(3,N)*Z2 + PK(4,N)*Z3+IX40
63      1 PK(5,N)*Z4 +PK(6,N)*Z5)IX40
64      DPR=PRES*(PK(2,N)+PK(3,N)*DZ2+PK(4,N)*DZ3+PK(5,N)*DZ4+PK(6,N)*DZ5)
65      C***** DENSITY FOR (28000 TO 90000)IX40
66      150 DENS=34.83676*(PRES/TEMPK)IX40
67      DD= (DPR/PRES-DTK/TEMPK)
68      160 TEMPV=TEMPKIX40
69      DTV=DTK
70      C***** SPEED OF SOUND (0 TO 90000)IX40
71      170 SPDSO=20.0468*SQRT(TEMPV)
72      DSO=.5*DTV/TEMPV
73      C***CALCULATIONS COMPLETE , RETURN TO MAIN PROGRAMIX40
74      GO TO 500IX40
75      300 TEMPK=180.65IX40
76      DTK=0.
77      C***** PRESSURE FOR (83004 TO 90000)IX40
78      PRES=PBASE*EXP((-1.373301523E12*( Z -83004.))/(180.65*(6344860.+ZIX40
79      1) *(6344860.+83004.)))IX40
80      DPR=180.65*6427864.*(6344860.+Z)
81      DPR=PRES*((Z-83004.)/(6344860.+Z)-1.)*1.373301523E12/DPR
82      GO TO 150IX40
83      C***** MOLECULAR TEMPERATURE FOR (90000 TO 700000)IX40
84      400 TEMPM=TMB(N)+LMB(N)*(Z-ZB(N))
85      DTM=LMB(N)
86      C***** KINETIC TEMPERATURE FOR (90000 TO 700000)IX40
87      U=TMB(N)/(TMB(N)+LMB(N)*(Z-ZB(N)))
88      V=1.373301523E12/(LMB(N)*(Z+6344860.)*(6344860.+ZB(N)))
89      PRES=PB(N)*U**V
90      DPR=-PRES*(V*LMB(N)*U/TMB(N)+ALOG(U)*V/(6344860.+Z))
91      DENS= 34.83676*PRES/TEMPMIX40
92      DD= (DPR/PRES-DTM/TEMPM)
93      SPDSO=20.0468*SQRT(TEMPM)IX40
94      DSO=.5*DTM/TEMPM
95      500 PR(1)=ZIX40
96      PR(2)=PRES*10000.
97      PR(3)=DPR*10000.
98      PR(4)=DD
99      PR(5)=DSO
100     PR(6)=DENSIX40
101     PR(9)=SPDSOIX40
102     RETURNIX40
103     DATA PBASE/6.23101759E-5/IX40
104     DATA (ZI(1),I=1,5)/10832.1,17853.3,28000.,49000.,83004./IX40
105     DATA ( PK(1) ,I=1,30 )/1.6871582E-2,-1.1425176E-4,-1.3612527IX40
106     XE-9,7.3624145E-14,-1.0800315E-17,3.3046432E-22,-7.9910777E-2,-8.10IX40
107     X46438E-5,-5.5522383E-9,3.1116969E-13,-1.6687827E-17,3.8319351E-22,IX40
108     X9.8414277E-1,-2.6976917E-4,8.5227541E-9,-3.9620263E-13,1.0146471E-IX40
109     X17,-1.0264318E-22,IX40
110     X1.14118495E1,-4.11497477E-4,1.33664855E-8,-3.59518975E-13,IX40
111     X5.10097254E-18,-2.89055894E-23,IX40
112     X9.99324461,-2.58298177E-4,3.76139346E-9,-4.20887236E-14,IX40
113     X1.60182148E-19,-1.92508927E-25/IX40

```

```

114 DATA ( RHOK(I) , I=1,18 )/1.3302117E-2,-8.8502064E-5,-4.21430IX40
115 X56E-9,5.9517557E-13,-3.9744789E-17,7.8771273E-22,1.2667122E-1, IX40
116 X-1.3373147E-4,2.0667371E-9,2.3396109E-13,-3.2562503E-17,7.9035209EIX40
117 X-22,9.2751266E-1,-1.4349679E-4,-2.8271736E-9,4.7480092E-14, IX40
118 X1.8863246E-18,-4.2702411E-23/ IX40
119 DATA ( TK(I) , I=1,30 )/2.9667877E2,-6.7731001E-3,8.4619805E-IX40
120 X7,-1.7004049E-10,1.1451454E-14,-2.4898788E-19, IX40
121 X2.6892151E2,4.3075352E-3,-8.9159672E-7,-2.8929791E-11,5.0724856E-1IX40
122 X5,-1.1490372E-19, IX40
123 X3.7064557E2,-3.2858965E-2,2.0645636E-6,-4.3283944E-11,-5.7507242E-IX40
124 X17,8.2924583E-21, IX40
125 X2.044798E1,2.07698384E-2,-8.63038789E-7,1.66392417E-11, IX40
126 X-9.30076185E-17,-4.09005108E-22, IX40
127 X-4.98865953E2,3.92137281E-2,-4.95180601E-7,-3.26219854E-12, IX40
128 X 9.66650364E-17,-4.78844279E-22/ IX40
129 DATA ( VTK(I) , I=1,18 )/2.9937265E2,-7.717626E-3,9.4867202E-IX40
130 X7,-1.7136592E-10,1.1074297E-14,-2.3294094E-19, IX40
131 X2.6892151E2,4.3075352E-3,-8.9159672E-7,-2.8929791E-11,5.0724856E-1IX40
132 X5,-1.1490372E-19, IX40
133 X3.7064557E2,-3.2858965E-2,2.0645636E-6,-4.3283944E-11,-5.7507242E-IX40
134 X17,8.2924583E-21/ IX40
135 DATA ( ZB(I), I=1,14)/ IX40
136 X9.E4,1.E5,1.1E5,1.2E5,1.5E5,1.6E5,1.7E5,1.9E5,2.3E5,3.E5,4.E5,5.E5IX40
137 X,6.E5,7.E5/ IX40
138 DATA ( TMB(I), I=1,14)/ 180.65,210.65,260.65,360.65,960.65, IX40
139 X110.65,1210.65,1350.65,1550.65,1830.65,2160.65,2420.65,2590.65, IX40
140 X2700.65/ IX40
141 DATA ( LMB(I), I=1,14)/ 3.E-3,5.E-3,10.E-3,20.E-3,15.E-3,10.E-3,IX40
142 X7.E-3,5.E-3,4.E-3,3.3E-3,2.6E-3,1.7E-3,1.1E-3,1.1E-3/ IX40
143 DATA ( PB(I), I=1,14)/.172244361E-4,.315971712E-5,.774389807E-6, IX40
144 X.265977111E-6,.535849383E-7,.391284945E-7,.295911117E-7, IX40
145 X.178715656E-7,.739258171E-8,.200573116E-8,.430456606E-9, IX40
146 X.117315480E-9,.370198961E-10,.128115330E-10/ IX40
147 END

```

3 .L.O.

SPLINE

LIFTING-ROBOT LISTINGS

TPF\$.SPLINE
UR 0018-05/17-11:51

SPLINE

LIFTING-ROBOT LISTINGS

OT*TPF\$,SPLINE

```

1      SUBROUTINE SPLINE(L,M,N,X,XT,Y1,Y1T,Y2,Y2T,Y3,Y3T,Y1D,Y2D,Y3D)
2      DIMENSION XT(1),Y1T(1),Y2T(1),Y3T(1)
3      IF(M.LT.1.OR.M.GT.N)M=1
4      IF(X-XT(M).GT.0.)GO TO 2
5      C      GET HERE IF X.LT.XT(M)
6          1 IF(M.EQ.1)GO TO 4
7          M=M-1
8          IF(X-XT(M))1,4,4
9      C      GET HERE IF X.GT.XT(M)
10         2 IF(M.EQ.N)GO TO 5
11         M=M+1
12         IF(X-XT(M).GT.0.)GO TO 2
13      C      GOT HERE WORKING UP
14         5 M=M-1
15      C      GOT HERE WORKING DOWN
16         4 MP1=M+1
17         MM1=M-1
18         MP2=MP1+1
19         DX=X-XT(M)
20         DLX=XT(MP1)-XT(M)
21         DLXI=1./DLX
22         R=DX*DLXI
23         R2=R*R
24         S=(Y1T(MP1)-Y1T(M))*DLXI
25         T=(Y2T(MP1)-Y2T(M))*DLXI
26         U=(Y3T(MP1)-Y3T(M))*DLXI
27         IF(M.LT.N-1)GO TO 31
28      C      AT LAST POINT
29         SP1=S
30         TP1=T
31         UP1=U
32         GO TO 32
33      31 CONTINUE
34         DLXPI=1./(XT(MP2)-XT(MP1))
35         SP1=(Y1T(MP2)-Y1T(MP1))*DLXPI
36         TP1=(Y2T(MP2)-Y2T(MP1))*DLXPI
37         UP1=(Y3T(MP2)-Y3T(MP1))*DLXPI
38         IF(M.GT.1)GO TO 32
39      C      AT FIRST POINT
40         SM1=S
41         TM1=T
42         UM1=U
43         GO TO 33
44      32 CONTINUE
45         DLXMI=1./(XT(M)-XT(MM1))
46         SM1=(Y1T(M)-Y1T(MM1))*DLXMI
47         TM1=(Y2T(M)-Y2T(MM1))*DLXMI
48         UM1=(Y3T(M)-Y3T(MM1))*DLXMI
49      33 CONTINUE
50         AM2G=.5*(SP1+SM1)-S
51         BM2G=.5*(TP1+TM1)-T
52         CM2G=.5*(UP1+UM1)-U
53         AM3G=.5*(SP1-S)+SM1-S
54         BM3G=.5*(TP1-T)+TM1-T
55         CM3G=.5*(UP1-U)+UM1-U
56         A=.5*(S+SM1)

```

SPLINE

LIFTING-ROBOT LISTINGS

```

57      B=.5*(T+TM1)
58      C=.5*(U+UM1)
59      Y1=Y1T(M)+(AM2G*DX-AM3G*DLX)*R2+A*DX
60      Y2=Y2T(M)+(BM2G*DX-BM3G*DLX)*R2+B*DX
61      Y3=Y3T(M)+(CM2G*DX-CM3G*DLX)*R2+C*DX
62      IF (L.EQ.1)RETURN
63      Y1D=3.*AM2G*R2-2.*AM3G*R+A
64      Y2D=3.*BM2G*R2-2.*BM3G*R+B
65      Y3D=3.*CM2G*R2-2.*CM3G*R+C
66      RETURN
67      END

```

.L,0.

SYSDER

LIFTING-ROBOT LISTINGS

1,5 TPF\$.SYSDER
PUR 0018-05/17-11:51

SYSDER

LIFTING-ROBOT LISTINGS

JT*TPF\$.SYSDER

```

1  SUBROUTINE SYSDER
2  COMMON/GRAV/GTO,G11,G22,G23,G33
3  COMMON/DFEQS/N,XN,X,XH,II,YN(6),Y(6),Y1(6),DY(6)
4  COMMON/AGEN2/AA,SA,CA,ALF,SIG,SSIG,CSIG,CHIP,SCHIP,CCHIP,CHIY,
5  1 SCHIY,CCHIY,THZ,STH,CTH,THGT,THETL,STHL,CTHL,DPHIZ,GR,GTH,
6  2 RTHE,RTHZ,R,R2,VR,THR,XM,XMDOT,SW,SU,SV,WZ,UZ,VZ,XZ,YZ,ZZ,WRZ,
7  3 VIV(8),DVAR(7),DELXDW(15,3),DELXDR(7),DELXD(15,7),EXIT,
8  4 DRAG1,CHIDOT,WZERO,GZERO,ALT,RHO,PAM,SS,XMACH,Q,QDOT,CAX,CNP,CN,
9  5 FAA,FAN,FX,FY,FZ,A(3,3),BL(3,3),BO(3,3),D(3,3),DQDX(6),CASE,
10 6 CT2,UMF,A12W,A22W,A32W,XJEXT,GNU(10),KINDB,KRDERB,BEU,BYL,
11 7 BHMAX(15),BHMN,BSTEP(15)
12
13 C      THIS SUBROUTINE SPECIFIES THE SYSTEM DIFFERENTIAL EQUATIONS IN
14 C
15 C      ORBIT
16 C
17     II = 1
18 1  CALL FORKM
19     IF(II.NE.6) GO TO 2
20     RETURN
21 2  R2= Y(4)*Y(4) + Y(5)*Y(5) + Y(6)*Y(6)
22     R = SQRT(R2)
23     CTH = (A(1,2)*Y(4) + A(2,2)*Y(5) + A(3,2)*Y(6))/R
24     CT2 = CTH*CTH
25     CALL AGE0(1)
26     DO 3 I = 1,3
27     J = I + 3
28     DY(I) = Y(J)*G11 - A(I,2)*GTO
29 3  DY(J) = Y(I)
30     GO TO 1
31     END

```

.L,0.

TARGET

LIFTING-ROBOT LISTINGS

T, S TPF, TARGET
UR 0018-05/17-11:51

TARGET

LIFTING-ROBOT LISTINGS

OT*TPF\$.TARGET

```

1  SUBROUTINE TARGET(T)
2  COMMON/CONST/RAD,PI,RE,FLAT,CJ,H,DJ,CMUE,OMEGA,ALT1,ALT2,PSL
3  COMMON/TARPO/TARPO(6),TAR(6),TAULN(2)
4  DOUBLEPRECISION      YT(6),XMUE,TAUC
5  XMUE=CMUE
6  TAUC=T
7  WRITE(6,2)T
8  WRITE(6,2)(TARPO(I),I=1,6)
9  CALL ATCST(TARPO,YT,XMUE,TAUC)
10 DO 1 I=1,6
11 1 TAR(I)=YT(I)
12  WRITE(6,2)(TAR (I),I=1,6)
13 2 FORMAT(2X,6E15.8)
14 RETURN
15 END

```

.L,0.

DESOLV

LIFTING-ROBOT LISTINGS

1,S TPF\$.DESOLV
PUR 0018-05/17-11:51

LIFTING-ROBOT LISTINGS

DT*TPF\$.DESOLV

```

1  SUBROUTINE DESOLV (MXEQ,NACT,DR1,DR2,ENDS,NP,KERR,HNOM,HBK,BLK,
2  C
3  C      *      *      *      *      *      *      *      *
4  C  DESOLV INTEGRATION PACKAGE PROGRAMMED BY JOAN SHEEHAN,
5  C  NOVEMBER 1966, UNDER SUPPORT WORK ORDER WITH MSFC, HUNTSVILLE,
6  C  ALABAMA.
7  C      *      *      *      *      *      *      *      *
8  C
9  X      ITYP, LL, EUP,ELO,DTMX,DTMN,YLO,
10 X      NOTRG,
11 X      NTR1, RT1, VAR1, ZV1,  NTR2, RT2, VAR2, ZV2,
12 X      NTR3, RT3, VAR3, ZV3,  NTR4, RT4, VAR4, ZV4,
13 X      NTR5, RT5, VAR5, ZV5,  NTR6, RT6, VAR6, ZV6,
14 X      NTR7, RT7, VAR7, ZV7,  NTR8, RT8, VAR8, ZV8,
15 X      NTR9, RT9, VAR9, ZV9,  NTR10,RT10,VAR10,ZV10,
16 X      NTR11,RT11,VAR11,ZV11 )
17 C      WHERE MXEQ = THE TOTAL NUMBER OF POSSIBLE FIRST ORDER DIFFER-
18 C                  ENTIAL EQUATIONS TO BE SOLVED
19 C      NACT = ACTUAL NUMBER OF DIFFERENTIAL EQUATIONS TO SOLVED
20 C      DR1 = ENTRY POINT TO ROUTINE THAT CARRIES OUT THE
21 C           PORTION OF THE DERIVATIVE EVALUATION INVOLVING
22 C           THE INDEPENDENT VARIABLE ONLY
23 C           DR1 IS OPTIONAL. PUT A ZERO IF NOT DESIRED.
24 C      DR2 = ENTRY POINT TO THE ROUTINE THAT CARRIES OUT THE
25 C           REMAINING CALCULATIONS NECESSARY TO COMPUTE ALL
26 C           NACT DERIVATIVES AND STORES THEM IN HBK
27 C      ENDS = ENTRY POINT TO ROUTINE ENTERED AT THE END OF A
28 C           FULL INTEGRATION STEP - OPTIONAL.
29 C           PUT A ZERO IF NO END-OF-STEP ROUTINE IS DESIRED.
30 C      NP = ONE OF THE FOLLOWING CODE NUMBERS DEPENDING ON
31 C          THE OPTIONS DESIRED
32 C          //////////////////////////////////
33 C          VALUES OF NP / DR1 / ENDS /
34 C          //////////////////////////////////
35 C                  1 / NO / YES /
36 C                  2 / YES / YES /
37 C                  5 / NO / NO /
38 C                  6 / YES / NO /
39 C          //////////////////////////////////
40 C
41 C      KERR = ERROR INDICATOR. NORMAL RETURN WHEN KERR=0
42 C                  ERROR RETURN WHEN KERR=1
43 C      HNOM = NOMINAL STEP SIZE
44 C      HBK = THE FIRST WORD OF A BLOCK OF STORAGE WITH HBK(1)
45 C           STARTING IN AN EVEN LOCATION AND -
46 C           HBK(1) = CURRENT STEP SIZE
47 C           HBK(2) = NOT USED
48 C           HBK(3) = INDEPENDENT VARIABLE IN SINGLE
49 C                   PRECISION (MUST BE PRESET)
50 C           HBK(4) = NOT USED
51 C           HBK(5) TO HBK(MXEQ+4) = DEPENDENT VARIABLES
52 C           HBK(MXEQ+5) TO HBK(2*MXEQ+4) = DERIVATIVES
53 C      BLK = THE FIRST WORD OF A BLOCK OF STORAGE THAT MUST BE
54 C           EQUIVALENT TO HBK(2*MXEQ+5) AND MUST HAVE TWO
55 C           DIMENSIONS (MXEQ,LL) WHERE LL IS DEFINED BELOW
56 C      ITYP = INDICATOR FOR THE TYPE OF INTEGRATION TO BE USED

```

DESOLV

LIFTING-ROBOT LISTINGS

```

57      C      ITYP=1, VARIABLE STEP ADAMS-MOULTON          DSLV
58      C      ITYP=2, FIXED STEP RUNGE-KUTTA              DSLV
59      C      ITYP=3, FIXED STEP ADAMS PREDICTOR ONLY      DSLV
60      C      (AN EXTERNAL CHANGE OF HNOM CAUSES A RESTART ) DSLV
61      C      LL   = THE VARIABLE DIMENSION USED IN BLK(MXEQ,LL) DSLV
62      C      LL=5 WHEN ITYP = 2                          DSLV
63      C      LL = 3*(M+2) +4 WHEN ITYP = 1 OR 3          DSLV
64      C      M IS THE ORDER OF THE DIFFERENCES USED      DSLV
65      C      EUP  = UPPER LIMIT FOR STEP SIZE CONTROL    DSLV
66      C      ELO  = LOWER LIMIT FOR STEP SIZE CONTROL    DSLV
67      C      DTMX = MAXIMUM STEP SIZE ALLOWED            DSLV
68      C      DTMN = MINIMUM STEP SIZE ALLOWED            DSLV
69      C      YLO  = THE SMALLEST VALUE OF A DEPENDENT VARIABLE THAT DSLV
70      C      WILL AFFECT THE AUTOMATIC STEP SIZE LOGIC   DSLV
71      C      NOTRG= THE NUMBER OF TRIGGERS TO FOLLOW      DSLV
72      C      THE MAXIMUM NO OF TRIGGERS IS SET AT 11     DSLV
73      C      NTR- = PLUS OR MINUS AN INTEGER 1 OR 2      DSLV
74      C      PLUS SIGN MEANS TRIGGER IS ON               DSLV
75      C      MINUS SIGN MEANS TRIGGER IS OFF             DSLV
76      C      INTEGER 1 INDICATES NO DISCONTINUITY AFTER  DSLV
77      C      EXECUTING THE TRIGGER ROUTINE               DSLV
78      C      INTEGER 2 INDICATES A DISCONTINUITY AND FORCES DSLV
79      C      A RESTART                                   DSLV
80      C      RT-  = ENTRY POINT OF THE TRIGGER INTERRUPT ROUTINE DSLV
81      C      VAR- = NAME OF THE VARIABLE BEING TESTED    DSLV
82      C      ZV-  = THE VALUE OF VAR AT WHICH THE TRIGGER INTERRUPT DSLV
83      C      ROUTINE IS TO BE EXECUTED                   DSLV
84      C
85      DIMENSION KERR(1)                                  DSLV
86      COMMON/MOD1/ACO(10),BCO(10)                       DSLV
87      COMMON /DES1/                                       DSLV
88      * KR(1), NMX, NEQ, NOLD, M, MP1, MP2, NN, KIND, KON, DSLV
89      * IFL, NTRG, NIVT, NDVT, NTR,NSTART,NBDIF, NNT, NTT,IDL(11),DSL
90      * IDL2, JJSS,J1000,J2000,J4000,J7000,IRST,EOST, HD, SMA, DSLV
91      * ERROR,REDOT,STEPNG,                               DSLV
92      * EU(1), EL, HMX, HMN, YL, A, ENP1, D, B, SKP,      DSLV
93      * SK, HC, DELU, DUSC, HOLD, VAR, ZV, HM, HP, R,     DSLV
94      * Q, DELT, HMU, FMU,CK(10),QQ(10),XLJ(11),RU(11),WU(11) DSLV
95      COMMON /DES2/ T, TSV, TGO, TMIN, TRIP, TL, TR      DSLV
96      DOUBLE PRECISION T, TSV, TGO, TMIN, TRIP, TL, TR   DSLV
97      C
98      LOGICAL EOST, SMA, HD, IRST, ERROR                 DSLV
99      LOGICAL STEPNG                                     DSLV
100     C
101     DIMENSION HBK(1), BLK(MX,0,LL)                      DSLV
102     C
103     DIMENSION HNOM(1), NACT(1), NP(1)                   DSLV
104     DIMENSION VAR1(1),VAR2(1),VAR3(1),VAR4(1),VAR5(1),VAR6(1), DSLV
105     X VAR7(1),VAR8(1),VAR9(1),VAR10(1),VAR11(1)         DSLV
106     DIMENSION ZV1(1), ZV2(1), ZV3(1), ZV4(1), ZV5(1),ZV6(1), DSLV
107     X ZV7(1), ZV8(1), ZV9(1), ZV10(1), ZV11(1)         DSLV
108     DIMENSION NTR1(1),NTR2(1),NTR3(1),NTR4(1),NTR5(1),NTR6(1), DSLV
109     X NTR7(1),NTR8(1),NTR9(1),NTR10(1),NTR11(1)        DSLV
110     C
111     DATA BIG/1.0E30/                                    DSLV
112     C
113     GO TO 400

```

DESOLV

LIFTING-ROBOT LISTINGS

114	402	CONTINUE	DSL
115		MXEQ=MXEQ	DSL
116		NACT=NACT	DSL
117		NP=NP	DSL
118		KERR=KERR	DSL
119		HNOM=HNOM	DSL
120		IDDI=2*MXEQ+4	DSL
121	C		DSL
122		DO 152 ID=1, IDDI	DSL
123	152	HBK(ID)=HBK(ID)	DSL
124		BLK(1,1)=BLK(1,1)	DSL
125		ITYP=ITYP	DSL
126		LL=LL	DSL
127		EUP=EUP	DSL
128		ELO=ELO	DSL
129		DTMX=DTMX	DSL
130		DTMN=DTMN	DSL
131		YLO=YLO	DSL
132		NOTRG=NOTRG	DSL
133		NTR1(1)=NTR1(1)	DSL
134		NTR2(1)=NTR2(1)	DSL
135		NTR3(1)=NTR3(1)	DSL
136		NTR4(1)=NTR4(1)	DSL
137		NTR5(1)=NTR5(1)	DSL
138		NTR6(1)=NTR6(1)	DSL
139		NTR7(1)=NTR7(1)	DSL
140		NTR8(1)=NTR8(1)	DSL
141		NTR9(1)=NTR9(1)	DSL
142		NTR10(1)=NTR10(1)	DSL
143		NTR11(1)=NTR11(1)	DSL
144		VAR1(1)=VAR1(1)	DSL
145		VAR2(1)=VAR2(1)	DSL
146		VAR3(1)=VAR3(1)	DSL
147		VAR4(1)=VAR4(1)	DSL
148		VAR5(1)=VAR5(1)	DSL
149		VAR6(1)=VAR6(1)	DSL
150		VAR7(1)=VAR7(1)	DSL
151		VAR8(1)=VAR8(1)	DSL
152		VAR9(1)=VAR9(1)	DSL
153		VAR10(1)=VAR10(1)	DSL
154		VAR11(1)=VAR11(1)	DSL
155		ZV1(1)=ZV1(1)	DSL
156		ZV2(1)=ZV2(1)	DSL
157		ZV3(1)=ZV3(1)	DSL
158		ZV4(1)=ZV4(1)	DSL
159		ZV5(1)=ZV5(1)	DSL
160		ZV6(1)=ZV6(1)	DSL
161		ZV7(1)=ZV7(1)	DSL
162		ZV8(1)=ZV8(1)	DSL
163		ZV9(1)=ZV9(1)	DSL
164		ZV10(1)=ZV10(1)	DSL
165		ZV11(1)=ZV11(1)	DSL
166	400	CONTINUE	DSL
167	C		DSL
168		ACO(1)=1.	DSL
169		ACO(2)=.5	DSL
170		ACO(3)=5./12.	DSL

DESOLV

LIFTING-ROBOT LISTINGS

171	ACO(4)=3./8.	DSL
172	ACO(5)=251./720.	SLV
173	ACO(6)=95./288.	DSL
174	ACO(7)=19087./60480.	SLV
175	ACO(8)=36799./120960.	DSL
176	ACO(9)=1070017./3628800.	SLV
177	ACO(10)=.287075448	DSL
178	BCO(1)=1.	SLV
179	BCO(2)=-.5	DSL
180	BCO(3)=-1./12.	SLV
181	BCO(4)=-1./24.	DSL
182	BCO(5)=-19./720.	SLV
183	BCO(6)=-3./160.	DSL
184	BCO(7)=-863./60480.	SLV
185	BCO(8)=-275./24192.	DSL
186	BCO(9)=-33953./3628800.	SLV
187	BCO(10)=-.78925542E-2	DSL
188	NMX = MXEQ	SLV
189	KIND = ITYP	DSL
190	NTRG = NOTRG	SLV
191	EU = EUP	DSL
192	EL = ELO	SLV
193	HMX = DTMX	DSL
194	HMN = DTMN	SLV
195	YL = YLO	DSL
196	C	SLV
197	50 MXNTRG = 11	DSL
198	IF (NTRG .LE. MXNTRG) GO TO 52	SLV
199	51 FORMAT (// 52H MAXIMUM NUMBER OF TRIGGERS THIS DECK WILL HANDLE IS	DSL
200	X, I4/ 5X14HRECOMPILE F41R)	SLV
201	WRITE (6,51) MXNTRG	DSL
202	KERR = 1	SLV
203	RETURN	DSL
204	52 CONTINUE	SLV
205	C INITIALIZATION	DSL
206	C CALCULATE DELTA U SCALE FACTOR = 0.596E-7	SLV
207	TEMP=2**24	DSL
208	DUSC = 1.0/TEMP	SLV
209	STEPNG = .FALSE.	DSL
210	IF (KERR .EQ. 100) STEPNG = .TRUE.	SLV
211	KERR = 0	DSL
212	KR = 0	SLV
213	DO 55 J=1,MXNTRG	DSL
214	55 IDL(J) = 0	SLV
215	HDK(1) = HNOM	DSL
216	HOLD = HNOM	SLV
217	C	DSL
218	C	SLV
219	C POINT FOR RESTART	DSL
220	100 CONTINUE	SLV
221	102 FORMAT (///5H NP =I3, 21H - UNACCEPTABLE VALUE)	DSL
222	103 FORMAT (///7H ITYP =, I3, 11HNOT ALLOWED)	SLV
223	IF (NP .EQ. 1 .OR. NP .EQ. 2 .OR. NP .EQ. 5 .OR. NP .EQ. 6)	DSL
224	X GO TO 105	SLV
225	WRITE (6,102) NP	DSL
226	KERR = 1	SLV
227	RETURN	DSL

DESOLV

LIFTING-ROBOT LISTINGS

```

228 105 IF (KIND .EQ. 1 .OR. KIND .EQ. 2) GO TO 106          DSLV
229 IF (KIND .EQ. 3) GO TO 106                               DSLV
230 WRITE (6,103) KIND                                       DSLV
231 KERR = 1                                                  DSLV
232 RETURN                                                    DSLV
233 C                                                         DSLV
234 106 CONTINUE                                              DSLV
235 NOLD = NACT                                               DSLV
236 NEQ = NACT                                               DSLV
237 EOST = .FALSE.                                           DSLV
238 NN = NP-4                                                 DSLV
239 IF (NP .GT. 2) GO TO 108                                  DSLV
240 EOST = .TRUE.                                             DSLV
241 NN = NP                                                    DSLV
242 108 IRST = .FALSE.                                         DSLV
243 C                                                         DSLV
244 C INITIALIZE FOR RUNGE-KUTTA OR ADAMS-MOULTON START      DSLV
245 C KIND = 1 FOR ADAMS-MOULTON                             DSLV
246 C KIND = 2 FOR RUNGE-KUTTA                               DSLV
247 C KIND = 3 FOR FIXED STEP ADAMS OPEN FORMULA           DSLV
248 C                                                         DSLV
249 GO TO (110, 112, 110), KIND                               DSLV
250 C VARIABLE STEP ADAMS-MOULTON                             DSLV
251 110 TRIP = HBK(3)                                          DSLV
252 NSTART = 0                                                DSLV
253 M=(LL-10)/3                                               DSLV
254 MP1 = M+1                                                  DSLV
255 MP2 = M+2                                                  DSLV
256 C COUNTER FOR SWITCHING TO A-M AFTER M+2 R-K STEPS     DSLV
257 NBDIF = 0                                                 DSLV
258 GO TO 115                                                 DSLV
259 C                                                         DSLV
260 C FIXED STEP SIZE RUNGE-KUTTA THROUGHOUT                 DSLV
261 112 TRIP = BIG                                             DSLV
262 NSTART = 100                                              DSLV
263 M = 0                                                      DSLV
264 MP1=0                                                      DSLV
265 MP2=0                                                      DSLV
266 115 KON = 2                                               DSLV
267 C KON IS CONTROL FOR R-K LOGIC OR A-M IN MAIN LOOP     DSLV
268 C                                                         DSLV
269 ERROR = .FALSE.                                           DSLV
270 MMP2 = MP1+1                                              DSLV
271 CALL DPIP (HBK(5),HBK(5),HBK(NMX+5),HBK(2*NMX+5),HBK(4*NMX+5), DSLV
272 * BLK(1,5),BLK 1,MP2+5),BLK(1,2*MP2+5), MxEQ,MMP2,     DSLV
273 X HBK, HNOM, NACT, DR1, DR2, ENDS,                        DSLV
274 X NTR1, RT1, VAR1, ZV1, NTR2, RT2, VAR2, ZV2,            DSLV
275 X NTR3, RT3, VAR3, ZV3, NTR4, RT4, VAR4, ZV4,            DSLV
276 X NTR5, RT5, VAR5, ZV5, NTR6, RT6, VAR6, ZV6,            DSLV
277 X NTR7, RT7, VAR7, ZV7, NTR8, RT8, VAR8, ZV8,            DSLV
278 X NTR9, RT9, VAR9, ZV9, NTR10,RT10,VAR10,ZV10,           DSLV
279 X NTR11,RT11,VAR11,ZV11 )                                DSLV
280 C                                                         DSLV
281 C TO RESTART                                              DSLV
282 IF (IRST) GO TO 100                                        DSLV
283 C SUBROUTINE INTERNAL ERROR                                DSLV
284 IF (ERROR) KERR = 1                                        DSLV

```

DESOLV

LIFTING-ROBOT LISTINGS

285
286
287
288
289

```
C      IF (KR .EQ. 100) KERR=100
C      (KR IS RESET TO ZERO IN DESOLV INITIALIZATION)
9999 RETURN
      END
```

DSL
DSL
DSL
DSL
DSL

3

.L.O.

DPIR

LIFTING-ROBOT LISTINGS

T,S 1PF\$.DPIR
PUR 0018-05/17-11:51

DPIR

LIFTING-ROBOT LISTINGS

01*TPF3.DPIR

```

1      SUBROUTINE DPIR (TC,Y,DY,PY,YN,DELZ,DELY,DELX, NNN,LOR,
2      X      H, HNOM, N, DER1, DER2, EOS,
3      X      NTR1, RT1, VAR1, ZV1, NTR2, RT2, VAR2, ZV2,
4      X      NTR3, RT3, VAR3, ZV3, NTR4, RT4, VAR4, ZV4,
5      X      NTR5, RT5, VAR5, ZV5, NTR6, RT6, VAR6, ZV6,
6      X      NTR7, RT7, VAR7, ZV7, NTR8, RT8, VAR8, ZV8,
7      X      NTR9, RT9, VAR9, ZV9, NTR10,RT10,VAR10,ZV10,
8      X      NTR11,RT11,VAR11,ZV11 )
9      C
10     DIMENSION TC(2), Y(NNN), DY(NNN), PY(NNN), YN(NNN)
11     DOUBLE PRECISION PY, YN
12     DIMENSION DELX(NNN,LOR), DELY(NNN,LOR), DELZ(NNN,LOR)
13     C
14     DIMENSION H(1), HNOM(1), N(1)
15     DIMENSION VAR1(1),VAR2(1),VAR3(1),VAR4(1),VAR5(1),VAR6(1),
16     X     VAR7(1),VAR8(1),VAR9(1),VAR10(1),VAR11(1)
17     DIMENSION ZV1(1), ZV2(1), ZV3(1), ZV4(1), ZV5(1),ZV6(1),
18     X     ZV7(1), ZV8(1), ZV9(1), ZV10(1), ZV11(1)
19     DIMENSION NTR1(1),NTR2(1),NTR3(1),NTR4(1),NTR5(1),NTR6(1),
20     X     NTR7(1),NTR8(1),NTR9(1),NTR10(1),NTR11(1)
21     COMMON/MOD1/ACO(10),BCO(10)
22     DATA BIG/1.0E30/
23     C
24     COMMON /DES1/
25     *   KR(1), NMX, NEQ, NOLD, M, MP1, MP2, NN, KIND, KON,
26     *   IFL, NTRG, NIVT, NDVT, NTR,NSTART,NBDIF, NNT, NTT,IDL(11),
27     *   IDL2, JJSS,J1000,J2000,J4000,J7000, IRST, EOST, HD, SMA,
28     *   ERROR,REDOT,STEPNG,
29     *   EU(1), EL, HMX, HMN, YL, A, ENP1, D, B, SKP,
30     *   SK, HC, DELU, DUSC, HOLD, VAR, ZV, HM, HP, R,
31     *   Q, DELT, HMU, FMU,CK(10),QQ(10),XLJ(11), RJ(11), WJ(11)
32     COMMON /DES2/ T, TSV, TGO, TMIN, TRIP, TL, TR
33     DOUBLE PRECISION T, TSV, TGO, TMIN, TRIP, TL, TR
34     LOGICAL EOST, SMA, HD, IRST, ERROR
35     LOGICAL STEPNG
36     LOGICAL ODD, REDOT
37     C
38     GO TO 118
39     3525 TC(1)=TC(1)
40     Y(1)=Y(1)
41     DY(1)=DY(1)
42     PY(1)=PY(1)
43     YN(1)=YN(1)
44     DELZ(1,1)=DELZ(1,1)
45     DELY(1,1)=DELY(1,1)
46     DELX(1,1)=DELX(1,1)
47     NNN=NNN
48     LOR=LOR
49     H=H
50     HNOM=HNOM
51     N=N
52     NTR1(1)=NTR1(1)
53     NTR2(1)=NTR2(1)
54     NTR3(1)=NTR3(1)
55     NTR4(1)=NTR4(1)
56     NTR5(1)=NTR5(1)

```

DPIR

LIFTING-ROBOT LISTINGS

```

57      NTR6(1)=NTR6(1)
58      NTR7(1)=NTR7(1)
59      NTR8(1)=NTR8(1)
60      NTR9(1)=NTR9(1)
61      NTR10(1)=NTR10(1)
62      NTR11(1)=NTR11(1)
63      VAR1(1)=VAR1(1)
64      VAR2(1)=VAR2(1)
65      VAR3(1)=VAR3(1)
66      VAR4(1)=VAR4(1)
67      VAR5(1)=VAR5(1)
68      VAR6(1)=VAR6(1)
69      VAR7(1)=VAR7(1)
70      VAR8(1)=VAR8(1)
71      VAR9(1)=VAR9(1)
72      VAR10(1)=VAR10(1)
73      VAR11(1)=VAR11(1)
74      ZV1(1)=ZV1(1)
75      ZV2(1)=ZV2(1)
76      ZV3(1)=ZV3(1)
77      ZV4(1)=ZV4(1)
78      ZV5(1)=ZV5(1)
79      ZV6(1)=ZV6(1)
80      ZV7(1)=ZV7(1)
81      ZV8(1)=ZV8(1)
82      ZV9(1)=ZV9(1)
83      ZV10(1)=ZV10(1)
84      ZV11(1)=ZV11(1)
85      100 IRST = .TRUE.
86      RETURN
87      C
88      C      SET INITIAL VALUES OF STORAGE AREAS TO ZERO
89      118 CONTINUE
90      DO 120 I=1,NMX
91      PY(I)=Y(I)
92      120 YN(I) = 0.
93      HD = .TRUE.
94      SMA = .TRUE.
95      GO TO (121, 126, 121), KIND
96      C
97      121 DO 125 I=1,NMX
98      DO 125 J=1,MP2
99      DELX(I,J) = 0.
100     DELY(I,J) = 0.
101     125 DELZ(I,J) = 0.
102     A = -ACO(MP1)/BCO(MP2)
103     126 HC = H
104     C
105     C      TO SET UP INDICATOR BLOCK FOR TRIGGERS THAT ARE DEPENDENT
106     C      (IDL = 1) OR INDEPENDENT (IDL = 0), ALSO CALCULATES THE
107     C      NUMBER OF EACH
108     NIVT = 0
109     IF (NTRG) 131, 131, 128
110     128 TEMP=TC(1)
111     TC(1)=-BIG
112     JUSS = 2
113     DO 130 J=1,NTRG

```

```

114      IDL(J) = 1
115      JNT=J
116      GO TO 8000
117      58 IF (VAR .NE. -BIG) GO TO 60
118 C      THIS INDICATES A T-STOP, SINCE T(1)
119 C      WAS SET EQUAL TO -BIG AT THE BEGINNING OF THIS LOOP
120      NIVT = NIVT+1
121      IDL(J)=0
122      60 CONTINUE
123      XLJ(J) = 0.
124      RJ(J) = 0.
125      130 WJ(J) = 0.
126      TC(1) = TEMP
127      131 T = TC(1)
128      62 NDVT = NTRG - NIVT
129 C
130 C      CALC DERIVATIVES AT THE CURRENT TIME
131      GO TO (172, 171), NN
132      171 CALL DER1
133      172 CALL DER2
134 C      GO TO ROUTINE TO GET THE CURRENT TMIN AND TRIGGER ROUTINE
135 C      AND NUMBER THAT GOES WITH IT. ROUTINE IS AT 1000
136      J1000 = 1
137      GO TO 1000
138      205 CONTINUE
139 C
140 C      GO TO END-OF-STEP CALCULATIONS, IF ANY
141      IF (EOST) CALL EOS
142      IF (KR .NE. 0) GO TO 4910
143      DELU = T
144      IF (HC .GT. DELU) DELU=HC
145      DELU = DUSC*DELU
146 C
147 C***** MAIN LOOP *****
148 C
149      300 CONTINUE
150      INDIS = 0
151 C      TEST FOR CALCULATION ERROR
152      310 IF (TMIN + DELU .LT. T) GO TO 9000
153      TS = DABS(TMIN-T)
154      IF (TS .GT. DELU) GO TO 350
155 C
156 C      EXECUTE INDEPENDENT VARIABLE TRIGGER ROUTINE
157      NNT = NTT
158      J4000 = 1
159      GO TO 4000
160 C      TEST FOR A DISCONTINUITY CAUSING A RESTART
161      320 IF (INDIS) 100, 325, 100
162 C      AFTER EXECUTING TRIGGER, RE-CALC. NEW TMIN
163      325 J1000 = 2
164      GO TO 1000
165      330 GO TO 310
166 C
167 C      CALCULATE L'S FOR EACH Y-STOP
168      350 IF (NDVT .EQ. 0) GO TO 380
169      TL = T
170      JUSS = 3

```

UPIR

LIFTING-ROBOT LISTINGS

```

171      DO 375 J=1,NTRG
172      UNT=J
173      IF (IDL(J)) 360, 375, 360
174      360 GO TO 8000
175      365 IF (NTR) 375, 370, 370
176      370 XLJ(J) = VAR - ZV
177      IF (XLJ(J)) 375,371,375
178      371 IDL(J)=3
179      GO TO 3500
180      375 CONTINUE
181  C
182  C  MOVE ONE STEP FORWARD, METHOD DEPENDS ON KON
183      380 GO TO ( 5000 , 6000 ) , KON
184  C
185  C  ADAMS - MOULTON CONTROL LOGIC
186  C
187      5000 CONTINUE
188      IF (DABS(T-TGO) .GT. DELU) GO TO 5400
189  C  MOVE FORWARD ONE HC STEP
190      J5500 = 1
191      GO TO 5500
192  C  ON RETURN FROM ADAMS TEST FOR STEP-SIZE FLAG
193      5020 GO TO (5100, 5200, 5300), IFL
194  C
195  C  NORMAL RETURN, LEAVE STEP SIZE ALONE
196      5100 CONTINUE
197      IF (STEPNG) GO TO 5150
198      5105 TGO = T
199      DELU = T
200      IF (HC .GT. DELU) DELU=HC
201      DELU = DUSC*DELU
202      GO TO 5400
203  C
204  C  AT END OF FIRST HC AFTER SWITCHING TO ADAMS MODE, TEST HC TO SEE
205  C  IF INITIAL STEP WAS TOO BIG. IF SO RETURN CONTROL TO CALLING PROG
206      5110 FORMAT (//22H INITIAL STEP SIZE OF, 1PE12.5,
207      *          17H HAS BEEN CUT TO, E12.5)
208      5150 STEPNG = .FALSE.
209      IF (HC .GE. HSWICH) GO TO 5105
210  C  STEP HAS BEEN CUT DOWN
211      WRITE (6,5110) HSWICH, HC
212      KR = 100
213      GO TO 9999
214  C
215  C  STEP SIZE IS TOO BIG. HALVE STEP SIZE AND RESET CONDITIONS AT
216  C  BEGINNING OF STEP. DIFFERENCE TABLES MUST BE REGENERATED FOR HC/
217      5200 HC = HC/2.0
218      H = HC
219  C
220  C  USING DIFFERENCES IN DELX GET (M+1) DERIVATIVES BACK FROM THIS
221  C  TIME AND STORE IN DELZ. (DELX STORAGE DESTROYED)
222      DO 5220 I=1,NEQ
223      DO 5210 J=1,MP1
224      MM= MP2-J
225      DELZ(I,J)=DELX(I,1)
226      DO 5210 K=1,MM
227      5210 DELX(I,K) = DELX(I,K) - DELX(I,K+1)

```

```

228 5220 DELZ(I,MP2) = DELX(I,1)
229 C
230 C USING THESE DERIVATIVES IN DELZ AND DERIVATIVES AT STEPS HALF WAY
231 C BETWEEN THESE (WHICH ARE CALCULATED USING BACKWARD DIFFERENCE
232 C INTERPOLATION), SET UP A TABLE OF (M+1) BACK DERIVATIVES
233 C USING HC/2 AND STORE IN DELY
234 FM=M
235 FMP1=MP1
236 C CALC. (M+1) FACTORIAL
237 FFMP1=1.0
238 DO 5230 J=2,MP1
239 FJ=J
240 5230 FFMP1=FFMP1*FJ
241 C
242 ODD= .FALSE.
243 DO 5270 NM=1,MP2
244 IF (ODD) GO TO 5250
245 C
246 C USE Y DOT FROM DELZ (N=0,2,4,---)
247 NMM= NM/2 + 1
248 DO 5240 I=1,NEQ
249 5240 DELY (I,NM) = DELZ(I,NMM)
250 GO TO 5270
251 C
252 C CALC. Y DOT HALF WAY BETWEEN (N=1,3,5, ---)
253 5250 FN=NM-1
254 FMU=-FN/2.0
255 PROD=1.
256 DO 5255 J=1,MP1
257 FJ=J
258 5255 PROD = PROD*(FMU+FJ)
259 C CALC. A(0)
260 DO 5265 I=1,NEQ
261 AK = PROD/FFMP1
262 SUM = AK*DELZ(I,1)
263 C
264 DO 5260 K=2,MP2
265 FK=K-2
266 AK=-AK*(FMU+FK)/(FMU+FK+1.) * (FMP1-FK)/(FK+1.)
267 5260 SUM = SUM+ AK*DELZ(I,K)
268 5265 DELY (I,NM) = SUM
269 C
270 5270 ODD = .NOT. ODD
271 C
272 C NOW HAVE (M+1) BACK DERIVATIVES AT HC/2 SAVED IN DELY
273 C RE-GENERATE NEW BACKWARD DIFFERENCE TABLES IN DELX
274 K1=MP2
275 5275 DO 5280 I=1,NEQ
276 TP1 = DELY(I,K1)
277 DO 5280 K=1,MP2
278 TP2=TP1
279 TP1=TP1-DELX(I,K)
280 5280 DELX(I,K) = TP2
281 K1=K1-1
282 IF (K1) 5285, 5285, 5275
283 C
284 C GO TO INTEGRATE ONE ADAMS STEP AGAIN WITH HALF THE STEP SIZE.

```

UPIR

LIFTING-ROBOT LISTINGS

```
285 C          SETTING J5500=1 FORCES RETURN TO STATEMENT 5020
286 5285 J5500=1
287     GO TO 5500
288 C
289 C          STEP SIZE IS SMALL.  PREPARE TO DOUBLE HC.
290 5300 HD = .FALSE.
291     FMP1 = MP1
292     TRIP = T + FMP1*HC
293     NSTART = -1
294 C
295 C
296 C          MOVE DELX TO DELZ AND USING THE DIFFERENCES FROM HERE GET THE
297 C          M+1 BACK DERIVATIVES AND SAVE THEM IN DELY
298 C
299 C          DEL(T-HC)**R = DEL(T)**R - DEL(T)**(R+1)
300 DO 5310 I=1, NEQ
301 DO 5310 K=1, MP2
302 5310 DELZ(I,K) = DELX(I,K)
303 C
304 C
305 DO 5350 I=1, NEQ
306 DO 5330 J=1, MP1
307 MM = MP2 - J
308 DELY(I,J) = DELZ(I,1)
309 DO 5330 K=1, MM
310 5330 DELZ(I,K) = DELZ(I,K)-DELY(I,J)
311 5350 DELY(I,MP2) = DELZ(I,1)
312 C
313     GO TO 5100
314 C
315 C
316 5400 IF (TMIN .LE. (TGO-DELU)) GO TO 5430
317 C          ALL T-STOPS IN THIS STEP HAVE BEEN EXECUTED
318 C          SET CONDITIONS BACK TO TGO
319     T = TGO
320     TC=T
321 DO 5410 I=1,NEQ
322 Y(I) = PY(I)
323 5410 DY(I) = DELX(I,1)
324 C
325     IF (REDOT) GO TO 5425
326     GO TO (5416,5415), NN
327 5415 CALL DER1
328 5416 CALL DER2
329     REDOT = .TRUE.
330 5425 IF (EOST) CALL LOS
331     IF (KR .NE. 0) GO TO 4910
332     GO TO 6100
333 C
334 C          INTERPOLATE FOR CONDITIONS AT TMIN
335 5430 T = TMIN
336     TC = T
337     J7000 = 1
338     GO TO 7000
339 5450 GO TO 6100
340 C
341 C          USE RUNGE-KUTTA THIS STEP
```

```

342 C      CALCULATE STEP SIZE TO USE
343 6000 CONTINUE
344     DT = AMIN1( H,TS)
345     J2000 = 1
346     GO TO 2000
347 6010 TGO = T
348     DELU = T
349     IF (HC .GT. DELU) DELU=HC
350     DELU = DUSC*DELU
351 C
352 C      GO TO END-OF-STEP
353     IF (EOST) CALL EOS
354     IF (KR .NE. 0) GO TO 4910
355 C      TEST FOR ANY Y-STOP IN THIS STEP
356 C
357 C      FLAG ROUTINE
358 6100 CONTINUE
359     IDL2 = 0
360     IF (NDVT) 6105, 300, 6105
361 6105 JUSS = 4
362 C      FLAG TRIGGERS THAT HAVE HAD A CHANGE OF SIGN IN THIS STEP BY
363 C      SETTING THE INDICATOR IDL = 2
364     DO 6150 J=1,NTRG
365     JNT=J
366     IF (IDL(J)) 6110, 6150, 6110
367 6110 GO TO 8000
368 6120 IF (NTR) 6150, 6130, 6130
369 6130 TEMP = VAR - ZV
370     IF (SIGN(1.,TEMP) .NE. SIGN(1.,XLJ(J)) ) GO TO 6140
371     IDL(J) = 1
372     GO TO 6150
373 6140 IDL(J) = 2
374     IDL2 = IDL2+1
375     RJ(J) = TEMP
376     WJ(J) = TEMP
377 6150 CONTINUE
378 C
379     IF (IDL2 .EQ. 0) GO TO 300
380 C
381 C
382 C      SEARCH ROUTINE
383 C      SEARCH AND FLAG FIRST Y-STOP IN THIS TIME STEP
384 3000 CONTINUE
385 C
386 3010 HM = BIG
387     HP = BIG
388     JUSS = 5
389     DO 3100 J=1,NTRG
390     IF (IDL(J) .LT. 2) GO TO 3100
391     JNT=J
392     GO TO 8000
393 3110 IF (NTR) 3100, 3115, 3115
394 3115 WJ(J) = VAR - ZV
395     IF (SIGN(1.,WJ(J)) .NE. SIGN(1.,XLJ(J))) GO TO 3120
396     IF (SIGN(1.,WJ(J)) .NE. SIGN(1.,RJ(J))) GO TO 3140
397 C      UNFLAG DEPENDENT VARIABLE J
398     IDL(J) = 1

```

DPIR

LIFTING-ROBOT LISTINGS

```

399      GO TO 3100
+00      C
+01      3120 R = WJ(J)*(TL-T)/(WJ(J)-XLJ(J))
+02          IF (R .LT. HM) HM = R
+03      GO TO 3150
+04      C
+05      3140 R = WJ(J)*(TR-T)/(WJ(J)-RJ(J))
+06          IF (R .LT. HP) HP = R
+07      3150 Q = R
+08          IDL(J) = 2
+09          IF (ABS(Q) .LT. DELU) IDL(J) = 3
+10      3100 CONTINUE
+11      C
+12      3180 IF (HM .EQ. BIG) GO TO 3220
+13          DO 3200 J=1,NTRG
+14      3200 RJ(J) = WJ(J)
+15          TR = T
+16          DELT = HM
+17          GO TO 3260
+18      3220 DO 3240 J=1,NTRG
+19      3240 XLJ(J) = WJ(J)
+20          TL = T
+21          DELT = HP
+22      3260 IF (ABS(DELT) .LT. DELU) GO TO 3500
+23      C
+24      C      NOT CONVERGED
+25          GO TO (3300, 3400), KON
+26      C      FOR ADAMS-MOULTON, INTERPOLATE FOR Y'S AT T + DELT
+27      C      GET DERIVATIVES AND EOS CONDITIONS
+28      3300 T = T + DELT
+29          TC=T
+30          J7000 = 2
+31          GO TO 7000
+32      3350 GO TO 3000
+33      C
+34      C      FOR RUNGE-KUTTA, INTEGRATE TO T + DELT
+35      3400 DT = DELT
+36          J2000 = 2
+37          GO TO 2000
+38      3450 IF (EOST) CALL EOS
+39          GO TO 3000
+40      C
+41      C      SEARCH HAS CONVERGED WITHIN ACCEPTABLE TOLERANCE ON ONE OF
+42      C      DEPENDENT VARIABLE TRIGGERS
+43      3500 CONTINUE
+44          INDIS = 0
+45          DO 3540 J=1,NTRG
+46          JJSS = 6
+47          JNT=J
+48          GO TO 8000
+49      3520 IF (NTR) 3540,3530,3530
+50      3530 IF (IDL(J) .NE. 3) GO TO 3540
+51      C      GO TO EXECUTE TRIGGER
+52          NNT = J
+53          IDL(J) = 1
+54          J4000 = 2
+55          GO TO 4000

```

DPIR

LIFTING-ROBOT LISTINGS

```

+56      3540 CONTINUE
+57      C
+58      C      CALC. NEW TMIN AND NTT
+59          J1000 = 5
+60          GO TO 1000
+61      3600 CONTINUE
+62      C
+63          GO TO (3680, 3700), KON
+64      C      TEST FOR DISCONTINUITIES, CAUSES A RESTART IN ADAMS
+65      3680 IF (INDIS.EQ. 0) GO TO 350
+66          GO TO 100
+67      C
+68      C      TEST FOR DISCONTINUITIES, CAUSES DERIVATIVES TO BE RE-CALCULATED
+69      C          IN KUTTA
+70      3700 TGO = T
+71          IF (INDIS.EQ. 0) GO TO 300
+72      C
+73          GO TO (3752, 3751), NN
+74      3751 CALL DER1
+75      3752 CALL DER2
+76          IF (EOST) CALL EOS
+77          IF (KR.NE. 0) GO TO 4910
+78          GO TO 300
+79      C
+80      C
+81      C*****
+82      C ROUTINE TO CALCULATE CURRENT TMIN FROM LIST OF T-STOP
+83      C      ALSO CHECKS FOR ANY CHANGE IN ARGUMENTS THAT WILL CAUSE A RESTART
+84      C          STARTS AT 1000 AND RETURNS BY A CALCULATED GO TO USING J1000
+85      C
+86      1000 CONTINUE
+87          TMIN = TRIP
+88          NTT = NSTART
+89          IF (NIVT) 1060, 1060, 1010
+90      1010 JJSS = 1
+91          DO 1050 J=1,NTRG
+92          JNT=J
+93          IF (IDL(J)) 1050, 1020, 1050
+94      1020 GO TO 8000
+95      1030 IF (NTR) 1050, 1040, 1040
+96      1040 IF (ZV.GT.TMIN)GO TO 1050
+97          IF (ABS(ZV-TMIN).LE.DELU)GO TO 1050
+98          TMIN = ZV
+99          NTT = J
500      1050 CONTINUE
501      1060 CONTINUE
502      C      CHECK POSSIBILITY OF A RESTART DUE TO A CHANGE IN ARGUMENTS
503          IF (NOLD = N) 100, 1075, 1070
504      1070 NOLD = N
505          NEQ = N
506      1075 IF (HNOM.EQ. HOLD) GO TO 1080
507      C      THE NOMINAL STEP SIZE HAS BEEN CHANGED EXTERNALLY
508      C          MUST RESTART.
509          H = HNOM
510          HOLD = HNOM
511          GO TO 100
512      C

```

DPIR

LIFTING-ROBOT LISTINGS

```

1080 GO TO (205,330,5505,5570,3600), J1000
C
C*****
513 1080 GO TO (205,330,5505,5570,3600), J1000
514 C
515 C*****
516 C  FOURTH ORDER RUNGE-KUTTA ROUTINE
517 C    TO MOVE ONE STEP FROM T TO T + DT (INITIALIZE DT BEFORE CALLING)
518 C    USES YN TO SAVE PY(T)
519 C    PY AND Y CALC. TOGETHER TO T+DT
520 C    DELZ(I,1) TO SAVE SUM OF K'S
521 C    RETURN IS BY J2000
522 C
523 2000 TSV = T
524 PDT = DT/2.
525 C    SAVE INITIAL Y'S IN YN
526 DO 2002 I=1,NEQ
527 2002 YN(I) = PY(I)
528 C
529 C    OUTER LOOP
530 DO 2060 L=1,4
531 C    INNER LOOP  FOR ALL N EQUATIONS
532 DO 2050 I=1,NEQ
533 SK = DT*DY(I)
534 GO TO ( 2010, 2020, 2030, 2040), L
535 C    L=1, SK=K1
536 2010 DELZ(I,1) = SK
537 2015 PY(I) = YN(I) + SK/2.
538 Y(I) = PY(I)
539 GO TO 2050
540 C    L=2, SK=K2
541 2020 DELZ(I,1) = DELZ(I,1)+2.*SK
542 GO TO 2015
543 C    L=3, SK=K3
544 2030 DELZ(I,1) = DELZ(I,1) + 2.*SK
545 PY(I) = YN(I) + SK
546 Y(I) = PY(I)
547 GO TO 2050
548 C    L=4, SK=K4
549 2040 PY(I) = YN(I) + (DELZ(I,1) + SK)/6.
550 Y(I) = PY(I)
551 2050 CONTINUE
552 C
553 GO TO ( 2051, 2057, 2053, 2057), L
554 2051 T = TSV + PDT
555 GO TO 2055
556 2053 T = TSV + DT
557 2055 TC = T
558 GO TO ( 2057, 2056), NN
559 2056 CALL DER1
560 2057 CALL DER2
561 2060 CONTINUE
562 C
563 C    RETURN
564 GO TO ( 6010, 3450), J2000
565 C*****
566 C*****
567 C
568 C    CALLS TO EXECUTE TRIGGER ROUTINES.  NUMBER IS SET IN NNT
569 C    NNT MAY EQUAL ZERO WHEN GETTING THE FIRST M+2

```

DPIR

LIFTING-ROBOT LISTINGS

```

C          DIFFERENCES BY RUNGE-KUTTA BEFORE SWITCHING TO ADAMS
C
C          NNT MAY EQUAL -1 WHEN ENOUGH DIFFERENCES ARE SAVED
C          TO DOUBLE THE STEP SIZE
C          RETURN IS BY J4000
C
576 4000 CONTINUE
577     IF (NNT) 4500, 4400, 4005
578 C
579 4005 IF (NNT .GT. NTRG) GO TO 4300
580     GO TO (4010,4020,4030,4040,4050,4060,4070,4080,4090,4100,
581     X      4110), NNT
582 C
583 4010 CALL RT1
584     GO TO 4900
585 4020 CALL RT2
586     GO TO 4900
587 4030 CALL RT3
588     GO TO 4900
589 4040 CALL RT4
590     GO TO 4900
591 4050 CALL RT5
592     GO TO 4900
593 4060 CALL RT6
594     GO TO 4900
595 4070 CALL RT7
596     GO TO 4900
597 4080 CALL RT8
598     GO TO 4900
599 4090 CALL RT9
600     GO TO 4900
601 4100 CALL RT10
602     GO TO 4900
603 4110 CALL RT11
604     GO TO 4900
605 C
606 4310 FORMAT (///31H TRYING TO CALL TRIGGER ROUTINE, I3,
607     X      21H WHICH IS NOT DEFINED/)
608 4300 WRITE (6,4310) NNT
609     GO TO 9900
610 C
611 C
612 C          ROUTINE TO EXECUTE WHEN TAKING THE FIRST M+2 RUNGE-KUTTA STEPS
613 C          AND GENERATING THE NECESSARY DIFFERENCE TABLES.
614 C          WHEN NBDIF = M+2 THE TRIGGER IS SHUT OFF AND CONTROL IS
615 C          TRANSFERED TO ADAMS CONTROL.
616 4400 CONTINUE
617     TRIP = T + HC
618 C          UPDATE DIFFERENCES TABLES
619     DO 4410 I=1,NEQ
620         TP1 = DY(I)
621     DO 4410 K=1,MP2
622         TP2 = TP1
623         TP1 = TP1 - DELX(I,K)
624 4410 DELX(I,K) = TP2
625 C
626     NBDIF = NBDIF+1

```

DPIR

LIFTING-ROBOT LISTINGS

```

027      IF (NBDIF .LT. MP2) GO TO 4995
028      C
029      C      WHEN ENOUGH STEPS HAVE BEEN TAKEN, SWITCH TO ADAMS-MOULTON
030      KON = 1
031      HSWICH = HC
032      TRIP = BIG
033      NSTART = 100
034      GO TO 325
035      C
036      C      ROUTINE TO DOUBLE THE STEP SIZE AFTER ENOUGH DIFFERENCES
037      C      HAVE BEEN SAVED
038      C      TIME OF THIS TRIGGER IS (M+1)*HC AFTER THE TIME WHEN THE DOUBLING
039      C      PROCEDURE WAS SET OFF
040      4500 CONTINUE
041      HC = 2.*HC
042      H = HC
043      HD = .TRUE.
044      TRIP = BIG
045      NSTART = 100
046      C
047      C      USING DIFFERENCES IN DELX GET (M+1) DERIVATIVES BACK FROM THIS
048      C      TIME AND STORE THESE IN DELZ
049      DO 4530 I=1,NEQ
050      DO 4520 J=1,MP1
051      MM = MP2-J
052      DELZ(I,J) = DELX(I,1)
053      DO 4520 K=1,MM
054      4520 DELX(I,K) = DELX(I,K) - DELX(I,K+1)
055      4530 DELZ(I,MP2) = DELX(I,1)
056      C
057      C      FROM DERIVATIVES IN DELY AND DELZ, SET DELY= BACK DERIVATIVES
058      C      WITH 2*HC TIME INTERVAL BETWEEN
059      ASSIGN 4542 TO ISW
060      I1 = MP2+1
061      I2 = MP2+2
062      4535 I1 = I1-1
063      IF (I1) 4560, 4560, 4536
064      4536 I2 = I2-2
065      IF (I2) 4538, 4538, 4540
066      C      SWITCH TO DELZ TABLES
067      4538 ASSIGN 4544 TO ISW
068      I2 = I2 + MP1
069      C
070      C      IN SAME LOOP, RE-CALC. NEW DIFFERENCE TABLES IN DELX BASED ON NEW
071      C      TIME STEP
072      4540 DO 4550 I=1,NEQ
073      GO TO ISW, (4542,4544)
074      4542 DELY(I,I1) = DELY(I,I2)
075      GO TO 4545
076      4544 DELY (I,I1) = DELZ(I,I2)
077      4545 TP1 = DELY(I,I1)
078      DO 4550 K=1,MP2
079      TP2 = TP1
080      TP1 = TP1-DELX(I,K)
081      4550 DELX(I,K) = TP2
082      C
083      GO TO 4535

```

DPIR

LIFTING-ROBOT LISTINGS

```

584 C
585 4560 GO TO 4995
586 C
587 C
588 C AFTER EXECUTING AN EXTERNAL TRIGGER ROUTINE, COME HERE
589 4900 CONTINUE
590 C HAS RTMRK BEEN CALLED
591 IF (KR .EQ. 0) GO TO 4920
592 C YES, RETURN TO CALLING PROGRAM
593 4910 KR = 0
594 GO TO 9999
595 C
596 C TEST FOR A DISCONTINUITY AFTER TRIGGER EXECUTION
597 4920 JNT = NNT
598 JSS = 7
599 GO TO 8000
700 4930 CONTINUE
701 C
702 C IF THERE IS A DISCONTINUITY, SET INDICATOR
703 IF ( IABS(NTR) .EQ. 2) INDIS = 1
704 C
705 4995 GO TO ( 320 , 3540 ) , J4000
706 C
707 C
708 C*****
709 C
710 C ADAMS - MOULTON M-TH ORDER PREDICTOR - CORRECTOR ROUTINE WITH
711 C AUTOMATIC STEP SIZE CONTROL
712 C MOVES ONE STEP FORWARD USING CURRENT STEP SIZE, HC
713 C RETURNS WITH AN INDICATOR, IFL
714 C
715 C
716 C IF IFL = 1, NORMAL RETURN
717 C TIME IS UPDATED
718 C Y'S ARE IN Y AND PY
719 C Y DERIVATIVES ARE IN DY
720 C BACK DIFFERENCES UPDATED AND IN DELX
721 C
722 C IF IFL = 2, INDICATES STEP SIZE IS TOO BIG
723 C TIME IS SET BACK TO BEGINNING OF STEP
724 C OLD Y'S ARE PUT BACK INTO Y AND PY
725 C OLD BACK DIFFERENCES ARE STILL IN DELX
726 C OLD DERIVATIVES SET IN DY FROM FIRST COL. OF DELX
727 C END-OF-STEP CONDITIONS ARE RE-CALCULATED
728 C ANY DOU LING INDICATORS ARE SHUT OFF
729 C
730 C IF IFL = 3, INDICATES STEP SIZE IS TOO SMALL
731 C STATE OF STORAGE IS THE SAME AS FOR NORMAL RETURN
732 C
733 C RETURN IS BY J5500
734 C
735 5500 CONTINUE
736 REDOT = .TRUE.
737 IF (SMA) GO TO 5505
738 C GET NEW TMIN AND NTT
739 SMA = .TRUE.
740 J1000 = 3

```

```

1      GO TO 1000
2      5505 TSV = T
3      C      SAVE CURRENT Y'S IN YN AND CALC PREDICTED Y'S
4      C
5          DO 5520 I=1,NEQ
6          YN(I) = PY(I)
7          SK = 0.
8      C      SUM FROM K=0 TO M
9          DO 5510 K=1,MP1
10         5510 SK = SK + ACO(K)*DELX(I,K)
11         PY(I) = YN(I) + HC*SK
12         5520 Y(I) = PY(I)
13      C      CALCULATE DERIVATIVES OF PREDICTED Y'S AT T + HC
14         T = TSV + HC
15         TC=T
16         GO TO (5526, 5525), NN
17         5525 CALL DER1
18         5526 CALL DER2
19      C
20         IFL=1
21         GO TO (5528, 9900, 5590), KIND
22      C      LEAVE DIFFERENCES AT BEG. OF STEP IN DELX
23      C      UPDATE DIFFERENCE TABLES FROM DELX USING PREDICTED
24      C      DERIVATIVES IN DELZ
25         5528 CONTINUE
26         DO 5530 I=1,NEQ
27         TP1 = DY(I)
28         DO 5530 K=1,MP2
29         TP2 = TP1
30         TP1 = TP1 - DELX(I,K)
31         5530 DELZ(I,K) = TP2
32      C
33      C      CALCULATE CORRECTED Y'S AND GET LARGEST ERROR TERM
34         ENP1 = 0.
35         DO 5550 I=1,NEQ
36         SKP=0.
37         SK = 0.
38         DO 5540 K=1,MP1
39         SKP = SKP+ACO(K)*DELX(I,K)
40         5540 SK = SK + BCO(K)*DELZ(I,K)
41         PY(I) = YN(I) + HC*SK
42         Y(I) = PY(I)
43         D = ABS(Y(I))
44         IF (D .LE. YL) D = YL
45         B = ABS(HC*(SKP-SK))/D/A
46         5550 IF (B .GT. ENP1) ENP1 = B
47      C      CALC. ERIVATIVES OF CORRECTED Y'S
48         CALL DER2
49      C      TEST ERROR LIMITS FOR AUTOMATIC STEP SIZE CONTROL
50         IF (ENP1 .LE. EL) GO TO 5580
51      C
52         IF (ENP1 .LE. EU) GO TO 5590
53      C      STEP SIZE IS TOO BIG
54         IF (HC/2. .LT. HMN) GO TO 5590
55      C      RESET EVERYTHING TO CONDITIONS AT BEGINNING OF STEP
56      C      INCLUDING END OF STEP CONDITIONS
57      C      Y(I) AND PY(I) FROM YN(I)

```

DPIR

LIFTING-ROBOT LISTINGS

```

798 C          DY(I) FROM 1ST COLUMN OF DELX
799     T = TSV
800     TC=T
801     DO 5560 I=1,NEQ
802     Y(I) = YN(I)
803     PY(I) = YN(I)
804     5560 DY(I) = DELX(I,1)
805     IF (EOST) CALL EOS
806 C
807     IFL = 2
808     HD = .TRUE.
809     TRIP = BIG
810     NSTART = 100
811     J1000 = 4
812 C          TO RECALCULATE TMIN AND NTT
813     GO TO 1000
814     5570 GO TO 5599
815 C
816 C     STEP SIZE IS TOO SMALL
817     5580 IF ( .NOT. HD) GO TO 5590
818     IF (2.*HC .GT. HMX) GO TO 5590
819     IFL = 3
820     SMA = .FALSE.
821 C
822 C     UPDATE DIFFERENCE TABLES USING CORRECTED DERIVATIVES
823     5590 DO 5595 I=1,NEQ
824     TP1 = DY(I)
825     DO 5595 K=1,MP2
826     TP2 = TP1
827     TP1 = TP1 - DELX(I,K)
828     5595 DELX(I,K) = TP2
829 C
830 C     RETURN
831     5599 GO TO (5020), J5500
832 C
833 C*****
834 C
835 C ROUTINE TO INTERPOLATE USING DIFFERENCE FORMULAS FOR THE Y'S AT TIME
836 C SET IN T AND USING DIFFERENCE TABLES EVALUATED AT TGO. BEFORE
837 C RETURNING THE DERIVATIVES ARE CALCULATED AND STORED IN DY AND
838 C THE END-OF-STEP ROUTINE HAS BEEN EXECUTED.
839 C     RETURN IS BY J7000
840 C
841     7000 CONTINUE
842     REDOT = .FALSE.
843     HMU = TGO - T
844     FMU = HMU/HC
845     FACTJ = 1.0
846     FNUM = 1.0
847     DO 7020 J=1,M
848     FJ = J
849     FACTJ=FACTJ*(FJ+1.)
850     FNUM = FNUM*(FMU-FJ)
851     7020 QQ(J) = (-1.)*J * FNUM/FACTJ
852 C     CALCULATE COEFFICIENTS OF DIFFERENCES
853     DO 7050 K=1,MP1
854     CK(K) = BCO(K)

```

DPIK LIFTING-ROBOT LISTINGS

```

355      IF (K .EQ. 1) GO TO 7050
356      KK = K-1
357      DO 7040 J=1, KK
358      KJ = K-J
359      7040 CK(K) = CK(K) + QQ(J)*BCO(KJ)
360      7050 CONTINUE
361      C
362      C      CALCULATE NEW Y'S AT T
363      DO 7080 I=1, NEQ
364      SUM = 0.
365      DO 7060 K=1, MP1
366      7060 SUM = SUM + CK(K)*DELX(I,K)
367      7080 Y(I) = PY(I) - HMU*SUM
368      C
369      C      CALCULATE DERIVATIVES
370      GO TO (7092, 7091), NN
371      7091 CALL DER1
372      7092 CALL DER2
373      C      GET EOS CONDITIONS
374      IF (EOST) CALL EOS
375      IF (KR .NE. 0) GO TO 4910
376      C
377      GO TO (5450, 3350), J7000
378      C
379      C
380      C*****
381      C      ROUTINE TO SET UP TRIGGER VARIABLES WITH CURRENT VALUES
382      8000 GO TO (1,2,3,4,5,6,7,8,9,10,11), JNT
383      C      RETURN TO THESE STATEMENT NOS. DEPENDING ON VALUE OF JJSS
384      C      JJSS = 1 , 2 , 3 , 4 , 5 , 6 , 7
385      8100 GO TO (1030,58,365,6120,3110,3520,4930), JJSS
386      C
387      1 NTR = NTR1
388      VAR = VAR1
389      ZV = ZV1
390      GO TO 8100
391      2 NTR = NTR2
392      VAR = VAR2
393      ZV = ZV2
394      GO TO 8100
395      3 NTR = NTR3
396      VAR = VAR3
397      ZV = ZV3
398      GO TO 8100
399      4 NTR = NTR4
400      VAR = VAR4
401      ZV = ZV4
402      GO TO 8100
403      5 NTR = NTR5
404      VAR = VAR5
405      ZV = ZV5
406      GO TO 8100
407      6 NTR = NTR6
408      VAR = VAR6
409      ZV = ZV6
410      GO TO 8100
411      7 NTR = NTR7

```

DPIR

LIFTING-ROBOT LISTINGS

```

012      VAR = VAR7
013      ZV = ZV7
014      GO TO 8100
015      8 NTR = NTR8
016      VAR = VAR8
017      ZV = ZV8
018      GO TO 8100
019      9 NTR = NTR9
020      VAR = VAR9
021      ZV = ZV9
022      GO TO 8100
023      10 NTR = NTR10
024      VAR = VAR10
025      ZV = ZV10
026      GO TO 8100
027      11 NTR = NTR11
028      VAR = VAR11
029      ZV = ZV11
030      GO TO 8100
031      C
032      C*****
033      C*****
034      C
035      C
036      9000 WRITE (6,9010) TMIN, T, DELU
037      9010 FORMAT (///10H T(MIN) = ,1PE10.3,28H IS LESS THAN CURRENT TIME, ,
038      X          E10.3, 17H, MINUS DELTA U, ,E10.3/)
039      C
040      9900 ERROR = .TRUE.
041      9999 RETURN
042      END

```

.L.O.

RTMRK

LIFTING-ROBOT LISTINGS

.S TPF\$.RTMRK
PUR 0018-05/17-11:51

RTMRK

LIFTING-ROBOT LISTINGS

DT*TPF\$.RTMRK

```
1      SUBROUTINE RTMRK
2      C
3      COMMON /DES1/
4      *   KR(1), NMX, NEQ, NOLD,   M, MP1, MP2, NN, KIND, KON,
5      *   IFL, NTRG, NIVT, NDVT, NTR,NSTART,NBDIF, NNT, NTT,IDL(11),
6      *   IDL2, JJSS,J1000,J2000,J4000,J7000, IRST, EAST, HU, SMA,
7      *   ERROR,REDOT,STEPNG,
8      *   EU(1), EL, HMX, HMN, YL, A, ENP1, D, B, SKP,
9      *   SK, HC, DELU, DUSC, HOLD, VAR, ZV, HM, HP, R,
10     *   Q, DELT, HMU, FMU,CK(10),QQ(10),XLJ(11), RJ(11), WJ(11)
11     KR(1)=2
12     C
13     RETURN
14     END
```

.L,0.





UNIVAC-1108-TIME-SHARING-EXEC*VER.26.70.98H006*UNIVAC-COMPUTER-CENTER*UNIVERSITY-OF-AL.

34567890123456789012345678901234567890123456789012345678901234567
34567890123456789012345678901234567890123456789012345678901234567