Minimum Key Rendezvous

AUTOCHK Entered from "GOTOPOOH" and "NDUTINPT"

If bit 7(AUTOSEQ) of FLGWRD10 = 0:

Return

Set restart group 4 to cause a start at next line (priority 13_8)
(Tag here "AUTOCHK1"; from "PIKUP20")

Proceed to address specified by AUTPOINT

REND30S Entered from "V37" if input program number in range 31-38 and [REFSMMAT] available

TEMPMM = MMNUMBER + 50 (makes 31-38 into 81-88)

If bit 9(UTFLAG) of FLAGWRD8 = 1:

Set bit 5(TRACKFLG) of FLAGWRD1 = 0

Set bit 9(UTFLAG) of FLAGWRD8 = 0

If bit 7(RNDVZFLG) of FLAGWRD0 = 0:

Set bit 7(AUTOSEQ) of FLGWRD10 = 1

TS = 20

Perform "AUTOSET" ("NDUTINPT", via "AUTOCHK", returns to next line)

MMNUMBER = TEMPMM

Proceed to "AUTO37"

MINKDISP

AUTPOINT = Return address

TS = (MMNUMBER - 50) and perform "NEWMODEX"

Perform "RELDSP"

TS = 00017_8

Proceed to "GOPERF1": if terminate, proceed to "GOTOPOOH"
if proceed, proceed to "STARTAUT"
otherwise, proceed

Set bit 7(AUTOSEQ) of FLGWRD10 = 0

Proceed to address specified by AUTPOINT

STARTAUT Entered from "MINKDISP"

If bit 8(AZIMFLAG) of FLGWRD11 = 0:

Set bit 8(AZIMFLAG) of FLGWRD11 = 1 (effect of "option 4")

If bit 11(HDSUPFLG) of FLGWRD10 = 1:

AZIMANGL = 0

If bit 11(HDSUPFLG) of FLGWRD10 = 0:

AZIMANGL = $\frac{1}{2}$ (i.e. 180°)

If bit 1(RENDWFLG) of FLAGWRD5 = 0:

Set bit 5(MANEUFLG) of FLGWRD10 = 1

Set bit 4(PTV93FLG) of FLGWRD10 = 1

Set bit 1(PCFLAG) of FLGWRD10 = 0

Set bit 7(AUTOSEQ) of FLGWRD10 = 1

Proceed to address specified by AUTPOINT

AUTOSET

AUTTEMP = Return address

MMNUMBER = TS

Set restart group 4 to cause a start at next line

AUTPOINT = AUTTEMP

Proceed to "AUTO37"

BURNHOW

AUTTEMP = Return address

If $|\underline{\text{DELVLVC}}| - K_{dv40d41} < 0$:

TS = 41

Proceed to second line of "AUTOSET"

TS = 40

Proceed to second line of "AUTOSET"

P81

```
Perform "MINKDISP"
TS = 31
Perform "AUTOSET"
Perform "BURNHOW"
Proceed to second line of "P82"
```

P82

```
Perform "MINKDISP"
TS = 32
Perform "AUTOSET"
Perform "BURNHOW"
Proceed to second line of "P83"
```

P83

```
Perform "MINKDISP"
TS = 33
Perform "AUTOSET"
Perform "BURNHOW"
Proceed to second line of "P84"
```

P84

```
Perform "MINKDISP"
TS = 34
Perform "AUTOSET"
Perform "BURNHOW"
Proceed to second line of "P85"
```

P85

```
Perform "MINKDISP"
TS = 35
```

Perform "AUTOSET"

Perform "BURNHOW"

Proceed to second line of "P86"

P86

Perform "MINKDISP"

TS = 36

Perform "AUTOSET"

Perform "BURNHOW"

TS = 36

Perform "AUTOSET"

Perform "BURNHOW"

Proceed to second line of "P87"

P87

Perform "MINKDISP"

TS = 37

Perform "AUTOSET"

TS = 48

Perform "AUTOSET"

Proceed to "GOTOPOOH"

P88

Perform "MINKDISP"

TS = 38

Perform "AUTOSET"

If $|\underline{\text{DELVLVC}}| = 0$: (all components below 2^{-14} m/cs $\approx$ 0.02 fps)

Proceed to "GOTOPOOH"

TS = 52

```
Perform "AUTOSET"
If bit 1(PCFLAG) of FLGWRD10 = 1:    (reset in "GYCRS" if torque)
     TS = 41
     Perform "AUTOSET"
     Proceed to "GOTOPOOH"
Perform "BURNHOW"
Set bit 15(PCMANFLG) of FLGWRD10 = 1       (Tag here "P86CONT2")
TS = 20
Perform "AUTOSET"       ("PIKUP20", since PCMANFLG = 1, after R61 exits)
Set bit 15(PCMANFLG) of FLGWRD10 = 0
TS = 52
Perform "AUTOSET"
Proceed to "GOTOPOOH"
```

AUTOW Entered from "REND4" if bit 7(AUTOSEQ) of FLGWRD10 = 1

```
If bit 5(MANEUFLG) of FLGWRD10 = 1:
     If bit 4(PTV93FLG) of FLGWRD10 = 1:      (Tag here "AUTOW3")
          Proceed to "REND5C"  (initialize W matrix)
     If bit 3(TPIMNFLG) of FLGWRD10 = 1:
          If bit 2(FULTKFLG) of FLGWRD10 = 0: (have both sensors)
               Proceed to "REND5C"
     COUNT3MK = 1          (Tag here "AUTOW4")
     Proceed to "AUTOW2"
If COUNT3MK ≠ 0:
     Proceed to "AUTOW2"
```

If bit 3(TPIMNFLG) of FLGWRD10 = 1: (Tag here "AUTOW1A")

If bit 8(P35FLAG) of FLGWRD10 = 1: (set in P36)

Proceed to "REND5C" (initialize W matrix)

Proceed to "AUTOW2"

TS = T_{ig} (Tag here "AUTOW1")

If bit 1(PCFLAG) of FLGWRD10 = 1:

TS = T_{cdh} (i.e. TNSR)

If (MARKTIME - $C_{wrdtime}$ - AGEOFW) $<$ 0: (matrix not old enough)

Proceed to "AUTOW2"

If (OLDMKTME - MARKTIME + $C_{minblktm}$) $<$ 0: (too long since last mark)

COUNT3MK = 1

Proceed to "AUTOW2"

TS_1 = MARKTIME - TS

If $TS_1 \geq 0$: (note that this means if delay answering initial P3x display, may force initialization)

Set bit 4(PTV93FLG) of FLGWRD10 = 0

Proceed to "REND5C" (initialize W matrix)

If (TS_1 + $C_{fincmptm}$ + $C_{tbefcomp}$) $<$ 0:

Set bit 4(PTV93FLG) of FLGWRD10 = 0

Proceed to "REND5C"

If (TS + $K_{3mincon}$ - $C_{fincmptm}$ + $C_{brnblktm}$ - AGEOFW - $C_{maxwtime}$) $\geq$ 0:

Set bit 4(PTV93FLG) of FLGWRD10 = 1

Proceed to "AUTOW2"

AUTOW2

If bit 1(RENDWFLG) of FLAGWRD5 = 0:

Proceed to "REND5C"

Set bit 1(RENDWFLG) of FLAGWRD5 = 1 (unnecessary)

Proceed to "REND7"

P52AUTO Entered from "PROG52" if bit 7(AUTOSEQ) of FLGWRD10 = 1

If bit 1(PCFLAG) of FLGWRD10 = 1: (i.e. before burn)

TS = (cos CDU_y) sgn $DELVLVC_y$

Set bit 6(TCOMPFLG) of FLGWRD10 = 0

If TS < 0:

Set bit 6(TCOMPFLG) of FLGWRD10 = 1

$\underline{TS} = \frac{1}{2}\ REFSMMAT_3$ (Tag here "P52AUTO1")

If bit 1(PCFLAG) of FLGWRD10 = 0: (after burn)

$\underline{TS} = -\ \underline{TS}$

If bit 6(TCOMPFLG) of FLGWRD10 = 1:

$\underline{TS} = -\ \underline{TS}$

$\underline{X}_{smd} = unit(\frac{1}{2}\ REFSMMAT_0 + \underline{TS})$

$\underline{Z}_{smd} = REFSMMAT_6$

$\underline{Y}_{smd} = unit(\underline{Z}_{smd} * \underline{X}_{smd})$

Set bit 4(PFRATFLG) of FLAGWRD2 = 1

Proceed to "P52D"

PERF20 Entered from "P52D" if bit 7(AUTOSEQ) of FLGWRD10 = 1

$TS = 20_8$

Proceed to "GOPERF1": if terminate, proceed to "GOTOPOOH"
if proceed, proceed to "GYCRS"
otherwise, proceed

If bit 1(PCFLAG) of FLGWRD10 = 1: (before burn)

Proceed to second line of "GOTOPOOH"

Perform "ALARM" (pattern 0402_8)

Proceed to "PERF20"

Quantities in Computations

See also list of major variables and list of routines

AGEOFW: See Measurement Incorporation.

AUTPOINT: Single precision cell used for return address storage in the minimum key rendezvous logic. It could be considered as a "pointer" indicating the progress of the computations through the P8x "driver".

AUTTEMP: Single precision cell used as temporary storage for information to be placed in AUTPOINT (to simplify some program logic and for the sake of restart protection).

AZIMANGL: See Orbital and Rendezvous Navigation.

$C_{brnblktm}$: Single precision erasable memory constant, program notation "BRNBLKTM", scale factor B28, units centi-seconds (least increment 163.84 seconds). Used in determination of age of W matrix following next maneuver, for $C_{maxwtime}$ check.

$C_{fincmptm}$: Single precision erasable memory constant, program notation "FINCMPTM", scale factor B28, units centi-seconds (least increment 163.84 seconds). Used to allow for length of time of final targeting computation.

$C_{maxwtime}$: Single precision erasable memory constant, program notation "MAXWTIME", scale factor B28, units centi-seconds (least increment 163.84 seconds). Used to determine if W matrix age will be excessive after next maneuver.

$C_{minblktm}$: Single precision erasable memory constant, program notation "MINBLKTM", scale factor B28, units centi-seconds (least increment 163.84 seconds). Used to check whether time since last mark is excessive (if so, COUNT3MK is set to 1).

$C_{tbefcomp}$: Single precision erasable memory constant, program notation "TBEFCOMP", scale factor B28, units centi-seconds (least increment 163.84 seconds). Used to check if a sufficient time remains before the final targeting computation so that W matrix reinitialization should take place.

$C_{wrdtime}$: Single precision erasable memory constant, program notation "WRDTIME", scale factor B28, units centi-seconds (least increment 163.84 seconds). Used to check if sufficient time has elapsed since the previous W matrix initialization.

COUNT3MK: Single precision counter, scale factor B14, units counts, used to control the initialization of the W matrix after three marks accumulated (hence the name). It is incremented in "REND12", where it is also set zero after reaching 3. If non-zero, checks in "AUTOW", except those immediately following a burn, are bypassed.

DELVLV$\underline{C}$: See Burn Control.

$K_{3mincon}$: Single precision constant, program notation "3MINCON", scale factor B28, units centi-seconds. Used double precision with octal value 00001_8 00002_8, corresponding to 163.86 seconds or 2.731 minutes.

$K_{dv40d41}$: Constant, program notation "DV40/41", scale factor B7, units meters/centi-second. Value is 0.03048×2^{-7}, corresponding to $10 \times 0.3048 \times 0.01 \times 2^{-7}$, where first term is value in fps, second converts to meters, third to centi-seconds, and fourth is scale factor.

MARKTIME: See Measurement Incorporation.

MMNUMBER: See General Program Control.

OLDMKTME: See Measurement Incorporation.

T_{cdh}: See Rendezvous Computations.

TEMPMM: Temporary storage for program number to be used in searching the program tables in V37 logic (in range 81-88), scale factor B14, single precision. Used in "REND30S" to allow start-up of P20 and then initiation of the desired program.

$\underline{X}_{smd}$, $\underline{Y}_{smd}$, $\underline{Z}_{smd}$: See Inflight Alignment.

Noun Definitions

DPTEST

If TS_1 = 4, 5, 7, 10, or 13, indicate double precision (return to calling address +2)

If TS_1 = 0, 1, 2, 3, 6, 8, 9, 11, 12, or 14, indicate single precision (return to calling address +1)

GTSFOUT

SFTEMP1 = $K_{sfot_{TS}}$

Return

GTSFIN

SFTEMP1 = $K_{sfin_{TS}}$

Return

LODNNTAB

NNADTEM = $K_{ntb_{NOUNREG}}$

NNTYPTEM = $K_{nty_{NOUNREG}}$

If NOUNREG < K_{mxcn}:

MIXBR = 1

Return

MIXBR = 2

TS = NOUNREG - 40

RUTMXTEM = $K_{rtmtb_{TS}}$

TS = bits 10-1 of NNADTEM

IDAD1TEM = $K_{idtb_{TS}}$

IDAD2TEM = $K_{idtb_{TS+1}}$

IDAD3TEM = $K_{idtb_{TS+2}}$

Return

NOTE: IDAD$\underline{I}$TEM$_i$ refers to:

IDAD1TEM if i = 1
IDAD2TEM if i = 2
IDAD3TEM if i = 3

DECDSP3

Proceed to address given by the following table:

TS_1	Address
0	"DSPALARM" (after setting DSPCOUNT = -19)
1	"DSPDCEND"
2	"DEGOUTSF"
3	"ARTOUTSF"
4	"DP1OUTSF"
5	"DP2OUTSF"
6	"OPDEGOUT"
7	"DP3OUTSF"
8	"HMSOUT"
9	"M/SOUT"
10	"DP2OUTSF"
11	"AROUT1SF"
12	"2INTOUT"
13	"DPFRACOT"
14	"ATTERROR"

DEGOUTSF

If MPAC+0 $\leq$ -0:

Set bit 15 of MPAC+0 = 0

SFTEMP1 = K_{dgtb_0}

$MPAC_{dp} = K_{dgtb_0}$ MPAC+0 + SFTEMP1

Proceed to "DSPDCEND"

ARTOUTSF (no shift)

If MPAC+0 = -0:

$MPAC_{dp}$ = -0

Proceed to "DSPDCEND"

$MPAC_{dp}$ = SFTEMP1 MPAC+0

Proceed to "DSPDCEND"

DP1OUTSF (left shift of 14)

Perform "DPOUT"

$MPAC_{dp} = 2^{14}\ MPAC_{tp}$ (left shift of 14, overflow information lost)

Proceed to "DSPDCEND"

DP2OUTSF (no shift)

Perform "DPOUT"

Proceed to "DSPDCEND"

OPDEGOUT

TS = MPAC+0

If TS $\geq$ +0:

MPAC+0 = MPAC+0 + K_{20b} (overflows propagate to bit 15)

If TS $\leq$ -0:

MPAC+0 = MPAC+0 + K_{20b}

If MPAC+0 $>$ 0: (includes case where TS = -0)

MPAC+0 = MPAC+0 + K_{ngl}

If MPAC+0 $\leq$ -0:

Set bit 15 of MPAC+0 = 0

SFTEMP1 = K_{dgtb_2}

$MPAC_{dp} = K_{dgtb_2}$ MPAC+0 + SFTEMP1

Proceed to "DSPDCEND"

DP3OUTSF (left shift of 7)

Perform "DPOUT"

$MPAC_{tp} = 2^7$ $MPAC_{tp}$ (left shift 7, OVFINP set if overflow)

Proceed to "DSPDCEND"

HMSOUT

If MIXBR = 1:

TS = NOUNADD

If MIXBR = 2:

EBANK = bits 11-9 of $IDADITEM_{DECOUNT+1}$ (i.e. IDAD2TEM for DECOUNT = 1)

TS = 1400_8 + bits 8-1 of $IDADITEM_{DECOUNT+1}$

$MPAC_{dp} = E_{TS_{dp}}$, with sign agreement forced

Perform "SEPSECNR"

$MPAC_{dp} = K_{scn2}\ MPAC_{dp}$

DSPCOUNT = 4

Perform "DSPDECWD" (display seconds in R3)

Perform "SEPMIN"

TS_2 = MPAC+0 (whole hours)

If MPAC+1 = -0:

$MPAC_{dp} = -0$

If MPAC+1 $\neq$ -0:

$MPAC_{dp} = K_{mncn2}$ MPAC+1

DSPCOUNT = 9

Perform "DSPDECWD" (display minutes in R2)

If $TS_2 = -0$:

$MPAC_{dp} = -0$

If $TS_2 \neq -0$:

$MPAC_{dp} = K_{hrcn1}\ TS_2$

DSPCOUNT = 14

Perform "DSPDECWD" (display hours in R1)

Proceed to address specified by ENTRET

M/SOUT

If MIXBR = 1:

TS = NOUNADD

If MIXBR = 2:

EBANK = bits 11-9 of $IDAD\underline{I}TEM_{DECOUNT+1}$ (i.e. IDAD2TEM for DECOUNT = 1)

$TS = 1400_8$ + bits 8-1 of $IDAD\underline{I}TEM_{DECOUNT+1}$

$MPAC_{dp} = E_{TS_{dp}}$, with sign agreement forced

$TS = MPAC_{dp}$

If $|TS| \geq K_{mscn12}$:

$MPAC_{dp} = K_{mscn3}$ sgn MPAC+0

Perform "SEPSECNR"

If $|TS| < K_{mscn12}$:

$MPAC_{dp} = MPAC_{dp} + K_{rndcn}$ sgn MPAC+1

Perform "SEPSECNR"

$MPAC_{dp} = K_{hisec}$ $MPAC_{dp}$

DSPCOUNT = DSPCOUNT - 3

Perform "DSPDC2NR" (seconds in digits 4 and 5)

CODE = 0

COUNT = $K_{rd_{DECOUNT}}$ - 2

Perform "DSPIN" (blank digit 3)

Perform "SEPMIN"

$MPAC_{dp} = K_{himin}$ MPAC+1

DSPCOUNT = $K_{rd_{DECOUNT}}$

Perform "DSPDC2NR" (minutes in digits 1 and 2)

Proceed to second line of "DSPDCEND"

SEPSECNR

$MPAC_{tp} = K_{scn1}$ $MPAC_{dp}$

HITEMOUT = $MPAC_{dp}$

$MPAC_{tp} = 2^2$ $MPAC_{tp}$ (left shift 2, OVFINP set if overflow)

$MPAC_{dp} = 2^{14}$ $MPAC_{tp}$ (left shift 14, overflow information lost, leaves seconds in $MPAC_{dp}$)

Return

SEPMIN

$MPAC_{dp}$ = HITEMOUT

Set bits 12-1 of MPAC+1 = sign bit (i.e. zero magnitude, masking out former information on seconds)

$MPAC_{tp} = K_{mncnl}\ MPAC_{dp}$

Return

AROUT1SF (left shift of 14, single precision operand)

If MPAC+0 = -0:

$MPAC_{dp}$ = -0

Proceed to "DSPDCEND"

$MPAC_{tp}$ = SFTEMP1 MPAC+0

$MPAC_{dp} = 2^{14}\ MPAC_{tp}$ (left shift of 14, overflow information lost)

Proceed to "DSPDCEND"

2INTOUT

Perform "5BLANK" (blanks all digits)

Perform "+0N"

TS = MPAC+0

Perform "DSPDECVN" (puts first cell in digits 1 and 2)

DSPCOUNT = $K_{rd_{DECOUNT}}$ - 3

If MIXBR = 1:

TS = NOUNADD

If MIXBR = 2:

EBANK = bits 11-9 of $IDADITEM_{DECOUNT+1}$

TS = 1400_8 + bits 8-1 of $IDADITEM_{DECOUNT+1}$

$MPAC_{dp} = E_{TS_{dp}}$

TS = MPAC+1

Perform "DSPDECVN" (puts second cell in digits 4 and 5)

Proceed to second line of "DSPDCEND"

DPFRACOT

If MIXBR = 1:

TS = NOUNADD

If MIXBR = 2:

EBANK = bits 11-9 of $IDADITEM_{DECOUNT+1}$

TS = 1400_8 + bits 8-1 of $IDADITEM_{DECOUNT+1}$

$MPAC_{dp} = E_{TS_{dp}}$

Proceed to "DSPDCEND"

PUTDCSF2

Proceed to address given by the following table:

TS_1	Address
0	"ALMCYCLE" (noun octal only)
1	"BINROUND"
2	"DEGINSF"
3	"ARTHINSF"
4	"DPINSF"
5	"DPINSF2"
6	"OPTDEGIN"
7	"DPINSF"
8	"HMSIN"
9	"DSPALARM" (can't load MMBSS)
10	"DPINSF4"
11	"ARTIN1SF"
12	"DSPALARM" (can't load XXBYY)
13	"DPFRACIN"
14	"DSPALARM" (no need to load NO4)

DEGINSF

$MPAC_{tp} = K_{dgcnl}\ MPAC_{dp}$

$MPAC_{tp} = MPAC_{tp} + K_{btll}$ sgn MPAC+1 If overflow, set OVFINP

$MPAC_{tp} = 2^1\ MPAC_{tp}$ (left shift 1) If overflow, set OVFINP

Proceed to "DEGINSF2"

DEGINSF2

$MPAC_{tp} = 2^1\ MPAC_{tp}$ (left shift 1) If overflow, set OVFINP

If OVFINP set (i.e. non-zero), proceed to "ALMCYCLE"

$MPAC_{tp} = 2^1\ MPAC_{tp}$ (left shift 1)

If overflow takes place:

OVFINP = 1 sgn MPAC+0

If MPAC+0 = -0:

MPAC+0 = +0

If MPAC+0 < 0:

MPAC+0 = MPAC+0 + 00001_8 (for twos complement)

If OVFINP = 0:

TS = MPAC+0

Proceed to address specified by DECRET

If OVFINP > 0:

Set bit 15 of MPAC+0 = 1

TS = MPAC+0

Proceed to address specified by DECRET

If MPAC+0 = +0: (OVFINP is <0 if come here)

MPAC+0 = 40000_8

TS = MPAC+0

Proceed to address specified by DECRET

Set bit 15 of MPAC+0 = 0

TS = MPAC+0

Proceed to address specified by DECRET

ARTHINSF (left shift of 14)

$MPAC_{tp}$ = SFTEMP1 $MPAC_{dp}$

If MPAC+0 $\neq$ 0, proceed to "ALMCYCLE"

$MPAC_{dp} = 2^{14}$ $MPAC_{tp}$ (left shift of 14)

Proceed to "BINROUND"

DPINSF2 (left shift of 7)

$MPAC_{tp}$ = SFTEMP1 $MPAC_{dp}$

$MPAC_{tp} = 2^{7}$ $MPAC_{tp}$ (left shift of 7, OVFINP set if overflow)

Proceed to second line of "DPINSF"

OPTDEGIN

If MPAC+0 $\leq$ -0, proceed to "ALMCYCLE"

$MPAC+0 = MPAC+0 + K_{ngp2}$

$MPAC_{tp} = K_{dgcn2}\ MPAC_{dp}$

$MPAC_{tp} = MPAC_{tp} + K_{btl2}$ If overflow, set OVFINP

Proceed to "DEGINSF2"

HMSIN

If bits 5-3 of DECBRNCH $\neq 111_2$: (3 decimal inputs not received)

VERBSAVE = -25 (force verb 25)

Proceed to "ALMCYCLE"

$MPAC_{tp} = K_{whecn}\ MPAC_{dp}$ ($MPAC_{dp}$ contains hours)

$MPAC_{tp} = MPAC_{tp} + MPAC+2$

If MPAC+0 $\neq$ 0, proceed to "ALMCYCLE"

$MPAC_{tp} = K_{hrcon}\ MPAC+1$

If MPAC+0 $\neq$ 0, proceed to "ALMCYCLE" (input exceeded 745 hours)

$TS = 2^{14}\ MPAC_{tp}$ (left shift 14)

$MPAC_{dp}$ = (YREG, YREGLP) ($MPAC_{dp}$ loaded with minutes)

$MPAC_{tp} = K_{whecon}\ MPAC_{dp}$

$MPAC_{tp} = MPAC_{tp} + MPAC+2$

If MPAC+0 $\neq$ 0, proceed to "ALMCYCLE"

If $|MPAC+1| > K_{59min}$, proceed to "ALMCYCLE"

$TS = TS + K_{mincon}\ MPAC+1$

If $|TS| \geq 2^{28}$ centi-seconds, proceed to "ALMCYCLE"

$MPAC_{dp}$ = (ZREG, ZREGLP) ($MPAC_{dp}$ loaded with centi-seconds)

$MPAC_{tp} = K_{whecon}\ MPAC_{dp}$

$MPAC_{tp} = MPAC_{tp} + MPAC+2$

If MPAC+0 $\neq$ 0, proceed to "ALMCYCLE"

If $|MPAC+1| > K_{5999sc}$, proceed to "ALMCYCLE"

$TS = TS + MPAC_{dp}$

If $|TS| \geq 2^{28}$ centi-seconds, proceed to "ALMCYCLE"

Force sign agreement of TS (maximum time is 2^{28} - 1 centi-seconds, i.e. 745 h, 39 m, 14.55 sec; or 31 d, 1 hr, 39 m, 14.55 s)

$E_{NOUNADD} = TS_{dp}$

Proceed to "LOADLV"

DPINSF4 (left shift 3)

$MPAC_{tp} = SFTEMP1\ MPAC_{dp}$

$MPAC_{dp} = 2^3\ MPAC_{tp}$ (left shift of 3, OVFINP set if overflow)

Proceed to second line of "DPINSF"

ARTIN1SF (no shift)

$MPAC_{tp} = SFTEMP1\ MPAC_{dp}$

Proceed to "BINROUND"

DPFRACIN

Proceed to 4th line of "DPINSF"

ATTERROR

MPAC+0 = - MPAC+0

Proceed to "ARTOUTSF"

Quantities in Computations

See also list of major variables and list of routines

CODE, COUNT: See Data Input/Output.

DECBRNCH, DECOUNT, DECRET: See Data Input/Output.

DSPCOUNT: See Data Input/Output.

EBANK, ENTRET: See Data Input/Output.

HITEMOUT: Value of number of minutes in time quantity (computed in "SEPSECNR" for use in "SEPMIN"), scale factor B16, units minutes. Least significant 12 bits of least significant half contain fractional portion of minutes, and therefore must be blanked in "SEPMIN" for proper computation of the number of integral minutes in the time argument. Least significant half "LOTEMOUT".

IDADITEM (I = 1,2,3): Temporary storage for information from mixed-noun tables (see below) for first (R1), second (R2), and third (R3) components of noun. The R1 component, of course, is displayed in register R1 on the DSKY, etc.

K_{20b}: Single precision constant, program notation "20BIAS", scale factor B-3, units revolutions. Octal value is 16040_8, corresponding to about 19.7754°.

K_{59min}: Single precision constant, program notation "59MIN", scale factor B14, units minutes. Value is 00073_8, corresponding to 59 minutes.

K_{5999sc}: Single precision constant, program notation "59.99SEC", scale factor B14, units centi-seconds. Value is 13557_8, corresponding to 5999 centi-seconds (or 59.99 seconds).

K_{btl1}: Single precision constant, program notation "BITl1". As used, value corresponds to $(\frac{1}{2} \times 2^{-14})$ of full scale, or one-half the least increment on the single precision result for "DEGINSF".

K_{btl2}: Single precision constant, program notation "BITl2". As used, value corresponds to $(\frac{1}{2} \times 2^{-14})$ of full scale, or one-half the least increment on the single precision result for "OPTDEGIN".

K_{dgcnl}: Constant, program notation "DEGCONl", scale factor B3, value $5.555555555 \times 2^{-3}$. Value corresponds to $(1000/180) \times 2^{-3}$, where first term converts from XXX.XX° to B-1 revolutions, and second is the constant's scale factor.

K_{dgcn2}: Constant, program notation "DEGCON2", scale factor B2, value $2.222222222 \times 2^{-2}$. Value corresponds to $(100/45) \times 2^{-2}$, where first term converts from XX.XXX° to B-3 revolutions, and second is the constant's scale factor.

K_{dgtb_0}: Constant, program notation "DEGTAB", scale factor B0, octal value 05605_8 03656_8, corresponding to decimal 0.18. Value equivalent to 180/1000, to convert from B-1 revolutions to XXX.XX°.

K_{dgtb_2}: Constant, program notation "DEGTAB +2", scale factor B0, octal value 16314_8 31463_8, corresponding to decimal 0.45. Value equivalent to 45/100, to convert from B-3 revolutions to XX.XXX°.

K_{himin}: Single precision constant, program notation "HIMINCON", scale factor B0, value 23346_8. Value corresponds to $(0.6 + 2^{-7})$.

K_{hisec}: Single precision constant, program notation "HISECON", scale factor B0, value 23147_8. Value corresponds to $(0.6 + 2^{-14})$.

K_{hrcn1}: Constant, program notation "HRCON1", scale factor B-14, value 0.16384. Value corresponds to $10^{-5} \times 2^{14}$.

K_{hrcon}: Constant, program notation "HRCON", scale factor B28, units centi-seconds. Value is 00025_8 37100_8, corresponding to 360,000 (or 3600 seconds).

K_{idtb_i}: Table of constants for mixed noun information, program notation "IDADDTAB", giving address and scaling routine information. See information below.

K_{mincon}:Single precision constant, program notation "MINCON", scale factor B14, units centi-seconds. Value is 13560_8, corresponding to decimal 6000 (or 60 seconds).

K_{mncn1}: Constant, program notation "MINCON1", scale factor B-2, value 02104_8 10422_8, corresponding to $(1/60 \times 2^2 + 2^{-28})$, to give hours scaled B14 in MPAC+0 when return from "SEPMIN" (and fraction of an hour, B0, in MPAC+1).

K_{mncn2}: Constant, program notation "MINCON2", scale factor B0, value 00011_8 32445_8. Value corresponds to 60×10^{-5}.

K_{mscn3}: Constant, program notation "M/SCON3", scale factor B28, units centi-seconds. Value is 00025_8 37016_8, corresponding to 359,950 centi-seconds (or 59 minutes 59.5 seconds).

K_{mscn12}: Notation assigned to effect in program of constants "M/SCON1" and "M/SCON2", both single precision with octal values 77753_8 and 41126_8 respectively. The constants information, for program convenience, is stored in negative form (with magnitude of each decremented by one least increment for convenience in forming the absolute value). The net effective value in the program of the combined constant (scale factor B28, units centi-seconds) is 00025_8 36652_8, corresponding to 359,850 centi-seconds (59 minutes 58.5 seconds). A value of time of this value or more would be displayed in "M/SOUT" as 59 59, with appropriate sign.

K_{mxcn}: Single precision constant, program notation "MIXCON", scale factor B14, value 00050_8, corresponding to a noun of 40. Nouns of this value or above are considered "mixed nouns".

K_{ng1}: Single precision constant, program notation "NEG1", scale factor B-3, units revolutions, used to convert from twos complement to ones complement information. Value is -2^{-14} (minus one least increment).

K_{ngp2}: Single precision constant, program notation "NEG.2", value 71527_8. Octal value equivalent to -6250_8: since the constant is added to information scaled XX.XXX°, value corresponds to 100 x (- 0.197754), or - 19.7754° (cf. K_{20b}).

K_{ntb_i}: Table of constants, program notation "NNADTAB", for loading NNADTEM. See information below.

K_{nty_i}: Table of constants, program notation "NNTYPTAB", for loading NNTYPTEM. See information below.

K_{rd_I}: See Data Input/Output.

K_{rndcn}: Constant, program notation "RNDCON -1", scale factor B28, units centi-seconds. Value is 00000_8 00062_8, corresponding to 50 centi-seconds (0.5 second).

K_{rtmtb_i}: Table of constants used to specify scaling routines for mixed nouns, program notation "RUTMXTAB". See information below (is used to load RUTMXTEM). First table cell is for noun 40.

K_{scn1}: Constant, program notation "SECON1", scale factor B-12, value $1.666666666E\text{-}4 \times 2^{12}$, corresponding to $(1/6000) \times 2^{12}$ (to convert centi-seconds scaled B28 to minutes scaled B16, cf. HITEMOUT).

K_{scn2}: Constant, program notation "SECON2", scale factor B0, value 01727_8 01217_8. Value corresponds to 60×10^{-3}.

K_{sfin_0}: First of a set of input constants selected in "GTSFIN", located in consecutive cells starting at "SFINTAB". Value is 00006_8 03240_8, scale factor B28. Value corresponds to $10^5 \times 2^{-28}$, to convert from the fraction produced by "NUM" to an integer with scale factor B28.

K_{sfin_1}: Constant, value 0, not assigned.

K_{sfin_2}: Constant, value 0, used to set initial condition for SFTEMP1 for use in "DEGINSF" and "OPDEGIN". The scaling routine itself selects the appropriate scaling constants.

K_{sfin_3}: Constant, value 10707_8 03435_8, scale factor B0. Value corresponds to one least increment more than (100 x 1/360), to convert from an input of XX.XXX° to B0 revolutions. Since there are 2^{21} gyro pulses per revolution, the constant could also be considered to convert to gyro pulses, scale factor B21.

K_{sfin_4}: Constant, scale factor B3, value 13070_8 34345_8. Value corresponds to one least increment more than $(1000 \times 1/360) \times 2^{-3}$, to convert from XXX.XX° to B3 revolutions (hence use Routine #10, "DPINSF4", to shift result left 3 for a scaling of B0 revolutions).

K_{sfin_5}: Constant, value 00005_8 21616_8, scale factor B14. Value corresponds to $(1000 \times 1/180) \times 2^{-14}$, to convert from XXX.XX° to B-1 revolutions (after performing a left shift of 14).

K_{sfin_6}: Constant, value 26113_8 31713_8, scale factor B0. Value corresponds to $10^5 \times 0.45359237 \times 2^{-16}$, to convert between XXXXX. pounds and B16 kilograms (single precision).

K_{sfin_7}: Constant, value 00070_8 20460_8, scale factor B0. Value corresponds to $10^3 \times 1852 \times 2^{-29}$, to convert between XXX.XX nmi and meters scaled B29 (there are 1852 meters in a nautical mile).

K_{sfin_8}: Constant, value 01065_8 05740_8, scale factor B0. Value corresponds to $10^4 \times 1852 \times 2^{-29}$, to convert between XXXX.X nmi and meters scaled B29 (there are 1852 meters in a nautical mile). See K_{sfot_8}.

K_{sfin_9}: Constant, value 11414_8 31463_8, scale factor B3. Value corresponds to $10^5 \times 10^{-2} \times 0.3048 \times 2^{-7} \times 2^{-3}$, for the net value to convert from XXXXX. feet/second to B10 meters/centi-second (hence use Routine #10, "DPINSF4", to shift result left 3 for scaling of B7 meters/centi-second).

$K_{sfin_{10}}$: Constant, value 07475_8 16051_8, scale factor B0. Value corresponds to $10^4 \times 10^{-2} \times 0.3048 \times 2^{-7}$, in order to convert between XXXX.X fps and meters/centi-second with scale factor B7.

$K_{sfin_{11}}$: Constant, value 00001_8 03434_8, scale factor B14. Value corresponds to $10^2 \times (1/90) \times 2^{-14}$, to convert from XX.XXX° to B-2 revolutions (after performing a left shift of 14 via Routine #3, "ARTHINSF").

$K_{sfin_{12}}$: Constant, value 00002_8 22245_8, scale factor B14. Value corresponds to $10^3 \times (3600/85.41) \times 2^{-14} \times 2^{-14}$ minus one least increment, to convert between XXX.XX° and B14 "CDU actuator pulses" (after performing a left shift of 14 via Routine #3, "ARTHINSF"). There are 85.41 arc seconds per CDU actuator pulse.

$K_{sfin_{13}}$: Constant, value 00014_8 35607_8, scale factor B14. Value corresponds to $10^7 \times 1.355817948 \times 2^{-20} \times 2^{-14}$, to convert between XXXXXbb. slug-feet^2 and B20 kilogram-meters^2 (after performing a left shift of 14 via Routine #3, "ARTHINSF"). The "bb" means that the input is in units of 100 slug-feet^2. Same constant could also be used to convert between XXXXXbb. foot-pounds and B20 newton-meters. Constant is not used (since functions are computed in "FIXCW" as determined from inputs of vehicle mass).

$K_{sfin_{14}}$: Constant, value 07606_8 06300_8, scale factor B3. Value corresponds to one least increment more than $10^5 \times (1/25766.1973) \times 2^{-1} \times 2^{-3}$, to convert between XXXXX. fps and B1 VSAT units (used in entry, where 1 VSAT = 25766.1973 fps), after performing a left shift of 3 via Routine #10, "DPINSF4".

$K_{sfin_{15}}$: Constant, value 16631_8 11307_8, scale factor B0. Value corresponds to $10^4 \times (1/21622.4965)$, to convert between XXXX.X nmi and B0 revolutions. The denominator conversion value corresponds to $2\pi \times 6373338 / 1852$, i.e. an earth radius equal to the value of K_{rpad} (see Burn Control) used to derive circumference.

$K_{sfin_{16}}$: Constant, value 12000_8 00000_8, scale factor B7. Value corresponds to $10^3 \times (1/25) \times 2^{-7}$, to convert between XXX.XX g's and B0 "G-units" (25 g's, as used in Entry Computations), after performing a left shift of 7 via Routine #5, "DPINSF2".

$K_{sfin_{17}}$: Constant, value 27176_8 14235_8, scale factor B0. Value corresponds to $10^4 \times (1/3441.3272) \times 2^{-2}$, to convert between XXXX.X nmi and radians scaled B2 (using the pad radius given for $K_{sfin_{15}}$).

$K_{sfin_{18}}$: Constant, value 30480×2^{-19}, scale factor B0. Value corresponds to $10^5 \times 0.3048 \times 2^{-19}$, to convert between XXXXX. feet and meters scaled B19 (there are 0.3048 meters in one foot).

$K_{sfin_{19}}$: Constant, value 30.48×2^{-7}, scale factor B7. Value corresponds to $10^4 \times 0.3048 \times 10^{-2} \times 2^0 \times 2^{-7}$, to convert between XXXX.X fps and B0 meters/centi-second (first term is for XXXX.X fps, second converts to meters, third converts to centi-seconds, fourth is for scaling of final answer, and fifth is scale factor of constant), after a left shift of 7 via Routine #5, "DPINSF2".

$K_{sfin_{20}}$: Constant, value $1.42213543 \times 2^{-14}$, scale factor B14. Value corresponds to $10 \times (1/360) \times (1/10) \times 2^9 \times 2^{-14}$, to convert between X.XXXX degrees/sec and revolutions/deci-second scaled B-9 (first term is for X.XXXX degrees/second, second converts to revolutions, third converts to deci-seconds, fourth is for scaling of final answer, and fifth is scale factor of constant), after a left shift of 14 via Routine #3, "ARTHINSF". Value should be 1.422222.

$K_{sfin_{21}}$: Constant, value $4.62962963 \times 2^{-14}$, scale factor B14. Value corresponds to $10^5 \times (1/360) \times (1/60) \times 2^0 \times 2^{-14}$, to convert between XXXXX. arc minutes and revolutions scaled B0 (first term is for XXXXX. arc minutes, second converts to revolutions from degrees, third converts to degrees from minutes, fourth is for scaling of final answer, and fifth is scale factor of constant), after a left shift of 14 via Routine #3, "ARTHINSF".

$K_{sfin_{22}}$: Constant, value $0.3048E2 \times 2^{-6}$, scale factor B0. Value corresponds to $10^4 \times 0.3048 \times 10^{-2} \times 2^{-6}$, to convert between XXXX.X fps and meters/centi-second scaled B6 (first term is for XXXX.X fps, second converts to meters, third to centi-seconds, and fourth is for scaling of final answer).

$K_{sfin_{23}}$: Constant, value $1852E3 \times 2^{-23}$, scale factor B3. Value corresponds to $10^3 \times 1852 \times 2^{-20} \times 2^{-3}$, to convert between XXX.XX nmi and meters scaled B20 (first term is for XXX.XX nmi, second converts from nautical miles to meters, third is for scaling of final answer, and fourth is scaling for constant), after a left shift of 3 via Routine #10, "DPINSF4".

$K_{sfin_{24}}$: Constant, value 0.022222222, scale factor B0. Value corresponds to $10 \times (1/360) \times (1/10) \times 2^3$, to convert between X.XXXX degrees/second and revolutions/deci-second scaled B-3 (first term is for X.XXXX degrees/second, second converts to revolutions, third converts to deci-seconds, and fourth is for scaling of final answer).

K_{sfot_0} : First of a set of output constants selected in "GTSFOUT", located in consecutive cells starting at "SFOUTAB". Value is 05174_8 13261_8, scale factor B0, with value corresponding to $10^{-5} \times 2^{14}$, to convert from an integer with scale factor B14 to XXXXX.

K_{sfot_1} : Constant, value 0, not assigned.

K_{sfot_2}: Constant, value 0, used to load SFTEMP1 with proper initial value for use in "DEGOUTSF" and "OPDEGOUT".

K_{sfot_3}: Constant, value 00714_8 31463_8, scale factor B7. Value corresponds to $(360) \times 10^{-2} \times 2^{-7}$, to convert from B0 revolutions to XX.XXX°. Since there are 2^{21} gyro pulses per revolution, the constant could also be considered to convert from gyro pulses scaled B21.

K_{sfot_4}: Constant, value 13412_8 07534_8, scale factor B0. Value corresponds to $(360) \times 10^{-3}$, to convert from B0 revolutions to XXX.XX°.

K_{sfot_5}: Constant, value 05605_8 03656_8, scale factor B0. Value corresponds to $(180) \times 10^{-3}$, to convert from B-1 revolutions to XXX.XX°.

K_{sfot_6}: Constant, value 00001_8 16170_8, scale factor B14. Value corresponds to $2^{16} \times (1/0.45359237) \times 10^{-5} \times 2^{-14}$, to convert between B16 kilograms and XXXXX. pounds.

K_{sfot_7}: Constant, value 00441_8 34306_8, scale factor B14. Value corresponds to $2^{29} \times (1/1852) \times 10^{-3} \times 2^{-14}$, to convert between B29 meters and XXX.XX nmi.

K_{sfot_8}: Constant, value 07176_8 21603_8, scale factor B7. Value corresponds to $(2^{29}) \times (1/1852) \times 10^{-4} \times 2^{-7}$, to convert between B29 meters and XXXX.X nmi (there are 1852 meters in a nautical mile). For N73, converts from units of meters/100 (computed by P21) to XXXXXb. nmi.

K_{sfot_9}: Constant, value 15340_8 15340_8, scale factor B0 (equality of two halves of constant is correct). Value corresponds to $(100 \times 2^7/0.3048) \times 10^{-5}$, to convert from the navigation scaling of velocity (B7 meters/centi-second) to XXXXX. fps.

$K_{sfot_{10}}$: Constant, value 01031_8 21032_8, scale factor B7. Value corresponds to $(100 \times 2^7/0.3048) \times 10^{-4} \times 2^{-7}$ plus one least increment, to convert from the navigation scaling of velocity (B7 meters/centi-second) to XXXX.X fps.

$K_{sfot_{11}}$: Constant, value 34631_8 23146_8, scale factor B0. Value corresponds to $(90) \times 10^{-2}$, to convert from B-2 revolutions to XX.XXX°.

$K_{sfot_{12}}$: Constant, value 14340_8 24145_8, scale factor B0. Value corresponds to $(85.41/3600) \times 2^{14} \times 10^{-3}$, to convert between B14 "CDU actuator pulses" and XXX.XX°. There are 85.41 arc seconds per CDU actuator pulse.

$K_{sfot_{13}}$: Constant, value 02363_8 03721_8, scale factor B0. Value corresponds to $2^{20} \times (1/1.355817948) \times 10^{-7}$, to convert between B20 kilogram-meters^2 and XXXXXbb. slug-feet^2 (the "bb" means that output is in units of 100 slug-feet^2). Same constant could also be used to convert between B20 newton-meters and XXXXXbb. foot-pounds. Constant is not used (see $K_{sfin_{13}}$).

$K_{sfot_{14}}$: Constant, value 20373_8 02122_8, scale factor B0. Value corresponds to $2^1 \times 25766.1973 \times 10^{-5}$, to convert between B1 VSAT units and XXXXX. fps (one VSAT is 25766.1973 fps, used in entry).

$K_{sfot_{15}}$: Constant, value 00424_8 30446_8, scale factor B7. Value corresponds to $21622.4965 \times 10^{-4} \times 2^{-7}$, to convert between B0 revolutions and XXXX.X nmi (see $K_{sfin_{15}}$).

$K_{sfot_{16}}$: Constant, value 00631_8 23146_8, scale factor B0. Value corresponds to 25×10^{-3}, to convert between B0 "G-units" (25 g's, as used in entry computations) and XXX.XX g's.

$K_{sfot_{17}}$: Constant, value 00260_8 06213_8, scale factor B7. Value corresponds to $3441.3272 \times 2^{28} \times 10^{-4} \times 2^{-7}$, to convert between B2 radians and XXXX.X nmi (using pad radius, see $K_{sfin_{17}}$).

$K_{sfot_{18}}$: Constant, value 17.2010499×2^{-7}, scale factor B7. Value corresponds to $2^{19} \times (1/0.3048) \times 10^{-5} \times 2^{-7}$, to convert between B19 meters and XXXXX. feet (with a left shift of 7 places due to scaling of the constant).

$K_{sfot_{19}}$: Constant, value 0.032808399, scale factor B0. Value corresponds to $2^0 \times 10^2 \times (1/0.3048) \times 10^{-4}$, to convert between B0 meters/centi-second and XXXX.X fps: first term is for original scaling, second converts to seconds, third converts to feet, and fourth is display scale.

$K_{sfot_{20}}$: Constant, value 0.703167914, scale factor B0. Value corresponds to $\sim 2^{-9} \times 360 \times 10 \times 10^{-1}$, to convert between revolutions/deci-second and X.XXXX degrees/second: first term is for original B-9 scaling of rev/deci-second, second converts to degrees, third converts to seconds, and fourth is display scale. Value of constant should be 0.703125: actual value is reciprocal of $K_{sfin_{20}}$.

$K_{sfot_{21}}$: Constant, value 0.216, scale factor B0. Value corresponds to $360 \times 60 \times 10^{-5}$, to convert between B0 revolutions and XXXXX. arc minutes (first term converts to degrees, second to minutes, and third is display scale).

$K_{sfot_{22}}$: Constant, value $3.280839896E-2 \times 2^{-1}$, scale factor B7. Value corresponds to $2^6 \times (1/0.3048) \times 10^2 \times 10^{-4} \times 2^{-7}$, to convert between B6 meters/centi-second and XXXX.X fps: first term is for original scaling, second converts to meters, third converts to seconds, fourth is display scale, and fifth is scaling of constant.

$K_{sfot_{23}}$: Constant, value $0.5399568E-6 \times 2^{20}$, scale factor B0. Value corresponds to $2^{20} \times (1/1852) \times 10^{-3}$, to convert between B20 meters and XXX.XX nmi.

$K_{sfot_{24}}$: Constant, value 45×2^{-7}, scale factor B7. Value corresponds to $2^{-3} \times 360 \times 10 \times 10^{-1} \times 2^{-7}$, to convert between revolutions/deci-second scaled B-3 and X.XXXX degrees/second: first term is for original scaling, second converts to degrees, third converts to seconds, fourth is display scale, and fifth is scaling of constant.

K_{whecon}: Constant, program notation "WHOLECON", value 00006_8, 03240_8, scale factor B28. Value corresponds to $10^5 \times 2^{-28}$ $_8$(to "compensate" for the scaling done in "NUM").

MIXBR: Single precision cell, scale factor B14, set in "LODNNTAB" to 1 for a "normal" noun and to 2 for a "mixed" noun.

NNADTEM: Temporary storage cell, single precision, loaded in "LODNNTAB" with the value of K_{ntb_I} corresponding to NOUNREG contents. See information below.

NNTYPTEM: Temporary storage cell, single precision, loaded in "LODNNTAB" with the value of K_{nty_I} corresponding to NOUNREG contents. See information below.

NOUNADD, NOUNREG: See Data Input/Output.

OVFINP: See Data Input/Output.

RUTMXTEM: Temporary storage cell, single precision, loaded in "LODNNTAB" with value of K_{rtmtb_I} for "mixed nouns" only. See information below.

SFTEMP1: Temporary storage cell used to contain the appropriate value of K_{sfin_I} or K_{sfot_I}, used for angle bias information (if any) in "DEGOUTSF" and "OPDEGOUT".

VERBSAVE: See Data Input/Output.

YREG, YREGLP: See Data Input/Output (YREGLP described for LPREG).

ZREG, ZREGLP: See Data Input/Output (ZREGLP described for LPREG).

Noun Table Interpretation

"Normal" Nouns
MIXBR = 1

K_{nty} bits 15-11 contain the "component code number", interpreted as follows:

Bit 15 is 1 if no loading of information by use of the noun with verbs 24 or 25 is allowed.

Bit 14 is 1 if only decimal input/output by use of the noun is allowed (i.e. no octal verbs may be used).

Bit 13 is not assigned.

Bits 12-11 give information on the number of components:

00_2 for one component

01_2 for two components

10_2 for three components

K_{nty} bits 10-6 contain the "scale factor routine code number", used in "DECDSP3" and "PUTDCSF2" to transfer to the proper scaling routine.

K_{nty} bits 5-1 contain the "scale factor constant code number", used in "GTSFOUT" and "GTSFIN" to select the proper scaling constant.

K_{ntb} gives the machine address information with the following interpretations:

+0 if noun not assigned.

-0 if previous address to be incremented by +1.

-1 if a channel is to be loaded or read (channel number is supplied as another input).

<-1 (usually 40000_8) if an octal erasable memory address is to be supplied as another input.

>+0 if the octal erasable memory address is given (for the first component).

"Mixed" Nouns
MIXBR = 2

K_{nty} bits 5-1, 10-6, and 15-11 contain the "scale factor constant code number", used in "GTSFOUT" and "GTSFIN", for component #1, #2, and #3 respectively.

K_{ntb} bits 15-11 contain the "component code number", with the same bit assignments as for bits 15-11 of K_{nty} for normal nouns.

K_{ntb} bits 10-1 contain the relative address (with respect to the start of the K_{idtb} table, program notation "IDADDTAB") of the information in that table for the first component of the noun. Is 3(NOUNREG -40).

K_{idtb} gives the octal erasable memory address for the individual component (if the noun is three components, three consecutive cells are used).

K_{rtmtb} bits 5-1, 10-6, and 15-11 contain the "scale factor routine code number", used in "DECDSP3" and "PUTDCSF2", for component #1, #2, and #3 respectively.

Notes for Noun Table Information

1. A decimal display of "hr,mn,sc" appears as: 00XXX. hr
   000XX. mn
   0XX.XX sc

2. A decimal single-register display of "mn, ,sc" appears with minutes in the first two digits of the register and seconds in the last two digits. The middle (third) digit is blank. Maximum magnitude is 59 59.

3. Vector-type quantities are displayed with the first (e.g. X) component in R1, the second component in R2, and the third in R3.

4. Single component nouns appear in R1 only.

5. If an output quantity is larger than the capacity of the display register scaling, it generally will be displayed modulo that capacity: this assumes, of course, that the basic cell itself as stored in memory has not overflowed. If velocity increment of 12000 fps experienced on an XXXX.X fps scale for display, it would be expected to show 2000(.)0. This arises from the fact that the K_{sfot} constants convert the information into the range 0 - 0.99999, with subsequent display output generated by multiplying by 10 and using the integral part of the result (in "DSPDCWD1"). Multiplication by the K_{sfot} constant takes place before any scaling shifts.

Noun Table Information

Noun	Cell	No Load	Dec. Only	Number	Routine	Constant	Decimal Display	Quantity
00	+0			0	0	0	---	Not assigned.
01	40000_8			3	1	0	.XXXXX	Address supplied.
02	40000_8			3	3	0	XXXXX.	Address supplied.
03	40000_8			3	2	2	XXX.XX^o	Address supplied.
04	AK	x	x	3	14	5	XXX.XX^o	FDAI error needles.
05	DSPTEM1 DSPTEM1+1			2	3	5	XXX.XX^o	Angle error. Sighting angle.
06	OPTION1 OPTION2			2	0	0	Octal	Option codes.
07	XREG YREG ZREG			3	0	0	Octal	Address of word. Bits to be changed. If $\leq$0, reset; if >0, set bits: see "ABCLOAD" (V25 only).
08	ALMCADR ALMCADR+1 ERCOUNT			3	0	0	Octal	Alarm data.
09	FAILREG			3	0	0	Octal	Alarm codes.
10	-1			1	0	0	Octal	Channel supplied.
11	T_{csi}		x	3	8	0	hr,mn,sc	NCC ignition time.
12	OPTIONX OPTIONX+1			2	0	0	Octal	Option codes.
13	T_{cdh}		x	3	8	0	hr,mn,sc	NSR ignition time.
14	TRKAZ TRKEL			2	3	21	XXXXX. min	Star tracker angles.
15	-0			1	0	0	Octal	Increment address.
16	DSPTEMX		x	3	8	0	hr,mn,sc	Time of event.
17	CPHIX			3	2	2	XXX.XX^o	V63 FDAI base angles (V60E loads with N20).
18	THETAD			3	2	2	XXX.XX^o	Ball angles for attitude maneuver.

Noun	Cell	No Load	Dec. Only	Number	Routine	Constant	Decimal Display	Quantity
19	TRKAZOCT TRKAZOCT+1			2	0	0	Octal	Special star tracker octal azimuth display.
20	CDU			3	2	2	XXX.XX^{o}	Present ICDU angles.
21	PIPA			3	3	0	XXXXX. cnt	Accelerometer cells.
22	THETAD			3	2	2	XXX.XX^{o}	Desired ICDU angles.
23	THETAD			3	2	2	XXX.XX^{o}	Docking angles.
24	DSPTEM2+1$_{dp}$		x	3	8	0	hr,mn,sc	Delta time for clock.
25	DSPTEM1			3	3	0	XXXXX.	Checklist information.
26	N26dPRI N26d2CAD N26d2CAD+1			3	0	0	Octal	Priority/Delay and 2CADR address for V30 and V31.
27	SMODE			1	3	0	XXXXX.	Computer self-test switch.
28	NC2TIG		x	3	8	0	hr,mn,sc	NC2 ignition time.
29	DSPTEM1		x	1	2	2	XXX.XX^{o}	$\underline{X}_{sm}$ launch azimuth.
30	DSPTEM1			3	3	0	XXXXX.	Target codes.
31	AGEOFW		x	3	8	0	hr,mn,sc	Time of r/v W matrix.
32	mTPER		x	3	8	0	hr,mn,sc	Time from pericenter.
33	T_{ig}		x	3	8	0	hr,mn,sc	Time of "ignition".
34	DSPTEM1$_{dp}$		x	3	8	0	hr,mn,sc	Time of event.
35	T_{togo}		x	3	8	0	hr,mn,sc	Time from event.
36	T_{now}		x	3	8	0	hr,mn,sc	AGC clock.
37	T_{tpi}		x	3	8	0	hr,mn,sc	TPI ignition time.
38	T_{et}		x	3	8	0	hr,mn,sc	State vector time.
39	LASTTIG		x	3	8	0	hr,mn,sc	Last maneuver (P38).
40		x	x	3				
	T_{togo}				9	0	mn, ,sc	Time from event.
	VGDISP				7	10	XXXX.X fps	$\lvert \underline{V}_g \rvert$ value.
	DVTOTAL				7	10	XXXX.X fps	Velocity accumulated.

Noun	Cell	No Load	Dec. Only	Number	Routine	Constant	Decimal Display	Quantity
41				2				
	$DSPTEM1_{sp}$				2	2	XXX.XX°	Target azimuth.
	$DSPTEM1+1_{sp}$				3	11	XX.XXX°	Target elevation.
42			x	3				
	HAPO				7	8	XXXX.X nmi	Apo. altitude.
	HPER				7	8	XXXX.X nmi	Per. altitude.
	VGDISP				7	10	XXXX.X fps	Required velocity change.
43			x	3				
	LAT				10	4	XXX.XX°	Latitude.
	LONG				10	4	XXX.XX°	Longitude.
	ALT				7	8	XXXX.X nmi	Altitude.
44		x	x	3				
	HAPOX				7	8	XXXX.X nmi	Apo. altitude.
	HPERX				7	8	XXXX.X nmi	Per. altitude.
	TFF				9	0	mn, ,sc	Time from interface altitude.
45		x	x	3				
	VHFCNT/ TRKMKCNT				12	0	VH, ,TR	VHF marks (D1 and D2). Optics marks (D4 & D5).
	T_{togo}				9	0	mn, ,sc	Time from event.
	pMGA				10	4	XXX.XX°	Predicted middle gimbal angle (if +).
46				2				
	DAPDATR1				0	0	Octal	DAP code word #1.
	DAPDATR2				0	0	Octal	DAP code word #2.

Noun	Cell	No Load	Dec. Only	Number	Routine	Constant	Decimal Display	Quantity
47			x	2				
	CSMMASS				11	6	XXXXX. lbs	CSM mass.
	LEMMASS				11	6	XXXXX. lbs	LM mass.
48			x	2				
	PACTOFF				3	12	XXX.XX°	Pitch SPS trim.
	YACTOFF				3	12	XXX.XX°	Yaw SPS trim.
49			x	3				
	N49DISP				4	7	XXX.XX nmi	Position change.
	N49DISP+2				7	10	XXXX.X fps	Velocity change.
	N49DISP+4				3	0	XXXXX.	Source code (1 for optics, 2 for VHF).
50		x	x	3				
	RSPmRREC				7	15	XXXX.X nmi	Splash error.
	HPERX				7	8	XXXX.X nmi	Per. Altitude.
	TFF				9	0	mn, ,sc	Time from interface altitude.
51	+0			0	0	0	---	Not assigned.
52				1				
	ACTCENT				10	4	XXX.XX°	Active vehicle(CSM) central angle.
53			x	3				
	RANGE				4	7	XXX.XX nmi	Range from target.
	RRATE				7	10	XXXX.X fps	Range rate.
	RTHETA				10	4	XXX.XX°	Angle Phi (R34).

Noun	Cell	No Load	Dec. Only	Number	Routine	Constant	Decimal Display	Quantity
54			x	3				
	RANGE				4	7	XXX.XX nmi	Range from target.
	RRATE				7	10	XXXX.X fps	Range rate.
	RTHETA				10	4	XXX.XX^{o}	Angle Theta (R31/P37).
55			x	2				
	NN1				3	0	XXXXX.	Count.
	ELEV				10	4	XXX.XX^{o}	Elevation angle.
56		x	x	3				
	ADOT				7	24	X.XXXX^{o}/sec	Vehicle rate (from DAP).
57			x	3				
	HALFREVS				3	0	XXXXX.	Half revolutions.
	DHNCC				7	8	XXXX.X nmi	NCC altitude difference.
	DELH1				7	8	XXXX.X nmi	NSR altitude difference.
58			x	3		(NOTE that is not "No Load")		
	DELVTPI				7	10	XXXX.X fps	Delta-V for burn.
	DELVTPF				7	10	XXXX.X fps	Final phase burn.
	T2TOT3				9	0	mn, ,sc	TPI slip.
59			x	3				
	DVLOS				7	10	XXXX.X fps	Delta-V in line-of-sight coordinates.
60			x	3				
	GMAX				3	0	XXX.XX g	Max. drag predicted.
	VPRED				10	9	XXXXX. fps	Predicted velocity 400 kft above Fischer.
	GAMMAEI				10	4	XXX.XX^{o}	Predicted flight path angle with VPRED.

Noun	Cell	No Load	Dec. Only	Num-ber	Rou-tine	Con-stant	Decimal Display	Quantity
61			x	3				
	LATSPL				10	4	XXX.XX^{o}	Target latitude.
	LNGSPL				10	4	XXX.XX^{o}	Target longitude.
	HEADSUP				3	0	XXXXX.	Heads up/down.
62			x	3				
	VMAGI				10	9	XXXXX. fps	Inertial velocity.
	HDOT				10	9	XXXXX. fps	Altitude rate.
	ALTI				7	8	XXXX.X nmi	Altitude above base radius magnitude.
63		x	x	3				
	RTGO				7	15	XXXX.X nmi	Range from EMS altitude to splash.
	VIO				10	9	XXXXX. fps	Predicted velocity at EMS altitude.
	TTE				9	0	mn, ,sc	Time from EMS altitude.
64			x	3				
	D				5	16	XXX.XX g	Drag acceleration.
	VMAGI				10	9	XXXXX. fps	Inertial velocity.
	RTGON67				7	15	XXXX.X nmi	Range to target.
65			x	3				
	T_{st}				8	0	hr,mn,sc	Sampled AGC clock.
66			x	3				
	ROLLC				10	4	XXX.XX^{o}	Commanded roll angle.
	LATANG				7	17	XXXX.X nmi	Cross-range error.
	DNRNGERR				7	15	XXXX.X nmi	Down-range error.

Noun	Cell	No Load	Dec. Only	Number	Routine	Constant	Decimal Display	Quantity
67			x	3				
	RTGON67				7	15	XXXX.X nmi	Range to target.
	LAT				10	4	XXX.XX°	Latitude.
	LONG				10	4	XXX.XX°	Longitude.
68			x	3				
	ROLLC				10	4	XXX.XX°	Commanded roll angle.
	VMAGI				10	9	XXXXX. fps	Inertial velocity.
	RDOT				10	14	XXXXX. fps	Altitude rate.
69			x	3				
	ROLLC				10	4	XXX.XX°	Commanded roll angle.
	Q7				5	16	XXX.XX g	Drag at skip-out.
	VL				10	14	XXXXX. fps	Skip-out velocity.
70				1				
	STARCODE				0	0	Octal	Source/Body code.
71				1				
	STARCODE				0	0	Octal	Source/Body code.
72			x	3				
	FIXTIME				8	0	hr,mn,sc	R27 optimization time.
73				3				
	P21ALT				7	8	XXXXXb. nmi	Altitude (P21 output m/100).
	P21VEL				10	9	XXXXX. fps	Velocity.
	P21GAM				10	4	XXX.XX°	Flight path angle.
74				3				
	ROLLC				10	4	XXX.XX°	Commanded roll angle.
	VMAGI				10	9	XXXXX. fps	Inertial velocity.
	D				5	16	XXX.XX g	Drag acceleration.

Noun	Cell	No Load	Dec. Only	Number	Routine	Constant	Decimal Display	Quantity
75		x	x	3				
	DIFFALT				7	8	XXXX.X nmi	NSR Delta Altitude.
	T1TOT2				9	0	mn, ,sc	TPI-NSR time.
	T2TOT3				9	0	mn, ,sc	TPI-NOMTPI time.
76		x	x	3				
	SVEC				10	23	XXX.XX nmi	R27 current range.
	SVEC+2				7	22	XXXX.X fps	R27 current range rate.
	TFO				9	0	mn, ,sc	Time from R27 optimization.
77			x	3				
	OPVEC				10	23	XXX.XX nmi	R27 optimized range.
	OPVEC+2				7	22	XXXX.X fps	R27 optimized range rate.
	PHETA				10	4	XXX.XX^{o}	R27 angle.
78			x	3				
	UTYAW				10	4	XXX.XX^{o}	P20 "yaw" angle.
	UTPIT				10	4	XXX.XX^{o}	P20 "pitch" angle.
	AZIMANGL				10	4	XXX.XX^{o}	P20 "azimuth" angle.
79			x	2				
	RATEPTC				3	11	X.XXXX^{o}/sec	Rate ("R67START" divides by 10), P20 opt. #2.
	DBPTC				2	2	XXX.XX^{o}	Deadband for P20.
80		x	x	3				
	T_{togo}				9	0	mn, ,sc	Time from event.
	VGDISP				10	9	XXXXX. fps	$\lvert \underline{V}_g \rvert$ value.
	DVTOTAL				10	9	XXXXX. fps	Velocity accumulated.
81			x	3				
	DELVLVC				7	10	XXXX.X fps	Delta-$\underline{V}$ in local vert. coordinates.

Noun	Cell	No Load	Dec. Only	Number	Routine	Constant	Decimal Display	Quantity
82			x	3				
	VGNSR				7	10	XXXX.X fps	NSR velocity change.
83			x	3				
	DELVIMU				7	10	XXXX.X fps	Delta-V in control coordinates.
84			x	3				
	DVDSP1				7	10	XXXX.X fps	Next maneuver Delta V.
	DHDSP				7	8	XXXX.X nmi	Next maneuver Delta H.
	DVDSP2				7	10	XXXX.X fps	Third maneuver Delta V.
85			x	3				
	$\underline{V}_{gbody}$				7	10	XXXX.X fps	$\underline{V}_g$ in control coordinates.
86			x	3				
	DELVLVC				10	9	XXXXX. fps	Delta-V in local vert. coordinates.
87				3				
	DAPDATR3				0	0	Octal	Docked DAP configuration.
	CH5FAIL				0	0	Octal	Channel 5 jet inhibit.
	CH6FAIL				0	0	Octal	Channel 6 jet inhibit.
88			x	3				
	STARSAV3				13	0	.XXXXX	"Planet" vector.
89				2				
	DKRATE				3	20	X.XXXX°/sec	Docked DAP maneuver rate.
	DKDB				3	5	XXX.XX°	Docked DAP deadband.
90			x	3				
	YCSM				4	7	XXX.XX nmi	CSM out-of-plane pos.
	YDOTC				7	10	XXXX.X fps	CSM out-of-plane vel.
	YDOTL				7	10	XXXX.X fps	OWS out-of-plane vel.

Noun	Cell	No Load	Dec. Only	Number	Routine	Constant	Decimal Display	Quantity
91				2				
	CDUS				2	2	XXX.XX^{o}	Optics shaft angle.
	CDUT				6	2	XX.XXX^{o}	Optics trunnion angle.
92				2				
	SAC				2	2	XXX.XX^{o}	Desired optics shaft.
	PAC				6	2	XX.XXX^{o}	Desired optics trunnion.
93				3				
	OGC				7	3	XX.XXX^{o}	Gyro torquing angles.
94				2				
	MRKBUF1+3$_{sp}$				2	2	XXX.XX^{o}	Alternate LOS shaft.
	MRKBUF1+5$_{sp}$				6	2	XX.XXX^{o}	Alternate LOS trunnion.
95			x	3				
	NC1TIG				8	0	hr,mn,sc	NC1 ignition time.
96			x	3				
	RANGE				4	7	XXX.XX nmi	CSM out-of-plane pos.
	RRATE				7	10	XXXX.X fps	CSM out-of-plane vel.
	RRATE2				7	10	XXXX.X fps	OWS out-of-plane vel.
97				3				
	DSPTEM1$_{sp}$				3	0	XXXXX.	System test inputs.
98				3				
	DSPTEM2$_{sp}$				3	0	XXXXX.	System test results.
	DSPTEM2+1$_{sp}$				1	0	.XXXXX	System test results.
	DSPTEM2+2$_{sp}$				3	0	XXXXX.	System test results.
99			x	3				
	WWPOS				7	18	XXXXX. ft	Position W-matrix error/initialization.
	WWVEL				5	19	XXXX.X fps	Velocity W-matrix error/initialization.
	WWOPT				3	0	XXXXX.	Option code.

Quantities in Noun Tables

ACTCENT: See Rendezvous Computations. N52

ADOT: See Digital Autopilot RCS Routines. N56

AGEOFW: See Measurement Incorporation. N31

AK: See Digital Autopilot Interface Routines. NO4

ALMCADR: See General Program Control. NO8

ALT: See Coordinate Transformations. N43

ALTI: See Boost Computations. N62

AZIMANGL: See Orbital and Rendezvous Navigation. N78

CDU: Major Variable. N20

CDUS, CDUT: See Optics Computations. N91

CH5FAIL, CH6FAIL: See Digital Autopilot Docked Jet Selection. N87

CPHIX: See Digital Autopilot RCS Routines. N17

CSMMASS: See Digital Autopilot Interface Routines. N47

D: See Entry Computations. N64, N74

DAPDATR1, DAPDATR2: See Digital Autopilot Interface Routines. N46

DAPDATR3: See Digital Autopilot Docked Jet Selection. N87

DBPTC: See Orbital and Rendezvous Navigation. N79

DELVIMU: See Display Computations. N83

DELVLVC: See Burn Control. N81, N86

DELH1: See Rendezvous Computations. N57

DELVTPF, DELVTPI: See Burn Control. N58

DHDSP: See Rendezvous Computations . N84

DHNCC: See Rendezvous Computations. N57

DIFFALT: See Rendezvous Computations. N75

DKDB, DKRATE: See Digital Autopilot RCS Routines. N89

DNRNGERR: See Entry Computations. N66

Internal Noun Uses

Listed below are the routines which cause the values of particular nouns to be displayed, together with the associated verb used for the initial display generation (which is sometimes written over with another verb).

Noun	Using Routine (Verb)
01	"OHWELL1" (21); "OHWELL2" (21); "SOPTION" (05)
02	"UPVERIFY" (21)
03	Manual initiation only (angle data, address-to-be-specified)
04	Manual initiation only (FDAI error needle information)
05	"R54" (06)
06	See Checklist and Option Codes
07	Manual initiation only (with V25, to change flag/channel bits)
08	Manual initiation only (Alarm data)
09	See the "D" Error Codes (used with V05)
10	Manual initiation only (channel-to-be-specified)
11	"P33" (06)
12	See Checklist and Option Codes
13	"P33" (06); "P34" (06)
14	"R53DISP" (06); "TRKSTAR" (06)
15	Manual initiation only (increment machine address)
16	"R36" (06); "V82GOFF1" (06)
17	Manual initiation only (V63 FDAI error base angles)
18	"TOBALL" (06); "TOBALL" (06 replaced by 50); "V89RECL" (06)
19	"TRKSTAR" (04)
20	"GYCRS" (16)
21	Manual initiation only (accelerometer cells)
22	"IMUATTCK" (25); "P51A" (06); "P52D" (06); "P62.1" (06); "R62DISP" (06); "UPCONTRL" (06); "VBCOARK" (25)

Noun	Using Routine (Verb)
23	"OAVECS" (06)
24	"ALINTIME" (25)
25	See Checklist and Option Codes
26	Manual initiation only (verb 30 and 31 parameters)
27	Manual initiation only (computer self-test switch)
28	"P32" (06)
29	"AZMTHCG1" (06)
30	"GCOMPVER" (05, written over by V06N41)
31	Manual initiation only (time of last r/v W matrix initialization)
32	Manual initiation only (time from pericenter computed in R30)
33	"P30" (06); "P38" (06); "P77" (06)
34	"PASSOUT" (06); "P20OPT" (06); "P21PROG1" (06); "P52B" (06)
35	Manual initiation only (h,m,s display of e.g.R1 of N40)
36	Manual initiation only (computer clock)
37	"P31" (06); "P32" (06); "P33" (06); "P34/P74C" (06); "P35" (06)
38	Manual initiation only (state vector time, T_{et})
39	"P38" (06)
40	"CLOCKJOB" (06 replaced by 97 or 99); "POSTBURN" (16); "P40SXTY" (06); "TIG-0" (06); "TIGAVEG" (06); "V97E" (06); "V97P" (06); "V99P" (06)
41	"GCOMPVER" (06, after N30 loads R3)
42	"P30" (06)
43	"LONGPASS" (06, R1 and R3 blanked); "PASSOUT" (06); "P21PROG2" (06)
44	"V82CALL" (16); "V82GOFLP" (16)
45	"P30" (16); "R23CSM1" (16 replaced by 53); "VN1645" (16)
46	"DONOUN46" (04)

Noun	Using Routine (Verb)
47	"DONOUN46" (06)
48	"DONOUN46" (06)
49	"RENDISP2" (06, priority display)
50	Manual initiation only (optional R30 display, R1 computed in P00 and P11 only)
52	Manual initiation only (central angle of transfer computed in P35 and P36)
53	"R31CALL" (16)
54	"R31CALL" (16)
55	"P34/P74C" (06); "P35" (06)
56	Manual initiation only (DAP-derived vehicle rates)
57	"P31" (06); "P32" (06, R1 blanked)
58	"P34/P74C" (06)
59	"P35/P75B" (06)
60	"NEWRNVN" (06)
61	"P61" (06); "P62.1" (06)
62	"VHHDOT" (06)
63	"NEWRNVN" (16)
64	"P63" (06)
65	Manual initiation only (sampled computer clock)
66	"PREFINAL" (06)
67	"P67.1" (16)
68	Manual initiation only (entry quantities)
69	"P65.1" (16)
70	"R51DSP" (01); "TRKSTAR" (01); "V1N70DSP" (01)
71	"R53C" (01)

Noun	Using Routine (Verb)
72	"P25CSM" (06); "V76CALL" (06); "72SHOWX" (06)
73	Manual initiation only (quantities computed in P21)
74	"INITROLL" (06)
75	"P33/P73B" (06)
76	"BOTHSHOW" (16); "P47BODY" (16)
77	"BOTHSHOW" (16); "P47BODY" (16)
78	"DOV6N78" (06); "V64PERF" (06); "V89CALL" (06, R3 blanked)
79	"DOV6N78" (06)
80	Manual initiation only (R2 and R3 different scales from N40)
81	"NC12H" (06); "N81DISP" (06); "P30" (06); "P33/P73B" (06); "P34/P74C" (06); "P38RECYC" (06); "P77" (06)
82	"NCCDMP7+17" (06)
83	"P47BODY" (16)
84	"NC12DH" (06)
85	"P40SXTY" (06); "TIGAVEG" (16); "TIGNOW" (16)
86	Manual initiation only (different scaling from N81)
87	"DKDISP" (05)
88	"PLANET" (06); "V1N70DSP" (06)
89	"DONOUN89" (06, R3 blanked)
90	Manual initiation only (out-of-plane parameters)
91	Manual initiation only (optics CDU angles)
92	"R53CHK" (06); "VBCOARK" (24)
93	"GCOMPVER" (06); "IMUFINEK" (25); "R55" (06)
94	"R23CSM" (06); "R56" (06); "V64PERF" (06)
95	"P31" (06)

Noun	Using Routine (Verb)
96	"R36" (06)
97	Manual initiation only (DSPTEM1 cells for system test inputs)
98	"SHOW" (06)
99	"GOTOPOOH" (37, which then blanks noun); "V67CALL" (06)

Optics Computations

OPTTEST Entered from "PROCEEDE" every 0.24 seconds

TS = CDUS (Tag here "OPTDRIVE")

$TS_1 = |TS| - K_{45dg}$

If $TS_1 \leq 0$:

ZONE = 0

If $TS_1 > 0$:

If ZONE = 0:

ZONE = TS

If OPTIND $\leq$ -0:

If OPTIND < -0:

Set bit 8(TVC Enable) of channel 12 = 0

Resume

If SWSAMPLE < 0: (Zero optics mode)

Resume

If SWSAMPLE = 0: (Manual mode)

If bit 8(TVC Enable) of channel 12 = 0: (Tag here "TVCBCK")

Set bit 8(TVC Enable) of channel 12 = 1

Resume

If bit 10(ZROPTFIN) of OPTMODES = 0: (Tag here "RATEDRV1")

Perform "ALARM" (pattern 0120_8)

If bit 2(Enable Optics CDU Error Counters) of channel 12 = 0:

Set bit 2(Enable Optics CDU Error Counters) of channel 12 = 1

Resume

COMMANDS = (DESOPTS - CDUS), rescaled to B1 revolutions. The ones complement difference of the twos complement numbers is formed, and a rounded shift employed in the rescaling from B-1 to B1 revolutions.

TS = DESOPTT - CDUT (treated as ones complement numbers)

If $|TS| < 2^{-3}$ rev (45^{o}):

COMMANDT = (DESOPTT - CDUT), rescaled to B-1 revolutions. The ones complement difference of the twos complement numbers is formed, and a rounded shift employed in the rescaling from B-3 to B-1 revolutions.

If $|TS| \geq 2^{-3}$ rev (45^{o}):

COMMANDT = +MAX sgn TS

OPTIND = +0

$TS_1 = 0$ (ITEMP1 tag)

If $|CDUS| - K_{90dg} > 0$: (i.e. 2nd or 3rd quadrant)

If ZONE $\neq$ 0: (should be)

If sgn ZONE = sgn COMMANDS: (+0 is +, -0 is -)

$TS_2 = |DESOPTS|$

If sgn ZONE $\neq$ sgn COMMANDS:

TS_2 = +MAX

If $TS_2 - K_{90dg} \leq 0$: (i.e. DESOPTS 1st or 4th quad.)

COMMANDS = - COMMANDS

If SWSAMPLE = 0: (Manual mode; Tag "CMDSETUP")

CDUSCMD = -0 (Tag here "RATEDRV2")

CDUTCMD = -0

Set bits 12-11 (Gate outputs from CDUiCMD, i = T, S) of channel 14 = 1

Resume

Set bit 8(TVC Enable) of channel 12 = 0

Perform the following for i = S,T: (Tag here "CMDSET")

If COMMANDi = 0:

CDUiCMD = -0

If COMMANDi $\neq$ 0:

$TS_1 = TS_1 + 1$ (non-zero command required)

$TS = |COMMANDi| + K_{mxpsl}$

If TS $>$ 0:

$CDUiCMD = - K_{mxps}$ sgn COMMANDi

If TS $\leq$ 0:

CDUiCMD = COMMANDi

If $TS_1 > 0$:

Set bits 12-11 (Gate outputs from CDUiCMD, i = T, S) of channel 14 = 1

Resume

OPTMON Entered from "PROCEEDE" every 0.48 seconds

TS = bit 7(Optics CDU Fail complement) of channel 30

If TS $\neq$ bit 7(OCDUFBIT) of OPTMODES:

Perform "OCDUFTST"

If OPTIND = -0:

Resume (note that bit 7 of OPTMODES not updated)

If OPTIND $<$ 0: (set -1 by e.g. "INITSUBA")

Set bit 2(Enable Optics CDU Error Counters) of channel 12 = 0

Set bit 7(OCDUFBIT) of OPTMODES = TS

Set bits 5-4(OPMD1BIT, OPMD2BIT) of OPTMODES = bits 5-4 (Computer Control of Optics complement and Zero Optics Mode complement) of channel 33

If bits 5-4 of C31FLWRD $\neq 00_2$:

Set bits 5-4 of OPTMODES = bits 5-4 of C31FLWRD

If bits 5-4 of OPTMODES $= 11_2$:

SWSAMPLE = 0 (Manual mode)

```
If bits 5-4 of OPTMODES ≠ 11₂:
     If bit 5(OPMD1BIT) of OPTMODES = 0:
          SWSAMPLE = 15     (Computer Control)
     If bit 5(OPMD1BIT) of OPTMODES = 1:
          SWSAMPLE = -1
If DESOPMOD > 0:        (previous cycle Computer Control; Tag "PROCESSW")
     If SWSAMPLE > 0:        (still Computer Control; Tag "CSCDES")
          DESOPMOD = SWSAMPLE
          Resume
     If SWSAMPLE < 0:     (from Computer Control to Zero)
          Proceed to "CSCTOZOP"
     Proceed to "CSCTOMAN"   (from Computer Control to Manual)
If DESOPMOD = 0:        (previous cycle Manual mode)
     If SWSAMPLE > 0:  (from Manual to Computer Control; Tag "MANUDES")
          WTOPTION = 0
          ZOPTCNT = 0
          Proceed to "CSCTOMAN"
     If SWSAMPLE = 0:      (still Manual)
          WTOPTION = WTOPTION - 1, limited ≥ +0
          DESOPMOD = SWSAMPLE
          Resume
     If WTOPTION = 0:        (from Manual to Zero)
          Proceed to "CSCTOZOP"
     Proceed to second line of "CSCTOZOP"
```

```
If DESOPMOD < 0:   (as it will)   (previous cycle Zero optics)
    If SWSAMPLE > 0:    (from Zero to Computer Control; Tag "ZOPTDES")
        If bit 3(ZOPTCS) of OPTMODES = 0:    (Tag "ZTOCSC")
            Proceed to "CSCTOMAN"
        Perform "ALARM"  (pattern 0116g)
        Set bits 3-2 (ZOPTCS, OCDUINHT) of OPTMODES = 0
        Set bit 1(Zero Optics CDU) of channel 12 = 0
        WTOPTION = 0
        ZOPTCNT = 0
        Proceed to "CSCTOMAN"
    If SWSAMPLE = 0:     (from Zero to Manual)
        If bit 3(ZOPTCS) of OPTMODES = 0:    (Tag "ZTOMAN")
            Proceed to "CSCTOMAN"
        Perform "ALARM"  (pattern 0116g)
        WTOPTION = 11
        Set bits 3-2 (ZOPTCS, OCDUINHT) of OPTMODES = 0
        Set bit 1(Zero Optics CDU) of channel 12 = 0
        Proceed to "CSCTOMAN"
    If bit 3(ZOPTCS) of OPTMODES = 0:    (still Zero)
        DESOPMOD = SWSAMPLE
        Resume
    If ZOPTCNT > 0:
        ZOPTCNT = ZOPTCNT - 1
        DESOPMOD = SWSAMPLE
        Resume
    Set bit 1(Zero Optics CDU) of channel 12 = 1  (Tag "SETZOEND")
```

(If DESOPMOD < 0):

Call "ENDZOPT" in 0.20 seconds

DESOPMOD = SWSAMPLE

Resume

ENDZOPT

CDUS = 0

ZONE = 0

CDUT = - K_{20degs}

Set bit 1(Zero Optics CDU) of channel 12 = 0

Delay 0.20 seconds

Set bit 10(ZROPTFIN) of OPTMODES = 1

Set bits 3-2 (ZOPTCS, OCDUINHT) of OPTMODES = 0

Perform "OCDUFTST"

End of task

OCDUFTST

If bit 7(Optics CDU Fail complement) of channel 30 = 0:

If bit 2(OCDUINHT) of OPTMODES = 1:

Return

If bit 8 of DSPTAB+11 = 0:

Set bit 8(Tracker) of DSPTAB+11 = 1, and flag for output at next opportunity

Return

If bit 1(LMPTSTBT) of IMODES33 = 1:

Return

If bit 8 of DSPTAB+11 = 1:

Set bit 8(Tracker) of DSPTAB+11 = 0, and flag for output at next opportunity

Return

CSCTOZOP

ZOPTCNT = 32

WTOPTION = 0

Set bits 3-2 (ZOPTCS, OCDUINHT) of OPTMODES = 1

Set bit 8(TVC Enable) of channel 12 = 0

Set bit 2(Enable Optics CDU Error Counters) of channel 12 = 0

DESOPMOD = SWSAMPLE

Resume

CSCTOMAN

If OPTIND ≫ +0:

Set bit 2(Enable Optics CDU Error Counters) of channel 12 = 0

Call "ECENAB" in 0.06 seconds

DESOPMOD = SWSAMPLE

Resume

ECENAB

If SWSAMPLE = 0: (Manual mode)

Set bit 8(TVC Enable) of channel 12 = 1

Set bit 2(Enable Optics CDU Error Counters) of channel 12 = 1

End of task

SXTMARK

Perform "TESTMARK"

MARKINDX = 1

OPTCADR = Return address (to routine calling "SXTMARK")

Proceed to "MKVB51"

TESTMARK Entered from "R56" and "SXTMARK"

If bits 3-2 of EXTVBACT $\neq 00_2$:

Proceed to "BAILOUT" (pattern 31211_8)

Set bit 2 of EXTVBACT = 1

Return

MKRELEAS (Tag also "MKRLEES")

Inhibit interrupts

OPTIND = -1

MARKINDX = 0

Set bit 4(MARKFLG) of FLAGWRD1 = 0

Release interrupts

Return

MKVB51

Perform "KLEENEX"

Proceed to "MKVBDSP"

MKVBDSP

TS = 5100_{vn}

Proceed to "GOMARK4": if terminate, proceed to "TERMSXT"
if proceed, proceed to "ENTANSWR"
otherwise, proceed to "MKVB5X"

ENTANSWR

Establish "ENDEXT" (priority 24_8)

Change priority of present job to 13_8 (allows "ENDEXT" to be executed)

Proceed to address specified by OPTCADR

MKVB5X

If MARKINDX > 0:

Proceed to "MKVB51"

Proceed to "MKVB50"

MKVB50

DSPTEM1 = 00016_g

TS = 5025_{vn}

Proceed to "GOMARK4": if terminate, proceed to "TERMSXT"
if proceed, proceed to "ENTANSWR"
otherwise, proceed to "MKVB5X"

TERMSXT

Perform "CLEARMRK"

Perform "MKRELEAS"

If MODREG = 03:

Proceed to "GCOMP5"

Proceed to "GOTOPOOH"

MARKRUPT Entered after receipt of program interrupt #6, navigation panel DSKY key code or optics mark/mark reject

MKCDUT = CDUT

MKCDUS = CDUS

MKCDUY = CDU_y

MKCDUZ = CDU_z

MKCDUX = CDU_x

MKT2T1 = T_{now}

T_{st} = MKT2T1

If bit 6(Optics Mark) of channel 16 = 1:

TS = $C_{cduchkwd}$, limited $\geq$ 1 centi-second (tag here "MARKIT")

Call "MARKDIF" in TS centi-seconds

Resume

If bit 7(Optics Mark Reject) of channel 16 = 1:

Proceed to "MKREJECT"

TS_1 = bits 5-1 (DSKY input) of channel 16

If $TS_1 \neq 0$:

Proceed to "KEYCOM"

Perform "ALARM" (pattern 0113_g)

Resume

MARKDIF

If $C_{cduchkwd} > 0$:

If any $|MKCDUi - CDU_i| + K_{m3bt} > 0$: (i = X,Y,Z)

Perform "ALARM" (pattern 0121_g)

End of task

If bit 14(R21MARK) of FLAGWRD2 = 1: (Tag here "MARKCONT")

Proceed to "PUTMARK"

If MARKINDX = 0:

Perform "ALARM" (pattern 0114_g)

End of task

MARKINDX = MARKINDX - 1 (Tag here "MARK2")

Set bit 4(MARKFLG) of FLAGWRD1 = 1

Proceed to "PUTMARK"

PUTMARK

MRKBUF1+i = MKj (i = 0-6; j = $T2T1_{dp}$, CDUY, CDUS, CDUZ, CDUT, CDUX)

If bit 14(R21MARK) of FLAGWRD2 = 1:

End of task

Proceed to "MARKDONE"

MARKDONE

If MARKINDX = 0:

Established "MKVB5X" (priority 22_g)

End of task

MKREJECT

If bit 14(R21MARK) of FLAGWRD2 = 1:

If MRKBUF1 $\geq$ 0: (i.e. last mark not yet being processed by R22)

MRKBUF1 = -1

Resume

If bit 7(R22CAFLG) of FLAGWRD9 = 1: (set by "REND1")

Set bit 12(REJCTFLG) of FLGWRD10 = 1 (for "REND7" use)

Resume

If bit 4(MARKFLG) of FLAGWRD1 = 0:

Perform "ALARM" (pattern 0110_8)

Resume

Set bit 4(MARKFLG) of FLAGWRD1 = 0 (Tag here "REJECT3")

MARKINDX = MARKINDX + 1

Establish "MKVBDSP" (priority 22_8)

Resume (Note that "KLEENEX" bypassed, so may have more than F V51 on DSKY, e.g. N25 and R1 = 00016_8 also)

R23CSM Established by "GOTOR23" for a V54E

Set bit 14(R21MARK) of FLAGWRD2 = 0

MRKBUF1 = -1

Change priority of present job to 27_8 (higher than R22)

TS = 0694_{vn}

Proceed to "GOXDSPF": if terminate, proceed to "R21END"
if proceed, proceed
otherwise, proceed to previous line

Proceed to "R23CSM1"

R23CSM1

TS = 5345_{vn}

Proceed to "MARKMONR": if terminate, proceed to "R21END"
if proceed, proceed to "R21END"
otherwise, proceed

Inhibit interrupts

$MRKBUF1_{dp} = T_{now}$

$MRKBUF1+2 = CDU_y$ (MRKBUF1+3 and MRKBUF1+5 loaded by N94 in "R23CSM")

$MRKBUF1+4 = CDU_z$

$MRKBUF1+6 = CDU_x$

Release interrupts

Proceed to "R23CSM1" ("V86PERF", if done promptly, can be used to reject the mark)

R21END

Perform "KLEENEX"

MRKBUF1 = -1

Change priority of present job to 16_8 (allow R22 to finish if processing mark, with proper R21MARK bit value)

Set bit 14(R21MARK) of FLAGWRD2 = 1

Proceed to "ENDEXT"

R52 Entered from "PIKUP20" and "R51DSP"

SAVQR52 = Return address

DESOPTT = CDUT

DESOPTS = CDUS

OPTIND = +0

Set bit 6(SGTMK) of FLAGWRD0 = 0 (notation R53FLAG)

If bit 10 (LMTRG) of FLAGWRD1 = 1: (notation TARG1FLG, means tracking OWS)

Proceed to "R52H"

Set bit 15(TERMIFLG) of FLAGWRD7 = 0

Proceed to "R52C"

R52C

If SWSAMPLE > 0: (i.e. Computer Control of optics)

Proceed to "R52D"

If bit 6(SGTMK) of FLAGWRD0 = 0: (Tag here "R52M")

Establish "R53JOB" (priority 24_8)

Proceed to "R52F"

R52F

TS = 0.50 second

Delay TS seconds (by putting job to sleep via "DELAYJOB")

If bit 10(LMTRG) of FLAGWRD1 = 1:

Proceed to "R52H"

If bit 15(TERMIFLG) of FLAGWRD7 = 1: (set by end of "R53" package)

Proceed to address specified by SAVQR52

Proceed to "R52C"

R52H (Entered only if tracking OWS, from "R52" and "R52F")

Perform "R61CSM"

If bit 5(TRACKFLG) of FLAGWRD1 = 0:

Proceed to address specified by SAVQR52

If bit 7(UPDATFLG) of FLAGWRD1 = 1:

Proceed to "R52D"

TS = 1.80 seconds

Proceed to second line of "R52F"

R52D

If bit 10(LMTRG) of FLAGWRD1 = 1:

$T_{decl} = T_{now} + K_{2p4secdp}$ (Tag here "LEM52")

Perform "LEMCONIC"

$\underline{STAR} = \underline{R}_{att}$

$T_{decl} = T_{att}$

Perform "CSMCONIC"

$\underline{TS} = \text{unit}(\underline{STAR} - \underline{R}_{att})$

Proceed to "COM52"

If STARIND = 0:

TS = STARSAV2

If STARIND > 0:

TS = STARSAV1

Proceed to "COM52"

COM52

STAR = unit([REFSMMAT] TS)

Perform "CDUTRIG"

Perform "CALCSXA"

If bit 7(CULTFLAG) of FLAGWRD3 = 1:

If bit 10(LMTRG) of FLAGWRD1 = 0: (Tag here "R52L")

TS = 0404_g

Perform "PRIOLARM": if terminate, proceed to "TERM52"
if proceed, skip next line
otherwise, skip next line

End of job

Proceed to "R52F"

PACTEMP = $K_{50dgtr_{sp}}$

If bit 7(CULTFLAG) of FLAGWRD3 = 0:

If (PAC - K_{50dgtr}) < 0 and (PAC - $K_{20dgsmn}$) ≥ 0:

PACTEMP = PAC

Skip next line

PACTEMP = $K_{50dgtr_{sp}}$

If bit 10(LMTRG) of FLAGWRD1 = 1: (Tag here "R52JA")

Proceed to "R52E"

Proceed to "R53CHK"

R53CHK

If bit 6(SGTMK) of FLAGWRD0 = 0:

TS = 0692_{vn}

Perform "GODSPR"

Proceed to "R52E"

R52E

DESOPTT = PACTEMP

DESOPTS = SAC

Proceed to "R52F"

R53JOB Established by "R52C"

Perform "R53"

End of job

TERM52 Entered for V34E response to 0404_{g} alarm display

Perform "KLEENEX"

Proceed to "TERMSXT"

R53

R53EXIT = Return address

Set bit 6(SGTMK) of FLAGWRD0 = 1 (program notation R53FLAG)

Perform "SXTMARK"

If MARKINDX > 0: (no marks taken)

Proceed to third line of "R53"

Perform "MKRELEAS" (Tag here "R53A1")

Proceed to "R53C1"

R53C1 Entered from "R53" and "R56"

TS = 0

Perform "CLEANDSP"

If (MODREG - 32) $\leq$ 0: (should not be)

Set bit 15(TERMIFLG) of FLAGWRD7 = 1

Proceed to address specified by R53EXIT

If bit 12(P50.1FLG) of FLAGWRD0 = 1:

Set bit 15(TERMIFLG) of FLAGWRD7 = 1

Proceed to address specified by R53EXIT

Proceed to "R53C"

R53C

TS = 0171_{vn}

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
if proceed, proceed
otherwise, proceed to previous line

If STARCODE $\leq$ -0 or if (bits 6-1 of STARCODE) $> 47_8$:

Set bit 7(Operator Error) of channel 11 = 1

Proceed to "R53C"

If bit 7 of STARCODE = 1:

If STARCODE $\neq 00146_8$: (i.e. sun not specified)

Set bit 7(Operator Error) of channel 11 = 1

Proceed to "R53C"

TS = 6 (bits 6-1 of STARCODE)

If STARIND = 0:

BESTI = TS

If STARIND = 1:

BESTJ = TS

If MODREG $\neq$ 50:

If bit 8 of STARCODE = 0: (e.g. not of form 002xx)

Set bit 15(TERMIFLG) of FLAGWRD7 = 1

Proceed to address specified by R53EXIT

Proceed to "R53DISP"

R53DISP

TS = 0614_{vn}

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
if proceed, proceed
otherwise, proceed to previous line

TS = TRKAZ, rescaled (by doubling) to B-1 rev.

MRKBUF1+3 = TS, modulo 2^{-1} rev. (180^{o})

If $|TS| \geqslant \frac{1}{2}$: (i.e. bigger than 180^{o})

Set bit 7(Operator Error) of channel 11 = 1

Proceed to "R53DISP"

TS = TRKEL, rescaled (by doubling) to B-1 rev.

MRKBUF1+5 = TS, modulo 2^{-1} rev. (180^{o})

If $|TS| \geqslant \frac{1}{2}$: (i.e. bigger than 180^{o})

Set bit 7(Operator Error) of channel 11 = 1

Proceed to "R53DISP"

Set bit 15(TERMIFLG) of FLAGWRD7 = 1

Proceed to address specified by R53EXIT

R56 Entered from "P51B" and "R51DSP"

R53EXIT = Return address

TS = 0694_{vn}

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
if proceed, proceed
otherwise, proceed to previous line

Perform "TESTMARK" (Tag here "R56A")

TS = 0

Perform "CLEANDSP"

TS = 5300_{vn}

Proceed to "GOMARK2": if terminate, proceed to "GOTOPOOH"
if proceed, proceed to previous line
otherwise, proceed

SAC = MRKBUF1+3 (loaded in R1 of N94)

PAC = MRKBUF1+5 (loaded in R2 of N94)

Inhibit interrupts (SAC and PAC loading done in interpretive language, maximizing the delay between ENTR response and the sampling of CDU angles. Also can be delayed if R67 active, since higher priority than P54/R56).

$MRKBUF1_{dp} = T_{now}$

$MRKBUF1+2 = CDU_y$

$MRKBUF1+4 = CDU_z$

$MRKBUF1+6 = CDU_x$

Release interrupts

Perform "CLEARMRK"

$TS = 00016_8$

Proceed to "GOPERF1": if terminate, proceed to "GOTOPOOH"
if proceed, proceed
otherwise, proceed to 4th line of "R56"

Proceed to "R53C1"

See also list of major variables and list of routines

BESTI, BESTJ: See Inflight Alignment.

$C_{cduchkwd}$: Single precision erasable memory constant, program notation "CDUCHKWD", scale factor B14, used to specify (if positive non-zero) the number of centi-seconds delay before "MARKDIF" is performed after receiving an optics mark button input. If the cell is zero or negative, the delay is 0.01 second.

C31FLWRD: See Digital Autopilot RCS Routines.

CDUS: Single precision value of counter input cell 0036_8, containing the optics CDU shaft angle (also identified with the X axis), scale factor B-1, units revolutions, in twos complement.

CDUT: Single precision value of counter input cell 0035_8, containing the optics CDU trunnion angle (also identified with the Y axis), scale factor B-3, units revolutions, in twos complement. See K_{20degs}.

CDUiCMD (i = T, S): Single precision value of computer special erasable memory cells 0053_8 - 0054_8 respectively. Pulses are generated based on the contents of these cells if bits 12-11 respectively of channel 14 = 1, and the optics CDU error counter is loaded from the 3200 pps pulse train information if bit 2 of channel 12 = 1. For the output to give one revolution, 2^{15} pulses from CDUTCMD, and 2^{13} pulses from CDUSCMD, are required. The same cells are also used for TVCYAW and TVCPITCH commands for the SPS engine (see Digital Autopilot Interface Routines).

COMMANDT, COMMANDS: Raw values of trunnion and shaft commands computed in "OPTTEST", same scaling as CDUiCMD. The sign of COMMANDS may be reversed for shaft-stop avoidance purposes.

DESOPMOD: Single precision value of SWSAMPLE the previous time that "OPTMON" was performed, used to determine if a change has taken place in the setting of the optics mode switches (Computer Control, Manual, or Zero).

DESOPTT, DESOPTS: Single precision values of desired CDUT and CDUS respectively, twos complement with same scaling as CDUT and CDUS respectively. They are used in "OPTTEST" to generate optics driving commands (if OPTIND and SWSAMPLE are satisfactory).

EXTVBACT: See Verb Definitions.

IMODES33: See IMU Computations.

$K_{2p4secdp}$: Constant, program notation "2.4SECDP", scale factor B28, units centi-seconds. Value is 240 x 2^{-28}, corresponding to 2.4 seconds.

K_{20degs}: Single precision constant, program notation "20DEGS", scale factor B-3, units revolutions. Value is 7199 x 2^{-14}, but since used (in "ENDZOPT") in complement form, value equivalent to 7200 x 2^{-14}, or 19.7754°. When the optics are "zeroed", the CDUT cell is loaded with - K_{20degs}. Consequently, to find the "true" trunnion angle, this 19.7754° value must be added, and to convert "true" to actual cell contents this 19.7754° value must be subtracted. See e.g. "CALCSXA". The octal value of - K_{20degs} is 61740_8.

$K_{20dgsmn}$: Constant, program notation "20DEGSMN", scale factor B-3, units revolutions. Value is -7199 x 2^{-14} (as a ones complement number), equivalent to -19.7754° (see K_{20degs}). Causes limiting of trunnion value if the derived "true" angle is negative.

K_{45dg}: Single precision constant, program notation "13,14,15", scale factor B-1, units revolutions. Value is 70000_8, corresponding to a value of 10000_8 as used in program (after a one-bit correction for convenience in forming absolute value), or 45°. If CDUS is = or less than this value, then ZONE is set 0; if greater and ZONE is zero, then ZONE is set to CDUS.

K_{50dgtr}: Constant, program notation "38TRDEG", scale factor B-3, units revolutions. Value is 0.66666667, corresponding to 30° in the value of CDUT, or a "true" trunnion angle of 30 + 19.7754 = 49.7754° (see K_{20degs}). Constant formerly was 0.4 (a "true" value of 37.7754°), hence the notation.

K_{90dg}: Single precision constant, program notation "NEG1/2", scale factor B-1, units revolutions. Value is 57777_8, corresponding to a value of 20001_8 as used in the program (after a one-bit correction for convenience in forming absolute value), or about 90°. If CDUS magnitude is less than 90°, then there is considered to be no problem with the optics stops.

K_{m3bt}: Single precision constant, program notation "NEG2", scale factor B-1, units revolutions. Value is -2 x 2^{-14}, but used in program in such a way (for convenience in forming absolute value) that the effective value in program is -3, or about -0.033°. A change in an IMU CDU angle of more than 3 least increments in $C_{cduchkwd}$ centi-seconds causes the optics mark to be rejected in "MARKDIF", provided $C_{cduchkwd}$ is positive non-zero.

K_{mxps}: Single precision constant, program notation "MAXPLS", scale factor B14, units optics CDU output pulses. Value is -83 x 2^{-14}. See CDUiCMD above for pulses/revolution: note that B1 revolutions is equivalent to B14 pulses if 2^{13} pulses = 1 revolution.

K_{mxpsl}: Single precision constant, program notation "MAXPLS1", scale factor B14, units optics CDU output pulses. Value is -82×2^{-14}, but used in program (for convenience in forming absolute value) in such a way that effective value is -83 (see K_{mxps}).

MARKINDX: Single precision cell, scale factor B14, loaded in "SXTMARK" with 1, and decremented in "MARKDIF" so as to contain the count of the number of marks remaining (i.e. decremented to 0: cell a hold-over from earlier design requiring a capability for more than a single mark in some programs). It is incremented in "MKREJECT" when a mark is rejected. "R53" checks the cell and if non-zero re-enters the marking routine, so as to force a mark to be processed.

MKCDUi, MKT2T1: Set of buffer cells loaded in "MARKRUPT" with optics and IMU CDU angles (with time in MKT2T1, double precision), and used in "PUTMARK" to load appropriate cells. The scaling of the cells is the same as that of the quantities with which they are loaded, and their sequence is $MKT2T1_{dp}$, MKCDUY, MKCDUS, MKCDUZ, MKCDUT, and MKCDUX.

MRKBUF1: Set of cells loaded in "PUTMARK" with optics mark information. The cells have data in the same sequence as described above for MKCDUi, MKT2T1. For common use of processing routines, "R23CSM1" and "R56" load these cells with information in the same format as that done for normal optics marks. They are also loaded in "R53DISP" (after the N14 star tracker specification) with TRKAZ (in MRKBUF1+3) and TRKEL (in MRKBUF1+5). Note in this case, however, MRKBUF1+5 has information scaled B-1 revolutions, not B-3 revolutions as it has when it is used for CDUT information (either prime optics or equivalent N94 R2 input). For R22 use, a setting of MRKBUF1 (first cell) to a negative number indicates that no mark data are in the cells. For R22, a mark reject (or V86E) must be done sufficiently promptly (e.g. within 7 seconds) after a mark to avoid having it be incorporated (barring the N49 display). The notation "MRKBUF1" refers to the first cell of the set, i.e. MRKBUF1+0. MRKBUF1+3 is displayed in R1 of N94 and MRKBUF1+5 in R2 of N94.

OPTCADR: Single precision cell used to contain return address for "SXTMARK".

OPTIND: Single precision control cell, scale factor B14, used to control the performance of "OPTTEST". If it is negative, including -0, driving of the optics is bypassed. A setting of -0 means that the optics are "reserved": this setting is done in "S40.6" when the output channel bits are set for TVC use (see CDUiCMD above). The setting to -1 is done in e.g. "INITSUBA" in conjunction with V37 processing.

OPTMODES: Single precision flag word used for control of "T4RUPT" computations associated with the optics hardware. The individual bits have the following meanings:

Bit	Symbol	Meaning
10	ZROPTFIN	Zeroing of optics has been completed since last fresh start or restart if bit is 1.

Bit	Symbol	Meaning
7	OCDUFBIT	Nominal last sampled value of channel 30 bit 7 (0 if an optics CDU fail indication has been generated by the optics CDU hardware). If OPTIND = -0, the value of the bit is not changed in "OPTMON", although a channel 30 bit 7 change will cause "OCDUFTST" to be entered to change bit 8 of DSPTAB+11 (the Tracker light).
5	OPMD1BIT	Last sampled value of channel 33 (or C31FLWRD) bit 5: zero if Computer Control mode of optics desired.
4	OPMD2BIT	Last sampled value of channel 33 (or C31FLWRD) bit 4: zero if Zero Optics mode of optics desired. If bits 5-4 = 11_2, Manual mode of optics desired.
3	ZOPTCS	Bit set 1 in "CSCTOZOP" to indicate that the wait for zeroing of optics is in process. It is set zero in "ENDZOPT", after the completion of the necessary wait.
2	OCDUINHT	Bit set 1 to inhibit generation of Tracker alarm (bit 8 of DSPTAB+11) within "OCDUFTST" routine.

PAC: See Coordinate Transformations.

PACTEMP: Single precision cell, same scaling as PAC, used to retain the value of PAC derived in R52, with limiting if necessary, that is subsequently loaded into DESOPTT.

R53EXIT: Single precision cell containing return address from "R53" or "R56" (allowing "R53C1" to be used by both).

SAC: See Coordinate Transformations.

SAVQR52: Single precision cell containing return address from "R52".

STAR: See Coordinate Transformations. Used in "R52D" to retain temporarily the OWS position vector (scale factor B29, units meters).

STARCODE: See Inflight Alignment.

STARIND: See Inflight Alignment. It is set 0 in "R00".

STARSAV1, STARSAV2: See Inflight Alignment.

SWSAMPLE: Single precision cell containing information on the value of the optics hardware mode last sampled in "OPTMON" (DESOPMOD is loaded with the same value before termination of the routine), scale factor B14. A positive non-zero value (e.g. 15) means Computer Control mode; a zero value means Manual mode; and a negative value (-1) means Zero Optics mode.

T_{st}: See Data Input/Output.

TRKAZ, TRKEL: See Inflight Alignment.

WTOPTION: Single precision cell, scale factor B14, set to 11 in "OPTMON" if switch from Zero to Manual optics mode with zeroing not yet complete (alarm 0116_8 is generated also). It is decremented by 1, with a lower limit of +0, when "OPTMON" is entered with previous and present cycle modes both Manual. If switch back to Zero from Manual before counter decremented to 0 (which occurs in about 5.3 seconds), then ZOPTCNT is not reset to 32. Cell set 0 in "CSCTOZOP" and in "OPTMON" if go from Zero to Computer Control.

ZONE: Single precision cell set 0 in "ENDZOPT" or in "OPTTEST" if CDUS magnitude less than 45°, and set in "OPTTEST" if CDUS magnitude exceeds 45° and cell presently 0. It is used in "OPTTEST" in the implementation of the optics shaft stop monitor (if non-zero, only its sign is employed). Cell is required because optics shaft stops are at $\pm 270^\circ$ from "zero" ("zero" is the position at which optics are left after zero mode employed). If, after zeroing, were to drive optics towards e.g. $+70^\circ$, then when CDUS exceeded 45° ZONE would be set to positive non-zero. If continued to drive in the same direction, e.g. to $+100^\circ$, then $+140^\circ$, then $+170^\circ$ (always with increments of less than 180° from present CDUS, so go "shortest" way, namely in same direction), then ZONE would remain positive non-zero and optics would continue to drive. If, from the 170° point, the next DESOPTS were -160° (or, equivalently, $+200^\circ$), then "shortest distance" would again be the 30° across the 180° point, and the optics would be driven there, with ZONE remaining untouched (CDUS reading would be "-160°" when reach DESOPTS). Note that, however, if had started at the "zero" point, would have driven optics in the opposite direction (i.e. though -30, -70, etc.). If the next DESOPTS were e.g. -100°(i.e. $+260^\circ$), then here the optics would again be driven in the same direction, with ZONE unchanged, from -160° to -100°, or only 10° away from the stop at $+270^\circ$ (note that "normal" method of reaching -100°, from "zero", would have gone the other way). At the -100° (actually $+260^\circ$ as far as how shaft angle got there is concerned) point, ZONE is still positive non-zero. If the next DESOPTS were e.g. -60° (i.e. $+300^\circ$), this would generate a COMMANDS of $(-60^\circ) - (-100^\circ) = +40^\circ$: to drive there directly, however, would require violating the hardware stop at 270°, and hence the stop monitor logic is invoked. With these conditions, all the conditions at the top of page OPTC-2 are satisfied, since CDUS magnitude (100°) exceeds 90°; ZONE $\neq 0$; the sign of ZONE = sign of COMMANDS (both positive); and magnitude of DESOPTS is 60° (less than 90°). Consequently, the COMMANDS polarity is reversed, so that instead of driving towards -60°, the optics are driven towards -140° (-100 + -40). Assuming that DESOPTS remains -60°, the next attempt would be to drive towards -220° ($+140^\circ$, i.e. -140 -(-60 --140)), and so on. When CDUS is driven below 90°, then shaft stop logic is no longer involved, since "shortest path" is likewise one that avoids the optics shaft

stops. Finally, when optics reaches the -60° point (the "long way"), ZONE would have been reset to 0 and then to negative non-zero, to protect the stop at -270° ($+90^{\circ}$). If the value of DESOPTS were to "chatter" near a magnitude of 270, then the optics themselves would be driven in alternating directions: in this case, manual selection of optics zeroing should be done.

ZOPTCNT: Single precision cell, scale factor B14, set to 32 at the start of optics zeroing (assuming no WTOPTION effect), to give an optics zeroing period of $33 \times 0.48 + 0.4 \approx 16.2$ seconds. An alarm (pattern 116_8) is generated if switch out of zeroing before completion of initial part of delay (about 15.8 seconds from when switch first sensed to be in zero mode). See WTOPTION (no setting of WTOPTION is made if switch to Computer Control: it is reset to 0 instead).

Orbital Integration

STATEINT — Called by "ENDINT", and restarted via group 2.3

Establish "STATINT1" (priority 05_8)

End of task

STATINT1 — Established by "NDUTINPT" if bit 9(UTFLAG) of FLAGWRD8 = 1, by "STATEINT", and by restart group 2.5 (in "R00" for P00)

If bit 5(QUITFLAG) of FLAGWRD9 = 1: (set by "VERB96")

Make restart group 2 inactive

Set bit 5(QUITFLAG) of FLAGWRD9 = 0

End of job (must key e.g. V37E00E to get periodic integration)

$T_{decl} = T_{now}$

Set bit 3(V96ONFLG) of FLAGWRD8 = 0

Perform "INTSTALL"

Set bit 1(NODOV37) of FLAGWRD2 = 1

Set bit 5(STATEFLG) of FLAGWRD3 = 1

Set bits 4(CONICINT) and 1(WMATINT) of FLAGWRD3 = 0

Set bit 9(POOFLAG) of FLAGWRD3 = 1

Set bit 3(CSMINTSW) of FLAGWRD3 = 1 (Tag here "STATEUP")

Set bit 8(PRECIFLG) of FLAGWRD3 = 0

Perform "INTEGRV"

$T_{decl} = T_{etcm}$

Perform "INTSTALL"

Set bit 3(CSMINTSW) of FLAGWRD3 = 0

Set bit 5(STATEFLG) of FLAGWRD3 = 1

Set bits 4(CONICINT) and 1(WMATINT) of FLAGWRD3 = 0

If bit 1(RENDWFLG) of FLAGWRD5 = 1:

Set bit 1(WMATINT) of FLAGWRD3 = 1

Set bit 8(PRECIFLG) of FLAGWRD3 = 1

Perform "INTEGRV"

Set bit 1(NODOV37) of FLAGWRD2 = 0

Proceed to "ENDINT"

ENDINT

Set bit 5(STATEFLG) of FLAGWRD3 = 0

Set restart group 2 to phase 3 (2.3, causing "STATEINT" to be called)

Call "STATEINT" in K_{600sc} seconds

End of job

CSMPREC

Perform "INTSTALL"

IRETURN = Return address (to routine calling "CSMPREC")

Set bit 3(CSMINTSW) of FLAGWRD3 = 1

Set bit 8(PRECIFLG) of FLAGWRD3 = 1

Set bits 4(CONICINT) and 1(WMATINT) of FLAGWRD3 = 0

Proceed to second line of "INTEGRV"

LEMPREC

Perform "INTSTALL"

IRETURN = Return address (to routine calling "LEMPREC")

Set bit 3(CSMINTSW) of FLAGWRD3 = 0

Proceed to 4th line of "CSMPREC"

CSMCONIC

Perform "INTSTALL"

IRETURN = Return address (to routine calling "CSMCONIC")

Set bit 3(CSMINTSW) of FLAGWRD3 = 1

Set bit 1(WMATINT) of FLAGWRD3 = 0

Set bit 4(CONICINT) of FLAGWRD3 = 1

Proceed to second line of "INTEGRV"

LEMCONIC

Perform "INTSTALL"

IRETURN = Return address (to routine calling "LEMCONIC")

Set bit 3(CSMINTSW) of FLAGWRD3 = 0

Proceed to 4th line of "CSMCONIC"

INTEGRVS

Set bit 8(PRECIFLG) of FLAGWRD3 = 1

IRETURN = Return address (to routine calling "INTEGRVS")

TDELTAV = 0

TNUV = 0

Perform "RECTIFY"

Set bit 1(WMATINT) of FLAGWRD3 = 0

Set bit 13(NEWIFLG) of FLAGWRD8 = 1

Proceed to "ALOADED"

INTSTALL

QPRET = Return address (since enter in interpretive language)

If bits 14(INTINUSE) and 13(INTGRAB) of FLGWRD10 $\neq 00_2$:

Put present job to sleep (starting address id = 2nd line of "INTSTALL", i.e. check of bits)

Set bit 14(INTINUSE) of FLGWRD10 = 1

Proceed to address specified by QPRET (of present job, of course)

INTWAKE

If bit 13(INTGRAB) of FLGWRD10 = 1: (means restarted)

TBASE2 = QPRET of present job

Set restart group 2 to resume computations at next step

QPRET (of present job) = TBASE2

If bit 13(INTGRAB) of FLGWRD10 = 0: (i.e. got a restart)

Proceed to address specified by QPRET (of present job)

Awaken all jobs with starting address id = 2nd line of "INTSTALL" (Tag here "INTWAKEO")

Set bits 14(INTINUSE) and 13(INTGRAB) of FLGWRD10 = 0

Proceed to address specified by QPRET (of present job)

AVETOMID (Entered from "AVGEND")

EGRESS = Return address

If bit 1(RENDWFLG) of FLAGWRD5 = 1:

$TS_1 = T_{pptm}$ (holds in MPAC) (tag here "INT/W")

Perform "INTSTALL"

Set bit 1(WMATINT) of FLAGWRD3 = 1

Set bit 1(AVEMIDSW) of FLAGWRD9 = 1

Set bit 3(CSMINTSW) of FLAGWRD3 = 1

$T_{dec1} = TS_1$

Perform "INTEGRV"

X2 = 0 (tag here "SETCOAST")

Perform "INTSTALL"

Set bit 3(CSMINTSW) of FLAGWRD3 = 0

$\underline{R}_{rect} = \underline{R}$

$\underline{RCV} = \underline{R}_{rect}$

$T_{dec1} = T_{pptm}$

$T_{et} = T_{pptm}$

$\underline{V}_{rect} = \underline{V}$

$VC\underline{V} = \underline{V}_{rect}$

$TDELTA\underline{V} = 0$

$TNU\underline{V} = 0$

$T_c = 0$

XKEP = 0 (notation also "XPREV")

Perform "MOVEACSM"

Set bit 5(STATEFLG) of FLAGWRD3 = 1

Set bits 4(CONICINT) and 1(WMATINT) of FLAGWRD3 = 0

Perform "INTEGRV"

Proceed to address specified by EGRESS

MIDTOAV1 R41 entrance from "P40S/SV" (P40/P41)

IRETURN1 = Return address

Perform "INTSTALL"

Set bit 3(MID1FLAG) of FLAGWRD9 = 1

$TS = T_{now} + K_{timedt}$

If $(T_{decl} - TS) < 0$: (i.e. desired time less than K_{timedt} from now)

Set bit 3(MID1FLAG) of FLAGWRD9 = 0

IRETURN1 = IRETURN1 + 1

Perform "ALARM" (pattern 1703_8)

$TTOADD = K_{timedt}$

Proceed to 5th line of "MIDTOAV2"

Proceed to 6th line of "MIDTOAV2"

MIDTOAV2 R41 entrance from "P47CSM" (P47/P48) and "S61.1" (P61/P62)

IRETURN1 = Return address

Perform "INTSTALL"

Set bit 3(MID1FLAG) of FLAGWRD9 = 0

TTOADD = K_{timedt}

$T_{decl} = T_{now} + TTOADD$ (tag here "ENTMID2")

Set bit 1(WMATINT) of FLAGWRD3 = 0 (tag here "ENTMID1")

Set bit 3(CSMINTSW) of FLAGWRD3 = 1

Set bit 4(CONICINT) of FLAGWRD3 = 0

Set bit 2(MIDAVFLG) of FLAGWRD9 = 1 (causes "DIFEQ+2" to transfer to "CKMID2")

Perform "INTEGRV"

$\underline{R}_{n1} = \underline{R}_{att}$

$\underline{V}_{n1} = \underline{V}_{att}$

$T_{pptm1} = T_{att}$

Inhibit interrupts

$TS = T_{pptm1} - T_{now}$, with sign agreement forced

Proceed to address specified by IRETURN1

CKMID2 Entered from "DIFEQ+2" if bit 2 of FLAGWRD9 = 1

If bit 3(MID1FLAG) of FLAGWRD9 = 0:

If $|T_{dec} - T_{et}| - K_{3csecs} < 0$:

If $(T_{et} - T_{now} - K_{5p6secs}) \geq 0$, proceed to "A-PCHK"

TTOADD = TTOADD + K_{timedt}

$T_{dec} = T_{now} + TTOADD$

Proceed to "TESTLOOP"

$TS = T_{now} + K_{timedt}$

If $T_{dec} - TS \gg 0$:

Proceed to "TESTLOOP"

Set bit 3(MIDlFLAG) of FLAGWRD9 = 0

IRETURN1 = IRETURN1 + 1

Perform "ALARM" (pattern 1703_8)

$TTOADD = K_{timedt}$

$T_{dec} = T_{now} + TTOADD$

Proceed to "TESTLOOP"

MOVEACSM

$\underline{R}_{rectcm} = \underline{R}_{rect}$

$\underline{V}_{rectcm} = \underline{V}_{rect}$

$T_{etcm} = T_{et}$

$DELTA\underline{V}_{cm} = TDELTA\underline{V}$

$NU\underline{V}_{cm} = TNU\underline{V}$

$RC\underline{V}_{cm} = RC\underline{V}$

$VC\underline{V}_{cm} = VC\underline{V}$

$T_{ccm} = T_c$

$XKEP_{cm} = XKEP$ (notation also "XPREV")

Return

MOVEALEM

$\underline{R}_{rectlm} = \underline{R}_{rect}$

$\underline{V}_{rectlm} = \underline{V}_{rect}$

$T_{etlm} = T_{et}$

$\underline{DELTAV}_{lm} = TDELTA\underline{V}$

$NU\underline{V}_{lm} = TNU\underline{V}$

$RC\underline{V}_{lm} = RC\underline{V}$

$VC\underline{V}_{lm} = VC\underline{V}$

$T_{clm} = T_c$

$XKEP_{lm} = XKEP$ (notation also "XPREV")

Return

MOVEPCSM

$\underline{R}_{rect} = \underline{R}_{rectcm}$

$\underline{V}_{rect} = \underline{V}_{rectcm}$

$T_{et} = T_{etcm}$

$TDELTA\underline{V} = \underline{DELTAV}_{cm}$

$TNU\underline{V} = NU\underline{V}_{cm}$

$RC\underline{V} = RC\underline{V}_{cm}$

$VC\underline{V} = VC\underline{V}_{cm}$

$T_c = T_{ccm}$

$XKEP = XKEP_{cm}$ (XKEP notation also "XPREV")

Return

MOVEPLEM

$\underline{R}_{rect} = \underline{R}_{rectlm}$

$\underline{V}_{rect} = \underline{V}_{rectlm}$

$T_{et} = T_{etlm}$

$TDELTA\underline{V} = \underline{DELTAV}_{lm}$

$TNU\underline{V} = NU\underline{V}_{lm}$

$RC\underline{V} = RC\underline{V}_{lm}$

$VC\underline{V} = VC\underline{V}_{lm}$

$T_c = T_{clm}$

$XKEP = XKEP_{lm}$ (XKEP notation also "XPREV")

Return

INTEGRV

IRETURN = Return address (to routine calling "INTEGRV")

Set bit 13(NEWIFLG) of FLAGWRD8 = 1

If bit 3(CSMINTSW) of FLAGWRD3 = 1:

Perform "MOVEPCSM"

Proceed to "ALOADED"

Perform "MOVEPLEM"

Proceed to "ALOADED"

ALOADED

Reset overflow indicator

$T_{dec} = T_{decl}$

If bit 4(CONICINT) of FLAGWRD3 = 0, proceed to "TESTLOOP"

$TAUORB = T_{dec} - T_{et}$

Perform "RECTIFY"

Perform "KEPPREP"

$T_{et} = T_{et} + T_{c}$

Proceed to "RECTOUT"

A-PCHK

If bit 5(STATEFLG) of FLAGWRD3 = 0, proceed to "RECTOUT"

Set bit 5(STATEFLG) of FLAGWRD3 = 0

Set bit 13(INTGRAB) of FLGWRD10 = 1 (means restart phase change)

If bit 3(CSMINTSW) of FLAGWRD3 = 1:

Perform "MOVEACSM"

If bit 1(AVEMIDSW) of FLAGWRD9 = 0:

$\underline{R} = RC\underline{V} + TDELTA\underline{V}$

$\underline{V} = VC\underline{V} + TNU\underline{V}$

$T_{pptm} = T_{et}$

Proceed to "RECTOUT"

Perform "MOVEALEM" (bit 3 of FLAGWRD3 = 0, meaning OWS integration)

$\underline{R}_{other} = RC\underline{V} + TDELTA\underline{V}$

$\underline{V}_{other} = VC\underline{V} + TNU\underline{V}$ Time tag is in T_{etlm}

Proceed to "RECTOUT"

RECTOUT

Perform "RECTIFY"

$\underline{R}_{att} = \underline{R}_{rect}$	Push-down list 0D-5D
$\underline{V}_{att} = \underline{V}_{rect}$	6D-11D
$T_{att} = T_{et}$	12D-13D
$\underline{R}_{att1} = \underline{R}_{rect}$	14D-19D
$\underline{V}_{att1} = \underline{V}_{rect}$	20D-25D
$MUpPp = K_{mutab_0}$	26D-27D

X1 = -2

Proceed to "INTEXIT"

INTEXIT

Reset overflow indicator

Set bits 2(MIDAVFLG) and 1(AVEMIDSW) of FLAGWRD9 = 0

Set bits 5(STATEFLG) and 8(PRECIFLG) of FLAGWRD3 = 0

X2 = 0

QPRET = IRETURN

Proceed to "INTWAKE" (returns to address in QPRET)

TESTLOOP

If bit 5(QUITFLAG) of FLAGWRD9 = 1: (set by "VERB96"; reset by "STATINT1")

Set bit 5(STATEFLG) of FLAGWRD3 = 0

Proceed to "INTEXIT"

$TS_1 = K_{p3d}\sqrt{|\underline{RCV}|^3 / K_{mutab_0}}$ (scaled B28 in units of centi-seconds)

$TS_2 = TS_1$ shifted right 7 places (giving scale factor B35, or least increment of 128 centi-seconds, with low-order bits of TS_1 lost)

$TS_3 = TS_2$ shifted left 15 places (giving scale factor B20 centi-seconds)

If $K_{dt2mx} - TS_3 < 0$, or if overflow taken place:

$TS_3 = K_{dt2mx}$ (TS_3 in push-down list location 12D)

$TS = T_{dec} - T_{et}$, with sign agreement forced

TS = TS, shifted left 8 places (scale factor B20, units centi-sec)

If $|TS| - TS_3 \geq 0$, or if overflow has taken place:

$TS = TS_3$ sgn TS

If $|TS| - K_{dt2mn} < 0$:

Proceed to "A-PCHK"

$DTd2 = \frac{1}{2} TS$ (accomplished by considering scaling of DTd2 B19, whereas TS is scaled B20)

If bit 9(POOFLAG) of FLAGWRD3 = 0, proceed to "TIMESTEP"

If bit 8(PRECIFLG) of FLAGWRD3 = 1, proceed to "TIMESTEP" (not CSM from "STATINT1")

If $TS - TS_3 < 0$, proceed to "A-PCHK" (equals case does <u>not</u> transfer)

If bit 13(NEWIFLG) of FLAGWRD8 = 0, proceed to "TIMESTEP"

Set bit 13(NEWIFLG) of FLAGWRD8 = 0 (if get here, came from "STATINT1")

$TS_4 = T_{dec} - T_{et}$ (scale factor B28, units centi-seconds)

If $TS_4 < 0$, proceed to "INTEXIT" (no backwards integration)

TS_5 = TS, shifted right 6 places (rounded shift, scale factor B26)

If $TS_4 - 4\,TS_5 < 0$, proceed to "INTEXIT" ("4" because of scaling)

Proceed to "TIMESTEP" (at least 4 time steps behind)

RECTIFY

$\underline{R}_{rect} = RC\underline{V} + (TDELTA\underline{V}$, shifted right 7 places)

$RC\underline{V} = \underline{R}_{rect}$

$\underline{V}_{rect} = VC\underline{V} + (TNU\underline{V}$, shifted right 4 places)

$VC\underline{V} = \underline{V}_{rect}$

$TDELTA\underline{V} = 0$

$TNU\underline{V} = 0$

$T_c = 0$

XKEP = 0 (notation also "XPREV")

Return

TIMESTEP

If $|TDELTA\underline{V}| - K_{3d4} \geq 0$, or if overflow taken place:

Perform "RECTIFY"

Proceed to "INTGRATE"

If $|TDELTA\underline{V}|/|RC\underline{V}| - K_{recrat} \geq 0$: ($|TDELTA\underline{V}|$ shifted right 7 places)

Perform "RECTIFY"

Proceed to "INTGRATE"

If $|TNU\underline{V}| - K_{3d4} \geq 0$, or if overflow has taken place:

Perform "RECTIFY"

Proceed to "INTGRATE"

(Note that sensing overflow resets overflow indicator in all cases)

Proceed to "INTGRATE"

INTGRATE

Set bit 14(JSWITCH) of FLAGWRD0 = 0

$Z\underline{V} = TNU\underline{V}$

$Y\underline{V} = TDELTA\underline{V}$

DIFEQCNT = 0

$ALPHA\underline{V} = Y\underline{V}$

H = 0

Proceed to "ACCOMP"

ACCOMP

$\underline{FV} = 0$

$\underline{BETAV} = \underline{RCV} + \underline{ALPHAV}$ (ALPHAV shifted right 7 places)

If bit 1(WMATINT) of FLAGWRD3 = 1:

$\underline{VECTAB}_{\text{-DIFEQCNT}} = \underline{BETAV}$

$ALPHAM = |\underline{ALPHAV}|$

$\underline{ALPHAV} = \text{unit}\underline{ALPHAV}$

Perform "GAMCOMP"

$\underline{ALPHAV} = \underline{BETAV}$

ALPHAM = BETAM

Proceed to "OBLATE"

GAMCOMP

$BETAM = |\underline{BETAV}|$

$\underline{BETAV} = \text{unit}\underline{BETAV}$

RHO = ALPHAM / BETAM computed in quasi-floating point fashion, using K_{ascale_0} for scaling information

$LIIQ = RHO\ (RHO - 2\ \underline{ALPHAV} \cdot \underline{BETAV})$ (same as $(\underline{A} - 2\ \underline{B}) \cdot \underline{A}/\underline{B}^2$ for non-unit vectors)

$$FOFQ = LIIQ\ \frac{3 + 3\ LIIQ + LIIQ^2}{1 + (1 + LIIQ)^{3/2}}$$

$\underline{TS} = \underline{ALPHAV} + (FOFQ/RHO)\ \underline{BETAV}$ (scaled B4)

$$\underline{TS}_1 = -K_{\text{mutab}_0}\ \frac{RHO}{BETAM^2\ (1 + LIIQ)^{3/2}}\ \underline{TS}$$ (computed quasi-floating point)

Reset overflow indicator

$\underline{FV} = \underline{FV} + \underline{TS}_1$ (using K_{ascale_0} and K_{ascale_1} for scaling data)

If overflow has taken place, proceed to "GOBAQUE"

Return

GOBAQUE Entered if overflow from "DIFEQ+2", "GAMCOMP", and "OBLATE"

If $|\text{TDELTA}\underline{V}| = 0$: (all components below 2 meters)

Proceed to "POODOO" (pattern 20430_8)

TAUORB = $T_c - H$

$T_{et} = T_{et} - H$

Perform "KEPPREP"

Perform "RECTIFY"

Proceed to "TESTLOOP"

OBLATE

COSPHI = $\text{ALPHA}\underline{V} \cdot \underline{C}_{unitw}$

$P_2' = 3\ \text{COSPHI}$

$P_3' = \frac{1}{2}\ (15\ \text{COSPHI}^2 - 3)$

$P_4' = (1/3)\ (7\ P_3'\ \text{COSPHI} - 4\ P_2')$

$P_5' = \frac{1}{4}\ (9\ P_4'\ \text{COSPHI} - 5\ P_3')$

$$\underline{TS} = \left(P_3' + \frac{K_{j3j2}}{\text{ALPHAM}}\left(P_4' + \frac{K_{j4j3}}{\text{ALPHAM}}\ P_5'\right)\right)\ \text{ALPHA}\underline{V} \qquad \text{(scaled B6)}$$

$$\underline{TS} = \underline{TS} - \left(P_2' + \frac{K_{j3j2}}{\text{ALPHAM}}\left(P_3' + \frac{K_{j4j3}}{\text{ALPHAM}}\ P_4'\right)\right)\ \underline{C}_{unitw}$$

$$\underline{TS} = \frac{K_{j2}}{\text{ALPHAM}^4}\ \underline{TS} \qquad \text{computed quasi-floating point}$$

Reset overflow indicator

$\underline{TS} = \underline{FV} + \underline{TS}$

If overflow indicator set, proceed to "GOBAQUE"

$\underline{FV} = \underline{TS}$

Proceed to "NBRANCH"

NBRANCH

If DIFEQCNT = -24, proceed to "DIFEQ+2"

If DIFEQCNT = 0:

$\underline{PHIV} = \underline{FV}$

If DIFEQCNT = -12:

$\underline{PSIV} = \underline{PHIV} + 4\ \underline{FV}$

$\underline{PHIV} = \underline{PHIV} + 2\ \underline{FV}$

H = H + DTd2

DIFEQCNT = DIFEQCNT - 12

$\underline{ALPHAV} = \underline{YV} + H\ (\underline{ZV} + \frac{1}{2}\ H\ \underline{FV})$

If bit 14(JSWITCH) of FLAGWRDO = 1, proceed to "DOW.."

TS = DTd2, shifted right 9 places ($\frac{1}{2}$ interval least increment 1 cs)

If DIFEQCNT = -24:

Round TS to double precision (otherwise, a truncated shift)

$TAUORB = T_c + TS$

$T_{et} = T_{et} + TS$

Perform "KEPPREP"

Proceed to "ACCOMP"

KEPPREP (Entered from "ALOADED", "GOBAQUE", and "NBRANCH")

$A_5 = \frac{1}{2}\ \text{unit}\underline{RCV} \cdot \underline{VCV}$

$QARG = (TAUORB - T_c)/|\underline{RCV}|$

$TS = (1/6)\ QARG^2\ (K_{mutab_0} - |\underline{VCV}|^2\ |\underline{RCV}|)/|\underline{RCV}|$

$XKEPNEW = XKEP + \sqrt{K_{mutab_0}}\ QARG\ (1 - A_5\ QARG + 2(A_5\ QARG)^2 + TS)$

(XKEP also has notation "XPREV")

KEPRTN = Return address (to routine entering "KEPPREP")

Proceed to "KEPLERN" (with overflow indicator reset just before adding the "1" factor to other terms)

DIFEQ+2

$\underline{YV} = \underline{YV} + H(\underline{ZV} + PHI\underline{V}\ H/6)$

$\underline{ZV} = \underline{ZV} + H(PSI\underline{V} + \underline{FV})/6$

If bit 14(JSWITCH) of FLAGWRD0 = 1:

$\underline{W}_{27\text{-COLREG}} = \underline{ZV}$ (shifted left 3 places, compensating for "NEXTCOL" right shift)

$\underline{TS} = \underline{YV}$ (shifted left 3 places)

If overflow indicator set:

Set bit 1(WMATINT) of FLAGWRD3 = 0

Set bit 1(RENDWFLG) of FLAGWRD5 = 0

Set bit 5(STATEFLG) of FLAGWRD3 = 1

Perform "ALARM" (pattern 0421_8)

Proceed to "TESTLOOP"

$\underline{W}_{\text{-COLREG}} = \underline{TS}$

If COLREG $\geq$ 0:

$T_{decl} = T_{dec}$

Proceed to third line of "INTEGRV"

COLREG = COLREG +3

Proceed to "NEXTCOL"

If overflow indicator set, proceed to "GOBAQUE" (sensing it resets)

$TNU\underline{V} = \underline{ZV}$

$TDELTA\underline{V} = \underline{YV}$

If bit 2(MIDAVFLG) of FLAGWRD9 = 1, proceed to "CKMID2"

If bit 1(WMATINT) of FLAGWRD3 = 0, proceed to "TESTLOOP"

Set bit 13(INTGRAB) of FLGWRD10 = 1

If bit 3(CSMINTSW) of FLAGWRD3 = 1:

Perform "MOVEACSM"

If bit 1(AVEMIDSW) of FLAGWRD9 = 0:

$\underline{R} = \underline{RCV} + \underline{TDELTAV}$

$\underline{V} = \underline{VCV} + \underline{TNUV}$

$T_{pptm} = T_{et}$

If bit 3(CSMINTSW) of FLAGWRD3 = 0:

Perform "MOVEALEM"

$\underline{R}_{other} = \underline{RCV} + \underline{TDELTAV}$

$\underline{V}_{other} = \underline{VCV} + \underline{TNUV}$ (T_{etlm} is time tag)

Set bit 14(JSWITCH) of FLAGWRD0 = 1

COLREG = -15

Proceed to "NEXTCOL"

NEXTCOL

$\underline{YV} = \underline{W}_{-COLREG}$ (shifted right 3 places)

$\underline{ZV} = \underline{W}_{27-COLREG}$ (shifted right 3 places)

DIFEQCNT = 0

$\underline{ALPHAV} = \underline{YV}$

H = 0

Proceed to "DOW.."

DOW..

$\underline{TS} = \underline{VECTAB}_{-DIFEQCNT}$

$$\underline{FV} = \frac{3\ (\underline{ALPHAV} \cdot \underline{unitTS})\ \underline{unitTS} - \underline{ALPHAV}}{|\underline{TS}|^3}\ K_{mutab_0}$$

Proceed to "NBRANCH"

Quantities in Computations

See also list of major variables and list of routines

A_5: Intermediate quantity used in "KEPPREP", scale factor B7, stored in push-down list location 4D (corresponds to gamma times square root of mu).

ALPHAM: Value of magnitude of ALPHAV (before ALPHAV made a unit vector), same scaling and units.

ALPHAV: Cell used for several purposes. When initially enter "ACCOMP", contains position deviation scaled B22 meters. Subsequently loaded with unit(ALPHAV), scale factor B1. Is then set to unit position vector (originally BETAV information), with scaling for ALPHAM correspondingly B29, and retains this setting when "OBLATE" entered. When W matrix computations are done, is used to contain value of YV (W-COLREG) information. Can also be used as internal communication cells with coordinate transformation routines.

BETAM: Value of magnitude of BETAV (before BETAV made a unit vector), same scaling and units.

BETAV: Vector giving vehicle position with respect to body whose acceleration effect is being computed, units meters. In "ACCOMP", initial scale factor B29.

C unitw: See General Program Control.

COLREG: Single precision cell ("column register") used to control the indexing in "DIFEQ+2" and "NEXTCOL" for the appropriate number of elements of the W matrix, scale factor B13 (since W matrix elements double precision).

COSPHI: Argument for "OBLATE" equations, program notation "COSPHI/2", scale factor B1, stored in push-down list location 18D. It is the cosine of the angle between the unit polar vector and the position vector.

DELTAV cm, DELTAV lm: "Permanent" values of TDELTAV for CSM and OWS respectively, same scaling as TDELTAV. Program notation DELTACSM and DELTALEM.

DIFEQCNT: Single precision cell, scale factor B14, used for program control purposes, having values 0, -12, and -24 at the beginning, middle, and end of each integration step. It is negative for convenience in using interpreter indexing orders.

DTd2: Value of limited time increment for orbital integration, scale factor B19, units centi-seconds (is actually 1/2 the value of the net increment, e.g., beginning to middle or middle to end). It is derived in "TESTLOOP".

EGRESS: Single precision cell used to retain return address information (e.g., from "AVETOMID").

FOFQ: Function of LILQ used in "GAMCOMP", scale factor (for FOFQ/RHO) B3. Program does not actually divide by RHO, but instead computes FOFQ/RHO directly: FOFQ computation shown as given, however, to facilitate comparison with published equation information.

FV: Value of disturbing acceleration (second time derivative of position deviation), scale factor B-16 in meters/cs^2. Same cell used in "DOW.." for W-matrix updating, where scale factor is 38 less than ALPHAV scale factor.

H: Value of time since beginning of integration step, scale factor B19, units centi-seconds.

IRETURN: Single precision cell used to retain return address information from the subroutine entered to use the orbital integration package (after "INTSTALL" logic has given control of the package to the routine used).

IRETURN1: Single precision cell used to retain return address information from "MIDTOAV1" and "MIDTOAV2". If "MIDTOAV1" is entered (meaning that integration to a specified time is desired) but insufficient time is available to do the integration, then the time is slipped and IRETURN1 is incremented by 1 (to return to calling address +2) for appropriate disposition by caller. If enter "MIDTOAV2", cell is not incremented.

KEPRTN: See Conic Routines.

K_{3csecs}: Constant, program notation "3CSECS", scale factor B28, units centi-seconds, used as exit tolerance from "CKMID2". Value is 3 x 2^{-28}, corresponding to 0.03 seconds (meaning an exit if time difference is 0-2 centi-seconds).

K_{3d4}: Constant, program notation "3/4", used in "TIMESTEP" to check if rectification is required. Value corresponds to 0.75, with scaling same as that of the vector with which compared. Hence will rectify if TDELTAV magnitude is 0.75 x 2^{22} meters or more, or if TNUV magnitude $\geq$ 0.75 x 2^{3} meters/csec.

$K_{5p6secs}$: Constant, program notation "5.6SECS", scale factor B28, units centi-seconds. Value is 560 x 2^{-28}, corresponding to 5.6 seconds (to ensure that calling routine can successfully blank DSKY for 5 seconds).

K_{600sc}: Constant, program notation "600SECS", scale factor B28, units centi-seconds. Value is 60000 x 2^{-28}, corresponding to 600 seconds.

K_{ascale_i}: Set of single precision constants used to control performance of "GAMCOMP" as selected by the value of index. Program notation for i = 0 is "ASCALE". Value for i = 0 gives the difference between ALPHAM and BETAM scalings, for use in determining the proper scaling of RHO:

i	Value	ALPHAM scale	BETAM scale	
0	-7	22	29	(earth, primary)

Value for i = 1 is used to determine (with RHO scaling information above) the proper scaling of FV information: B-16. The value is (16 - 2 x BETAM scaling + mu scaling): the "2" factor is required because $BETAM^2$ is in computations.

i	Value	2 x BETAM scale	Mu scale	
1	-6	58	36	(earth, primary)

K_{dt2mn}: Constant, program notation "DT/2MIN", scale factor B20, units centi-seconds. Value is 3 x 2^{-20}, corresponding to 0.03 seconds (hence "TESTLOOP" exits, since original input B28 centi-seconds, if time difference is 0-2 centi-seconds). Could also be considered a minimum for DTd2, in which case value would correspond (scale factor B19) to 0.015 seconds.

K_{dt2mx}: Constant, program notation "DT/2MAX", scale factor B20, units centi-seconds. Value is 4000E2 x 2^{-20}, corresponding to 4000 seconds. Could also be considered a maximum of DTd2, in which case value would correspond (scale factor B19) to 2000 seconds.

K_{j2}: Constant, program notation "J2REQSQ", scale factor B72. Value is 1.75501139E21 x 2^{-72}, corresponding to 3.986032E10 x $(6.378165E6)^2$ x (1082.3E-6) x 2^{-72}, where first term is earth μ (in $meters^3/cs^2$), second is square of earth's radius (gravitational, in meters), and third is second harmonic of earth's potential function (note that 1.5 times it = 1.62345E-3), while 4th is scale factor.

K_{j3j2}: Constant, program notation "2J3RE/J2", scale factor B27. Value is -0.1355426363E5 x 2^{-27}, corresponding to -0.23E-5 x 6.378165E6 / 1082.3E-6 x 2^{-27}, where first term is third harmonic of earth's potential function, second is earth radius in meters, and 3rd is second harmonic, while fourth is scaling.

K_{j4j3}: Constant, program notation "J4REQ/J3", scale factor B26. Value is 0.4991607391E7 x 2^{-26}, corresponding to -1.8E-6 x 6.378165E6 / -0.23E-5 x 2^{-26}, where first term is fourth harmonic of earth's potential function, second is earth radius in meters, 3rd is third harmonic, and fourth is scaling.

K_{mutab_0}: See Conic Routines.

K_{p3d}: Constant, program notation ".3D", scale factor B2, value 0.3 x 2^{-2}.

K_{recrat}: Constant, program notation "RECRATIO", scale factor B0, value 0.01.

K_{timedt}: Constant, program notation "TIMEDELT", scale factor B28, units centi-seconds. Value is 1250 x 2^{-28}, corresponding to 12.5 seconds.

LILQ: Value for argument of FOFQ, scale factor B2, stored in push-down list location 8D.

LOCCTR: See Display Interface Routines.

MUpPp: Value of μ for the earth left in push-down list location 26D by "RECTOUT", scale factor B36. Program notation "MU(P)".

NUV_{cm}, NUV_{lm}: "Permanent" values of TNUV for CSM and OWS respectively, same scaling as TNUV.

P_2', P_3', P_4', P_5': Legendre polynomial derivatives computed in "OBLATE", scale factors B6, B5, B7, and B10 respectively, stored in push-down list locations 0D, 2D, 4D, and 6D (partially) respectively.

PHIV: Storage for information used in Nystrom numerical integration, scaled (for state vector updates) B-13.

PSIV: Storage for information used in Nystrom numerical integration, scaled (for state vector updates) B-13.

QARG: Intermediate quantity used in "KEPPREP", computed in quasi-floating point fashion, with scaling (for numerator and denominator already normalized) B0. It corresponds to s divided by the square root of mu, and is stored in push-down list location 4D. Hence A_5 QARG has scale factor B7.

QPRET: Single precision cell assigned to each job in the interpretive language, which may be used to retain return address information. Cell is preloaded with proper exit address from orbital integration system of routines before transfer to "INTWAKE"; before using the orbital integration package, subroutine "INTSTALL" is performed which returns immediately to the calling routine if the orbital integration package is not already in use by another user (at a lower job priority), and otherwise retains in QPRET the return address to the routine calling "INTSTALL".

$\underline{R}_{nl}$: See General Program Control.

$\underline{R}_{other}$: Position vector of "other" vehicle (i.e., OWS) transmitted on downlink, scale factor B29, units meters.

$\underline{R}_{rect}$: Value of position vector within orbital integration (when last rectification or equivalent function was performed), scale factor B29, units meters.

$\underline{R}_{rectcm}$, $\underline{R}_{rectlm}$: "Permanent" values of $\underline{R}_{rect}$ for CSM and OWS respectively, same scale factor and units as $\underline{R}_{rect}$.

RC$\underline{V}$: Value of conic position vector within orbital integration routine, scale factor B29, units meters. Also used as communication cell with orbital integration package (when entered via "INTEGRVS") to specify the position component of the state vector to be integrated.

RC$\underline{V}_{cm}$, RC$\underline{V}_{lm}$: "Permanent" value of RC$\underline{V}$ for CSM and OWS respectively, same scale factor and units as RC$\underline{V}$.

RHO: Ratio of magnitudes of ALPHA$\underline{V}$ and BETA$\underline{V}$ (before becoming unit vectors) computed in "GAMCOMP", scale factor B1, stored in push-down list location 4D.

T_c: Time (as of start of integration cycle) since last rectification, scale factor B28, units centi-seconds. Set 0 in "RECTIFY", and incremented in Kepler computations (see Conic Routines).

T_{ccm}, T_{clm}: "Permanent" value of T_c for CSM and OWS respectively, same units and scaling.

T_{dec}: Value of "decision time", i.e., final value of time to which state vector must be integrated, scale factor B28, units centi-seconds.

T_{et}: Value of the time associated with the orbital integration routine state vector, scale factor B28, units centi-seconds. It is incremented in "NBRANCH".

T_{etcm}, T_{etlm}: "Permanent" value of T_{et} for CSM and OWS state vectors respectively, same units and scaling. T_{etlm} is called "TETLEM", "TETOTHER", and "T-OTHER" in program.

T_{pptml}: See IMU Computations.

TAUORB: Time within integration cycle since last rectification (or total conic integration time required), scale factor B28, units centi-seconds. Program notation is "TAU.".

TBASE2: Single precision cell used generally for retention of time base information for restart group 2 (see 3420.5-27), for waitlist restarts. Used in "INTWAKE" to permit retention of QPRET value for restart purposes (since QPRET of job involved would be lost if a restart generated, because it generally is in VAC area as described in 3420.5-27).

TDELTA$\underline{V}$: Vector position deviation (from conic) within orbital integration package, scale factor B22, units meters.

TNU$\underline{V}$: Vector velocity deviation (from conic) within orbital integration package, scale factor B3, units meters/centi-second.

TTOADD: Value of time increment used in "CKMID2" if MID1FLAG = 0, program notation "T-TO-ADD", scale factor B28, units centi-seconds. Normal value is the same as K_{timedt}, but it is incremented by that constant if the "CKMID2" check using $K_{5p6secs}$ fails.

$\underline{V}_{nl}$: See General Program Control.

$\underline{V}_{other}$: Velocity vector of "other" vehicle (i.e., OWS) transmitted on downlink, scale factor B7, units meters/centi-second.

$\underline{V}_{rect}$: Value of velocity vector within orbital integration (when last rectification or equivalent function was performed), scale factor B7, units meters/centi-second.

$\underline{V}_{rectcm}$, $\underline{V}_{rectlm}$: "Permanent" value of $\underline{V}_{rect}$ for CSM and OWS respectively, same units and scaling.

VC$\underline{V}$: Value of conic velocity vector within orbital integration routine, scale factor B7, units meters/centi-second. Also used as communication cell with orbital integration package (when entered via "INTEGRVS") to specify the velocity component of the state vector to be integrated.

VCV_{cm}, VCV_{lm}: "Permanent" value of VC$\underline{V}$ for CSM and OWS respectively, same scale factor and units as VC$\underline{V}$.

$VECTAB_i$: Temporary storage for values of position vector of vehicle, loaded in "ACCOMP" and used in "DOW.." for orbital integration updating of W matrix.

$\underline{W}_i$: See Measurement Incorporation.

XKEP: Value of quantity used in previous cycle through Kepler's equation (program notation also "XPREV"), scale factor B17. Set 0 e.g. in "RECTIFY". Units are $\sqrt{\text{meters}}$.

$XKEP_{cm}$, $XKEP_{lm}$: "Permanent" value of XKEP for CSM and OWS respectively, same scaling.

XKEPNEW: "New" value of XKEP computed in "KEPPREP", scale factor B17.

YV: Communication cell with integration routines, set to TDELTAV in "INTGRATE" and also used for W matrix updating.

ZV: Communication cell with integration routines, set to TNUV in "INTGRATE" and also used for W matrix updating.

PROG20

Perform "RO2BOTH"

DBPTC = ADB

Set bit 12(R27FLAG) of FLGWRD11 = 0

Set bit 8(CSMUPDAT) of FLAGWRD1 = 1

Set bit 15(V50N18FL) of FLAGWRD3 = 1 (already 1 from "ISITPOO")

Set bit 5(TRACKFLG) of FLAGWRD1 = 1

OPTION2 = 0

Option 0: OWS/"VECPOINT"
1: Celes.Body/"VECPOINT"
2: Rotation
4: OWS/3 axis
5: Celes.Body/3 axis

If bit 7(AUTOSEQ) of FLGWRD10 = 0:

TS = 00024_8

Proceed to "GOPERF4": if terminate, proceed to "GOTOPOOH"
if proceed, proceed
otherwise, proceed to previous line

If (bits 2-1 of OPTION2) $\neq 00_2$: (e.g. options 1, 2, or 5)

Proceed to "DOV6N78"

UTPIT = $K_{35degang}$ (constant is -35°) (Tag here "PRELOD78")

UTYAW = 0

AZIMANGL = 0

If bit 7(AUTOSEQ) of FLGWRD10 = 1:

Proceed to "P20OPT"

Proceed to "DOV6N78"

DOV6N78

TS = 0678_{vn}

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
if proceed, proceed
otherwise, proceed to previous line

TS = 0679_{vn}

Perform "VNFLASHR": (if terminate, proceed to "GOTOPOOH")
if proceed, skip next 3 lines
otherwise, proceed to previous line

If bit 2 of OPTION2 = 0: (e.g. options 0, 1, 4, or 5)

TS = 101_2 and perform "BLANKET" (R3BLNK and R1BLNK)

End of job

If bit 3 of OPTION2 = 1: (e.g. option 4 or 5) (Tag here "ENDV6N79")

Set bit 8(AZIMFLAG) of FLGWRD11 = 1

Proceed to second line of "P20OPT"

Proceed to "P20OPT"

P20OPT

Set bit 8(AZIMFLAG) of FLGWRD11 = 0

If (bits 2-1 of OPTION2) = 00_2: (e.g. option 0 or 4)

Set bit 9(UTFLAG) of FLAGWRD8 = 0

Set bit 7(UPDATFLG) of FLAGWRD1 = 1

Proceed to "NDUTINPT"

OPTNTYPE = (bits 2-1 of OPTION2) - 1 (Tag here "UTKINPUT")

Set bit 9(UTFLAG) of FLAGWRD8 = 1

Set bit 7(RNDVZFLG) of FLAGWRD0 = 0

If OPTNTYPE > 0: (e.g. option 2)

TS = 0634_{vn} (Tag here "TYPE2IN")

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
if proceed, proceed
otherwise, proceed to previous line

R67TIME = DSPTEM1

Proceed to "NDUTINPT"

Proceed to "V1N70DSP" (e.g. option 1 or 5)

V1N70DSP

TS = 0170_{vn}

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
if proceed, proceed
otherwise, proceed to previous line

If STARCODE < 0: (contrary to e.g. "R51DSP", -0, i.e. 77777_8, is not rejected here)

Set bit 7(Operator Error) of channel 11 = 1

Proceed to "V1N70DSP"

If STARCODE > 47_8:

Set bit 7(Operator Error) of channel 11 = 1

Proceed to "V1N70DSP"

UTSTARNO = (bits 6-1 of STARCODE) (note get 77_8, not defined, for an original input of -0)

If UTSTARNO > 0: (includes original -0 STARCODE input: will be treated as "earth" in "UTAREA1")

Proceed to "NDUTINPT"

TS = 0688_{vn}

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
if proceed, proceed
otherwise, proceed to previous line

PLANVCUT = STARSAV3

Proceed to "NDUTINPT"

NDUTINPT

Set restart group 2 to cause a start at next line (priority 26_8 to PHSPRDT2)

If bit 9(UTFLAG) of FLAGWRD8 = 1:

Establish "STATINT1"(priority 05_8) (Tag here "INTSETUP")

Set restart group 2 to phase 5(i.e. 2.5, causing "STATINT1" to be established with priority 05_8 if a restart)

Set restart group 1 to phase 11(i.e. 1.11, causing "PIKUP20" to be established with priority 10_8 if a restart)

Proceed to "PIKUP20"

MARKTIME = T_{now} (communication cell with "SETINTG")

Perform "SETINTG"

If bit 8(CSMUPDAT) of FLAGWRD1 = 1:

    Set bit 3(CSMINTSW) of FLAGWRD3 = 0

Perform "INTEGRV"

Set restart group 2 to start at next line

Perform "SETINTG"

If bit 1(RENDWFLG) of FLAGWRD5 = 1:

    Set bit 1(WMATINT) of FLAGWRD3 = 1

If bit 8(CSMUPDAT) of FLAGWRD1 = 0:

    Set bit 3(CSMINTSW) of FLAGWRD3 = 0

Perform "INTEGRV"

Set bit 7(RNDVZFLG) of FLAGWRD0 = 1

Establish "R22" (priority 26_8)

Set restart group 2 to phase 7 (i.e. 2.7, causing "R22" to be established with priority 10_8 if a restart)

Set restart group 1 to phase 11 (i.e. 1.11, causing "PIKUP20" to be established with priority 10_8 if a restart)

Perform "AUTOCHK" (returns immediately if bit 7(AUTOSEQ) of FLGWRD10 is zero)

Proceed to "PIKUP20"

PIKUP20 (Entered from "NDUTINPT" and due to restart group 1.11)

Change priority of present job to 14_8

If bit 5(TRACKFLG) of FLAGWRD1 = 0:

    Proceed to "FIXDB"

If bit 13(REFSMFLG) of FLAGWRD3 = 0:

    Proceed to "FIXDB"

R61CNTR = 0

If bit 9(UTFLAG) of FLAGWRD8 = 1:

Proceed to "CALLR6X"

Call "CTASK" in 0.01 second

Set bit 14(R21MARK) of FLAGWRD2 = 1

Perform "R61CSM"

If bit 15(PCMANFLG) of FLGWRD10 = 1:

Proceed to third line of "AUTOCHK"

Proceed to "P20TRACK"

P20TRACK

Set bit 10(LMTRG) of FLAGWRD1 = 1

Perform "R52"

Perform "MKRELEAS"

Proceed to "FIXDB"

PROG21

OPTION2 = 1

Set bit 5(TRACKFLG) of FLAGWRD1 = 1

TS = 00002_8

Proceed to "GOPERF4": if terminate, proceed to "GOTOPOOH"
if proceed, proceed
otherwise, proceed to previous line

DSPTEM1 = 0

Proceed to "P21PROG1"

P21PROG1

TS = 0634_{vn}

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
if proceed, proceed
otherwise, proceed to previous line

TS = DSPTEM1

If TS = 0:

TS = T_{now}

Proceed to "P21PROG2"

P21PROG2

$T_{decl} = TS$

Perform "INTSTALL"

If bit 12(P21FLAG) of FLAGWRD2 = 0: (set 0 e.g. in "INITSUB")

Set bit 3(CSMINTSW) of FLAGWRD3 = 1

If |OPTION2| ≥ 2:

Set bit 3(CSMINTSW) of FLAGWRD3 = 0

Set bits 4(CONICINT) and 1(WMATINT) of FLAGWRD3 = 0

Perform "INTEGRV"

If bit 12(P21FLAG) of FLAGWRD2 = 1:

$\underline{RCV} = \underline{P21BASER}$ (tag here "P21CONT")

$\underline{VCV} = \underline{P21BASEV}$

$T_{et} = P21TIME$

Set bit 1(WMATINT) of FLAGWRD3 = 0 (Note that bit 4 not set.)

Perform "INTEGRVS"

$P21TIME = T_{att}$ (Tag here "P21VSAVE")

$\underline{P21BASER} = \underline{R}_{attl}$

$\underline{P21BASEV} = \underline{V}_{attl}$

If bit 10(NEWTFLAG) of FLAGWRD5 = 1: (i.e. from P29)

Set bit 10(NEWTFLAG) of FLAGWRD5 = 0

Proceed to "HOP29DSP"

If bit 1(P29FLAG) of FLAGWRD0 = 1:

Proceed to "LONGPASS"

$P21VEL = |\underline{P21BASEV}|$ (X2 used to shift, but 0 from "INTEXIT")

$P21GAM = \sin^{-1} \left((unit\underline{R}_{att} \cdot \underline{V}_{att}) / P21VEL \right)$

Set bit 12(P21FLAG) of FLAGWRD2 = 1

$ALPHA\underline{V} = \underline{R}_{att}$

Set bit 13(ERADCOMP) of FLAGWRD1 = 0

$TS = T_{att}$

Perform "LAT-LONG"

$P21ALT = K_{kp01}$ ALT

$TS = 0643_{vn}$

Proceed to "GOFLASH": if terminate, proceed to "GOTOPOOH"
if proceed, proceed to "GOTOPOOH"
otherwise, proceed

$DSPTEM1_{dp} = P21TIME + K_{600sec}$

Proceed to "P21PROG1"

P25CSM

Set bit 4(P25FLAG) of FLGWRD11 = 1

Set bit 7(RNDVZFLG) of FLAGWRD0 = 0

$TS = 0672_{vn}$

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
if proceed, proceed
otherwise, proceed to previous line

Perform "N72STUFF"

Call "READCYCL" in 0.01 second

Call "CTASK" in 0.01 second

Delay 2 seconds (by putting job to sleep via "DELAYJOB")

Proceed to "BOTHSHOW"

72SHOWX

TS = 0672_{vn}

Proceed to "VNFLASH": (if terminate, proceed to "GOTOPOOH")
if proceed, proceed
otherwise, proceed to previous line

Perform "N72STUFF"

Proceed to "BOTHSHOW"

BOTHSHOW Entered from "P25CSM" and "72SHOWX"

If bit 10(NOUNFLG) of FLGWRD11 = 0:

TS = 1676_{vn}

Proceed to "GOFLASH": if terminate, proceed to "GOTOPOOH"
if proceed, proceed
otherwise, proceed to "72SHOWX"

Set bit 10(NOUNFLG) of FLGWRD11 = 1 (tag here "77SHOW")

TS = 1677_{vn}

Proceed to "GOFLASH": if terminate, proceed to "GOTOPOOH"
if proceed, proceed
otherwise, proceed to "72SHOWX"

Proceed to "GOTOPOOH"

READCYCL Called by "P25CSM" and "R27JOB"

Establish "R27JOB" (priority 20_8)

End of task

R27JOB Established by "READCYCL"

Perform "R27GO"

Call "READCYCL" in 2 seconds

End of job

TIMETASK Called by "CTASK"

TFO = FIXTIME - BUFTIME (written over below while still in task)

If TFO $\neq$ 0: (e.g. manual N72 input to change FIXTIME)

Perform "N72STUFF"

Proceed to "CTASK"

CTASK Called by "PIKUP20", "P25CSM", "P47CSM" (P48), and "RELINUS"

Call "TIMETASK" in 1 second

TFO = T_{now} - FIXTIME

If |TFO| > 0.64 second:

End of task

If bit 5(TRACKFLG) of FLAGWRD1 = 0:

End of task

CDUSNAP = CDU

CDUSNAPT = CDUT

CDUSNAPS = CDUS

Establish "PHIJOB" (priority 27_8)

End of task

N72STUFF Entered from "P25CSM", "P48CSM","TIMETASK", and "72SHOWX"

Inhibit interrupts

Set bits 10(NOUNFLG) and 3(TDFLAG) of FLGWRD11 = 0

PHETA = K_{dpm01}

If $FIXTIME_{sp}$ = 0:

Set bit 10(NOUNFLG) of FLGWRD11 = 1

Set bit 9(N77FLAG) of FLGWRD11 = 0

Release interrupts

BUFTIME = FIXTIME

Return

PHIJOB Established by "CTASK"

Delay 0.5 second (by putting job to sleep via "DELAYJOB")

Perform "PHICOMP"

Set bit 6(SNAPFLAG) of FLGWRD11 = 0

End of job

PHICOMP Entered from "PHIJOB" and "THISJAZZ"

TS_2+3 = CDUSNAPS

TS_2+5 = CDUSNAPT

X1 = - "TS_2"

Perform "SXTNB"

VECLOS = TS

If bit 7(FIXFLAG) of FLGWRD11 = 1:

If $(T_{now}$ - FIXTIME) $\geq$ 0:

Set bit 3(TDFLAG) of FLGWRD11 = 1 (Tag here "TDSNAP")

T_{decl} = FIXTIME

Perform "CSMPREC"

Proceed to "PREANGLE"

$T_{decl} = T_{now}$ (Tag here "NOWCOMP")

Perform "CSMCONIC"

Proceed to "PREANGLE"

PREANGLE

$\underline{TS}_1 = \underline{R}_{att}$ (in 18D) (unit$\underline{TS}_1$ loaded into OD)

$\underline{TS}_2 = \underline{V}_{att}$ (in 24D)

CDUSPOT = (CDUSNAP$_y$, CDUSNAP$_z$, CDUSNAP$_x$)

Perform "QUICTRIG"

TS = VECLOS

Perform "ANGLER"

Return

ANGLER Enter with TS_1 r, 18D; TS_2 v, 24D; unitTS_1, OD, from "PREANGLE" and "THISJAZZ".

Perform "TRG*NBSM" starting at 2nd line (cf. "COMPDISP" equations)

$TS = TS\ [REFSMMAT]$ (in 6D)

$TS_3 = \text{unit}\left(TS - (TS \cdot \text{unit}TS_1)\ \text{unit}TS_1\right)$ (in 12D)

$TS_4 = \left(\text{unit}(TS_1 * TS_2) * TS_1\right) \cdot TS_3$

$PHETA = \cos^{-1}\left((TS_3 \cdot TS)\ \text{sgn}\ TS_4\right)$

If $(TS_1 \cdot TS) < 0$: (note V16 monitor could observe the uncorrected value)

$PHETA = (1 - 2^{-28}) - PHETA$

Return

P29

Set bit 1(P29FLAG) of FLAGWRDO = 1

Proceed to "PROG21" ("P21PROG2" exits to "LONGPASS" since P29FLAG is set)

LONGPASS

$TS = 0643_{vn}$

Perform "VNFLASHR": (if terminate, proceed to "GOTOPOOH")
if proceed, skip next 2 lines
otherwise, proceed to previous line

$TS = 101_2$ and perform "BLANKET" (R3BLNK, R1BLNK)

End of job

LONGFOR = LONG (LONG contents destroyed)

Proceed to "HOP29DSP"

HOP29DSP

$TS_1 = \text{unit}Z$

TS_2 = P21TIME (tag also P29BASET)

PASSTIME = TS_2

Set bit 1(P29FLAG) of FLAGWRDO = 1

DELTLONG = 0

Perform "RP-TO-R" (leaves $\underline{TS}$ with Z polar vector)

$\underline{MUSUBE}$ = unit($\underline{TS}$ * P21BASE$\underline{R}$) (P21BASE$\underline{R}$ tag also P29BASE$\underline{R}$)

$\underline{MUSUBC}$ = unit($\underline{TS}$ * $\underline{MUSUBE}$)

$\underline{TS}_1$ = unit(P21BASE$\underline{R}$ * P21BASE$\underline{V}$) (P21BASE$\underline{V}$ tag also P29BASE$\underline{V}$)

$\underline{MUSUBS}$ = unit($\underline{TS}_1$ * P21BASE$\underline{R}$)

$\underline{MUSUBN}$ = $\underline{TS}_1$ sgn($\underline{TS}_1$ · $\underline{TS}$) ($\underline{TS}$ from "RP-TO-R")

$\underline{TS}$ = P21BASE$\underline{R}$

Proceed to "HOPALONG"

HOPALONG

ALPHA$\underline{V}$ = $\underline{TS}$

TS = PASSTIME

Set bit 13(ERADCOMP) of FLAGWRD1 = 0

Perform "LAT-LONG"

LNGERR = (LONGFOR - LONG), modulo 1 revolution

$TS_1 = |LNGERR| - K_{epsilong}$

If $TS_1 < 0$:

Proceed to "PASSOUT"

$TS_2 = TS_1 + K_{twiceeps}$

If $TS_2 \geq 1$: (the 1 is revolutions)

Proceed to "PASSOUT"

If bit 1(P29FLAG) of FLAGWRD0 = 0: (i.e. not first iteration)

If $(TS_2 - \frac{1}{2}) \geq 0$: (Tag here "MODULO")

LNGERR = $(TS_2 - \frac{1}{2} - K_{epsilong} - \frac{1}{2})$ sgn LNGERR

(same as
LNGERR - 1 sgn LNGERR)

If bit 1(P29FLAG) of FLAGWRD0 = 1: (first iteration)

Set bit 1(P29FLAG) of FLAGWRD0 = 0

If LNGERR < 0:

LNGERR = LNGERR + $(1 - 2^{-28})$

FUDGE = K_{fearth}

Reset overflow indicator (Tag here "THETCOMP")

TS = FUDGE LNGERR + DELTLONG

If overflow has taken place: (i.e. $|TS| \geq 1$ rev)

TS = P21TIME + K_{600sec} (Tag here "ADDTEN")

Set bit 10(NEWTFLAG) of FLAGWRD5 = 1

Proceed to "P21PROG2" (exits to "HOP29DSP")

DELTLONG = TS

$\underline{MUSUBD} = \text{unit}\left((\underline{MUSUBE} \cos DELTLONG + \underline{MUSUBC} \sin DELTLONG) * \underline{MUSUBN} \right)$

$ORBDLT = \left(\cos^{-1} (\text{unit}\underline{P21BASER} \cdot \underline{MUSUBD}) \right) \text{sgn} (\underline{MUSUBD} \cdot \underline{MUSUBS})$

SNTH = sin ORBDLT

CSTH = cos ORBDLT

$\underline{RVEC} = \underline{P21BASER}$

$\underline{VVEC} = \underline{P21BASEV}$

Set bit 9(RVSW) of FLAGWRD7 = 0 (new $\underline{R}$, $\underline{V}$ desired)

Perform "TIMETHET"

PASSTIME = T + P21TIME

$\underline{TS} = \underline{TS}_r$ (value from "TIMETHET")

Proceed to "HOPALONG"

PASSOUT

DSPTEM1 = PASSTIME

$TS = 0634_{vn}$

Proceed to "GOFLASH": if terminate, proceed to "GOTOPOOH"
if proceed, proceed
otherwise, proceed to "LONGPASS"

$TS = 0643_{vn}$

Proceed to "GOFLASH": if terminate, proceed to "GOTOPOOH"
if proceed, proceed
otherwise, proceed to "P29"

Proceed to "GOTOPOOH"

See also list of major variables and list of routines

ADB: See Digital Autopilot Interface Routines.

ALPHAV: See Coordinate Transformations.

ALT: See Coordinate Transformations.

AZIMANGL: Value of "rotation angle" used in options 4 and 5 (3-axis) of P20, scale factor BO, units revolutions. It is loaded via R3 of N78, and is also modified in "STARTAUT" when a minimum key rendezvous sequence is started. Cell has no effect unless bit 8(AZIMFLAG) of FLGWRD11 = 1; a "heads up" value is 0^{o} and a "heads down" 180^{o}.

BUFTIME: Value of FIXTIME sampled in "N72STUFF", scale factor B28, units centi-seconds, used in "TIMETASK" to determine if a change in FIXTIME made since previous cycle.

CDUS: See Optics Computations.

CDUSNAP: Value of CDU sampled in "CTASK" or "THISJAZZ" and used in "PREANGLE", same scaling and units as CDU (single precision, B-1 revolutions).

CDUSNAPS: Value of CDUS sampled as described for CDUSNAP, same scaling as CDUS.

CDUSNAPT: Value of CDUT sampled as described for CDUSNAP, same scaling as CDUT.

CDUSPOT: See Coordinate Transformations.

CDUT: See Optics Computations.

CSTH: See Conic Routines.

DBPTC: Single precision value of deadband used for P20 maneuvers, scale factor B-1, units revolutions. It is loaded into ADB in "R61CSM" and "R67START" (unless a value of zero is indicated, in which case the 0.5^{o} deadband is used). It is initialized to the current ADB at the start of "PROG20", and can be loaded via R2 of N79.

DELTLONG: Change in longitude computed in P29 from the time that "HOPALONG" was initiated, scale factor BO, units revolutions. It is the change from the vector reflected in e.g. P21BASER.

FIXTIME: Value of R27 optimization time loaded by N72, scale factor B28, units centi-seconds. A value of zero means no optimization is to be done. It is incremented in "WUNZMOR" after completion of an optimization pass.

FUDGE: Factor used to convert LNGERR information to DELTLONG, scale factor B1. It approximates the ratio of the number of revolutions of the spacecraft with respect to inertial space to the number with respect to a fixed longitude in equal time periods.

$K_{35degang}$: Constant, program notation "35DEGANG", scale factor B0, units revolutions. Value is -0.097222222222, corresponding to -35^{o}. Used in "PROG20" to initialize UTPIT for rendezvous options.

K_{600sec}: Constant, program notation "600SEC", scale factor B28, units centi-seconds. Value is 60000 x 2^{-28}, corresponding to 600 seconds or 10 minutes.

K_{dpm01}: See Burn Control.

$K_{epsilong}$: Constant, program notation "EPSILONG", scale factor B0, units revolutions. Value is 0.2777778E-4, corresponding to 0.01^{o} (the P29 convergence criterion).

K_{fearth}: Constant, program notation "FEARTH", scale factor B1, used to load FUDGE in P29. Value is 1.06666667 x 2^{-1}, where first term is nominal value (i.e. 16/15) and second is scale factor. Corresponds roughly to proper FUDGE value for an earth-orbiting spacecraft with a period of 90 minutes (16 revs per 24 hours).

K_{kp01}: Constant, program notation "K.01", scale factor B0, value 0.01. Used to convert output from "LAT-LONG" from units of meters to units of meters/100, so that DSKY display scale, usually XXXX.X nmi, will be XXXXXb. nmi instead for P21ALT.

$K_{twiceeps}$: Constant, program notation "TWICEEPS", scale factor B0, units revolutions. Value is 0.5555556E-4, corresponding to 0.02^{o}, or twice the value of $K_{epsilong}$.

LNGERR: Value of longitude error in P29, scale factor B0, units revolutions. Stored in push-down list location 0D.

LONG: See Coordinate Transformations.

LONGFOR: Value of desired longitude (entered into R2 of N43) in P29, scale factor B0, units revolutions.

MARKTIME: See Measurement Incorporation.

MUSUBC: Unit vector, scale factor B1, perpendicular to polar vector and MUSUBE, used in P29.

MUSUBD: Unit vector, scale factor B1, used in P29 iteration to compute ORBDLT. Stored in push-down list locations 0D and 6D.

MUSUBE: Unit vector, scale factor B1, in easterly direction, used in P29.

MUSUBN: Unit vector, scale factor B1, in plane defined by angular momentum (and same hemisphere as north polar axis), used in P29.

MUSUBS: Unit vector, scale factor B1, in direction of tangential velocity for P29 computations.

OPTION2: See Display Interface Routines.

OPTNTYPE: Single precision cell, scale factor B14, used to retain information on the P20 option selected: it is set equal to (bits 2-1 of OPTION2) - 1, hence zero for selection of option 1/5(celestial body) and 1 for selection of option #2 (rotation). Not loaded for option 0/4 (rendezvous).

ORBDLT: Required angle change (converted via "TIMETHET" to a time and a new position vector) in the P29 loop, scale factor B0, units revolutions. Stored in push-down list location 0D.

P21ALT: Value of K_{kp01} ALT computed in "P21PROG2" for (optional) display in R1 of N73, scale factor B29, units (meters/100): see K_{kp01}.

P21BASER: Value of P21/P29 "base" vector for position (notation also P29BASER), scale factor B29, units meters. In P21, loaded after completion of integration to specified input time, and used to initialize the integration if bit 12(P21FLAG) of FLAGWRD2 = 1, thus permitting computation time to be saved if it is desired to iterate about a point which is a number of orbital integration time steps removed from the "permanent" CSM or OWS state vector.

P21BASEV: Value of P21/P29 "base" vector for velocity (notation also P29BASEV), scale factor B7, units meters/centi-second. See P21BASER.

P21GAM: Value of flight path angle computed in "P21PROG2" for (optional) display in R3 of N73 in P21, scale factor B0, units revolutions.

P21TIME: Cell used to retain time information (notation also P29BASET), scale factor B28, units centi-seconds. Used to contain the time tag of P21BASER and P21BASEV, and to permit the incrementing of the time associated with the N34 display.

P21VEL: Value of velocity computed in "P21PROG2" for (optional) display in R2 of N73 in P21, scale factor B7, units meters/centi-second.

PASSTIME: Time associated with current iteration in P29 loop, scale factor B28, units centi-seconds. When convergence criterion is satisfied, information in this cell is loaded in DSPTEM1 for N34 display.

PHETA: Value of angle computed in "ANGLER" (initialized to -0.01° in "N72STUFF"), scale factor B0, units revolutions, displayed in R3 of N77. It is analogous to R3 of N54 (R31) in P48, and to R3 of N53 (R34) in R22; in P25, it remains at -0.01°. Also set to -0.01° in "SNAPPY".

PLANVCUT: Value of STARSAV3 (N88) information sampled in "V1N70DSP" for use in options 1 or 5 of P20 (display generated if N70 = +0). Can be with arbitrary (but consistent) scaling.

R61CNTR: Single precision cell, scale factor B14, used for control of "R61CSM", and as a flag (if negative) that this routine has called R60 (requiring DSKY priority displays for the performance of the maneuver). Positive values cause the cell to be decremented and performance of the remainder of the routine to be bypassed; the setting to 3 causes the routine to be performed every fourth time it is entered, assuming that R61CNTR is not set 0 by the calling routine.

R67TIME: Value of the time at which maneuver is to be started for option 2 of P20, loaded in "P20OPT" from N34 information, scale factor B28, units centi-seconds. Used in "CALLR6X" to determine when "R67" should be called, for OPTNTYPE > 0 (i.e. option 2).

RCV: See Orbital Integration.

RVEC: See Conic Routines.

SNTH: See Conic Routines.

STARCODE: See Inflight Alignment.

STARSAV3: See Inflight Alignment.

T: See Conic Routines.

T_{et}: See Orbital Integration.

TFO: Time from optimization, scale factor B28, units centi-seconds, displayed in R3 of N76. It is updated in "CTASK" (and used in "TIMETASK" for temporary storage purposes).

UTPIT: Value of "pitch angle" used in P20, scale factor B0, units revolutions. It is loaded via R2 of N78, with a value of -35° the normal rendezvous initialization (preferred tracking axis) value.

UTSTARNO: Value of bits 6-1 of STARCODE sampled in "V1N70DSP", scale factor B14, for use in options 1 or 5 of P20. It is single precision.

UTYAW: Value of "yaw angle" used in P20, scale factor B0, units revolutions. It is loaded via R1 of N78, with a value of zero the normal rendezvous initialization: zeros for UTPIT and UTYAW correspond to +X axis tracking.

VCV: See Orbital Integration.

VECLOS: Temporary storage for sighting information in "PHICOMP", scale factor B1, used to compute required angle information (optics line of sight information).

VVEC: See Conic Routines.

GTSCPSS Entered via V37 E 01 E

If bit 12(NODOP01) of FLAGWRD1 = 1:

Proceed to "POODOO" (pattern 21521_8)

Change priority of present job to 20_8 (was established at 13_8)

GEOCOMP1 = 1 (Logic checking this cell not shown)

ldPIPADT = K_{ldppgt}

LENGTHOT = K_{bt8}

1SECXT1 = K_{ld2scx}

PREMTRXC = 1

$PERFDLAY_{dp}$ = 1 (centi-second)

NEWAZMTH = LAUNCHAZ

OLDAZMTH = LAUNCHAZ

Perform "POSN17C"

Perform "IMUZERO" (Tag here is "GEOIMUTT")

Perform "IMUSTALL": if error return, proceed to "SOMERR2"
otherwise, proceed

NDXCTR = 0

$\underline{X}_{dc} = (1, 0, 0)$

$\underline{Y}_{dc} = (0, \sin C_{azmth}, \cos C_{azmth})$

$\underline{Z}_{dc} = (0, -\cos C_{azmth}, \sin C_{azmth})$

Perform "CALCGA" (comes here if MODREG $\neq 3$, since should still be 1 at this point)

Perform "IMUCOARS"

If bit 14(GLOKFAIL) of FLAGWRD3 = 1:

NDXCTR = NDXCTR + 1

Set bit 14(GLOKFAIL) of FLAGWRD3 = 0

Perform "IMUSTALL": if error return, proceed to "SOMERR2"
otherwise, proceed

If NDXCTR $>$ 0:

Proceed to "PIPACHK" (a check of PIPA's)

Perform "IMUFINE"

Perform "IMUSTALL": if error return, proceed to "SOMERR2"
otherwise, proceed

Call "GOESTIMS" in $PERFDLAY_{dp}$ centi-seconds (set to 1 above)

Put present job to sleep (starting address id = "ESTIMS")

GOESTIMS

Awaken job with starting address id = "ESTIMS"

End of task

SOMERR2

Perform "ALARM" (pattern 1601_8)

Set bit 8(IMUSE) of FLAGWRD0 = 0 (tag here is "ENDTEST1")

TS = -0 and perform "NEWMODEX" (blanks program register)

Perform "MKRELEAS"

Proceed to "ENDEXT"

POSN17C

$\underline{X}_{sm}$ = (0, - cos NEWAZMTH, sin NEWAZMTH)

$\underline{Y}_{sm}$ = (0, sin NEWAZMTH, cos NEWAZMTH)

$\underline{Z}_{sm}$ = (-1, 0, 0)

Return

ESTIMS Entered from awakened job via "GOESTIMS", or from end of "ALFLT" for azimuth change

Inhibit interrupts

GTSWTLT1 = TIME1

PIP$\underline{A}$ = 0

Release interrupts

$INTVEC1_y$ = 0

$INTVEC1_z$ = 0

$FILDELV1_y$ = 0

$FILDELVl_z = 0$

$THETA\underline{N} = 0$

GCOMPSW = 0

$GCOM\underline{P} = 0$

$DEL\underline{V} = 0$

$ERVECTO\underline{R} = K_{omegms}(\sin C_{atd}, -\cos C_{atd}, 0)$

$T_{mark} = T_{now}$

$ERCOM\underline{P} = 0$

ERECTIME = LENGTHOT

TS = 02 and perform "NEWMODEX"

Perform "PIPUSE"

LENGTHOT = 9

Proceed to "SLEEPIE"

SLEEPIE

Perform "CHKCOMED"

Perform "SETGWLST"

End of job

CHKCOMED

Inhibit interrupts

If MODREG = 7:

Return

If bit 5(Liftoff complement) of channel 30 = 1:

If bit 5(BKUPLO) of FLAGWRD5 = 0: (set 1 by "LFTFLGON" for a V75E)

Release interrupts

Return

Set priority of present job to 22_8 (was established at that anyhow)

Proceed to "P11"

SETGWLST

Inhibit interrupts

TS = GTSWTLT1 - TIME1

If TS > 0:

TS = TS - 163.83 seconds (should be 163.84)

TS = TS + 1SECXT1

If TS ≤ 0:

TS = 0.04 seconds

Call "ALLOOP" in TS seconds

Return (interrupts released e.g. by End of job)

ALLOOP

GTSWTLT1 = TIME1

Set restart group 5 to cause restart at next line

Set $\underline{DELV}_{sp}$ = $\underline{PIPA}$ and $\underline{PIPA}$ = +0 (no special restart provisions)

Set restart group 5 to cause restart at next line

Establish "ALFLT" (priority 22_8)

End of task

ALFLT

Perform "CHKCOMED"

Perform "1/PIPA"

$\underline{TS}$ = $\underline{DELV}$ $\left[X_{sm}\right]$

$DPIPAY = -TS_y$

$DPIPAZ = TS_z$

$FILDELV1_y = FILDELV1_y + K_{geocl}\,(DPIPAY - FILVELV1_y)$ (Tag here is "ALWAYSG")

$INTVEC1_y = INTVEC1_y + FILDELV1_y$

$FILDELV1_z = FILDELV1_z + K_{geocl}\,(DPIPAZ - FILDELV1_z)$

$INTVEC1_z = INTVEC1_z + FILDELV1_z$

If ERECTIME $\neq$ 0:

$\text{THETAN}_y = \text{THETAN}_y - K_{geoc5}\ (\text{FILDELVL}_z + K_{geoc2}\ \text{INTVECL}_z)$

$\text{THETAN}_z = \text{THETAN}_z - K_{geoc5}\ (\text{FILDELVL}_y + K_{geoc2}\ \text{INTVECL}_y)$

If ERECTIME = 0:

$\text{THETAN}_x = \text{THETAN}_x + K_{pipgyr}\ \text{FILDELVL}_y$

$\text{THETAN}_y = \text{THETAN}_y - K_{geoc3}\ \text{FILDELVL}_z - K_{geoc4}\ \text{INTVECL}_z$

$\text{THETAN}_z = \text{THETAN}_z - K_{geoc3}\ \text{FILDELVL}_y$

If LENGTHOT $>$ 0:

LENGTHOT = LENGTHOT - 1

Proceed to "SLEEPIE"

Perform "CHKCOMED"

If LGYRO $>$ 0: (gyros torquing, e.g. from "1/PIPA" compensation)

Proceed to "SLEEPIE"

$\underline{\text{ERCOMP}} = \underline{\text{ERCOMP}} + \left[X_{sm}\right]\ \underline{\text{THETAN}}$

Note that restart protection from here onward is priority 20_8 (job established at 22_8).

Perform "EARTHR*"

$\underline{\text{THETAN}} = 0$

If PREMTRXC $>$ 0: (set 0 by "AZMTHCG1")

ERECTIME = ERECTIME - 1, limited $\geq$ 0

LENGTHOT = 9

Proceed to "SLEEPIE"

TS = LAUNCHAZ - OLDAZMTH

If TS = 0:

PREMTRXC = 1

ERECTIME = ERECTIME - 1, limited $\geq$ 0

LENGTHOT = 9

Proceed to "SLEEPIE"

PREMTRXC = 1

NEWAZMTH = LAUNCHAZ

$ERCOMP_z$ = TS

Perform "POSN17C" (zero elements not reset 0)

OLDAZMTH = NEWAZMTH

LENGTHOT = K_{bt7}

TS = "ERCOMP"

Perform "IMUPULSE"

Perform "IMUSTALL": if error return, proceed to "SOMERR2"
otherwise, proceed

Proceed to "ESTIMS"

EARTHR*

$TS = T_{now}$

$TS_1 = TS - T_{mark}$

If $TS_1 < 0$:

$TS_1 = TS_1 + 2^{28}$ centi-seconds

$\underline{ERCOMP} = \underline{ERCOMP} + \left[X_{sm}\right] TS_1 \ \underline{ERVECTOR}$

$T_{mark} = TS$

TS = "ERCOMP"

Perform "IMUPULSE"

Perform "IMUSTALL": if error return, proceed to "SOMERR2"
otherwise, proceed

Return

AZMTHCG1 Established by "CHAZFOGC" for V78E, with priority 16_8

DSPTEM1 = NEWAZMTH, converted to single precision twos complement,
scale factor B-1, units revolutions.

Perform "CLEANDSP"

TS = 0629_{vn}

Proceed to "GOFLASH": if terminate, skip next line
if proceed, proceed
otherwise, proceed to previous line

LAUNCHAZ = DSPTEM1, converted to double precision ones complement, scale factor BO, units revolutions

PREMTRXC = 0

Proceed to "PINBRNCH"

GCOMPVER Established by "CKOPTVB" for V65E, with priority 16_g

TS = 03 and perform "NEWMODEX"

$\underline{X}_{dc} = (1, 0, 0)$

$\underline{Y}_{dc} = (0, \sin C_{azmth}, \cos C_{azmth})$

$\underline{Z}_{dc} = (0, -\cos C_{azmth}, \sin C_{azmth})$

Perform "MKRELEAS" (comes here if MODREG = 3, as it should)

Perform the following for i = 1 and then i = 2:

DSPTEM1+2 = i

DSPTEM1+0 = TAZ_i

DSPTEM1+1 = TEL_i

TS = 0530_{vn}

Perform "GODSPRET"

TS = 0641_{vn}

Proceed to "GOFLASH": if terminate, proceed to "GCOMP5"
if proceed, proceed
otherwise, proceed to 3rd previous line

TAZ_i = DSPTEM1+0

TEL_i = DSPTEM1+1

STARCODE = 1

TS = 0

Perform "TARGDRVE"

STARAD = $[X_{sm}]$ (sin TEL_1, - cos TAZ_1 cos TEL_1, sin TAZ_1 cos TEL_1)

STARBD = $[X_{sm}]$ (sin TEL_2, - cos TAZ_2 cos TEL_2, sin TAZ_2 cos TEL_2)

X1 = - "MRKBUF1"

CDUSPOT = E_{2-X1}

Perform "SXTNB" (here if bit 11 (ATMFLAG) of FLAGWRD0 = 0, as it should)

Perform "TRG*NBSM"

LOSVEC = TS

STARCODE = 2 (Tag here "NEXBNKSS")

TS = 6

Perform "TARGDRVE"

X1 = - "MRKBUF1"

CDUSPOT = E_{2-X1}

Perform "SXTNB" (here if bit 11 (ATMFLAG) of FLAGWRD0 = 0, as it should)

Perform "TRG*NBSM"

STARBC = TS

STARAC = LOSVEC

Perform "AXISGEN"

Perform "CALCGTA"

TS = 0693_{vn}

Proceed to "GOFLASH": if terminate, proceed to "GCOMP5"
if proceed, proceed
otherwise, proceed to previous line

ERCOMP = ERCOMP + OGC

Proceed to "GCOMP5"

GCOMP5

TS = 02 and perform "NEWMODEX"

End of job

TARGDRVE

If TS = 0:

$\underline{STAR}$ = (sin TEL_1, - cos TAZ_1 cos TEL_1, sin TAZ_1 cos TEL_1)

If TS = 6:

$\underline{STAR}$ = (sin TEL_2, - cos TAZ_2 cos TEL_2, sin TAZ_2 cos TEL_2)

Perform "SXTANG"

DESOPTS = SAC_{sp}

DESOPTT = PAC_{sp}

Proceed to "RETARG"

RETARG

OPTIND = +0

Perform "SXTMARK"

If MARKINDX > 0: (no mark obtained)

Proceed to "RETARG"

Perform "MKRELEAS"

Return (to routine calling "TARGDRVE")

Quantities in Computations

See also list of major variables and list of routines

ldPIPADT: See IMU Computations.

lSECXTl: Single precision quantity, scale factor Bl4, units centi-seconds, giving required period of computations for "ALLOOP". Set to 0.5 seconds for gyro compassing.

C_{atd}: Erasable memory (double precision) constant, program notation "LATITUDE", scale factor BO, units revolutions. It gives the "local vertical astronomical latitude" of the pad.

C_{azmth}: Erasable memory (double precision) constant, program notation "AZIMUTH", scale factor BO, units revolutions. It gives the "azimuth of the vehicle Z-axis east of north."

CDUSPOT: See Coordinate Transformations.

DESOPTS, DESOPTT: See Optics Computations.

DPIPAi (i = Y,Z): Value of accelerometer output modified for use in gyrocompassing. The y axis of this system is south and the z axis is east (from $[X_{sm}]$). Scale factor of DPIPAi is Bl4, units accelerometer counts.

ERCOMP: Value of gyro compensation to be sent to gyros, scale factor B21, units pulses (or scale factor BO, units revolutions, since one pulse is 2^{-21} revolution). Program notation also "ERCOMPl".

ERECTIME: Single precision length of time to be spent in erection phase, scale factor Bl4, units of five-second gyro compassing cycles. It is set to LENGTHOT in "ESTIMS". Program notation also "ERECTIMl".

ERVECTOR: Earth rotation vector initialized in "ESTIMS", scale factor Bl, units gyro pulses/centi-second.

$FILDELVl_{y,z}$: Filtered velocity in the y and z directions, scale factor Bl4, units accelerometer counts. Here "y" is north.

GCOMP, GCOMPSW: See IMU Computations.

GEOCOMPl: Single precision cell set positive non-zero to indicate that gyro-compassing computations are being performed. The logic that checks this cell is not shown in this writeup. See Testing Routines.

GTSWTLTl: Single precision cell, scale factor Bl4, units centi-seconds, giving the value of TIMEl when "ALLOOP" last entered, and used to construct proper waitlist value for the next call in "SETGWLST".

$INTVEC1_{y,z}$: Summed values of $FILDELV1_y$ and $FILDELV1_z$ respectively, scale factor B14, units accelerometer counts.

K_{ld2scx}: Single precision constant, program notation "1/2SECX", scale factor B14, units centi-seconds. Value is 50×2^{-14}, corresponding to 0.5 second.

K_{ldppgt}: Single precision constant, program notation "1/PIPAGT", scale factor B8, units centi-seconds. Value is 06200_8, corresponding to 0.5 seconds.

K_{bt7}: Single precision constant, program notation "BIT7", scale factor B14, units of five-second gyro compassing cycles. Octal value is 00100_8, corresponding to decimal 64, used to load LENGTHOT, and thence ERECTIME, for a change in input azimuth: value gives $64 \times 5 =$ 320 seconds for vertical erection.

K_{bt8}: Single precision constant, program notation "BIT8", scale factor B14, units of five-second gyro compassing cycles. Octal value is 00200_8, corresponding to decimal 128, used to load LENGTHOT, and thence ERECTIME, at start of P01 (in "GTSCPSS"): value gives $128 \times 5 =$ 640 seconds for vertical erection at start of P02.

K_{geoc1}: Constant, program notation "GEOCONS1", scale factor B0, value 0.1.

K_{geoc2}: Constant, program notation "GEOCONS2", scale factor B0, value 0.005.

K_{geoc3}: Constant, program notation "GEOCONS3", scale factor B7, units gyro pulses/accelerometer pulse. Value is 0.062, corresponding to a "true" value of 7.936.

K_{geoc4}: Constant, program notation "GEOCONS4", scale factor B7, units gyro pulses/accelerometer pulse. Value is 0.0003, corresponding to a "true" value of 0.0384.

K_{geoc5}: Constant, program notation "GEOCONS5", scale factor B7, units gyro pulses/accelerometer pulse. Value is 0.5, corresponding to a "true" value of 64.

K_{omegms}: Constant, program notation "OMEG/MS", scale factor B0, units gyro pulses/centi-second. Value is 0.2433904$8_8$, corresponding approximately to $(1/86164.0932) \times 10^{-2} \times 2^{21}$, where first term is earth rotation period in seconds (used to derive constant), second converts to centi-seconds, and third is number of gyro torquing pulses in one revolution.

K_{pipgyr}: Dummy constant used to indicate change in units between accelerometer pulses and gyro torquing pulses, scale factor B7. Value is 1.00, corresponding to 2^7 or 128.

LAUNCHAZ: Value of desired launch azimuth, scale factor BO, units revolutions. Since is used at the start of PO1, should form part of prelaunch erasable load, although it can also be updated in "AZMTHCG1" (via V78E). Program notation also "LUNCHAZ1".

LENGTHOT: Single precision cell, scale factor B14, loaded in "GTSCPSS" with K_{bt8} (for loading into ERECTIME) and at the end of "ALFLT", if a new azimuth specification received, with K_{bt7}. It is also used as a counter of the number of half-seconds that have elapsed since the previous entrance to "EARTHR*", to control the nominal gyro compassing cycle of five seconds (via an initial setting to 9 and a 1SECXT1 setting to 0.5 seconds).

LGYRO: See IMU Computations.

LOSVEC: Unit vector, scale factor B1, serving in "GCOMPVER" as a temporary storage cell for the sighting vector to the first target.

MARKINDX, MRKBUF1: See Optics Computations.

NDXCTR: Single precision cell, scale factor B14, used in "GTSCPSS" to retain information on bit 14(GLOKFAIL) of FLAGWRD3, which should be 0 if do PO1.

NEWAZMTH: Communication cell with "POSN17C", scale factor BO, units revolutions, used to compute the required orientation of the stable member. Program notation also "NEWAZ1". Set to LAUNCHAZ in "GTSCPSS" and "ALFLT" (if an input azimuth change).

OGC: See Coordinate Transformations.

OLDAZMTH: Value of azimuth angle presently reflected in computations, scale factor BO, units revolutions. If, with PREMTRXC = 0, LAUNCHAZ = OLDAZMTH, it is concluded that no input change took place, and no re-initialization (via entrance to "ESTIMS") of gyro compassing is done. OLDAZMTH is set as described for NEWAZMTH, but at a slightly different time for restart considerations.

OPTIND: See Optics Computations.

PAC: See Coordinate Transformations.

PERFDLAY: Communication cell with routine calling "GOESTIMS", set to one centi-second at start of "GTSCPSS", scale factor B28, units centi-seconds (the "LONGCALL" entrance to the waitlist system is used).

PIPA: See IMU Computations.

PREMTRXC: Single precision cell, scale factor B14, initialized to 1 in "GTSCPSS" and set 0 at the end of "AZMTHCG1". If sensed as 0 in "ALFLT"(when LENGTHOT has run down), then, if LAUNCHAZ $\neq$ OLDAZMTH, gyro compassing is re-initialized. In any event, PREMTRXC is reset to 1. Program notation also "PREMTRX1".

SAC: See Coordinate Transformations.

STAR: See Coordinate Transformations.

STARAC, STARAD, STARBC, STARBD: See Coordinate Transformations.

STARCODE: See Inflight Alignment (setting in "GCOMPVER" is not functional, since the cell is not displayed by the mark routine).

T_{mark}: Value of time when previous earth-rate compensation was made, scale factor B28, units centi-seconds.

TAZ_1, TAZ_2, TEL_1, TEL_2: Single precision erasable memory cells (which could be set as part of erasable memory load) giving azimuth (TAZ_i) and elevation(TEL_i) for targets #1 and #2 used in P03. For azimuth information, scale factor is B-1 in twos complement; for elevation, scale factor is B-2: in both cases, units are revolutions.

THETAN: Value of required number of gyro torquing pulses (before rotation by $[X_{sm}]$, and hence in the vertical, south, east system), scale factor B21, units pulses. Program notation also "THETAN1".

PRECSET

$T_{dec2} = T_{dec1}$

Perform "LEMPREC"

$\underline{R}_{pass3} = \underline{R}_{att}$

$\underline{V}_{pass3} = \underline{V}_{att}$

$T_{dec1} = T_{dec2}$

Perform "CSMPREC"

$\underline{R}_{act3} = \underline{R}_{att}$

$\underline{V}_{act3} = \underline{V}_{att}$

$\underline{UP1} = unit(unit\underline{R}_{pass3} * \underline{V}_{pass3})$

$CMYDOT = \underline{V}_{act3} \cdot \underline{UP1}$

$AUTOY = \underline{R}_{act3} \cdot \underline{UP1}$

$\underline{UNRM} = unit(\underline{R}_{act3} * \underline{V}_{act3})$

$LMYDOT = \underline{V}_{pass3} \cdot \underline{UNRM}$

Return

S33/34.1 Entered from "P33/P73B" (P34) and "P34/P74C" (P35)

TITER = -16383

$SECMAX = K_{max250}$

$\underline{R}_{aprec} = \underline{R}_{act3}$

$\underline{V}_{aprec} = \underline{V}_{act3}$

$\underline{R}_{pprec} = \underline{R}_{pass3}$

$\underline{V}_{pprec} = \underline{V}_{pass3}$

Proceed to "ELCALC"

ELCALC

$\underline{ULOS} = \text{unit}(\underline{R}_{pass3} - \underline{R}_{act3})$

$\underline{UNRM} = \text{unit}(\underline{R}_{act3} * \underline{V}_{act3})$

$\underline{UP} = \text{unit}\left(\underline{ULOS} - (\underline{ULOS} \cdot \text{unit}\underline{R}_{act3})\ \text{unit}\underline{R}_{act3}\right)$

$TS_1 = (\underline{UNRM} * \underline{R}_{act3}) \cdot \underline{UP}$

$TS = \cos^{-1}\left((\underline{UP} \cdot \underline{ULOS})\ \text{sgn}\ TS_1\right)$ (elevation angle)

If $(\underline{ULOS} \cdot \underline{R}_{act3}) < 0$:

$TS = (1 - 2^{-28}) - TS$

If bit 14(ITSWITCH) of FLAGWRD7 = 0:

$T_{tpi} = T_{tpi} + NOMTPI$

If bit 7(HAVEELEV) of FLAGWRD2 = 0:

ELEV = TS (TS derived above, elevation angle)

TS = 0 (non-error return flag)

Return (to routine calling "S33/34.1")

DELELO = DELEL

DELEL = TS - ELEV

If $|DELEL| - K_{eeps} < 0$:

TS = 0 (non-error return flag)

Return (to routine calling "S33/34.1")

TS = $\frac{1}{2}$ TITER (Tag here "FIGTIME")

If $TS_{sp} = 0$:

Return (to routine calling "S33/34.1": TS $\neq$ 0, indicating error)

$TITER = TS_{sp}$

$TS = (\frac{1}{2} - ELEV)\ \text{sgn}\ (|\underline{R}_{pass3}| - |\underline{R}_{act3}|)$ (The $\frac{1}{2}$ is 180°)

If TS < 0: ($|\underline{R}_{pass3}|$ in 12D; $|\underline{R}_{act3}|$ in 14D; units 0D & 6D)

Return (to routine calling "S33/34.1": TS $\neq$ 0, indicating error)

$PD28CS = -|\underline{R}_{act3}| \cos(\frac{1}{2} - ELEV) / |\underline{R}_{pass3}|$

$TS = 1 - |PD28CS|$

If $TS < 0$:

Return (to routine calling "S33/34.1": TS $\neq$ 0, indicating error)

$PDOMEGA = |\underline{R}_{pass3}| \left(unit(UNR\underline{M} * unit\underline{R}_{act3})\right) \cdot \underline{V}_{act3}$

$T\underline{S} = unit\underline{R}_{pass3} * \underline{V}_{pass3}$

$PDOMEGP = |\underline{R}_{act3}| \left(unit(T\underline{S} * unit\underline{R}_{pass3})\right) \cdot \underline{V}_{pass3}$

$TS_1 = (unit\underline{R}_{act3} * unit\underline{R}_{pass3}) \cdot UNR\underline{M}$

$PDALFMPI = \left(\cos^{-1}(unit\underline{R}_{pass3} \cdot unit\underline{R}_{act3})\right) \text{ sgn } TS_1 - \frac{1}{2} + ELEV$

$TS_2 = (\frac{1}{2} - \cos^{-1} PD28CS) \text{ sgn}(|\underline{R}_{pass3}| - |\underline{R}_{act3}|)$ ($\frac{1}{2}$ is 180°)

$$PDDELTM = K_{twopi} \frac{|\underline{R}_{act3}| \, |\underline{R}_{pass3}| \, (PDALFMPI + TS_2)}{PDOMEGA - PDOMEGP}$$

If $|PDDELTM| - SECMAX \geq 0$:

PDDELTM = SECMAX sgn PDDELTM

If TITER < 0: (i.e. first pass) (Tag here "OKMAX")

TITER = 37777_8

DELTEEO = PDDELTM

NOMTPI = NOMTPI + DELTEEO

Proceed to "ADTIME+3"

If (DELEL DELELO) < 0: (i.e. sign change in angle)

SECMAX = SECMAX / 3

DELTEEO = $-\frac{1}{2}$ |PDDELTM| sgn DELTEEO

NOMTPI = NOMTPI + DELTEEO

Proceed to "ADTIME+3"

If $|DELELO| - |DELEL| \geq 0$:

DELTEEO = $|PDDELTM|$ sgn DELTEEO

NOMTPI = NOMTPI + DELTEEO

Proceed to "ADTIME+3"

TS = DELTEEO

DELTEEO = $-\frac{1}{2}$ DELTEEO

NOMTPI = NOMTPI - TS + DELTEEO (same as NOMTPI - 1.5 $DELTEEO_{n-1}$)

Proceed to "ADTIME+3"

ADTIME+3

Perform "INTSTALL"

Set bit 4(CONICINT) of FLAGWRD3 = 0

If NOMTPI $\neq$ 0: (as it would be expected to be)

Set bit 4(CONICINT) of FLAGWRD3 = 1

T_{decl} = NOMTPI

T_{et} = 0

$RC\underline{V} = \underline{R}_{aprec}$

$VC\underline{V} = \underline{V}_{aprec}$

Perform "INTEGRVS"

$\underline{R}_{act3} = \underline{R}_{att}$

$\underline{V}_{act3} = \underline{V}_{att}$

Perform "INTSTALL"

Set bit 4(CONICINT) of FLAGWRD3 = 0

If NOMTPI $\neq$ 0:

Set bit 4(CONICINT) of FLAGWRD3 = 1

T_{decl} = NOMTPI

T_{et} = 0

$RC\underline{V} = \underline{R}_{pprec}$

$VC\underline{V} = \underline{V}_{pprec}$

Perform "INTEGRVS"

$\underline{R}_{pass3} = \underline{R}_{att}$

$\underline{V}_{pass3} = \underline{V}_{att}$

Proceed to "ELCALC"

S34/35.2 Entered from "P34/P74C" (P35) and "P35/P75B" (P36)

SUBEXIT = Return address

Perform "INTSTALL"

Set bit 4(CONICINT) of FLAGWRD3 = 0

If $NN1_{sp} = 0$:

Set bit 4(CONICINT) of FLAGWRD3 = 1

$T_{decl} = T_{pass4}$

$T_{et} = INTIME$

$RC\underline{V} = \underline{R}_{pass3}$

$VC\underline{V} = \underline{V}_{pass3}$

Perform "INTEGRVS"

$\underline{R}_{targ} = \underline{R}_{att}$

Proceed to "S3435.25"

S3435.25

$\underline{V}_{pass4} = \underline{V}_{att}$

$TS_1 = (unit\underline{R}_{act3} * unit\underline{R}_{targ}) \cdot UNR\underline{M}$

$TS = \left(\cos^{-1}(unit\underline{R}_{act3} \cdot unit\underline{R}_{targ})\right) \text{ sgn } TS_1$

If TS < 0:

$TS = TS + (1 - 2^{-28})$

ACTCENT = TS

$DELLT4 = T_{pass4} - INTIME$

$TS_1 = NN1_{sp}$

$TS_2 = K_{epsfour}$

$\underline{R}_{init} = \underline{R}_{act3}$

$\underline{V}_{init} = \underline{V}_{act3}$

Perform "INITVEL"

$TS_2 = -$ UNRM

$TS_3 = -$ unitR_{act3}

$TS_1 = TS_3 *$ UNRM

$$DELVLVC = \begin{bmatrix} TS_1 \\ TS_2 \\ TS_3 \end{bmatrix} DELVEET3$$

$TS_1 =$ ULOS

$TS_3 = -$ unit(ULOS * UNRM)

$TS_2 = TS_3 *$ ULOS

$$DVLOS = \begin{bmatrix} TS_1 \\ TS_2 \\ TS_3 \end{bmatrix} DELVEET3$$

Proceed to address specified by SUBEXIT

S34/35.3 Entered from "P34/P74C" (P35) and "P35/P75B" (P36) if new velocity components input

Perform "INTSTALL"

Set bit 4(CONICINT) of FLAGWRD3 = 0

Set bit 4(CONICINT) of FLAGWRD3 = 1 (superseding previous line)

$T_{decl} = T_{pass4}$

$T_{et} = T_{ig}$

$RCV = \underline{R}_{act3}$

$VCV = \underline{V}_{act3} + DELVEET3$

Perform "INTEGRVS"

$\underline{R}_{targ} = \underline{R}_{att}$

$TS_2 = -$ UNRM

$TS_3 = -$ unit$\underline{R}_{act3}$

$TS_1 = TS_3 *$ UNRM

$$\text{DELVLVC} = \begin{bmatrix} TS_1 \\ TS_2 \\ TS_3 \end{bmatrix} \text{DELVEET3}$$

$TS_1 =$ ULOS

$TS_3 = -$ unit(ULOS * UNRM)

$TS_2 = TS_3 *$ ULOS

$$\text{DVLOS} = \begin{bmatrix} TS_1 \\ TS_2 \\ TS_3 \end{bmatrix} \text{DELVEET3}$$

Return

INITVEL Entered with TS_1, OD, set to iterations; TS_2, 2D, angle to 180°

Set bit 2(GUESSSW) of FLAGWRD1 = 1

$\underline{R}_{targ1} = \underline{R}_{targ}$ (Tag here "HAVEGUES", entered from "S40.9")

ITCTR = -1

COZY4 = cos TS_2 (TS_2 set before enter, cell 2D, BO revs.)

VTARGTAG = TS_1 (TS_1 set before enter, cell OD, B14)

R1VEC = $\underline{R}_{init}$

R2VEC = $\underline{R}_{targ1}$

TDESIRED = DELLT4

UN = unit(unit$\underline{R}_{init}$ * $\underline{V}_{init}$)

COZY4 = COZY4 + unit$\underline{R}_{targ1}$ · unit$\underline{R}_{init}$

Set bit 10(NORMSW) of FLAGWRD7 = 0

Proceed to "INITVEL2"

INITVEL2

If COZY4 < 0:

Set bit 10(NORMSW) of FLAGWRD7 = 1

$\underline{R2VEC} = |\underline{R2VEC}|\ \text{unit}\left(\underline{R2VEC} - (\underline{R2VEC} \cdot \underline{UN})\ \underline{UN}\right)$

If ITCTR < 0: (i.e. first pass)

$\underline{R}_{targ1} = \underline{R2VEC}$

$\underline{TS} = -\ \text{unitR1VEC} * \text{unitR2VEC}$

If $TS_z < 0$:

$\underline{TS} = -\ \underline{TS}$

$\text{GEOMSGN} = 22437_8\ \text{sgn}\left((\underline{TS} * \text{unitR1VEC}) \cdot \text{unitR2VEC}\right)$

Perform "LAMBERT"

Set bit 2(GUESSSW) of FLAGWRD1 = 0

$\underline{V}_{iprime} = \underline{VVEC}$

If VTARGTAG = 0:

Proceed to "INITVEL7"

Perform "INTSTALL"

$\underline{R1VEC} = \underline{R}_{init}$

$\underline{RCV} = \underline{R}_{init}$

$\underline{VCV} = \underline{V}_{iprime}$

$T_{et} = \text{INTIME}$

Set bit 4(CONICINT) of FLAGWRD3 = 0

$T_{decl} = \text{INTIME} + \text{DELLT4}$

Perform "INTEGRVS"

$\underline{V}_{target} = \underline{V}_{att1}$

ITCTR = ITCTR + 1

If ITCTR = VTARGTAG:

$\underline{R}_{targ1} = R2VEC$

Proceed to "INITVEL7"

$R2VEC = R2VEC + \underline{R}_{targ1} - \underline{R}_{att1}$

Proceed to "INITVEL2"

INITVEL7

$DELVEET3 = \underline{V}_{iprime} - \underline{V}_{init}$

$\underline{V}_{tprime} = \underline{V}_{target}$ ($\underline{V}_{target}$ computed in "INITV" if VTARGTAG = 0)

$\underline{R}_{targ} = \underline{R}_{targ1}$

Set bit 8(XDELVFLG) of FLAGWRD2 = 0

Return (to routine calling "INITVEL")

CDHMVR Entered from "P33/P73B" (P34)

$UNVEC = unit\underline{R}_{act2}$

$CSTH = unit\underline{R}_{pass2} \cdot UNVEC$

$SNTH = \sqrt{1 - CSTH^2}\ \ sgn\left((-\underline{R}_{act2} * \underline{R}_{pass2}) \cdot UP\underline{1} \right)$

$RVEC = \underline{R}_{pass2}$

Set bit 9(RVSW) of FLAGWRD7 = 0 (means new $\underline{r}$, $\underline{v}$ are desired)

$VVEC = \underline{V}_{pass2}$

Perform "TIMETHET"

$PD18\underline{V} = \underline{TS}_{v}$

$PD02R = |\underline{TS}_{r}|$

$DIFFALT = PD02R - |\underline{R}_{act2}|$

PDSEMAP = R1 / RdA (semi-major axis of passive vehicle)

PDSEMAA = PDSEMAP - DIFFALT

$PDVAV = (PD18\underline{V} \cdot UNVE\underline{C})\ (PDSEMAP / PDSEMAA)^{3/2}$

$TS = 2\ K_{rtmu} / |\underline{R}_{act2}| - K_{rtmu} / PDSEMAA$

$PDVAH = \sqrt{TS - PDVAV^2}$

$\underline{V}_{act3} = PDVAH\ unit(UP\underline{1} * UNVE\underline{C}) + PDVAV\ UNVE\underline{C}$

$DELVEET\underline{2} = \underline{V}_{act3} - \underline{V}_{act2}$

Return

NC12ST Entered from "NC12H", "P31", and "P32"

$PH1 = 9$ (the 9 is from most significant half of K_{25fds})

$CF = 0$

$PH2 = 0$

$DVH1 = K_{165fds}$

If bit 5(NC12FLG) of FLAGWRD0 = 1: (P32)

$NC = C_{nh1}$

$DVC = K_{165fds}$

If bit 5(NC12FLG) of FLAGWRD0 = 0: (P31)

$DVC = K_{225fds}$

$DVH2 = K_{25fds}$

$PI = 0.5$ (units of revolutions, i.e. 180^o)

$TN = TIMEC$

$T_{decl} = TIMEC$

Perform "CSMPREC"

Set bit 13(ITERFLG) of FLAGWRD2 = 0

$\underline{R}_{tig} = \underline{R}_{att}$

$\underline{R}_{ac} = \underline{R}_{att}$

$\underline{R}_{act2} = \underline{R}_{ac}$ (same cell)

$\underline{V}_{tig} = \underline{V}_{att}$

$\underline{V}_{ac} = \underline{V}_{att}$

$T_{decl} = TIMEC$

Perform "LEMCONIC"

$\underline{V}_{pc} = \underline{V}_{att}$

$\underline{R}_{pc} = \underline{R}_{att}$

Proceed to "PHSMCH"

PHSMCH Entered also from end of "NC12J"

$\underline{AM1} = \text{unit}(\underline{R}_{pc} * \underline{V}_{pc})$

$\underline{UP1} = \underline{AM1}$ (same cell)

$\underline{UR}_{pd} = \text{unit}\left[\left[\left(\text{unit}(\underline{R}_{ac} * \underline{V}_{ac})\right) * \underline{R}_{ac}\right] * \underline{AM1}\right]$

$TS_1 = PI - \cos^{-1}(\text{unit}\underline{R}_{pc} \cdot \underline{UR}_{pd})$

$TH2 = PI - TS_1 \operatorname{sgn}\left((\underline{UR}_{pd} * \underline{R}_{pc}) \cdot \underline{AM1}\right)$

If $TS_1 \geq 0$:

If $NC - 18 \geq 0$:

If $PI \neq 0$:

$\underline{TS}_1 = \underline{R}_{pc}$ (0D)

$\underline{TS}_2 = \underline{V}_{pc}$ (6D)

$TS_3 = TIMEC$ (12D)

$TS_4 = 1$ (B5 rev, 14D)

Perform "REVUP"

TN = TN - (RATEMP - TIMEC)

Proceed to "NC12J"

If $|TH2| - C_{epsl} < 0$: (Tag here "NC12B")

Proceed to "NC12F"

If $PH1 \neq 0$:

PH1 = PH1 - 1

Proceed to "NC12J"

$\underline{R}_{ac} = |\underline{R}_{ac}|$ unit $\left(\underline{R}_{ac} - (\underline{R}_{ac} \cdot \underline{AM1})\ \underline{AM1}\right)$ (Tag here "PSMALRM")

$\underline{R}_{act2} = \underline{R}_{ac}$ (same cell)

$TS = 600_8$

Proceed to "DISPALM"

NC12J Entered from "PHSMCH"

Set bit 9(RVSW) of FLAGWRD7 = 1 (new $\underline{r}$, $\underline{v}$ not desired)

CSTH = cos |TH2|

SNTH = sin |TH2|

$\underline{RVEC} = \underline{R}_{pc}$

$\underline{VVEC} = \underline{V}_{pc}$ sgn TH2

Perform "TIMETHET"

If PI $\neq$ 0:

PI = 0

$\underline{R}_{pc} = \underline{R}_{other}$

$\underline{V}_{pc} = \underline{V}_{other}$

$T_{pc} = T_{etlm}$

Set bit 2(NCINTFLG) of FLAGWRD0 = 0 (Tag here "NC12C")

$TS = T_{pc}$ (save for integration below)

TN = TN - T sgn TH2

$T_{pc} = TN$

$T_{decl} = TN$

Perform "INTSTALL"

Set bit 4(CONICINT) of FLAGWRD3 = 0

If bit 2(NCINTFLG) of FLAGWRD0 = 1: (should not be)

Set bit 4(CONICINT) of FLAGWRD3 = 1

$T_{et} = TS$ (TS saved above)

$\underline{VCV} = \underline{V}_{pc}$

$\underline{RCV} = \underline{R}_{pc}$

Perform "INTEGRVS"

$\underline{V}_{pc} = \underline{V}_{att}$

$\underline{R}_{pc} = \underline{R}_{att}$

Proceed to "PHSMCH"

NC12F Entered from "PHSMCH" if $|TH2| < C_{epsl}$

$PH1 = 0$

$\underline{R}_{ac} = |\underline{R}_{ac}|$ unit $\left(\underline{R}_{ac} - (\underline{R}_{ac} \cdot \underline{AM1})\ \underline{AM1}\right)$

$\underline{R}_{act2} = \underline{R}_{ac}$ (same cell)

$\underline{V}_{ac} = |\underline{V}_{ac}|$ unit $\left(\underline{V}_{ac} - (\underline{V}_{ac} \cdot \underline{AM1})\ \underline{AM1}\right)$

Set bit 2(NCINTFLG) of FLAGWRDO = 1

$T_{decl} = T_{tpi}$

Perform "INTSTALL"

Set bit 4(CONICINT) of FLAGWRD3 = 0

If bit 2(NCINTFLG) of FLAGWRDO = 1: (should be)

Set bit 4(CONICINT) of FLAGWRD3 = 1

$T_{et} = T_{pc}$

$\underline{VCV} = \underline{V}_{pc}$

$\underline{RCV} = \underline{R}_{pc}$

Perform "INTEGRVS"

Set bit 2(NCINTFLG) of FLAGWRDO = 1

$\underline{V}_{ptpi} = \underline{V}_{att}$

$\underline{R}_{ptpi} = \underline{R}_{att}$

DELH = DELH1

Perform "QRDTPI"

$\underline{R}_{afd} = \underline{R}_{j} - DELH1\ \underline{unitR}_{j}$

Proceed to "NC1LOOP"

NC1LOOP Also entered from end of "NC12G"

$\underline{TS}_1 = \underline{R}_{ac}$ (0D)

$\underline{TS}_2 = \underline{V}_{ac} + DVC\ unit(\underline{AM1} * \underline{R}_{ac})$ (6D)

$TS_3 = TIMEC$ (12D)

$TS_4 = NC$ (14D)

Perform "REVUP"

If bit 5(NC12FLG) of FLAGWRDO = 1: (P32)

$\underline{R}_{ah2} = \underline{R}_{att}$ (Tag here "NC2A")

$\underline{V}_{ah2} = \underline{V}_{att}$

TH2 = RATEMP

Proceed to "NC12D"

$\underline{R}_{ah1} = \underline{R}_{att}$ (P31 if get here)

$\underline{V}_{ah1} = \underline{V}_{att}$

TH1 = RATEMP

TS = CF

If TS $\neq$ 0: (Means have done "NC12G" at least once)

TS = 0.5

CI = TS

Proceed to "NC2LOOP"

NC2LOOP Also entered from end of "EHCOMP" (NCLPFLG = 0)

$\underline{TS}_1 = \underline{R}_{ah1}$ (0D)

$\underline{TS}_2 = \underline{V}_{ah1} + DVH1\ unit(\underline{AM1} * \underline{R}_{ah1})$ (6D)

$TS_3 = TH1$ (12D)

$TS_4 = C_{nh1}$ (14D)

Perform "REVUP"

$\underline{V}_{ah2} = \underline{V}_{att}$

TH2 = RATEMP

$\underline{TS} = \underline{R}_{att}$ (in OD)

Set bit 9(NCLPFLG) of FLAGWRD1 = 0

$\underline{R}_{ah2} = \underline{R}_{att}$

DELHTMP = DHNCC

Proceed to "EHCOMP"

NC12D Entered from "NC1LOOP" (P32) and "EHCOMP" (NCLPFLG = 0 & converged)

TS = CF

If TS ≠ 0: (Means have done "NC12G" at least once)

TS = 0.5

CI = TS

Proceed to "NCCLOOP"

NCCLOOP Also entered from "EHCOMP" (NCLPFLG = 1)

Set bit 2(NCINTFLG) of FLAGWRD0 = 1

$T_{as} = TH2 + C_{tcs}$

$T_{decl} = T_{as}$

Perform "INTSTALL"

Set bit 4(CONICINT) of FLAGWRD3 = 0

If bit 2(NCINTFLG) of FLAGWRD0 = 1: (should be)

Set bit 4(CONICINT) of FLAGWRD3 = 1

$T_{et} = TH2$

$\underline{VCV} = \underline{V}_{ah2} + DVH2\ unit(\underline{AM1} * \underline{R}_{ah2})$

$\underline{RCV} = \underline{R}_{ah2}$

Perform "INTEGRVS"

$\underline{V}_{as} = \underline{V}_{att}$

$\underline{R}_{as} = \underline{R}_{att}$

$\underline{TS} = \underline{R}_{att}$ (OD, same cell)

Perform "RADUP"

DELH = $|\underline{TS}_r| - |\underline{R}_{as}|$

$\underline{TS}_1 = \underline{TS}_r$

$\underline{TS}_2 = \underline{TS}_v$ (OD, same cell)

Perform "COE"

$\underline{V}_{asf} = PDV\underline{A}$

Set bit 2(NCINTFLG) of FLAGWRD0 = 1

$T_{decl} = T_{tpi}$

Perform "INTSTALL"

Set bit 4(CONICINT) of FLAGWRD3 = 0

If bit 2(NCINTFLG) of FLAGWRD0 = 1: (should be)

Set bit 4(CONICINT) of FLAGWRD3 = 1

$T_{et} = T_{as}$

$VC\underline{V} = \underline{V}_{asf}$

$RC\underline{V} = \underline{R}_{as}$

Perform "INTEGRVS"

$\underline{R}_{af} = \underline{R}_{att}$

$\underline{TS} = \underline{R}_{att}$

Set bit 9(NCLPFLG) of FLAGWRD1 = 1

DELHTMP = DELH1

Proceed to "EHCOMP"

NC12G Entered from "EHCOMP" (NCLPFLG = 1 & converged)

$PDEP = \cos^{-1}(\text{unit}\underline{R}_{af} \cdot \text{unit}\underline{R}_{afd})\ \text{sgn}\left((\underline{R}_{af} * \underline{R}_{afd}) \cdot \underline{AM1}\right)$

If $(|PDEP| - 4\,C_{epsl}) < 0$:

Proceed to "NC12OUT"

PDITX = DVC (2D)

PDITE = PDEP (4D)

PDITC = CF (6D)

EO = EOP

DVO = DVOC

Perform "ITER"

If bit 13(ITERFLG) of FLAGWRD2 = 1:

Proceed to "OLPALRM" (602_8 alarm)

CF = TS (updated PDITC)

DVC = PDITX

DVOC = DVO

EOP = EO

Proceed to "NC1LOOP"

EHCOMP Entered from "NC2LOOP" (NCLPFLG = 0) and "NCCLOOP" (NCLPFLG = 1)

RATEMP = $|\underline{TS}|$

Perform "RADUP"

PDEH12 = $|\underline{TS}_r|$ - RATEMP - DELHTMP

PDITP = PH1 (0D)

If bit 9(NCLPFLG) of FLAGWRD1 = 1:

PDITP = PH2

If PDITP $\neq$ 0:

If ($|$PDEH12$|$ - C_{eps2}) < 0:

If bit 9(NCLPFLG) of FLAGWRD1 = 0: (Tag here "LOOPX")

Proceed to "NC12D"

Proceed to "NC12G"

PDITX = DVH1 (2D) (Tag here "ITERUP")

If bit 9(NCLPFLG) of FLAGWRD1 = 1:

PDITX = DVH2

PDITE = PDEH12 (4D)

PDITC = CI (6D)

X1 = PSHFT1

If bit 9(NCLPFLG) of FLAGWRD1 = 1:

X1 = PSHFT2

Perform "ITER"

If bit 13(ITERFLG) of FLAGWRD2 = 1:

TS = 601_8 (Tag here "ILPALRM")

Proceed to "DISPALM"

CI = TS

If bit 9(NCLPFLG) of FLAGWRD1 = 1:

PSHFT2 = X1

DVH2 = PDITX

PH2 = PDITP

Proceed to "NCCLOOP"

DVH1 = PDITX (Tag here "NC2ITR")

PSHFT1 = X1

PH1 = PDITP

Proceed to "NC2LOOP"

NC12OUT Entered from start of "NC12G" and "OLPALRM" (PRO)

DELVLVC = (DVC, 0, 0)

$\underline{TS}_2$ = - $\underline{AM1}$

$\underline{TS}_3$ = - $\underline{unitR}_{as}$

$\underline{TS}_1$ = $\underline{TS}_3$ * $\underline{AM1}$

$$\underline{VGNSR} = \begin{bmatrix} \underline{TS}_1 \\ \underline{TS}_2 \\ \underline{TS}_3 \end{bmatrix} \quad (\underline{V}_{asf} - \underline{V}_{as})$$

$T_{cdh} = T_{as}$ (T_{cdh} notation also TNSR)

$T_{csi} = TH2$ (T_{csi} notation also TNCC)

If bit 5(NC12FLG) of FLAGWRDO = 1: (P32)

DVDSP1 = DVH2 (Tag here "NC2N80")

DVDSP2 = $|\underline{VGNSR}|$

DHDSP = DHNCC

Proceed to "NC12DH"

DVDSP2 = DVH2 (P31 if here)

DVDSP1 = DVH1

NC2TIG = TH1

$\underline{TS} = \underline{R}_{ah1}$

Perform "RADUP"

DHDSP = $|\underline{TS}_r| - |\underline{R}_{ah1}|$

Proceed to "NC12DH"

REVUP Entered with $\underline{TS}_1$ position, $\underline{TS}_2$ velocity, TS_3 time tag, TS_4 revs

$PDA = 1/\left(2/|\underline{TS}_1| - K_{ldmue}\ (\underline{TS}_2 \cdot \underline{TS}_2)\right)$

Set bit 2(NCINTFLG) of FLAGWRDO = 1

$RATEMP = TS_3 + TS_4\ K_{2pib3}\ K_{ldsqmu}\ \sqrt{PDA}\ PDA$

$T_{decl} = RATEMP$

Perform "INTSTALL"

Set bit 4(CONICINT) of FLAGWRD3 = 0

If bit 2(NCINTFLG) of FLAGWRDO = 1: (should be)

Set bit 4(CONICINT) of FLAGWRD3 = 1

$T_{et} = TS_3$

$VC\underline{V} = TS_{\underline{2}}$

$RC\underline{V} = TS_{\underline{1}}$

Perform "INTEGRVS"

Return

RADUP Entered with TS, OD, vector to be matched

$VVEC = V_{pc}$

$RVEC = R_{pc}$

$TS_{\underline{1}} = RVEC * VVEC$

$TS_2 = (RVEC * TS) \cdot TS_{\underline{1}}$

$TS = \left(\cos^{-1} (unitRVEC \cdot unitTS)\right)$ sgn TS_2

CSTH = cos TS

Set bit 9(RVSW) of FLAGWRD7 = 0 (new r, v desired)

SNTH = sin TS

Perform "TIMETHET"

Return (TS_r in MPAC; TS_v in OD)

ITER Entered from "EHCOMP", "GETRJ", and "NC12G"

If PDITC = 0:

TS = K_{1fds}

Proceed to "ITER3"

If (PDITC - 0.5) = 0:

TS = (PDITE / PDITP), shifted left X1 places

Proceed to "ITER3"

If PDITE - E0 = 0:

TS = K_{0p3fds} (constant is 3 fps for velocity)

Proceed to "ITER3"

PD8D = (PDITE - EO), with number of leading zeros stored in X1

PDITE = PD8D / (PDITX - DVO)

If (PDITC - 14) $\geq$ 0:

Set bit 13(ITERFLG) of FLAGWRD2 = 1

Return

TS = PDITE / PDITP

Proceed to "ITER3"

ITER3

PD8D = TS

EO = PDITE

DVO = PDITX

PDITX = DVO - PD8D (in 2D)

TS = PDITC + 1 (in 4D)

Return

DISP45A Entered from "P33" and "N81DISP"

Perform "VN1645"

$T_{decl} = T_{csi}$ (T_{csi} notation also TNCC)

INTIME = T_{csi}

Perform "CSMPREC"

$\underline{R}_{init} = \underline{R}_{att}$

$\underline{R}_{pass3} = \underline{R}_{att}$ (used to restore $\underline{R}_{init}$ in "NCCDMP7+17")

$\underline{V}_{init} = \underline{V}_{att}$

$\underline{V}_{act3} = \underline{V}_{att}$

$T_{decl} = T_{cdh}$ (T_{cdh} notation also TNSR)

Perform "LEMPREC"

Set bit 13(ITERFLG) of FLAGWRD2 = 0

$\underline{VC}V = \underline{V}_{att}$ (note that this violates "INTSTALL" lockout

$\underline{RC}V = \underline{R}_{att}$ constraints generally observed elsewhere)

$\underline{UP1} = unit(\underline{RC}V * \underline{VC}V)$

$T_{decl} = T_{tpi}$ (Tag here "NCCDMP4")

Perform "LEMPREC"

Set bit 2(NCINTFLG) of FLAGWRD0 = 0

$\underline{R}_{ptpi} = \underline{R}_{att}$

$\underline{V}_{ptpi} = \underline{V}_{att}$

DELH = DELH1

Perform "QRDTPI"

$\underline{TS}_1 = \underline{R}_j$

$\underline{TS}_2 = \underline{V}_j$

Perform "COE"

Perform "INTSTALL"

$\underline{RC}V = \underline{PDRA}$

$\underline{VC}V = \underline{PDVA}$

$T_{et} = T_{tpi}$

Set bit 4 (CONICINT) of FLAGWRD3 = 0

$T_{decl} = T_{cdh}$ (T_{cdh} notation also TNSR)

Perform "INTEGRVS"

$\underline{R}_{nsr} = \underline{R}_{att}$

$\underline{R}_{act2} = \underline{R}_{nsr}$ (same cell)

$\underline{R}_{targ} = \underline{R}_{att}$

$\underline{V}_{nsrp} = \underline{V}_{att}$

$DELLT4 = T_{cdh} - T_{csi}$ (T_{csi} notation also TNCC)

$TS_1 = 2$

$TS_2 = K_{15deg}$

Perform "INITVEL"

$\underline{TS} = \underline{V}_{nsrp} - \underline{V}_{tprime}$ (Tag here "NCCDMP7")

$\underline{V}_{init} = \underline{V}_{tprime}$

$\underline{R}_{init} = \underline{R}_{nsr}$

Perform "GET.LVC"

$DELVLVC_y = \underline{V}_{tprime} \cdot \underline{UP1}$

$\underline{VGNSR} = \underline{DELVLVC}$

Proceed to "NCCDMP7+17"

QRDTPI Entered from "NC12F" and "DISP45A"

DTQRD = 0

ITERCNT = 0

If $(ELEV - \frac{1}{2}) \geq 0$: (the $\frac{1}{2}$ is 180°)

ELEV = ELEV $- \frac{1}{2}$

COSE = cos ELEV

Proceed to "GETRJ"

GETRJ

Perform "INTSTALL"

Set bit 4(CONICINT) of FLAGWRD3 = 0

$\underline{RCV} = \underline{R}_{ptpi}$

$\underline{VCV} = \underline{V}_{ptpi}$

$T_{et} = T_{tpi}$

$T_{decl} = T_{tpi} + DTQRD$

If bit 2(NCINTFLG) of FLAGWRD0 = 1:

Set bit 4(CONICINT) of FLAGWRD3 = 1

Perform "INTEGRVS" (Tag here "GOPREC")

$\underline{R}_j = \underline{R}_{att}$ (same cell)

$\underline{V}_j = \underline{V}_{att}$ (same cell)

$\underline{URJ} = unit\underline{R}_{att}$

$RJMAG = |\underline{R}_{att}|$

$TS = (\underline{V}_{ptpi} * \underline{R}_{ptpi}) \cdot (unit\underline{R}_{ptpi} * \underline{URJ})$ (B38 m^2/cs)

If $TS \neq 0$:

$TS = \left(\cos^{-1} (unit\underline{R}_{ptpi} \cdot \underline{URJ})\right)$ sgn TS

$$PDET = \frac{1}{4} - \left[\sin^{-1} \left(\frac{(RJMAG - DELH)\ COSE}{|\underline{R}_{ptpi}|} \right) + ELEV \right] - TS$$

If $(|PDET| - C_{epsl}) < 0$:

Return (to routine calling "QRDTPI")

PDITX = DTQRD

PDITE = PDET

PDITC = ITERCNT

Perform "ITER"

If bit 13(ITERFIG) of FLAGWRD2 = 1:

$TS = 603_8$ (Tag here "QRDALRM")

Proceed to "DISPALM"

ITERCNT = TS

DTQRD = PDITX

Proceed to "GETRJ"

COE Entered with $\underline{TS}_1$ position; $\underline{TS}_2$ velocity

$PDRM = |\underline{TS}_1| - DELH$

$\underline{PDRA} = PDRM\ unit\underline{TS}_1$ (left in MPAC when exit)

$$PDCA = K_{mum42} / \left(2 K_{mum42} / |\underline{TS}_1| - |\underline{TS}_2|^2 \right)$$

$PDCAD = PDCA - DELH$

$$TS = (PDCA / PDCAD)^3 \quad (unit\underline{TS}_1 \cdot \underline{TS}_2)^2$$

$$PDVV = \sqrt{TS} \quad sgn\ (unit\underline{TS}_1 \cdot \underline{TS}_2)$$

$$TS_3 = K_{mum42} \ (\ 2\ PDCAD\ -\ PDRM)\ /\ (PDCAD\ \ PDRM)\ \ -\ TS \qquad (B14)$$

$$PDV\underline{A} = \sqrt{TS_3} \quad unit \left((unit\underline{TS}_1 * \underline{TS}_2) * unit\underline{TS}_1 \right) \ + \ PDVV\ unit\underline{TS}_1$$

Return

Quantities in Computations

See also list of major variables and list of routines

ACTCENT: Value of CSM (active vehicle) central angle of transfer computed in "S3435.25" (entered for P35 and P36), scale factor BO, units revolutions, in range 0 - 360°. It can be displayed at crew option in R1 of N52: if too close to 180°, then the maneuver parameters should be reconsidered.

AM$\underline{1}$: Unit vector, scale factor B1, computed in "PHSMCH" in direction perpendicular to passive vehicle plane. It is assigned to the same cell as UP$\underline{1}$, but a separate loading of UP$\underline{1}$ (for compatibility with subsequent computations) is shown for documentation convenience.

AUTOY: See Burn Control.

C_{epsl}: Erasable memory constant, program notation "EPS1" (alternate notation "DTHET"), scale factor BO, units revolutions. It is used as a loop convergence criterion in "PHSMCH", "NC12G" (where multiplied by 4 before use), and "GETRJ".

C_{eps2}: Erasable memory constant, program notation "EPS2", scale factor B29, units meters, used as a loop convergence criterion in "EHCOMP".

C_{nhl}: Single precision erasable memory constant, program notation "NH1", scale factor B5, units revolutions, used to initialize NC in "NC12ST" (for P32) and to specify number of revolutions for "REVUP" in "NC2LOOP". It specifies the number of revolutions between the NC2 (P32) and NCC (P33) maneuvers.

C_{tcs}: See Burn Control.

CF: Loop counter (contains integer values only), scale factor B5, units counts. It is initialized to zero at the start of "NC12ST" and incremented in "NC12G" logic (due to "ITER" computations).

CI: Loop counter, initialized to 0 or 0.5 in "NC1LOOP" and "NC12D", scale factor B5, units counts. The value of 0.5 is used as a first-pass indicator in "ITER" provided a non-zero value of partial (PDITP) is available.

CMYDOT: See Burn Control.

COSE: Value of cos ELEV computed in "QRDTPI", scale factor B1.

COZY4: Criterion used in "INITVEL2" to decide if input position vector forms an angle with the target position vector that is too close to 180°, scale factor B2. It is initially loaded with cos TS_2, where TS_2, stored in push-down list location 2D as an angle, BO revolutions, is angle away from 180° within which "INITVEL2" rotates the target vector to be in the plane of $\underline{R}_{init}$ and $\underline{V}_{init}$.

CSTH: See Conic Routines.

DELEL: Value of error between derived and specified values of elevation angle in "ELCALC", scale factor BO, units revolutions.

DELELO: Value of previous DELEL, scale factor BO, units revolutions, stored in 26D.

DELH: Altitude difference used as a communication cell with "QRDTPI" and "COE", scale factor B29, units meters.

DELH1: Value of altitude difference between orbits at ignition time for (TPI) NSR (P34), scale factor B29, units meters. It is displayed in R3 of N57, and would also be expected to form part of the prelaunch load.

DELHTMP: Communication cell with "EHCOMP" used to contain DHNCC (if from "NC2LOOP") or DELH1 (if from "NCCLOOP"), scale factor B29, units meters.

DELLT4: Communication cell with "INITVEL", containing desired time of flight from $\underline{R}_{init}$ to $\underline{R}_{targ}$, scale factor B28, units centi-seconds. It is loaded into TDESIRED for use in Lambert routine.

DELTEEO: Value of time increment information in "ELCALC" (generally the value added to NOMTPI), scale factor B28, units centi-seconds.

DELVEET1: Value of DELVEET2 sampled at start of "P33/P73B" (part of P34) for use subsequently to compute N81 display information there, scale factor B7, units meters/centi-second.

DELVEET2: Vector velocity change required for P34 burn (computed in "CDHMVR"), scale factor B7, units meters/centi-second.

DELVEET3: Vector velocity increment required to perform maneuver, scale factor B7, units meters/centi-second. It is loaded at end of "INITVEL7". DELVSIN is same cell, but for documentation convenience (interface with "VN1645") loading of DELVSIN into DELVEET3 shown where necessary. Also loaded in P35 ("P34/P74C") and P36 ("P35/P75B") from revised noun inputs if required.

DELVLVC: See Burn Control.

DHDSP: Value of altitude difference for next maneuver, scale factor B29, units meters, displayed in R2 of N84. It is loaded in "NC12OUT" and is used in "NC12DH" (for P32), after display, to load DHNCC.

DHNCC: Value of altitude difference between orbits at ignition time for NCC (P33), scale factor B29, units meters, displayed in R2 of N57. It is loaded from DHDSP in "NC12DH" for P32 (after the display of N84), and would also be expected to form part of the prelaunch load.

DIFFALT: Difference in altitude between OWS and CSM at the NSR (P34) maneuver, scale factor B29, units meters, computed in "CDHMVR". It is displayed in R1 of N75.

DTQRD: Value of time increment to T_{tpi} used in "QRDTPI" routine, scale factor B28, units centi-seconds.

DVC: Value of maneuver magnitude used in outer iteration loop (NC1 for P31, NC2 for P32), scale factor B7, units meters/centi-second. It is initialized in "NC12ST", used at start of "NC1LOOP" to compute velocity for input to "REVUP", is updated in "NC12G", and forms x component of DELVLV$\underline{C}$ in "NC12OUT".

DVDSP1: Value of velocity increment loaded in "NC12OUT" for display in R1 of N84, scale factor B7, units meters/centi-second. It is loaded with DVH1 for P31 and DVH2 for P32 (i.e. the P32 and P33 maneuver magnitudes respectively).

DVDSP2: Value of velocity increment loaded in "NC12OUT" for display in R3 of N84, scale factor B7, units meters/centi-second. It is loaded with DVH2 for P31 and $|\underline{VGNSR}|$ for P32 (i.e. the P33 and P34 maneuver magnitudes respectively).

DVH1: Value of velocity increment for NC2 maneuver (P32) initialized in "NC12ST", scale factor B7, units meters/centi-second. It is updated in "EHCOMP" if bit 9(NCLPFLG) of FLAGWRD1 = 0, and used to load DVDSP1 for P31, as well as to compute velocity in "NC2LOOP" for input to "REVUP".

DVH2: Value of velocity increment for NCC maneuver (P33) initialized in "NC12ST", scale factor B7, units meters/centi-second. It is updated in "EHCOMP" if bit 9(NCLPFLG) of FLAGWRD1 = 1, and used to load DVDSP1 for P32 and DVDSP2 for P31.

DVLO$\underline{S}$: See Burn Control.

DVO: Value of independent variable (PDITX) being iterated, stored in "ITER3" (also sampled and initialized in "NC12G"). For "ITER" entrances from "EHCOMP" and "NC12G", scale factor B7, units meters/centi-second; for entrance from "GETRJ", scale factor B28, units centi-seconds.

DVOC: Cell used to initialize and retain value of DVO in "NC12G", scale factor B7, units meters/centi-second.

ELEV: Value of elevation angle between the CSM/OWS vehicle line of sight and the CSM vehicle local horizontal at TPI ignition time, scale factor B0, units revolutions ("QRDTPI" reduces to range 0 - 180°). It is displayed in R2 of N55, and would also be expected to form part of the prelaunch load. An input value of 0 in P35 causes the angle to be derived from the specified time; a non-zero value causes the time to be computed in routine starting at "S33/34.1".

EO: Value of dependent variable (PDITE) being iterated, stored in "ITER3" (also sampled and initialized in "NC12G"). For entrances to "ITER" from "GETRJ" and "NC12G", scale factor is B0, units revolutions; from "EHCOMP", scale factor is B29, units meters.

EOP: Value of EO used to initialize and retain quantity in "NC12G", scale factor B0, units revolutions.

GEOMSGN: See Conic Routines. The magnitude of the number (provided that it is non-zero) is not significant, but only the sign: the magnitude loaded in "INITVEL2", for programming convenience, is the most significant half of K_{mutab_0} (see Conic Routines).

INTIME: Communication cell with "INITVEL" containing the time tag of the state vector in $\underline{R}_{init}$ and $\underline{V}_{init}$, scale factor B28, units centi-seconds. It is not required if VTARGTAG = 0.

ITCTR: Single precision counter, scale factor B14, of the number of iterations of "INITVEL" computations which have been performed. It is set to an initial condition of -1, so that a value of +1, for example, would mean that 2 passes through "LAMBERT" and "INTEGRVS" are performed, with the output of "LAMBERT" used in the "INTEGRVS" routine to refine the target position. The desired number of iterations is specified by the contents of VTARGTAG.

ITERCNT: Counter (contains integer values only), scale factor B5, units counts, of the number of iterations in "QRDTPI" loop (initialized to 0 in "QRDTPI" and updated in "GETRJ" due to "ITER" computations).

K_{Op3fds}: Constant, program notation "0.3F/S", used in "ITER" if PDITE = EO. When "ITER" entered from "EHCOMP" and "NC12G", scale factor is B7, units meters/centi-second; when entered from "GETRJ", scale factor is B28, units centi-seconds. Nominal value is 0.009144×2^{-7}, corresponding (for B7 meters/centi-second) to $3.0 \times 0.01 \times 0.3048 \times 2^{-7}$, where first term is value in fps (note that it is <u>ten times</u> the value suggested by the mnemonic), second converts to centi-seconds, third converts to meters, and fourth is scale factor. For units of centi-seconds, octal value is $00001_8\ 05350_8$, corresponding to 191.76 seconds.

K_{ldmue}: Constant, program notation "1/MUE", scale factor B-34, units cs^2/m^3. Value is $0.25087606E\text{-}10 \times 2^{34}$ (see K_{mutab_2} in Conic Routines).

K_{ldsqmu}: Constant, program notation "1/SQMU", scale factor B-17, units $(cs^2/m^3)^{\frac{1}{2}}$. Value is $0.50087529E\text{-}5 \times 2^{17}$ (see K_{mutab_6} in Conic Routines).

K_{1fds}: Constant, program notation "1F/S", used in "ITER" if PDITC = 0. When "ITER" entered from "EHCOMP" and "NC12G", scale factor is B7, units meters/centi-second; when entered from "GETRJ", scale factor is B28, units centi-seconds. Nominal value is 0.003048×2^{-7}, corresponding (for B7 meters/centi-second) to $1.0 \times 0.01 \times 0.3048 \times 2^{-7}$, where first term is value in fps, second converts to centi-seconds, third converts to meters, and fourth is scale factor. For units of centi-seconds, octal value is 00000_8 14370_8, corresponding to 63.92 seconds.

K_{15deg}: Constant, program notation "15DEG", scale factor B0, units revolutions. Value is 0.04166667, corresponding to 15 x (1/360), where first term is value in degrees and second converts to revolutions.

K_{165fds}: Constant, program notation "165F/S", scale factor B7, units meters/centi-second. Value is 0.50292×2^{-7}, corresponding to $165 \times 0.01 \times 0.3048 \times 2^{-7}$, where first term is value in fps, second converts to centi-seconds, third converts to meters, and fourth is scale factor.

K_{2pib3}: Constant, program notation "2PIB3", scale factor B3. Value is 6.2831853×2^{-3}, corresponding to $2 \times 3.14159265 \times 2^{-3}$, where first term is equation factor, second is π, and third is scale factor.

K_{25fds}: Constant, program notation "25F/S", scale factor B7, units meters/centi-second. Value is 0.07620×2^{-7}, corresponding to $25 \times 0.01 \times 0.3048 \times 2^{-7}$, where first term is value in fps, second converts to centi-seconds, third converts to meters, and fourth is scale factor. Octal value is 00011_8 30073_8, allowing a single precision load of most significant half (i.e. octal 11 or decimal 9) to be used to initialize PH1 at the start of "NC12ST" (PH1 then is B14).

K_{225fds}: Constant, program notation "225F/S", scale factor B7, units meters/centi-second. Value is 0.68580×2^{-7}, corresponding to $225 \times 0.01 \times 0.3048 \times 2^{-7}$, where first term is value in fps, second converts to centi-seconds, third converts to meters, and fourth is scale factor.

K_{eeps}: Constant, program notation "ELEPS", scale factor B0, units revolutions. Value is 0.27777777E-3, corresponding to 0.1^{o}.

$K_{epsfour}$: Constant, program notation "EPSFOUR", scale factor B0, units revolutions. Value is 0.0416666666, corresponding to 15^{o}.

K_{max250}: Constant, program notation "MAX250", scale factor B28, units centi-seconds. Value is $25E3 \times 2^{-28}$, corresponding to 250 seconds.

K_{mum42}: Constant, program notation "MU(-42)", scale factor B42, units meters^3/centi-seconds^2. Value is $3.986032E10 \times 2^{-42}$ (see K_{mutab_0} in Conic Routines).

K_{rtmu}: Constant, program notation "RTMU", scale factor B36, units meters^3/centi-seconds^2. Value is $3.986032E10 \times 2^{-36}$ (see K_{mutab_0} in Conic Routines).

K_{twopi}: Constant, program notation "TWOPI", scale factor B4, value $6.283185307 \times 2^{-4}$. Value corresponds to $2 \times 3.1415926535 \times 2^{-4}$, where first term is equation factor, second is π, and third is scale factor.

LMYDOT: See Burn Control.

NC: Number of revolutions between burn being targeted and subsequent burn, scale factor B5, units revolutions. It is set to C_{nhl} for P32 computations in "NC12ST". For P31 use, it is loaded with HALFREVS (modulo 32 revolutions or 64 half-revolutions) in "P31" (similar loading done in "P32", but written over by C_{nhl} subsequently).

NC2TIG: See Burn Control.

NN1: Value of number of offsets desired in Lambert targeting, scale factor B14, units counts, displayed by R1 of N55 (program notation also "NN"). A value of 0 (as done at start of "P35" before display) means that conic targeting is to be used; typical precision setting is 2. The setting is effective only for P35 and P36 use.

NOMTPI: Value of computed correction to T_{tpi}, scale factor B28, units centi-seconds, loaded in "ELCALC". This scheme permits $\underline{R}_{aprec}/\underline{R}_{pprec}$ state vector time tag to remain T_{tpi} while the iteration continues for the proper updated value of T_{tpi}.

PDO2R: Value of magnitude of OWS vehicle position vector used in "CDHMVR" to compute DIFFALT, scale factor B29, units meters, stored in push-down location 02D.

PD18V: Value of OWS velocity vector used in "CDHMVR", scale factor B7, units meters/centi-second. It is the velocity vector at the point corresponding to PDO2R (i.e. radially above/below CSM), and is stored in push-down location 18D.

PD28CS: Value of ratio of active to passive vehicle radius magnitudes times cos ELEV (i.e. times - cos ($\frac{1}{2}$ - ELEV)), scale factor B1, stored in push-down list location 28D.

PD8D: Temporary storage cell used in "ITER" routine, corresponding to push-down list location 8D, variable scaling and units. Left in "ITER3" with same scaling and units as PDITX.

PDA: Value of semi-major axis computed in "REVUP", scale factor B24, units meters, stored in push-down list.

PDALFMPI: Value of guidance equation quantity α minus π, scale factor B0, units revolutions, stored in push-down list location 18D.

PDCA: Value of semi-major axis computed in "COE", scale factor B29, units meters, stored in push-down list location 22D (axis of OWS).

PDCAD: Value of desired (CSM) semi-major axis computed in "COE", scale factor B28, units meters, stored in push-down list location 32D.

PDDELTM: Value of guidance equation quantity δt, scale factor B28, units centi-seconds, stored in push-down list location 12D.

PDEH12: Altitude error quantity computed in "EHCOMP", scale factor B29, units meters, stored in push-down list location 2D.

PDEP: Angular error between $\underline{R}_{af}$ and $\underline{R}_{afd}$ computed in "NC12G", scale factor BO, units revolutions, stored in push-down cell 2D, used as a check on phasing at TPI.

PDET: Value of difference between the desired central angle between the CSM and OWS position vectors and actual angle computed in "QRDTPI" routine, scale factor BO, units revolutions, stored in push-down list location 12D.

PDITC: Communication cell with "ITER" routine loaded with value of counter to be used to check for excessive iterations, scale factor B5, units counts. It is stored in push-down list location 6D. The updated count is left in 4D when exit from routine (from which the calling routine samples the value for storage in calling routine cell).

PDITE: Value of dependent variable used as input to "ITER", and stored in push-down list location 4D when enter. For entrances from "GETRJ" and "NC12G", scale factor is BO, units revolutions; for entrances from "EHCOMP", scaling is B29, units meters.

PDITP: Value of partial of dependent variable with respect to independent one in "ITER" package, required as input only for PDITC = 0.5 (i.e. for entrance to "ITER" from "EHCOMP"). Scaling information is variable, with information on number of leading zeros contained in X1 when enter routine. Stored in push-down list location 0D.

PDITX: Value of independent variable used as input to "ITER", and derived by it (as updated value). It is stored in push-down list location 2D. For entrances from "EHCOMP" and "NC12G", scale factor is B7, units meters/centi-second; for entrances from "GETRJ", scale factor is B28, units centi-seconds.

PDOMEGA: Value of CSM angular velocity information times radius magnitude ratio (for convenience in scaling), scale factor B37, stored temporarily in push-down list location 16D. Location subsequently modified to contain (PDOMEGA - PDOMEGP), with scale factor information in X1 also.

PDOMEGP: Value of OWS angular velocity information times radius magnitude ratio (for convenience in scaling: the reciprocal of the ratio used for PDOMEGA is employed), scale factor B37, stored as described under PDOMEGA.

PDRA: Position vector computed by "COE" routine, scale factor B29, units meters, stored initially in push-down list location 14D, but left in MPAC when exit from routine.

PDRM: Magnitude of PDRA computed by "COE" routine, scale factor B29, units meters, stored in push-down list location 12D.

PDSEMAA: Value of semi-major axis of CSM computed in "CDHMVR", scale factor B29, units meters, stored in push-down list location 4D.

PDSEMAP: Value of semi-major axis of OWS computed in "CDHMVR", scale factor B29, units meters, stored temporarily in push-down list location 4D (then that location used for PDSEMAA).

PDVA: Velocity vector computed by "COE" routine, scale factor B7, units meters/centi-second, left in push-down list location OD when exit from the routine.

PDVAH: Value of "horizontal" component of required CSM velocity vector after P34 (NSR) maneuver, computed in "CDHMVR". Scale factor is B7, units meters/centi-second, and stored in push-down location 10D.

PDVAV: Value of "vertical" (i.e. radial) component of required CSM velocity vector after P34 (NSR) maneuver, computed in "CDHMVR". Scale factor is B7, units meters/centi-second, and stored in push-down location 8D.

PDVV: Value of radial component of PDVA computed in "COE", scale factor B7, units meters/centi-second, stored in push-down list location 30D.

PH1: Cell used for two purposes. At start of "NC12ST", initialized to 9 (B14) for use as a loop counter in "PHSMCH". Set 0 at start of "NC12F" and then used in "EHCOMP" to retain previous PDITP (for use in initializing that quantity) if bit 9(NCLPFLG) of FLAGWRD1 = 0. Scaling there same as PDITP.

PH2: Cell initialized to 0 at start of "NC12ST" and then used in "EHCOMP" to retain previous PDITP (for use in initializing that quantity) if bit 9(NCLPFLG) of FLAGWRD1 = 1. Scaling same as PDITP.

PI: Cell used as a first-pass indicator and in computation of TH2 in "PHSMCH", scale factor B0, units revolutions. Initialized to 180^{o} in "NC12ST" and set 0 in "NC12J".

PSHFT1: Single precision cell, scale factor B14, used to contain scaling information for PH1 (determined in "ITER" routine), employed to initialize X1 (and retain its value) in "EHCOMP".

PSHFT2: Single precision cell, scale factor B14, used to contain scaling information for PH2 (determined in "ITER" routine), employed to initialize X1 (and retain its value) in "EHCOMP".

$\underline{R}_{ac}$: Value of CSM position vector at ignition time for next burn, scale factor B29, units meters. It is initially computed in "NC12ST" (time of TIMEC), and is rotated into OWS plane in "NC12F" (or, for an alarm exit, in "PHSMCH"). It is assigned the same cell as $\underline{R}_{act2}$, but for documentation convenience a separate loading of $\underline{R}_{act2}$ is shown where necessary.

$\underline{R}_{act2}$: Value of CSM position vector at ignition, scale factor B29, units meters. It occupies the same cell as $\underline{R}_{ac}$ and $\underline{R}_{nsr}$, but for documentation convenience separate loading is shown where necessary. It is also computed in "ADVANCE" for use in "CDHMVR". Notation arises because was formerly used for CDH state vector (with CSI being "#1").

$\underline{R}_{act3}$: Value of CSM position vector computed in "PRECSET" for specified input time, scale factor B29, units meters. It is used in the computations associated with TPI (P34, P35, and P36).

$\underline{R}_{af}$: Value of CSM position vector at TPI time computed in "NCCLOOP", scale factor B29, units meters, used in determining altitude error (via "EHCOMP" routine) and PDEP (in "NC12G").

$\underline{R}_{afd}$: Desired CSM position vector at TPI computed in "NC12F", scale factor B29, units meters.

$\underline{R}_{ah1}$: Value of CSM position vector at NC2 point (based on NC and DVC for P31) loaded in "NC1LOOP", scale factor B29, units meters. It is not loaded for P32.

$\underline{R}_{ah2}$: Value of CSM position vector at NCC point (based on NC and DVC for P32 in "NC1LOOP", and DVH1 and C_{nh1} for P31), scale factor B29, units meters.

$\underline{R}_{aprec}$: Value of $\underline{R}_{act3}$ at entrance to "S33/34.1", scale factor B29, units meters. It is used to form a standard initial condition for performance of conic integration in "ADTIME+3" by the amount of time in NOMTPI (to avoid buildup of error due to successive conic computations).

$\underline{R}_{as}$: Value of CSM position vector at NSR point computed in "NCCLOOP", scale factor B29, units meters.

$\underline{R}_{init}$: Value of position vector at start of burn (or of Lambert evaluation), used as a communication cell with "INITVEL", scale factor B29, units meters. If an iteration is required, the time tag for the state vector is in INTIME.

$\underline{R}_{j}$: Value of position vector determined by "QRDTPI" routine, scale factor B29, units meters, stored in push-down list location OD (same cell as $\underline{R}_{att}$, but separate loading shown for documentation convenience). It is the desired OWS vector determined by the routine.

$\underline{R}_{nsr}$: Value of CSM position vector at NSR time computed in "DISP45A", scale factor B29, units meters. It occupies the same cell as $\underline{R}_{act2}$, but for documentation convenience a separate loading of $\underline{R}_{act2}$ is shown where necessary.

$\underline{R}_{other}$: See Orbital Integration.

$\underline{R}_{pass2}$: Value of OWS state vector, scale factor B29, units meters, loaded in "ADVANCE" with $\underline{R}_{pass3}$ and used in "CDHMVR". See $\underline{R}_{act2}$.

$\underline{R}_{pass3}$: Value of OWS state vector, scale factor B29, units meters, loaded in "PRECSET" based on the input time argument. In "DISP45A" loaded with active vehicle state vector instead for use in "NCCDMP7+17" in restoring $\underline{R}_{init}$ (required due to erasable memory assignments).

$\underline{R}_{pc}$: Value of OWS vector at TIMEC, scale factor B29, units meters, computed in "NC12ST". It is subsequently updated in "NC12J" to achieve a "phase match" with the CSM state vector at NC1 time (P31) or NC2 (P32).

$\underline{R}_{pprec}$: Value of $\underline{R}_{pass3}$ at entrance to "S33/34.1", scale factor B29, units meters, used to form a standard initial condition for performance of conic integration in "ADTIME+3" (see $\underline{R}_{aprec}$).

$\underline{R}_{ptpi}$: Communication cell with "QRDTPI" giving OWS position vector at TPI time, scale factor B29, units meters.

$\underline{R}_{targ}$: Communication cell with "INITVEL" containing position of target vector (or vehicle) DELLT4 centi-seconds after the time in INTIME, scale factor B29, units meters. Loaded with updated target vector position (if a change made) before exit from routine.

$\underline{R}_{targl}$: Value of $\underline{R}_{targ}$ used within "INITVEL" package, scale factor B29, units meters.

$\underline{R}_{tig}$: See Burn Control.

R1: See Conic Routines.

R1VEC, R2VEC: See Conic Routines.

RATEMP: Cell used as communication cell with "REVUP", left when exit with time tag of state vector after required advancement, scale factor B28, units centi-seconds. Also used at start of "EHCOMP" to contain magnitude of calling argument vector, scale factor B29, units meters.

RCV: See Orbital Integration.

RdA: See Conic Routines.

RJMAG: Magnitude of $\underline{R}_j$ computed in "GETRJ", scale factor B29, units meters.

RVEC: See Conic Routines.

SECMAX: Value of maximum time increment used in iteration for TPI time for P34/P35, scale factor B28, units centi-seconds. It is initialized to K_{max250} in "S33/34.1".

SNTH: See Conic Routines.

SUBEXIT: Single precision cell used to retain return address information.

T: See Conic Routines.

T_{as}: NSR time loaded in "NCCLOOP", scale factor B28, units centi-seconds. Loaded into T_{cdh} in "NC12OUT".

T_{cdh}: Value of NSR time displayed by N13 (program notation also "TNSR"), scale factor B28, units centi-seconds. Loaded from T_{as} in "NC12OUT" and also initialized (using C_{tcs} and T_{csi}) in "P33" before initial display.

T_{csi}: Value of NCC time displayed by N11 (program notation also "TNCC"), scale factor B28, units centi-seconds. Loaded from TH2 in "NC12OUT".

T_{dec2}: Time to which integration is to be performed by "PRECSET", scale factor B28, units centi-seconds. Used to achieve the same T_{decl} setting for the CSM integration as for the OWS integration.

T_{et}: See Orbital Integration.

T_{etlm}: See Orbital Integration.

T_{pass4}: See Burn Control.

T_{pc}: Time tag associated with $\underline{R}_{pc}$ and $\underline{V}_{pc}$ updating in "NC12J", scale factor B28, units centi-seconds.

T_{tpi}: Value of Transfer Phase Initiation time, scale factor B28, units centi-seconds, displayed by N37. Can be derived in P35 if an elevation angle is specified (iteration starts from specified input time, however), and would also be expected to form part of the prelaunch load.

TDESIRED: See Conic Routines.

TH1: Time tag for NC2 maneuver computed in P31 (loaded in "NC1LOOP" based on "REVUP" results and loaded into NC2TIG for P31 in "NC12OUT"), B28 centi-sec.

TH2: Time tag for NCC maneuver computed in P32 (in "NC1LOOP") and P31 (in "NC2LOOP"), scale factor B28, units centi-seconds. It is loaded into T_{csi} in "NC12OUT". Prior to this use, employed in "PHSMCH"/"NC12J" area to contain the angle between the CSM and OWS vehicles, scale factor BO, units revolutions.

TIMEC: Time tag for the next maneuver loaded before entrance to "NC12ST" with NC1TIG for P31 and NC2TIG for P32, scale factor B28, units centi-seconds.

TITER: Single precision cell used in "S33/34.1" to protect against excessive iterations. Initialized to a negative number, and then reset in "ELCALC" (where used as flag for first pass) to 37777_8. Subsequently shifted right 1 place if DELEL not less than K_{eeps}: the 15th right shift will make the single precision value zero (the double precision value non-zero, serving as an error flag), causing an error exit.

TN: Time used for phase-match iteration, scale factor B28, units centi-seconds (initialized to TIMEC in "NC12ST" and updated in "PHSMCH"/ "NC12J").

ULOS: Value of unit vector, scale factor B1, in line-of-sight direction, i.e. in direction of $(\underline{R}_{pass3} - \underline{R}_{act3})$.

UN: See Conic Routines. It is computed in "INITVEL" as the perpendicular to plane defined by $\underline{R}_{init}$ and $\underline{V}_{init}$, but is written over in Lambert routine (subroutine "GEOM") if the NORMSW (bit 10 of FLAGWRD7) is 0, meaning that the transfer is not too close to 180° as specified by COZY4 information.

UNRM: Value of unit vector, scale factor B1, in direction $(\underline{R}_{act3} * \underline{V}_{act3})$.

UNVEC: Unit vector formed from $\underline{R}_{act2}$ at the start of "CDHMVR", scale factor B1.

UP: Unit vector, scale factor B1, giving direction of local horizontal plane of CSM at T_{tpi} for use in computing ELEV in "ELCALC", stored in push-down list location OD.

UP1: Unit vector, scale factor B1, in direction of $\underline{r} * \underline{v}$ of passive vehicle at ignition time (computed in "PRECSET", "PHSMCH", and "DISP45A" (where is at T_{cdh}, i.e. NSR time)). Cell occupies same location as AM1, but for documentation convenience separate loading shown where necessary.

$\underline{UR}_{pd}$: Vector, scale factor B1, computed in "PHSMCH" to be "colinear with the intersection of the OWS plane and the plane perpendicular to the CSM plane that contains the CSM position vector." It is a unit vector.

URJ: Value of unit$\underline{R}_j$ computed in "GETRJ", scale factor B1.

$\underline{V}_{ac}$: Value of CSM velocity vector at ignition time for next burn, scale factor B7, units meters/centi-second. It is initially computed in "NC12ST", and is rotated into OWS plane in "NC12F".

$\underline{V}_{act2}$: Value of CSM velocity vector, scale factor B7, units meters/centi-second, loaded in "ADVANCE" with velocity rotated into OWS plane.

$\underline{V}_{act3}$: Value of CSM velocity vector computed in "PRECSET" for specified input time, scale factor B7, units meters/centi-second. Also loaded at end of "CDHMVR" with required velocity after maneuver computed there.

$\underline{V}_{ahl}$: Value of CSM velocity vector at NC2 point (see $\underline{R}_{ahl}$), scale factor B7, units meters/centi-second.

$\underline{V}_{ah2}$: Value of CSM velocity vector at NCC point (see $\underline{R}_{ah2}$), scale factor B7, units meters/centi-second.

$\underline{V}_{aprec}$: Value of $\underline{V}_{act3}$ at entrance to "S33/34.1", scale factor B7, units meters/centi-second (see $\underline{R}_{aprec}$).

$\underline{V}_{as}$: Value of CSM velocity vector at NSR point computed in "NCCLOOP", scale factor B7, units meters/centi-second.

$\underline{V}_{asf}$: Coelliptic velocity required at NSR point, scale factor B7, units meters/centi-second, computed in "NCCLOOP". Used to compute VGNS$\underline{R}$ in "NC12OUT".

$\underline{V}_{init}$: Value of velocity vector at start of burn (or of Lambert evaluation), used as a communication cell with "INITVEL", scale factor B7, units meters/centi-second. If iteration is required, the time tag for the state vector is in INTIME.

$\underline{V}_{iprime}$: Value of required velocity computed by Lambert routine (i.e. VVE$\underline{C}$ value when return from routine) at the time INTIME. Scale factor B7, units meters/centi-second.

$\underline{V}_{j}$: Value of velocity vector determined by "QRDTPI" routine, scale factor B7, units meters/centi-second, stored in push-down list location 6D (same cell as $\underline{V}_{att}$, but separate loading shown for documentation convenience).

$\underline{V}_{nsrp}$: Value of velocity at T_{cdh} computed in "DISP45A", scale factor B7, units meters/centi-second, program notation "VNSR+"(after burn).

$\underline{V}_{other}$: See Orbital Integration.

$\underline{V}_{pass2}$: Value of OWS velocity vector (see $\underline{R}_{pass2}$), scale factor B7, units meters/centi-second.

$\underline{V}_{pass3}$: Value of OWS velocity vector, scale factor B7, units meters/centi-second, loaded in "PRECSET" based on the input time argument.

$\underline{V}_{pass4}$: Value of OWS velocity vector at intercept (T_{pass4}) time, scale factor B7, units meters/centi-second.

$\underline{V}_{pc}$: Value of OWS velocity vector updated as described for $\underline{R}_{pc}$, scale factor B7, units meters/centi-second.

$\underline{V}_{pprec}$: Value of $\underline{V}_{pass3}$ at entrance to "S33/34.1", used to form a standard initial condition for performance of conic integration in "ADTIME+3", scale factor B7, units meters/centi-second (see $\underline{R}_{aprec}$).

$\underline{V}_{ptpi}$: Communication cell with "QRDTPI" giving OWS velocity vector at TPI time, scale factor B7, units meters/centi-second.

$\underline{V}_{target}$: Value of CSM velocity at intercept (after doing the burn), scale factor B7, units meters/centi-second. If VTARGTAG is non-zero, the value is computed by precision integration in "INITVEL2"; if VTARGTAG is 0, value computed by "INITV" at the end of Lambert iteration.

$\underline{V}_{tig}$: See Burn Control.

$\underline{V}_{tprime}$: Value of $\underline{V}_{target}$ when return from "INITVEL", scale factor B7, units meters/centi-second. See also Burn Control.

VC$\underline{V}$: See Orbital Integration.

VGNS$\underline{R}$: Velocity to be gained for NSR (P34) maneuver, scale factor B7, units meters/centi-second, displayed by N82. It is loaded in "NC12OUT" and "DISP45A".

VTARGTAG: Single precision cell, scale factor B14, used to store the required number of Lambert iterations. Information originally set in push-down list location OD (called TS_1 in writeup), with a value of zero meaning that no iterations are required.

VVE$\underline{C}$: See Conic Routines.

S40.1 Entered from "P40S/F" (in P40/P41)

Set bit 7(FIRSTFLG) of FLAGWRD2 = 1

$\underline{BDT} = 0$

If bit 8(XDELVFLG) of FLAGWRD2 = 0:

Proceed to "S40.1B"

$DELVSAB = |\underline{DELVSIN}|$

$\underline{V}_{init} = \underline{V}_{tig}$

$\underline{TS}_1 = \text{unit}(\underline{V}_{tig} * \underline{R}_{tig})$ (stored in $\underline{UT}$ cell)

$\underline{R}_{init} = \underline{R}_{tig}$

$TS = K_{thetcon} \left|(\underline{V}_{tig} * \underline{R}_{tig})\right|$ (rounded multiplication, least increment in result 2^{16} m/cs)

$BURNANG = TS \ DELVSAB \ MASS_{dp} / (CAPF \ |\underline{R}_{tig}|^2)$

$\underline{TS}_2 = (\underline{DELVSIN} \cdot \underline{TS}_1) \ \underline{TS}_1$

$\underline{TS}_3 = \underline{DELVSIN} - \underline{TS}_2$ (in-plane component of specified velocity)

$\underline{V}_{gtig} = \underline{TS}_2 + |\underline{TS}_3| \left(\text{unit}(\underline{TS}_3 * \underline{TS}_1) \sin BURNANG + \text{unit}\underline{TS}_3 \cos BURNANG\right)$

$\underline{UT} = \text{unit}\underline{V}_{gtig}$

$\underline{TS} = \underline{V}_{gtig}$

Perform "GET.LVC"

Proceed to "S40.2,3"

S40.1B

$T_{decl} = T_{ig} - K_{twodt}$

$DELLT4 = T_{pass4} - T_{decl}$

Perform "AGAIN"

$\underline{TS}_4 = \underline{V}_{iprime}$ (stored in $\underline{UT}$ cell)

$T_{decl} = T_{ig}$

$DELLT4 = T_{pass4} - T_{decl}$

Perform "AGAIN"

$\underline{V}_{gtig} = \underline{DELVEET3}$

$\underline{TS} = \underline{V}_{gtig}$

Perform "GET.LVC"

$\underline{TS} = \underline{R}_{tig}$

Perform "CALCGRAV"

$\underline{TS} = \text{CSTEER}\left((\underline{V}_{iprime} - \underline{TS}_4)/K_{200cs} - \underline{GDT1}/K_{200cs} \right)$

$\underline{QFCT} = \underline{TS} - (\text{unit}\underline{V}_{gtig} \cdot \underline{TS})\ \text{unit}\underline{V}_{gtig}$

$TS_1 = \text{CAPF}/(\text{MASS}_{dp})$

$\underline{UT} = \text{unit}\left(\underline{QFCT} + \sqrt{TS_1^2 - |\underline{QFCT}|^2}\ \text{unit}\underline{V}_{gtig} \right)$

Proceed to "S40.2,3"

AGAIN

Perform "CSMPREC"

$\underline{R}_{tig} = \underline{R}_{att}$

$\underline{R}_{init} = \underline{R}_{att}$

$\underline{V}_{tig} = \underline{V}_{att}$

$\underline{V}_{init} = \underline{V}_{att}$

$TS_1 = 0$

$TS_2 = K_{eps45}$

If bit 10(NORMSW) of FLAGWRD7 = 0:

$TS_2 = K_{eps10}$

Perform "INITVEL"

Return

S40.2,3 Entered for P40/P41 from "S40.1" and "S40.1B"

$\text{POINTVS}\underline{M} = [\text{REFSMMAT}]\ \underline{UT}$

If bit 11(RCSBURN) of FLAGWRD1 = 1:

Proceed to "S40.2,3B"

$YAWANG = K_{trims}\ YACTOFF + K_{ybias}$

$PITANG = K_{trims}\ PACTOFF + K_{pbias}$

$\underline{X}_{smd}$ = (cos PITANG cos YAWANG, - cos PITANG sin YAWANG, sin PITANG)

$\underline{Z}_{smd}$ = (- sin PITANG cos YAWANG, sin PITANG sin YAWANG, cos PITANG)

$\underline{Y}_{smd}$ = (sin YAWANG, cos YAWANG, 0) Note x,z,y sequence & use below.

$\underline{SCAXIS}$ = (cos PITANG cos YAWANG, sin YAWANG, - sin PITANG cos YAWANG)

$\underline{TS}_1 = \underline{UT}$

$\underline{TS} = -\ unit\underline{R}_{tig} * \underline{UT}$

$\underline{TS}_2 = unit\underline{TS}$

If $|\underline{TS}| < K_{minmag}$:

$$\underline{TS}_2 = -\ unit\left((unit\underline{R}_{tig} + 0.125\ unit\underline{V}_{tig}) * \underline{UT}\right)$$

$\underline{TS}_3 = -\ \underline{TS}_2 * \underline{TS}_1$

$$[TS] = \begin{bmatrix} \underline{TS}_1 \\ \underline{TS}_2 \\ \underline{TS}_3 \end{bmatrix}$$

$\underline{X}_{smd} = \underline{X}_{smd}\ [TS]$

$\underline{Y}_{smd} = \underline{Y}_{smd}\ [TS]$

$\underline{Z}_{smd} = \underline{Z}_{smd}\ [TS]$

Proceed to "P40SXTY"

S40.2.3B

$\underline{SCAXIS} = unit\underline{X}$

$\underline{X}_{smd} = \underline{UT}$

$\underline{TS} = \underline{UT} * \underline{R}_{tig}$

$\underline{TS}_2 = unit\underline{TS}$

If $|\underline{TS}| < K_{minmagr}$:

$$\underline{TS}_2 = -\ unit\left((unit\underline{R}_{tig} + 0.125\ unit\underline{V}_{tig}) * \underline{UT}\right)$$

$\underline{Y}_{smd} = TS_2$

$\underline{Z}_{smd} = -(\underline{Y}_{smd} * \underline{X}_{smd})$

Proceed to "P40SXTY"

UPDATEVG Entered from "CALCN85" and "S40.8"

If bit 8(XDELVFLG) of FLAGWRD2 = 0:

If NBRCYCLS < 0: (set negative as initial condition and at end of "S40.9")

Proceed to "SETUP.9"

DELVSUM = DELVSUM + DELVREF

NBRCYCLP = NBRCYCLS + 1

NBRCYCLS = NBRCYCLP

Proceed to "VGCOMP"

VGCOMP

$\underline{V}_g = \underline{V}_{gtig} + BDT - DELVREF$ ($\underline{V}_{gtig}$ notation also "VGPREV")

VGDISP = $|\underline{V}_g|$

$\underline{V}_{gtig} = \underline{V}_g$

Return (to caller of "UPDATEVG")

SETUP.9

If bit 7(FIRSTFLG) of FLAGWRD2 = 0:

$\underline{V}_{gtig} = DELVEET3 + NBRCYCLP\ BDT - DELVSUM$ ($\underline{V}_{gtig}$ also has notation "VGPREV"; DELVEET3 also "VGTEMP")

Establish "S40.9" (priority 10_8) (Restart protect by group 1.5, see "REDO40.9")

$\underline{R}_{init} = \underline{R}$

$\underline{V}_{init} = \underline{V}$

$T_{nit} = T_{pptm}$

$DELLT4 = T_{pass4} - T_{nit}$

DELVSUM = 0

NBRCYCLS = 0

NBRCYCLP = 0

Proceed to "VGCOMP"

S40.8 Entered due to AVEGEXIT setting in "P40SXTY"

Perform "UPDATEVG"

MASSTMP = CSMMASS

If(|DELVREF| - $C_{dvthresh}$) < 0:

If bit 11(STEERSW) of FLAGWRD2 = 0: (Tag here "LOTHRUST")

Proceed to "SERVEXIT"

If bit 6(IDLEFAIL) of FLAGWRD1 = 1: (set/reset by "V97P")

Proceed to "SERVEXIT"

OMEGAC = 0

REPFRAC = -0

NVWORD1 = -0 (causes "CLOCKJOB" to generate V97 display)

Set bit 11(STEERSW) of FLAGWRD2 = 0

Proceed to "SERVEXIT"

CSMMASS = MASSTMP - C_{emdot} $K_{2secnds}$

If bit 11(STEERSW) of FLAGWRD2 = 0:

Proceed to "SERVEXIT"

If bit 15(SWTOVER) of FLAGWRD9 = 1: (Tag here "CGTRACK")

REPFRAC = $K_{frepfrac}$

Skip next 2 lines

j = (bit 14 of DAPDATR1) (1 for LM on, 0 for LM off)*

REPFRAC = $C_{erepfrac_j}$

TS_1 = BDT - DELVREF (Tag here "TGOCALC")

TS_2 = unitTS_1 · V_g

If $TS_2 \geq 0$:

Perform "ALARM" (pattern 1407_8) (Tag here "INCRSVG")

Proceed to "SERVEXIT"

*NOTE: "LM" references are from Apollo 15 program, and have been retained for mnemonic usefulness.

$T_{go} = K_{mfourdt}\ TS_2\ (1 + TS_2/K_{2vexh}) / (|\underline{TS}_1|) - C_{tdecay}$

$T_{ig} = T_{pptm} + T_{go}$ (for e.g. "CLOKTASK", giving GET of cutoff)

If $(T_{go} - K_{foursec}) < 0$:

Proceed to "S40.81"

$\underline{DELTAM} =$ CSTEER BD$\underline{T}$ - DELVREF

$\underline{TS} =$ [REFSMMAT] (unit$\underline{V}_g$ * unit$\underline{DELTAM}$)

Perform "*SMNB*"

$\underline{OMEGAC} =$ KPRIMEDT $\underline{TS}$

Proceed to "SERVEXIT"

S40.9 (Established if a Lambert burn by "SETUP.9")

$TS_1 = 0$

$TS_2 = K_{ep45}$

If bit 10(NORMSW) of FLAGWRD7 = 0:

$TS_2 = K_{ep10}$

Set bit 2(GUESSSW) of FLAGWRD1 = 0

Perform "INITVEL" (starting at second line) See "REDO40.9" for restarts.

If bit 7(FIRSTFLG) of FLAGWRD2 = 0: (set 1 in "S40.1")

$\underline{BDT} = (\underline{V}_{iprime} - \underline{V}_{rprev}) \left(K_{200cs} / (T_{nit} - T_{nitprev})\right) - \underline{GDT}$

$\underline{DELVEET3} = \underline{DELVEET3} + K_{earthmu} (T_{pptm} - NOMTIG)\ \underline{GOBL} / |\underline{R}|^2$

(DELVEET3 tag "VGTEMP")

$T_{nitprev} = T_{nit}$

$\underline{V}_{rprev} = \underline{V}_{iprime}$

Set bit 7(FIRSTFLG) of FLAGWRD2 = 0

NBRCYCLS = -1

End of job

REDO40.9 Entered if a restart encountered from time of establishing "S40.9" in "SETUP.9" until return from "INITVEL" in "S40.9", due to restart group 1.5

DELVSUM = 0

NBRCYCLP = 0

DELVEET3 = V_{gtig} (for "SETUP.9"restoration)

NBRCYCLS = -1

End of job

S40.13 Established by "TIG-5" and "V97E"

TS_1 = VGDISP (in PD 0D)

SBVGMAG = TS_1 / K_{kpip2} (double precision load, low half

PASTDV = SBVGMAG not meaningful here)

If bit 7(+X translation complement) of channel 31 = 0:

$TS = K_{2rcsf}$ / $MASS_{dp}$

If bit 15(2JETSFLG) of FLAGWRD1 = 0:

TS = 2 TS (i.e. 4-jet translation)

$TS_1 = TS_1 - TS$

$TS_2 = MASS_{dp}\ TS_1\ (1 - TS_1 / K_{2vexh})$

If $|TS_2| \geq 2^{14}$: (or other overflow) (TS_2 units same as $C_{eimplsec}$)

Set bit 3(SBFLAG) of FLAGWRD2 = 0

End of job

$TS_3 = K_{100bml4}\ (TS_2 - C_{eimplsec}) / C_{efimpl6}$ (B14, PD 2D)

If $(TS_3 - K_{500bml4}) \geq 0$:

Set bit 3(SBFLAG) of FLAGWRD2 = 0

End of job

If $TS_3 \geq 0$:

If bit 3(SBFLAG) of FLAGWRD2 = 1: (Tag here "S40.134")

$TS = MASS_{dp} - (TS_3 + K_{100bml4})\ C_{emdot_{dp}}$

$$DVTAIL = \frac{(C_{efimpl6} / K_{100b8})\ C_{tdecay_{dp}}}{K_{kpip2}\ TS}$$

PIPAPI = 0

$TZEROED = T_{pptml}+1$ (i.e. least significant half)

$PASTT = GDT1_{x_{sp}}$ (due to double precision store order)

If $TS_3 < 0$:

$TS_3 = C_{efimpl6}\ TS_3 / C_{efimp01}$ (B14)

Set bit 3(SBFLAG) of FLAGWRD2 = 0

Set bit 9(IMPULSW) of FLAGWRD2 = 1 (Tag here "S40.132")

Force sign agreement of TS_3

$T_{go} = (TS_3 + K_{100bml4})$, rescaled to scale factor B28

End of job

S41.1

Perform "CDUTRIG"

$\underline{TS} = [REFSMMAT]\ \underline{TS}$

Perform "*SMNB*"

$\underline{TS} = K_{tenb4}\ [QUADROT]\ \underline{TS}$ (now in control coordinates)

Return

SBTASK Initially called by "IGNITION"

PREST = TIME1

TS = PREST - TZEROED

If $TS \leq 0$:

TS = (bits 14-1 of TS) + 1 centi-second (compensates for TIME1 overflow) Note Maximum is 163.83 seconds.

TDELTA = TS

VELSS = 0

Perform the following for i = z, y, x:

$$TS = (PIPA_i + PIPAPI_i) + C_{pipascf_i}(PIPA_i + PIPAPI_i) - C_{pipabias_i} TDELTA$$

$$VELSS = VELSS + TS^2$$

If VELSS $\neq$ 0:

Perform "SQRTTASK"

TS = SBVGMAG - (2 PRESDV + DVTAIL) + PASTDV

If TS $\leq$ 0: (same as SBVGMAG - PRESDV - (PRESDV - PASTDV) - DVTAIL)

Proceed to "ENGOFF"

PASTT = PREST (Tag here "BURNMORE")

PASTDV = PRESDV (as "least significant half" of PASTT/PREST)

Call "SBTASK" in 0.10 second

End of task

ENGOFF

TS_1 = PREST - PASTT

If $TS_1 \leq 0$:

TS_1 = (bits 14-1 of TS_1) + 1 centi-second

TDELTA = TS_1

TS_2 = PRESDV - PASTDV

$TS_3 = (TS + TS_2)\ TDELTA / TS_2$ $(TS + TS_2) = (SBVGMAG^2 - PRESDV - DVTAIL)$

If $TS_3 \leq 0$:

Proceed to "ENGINOFF"

$T_{go}+1 = TS_3$

Call "ENGINOFF" in TS_3 centi-seconds

End of task

SQRTTASK

X = VELSS, considered to have scale factor B0

TS = X, normalized to lie between $\frac{1}{4}$ and <1. This is accomplished by shifting X left in increments of 2 shifts, and storing the required number of shifts divided by 2 in VRSSmSL

If $TS \geq \frac{1}{2}$, $TS_2 = K_{hislope}\ TS_{sp} + K_{hibias}$

If $TS < \frac{1}{2}$, $TS_2 = K_{loslope}\ TS_{sp} + K_{lobias}$

$TS_2 = (\frac{1}{2}\ TS_{sp}) / TS_2 + \frac{1}{2}\ TS_2$ (TS_2 single precision)

PRESDV = TS_2, shifted right by VRSSmSL places

Return

Quantities in Computations

See also list of major variables and list of routines

BDT: "Effect of rate of change of required velocity, and gravity, acting during the two-second computing interval, upon velocity-to-be-gained," scale factor B7, units meters/centi-second. Set 0 in "S40.1".

BURNANG: Value of one-half the central angle estimated to be traveled during the External Delta-V burn, scale factor B0, units revolutions, stored in push-down location 14D.

$C_{dvthresh}$: Single precision erasable memory constant, program notation "DVTHRESH", scale factor B-2, units meters/centi-second, giving velocity gate used in "S40.8" for concluding that engine failure has taken place. Program shifts constant right 9 places before use, to scale it B7 meters/centi-second (double precision). In order to avoid improper performance, constant should exceed the velocity increment obtained by ullage (in a 2-second interval).

$C_{eimplsec}$: Single precision erasable memory constant, program notation "EIMP1SEC", scale factor B14, units kilogram-meters/centi-second. When divided by mass of vehicle, would give the SPS impulse velocity acquired in the first second of burn. For a value in pound-seconds, PSEC, the fraction in the cell may be computed as PSEC x 10^{-2} x 9.80665 x 0.45359237 x 2^{-14}, where first term is value, second converts to centi-seconds, third is g, fourth converts from pounds to kilograms, and fifth is scale factor. Constant is used double precision, with least significant half the cell used for $C_{efimp01}$.

$C_{efimp01}$: Single precision erasable memory constant, program notation "EFIMP01", scale factor B14, units kilogram-meters/centi-second second. It gives the slope of the SPS impulse curve from 0 to 1 second. For a value in pound-seconds/second, PNDSSC, the fraction in the cell may be computed as PNDSSC x 10^{-2} x 9.80665 x 0.45359237 x 2^{-14}, where first term is value, second converts to centi-seconds, third is g, fourth converts from pounds to kilograms, and fifth is scale factor. Constant is used double precision, with least significant half the cell used for $C_{efimp16}$.

$C_{efimp16}$: Single precision erasable memory constant, program notation "EFIMP16", scale factor B14, units kilogram-meters/centi-second second. It gives the slope of the SPS impulse curve from 1 to 6 seconds (and may be computed by the formula given for $C_{efimp01}$ above). The constant is used double precision, with least significant half the cell used for C_{adpad} (see Entry Computations). Is equivalent to thrust, of course.

C_{emdot}: Single precision erasable memory constant, program notation "EMDOT", scale factor B3, units kilograms/centi-second, giving the nominal mass flow rate for the SPS engine. The least significant half is assigned to the most significant half of C_{rvar} (see Measurement Incorporation).

$C_{erepfrac_j}$: See Digital Autopilot TVC Routines.

$C_{pipabias}$: See IMU Computations.

$C_{pipascf}$: See IMU Computations.

C_{tdecay}: Single precision erasable memory constant, program notation "ETDECAY", scale factor B14, units centi-seconds, giving value of thrust decay time (i.e. equivalent full-thrust time). It is subtracted from the "raw" time-to-go, and therefore should be a positive number in the memory. Used in "S40.8" s.p. (for SPS burns). Least significant half is $C_{ekprime_0}$ (see Digital Autopilot TVC Routines), used in "S40.13".

CAPF: See Burn Control.

CSMMASS: See Digital Autopilot Interface Routines.

CSTEER: Scalar in cross-product steering law, scale factor B2. It is set to 1.0 in "P40CSM" (provided a Lambert burn specified); otherwise, it has a value of 0.

DAPDATR1: See Digital Autopilot Interface Routines.

DELLT4: See Rendezvous Computations.

DELTAM: Value of (CSTEER BDT - DELVREF), scale factor B7, units meters/centi-second. Its unit vector (scale factor B1) is stored temporarily in the push-down list.

DELVEET3: See Rendezvous Computations (DELVSIN occupies the same cells).

DELVREF: See General Program Control.

DELVSAB: Absolute value of DELVSIN used in "S40.1" in the computation of BURNANG for External Delta-V burns, scale factor B7, units meters/centi-second.

DELVSIN: See Burn Control.

DELVSUM: Value of sum of DELVREF outputs from Average-G computed in "UPDATEVG" for Lambert burn, used to correct V_{gtig} for accelerometer-output velocity gained since "S40.9" established. Scale factor B7, units meters/centi-second. Initialized to 0 in "SETUP.9".

DVTAIL: Single precision value of tail-off impulse computed in "S40.13", scale factor B14, units accelerometer pulses, used in "SBTASK". The initial loading done double precision (least significant half TDELTA).

GDT, GDT1: See General Program Control.

GOBL: See General Program Control.

K_{100b8}: Constant, program notation "100B+8", scale factor B8, units centi-seconds. Value is 100×2^{-8}, corresponding to one second (to convert centi-seconds to seconds).

$K_{100bml4}$: Constant, program notation "100B-14", scale factor B14, units centi-seconds. Value is 100 x 2^{-14}, corresponding to one second.

K_{200cs}: Constant, program notation "200CS", scale factor B12, units centi-seconds. Value is 200 x 2^{-12}, corresponding to 2 seconds.

$K_{500bml4}$: Constant, program notation "500B-14", scale factor B14, units centi-seconds. Value is 500 x 2^{-14}, corresponding to 5 seconds.

K_{2rcsf}: Constant, program notation "S40.135", scale factor B23, nominal value 69.6005183 x 2^{-23}. Value corresponds to 796 K_{frcs2} times 2^{-23}, i.e. the velocity increment (after dividing by MASS) for 7.96 seconds of two-jet RCS thrusting. See Burn Control for K_{frcs2} value. Octal value is 00000_8 04263_8, corresponding to 69.59375 x 2^{-23}. Before original ignition, "S40.13" is entered 5 seconds before ignition time, with last Average-G sample made 5.96 seconds before ignition. Program zeros channel 5 (ullage jets) at end of "IGNITION" routine, 2 seconds after ignition.

$K_{2secnds}$: Constant, program notation "FOURSEC +1", scale factor B13, units centi-seconds. Octal value is 00620_8 00000_8, corresponding to 200 x 2^{-13}, or two seconds.

K_{2vexh}: Constant, program notation "2VEXHUST", scale factor B7, units meters/centi-second. Value is 63.020792 x 2^{-7}, corresponding to 2 x 31.510396 x 2^{-7}, where first term is an equation factor of two, second is the exhaust velocity in meters/centi-second, and the third is the scale factor. Exhaust velocity of 3151.0396 meters/second (10338.05643 fps) corresponds to an I_{sp}, dividing by 9.80665, of about 321.32 seconds, or 20500/63.8.

$K_{earthmu}$: Constant, program notation "EARTHMU", scale factor B36, value -3.986032E10 x 2^{-36}, corresponding to earth $-\mu$ (in units of meters^3/centi-second^2).

K_{epl0}: Constant, program notation "EP4(10)L", scale factor B0, units revolutions. Value is 0.027777777, corresponding to ten degrees.

K_{ep45}: Constant, program notation "EP4(45)L", scale factor B0, units revolutions. Value is 0.125, corresponding to 45 degrees.

K_{epsl0}: Constant, program notation "EP4(10)H", scale factor B0, units revolutions. Value is 0.027777777, the same as K_{epl0} (duplicate storage in "high" part of memory).

K_{eps45}: Constant, program notation "EP4(45)H", scale factor B0, units revolutions. Value is 0.125, the same as K_{ep45}.

$K_{foursec}$: Constant, program notation "FOURSEC", scale factor B28, units centi-seconds. Value is 400 x 2^{-28}, corresponding to four seconds.

$K_{frepfrac}$: See Digital Autopilot TVC Routines.

K_{hibias}: Single precision constant, program notation "HIBIAS", scale factor B1, value 0.4192×2^{-1}. See square root routine in Mathematical Functions.

$K_{hislope}$: Single precision constant, program notation "HISLOPE", scale factor B0, value 0.5884.

K_{kpip2}: Constant, program notation "KPIP2", scale factor B-7, units meters/centi-second. Value is $5.85E\text{-}4 \times 2^{7}$, corresponding to $5.85 \times 10^{-2} \times 10^{-2} \times 2^{7}$, where first term is nominal accelerometer scale factor in units of centimeters/second per count, second converts to meters, third converts to centi-seconds, and fourth is scale factor.

K_{lobias}: Single precision constant, program notation "LOBIAS", scale factor B1, value 0.2974×2^{-1}.

$K_{loslope}$: Single precision constant, program notation "LOSLOPE", scale factor B0, value 0.8324.

$K_{mfourdt}$: Constant, program notation "-FOURDT", scale factor B16, units centi-seconds. Value is -800×2^{-18}, corresponding to $(-1) \times 200 \times 2^{-16}$, where first term is an equation factor, second is computing interval (i.e. 2 seconds), and third is scale factor.

K_{minmag}: See Attitude Maneuvers (equivalent effect achieved to use of explicit constant, by checking most significant half of magnitude for 0).

$K_{minmagr}$: Dummy constant used to show effect of checking for magnitude of vector information, scale factor B30, units meters. Value is equivalent to 2^{-14} (cf. K_{minmag}), or 2^{16} meters.

K_{pbias}: Constant, program notation "PBIAS", scale factor B0, units revolutions. Value is -0.00597222222, corresponding to -2.15 degrees (the "pitch mechanical bias with thrust on"). Hence a PACTOFF input of +2.15 degrees would be considered to line up the bell in pitch along +X.

K_{tenb4}: Constant, program notation "TENBNK14", scale factor B4, value 10×2^{-4}, corresponding to 10. Used to correct for the fact that matrix elements of [QUADROT] are multiplied by 0.1 because of the cycle rate during Attitude Maneuvers.

$K_{thetcon}$: Constant, program notation "THETACON", scale factor B6, value 0.31830989×2^{-8}. Value corresponds to $\frac{1}{2} \times (1/2\pi) \times 2^{-6}$, where first term is an equation factor of one-half (to obtain half the central angle traveled during the burn), second converts from radians to revolutions, and third is scale factor.

K_{trims}: Constant, program notation "TRIMSCAL", scale factor B-13, value 1.07975111×2^{-1}. Value corresponds to $85.41 \times 1/(360 \times 3600) \times 2^{13}$, where first term is bit weight in arc seconds for PACTOFF and YACTOFF, second converts to revolutions, and third is scale factor.

K_{twodt}: Constant, program notation "TWODT", scale factor B28, units centi-seconds. Value is 200 x 2^{-28}, corresponding to 2 seconds.

K_{ybias}: Constant, program notation "YBIAS", scale factor B0, units revolutions. Value is +0.00263888889, corresponding to +0.95 degrees (the "yaw mechanical bias with thrust on"). Hence a YACTOFF input of -0.95 degrees would be considered to line up the bell in yaw along +X.

KPRIMEDT: See Digital Autopilot TVC Routines.

MASS: See Digital Autopilot Interface Routines. Incorrect "double precision".

MASSTMP: Temporary storage for CSMMASS in "S40.8", scale factor B16, units kilograms, used to achieve restart protection there.

NBRCYCLP: Single precision counter, scale factor B14, of the number of cycles of DELVREF information that have been summed in DELVSUM. It is incremented in "UPDATEVG" and reset in "SETUP.9". It is used for restart protection purposes (with NBRCYCLS, which is in the preceding cell), and also to update $\underline{V}_{gtig}$ by the suitable amount of $\underline{BDT}$ for the time elapsed since "S40.9" was established.

NBRCYCLS: Single precision counter, scale factor B14, used in conjunction with NBRCYCLP. At the end of "S40.9", it is set to -1 as a flag to "UPDATEVG" that a new Lambert solution has been obtained. It is also set to an initial condition of -1 in "P40SXTY" (for P40 and P41) before Average-G is started.

NOMTIG: See Burn Control.

NVWORD1: See Burn Control.

OMEGAC: See Digital Autopilot TVC Routines.

PACTOFF: See Digital Autopilot TVC Routines.

PASTDV: Single precision value of PRESDV (accelerometer output information) from previous cycle, scale factor B14, units accelerometer counts. It is initialized in "S40.13", and loaded from PRESDV in "SBTASK".

PASTT: Single precision value of PREST (time tag for PRESDV) from previous cycle, scale factor B14, units centi-seconds, loaded from PREST at end of "SBTASK".

PIPA: See IMU Computations.

PIPAPI: Single precision cumulative sum of accelerometer output, scale factor B14, units accelerometer counts, initialized to zero in "S40.13" and incremented by accelerometer readout in "READACCS".

PITANG: Value of angle in pitch axis used in "S40.2,3", scale factor B0, units revolutions. It is used to calculate the desired position of the SPS engine bell in spacecraft coordinates.

POINTVSM: See Attitude Maneuvers.

PRESDV: Single precision magnitude of accelerometer output (compensated for accelerometer scale factor and bias errors) since PIPAPI zeroed, scale factor B14, units accelerometer counts.

PREST: Single precision time tag associated with PRESDV, scale factor B14, units centi-seconds.

QFCT: Value of quantity computed in "S40.1B", scale factor B-5, units meters/centi-second^2, stored in push-down list location 12D.

[QUADROT]: See Attitude Maneuvers.

$\underline{R}_{init}$: See Rendezvous Computations.

$\underline{R}_{tig}$: See Burn Control.

REPFRAC: See Digital Autopilot TVC Routines.

SBVGMAG: Single precision value of required velocity to be gained (measured at the time of "S40.13" performance), scale factor B14, units accelerometer counts. Loaded double precision, but least significant half subsequently used for DVTAIL.

SCAXIS: See Attitude Maneuvers.

T_{go}: Value of time-to-go computed in "S40.8" (and "S40.13"), scale factor B28, units centi-seconds. Least significant half loaded with required delay before cutoff (for restart protection purposes), scale factor B14.

T_{nit}: Value of time tag associated with state vector loaded by "SETUP.9", scale factor B28, units centi-seconds.

$T_{nitprev}$: Value of T_{nit} for previous entry into "S40.9", loaded at the end of "S40.9", scale factor B28, units centi-seconds. Used with T_{nit} to determine the proper divisor (with K_{200cs}) for difference of required velocities, in computation of BDT. Since accelerometer sampling at 2 second intervals, value used to divide velocity difference would be expected to be an integer (e.g. 2 if four seconds elapsed). Time difference shifted left 17 places after being formed, giving result modulo 2^{11} centi-seconds (20.48 seconds).

T_{pass4}: See Burn Control.

T_{pptml}: See IMU Computations.

TDELTA: Time interval information (corrected for possible overflow of TIME1, the least significant half of the computer clock), scale factor B14, units centi-seconds, single precision.

TZEROED: Value of least significant half of T_{pptml} sampled in "S40.13" giving the value of TIME1 associated with the zeroing of PIPAPI (since PIPA zeroed when T_{pptml} is loaded). It is single precision, scale factor B14, units centi-seconds.

$\underline{UT}$: Unit vector, scale factor B1, in the direction of desired thrust (expressed in reference coordinates). Cell also used for temporary storage purposes.

$\underline{V}_g$: Velocity-to-be-gained vector, scale factor B7, units meters/centi-second, in reference coordinates.

$\underline{V}_{gtig}$: See Burn Control (during Average-G running, retains the previous value of $\underline{V}_g$).

$\underline{V}_{init}$: See Rendezvous Computations.

$\underline{V}_{iprime}$: See Rendezvous Computations.

$\underline{V}_{rprev}$: Value of previous $\underline{V}_{iprime}$ (i.e. the one associated with $T_{nitprev}$), scale factor B7, units meters/centi-second.

$\underline{V}_{tig}$: See Burn Control.

VELSS: Square of "net" accelerometer output, including effects of scale factor and bias error compensation, scale factor B28, units of accelerometer counts^2. The square root is PRESDV.

VGDISP: See Burn Control.

VRSSmSL: Single precision counter, scale factor B14, units counts, used in "SQRTTASK" to count the number of pairs of shifts required for X in order to make it be in range $\frac{1}{4}$ to < 1.

X: Dummy argument used in "SQRTTASK" for documentation convenience (cf. square root routine in Mathematical Functions). The VELSS cells themselves are used to store X in program.

$\underline{X}_{smd}$, $\underline{Y}_{smd}$, $\underline{Z}_{smd}$: See Inflight Alignment. Program notation also $\left[X_{scref}\right]$.

YACTOFF: See Digital Autopilot TVC Routines.

YAWANG: Value of angle in yaw axis used in "S40.2,3", scale factor B0, units revolutions. Cf. PITANG.

DODOWNTM — Entered after receipt of program interrupt #8, telemetry end pulse (from telemetry system)

If bit 7(Word Order Code) of channel 13 = 0:

Perform "C13STALL"

Set bit 7(Word Order Code) of channel 13 = 1

Proceed to address specified by DNTMGOTO

DNPHASE1 — Initial condition set in "STARTSUB" for DNTMGOTO

SUBLIST = -1

DNECADR = -1

DNTMGOTO = "DNPHASE2"

CTLIST = $K_{dntable_{DNLSTCOD}}$ (Tag here is "NEWLIST")

Perform "C13STALL"

Set bit 7(Word Order Code) of channel 13 = 0

Channel 34 = - DNLSTCOD

Channel 35 = 77340_8 (i.e. - 00437_8)

Resume

DNPHASE2

If DNECADR > 0: (i.e. sending data)

Proceed to "FETCH2WD"

If SUBLIST > 0: (i.e. sending sublist)

Proceed to "NEXTINSL"

If CTLIST ≤ 0: (End of list reached, start over)

Proceed to 4th line of "DNPHASE1"

ADR = E_{CTLIST} (Tag here "NEXTINCL", get here e.g. next interrupt after doing "DNPHASE1")

If ADR > 0:

CTLIST = CTLIST + 1

If ADR < 0:

CTLIST = - CTLIST (end of list reached)

DNECADR = |ADR|

If DNECADR = "TIME2": (i.e. computer clock)

Perform "C13STALL"

Set bit 7(Word Order Code) of channel 13 = 0

Proceed to "SETWO+2"

SETWO+2

TS = DNECADR - 30000_8

If TS ≤ 0: (i.e. bits 14-12 of DNECADR less than 6)

Proceed to "FETCH2WD"

TS = TS - 04000_8

If TS > 0: (i.e. bits 14-12 of DNECADR are 7)

TS_1 = Channel # (DNECADR - 34000_8 + 1) (Tag here "DODNCHAN")

TS_2 = Channel # (DNECADR - 34000_8) (34000_8 subtracts bits 14-12 = 7)

DNECADR = -1

Channel 34 = TS_2

Channel 35 = TS_1

Resume

SUBADR = $E_{DNECADR - 30000_8}$ (bits 14-12 of DNECADR are 6; tag here is "DODNPTR")

If SUBADR > 0:

SUBLIST = DNECADR

Proceed to "NEXTINSL"

SUBLIST = DNECADR (SUBADR negative, meaning snapshot)

SUBADR = |SUBADR| - 1

TMINDEX = 0

Proceed to "SNAPLOOP"

SNAPLOOP

EBANK = bits 11-9 of SUBADR

TS = 1401_8 + (bits 8-1 of SUBADR) (1401_8 instead of 1400_8 because of decrement in original setting of SUBADR)

$DNTMBUFF_{TMINDEX} = E_{TS_{dp}}$

TMINDEX = TMINDEX + 2

SUBLIST = SUBLIST + 1

$SUBADR = E_{SUBLIST - 30000_8}$

If SUBADR > 0:

    SUBADR = SUBADR - 1

    Proceed to "SNAPLOOP"

SUBLIST = |SUBADR| - 1

DNECADR = -1

TS = SUBLIST

SUBLIST = -1

EBANK = bits 11-9 of TS

TS = 1401_8 + (bits 8-1 of TS)

(Channel 34, Channel 35) = $E_{TS_{dp}}$

Resume

FETCH2WD

EBANK = bits 11-9 of DNECADR

TS = (bits 8-1 of DNECADR)

DNECADR = DNECADR + 74001_8 (subtracts 1 from bits 14-12 and adds 2, for double precision, to bits 11-1: $74001_8 = - 04000_8 + 00002_8$)

TS = 1400_8 + TS

(Channel 34, Channel 35) = $E_{TS_{dp}}$

Resume

NEXTINSL

SUBADR = $E_{SUBLIST - 30000_8}$

If SUBADR > 0:

SUBLIST = SUBLIST + 1

If SUBADR < 0:

SUBLIST = -1 (end of list reached)

DNECADR = |SUBADR|

Proceed to "SETWO+2"

DNDUMPI Entered by the setting of DNTMGOTO to "DNDUMPI" in "DNEDUMP" for a V74E

DUMPLOC = 0

Perform "SENDID"

DNTMGOTO = "DNDUMP1"

Channel 34 = DUMPLOC

Channel 35 = TIME1

Resume

SENDID

DNTMGOTO = Return address

Perform "C13STALL"

Set bit 7(Word Order Code) of channel 13 = 0

Channel 34 = 1777_8

Channel 35 = 77340_8

Resume

DNDUMP1

DNTMGOTO = "DNDUMP"

EBANK = bits 11-9 of DUMPLOC

TS = 1400_8 + (bits 8-1 of DUMPLOC)

(Channel 34, Channel 35) = $E_{TS_{dp}}$ (Read by mask using -0 to avoid affecting shift register cells, 0020_8 - 0023_8)

Resume

DNDUMP

DUMPLOC = DUMPLOC + 2 (modulo 2^{14})

If bits 8-1 of DUMPLOC $\neq$ 0:

Proceed to second line of "DNDUMP1"

If bit 13 of DUMPLOC = 1: (2 complete dumps)

Proceed to "DNPHASE1"

Proceed to second line of "DNDUMPI"

VAC5STOR Entered from "BAILOUT", "GOPROG", and "POODOO" to save data in VAC area #5 for ground checking (after e.g. V74E)

$VAC5+3i_{dp}$ = $(LOC, BANKSET)_i$ (i = 0-6) Job register set starting addresses

$VAC5+2+3i$ = $PRIORITY_i$ (i = 0-6) Job priorities

$VAC5+19+2i_{dp}$ = (PHSNAMEi) (i = 1-6) 2CADR variable-type restarts starting addresses

VAC5+32+i = PHASEi (i = 1-6) Restart-group phase values

VAC5+39 = MPAC+3 Used in display interface routines for return address

$VAC5+40_{dp}$ = NEWLOC Contains starting address for job being established

VAC5+22 = NEWJOB Control of job selection

VAC5+26 = NEWPRIO Priority of new job

Return

NOTE: See 3420.5-27 for details of the significance of the above quantities. They are intended to be an aid to determining "what happened" when other methods (such as flag words or program alarms) give insufficient detail. Use of these quantities would require access to the program listing, and a detailed understanding of the coding implementation. Such details (absolute machine addresses, and all restart points) can be obtained from the listing as necessary; they are not included in this document.

Quantities in Computations

See also list of major variables and list of routines

ADR: Single precision dummy cell used for temporary storage of the address information taken from the cell whose address is in CTLIST (i.e. the "control", or master, telemetry list). If it is read as negative, this means that the end of the master list (the final entry) has been reached. When the necessary number of telemetry transmissions have taken place based on this last entry, the list is started again (CTLIST is complemented if ADR is found negative, for control in "DNPHASE2").

CTLIST: Single precision cell initialized in "DNPHASE1" with the starting address for the "control", or master, telemetry list, and subsequently incremented in "DNPHASE2" as quantities are read from this list. When the final quantity is read, CTLIST is complemented (see ADR) to cause list to be restarted again.

DNECADR: Single precision erasable memory cell used to contain the address (and other) information associated with the transmission of a series of words based upon a single entry into the control or sublist table. Bits 8-1 give the S-register portion of the word in E-memory (the most significant half), while bits 11-9 give the EBANK. If bits 14-12 are in the range 0-5, they control the transmission of 1-6 respectively consecutive erasable memory pairs. While this transmission is going on, DNECADR will be positive, causing branching at the start of "DNPHASE2" to "FETCH2WD", where bits 14-12 are decremented by 1 and bits 11-1 incremented by 2 (for double precision operation). When final cell has been sent based on a given table entry, DNECADR becomes negative, allowing further checks in "DNPHASE2".
If bits 14-12 of DNECADR are 6, this means a "sublist" (i.e. a list of telemetry quantities that can be common to several master lists, hence serving as a quasi "sub-routine" list), with bits 11-1 giving its starting address. Finally, if these bits are 7, they indicate transmission of channel information.

DNLSTCOD: Single precision cell, scale factor B14, loaded by various mission programs (e.g. in V37 processing, transition to P11 or P62, or start of P27) with the desired serial number (in range 0-3) of the master downlist to be sent. It is used in "DNPHASE1" to index $K_{dntable}$ to select the proper starting address, and for list id.

DNTMBUFF: Set of 12 erasable memory cells which can be loaded in "SNAPLOOP" with values of quantities (such as vehicle state vector) requiring sampling at a single point in the computation cycle. A "snapshot" is indicated by bits 14-12 of DNECADR being 6 (a sublist) and the first word in that list being negative. Looping continues in "SNAPLOOP" until the next negative word is found: this word is not used for address information to be placed into DNTMBUFF, but instead is used to read the memory directly and send the indicated word. Consequently, 7 double precision cells, e.g. position, velocity, and time, can be subject to the "snapshot" process at a given telemetry interrupt.

DNTMGOTO: Single precision cell containing address to which transfer is to be made when a telemetry interrupt (#8) is received. Except for initialization, it would be expected to remain at "DNPHASE2" for normal (i.e. not erasable memory dump) downlists.

DUMPLOC: Single precision cell (assigned same erasable memory cell as TMINDEX) used to control the performance of the erasable memory dump program. Initialized to zero at the start of "DNDUMPI", and subsequently incremented by 2 in "DNDUMP" to permit selection of the next set of erasable memory cells. Bits 8-1 (000_8 - 377_8) select the cell within the erasable memory bank, since each bank has 256 cells. Bits 11-9 (carries propagate from bit 8, of course) select the erasable memory bank (0 - 7). Bits 13-12 serve as a counter of the number of complete erasable dumps which have been performed (when bit 13 becomes 1, indicating 2 dumps, dumps cease).

EBANK: See Data Input/Output.

$K_{dntable_i}$: Table of single precision starting addresses for master telemetry downlists, used in "DNPHASE1" to load the proper initial conditions for CTLIST based upon the present value of DNLSTCOD. Program notation for first table entry is "DNTABLE". Values are:

i	Starting Address	Identification of List Data
0	"CMCSTADL"	Coast and Align
1	"CMENTRDL"	Entry and Update
2	"CMRENDDL"	Rendezvous & Prethrust
3	"CMPOWEDL"	Powered

SUBADR: Single precision dummy cell used for temporary storage of the address information taken from the cell whose address is in SUBLIST (i.e. a "sublist", see DNECADR). If it is read as negative, this means that the end of the sublist has been reached. When the necessary number of telemetry transmissions based on this last entry have taken place, the master list is used again (SUBLIST is set negative in "NEXTINSL", for control in "DNPHASE2"). Function is analogous to ADR's for a master list.

SUBLIST: Single precision cell initialized in "SETWO+2" to DNECADR for a non-snapshot list (for bits 14-12 of DNECADR = 6), giving then the starting address of the required sublist. While positive, "DNPHASE2" will transfer to "NEXTINSL" whenever DNECADR becomes negative: "NEXTINSL" increments SUBLIST until a negative readout (see SUBADR) is encountered, when SUBLIST is set to -1 to cause "DNPHASE2" to return to the master list. Function is analogous to CTLIST's for a master list. In "SNAPLOOP", it is used to index the readout of the snapshot sublist, and is left at -1 when "SNAPLOOP" is done to cause "DNPHASE2" to take next entry from master list (which should be a requirement to transmit DNTMBUFF cells).

TMINDEX: Single precision cell, scale factor B14, initialized to 0 in "SETWO+2" for use in indexing DNTMBUFF storage in "SNAPLOOP". It is assigned the same cell as DNECADR.

VAC5: First address in VAC area #5 (see 3420.5-27), the last to be assigned, and hence used in "VAC5STOR" to retain trouble-shooting information.

Information on the downlists appears on the following pages:

Telemetry Table Interpretation

To satisfy mission requirements, four different sets of downlinked memory cells can be specified. The mission program achieves the required specification by setting a quantity between 0 and 3 into DNLSTCOD (usually as part of the processing of a V37 program change). This quantity is used to index a table of fixed memory addresses in order to find the starting address of the particular list of downlinked memory cells required (powered flight, coast and align, etc.). In order to minimize the amount of fixed memory required for storage of this information, and maximize the flexibility of the information that is telemetered, a special storage format for the required addresses is used.

The table whose starting address is found by using DNLSTCOD is known as the "control" (or master) downlist. There is a separate such list for each different downlink set of information, and except for the first word in each 100-word set, the transmission of the information in the downlink set is under the control of information in this control list.

Within this control list, several different types of options can be employed:

a) A single (double precision, i.e. 2 consecutive erasable memory cells) downlink transmission can be specified. This is done by storing the ECADR (see 3420.5-27) of the first word in bits 11-1 of the fixed memory cell. The assembler operation is "1DNADR".

b) Two downlink transmissions (giving four cells in a row) can be specified. This is done by storing the ECADR for the first word in bits 11-1 of the fixed memory cell and putting a 1 in bit 12. The assembler operation in this case is "2DNADR".

c) Three downlink transmissions (giving six cells in a row) can be specified. Here again, the ECADR of the first word is in bits 11-1 of the fixed memory cell and a 1 is put in bit 13. The assembler operation in this case is "3DNADR".

d) Four, five, or six (giving 8, 10, or 12 cells in a row) can also be specified. The ECADR of the first word is in bits 11-1 of the fixed memory cell and bits 14-12 contain the (number - 1) of the transmissions desired: bit 14 is a 1 and bit 12 is a 1 for six transmissions, for example. The assembler operations are nDNADR.

e) A "sublist" can be specified. This permits a sequence of cells, such as state vector information, that may be common to more than one list to be specified only once (a sort of quasi "sub-routine" capability). In this case, bits 11-1 contain the address of the sublist (S-register portion, since in same bank) and bits 14-12 contain 6 (assembler operation is "DNPTR"). This particular assignment makes the 15-bit memory word of the form 3XXXX: rather than subtracting or masking out the "3" (as

indicated in this writeup), the coding takes advantage of the fact that this is the proper form for a "clear add" instruction (hence merely indexes on 00000_8).

f) A single downlink transmission of a pair of channels can be specified. In this case, bits 6-1 would contain the channel number and bits 14-12 would contain 7. The assembler operation for this is "DNCHAN", and advantage is taken of the "7" by indexing to form the required order, rather than subtracting or masking it out as indicated in this writeup.

g) The end of the control list (indicating that the list should be started again) is flagged by having the cell be negative (e.g. "-6DNADR" instead of "6DNADR").

If, per item e, a "sublist" in specified, then two options are available:

1. A "normal" sublist, in which case information can be stored in a manner exactly like the control list, except that option "e" should not be used (sublists cannot reference other sublists). The end of the sublist is flagged by a negative cell (as in option "g"), indicating in this case that information should again be taken from the control (master) list (at the line after the DNPTR).

2) A "snapshot" sublist, flagged by the fact that the first item in the sublist is negative (i.e. "-1DNADR"). The "snapshot" feature allows up to seven double precision words to be sampled at the same telemetry interrupt point, to achieve on the downlink a consistency of this information (e.g. position, velocity, and time of state vector). In this case, the remaining items in the sublist must be "1DNADR" form; the cell data specified by the first table entry through the next-to-last are stored in a special telemetry buffer (DNTMBUFF). The last entry in the sublist is negative, and in this case only is the first cell address transmitted. In all other cases, addresses in the lists are stored in the order in which they are transmitted. In order to cause transmission of the DNTMBUFF cells, the entry in the control list following the DNPTR cell for the snapshot should require transmission of DNTMBUFF cells (e.g. "6DNADR DNTMBUFF").

Except for specification of snapshot lists, which can occur only by DNPTR orders in the master (control) list, there is no maximum to the number of individual entries in either the control or sublists, except of course the convention that a complete telemetry cycle involves 100 double precision words. In addition, downlink processing format requires that word #51 (the computer clock) have a word order code bit of 0: a check for transmission of this word is made only for readouts from the master list (address 0024_8).

Information in Telemetered Words

There are four different downlists that may be transmitted during the flight (plus the special erasable memory dump initiated by a V74E). These four lists are implemented in the computer memory as the addresses of cells in erasable memory: the contents of the cells, in turn, can depend upon the phasing of the telemetry interrupts with respect to the other computations. In addition, cells are frequently time shared among different mission phases. The information below should be augmented by more detailed material on the telemetry data for items not covered.

Unless otherwise specified, the contents of the words below reflect both Channel 34 and Channel 35 (i.e. a double precision number). In those cases for which the information in the two channels is not closely related, "a" signifies the Channel 34 information and "b" the Channel 35 information.

Coast and Align List

This list, with starting address of "CMCSTADL", is selected for DNLSTCOD = 0. It is used in P00, P01, P02, P03, P06, P07, P50, P51, P52, P53, P54, and P55.

Word #	Quantity	Meaning
List Identification		
1a	77777_8	Identification of list (-0).
1b	77340_8	Special synchronization bits.
Snapshot Group #1		
		Words 2-8 are all sampled at the same telemetry interrupt time, with words 3-8 stored in DNTMBUFF for subsequent transmission. Cells for these words are those originally sampled.
2-4	$\underline{R}$	Position state vector (word 2 x component).
5-7	$\underline{V}$	Velocity state vector (word 5 x component).
8	T_{pptm}	State vector time.
Snapshot Group #2		
		Words 9-13 are all sampled at the same telemetry interrupt time, with words 10-13 stored in DNTMBUFF for subsequent transmission.

Word #	Quantity	Meaning
9, 10a	CDU	IMU CDU angles.
10b	CDUT	See Optics Computations.
11-13	ADOT	See Digital Autopilot RCS Routines (same cell used for e.g. TVC DAP observed rates).
Erasable Group #1		
		Words 14-44 are sampled as the telemetry interrupt for them is received.
14,15a	AK	See Digital Autopilot Interface Routines.
15b	RCSFLAGS	See Digital Autopilot Interface Routines.
16,17a	THETADX	See Digital Autopilot RCS Routines (these are not N22 quantities).
17b	$DELCDUX_{sp}$	See Attitude Maneuvers.
18	T_{ig}	Ignition time (or predicted cutoff time).
19a	BESTI	See Inflight Alignment.
19b	BESTJ	See Inflight Alignment.
20-23a	MARKDOWN	Cells MARKDOWN+0 to MARKDOWN+6 (see Measurement Incorporation).
23b	RM	See Measurement Incorporation.
24-27a	MARK2DWN	Cells MARK2DWN+0 to MARK2DWN+6 (see Inflight Alignment).
27b	$BVECTOR_{0_{y,sp}}$	See Measurement Incorporation.
28	HAPOX	See Display Computations.
29	HPERX	See Display Computations.
30a	PACTOFF	See Digital Autopilot TVC Routines.
30b	YACTOFF	See Digital Autopilot TVC Routines.
31-33	$\underline{V}_{gtig}$	See Burn Control.
34-36	$REFSMMAT\underline{}_0$	First row of [REFSMMAT].
37-39	$REFSMMAT\underline{}_3$	Second row of [REFSMMAT].

Word #	Quantity	Meaning
40-44	FLAGWRD0-FLAGWRD9	Program control flag words.
Display Table Group		
		Words 45-50 are sampled as the telemetry interrupt for them is received.
45-50	DSPTAB+0-DSPTAB+11	"Display table" information (DSPTAB+0 is 45a). See Data Input/Output.
Computer Clock		
		Word 51 is sampled when the telemetry interrupt for it is received.
51	T_{now}	Present value of computer clock (TIME2,TIME1).
Snapshot Group #3		
		Words 52-58 are all sampled at the same telemetry interrupt time, with words 53-58 stored in DNTMBUFF for subsequent transmission.
52-54	$\underline{R}_{other}$	See Orbital Integration.
55-57	$\underline{V}_{other}$	See Orbital Integration.
58	T_{etlm}	See Orbital Integration.
Snapshot Group #4		
		Words 59-63 are all sampled at the same telemetry interrupt time, with words 60-63 stored in DNTMBUFF for subsequent transmission.
59,60a	CDU	IMU CDU angles.
60b	CDUT	See Optics Computations.
61-63	ADOT	See Digital Autopilot RCS Routines (roll, pitch, yaw respectively).
Erasable Group #2		
		Words 64-90 are sampled as the telemetry interrupt for them is received.
64	OPTION1,2	See Display Interface Routines.
65	T_{et}	See Orbital Integration.

Word #	Quantity	Meaning
66,67a	THETADX	See Digital Autopilot RCS Routines.
67b	$DELCDUX_{sp}$	See Attitude Maneuvers.
68	$RSBBQ_{dp}$	See General Program Control.
69a	Channel 76	Not meaningful.
69b	Channel 77	Hardware restart information.
70a	C31FLWRD	See Digital Autopilot RCS Routines.
70b,71	FAILREG	FAILREG+0 to FAILREG+2 (see General Program Control).
72a	CDUS	See Optics Computations.
72b,73	PIPA	See IMU Computations.
74	OGC	See Coordinate Transformations.
75	IGC	See Coordinate Transformations.
76	MGC	See Coordinate Transformations.
77a	FLGWRD10	Program control flag word.
77b	FLGWRD11	Program control flag word.
78	T_{evt}	Event (e.g. liftoff or engine on/off) time.
79	LAUNCHAZ	See Prelaunch Alignment.
80a	OPTMODES	See Optics Computations.
80b	HOLDFLAG	See Digital Autopilot Interface Routines.
81a	LEMMASS	See Digital Autopilot Interface Routines.
81b	CSMMASS	See Digital Autopilot Interface Routines.
82a	DAPDATR1	See Digital Autopilot Interface Routines.
82b	DAPDATR2	See Digital Autopilot Interface Routines.
83,84a	ERROR	See Digital Autopilot RCS Routines.
84b	THETADX	See Digital Autopilot RCS Routines.
85-87	WBODY	See Digital Autopilot RCS Routines (same cells used for commanded rates in TVC).

Word #	Quantity	Meaning
88a	REDOCTR	See General Program Control.
88b,89	THETAD	Desired CDU angles (e.g. N22).
90a	IMODES30	See IMU Computations.
90b	IMODES33	See IMU Computations.
Channel Quantities		
		Words 91-94 are sampled as the telemetry interrupt for them is received.
91a	Channel 11	Outputs.
91b	Channel 12	Outputs.
92a	Channel 13	Outputs.
92b	Channel 14	Outputs.
93a	Channel 30	Inputs.
93b	Channel 31	Inputs.
94a	Channel 32	Inputs.
94b	Channel 33	Inputs.
Erasable Group #3		
		Words 95-100 are sampled as the telemetry interrupt for them is received.
95,96a	T_{eph}	See Boost Computations.
96b	$R_{other_{x,sp}}$	See Orbital Integration.
97a	SLOPE	See Digital Autopilot RCS Routines.
97b	ADB	See Digital Autopilot RCS Routines.
98a	DAPDATR3	See Digital Autopilot Docked Jet Selection.
98b	CH5FAIL	See Digital Autopilot Docked Jet Selection.
99a	CH6FAIL	See Digital Autopilot Docked Jet Selection.
99b	DKRATE	See Digital Autopilot RCS Routines.
100a	DKDB	See Digital Autopilot RCS Routines.
100b	WHICHDAP	See Digital Autopilot RCS Routines.

Entry and Update List

This list, with starting address of "CMENTRDL", is selected for DNLSTCOD = 1. It is used in P27, P62, P63, P64, P65, P66, and P67.

Word #	Quantity	Meaning
List Identification		
1a	77776_8	Identification of list (-1).
1b	77340_8	Special synchronization bits.
Snapshot Group #1		
		Words 2-8 are all sampled at the same telemetry interrupt time, with words 3-8 stored in DNTMBUFF for subsequent transmission.
2-4	R	Position state vector.
5-7	V	Velocity state vector.
8	T_{pptm}	State vector time.
Snapshot Group #2		
		Words 9-13 are all sampled at the same telemetry interrupt time, with words 10-13 stored in DNTMBUFF for subsequent transmission.
9, 10a	CDU	IMU CDU angles.
10b	CDUT	See Optics Computations.
11-13	ADOT	See Digital Autopilot RCS Routines (same cell used for e.g. TVC DAP observed rates). When Entry DAP is turned on, cells used for XPIPBUF and XOLDBUF (see General Program Control).
Erasable Group #1		
		Words 14-44 are sampled as the telemetry interrupt for them is received.
14,15a	AK	See Digital Autopilot Interface Routines.
15b	RCSFLAGS	See Digital Autopilot Interface Routines.

Word #	Quantity	Meaning
16,17a	THETADX	See Digital Autopilot RCS Routines.
17b	$DELCDUX_{sp}$	See Attitude Maneuvers. In entry, word 17 contains Q7 (see Entry Computations) and word 16 contains (QAXERR, RAXERR) (see Digital Autopilot Entry Routines).
18a	CMDAPMOD	See Digital Autopilot Entry Routines.
18b	PREL	See Digital Autopilot Entry Routines.
19a	QREL	See Digital Autopilot Entry Routines.
19b	RREL	See Digital Autopilot Entry Routines.
20	LdD1	See Entry Computations.
21-30	UPBUFF	See Uplink Processing (UPBUFF+0 through UPBUFF+19). Same cells used during entry DAP for CMTMTIME, SWdNDX, and ENDBUF (see Digital Autopilot Entry Routines). Words 29b and 30a are also V1 (see Entry Computations), and 30b AO_{sp} (see Entry Computations).
31a	COMPNUMB	See Uplink Processing.
31b	UPOLDMOD	See Uplink Processing.
32a	UPVERB	See Uplink Processing.
32b	UPCOUNT	See Uplink Processing.
33a	PAXERR1	See Digital Autopilot Entry Routines.
33b	ROLLTM	See Digital Autopilot Entry Routines.
34	LATANG	See Entry Computations.
35	RDOT	See Entry Computations.
36	THETAH	See Entry Computations.
37	LATSPL	See Display Computations.
38	LNGSPL	See Display Computations.
39a	ALFAd180	See Entry Computations.
39b	BETAd180	See Entry Computations.
40-44	FLAGWRD0-FLAGWRD9	Program control flag words.

Word #	Quantity	Meaning
Display Table Group		
		Words 45-50 are sampled as the telemetry interrupt for them is received.
45-50	DSPTAB+0-DSPTAB+11	"Display table" information. See Data Input/Output.
Computer Clock		
		Word 51 is sampled when the telemetry interrupt for it is received.
51	T_{now}	Present value of computer clock (TIME2,TIME1).
Snapshot Group #3		
		Words 52-58 are all sampled at the same telemetry interrupt time, with words 53-58 stored in DNTMBUFF for subsequent transmission.
52	T_{pptml}	See IMU Computations.
53-55	DELV	Sampled accelerometer output (if least significant halves = 0, is uncompensated; otherwise is compensated data).
56	TTE	See Display Computations. Also used for LdDCALC (see Entry Computations).
57	VIO	See Display Computations. Also used for LEWD (see Entry Computations).
58	VPRED	See Display Computations. Also used for VL (see Entry Computations).
Snapshot Group #4		
		Words 59-63 are all sampled at the same telemetry interrupt time, with words 60-63 stored in DNTMBUFF for subsequent transmission.
59,60a	CDU	IMU CDU angles.
60b	CDUT	See Optics Computations.
61-63	ADOT	See Digital Autopilot RCS Routines. See also words 11-13 above in this list (which are same cells).

Word #	Quantity	Meaning
Erasable Group #2		
		Words 64-90 are sampled as the telemetry interrupt for them is received.
64	OPTION1,2	See Display Interface Routines.
65	T_{et}	See Orbital Integration.
66, 67a	ERROR	See Digital Autopilot RCS Routines.
67b,68	THETADX	See Digital Autopilot RCS Routines. Words 66-68 also contain (single precision): VDTd180, mVTd180E,unreflected LCXd360, QAXERR, RAXERR, and $Q7_{sp}$ (see Entry Computations for Q7 and Digital Autopilot Entry Routines for the others).
69a	CMDAPMOD	See Digital Autopilot Entry Routines.
69b	PREL	See Digital Autopilot Entry Routines.
70a	QREL	See Digital Autopilot Entry Routines.
70b	RREL	See Digital Autopilot Entry Routines.
71-80	UPBUFF	See Uplink Processing (UPBUFF+0 through UPBUFF+19). See also words 21-30 above (same cells).
81a	LEMMASS	See Digital Autopilot Interface Routines.
81b	CSMMASS	See Digital Autopilot Interface Routines.
82a	DAPDATR1	See Digital Autopilot Interface Routines.
82b	DAPDATR2	See Digital Autopilot Interface Routines.
83a	ROLLTM	See Digital Autopilot Entry Routines.
83b	$ROLLC_{sp}$	See Entry Computations.
84a	OPTMODES	See Optics Computations.
84b	HOLDFLAG	See Digital Autopilot Interface Routines.
85-87	WBODY	See Digital Autopilot RCS Routines. Same cells used for commanded rates in TVC. They are also loaded with ASPSpTMp (see Entry Computations).
88a	REDOCTR	See General Program Control.

Word #	Quantity	Meaning
88b,89	THETAD	Desired CDU angles (e.g. N22). Same cells for RDOTREF and $VREF_{sp}$ (see Entry Computations).
90a	IMODES30	See IMU Computations.
90b	IMODES33	See IMU Computations.
Channel Quantities		
		Words 91-94 are sampled as the telemetry interrupt for them is received.
91a	Channel 11	Outputs.
91b	Channel 12	Outputs.
92a	Channel 13	Outputs.
92b	Channel 14	Outputs.
93a	Channel 30	Inputs.
93b	Channel 31	Inputs.
94a	Channel 32	Inputs.
94b	Channel 33	Inputs.
Erasable Group #3		
		Words 95-100 are sampled as the telemetry interrupt for them is received.
95	$RSBBQ_{dp}$	See General Program Control.
96a	Channel 76	Not meaningful.
96b	Channel 77	Hardware restart information.
97a	C31FLWRD	See Digital Autopilot RCS Routines.
97b,98	FAILREG	FAILREG+0 to FAILREG+2 (see General Program Control).
99a	FLGWRD10	Program control flag word.
99b	FLGWRD11	Program control flag word.
100a	GAMMAEI	See Display Computations. Same cell used for PREDANG (see Entry Computations), and GAMMAL.
100b	JJ	Loaded with $RTGO_{sp}$ (see Display Computations); JJ itself is in Entry Computations.

Rendezvous and Prethrust List

This list, with starting address of "CMRENDDL", is selected for DNLSTCOD = 2. It is used in P20, P21, P25, P29, P30, P31, P32, P33, P34, P35, P36, P37, P38, and P77 (and "P81" - "P88").

Word #	Quantity	Meaning
List Identification		
1a	77775_8	Identification of list (-2).
1b	77340_8	Special synchronization bits.
Snapshot Group #1		
		Words 2-8 are all sampled at the same telemetry interrupt time, with words 3-8 stored in DNTMBUFF for subsequent transmission.
2-4	R	Position state vector.
5-7	V	Velocity state vector.
8	T_{pptm}	State vector time.
Snapshot Group #2		
		Words 9-13 are all sampled at the same telemetry interrupt time, with words 10-13 stored in DNTMBUFF for subsequent transmission.
9,10a	CDU	IMU CDU angles.
10b	CDUT	See Optics Computations.
11-13	ADOT	See Digital Autopilot RCS Routines (same cell used for e.g. TVC DAP observed rates).
Erasable Group #1		
		Words 14-22 are sampled as the telemetry interrupt for them is received.
14,15a	AK	See Digital Autopilot Interface Routines.
15b	RCSFLAGS	See Digital Autopilot Interface Routines.
16,17a	THETADX	See Digital Autopilot RCS Routines.
17b	$DELCDUX_{sp}$	See Attitude Maneuvers.

Word #	Quantity	Meaning
18	T_{ig}	Ignition time (or predicted cutoff time).
19a	SLOPE	See Digital Autopilot RCS Routines.
19b	ADB	See Digital Autopilot RCS Routines.
20a	DAPDATR3	See Digital Autopilot Docked Jet Selection.
20b	CH5FAIL	See Digital Autopilot Docked Jet Selection.
21a	CH6FAIL	See Digital Autopilot Docked Jet Selection.
21b	DKRATE	See Digital Autopilot RCS Routines.
22a	DKDB	See Digital Autopilot RCS Routines.
22b	WHICHDAP	See Digital Autopilot RCS Routines.
Snapshot Group #3		
		Words 23-28 are all sampled at the same telemetry interrupt time, with words 24-28 stored in DNTMBUFF for subsequent transmission.
23	VHFTIME	See Measurement Incorporation.
24-27a	MARKDOWN	Cells MARKDOWN+0 to MARKDOWN+6 (see Measurement Incorporation).
27b	RM	See Measurement Incorporation.
28a	VHFCNT	See Measurement Incorporation.
28b	TRKMKCNT	See Measurement Incorporation.
Erasable Group #2		
		Words 29-44 are sampled as the telemetry interrupt for them is received.
29	T_{tpi}	See Rendezvous Computations.
30	SVEC+2	See Measurement Incorporation.
31	DELVTPF	See Burn Control.
32	T_{cdh}	See Rendezvous Computations (NSR time).
33	T_{csi}	See Rendezvous Computations (NCC time).

Word #	Quantity	Meaning
34	T_{pass4}	See Burn Control.
35-37	DELVLVC	See Burn Control.
38	RANGE	See Display Computations (same cell used for WWPOS, see Measurement Incorporation).
39	RRATE	See Display Computations (same cell used for HAPOX, Display Computations, and WWVEL, see Measurement Incorporation).
40-44	FLAGWRD0-FLAGWRD9	Program control flag words.
Display Table Group		
		Words 45-50 are sampled as the telemetry interrupt for them is received.
45-50	DSPTAB+0-DSPTAB+11	"Display table" information. See Data Input/Output.
Computer Clock		
		Word 51 is sampled when the telemetry interrupt for it is received.
51	T_{now}	Present value of computer clock (TIME2,TIME1).
Snapshot Group #4		
		Words 52-58 are all sampled at the same telemetry interrupt time, with words 53-58 stored in DNTMBUFF for subsequent transmission.
52-54	$\underline{R}_{other}$	See Orbital Integration.
55-57	$\underline{V}_{other}$	See Orbital Integration.
58	T_{etlm}	See Orbital Integration.
Snapshot Group #5		
		Words 59-63 are all sampled at the same telemetry interrupt time, with words 60-63 stored in DNTMBUFF for subsequent transmission.
59,60a	CDU	IMU CDU angles.
60b	CDUT	See Optics Computations.
61-63	ADOT	See Digital Autopilot RCS Routines.

Word #	Quantity	Meaning
Erasable Group #3		
		Words 64-90 are sampled as the telemetry interrupt for them is received.
64	OPTION1,2	See Display Interface Routines.
65	T_{et}	See Orbital Integration.
66,67a	THETADX	See Digital Autopilot RCS Routines.
67b	$DELCDUX_{sp}$	See Attitude Maneuvers.
68	$RSBBQ_{dp}$	See General Program Control.
69a	Channel 76	Not meaningful.
69b	Channel 77	Hardware restart information.
70a	C31FLWRD	See Digital Autopilot RCS Routines.
70b,71	FAILREG	FAILREG+0 to FAILREG+2 (see General Program Control).
72a	CDUS	See Optics Computations.
72b,73	PIPA	See IMU Computations.
74	NC1TIG	See Burn Control.
75	NC2TIG	See Burn Control.
76	DHDSP	See Rendezvous Computations.
77-79	DELVEET3	See Rendezvous Computations.
80a	OPTMODES	See Optics Computations.
80b	HOLDFLAG	See Digital Autopilot Interface Routines.
81a	LEMMASS	See Digital Autopilot Interface Routines.
81b	CSMMASS	See Digital Autopilot Interface Routines.
82a	DAPDATR1	See Digital Autopilot Interface Routines.
82b	DAPDATR2	See Digital Autopilot Interface Routines.
83,84a	ERROR	See Digital Autopilot RCS Routines.
84b	THETADX	See Digital Autopilot RCS Routines.
85-87	WBODY	See Digital Autopilot RCS Routines (same cells used for commanded rates in TVC).

Word #	Quantity	Meaning
88a	REDOCTR	See General Program Control.
88b,89	THETAD	Desired CDU angles (e.g. N22).
90a	IMODES30	See IMU Computations.
90b	IMODES33	See IMU Computations.
Channel Quantities		
		Words 91-94 are sampled as the telemetry interrupt for them is received.
91a	Channel 11	Outputs.
91b	Channel 12	Outputs.
92a	Channel 13	Outputs.
92b	Channel 14	Outputs.
93a	Channel 30	Inputs.
93b	Channel 31	Inputs.
94a	Channel 32	Inputs.
94b	Channel 33	Inputs.
Erasable Group #4		
		Words 95-100 are sampled as the telemetry interrupt for them is received.
95	RTHETA	See Display Computations (same cell used for WWOPT, see Measurement Incorporation, and RRATE2 and HPERX, Display Computations).
96	DVDSP1	See Rendezvous Computations.
97	DVDSP2	See Rendezvous Computations.
98	UTPIT	See Orbital and Rendezvous Navigation.
99	UTYAW	See Orbital and Rendezvous Navigation.
100a	FLGWRD10	Program control flag word.
100b	FLGWRD11	Program control flag word.

Powered List

This list, with starting address of "CMPOWEDL", is selected for DNLSTCOD = 3. It is used in P11, P40, P41, P47, P48, and P61.

Word #	Quantity	Meaning
List Identification		
1a	77774_8	Identification of list (-3).
1b	77340_8	Special synchronization bits.
Snapshot Group #1		
		Words 2-8 are all sampled at the same telemetry interrupt time, with words 3-8 stored in DNTMBUFF for subsequent transmission.
2-4	R	Position state vector.
5-7	V	Velocity state vector.
8	T_{pptm}	State vector time.
Snapshot Group #2		
		Words 9-13 are all sampled at the same telemetry interrupt time, with words 10-13 stored in DNTMBUFF for subsequent transmission.
9,10a	CDU	IMU CDU angles.
10b	CDUT	See Optics Computations.
11-13	ADOT	See Digital Autopilot RCS Routines. Same cells are used for OMEGAB (see Digital Autopilot TVC Routines).
Erasable Group #1		
		Words 14-29 are sampled as the telemetry interrupt for them is received.
14,15a	AK	See Digital Autopilot Interface Routines.
15b	RCSFLAGS	See Digital Autopilot Interface Routines.
16,17a	THETADX	See Digital Autopilot RCS Routines.
17b	$DELCDUX_{sp}$	See Attitude Maneuvers.

Word #	Quantity	Meaning
18	T_{ig}	Ignition time (or predicted cutoff time).
19a	SLOPE	See Digital Autopilot RCS Routines.
19b	ADB	See Digital Autopilot RCS Routines.
20a	DAPDATR3	See Digital Autopilot Docked Jet Selection.
20b	CH5FAIL	See Digital Autopilot Docked Jet Selection.
21a	CH6FAIL	See Digital Autopilot Docked Jet Selection.
21b	DKRATE	See Digital Autopilot RCS Routines.
22a	DKDB	See Digital Autopilot RCS Routines.
22b	WHICHDAP	See Digital Autopilot RCS Routines.
23	T_{go}	See Steering Computations. Should not be confused with T_{togo} (Burn Computations) of N40 etc.
24	T_{pptml}	See IMU Computations.
25-27	DELV	Sampled accelerometer output (if least significant half zero, is uncompensated; otherwise is compensated data).
28a	PACTOFF	See Digital Autopilot TVC Routines.
28b	YACTOFF	See Digital Autopilot TVC Routines.
29a	PCMD	See Digital Autopilot TVC Routines.
29b	YCMD	See Digital Autopilot TVC Routines.
Snapshot Group #3		
		Words 30-31 are sampled at the same telemetry interrupt time, with 31 stored in DNTMBUFF for subsequent transmission.
30a	CSTEER	See Steering Computations.
30b	RM	See Measurement Incorporation (also is least significant half of CSTEER).
31	MARKTIME	See Measurement Incorporation.

Word #	Quantity	Meaning
Erasable Group #2		
		Words 32-44 are sampled as the telemetry interrupt for them is received.
32	FIXTIME	See Orbital and Rendezvous Navigation.
33	DVTOTAL	See General Program Control.
34-36	$\underline{\text{REFSMMAT}}_0$	First row of [REFSMMAT].
37-39	$\underline{\text{REFSMMAT}}_3$	Second row of [REFSMMAT].
40-44	FLAGWRD0-FLAGWRD9	Program control flag words.
Display Table Group		
		Words 45-50 are sampled as the telemetry interrupt for them is received.
45-50	DSPTAB+0-DSPTAB+11	"Display table" information. See Data Input/Output.
Computer Clock		
		Word 51 is sampled when the telemetry interrupt for it is received.
51	T_{now}	Present value of computer clock (TIME2,TIME1).
Snapshot Group #4		
		Words 52-58 are all sampled at the same telemetry interrupt time, with words 53-58 stored in DNTMBUFF for subsequent transmission.
52-54	$\underline{R}_{other}$	See Orbital Integration.
55-57	$\underline{V}_{other}$	See Orbital Integration.
58	T_{etlm}	See Orbital Integration.
Snapshot Group #5		
		Words 59-63 are all sampled at the same telemetry interrupt time, with words 60-63 stored in DNTMBUFF for subsequent transmission.
59,60a	$\underline{\text{CDU}}$	IMU CDU angles.
60b	CDUT	See Optics Computations.

Word #	Quantity	Meaning
61-63	ADOT	See Digital Autopilot RCS Routines. Same cells are used for OMEGAB (see Digital Autopilot TVC Routines).
Erasable Group #3		
		Words 64-90 are sampled as the telemetry interrupt for them is received.
64,65a	AK	See Digital Autopilot Interface Routines.
65b	RCSFLAGS	See Digital Autopilot Interface Routines.
66,67a	THETADX	See Digital Autopilot RCS Routines.
67b	$DELCDUX_{sp}$	See Attitude Maneuvers.
68	$RSBBQ_{dp}$	See General Program Control.
69a	Channel 76	Not meaningful.
69b	Channel 77	Hardware restart information.
70a	C31FLWRD	See Digital Autopilot RCS Routines.
70b,71	FAILREG	FAILREG+0 to FAILREG+2 (see General Program Control).
72a	CDUS	See Optics Computations.
72b,73	PIPA	See IMU Computations.
74	ELEV	See Rendezvous Computations.
75	SVEC+2	See Measurement Incorporation.
76	T_{et}	See Orbital Integration.
77a	FLGWRD10	Program control flag word.
77b	FLGWRD11	Program control flag word.
78	T_{evt}	Event (e.g. liftoff or engine on/off) time.
79a	PCMD	See Digital Autopilot TVC Routines.
79b	YCMD	See Digital Autopilot TVC Routines.
80a	OPTMODES	See Optics Computations.
80b	HOLDFLAG	See Digital Autopilot Interface Routines.

Word #	Quantity	Meaning
81a	LEMMASS	See Digital Autopilot Interface Routines.
81b	CSMMASS	See Digital Autopilot Interface Routines.
82a	DAPDATR1	See Digital Autopilot Interface Routines.
82b	DAPDATR2	See Digital Autopilot Interface Routines.
83,84a	ERROR	See Digital Autopilot RCS Routines.
84b	THETADX	See Digital Autopilot RCS Routines.
85-87	WBODY	See Digital Autopilot RCS Routines. Same cells used for OMEGAC (see Digital Autopilot TVC Routines).
88a	REDOCTR	See General Program Control.
88b,89	THETAD	Desired CDU angles (e.g. N22).
90a	IMODES30	See IMU Computations.
90b	IMODES33	See IMU Computations.
Channel Quantities		
		Words 91-94 are sampled as the telemetry interrupt for them is received.
91a	Channel 11	Outputs.
91b	Channel 12	Outputs.
92a	Channel 13	Outputs.
92b	Channel 14	Outputs.
93a	Channel 30	Inputs.
93b	Channel 31	Inputs.
94a	Channel 32	Inputs.
94b	Channel 33	Inputs.
Erasable Group #4		
		Words 95-100 are sampled as the telemetry interrupt for them is received.
95-97	$\underline{V}_{gtig}$	See Burn Control.
98-100	DELVLVC	See Burn Control.

Special Erasable Memory Dump List

Input of a V74E causes DNTMGOTO to be switched so as to start at the next telemetry interrupt the transmission of a "dump" of the erasable memory (all cells are sent sequentially for two times, i.e. two complete passes through the erasable memory). The erasable memory hardware is divided into eight "banks" of 256 cells each, and a similar division is made for downlinking of information: the first two words are used for control data, and the next 128 contain the 256 words in the particular bank set identified by the first two words. The format of the information is:

Word #	Quantity	Meaning
List Identification		
1a	01777_g	Identification of list.
1b	77340_g	Special synchronization bits.
2a	DUMPLOC	See Telemetry (page TELE-7).
2b	TIME1	Least significant half of computer clock.
Erasable Memory Bank		
3-130	Cells in bank (addresses sent in consecutive increasing order).	

Testing Routines

SELFCHK Address set as initial condition for SELFRET in "STARTSB2"

Perform "SMODECHK" (will not return unless self-check is desired)

Proceed to "ERASCHK"

SMODECHK

SKEEP1 = Return address

Perform "CHECKNJ" (returns immediately if no new job waiting, and otherwise returns after doing job(s))

If SMODE = +0:

Proceed to second line of "SMODECHK"

If |SMODE| $\leq$ 8:

SCOUNT = SCOUNT + 1

If SMODE = -0, proceed to address specified by SKEEP1

If |SMODE| = 1, 2, 3, 6, 7, or 8: (10_8 = decimal 8)

Proceed to address specified by SKEEP1

If |SMODE| = 4, proceed to "ERASCHK"

If |SMODE| = 5, proceed to "ROPECHK"

SMODE = 0 (magnitude exceeded 8)

Proceed to "SELFCHK"

PRERRORS

If ERESTORE = 0, proceed to "ERRORS"

$E_{SKEEP7} = SKEEP5_{dp}$

ERESTORE = 0

Proceed to "ERRORS"

ERRORS

Inhibit interrupts (released upon return from alarm routine)

SFAIL = Return address (to routine calling "PRERRORS" or "ERRORS")

ALMCADR = SFAIL

ERCOUNT = ERCOUNT + 1

Perform "ALARM2" (pattern 1102_8)

If SMODE = -0:

Proceed to address specified by SFAIL

If SMODE > 0:

SMODE = +0

Proceed to "SELFCHK"

ERASCHK

SKEEP2 = 1 (flag to check cells 0061_8 - 1373_8)

EBANK = 0

SKEEP7 = 1461_8 (first cell in bank 0 to be checked, since cells 0 - 60_8 are "special" erasable cells)

SKEEP3 = 1777_8 (last cell)

Proceed to "ERASLOOP"

ERASLOOP

Inhibit interrupts

SKEEP4 = EBANK (non-zero EBANK loads bits 11-9)

$SKEEP5_{dp} = E_{SKEEP7_{dp}}$ (address also determined by EBANK)

ERESTORE = SKEEP7

E_{SKEEP7} = SKEEP7 (loads with own S-register address)

$E_{SKEEP7+1}$ = SKEEP7 + 1

If $E_{SKEEP7} - E_{SKEEP7+1} \neq -1$, perform "PRERRORS"

If ERESTORE $\neq$ 0:

$E_{SKEEP7_{dp}} = - E_{SKEEP7_{dp}}$

If $E_{SKEEP7+1} - E_{SKEEP7} \neq -1$, perform "PRERRORS"

If ERESTORE $\neq$ 0:

$E_{SKEEP7_{dp}} = SKEEP5_{dp}$

ERESTORE = 0

Release interrupts

Perform "CHECKNJ"

EBANK = bits 11-9 of SKEEP4

SKEEP7 = SKEEP7 + 1

If SKEEP3 - SKEEP7 $\neq$ 0:

Proceed to "ERASLOOP"

If SKEEP2 $>$ 0:

SKEEP2 = SKEEP2 - 1 (sets 0)

SKEEP7 = 0061_8 (unswitched erasable form of bank 0)

SKEEP3 = 1373_8 (SKEEP4 - SKEEP7 not checked, cells 1374 - 7)

Proceed to "ERASLOOP"

SKEEP2 = 1

EBANK = EBANK + 1, modulo 8 (7 + 1 = 0)

If EBANK = 2:

SKEEP7 = 1400_8

SKEEP3 = 1773_8 (cells 1374-7 not checked)

Proceed to "ERASLOOP"

If EBANK $\neq$ 0:

SKEEP7 = 1400_8

SKEEP3 = 1777_8

Proceed to "ERASLOOP"

EBANK = 3

Read out cells from 0060_8 to 0010_8 to check on their parity (addresses 0007_8 - 0000_8 have no parity bits), by a CS (Clear Subtract) order

Check cycle and shifting registers; if difficulty, perform "PRERRORS"

SCOUNT+1 = SCOUNT+1 + 1

Perform "SMODECHK"

Proceed to "ROPECHK" (if return from "SMODECHK")

ROPECHK

SKEEP6 = -0 (indicates "ROPECHK" option)

SKEEP4 = 0 (bank number)

SKEEP7 = 1 (counter for reading fixed-fixed banks)

SKEEP1 = 0 (sum)

SKEEP3 = 0 (address read)

SKEEP5 = 1 (counts two TC self words for end of bank data)

Proceed to "COMADRS"

SHOWSUM+2 Entered from "GOSHOSUM" for V91E

SKEEP6 = 1 (indicates "SHOWSUM" option)

SMODE = +0

SELFRET = "SELFCHK"

Proceed to second line of "ROPECHK"

COMADRS

$SKEEP2 = E_{SKEEP3,SKEEP4}$ (address determined by 2000_8 + SKEEP3 for S-register; bits 15-11 of SKEEP4 for FBANK; bits 7-5 of SKEEP4 for FEXT)

SKEEP1 = SKEEP1 + SKEEP2

If |SKEEP1| $\geq$ 16384:

SKEEP1 = SKEEP1 - 16383 sgn SKEEP1

TS = (2000_8 + SKEEP3) - SKEEP2 (zero if cell contains TC self order, since TC is op code "0")

Proceed to "ADRSCHK"

ADRSCHK

If bits 10-1 of SKEEP3 = 1777_8, proceed to "SOPTION" (just read last bank cell)

If SKEEP5 < 0, proceed to "SOPTION" (just read cell following two TC self orders)

If TS $\neq$ 0, SKEEP5 = 1

If TS = 0, SKEEP5 = SKEEP5 - 1 (goes +1, +0, -1)

If SKEEP6 = -0:

Perform "CHECKNJ"

If SKEEP6 $>$ 0:

Check NEWJOB cell for job of higher priority than present job (which has priority 30_8 from "KEYCOM"), and do that job if it exists, proceeding when present job again of highest priority.

SKEEP3 = SKEEP3 + 1

If SKEEP7 $\geq$ +0, proceed to "COMADRS"

Proceed to "FXADRS"

FXADRS

SKEEP2 = E_{SKEEP3} (S-register contents determined by SKEEP3 only for fixed-fixed memory cell)

SKEEP1 = SKEEP1 + SKEEP2

If |SKEEP1| $\geq$ 16384:

SKEEP1 = SKEEP1 - 16383 sgn SKEEP1

TS = SKEEP3 - SKEEP2

Proceed to "ADRSCHK"

SOPTION

TS = bits 15-11 of SKEEP4, cycled left 5 places (puts FBANK information in bits 5-1)

If bits 8-1 of SKEEP4 $\neq$ 0: (reading super-bank cells, i.e. $\geq 30_8$)

TS = bits 3-1 of TS + (bits 8-2 of SKEEP4 shifted right 1 place)

If SKEEP6 $>$ 0:

SKEEP3 = SKEEP2

SKEEP2 = TS

MPAC+2 = "SKEEP1"

(If SKEEP6 > 0):

TS = 0501_{vn}

Proceed to "GOXDSPF": if terminate, proceed
if proceed, proceed to "NXTBNK"
otherwise, proceed to second previous line (reload MPAC+2)

SKEEP1 = "SELFCHK"

Proceed to "ENDEXT"

SKEEP1 = |SKEEP1|

If SKEEP1 - TS - 1 ≠ -1, perform "PRERRORS"

Proceed to "NXTBNK"

NXTBNK

If SKEEP4 = K_{stbnk}:

If SKEEP6 > 0, proceed to second line of "ROPECHK"

Proceed to "SELFCHK"

Increment bits 14-11 of SKEEP4 by 1 (FBANK scaling)

If SKEEP4 overflows (i.e. previous value of these bits 17_8):

SKEEP4 = 20_8 (in bits 15-11)

If SKEEP4 bits 15-11 = 0 (i.e. previous value 37_8):

SKEEP4 = SKEEP4 + 60020_8 (sets FBANK to 30_8 in bits 15-11, and increments FEXT in bits 7-5)

If SKEEP4 = 60000_8: (bits 15-11 = 30_8)

SKEEP4 = 60060_8 (sets FEXT bits, 7-5, to 3)

If SKEEP7 > 0:

SKEEP7 = SKEEP7 - 1, limited ≥ +0

Proceed to fourth line of "ROPECHK"

If SKEEP7 = +0:

TS = 1

Proceed to "FXFX"

If SKEEP7 = -1:

TS = +0

Proceed to "FXFX"

If SKEEP7 = -0:

SKEEP7 = 64 (larger than number of remaining banks)

Proceed to fourth line of "ROPECHK"

FXFX

SKEEP7 = - TS

If TS ≠ 0:

SKEEP3 = 04000_8 (first cell in "bank 2")

If TS = 0:

SKEEP3 = 06000_8 (first cell in "bank 3")

SKEEP1 = 0

SKEEP5 = 1

Proceed to "FXADRS"

IMU Performance Testing

The computations associated with the IMU performance testing (P07) have been partially removed from fixed memory. For continuity purposes, the following information on the computations formerly done at the start of P07 (based on the H2 flight program) is provided.

1. Program display set to 07.
2. $DRIFTT_{sp}$ set 0.
3. GEOCOMP1 set 0 (logic checking for this cell = 0 not shown).
4. LENGTHOT set to 898 seconds.
5. 1SECXT1 and 1dPIPADT set to one second interval.
6. WANGI set to - cos C_{atd}; WANGO set to sin C_{atd}.
7. IMU coarse aligned to 0^o.
8. Computations progressed to "GEOIMUTT".

GEOIMUTT

Perform "IMUZERO"

Perform "IMUSTALL": if error return, proceed to "SOMERR2"
otherwise, proceed

NDXCTR = 0

WANGT = 0

$\underline{X}_{dc} = (1, 0, 0)$

$\underline{Y}_{dc} = (0, \sin C_{azmth}, \cos C_{azmth})$

$\underline{Z}_{dc} = (0, -\cos C_{azmth}, \sin C_{azmth})$

Perform "CALCGA" (here if MODREG $\neq 3$, since should be 7)

Perform "IMUCOARS"

If bit 14(GLOKFAIL) of FLAGWRD3 = 1:

NDXCTR = NDXCTR + 1

Set bit 14(GLOKFAIL) of FLAGWRD3 = 0

Perform "IMUSTALL": if error return, proceed to "SOMERR2"
otherwise, proceed

If NDXCTR > 0:

Proceed to "PIPACHK"

Perform "IMUFINE"

Perform "IMUSTALL": if error return, proceed to "SOMERR2"
otherwise, proceed

Call "GOESTIMS$_d$" in PERFDLAY$_{dp}$ centi-seconds

Put present job to sleep (starting address id = "ESTIMS$_d$")

GOESTIMS$_d$

Awaken job with starting address id = "ESTIMS$_d$"

End of task (since WANGT = 0, "ONCEMORE" will go to "TORQUE")

ESTIMS$_d$

Inhibit interrupts

GTSWTLT1 = TIME1

PIPA = 0

Release interrupts

Zero erasable memory cells 5,1453 - 5,1570: includes least significant half of DRIFTT, KGAINAZ, KGAINVRD, KGAINNSD, ANGX, ANGY, ANGZ, INTY, INTZ, DRIFTO, DRIFTI, VLAUN$_y$, VLAUN$_z$, ACCWD$_y$, ACCWD$_z$, POSNV$_y$, POSNV$_z$, and ALTIM.

GCOMPSW = 0

ALX1S = 144

CMPX1 = -1

KGAINPIP = K$_{soupy0}$

KGAINERC = K$_{soupy2}$

GCOMP = 0

DELV = 0

Proceed to "SLEEPIE$_d$"

SLEEPIE$_d$

If WANGT > 0:

Perform "EARTHR*"

Perform "CHKCOMED"

Inhibit interrupts

TS = GTSWTLT1 - TIME1

If TS > 0:

TS = TS - 163.83 seconds (should be 163.84)

TS = TS + 1SECXT1

If TS ≤ 0:

TS = 0.04 seconds

Call "ALLOOP$_d$" in TS seconds

Release interrupts

End of job

ALLOOP$_d$

GTSWTLT1 = TIME1

TS = ALTIM

Set restart grp. 5 to cause restart at next line

If TS = +0:

ALTIMS = +0

ALTIM = -0

If TS = -0:

ALTIM = +0

If TS $<$ 0: (should not be positive non-zero)

ALTIM = - (|ALTIM| - 1) (if was -1, result is -0)

Set DELV$_{sp}$ = PIPA and PIPA = +0 (no special restart provisions)

Set restart group 5 to cause restart at next line

Establish "ALFLT$_d$" (priority 22_8)

End of task

ALFLT$_d$

TS = DELV $\left[X_{sm}\right]$

DPIPAY = - TS$_y$

DPIPAZ = TS$_z$

If ALTIMS $\geq$ 0:

TS = 144 - ALX1S

ALTIM = ALFDK$_{TS}$ (for T seconds, set to -(T - 2))

ALTIMS = ALFDK$_{TS+1}$ (e.g. set to -1)

BUFTCPIP = ALFDK$_{TS+2}$ ("a_1")

BUFTCERC = ALFDK$_{TS+4}$ ("a_2")

(If ALTIMS $\geq$ 0):

BUFSLPAZ = $ALFDK_{TS+6}$ ("a_3")

BUFSLPVRD = $ALFDK_{TS+8}$ ("a_4")

BUFSLPNSD = $ALFDK_{TS+10}$ ("a_5")

ALX1S = ALX1S - 12

INTY = INTY - K_{pipasc} DPIPAY (i.e. "po_s", south PIPA)

$DELM_y$ = K_{vesc} $VLAUN_y$ - INTY ("ΔM_1")

INTZ = INTZ - K_{pipasc} DPIPAZ ("po_e", east PIPA)

$DELM_z$ = K_{vesc} $VLAUN_z$ - INTZ ("ΔM_2")

KGAINPIP = BUFTCPIP KGAINPIP ("K_1")

KGAINERC = BUFTCERC KGAINERC ("K_2")

INTY = INTY + KGAINPIP $DELM_y$ ("po_s") (tag "AIKLP")

KGAINAZ = KGAINAZ + BUFSLPAZ ("K_3")

ANGX = ANGX + 4 KGAINAZ $DELM_y$ ("alpha")

$VLAUN_y$ = $VLAUN_y$ + K_{ask0} $DELM_y$ ("v_{ls}")

ANGZ = ANGZ + KGAINERC $DELM_y$ ("gamma") (tag "AIKLP")

KGAINVRD = KGAINVRD + BUFSLPVRD ("K_4")

DRIFTO = DRIFTO + 4 KGAINVRD $DELM_y$ ("d_x")

$ACCWD_y$ = $ACCWD_y$ + K_{ask2} $DELM_y$ ("a_s")

INTZ = INTZ + KGAINPIP $DELM_z$ ("po_e") (tag "AIKLP")

KGAINNSD = KGAINNSD + BUFSLPNSD ("K_5")

DRIFTI = DRIFTI + 4 KGAINNSD $DELM_z$ ("d_y")

$VLAUN_z$ = $VLAUN_z$ + K_{ask0} $DELM_z$ ("v_{le}")

ANGY = ANGY + KGAINERC $DELM_z$ ("beta") (tag "AIKLP")

$ACCWD_z$ = $ACCWD_z$ + K_{ask2} $DELM_z$ ("a_e") (for indexing, a dummy "K_6", value 0, is generated)

$\underline{TS}$ = [TRANSM1] ($POSNV_y$, $VLAUN_y$, $ACCWD_y$) (tag "LOOSE")

($POSNV_y$, $VLAUN_y$, $ACCWD_y$) = $\underline{TS}$

$\underline{TS}$ = [TRANSM1] ($POSNV_z$, $VLAUN_z$, $ACCWD_z$)

$(POSNV_z,\ VLAUN_z,\ ACCWD_z) = TS$

$SNANGi = \sin K_{georgj}\ ANGi \quad (i = X,Y,Z)$

$CSANGi = \cos K_{georgj}\ ANGi \quad (i = X,Y,Z)$

Proceed to erasable memory cell 3400_8 (E7,1400)

NOTE: Following coding was in earlier programs (Sundisk) and is supplied for continuity purposes only. Fixed memory information is at "ONCEMORE" and onward.

$INTY = INTY + SNANGZ$ ("po_s")

$INTZ = INTZ + SNANGY\ CSANGZ$ ("po_e")

$WPLATO = DRIFTO - WANGO\ (CSANGY\ CSANGZ) - WANGI\ (SNANGX\ SNANGY + CSANGX\ CSANGY\ SNANGZ)$

$WPLATI = WANGT\ WANGI + WANGO\ SNANGZ + DRIFTI - WANGI\ CSANGX\ CSANGZ$

$WPLATT = DRIFTT - WANGO\ SNANGY\ CSANGZ + WANGI\ (SNANGX\ CSANGY - CSANGX\ SNANGY\ SNANGZ)$

WPLATi (i = O,I,T) are x,y,z components of $\underline{W}_{sm}$

$TS = (WPLATO\ CSANGY + WPLATT\ SNANGY)/\ CSANGZ$

$ANGX = ANGX + K_{georgk}\ TS$

$ANGY = ANGY + K_{georgk}\ (WPLATI + TS\ SNANGZ)$

$ANGZ = ANGZ + K_{georgk}\ (WPLATT\ CSANGY - WPLATO\ SNANGY)$

If overflow has taken place since start of job:

Proceed to "SOMEERRR"

NOTE: This ends coding taken from earlier programs (Sundisk); presumably the erasable coding now transfers to "ONCEMORE"

ONCEMORE

If LENGTHOT > 0:

LENGTHOT = LENGTHOT - 1

Proceed to "$SLEEPIE_d$"

If WANGT > 0:

$LOSVC2 = CDU_x$

$OG\underline{C} = \left[X_{sm}\right]\ (-K_{georgj})\ (ANGX,\ ANGY,\ ANGZ)$

TS = "OGC"

Perform "IMUPULSE"

Perform "IMUSTALL": if error return, proceed to "SOMERR2"
otherwise, proceed

If WANGT > 0:

Proceed to "VALMIS"

ERVECTOR = K_{omegms}(sin C_{atd}, - cos C_{atd}, 0)

$T_{mark} = T_{now}$

ERCOMP = 0

Proceed to "TORQUE"

SOMEERRR

Perform "ALARM" (pattern 1600_8)

Proceed to second line of "SOMERR2"

TORQUE

DSPTEM2 = 0 (forces R1 of N98 = 0)

DSPTEM2+1 = $DRIFTI_{sp}$

TS = POSITON - 1

$SOUTHDR_{TS} = DRIFTI_{sp}$

Perform "SHOW"

Proceed to "PIPACHK"

PIPACHK

Proceed to erasable memory cell 2000_8 (E4,1400)

VALMIS

DSPTEM2+1 = $DRIFTO_{sp}$

DSPTEM2+0 = 0 (forces R1 of N98 = 0)

Perform "SHOW"

Proceed to second line of "SOMERR2"

SHOW

DSPTEM2+2 = POSITON

TS 0698$_{vn}$

Proceed to "GOFLASH": if terminate, proceed to 2nd line of "SOMERR2"
if proceed, proceed
otherwise, proceed to "SHOW"

Return

See also list of major variables and list of routines

ldPIPADT: See IMU Computations.

lSECXTl: See Prelaunch Alignment (set to 1 second for gyro drift determination).

ACCWD$_y$, ACCWD$_z$: Value of "horizontal acceleration of launch vehicle" (due to sway) in north-south and east-west directions respectively, scale factor B9, units cm/sec^2.

ALFDK$_i$: Table of erasable memory quantities used in "ALFLT$_d$" to update values of parameters to be used for filtering in gyro drift computations. The table consists of five double precision constants, one single precision constant (the setting for ALTIM), and a reset value of ALTIMS (which could be e.g. -1 for all tables). Values must be initialized by an erasable memory load(with the first value at "ALFDK", octal cell 2022_8), with settings for ALTIM, ALTIMS, BUFTCPIP, BUFTCERC, BUFSLPAZ, BUFSLPVRD, and BUFSLPNSD stored in that order (first two single precision, remainder double precision). Scale factor of first two assumed B14, and the remainder assumed BO, in this writeup. See below for "typical" values (obtained from Sundisk program, when information was in fixed memory).

ALMCADR: See General Program Control.

ALTIM: Single precision value of time remaining prior to change in filter constants for drift measurements, scale factor B14, units seconds. To cause a set of gains to be used for T seconds, ALTIM is set to - (T - 2).

ALTIMS: Single precision flag cell set to 0 when a gain change should be made (see ALTIM), and then reset (e.g. to -1) when the gain change has been done, scale factor B14.

ALX1S: Single precision cell, scale factor B14, used to control selection of values from ALFDK$_i$ erasable memory table (set to 144 in "ESTIMS$_d$").

ANGX, ANGY, ANGZ: Values of determined angle changes about vertical, south, and east axes respectively, scale factor BO, units revolutions: they are "azimuth alignment angle","south axis leveling angle", and "east axis leveling angle" respectively (alpha, beta, gamma in official equation documentation).

BUFSLPAZ, BUFSLPNSD, BUFSLPVRD: Set of buffer cells used to contain the values of the slopes of the gains for azimuth angle, north-south drift, and vertical drift respectively, scale factor BO, read from ALFDK table set. Cells are ALDK+4, ALDK+8, and ALDK+6 respectively.

BUFTCERC, BUFTCPIP: Set of buffer cells used to contain the values of the "time constants" for the erection angles ("east axis leveling angle" and PIPA outputs, scale factor BO, as read from ALFDK table set. Cells are ALDK+2 and ALDK.

C_{atd}: See Prelaunch Alignment.

C_{azmth}: See Prelaunch Alignment.

CMPX1: Single precision cell, scale factor B14, used to set proper contents of index register X1 to permit use of an index loop (X1 is set successively to $\pm$ 1) to perform calculations in "$ALFLT_d$": use of the cell is not shown in this writeup.

CSANGi (i = X,Y,Z): Values of cosine of ANGX, ANGY, and ANGZ, scale factor B1, stored in push-down list locations 16D, 18D, and 20D respectively.

$DELM_y$, $DELM_z$: Value of measurement quantity in south and easterly directions used in drift test, scale factor B-2, units radians.

DPIPAY, DPIPAZ: See Prelaunch Alignment.

DRIFTI: Value of gyro drift measurement output displayed in "TORQUE", scale factor (assumed) B0, units "radians", giving the "south gyro drift".

DRIFTO: Value of gyro drift measurement output displayed in "VALMIS", scale factor (assumed) B0, units "radians", giving the "vertical gyro drift".

DRIFTT: Input "drift" to gyro drift determination routine (to separate "east gyro drift" from "azimuth error"), scale factor B0, units radians. It has only its most significant half loaded by calling routines, with the least significant half set 0.

EBANK: See Data Input/Output.

ERCOMP: See Prelaunch Alignment.

ERCOUNT: Single precision cell, scale factor B14, used to count the number of errors encountered in the self-check routine. The cell is initialized to 0 as part of a verb 36 fresh start, and may be read as the third component of noun 08.

ERESTORE: Single precision cell, initialized to 0 as part of a fresh start, used for control purposes in functions associated with the erasable memory checking ("ERASLOOP") portion of the self-check program. It is set to the S-register portion of the lower of the two erasable memory cells being checked (the same as SKEEP7) before these cells are altered. If a restart is encountered, the "GOPROG" routine checks that bits 15-11 of the word are zero (i.e. it is less than 2000_8, as required for erasable memory cells), and that it is equal to SKEEP7 (assuming, of course, the word in non-zero). Both conditions must be satisfied before SKEEP4 (for EBANK) and SKEEP7 address information are used to restore the cells; if one is not, then a fresh start is performed. Since

the ERESTORE cell (address 1360_8) is subject to erasable memory cell checks, when it does not necessarily contain the same data as SKEEP7, a fresh start may be forced even though SKEEP4 and SKEEP7 (neither of these cells are used for change purposes in the erasable memory check routine, since upper limit is 1373_8) is sufficient to permit restoration of erasable memory contents.

ERVECTOR: See Prelaunch Alignment.

FBANK: Hardware register cell (address 0004_8) used to contain the fixed memory bank number (if in range 30_8 - 37_8, FEXT is also used). See 3420.5-27 for details (only bits 15-11 are used).

FEXT: Computer hardware channel (channel 07, also referred to as SUPERBNK) used to select the appropriate fixed-memory bank for FBANK settings of 30_8 or more. Only bits 7-5 are used, with a setting of 3 selecting banks 3i; a setting of 4 selecting banks 4i (bank 43 is the last one in the computer). See 3420.5-27 for details.

GCOMP: See IMU Computations.

GCOMPSW: See IMU Computations.

GEOCOMP1: See Prelaunch Alignment. Set to +0 to indicate gyro drift measurement, but this logic not shown in this writeup. Instead, the zero-value case is in this writeup and the non-zero case in Prelaunch Alignment.

GTSWTLT1: See Prelaunch Alignment.

INTY, INTZ: Value of filtered accelerometer output (corrected for vehicle sway etc.) used in gyro drift test, scale factor B-2, units radians. Could also be considered to be "south" and "east" velocity increments expressed in units of g's (see K_{pipasc}).

K_{askO}: Constant, program notation "ALSK", scale factor B12, value 0.17329931. Value corresponds to 0.72402338 x 980.402 x 2^{-12}, where first term is "wind-induced sway velocity gain" (official equation documentation notation "K_7"), second converts DELM to units of cm/sec (i.e. units of VLAUN, cf. K_{pipasc}), and third term is scale factor.

K_{ask2}: Constant, program notation "ALSK +2", scale factor B12, value -0.00835370. Value corresponds to 0.03490074 x (-1) x 980.402 x 2^{-12}, where first term is "wind-induced sway accelerometer gain" (official equation documentation notation "K_8"), second is an equation factor, third converts to units of cm/sec^2 (cf. K_{pipasc}), and fourth term is scale factor.

K_{georgj}: Constant, program notation "GEORGEJ", scale factor B-2, value 0.63661977. Value corresponds to $(1/2\pi) \times 2^2$, to convert between radians and revolutions (the interpretive language trig functions require angle measurements in revolutions).

K_{georgk}: Constant formerly incorporated in fixed memory for use in IMU calibration computations. Value was 0.59737013, scale factor B-13. This corresponded to about $7.2921158E\text{-}5 \times 2^{13}$, where first term is earth rate in rad/sec (period of about 86164.0912 seconds).

K_{omegms}: See Prelaunch Alignment.

K_{pipasc}: Constant, program notation "PIPASC", scale factor B-7, value 0.76376833. Value corresponds to $5.85 \times (1/980.402) \times 2^7$, where first term is nominal accelerometer scale factor (cm/sec per count), second is normalization factor (acceleration due to gravity), and third is scale factor. For convenience in description, a fourth factor of "1/second" has been assumed reflected in this constant, giving for units of result (in INTY etc.) "radians".

K_{soupy0}: Constant, program notation "SOUPLY", scale factor B0, used in "$ESTIMS_d$" to initialize KGAINPIP. Value is 0.93505870.

K_{soupy2}: Constant, program notation "SOUPLY +2", scale factor B2, used in "$ESTIMS_d$" to initialize KGAINERC. Value is 0.26266423, corresponding to a "true value" of about 1.05065692.

K_{stbnk}: Single precision constant, program notation "LSTBNKCH", octal value 66100_8, corresponding to an FBANK value (bits 15-11) of 33_8 and an FEXT value (bits 7-5) of 4, i.e. a final "bank" readout of bank 43_8, the final computer hardware fixed memory bank.

K_{vesc}: Constant, program notation "VELSC", scale factor B-9, value -0.52223476. Value corresponds to $(-1) \times (1/980.402) \times 2^9$, where first term is an equation factor, second converts for acceleration due to gravity (cf. K_{pipasc}), and third is scale factor.

KGAINAZ, KGAINERC, KGAINNSD, KGAINPIP, KGAINVRD: Values of gains updated each cycle in gyro drift determination computations. KGAINERC and KGAINPIP are initialized to non-zero values in "$ESTIMS_d$" and multiplied by "time constants" for "erection angles" and "PIPA outputs" respectively, with scale factors due to initialization of B2 (KGAINERC) and B0. The others (KGAINAZ, KGAINNSD, and KGAINVRD) are initialized to 0 values in "$ESTIMS_d$", and are incremented each cycle to achieve varying gains for "azimuth angle", "north-south drift", and "vertical drift" respectively: all are considered to have scale factor B0 (see ALFDK). Program notation for the quantities is AIK+4, AIK+2, AIK+8, AIK+0, and AIK+6 respectively.

LENGTHOT: Single precision cell, scale factor B14, used to contain time duration information. Initialization must be done as part of erasable memory pre-load (in Prelaunch Alignment is initialized by coding).

LOSVC2: Single precision cell, program notation "LOSVEC +1", scale factor B-1, units revolutions, loaded with CDU_x in "ONCEMORE".

NDXCTR: Single precision cell, scale factor B14, initialized to 0 in "GEOIMUTT" and incremented to 1 for a "gimbal lock" return from "CALCGA" (angle of 60° or more).

NEWJOB: See General Program Control.

OGC: See Coordinate Transformations (used also as communication cell).

PERFDLAY: See Prelaunch Alignment (for gyro drift test, must be set manually to some value).

PIPA: See IMU Computations.

POSITON: Single precision cell, scale factor B14, used for indexing and display purposes (used in previous programs to select desired stable member orientation from fixed memory information). Must be loaded manually (inputs in "SHOW" do not change it).

$POSNV_y$, $POSNV_z$: Values of "horizontal displacement of launch vehicle" in south and east directions respectively, assumed scale factor B9, units cm ("assumed" since scaling of [TRANSM1] elements not known, but treated as B1).

SCOUNT, SCOUNT+1: Pair of single precision counters, scale factor B14, used to count the number of executions of portions of the self-check program. SCOUNT is incremented each time "SMODECHK" is entered with SMODE $\neq +0$ and with magnitude below 9; SCOUNT+1 is incremented each time "ERASLOOP" is completed (at the end of the test). Both quantities are modulo 2^{14}, and would have to be initialized manually since they are not preset as part of a fresh start (V36E), if a "true count" were desired. A value of SCOUNT = 3 (if set 0 before SMODE made e.g. 1) means that the self-test erasable and fixed memory checks have been completed. Addresses are 1366_8 and 1367_8 respectively.

SELFRET: See General Program Control.

SFAIL: Single precision cell used in "ERRORS" routine to retain return address information (and hence data on the cause of the self-check difficulty). Contains the same information as ALMCADR if no subsequent alarms (from sources other than self-check) are generated; it is not preset 0 by program unless error reset input.

SKEEP1: Self-check register #1, used to retain return address information from "SMODECHK" and the value of the bank sum while being formed. If the "SHOWSUM" option (V91E) is used, it contains the value of the sum as displayed in R1 (can be either + or -, and if + should be equal to the bank number; if minus should be the complement of the bank number). If self-check memory verification is done, is replaced by the magnitude of the sum before checking against the bank number.

SKEEP2: Self-check register #2, used in erasable memory check routine as a flag (if non-zero) to cause unswitched erasable to be checked (cells 0061_8 - 1372_8, plus 1373_8 partially) after completion of checks for each erasable bank (these cells are in banks 0-2, which should be read for any value of EBANK). Cell is used in fixed memory check routine to contain the contents of the fixed memory cell just read; in the "SHOWSUM" option, it contains the fixed memory bank number (00 - 43_8) displayed in R2, which should be the same as the magnitude of SKEEP1 (value of SKEEP1 is displayed in R1).

SKEEP3: Self-check register #3, used in erasable memory check routine to contain the value of the last address to be checked (i.e. one more than the final value of SKEEP7 actually used). The routine checks cells in pairs in ascending order (starting with the cell initialized in SKEEP7), and SKEEP3 specifies the final cell forming the upper half of a pair (SKEEP7 +1). Cell is used in fixed memory check to contain for fixed-fixed memory the S-register information (4000_8 - 7777_8), and for fixed-switchable memory the S-register information decreased by 2000_8. For the "SHOWSUM" option, cell is loaded with the final value of SKEEP2, which gives the final word read from memory (the "bugger word", designed to make the sum of all words in the memory, including itself, equal in magnitude to the bank number) for the bank, and is displayed in R3.

SKEEP4: Self-check register #4, used in erasable memory check routine to contain the erasable memory bank of the cells being checked (employed in "GOPROG" to reset EBANK, cf. ERESTORE). In fixed memory check routine, bits 15-11 are used to contain the appropriate setting for FBANK and bits 7-5 the appropriate value for FEXT.

SKEEP5: Self-check register #5, used with SKEEP6 in erasable memory check routine to retain the previous value of the cell being checked (to permit restoration of the cell in "GOPROG" or at the end of the test segment checking the cell). Used in fixed memory check routine to monitor for the presence of two TC self ("transfer control to the present step") orders, indicating that the following cell should be the final one entering the sum (bank sum also halted after last cell in bank has been read). If step not a TC self, SKEEP5 set to +1; if it is, cell is decreased by 1, and after reaching -1 the routine is halted for that bank after the next ("bugger", see SKEEP3) word is incorporated into the sum. Since the operation code for the TC order is 0, a "TC self" instruction (which, of course, would cause a hardware restart if encountered in the course of a program execution) appears as $0AAAA_8$, where AAAA is the S-register address of the cell in question, in range 2000_8 - 7777_8.

SKEEP6: Self-check register #6, used with SKEEP5 in erasable memory check routine to retain the previous value of the upper half of the pair of cells being checked (cf. SKEEP5). In the fixed memory check routine, is set to -0 to indicate that the "ROPECHK" option is used (automatic check for proper memory sum, part of self-check sequence, with no display unless difficulty); and is set to +1 to indicate that the "SHOWSUM" option is used (enabled by verb 91 from "GOSHOSUM", giving a DSKY display of each bank's sum, number, and "bugger word", with no automatic check for proper memory sum).

SKEEP7: Self check register #7, used in erasable memory check routine to contain the S-register portion of the erasable memory address of the lower half of the pair of cells being checked (see discussion with ERESTORE above). In fixed memory check, is used as an identification of when the two fixed-fixed banks (bank 2 and bank 3, S-register addresses starting at 4000_8 and 6000_8 respectively) are to be read.

SMODE: Single precision cell, scale factor B14, used to control the performance of the computer self-check routines, and examined whenever no active jobs are to be done (and, of course, no tasks). It is set to +0 as part of a fresh start, and can be loaded using N27. A +0 value causes the self-check routine to be bypassed; values of magnitude 9 or more cause SMODE to be reset to 0; a value of 4 causes the erasable memory check to be done; a value of 5 causes the fixed memory check to be done; and other values (-0, 1,2,3,6,7, or 8) cause the complete self-check to be done. If SMODE is positive, an error will cause it to be reset 0, while if it is negative the self-check will be started again (except for a value of -0, which causes the self-check computations to be started at the point after the failure, as determined by SFAIL).

SNANGi (i = X,Y,Z): Value of sine of ANGX, ANGY, and ANGZ, scale factor B-2, stored in push-down list locations 10D, 12D, and 14D respectively.

$SOUTHDR_i$: Indexed cell used in "TORQUE" to retain the value of $DRIFTI_{sp}$ for subsequent use by erasable memory program.

T_{mark}: See Prelaunch Alignment.

[TRANSM1] : Transformation matrix used as a "sway transition matrix", contained in erasable memory (must be initialized to values as part of an erasable memory load before running test). Assumed scaling in this writeup for all elements is B1 (after being used to perform multiplication, a left shift of 1 is done). Values from a previous program (Sundisk, which used fixed memory cells) were:

$$\begin{bmatrix} 0.47408845 & 0.23125894 & 0.14561689 \\ -0.06360691 & -0.16806746 & 0.15582939 \\ -0.06806784 & -0.75079894 & -0.24878704 \end{bmatrix}$$

These values are the "stored" ones, and must be multiplied by two to find the "true" values:

$$\begin{bmatrix} 0.94817690 & 0.46251788 & 0.29123378 \\ -0.12721382 & -0.33613492 & 0.31165878 \\ -0.13613568 & -1.50159788 & -0.49757408 \end{bmatrix}$$

$VLAUN_y$, $VLAUN_z$: Value of "horizontal velocity of launch vehicle" (due to sway) in north-south and east-west directions respectively, scale factor B9, units cm/sec.

WANGI: Value of (- cos C_{atd}), scale factor B0.

WANGO: Value of (sin C_{atd}), scale factor B0.

WANGT: Quantity set to 0 if no torquing is performed and to 1 (scale factor B0) if torquing is to be performed via "EARTHR*" at the start of "$SLEEPIE_d$". Program notation also "TORQNDX, TORQNDX+1". If it is non-zero, WANGI term is included in computation of WPLATI.

WPLATI, WPLATO, WPLATT: Quantities computed in "$ALFLT_d$", scale factor B0, normalized by earth rotation rate (hence are converted to angles, for the one-second gyro drift evaluation cycle, by multiplication by K_{georgk}). The "I" is south, the "O" is vertical ("azimuth"), and the "T" is east.

Values of ALFDK information from an earlier program (Sundisk)

Index	Time	ALTIM	BUFTCPIP	BUFTCERC	BUFSLPAZ	BUFSLPVRD	BUFSLPNSD
0	1-30	-28	.91230833	.81193187	-.00035882	-.00000029	.00013262
12	31-90	-58	.99122133	.98940595	-.00079010	-.00000265	.00043154
24	91-100	-8	.99971021	.99852047	.00042697	-.00000213	.00011864
36	101-200	-98	.99550063	.98992124	.00043452	-.00000401	-.00021980
48	201-450	-248	.99673264	.99365467	.00003767	-.00002317	-.00003305
60	451-790	-338	.99924362	.99888274	.00000064	-.00004012	-.00000195
72	791-1200	-408	.99963845	.99913162	.00000090	.00002927	-.00000026
84	1201-1700	-498	.99934865	.99868793	.00000055	.00001183	-.00000005
96	1701-2100	-398	.99947099	.99894799	.00000018	.00000300	-.00000001
108	2101-2700	-598	.99957801	.99916095	.00000007	.00000096	0
120	2701-3400	-698	.99966814	.99933952	.00000002	.00000028	0
132	3401-4000	-598	.99972716	.99945654	.00000001	.00000010	0

The "index" column gives the value of the ALFDK index required to obtain the ALTIM setting (see "$ALFLT_d$"). The columns are headed with the cells into which the $ALFDK_i$ information is loaded (ALTIMS, a single precision cell, could be loaded with the same value, e.g. -1, for all entries to the table).

V70UPDAT (verb 70)

UPVERBSV = 0

Proceed to second line of "V73UPDAT"

V71UPDAT (verb 71)

UPVERBSV = 1

Proceed to second line of "V73UPDAT"

V72UPDAT (verb 72)

UPVERBSV = 2

Proceed to second line of "V73UPDAT"

V73UPDAT (verb 73)

UPVERBSV = 3

Perform "TESTXACT"

If |MODREG| ≠ 0:

If bit 9(UTFLAG) of FLAGWRD8 = 1:

If MODREG = 20:

Proceed to "UPDATEOK"

If MODREG ≠ 2:

Set bit 7(Operator error) of channel 11 = 1

Set bit 3(Uplink activity) of channel 11 = 0

Proceed to "ENDEXT"

Proceed to "UPDATEOK"

UPDATEOK

UPOLDMOD = MODREG

UPVERB = UPVERBSV

UPCOUNT = 1

DNLSTCOD = 1 (Tag here "UPPART2")

TS = 27 and perform "NEWMODEX"

If UPVERB = 0 or 3:

COMPNUMB = 2

Proceed to "OHWELL2"

Proceed to "OHWELL1"

OHWELL1

MPAC+2 = "UPBUFF"

TS = 2101_{vn}

Proceed to "GOXDSPF": if terminate, proceed to second line of "UPOUT"
if proceed, proceed to second line of "OHWELL1"
otherwise, proceed

If MPAC+0 = 32, proceed to second line of "OHWELL1" (recycle verb)

If UPBUFF - 2 $\leq$ 0, proceed to second line of "OHWELL1"

If UPBUFF - 21 $\geq$ 0, proceed to second line of "OHWELL1"

COMPNUMB = UPBUFF+0

UPCOUNT = UPCOUNT +1

Proceed to "OHWELL2"

OHWELL2

MPAC+2 = "UPBUFF" + UPCOUNT - 1

TS = 2101_{vn}

Proceed to "GOXDSPF": if terminate, proceed to second line of "UPOUT"
if proceed, proceed to second line of "OHWELL2"
otherwise, proceed

If MPAC+0 = 32, proceed to second line of "OHWELL2" (recycle verb)

If COMPNUMB - UPCOUNT $>$ 0:

UPCOUNT = UPCOUNT + 1

Proceed to "OHWELL2"

Proceed to "UPVERIFY"

UPVERIFY

MPAC+2 = "UPTEMP"

TS = 2102_{vn}

Proceed to "GOXDSPF": if terminate, proceed to second line of "UPOUT"
if proceed, proceed to "UPSTORE"
otherwise, proceed

If MPAC+0 = 32, proceed to second line of "UPVERIFY" (recycle verb)

If UPTEMP $\leq$ 0, proceed to "UPVERIFY"

If COMPNUMB + 1 - UPTEMP $\leq$ 0, proceed to "UPVERIFY"

MPAC+2 = "UPBUFF" + UPTEMP - 1

Proceed to second line of "OHWELL2"

UPSTORE

Complement bit 3(VERIFLAG) of FLAGWRD7

If UPVERB $\geq$ 3:

$UPBUFF+8_{dp}$ = $UPBUFF+0_{dp}$

Perform "TIMEDIDL": if error return, proceed
otherwise, skip next line

Set bit 7(Operator error) of channel 11 = 1

Proceed to second line of "UPOUT"

Establish "UPJOB" (priority 30_8) (has a VAC area for "INTSTALL")

End of job

UPJOB

Perform "INTSTALL"

Set bit 13(INTGRAB) of FLGWRD10 = 1

If UPVERB = 0:

$UPBUFF+8_{dp}$ = - $UPBUFF+0_{dp}$

(If UPVERB = 0):

Perform "TIMEDIDL": if error return, proceed
otherwise, skip next 2 lines

Set bit 7 (Operator error) of channel 11 = 1

Proceed to "UPOUT"

$UPBUFF+10_{dp} = - UPBUFF+0_{dp}$

$UPBUFF+12_{dp} = - UPBUFF+0_{dp}$

Set TS = $UPBUFF+10_{dp}$ and $UPBUFF+10_{dp} = 0$

$T_{etcm} = T_{etcm} + TS$

Set TS = $UPBUFF+12_{dp}$ and $UPBUFF+12_{dp} = 0$

$T_{etlm} = T_{etlm} + TS$

Set TS = $UPBUFF+0_{dp}$ and $UPBUFF+0_{dp} = 0$

$T_{eph} = T_{eph} + TS$

Proceed to "UPOUT"

If UPVERB = 1:

EBANK = bits 11-9 of UPBUFF+1

UPTEMP = bits 8-1 of UPBUFF+1

TS = UPTEMP + COMPNUMB - 3

If |TS| > 0:

If bit 9 of TS = 1: (e.g. $\geq 400_8$, indicating next EBANK)

Set bit 7 (Operator error) of channel 11 = 1

Proceed to "UPOUT"

TS = COMPNUMB - 3

$TS_1 = 1400_8 + UPTEMP$

Perform the following for i = TS to i = 0: (interrupts inhibited)

$E_{TS_1+i} = UPBUFF_{2+i}$

i = i - 1

Proceed to "UPOUT"

If UPVERB = 2: (as it will)

If bit 1 of COMPNUMB = 0: (i.e. not an odd number)

Set bit 7 (Operator error) of channel 11 = 1

Proceed to "UPOUT"

TS = COMPNUMB - 2

Perform the following for i = TS to i = 1: (interrupts inhibited)

EBANK = bits 11-9 of $UPBUFF_i$

$TS_1 = 1400_8$ + bits 8-1 of $UPBUFF_i$

$E_{TS_1} = UPBUFF_{1+i}$

i = i -2

Proceed to "UPOUT"

TIMEDIDL

Set $UPBUFF+18_{dp} = T_{now}$ and $T_{now} = 0$

Set TS = $UPBUFF+8_{dp}$ and $UPBUFF+8_{dp} = 0$

TS = TS + $UPBUFF+18_{dp}$

If $|TS| \geq 2^{28}$:

Set TS = $UPBUFF+18_{dp}$ and $UPBUFF+18_{dp} = 0$

$T_{now} = T_{now}$ + TS

Return to calling address +1 (indicating an error)

Force sign agreement of TS

$T_{now} = T_{now}$ + TS

Return to calling address +2 (non-error return)

UPOUT

Proceed to "INTWAKEU" (which exits to next line)

TS = UPOLDMOD and perform "NEWMODEX"

TS = MODREG - 1

If |TS| = 1: (i.e. UPOLDMOD was 0 or 2)

DNLSTCOD = 0

If TS > 1: (i.e. UPOLDMOD was 20)

DNLSTCOD = 2

Set bit 3(Uplink activity) of channel 11 = 0

Proceed to "ENDEXT"

INTWAKEU

If UPSVFLAG ≠ 0:

$\underline{RCV} = \underline{R}_{rect}$

$\underline{VCV} = \underline{V}_{rect}$

$TDELTA\underline{V} = 0$

$TNU\underline{V} = 0$

$T_c = 0$

$XKEP = 0$

If UPSVFLAG ≥ 0:

Perform "MOVEACSM"

If bit 1(AVEMIDSW) of FLAGWRD9 = 0:

$\underline{R} = RC\underline{V} + TDELTA\underline{V}$

$\underline{V} = VC\underline{V} + TNU\underline{V}$

$T_{pptm} = T_{et}$

If UPSVFLAG < 0:

Perform "MOVEALEM"

$\underline{R}_{other} = RC\underline{V} + TDELTA\underline{V}$

$\underline{V}_{other} = VC\underline{V} + TNU\underline{V}$ (Time tag T_{etlm})

Set bit 1(RENDWFLG) of FLAGWRD5 = 0 (Tag here "INTWAKEX")

Channel 77 = 0 (resets restart monitor flip-flops)

UPSVFLAG = 0

QPRET = Return address (to line after next)

Perform "INTWAKE" (starting at 3rd from last line, awaken jobs)

Proceed to second line of "UPOUT"

See also list of major variables and list of routines

COMPNUMB: Single precision cell, scale factor B14, containing the total number of uplink data quantities to be sent. It is set to 2 for verbs 70 and 73; for verb 71 it is set to the first data quantity (and hence must be equal to the number of data words +1 (for address of first word) +1 (for its own setting)); for verb 72, it is likewise set to the first data quantity (since each data word is preceded by its address, it therefore must be 2 x data words +1). For verbs 71 and 72, values of COMPNUMB below 3 (the minimum for one data word) or above 20 are rejected: this gives for verb 71 a maximum of 18 single precision data words (or 9 double precision ones, sufficient e.g. for a 3 x 3 matrix), and for verb 72 a maximum of nine single precision data words.

DNLSTCOD: See Telemetry.

EBANK: See Data Input/Output.

MPAC+0: Cell loaded in display routine with the verb that is received (also used for other functions).

MPAC+2: Cell used for communication purposes with display routine, to provide the address into which a quantity (for e.g. noun 01 or 02) is to be loaded (also used for other functions).

QPRET: See Orbital Integration.

$\underline{R}_{other}$, $\underline{R}_{rect}$: See Orbital Integration.

RC$\underline{V}$: See Orbital Integration.

T_c: See Orbital Integration.

T_{eph}: See Boost Computations.

T_{et}, T_{etcm}, T_{etlm}: See Orbital Integration.

TDELTA$\underline{V}$, TNU$\underline{V}$: See Orbital Integration.

UPBUFF: Set of 20 cells (UPBUFF+0 through UPBUFF+19) used to contain the uplinked information as it is received. Also used for temporary storage of time information, after completion of the update sequence, for verbs 70 and 73. Cells are loaded during the entry phase of flight with guidance parameters for telemetry purposes. The address of UPBUFF+0 is 00304_8. The last 18 cells of the UPBUFF set also are used for storage of $[X_{smd}]$: consequently, if it is desired to uplink this preferred IMU alignment for use in P52/P54, this must be the last uplink that is sent before P52/P54 uses the data.

UPCOUNT: Single precision quantity, scale factor B14, containing the serial number of the UPBUFF cell to be loaded next. If cell UPBUFF+n is to be loaded, UPCOUNT = (n +1). When UPCOUNT = COMPNUMB, it is concluded that the basic load has been completed, and the option for "line-by-line" corrections is enabled, during which UPCOUNT does not change.

UPOLDMOD: Single precision cell, scale factor B14, used to contain the value of MODREG when an update verb is received, and used to restore its proper value (-0, 0, 2, or 20) after the end of the update sequence.

UPSVFLAG: Single precision cell, scale factor B14, assigned a cell in erasable memory just before the first component of $\underline{R}_{rect}$, and set by the update loading process to cause proper processing of a state vector update. The cell is checked in "INTWAKEU", and if non-zero it is concluded that a state vector update was done (if positive, a CSM state vector; if negative, an OWS state vector), and after processing the cell is reset 0 (which is also the initial condition set in "DOFSTART").

UPTEMP: Single precision cell used for temporary storage purposes. In the "line-by-line" correction mode, it contains the loaded value of the component serial number (defined as for UPCOUNT) into which the correction is to be loaded, scale factor B14. The address of UPTEMP is 00330_8.

UPVERB: Single precision quantity, scale factor B14, containing information on the update verb being performed: for verbs 70-73, it is set to 0-3 respectively.

UPVERBSV: Single precision cell, scale factor B14, used to retain information on the update verb received while a determination is made if it is allowed: if it is, then UPVERBSV is loaded into UPVERB.

$\underline{V}_{other}$, $\underline{V}_{rect}$: See Orbital Integration.

VC$\underline{V}$: See Orbital Integration.

XKEP: See Orbital Integration.

Verb 70

Verb 70 could be considered a "liftoff time update". Its transmission format is:

V 70 E XXXXX E XXXXX E (then V 33 E)

where the "X" information is an octal double precision increment to be added to T_{eph} and subtracted from the computer clock and the state vector times for CSM and OWS, scale factor B28, units centi-seconds. If liftoff were sensed 5 seconds late, for example, then the transmission should be:

V 70 E 77777 E 77013 E (V 33 E) ($-77013_8 = 764_8 = 500$ cs)

in order to correct the AGC time information to reflect the proper "liftoff time" (note that V70 keeps the sum of T_{eph} and the computer clock constant by changing them in opposite directions). If only the computer clock (T_{now}) is to be updated, then verb 73 could be used.

Verb 71

Verb 71 is used to perform a "contiguous block update". Its transmission format is:

V 71 E ii E AAAA E XXXXX E XXXXX E ... XXXXX E (V 33 E)

where the "ii" information is the setting for COMPNUMB, and is two more than the number of data words (i.e. number of XXXXX E's +2). The AAAA information is the erasable memory address into which the first XXXXX data word is to be loaded (successive words are loaded into successive cells), and must be sufficiently compatible with the value of ii so as to avoid requiring erasable memory bank switching. AAAA is in ECADR format, meaning that bits 11-9 give the E-bank number and bits 8-1 the address within the bank (which, for hardware reasons, is added to 1400_8 within the program). Since there is no lockout (aside from bank switching constraints) on the value of AAAA, caution must be observed to avoid destroying computer control cells (the "ENDSAFE" lockout present in e.g. Sundisk has been deleted).

Verb 72

Verb 72 is used to perform a "scatter update". Its transmission format is:

V 72 E ii E A_1 A_1 A_1 A_1 E XXXXX E A_2 A_2 A_2 A_2 E XXXXX E ... (V33E)

where the "ii" information is again the setting for COMPNUMB, and is equal to two times the number of data words plus 1 (and hence must be odd, as well as at least 3 and below 21, i.e. maximum 19 here). The A's are specified for each 15-bit word individually, with no constraints imposed by the software on addresses which may be changed.

Verb 73

Verb 73 is used to perform an "octal clock increment" (verb 55 can be used to perform the same function for decimal input of hours, minutes, and centi-seconds), and has the following transmission format:

V 73 E XXXXX E XXXXX E (then V 33 E)

where the "X" information is an octal double precision increment to be added to the computer clock (note that the V70 uplink time is subtracted from the computer clock), scale factor B28, units centi-seconds.

Sample Update Sequences

1. To load the components of [REFSMMAT] (double precision elements, scale factor B1), the following sequence could be used:

V 71 E 24 E 1717 E (COMPNUMB = 20; "REFSMMAT" = 3,1717)

XXXXX E XXXXX E (row 1 column 1)
XXXXX E XXXXX E (row 1 column 2)

etc.

XXXXX E XXXXX E (row 3 column 3)
V 33 E (accept)

2. To load the components of $[X_{smd}]$ (preferred IMU orientation, see Inflight Alignment), the same sequence as in #1 could be used, except that the address instead of being 1717 should be 0306("X_{smd}" = "UPBUFF" +2).

3. To load a CSM state vector update, the following sequence could be used:

V 71 E 21 E 1501 E (COMPNUMB = 17; "UPSVFLAG" = 3,1501)

00001 E
XXXXX E XXXXX E R_{rect_x}
Y-component
Z-component
XXXXX E XXXXX E V_{rect_x}
Y-component
Z-component
XXXXX E XXXXX E T_{et} value
V 33 E (accept)

4. To load an OWS state vector update, the same sequence as in #3 could be used, except that the UPSVFLAG setting, rather than being 00001, should be 77776.

5. To load an External Delta-V (P30) update, the following sequence could be used:

```
V 71 E 12 E  3404 E      (COMPNUMB = 10; "DELVLVC" = 7,1404)

XXXXX E XXXXX E      DELVLVC_x
 Y-component
 Z-component
XXXXX E XXXXX E      T_ig
   V 33 E            (accept)
```

6. To load an External Delta-V (P30) update for deorbit, the following sequence could be used:

```
V 71 E 16 E  3400 E      (COMPNUMB = 14; "LATSPL" = 7,1400)

XXXXX E XXXXX E      LATSPL
XXXXX E XXXXX E      LNGSPL
 (continue on
 as in item #5)
```

7. To perform an update of merely the entry parameters, the sequence of #6 could be used, stopping after LNGSPL (hence first few quantities would be V 71 E 6 E 3400 E).

8. To load new values for erasable memory constants (such as IMU compensation parameters), see the list of cells in Erasable Memory Prelaunch Load (which is arranged in sequence of increasing addresses).

VERBFAN

If (VERBREG - K_{st2cn}) < 0:

Proceed to address specified by $K_{vbt_{VERBREG}}$

TS = VERBREG - K_{st2cn}

Perform "RELDSP"

Proceed to address specified by $K_{st2fn_{TS}}$

ALM/END

Set bit 7(Operator error) of channel 11 = 1

Proceed to "PINBRNCH"

TESTXACT

If EXTVBACT > 0:

Proceed to "ALM/END"

If bits 14(PRIOIDLE), 12(PDSPFLAG), and 7(PROWTKEY) of FLAGWRD4 ≠ 0:

Proceed to "ALM/END" (priority display using DSKY)

EXTVBACT = 00024_8 (sets bits 5 and 3 to 1)

TS = -2 and perform "NVSUB": if busy, proceed (blank R1-R3
otherwise, proceed and noun)

Check for new job waiting to be performed, and do it if required

Return

VBTSTLTS (verb 35)

If MODREG ≠ 0:

Proceed to "ALM/END"

Inhibit interrupts (released in "DELAYJOB" performance)

Set bit 1(LMPTSTBT) of IMODES33 = 1

Set bits 7(Operator error), 6(Flash), 5(Key Release), 4(Temperature Caution), 3(Uplink Activity), and 1(ISS Warning) of channel 11 = 1

DSPTAB+11 = 00650_8 and flag for output at next opportunity
(bit 9 is Program alarm, bit 8 Tracker alarm, bit 6
Gimbal lock, and bit 4 No attitude) (Note if restart, "GOPROG" put IMU into coarse align)

Set bit 10(Test DSKY lights) of channel 13 = 1

Set DSPTAB+0 to DSPTAB+10 so that all registers display "8" and R1, R2, and R3 display plus signs, and flag for output at next opportunity

NOUT = 11

Delay K_{shots} seconds (by putting job to sleep via "DELAYJOB")

Proceed to "TSTLTS3"

TSTLTS3

Set bits 7, 4, 3, and 1 of channel 11 = 0 (interrupts inhibited during routine)

Set bit 10 of channel 13 = 0

TS = bit 4(Coarse align) of channel 12

DSPTAB+11 = TS, and flag for output at next opportunity (bit 4 is No attitude)
(Note that bit 9, Program alarm, set 0 even if FAILREG+0 $\neq$ 0)

Set bit 1(LMPTSTBT) of IMODES33 = 0

Set bits 13-11 (PIP2FLBT, DNLKFAIL, UPLKFAIL) of IMODES33 = 1

Set bit 15(TLIMBIT) of IMODES30 = 0

Set bits 13, 12, and 10 (IMUFLBIT, ICDUFLBT, PIPAFLBT) of IMODES30 = 1

Set bit 7(OCDUFBIT) of OPTMODES = 1

Establish "DSPMMJB" (priority 30_8)

MONSAVE1 = 40000_8 (sets bit 15 to 1)

Set bit 6(Flash) of channel 11 = 0

Perform "RELDSP"

If CADRSTOR $\neq$ 0:

Proceed to "PINBRNCH"

End of job

VBZERO (verb 40)

If IMUCADR ≠ 0:

Proceed to "ALM/END"

Perform "IMUZERO"

Perform "IMUSTALL": if error return, proceed
otherwise, proceed

Proceed to "PINBRNCH"

VBCOARK (verb 41)

If NOUNREG ≠ 20: (ICDU)

If NOUNREG ≠ 91: (OCDU)

Proceed to "ALM/END"

If NOUNREG = 20:

Perform "TESTXACT" (Tag here "IMUCOARK")

TS = 2522_{vn}

Proceed to "GOXDSPF": if terminate, proceed to "ENDEXT"
if proceed, proceed
otherwise, proceed

TS = 4100_{vn}

Perform "EXDSPRET"

If IMUCADR ≠ 0:

Set bit 7(Operator error) of channel 11 = 1

Proceed to "ENDEXT"

Perform "IMUCOARS"

Perform "IMUSTALL": if error return, proceed
otherwise, proceed

Proceed to "ENDEXT"

If MODREG ≠ 0: (N91) (Tag here "OPTCOARK")

Proceed to "ALM/END"

Perform "TESTXACT"

If SWSAMPLE $\leq$ 0: (switch not at computer position)

Set bit 7(Operator error) of channel 11 = 1

Perform "ALARM" (pattern 0115_8)

If OPTIND = -0: (Not expected in view of P00 restriction)

Perform "ALARM" (pattern 0117_8)

Proceed to "ENDEXT"

TS = 2492_{vn}

Proceed to "GOXDSPF": if terminate, proceed to "ENDEXT"
if proceed, proceed
otherwise, proceed

DESOPTS = SAC (loaded in R1 of N92)

DESOPTT = PAC (loaded in R2 of N92)

TS = 4100_{vn}

Perform "EXDSPRET"

OPTIND = 1

Proceed to "ENDEXT"

IMUFINEK (verb 42)

Perform "TESTXACT"

TS = 2593_{vn}

Proceed to "GOXDSPF": if terminate, proceed to "ENDEXT"
if proceed, proceed
otherwise, proceed

TS = 4200_{vn}

Perform "EXDSPRET"

If IMUCADR $\neq$ 0:

Set bit 7(Operator error) of channel 11 = 1

Proceed to "ENDEXT"

Perform "IMUFINE"

Perform "IMUSTALL": if error return, proceed to "ENDEXT"
otherwise, proceed

TS = "OGC"

Perform "IMUPULSE"

Perform "IMUSTALL": if error return, proceed
otherwise, proceed

Proceed to "ENDEXT"

IMUATTCK (verb 43)

If MODREG $\neq$ 0, proceed to "ALM/END"

If bits 4-5 (IMU Coarse align, IMU Zero) of channel 12 $\neq 00_2$:

Proceed to "ALM/END"

Perform "CKLFTBTS": if after liftoff, proceed
if before liftoff, skip next line

Perform "TESTXACT"

Set bits 6(Enable CDU IMU Error Counters) and 4 (IMU Coarse Align)
of channel 12 = 0

TS = 2522_{vn}

Proceed to "GOXDSPF": if terminate, proceed to "TRMATTCK"
if proceed, proceed
otherwise, proceed

Perform "NEEDLE11"

Perform "NEEDLER2"

Call "ATTCK1" in 0.02 seconds

Proceed to "TRMATTCK"

TRMATTCK

Perform "CKLFTBTS": if after liftoff, proceed to "ENDEXT"
if before liftoff, proceed

Proceed to "PINBRNCH"

ATTCK1

AK = THETAD

Perform "NEEDLES"

End of task

CKLFTBTS

If bit 5(BKUPLO) of FLAGWRD5 = 0:

If bit 5(Liftoff complement) of channel 30 = 1:

Return to calling address +2 (before liftoff)

Return to calling address +1 (after liftoff)

DKDISP (verb 44)

See Digital Autopilot Interface Routines

DKDAPON (verb 45)

See Digital Autopilot Interface Routines

STABLISH (verb 46)

See Digital Autopilot Interface Routines

LMTOCMSV (verb 47)

Establish "LMTOCM" (priority 10_8)

End of job

LMTOCM

Perform "INTSTALL"

Inhibit interrupts

$\underline{R}_{rectcm} = \underline{R}_{rectlm}$

$\underline{V}_{rectcm} = \underline{V}_{rectlm}$

$T_{etcm} = T_{etlm}$

$DELTAV_{cm} = DELTAV_{lm}$

$NUV_{cm} = NUV_{lm}$

$RCV_{cm} = RCV_{lm}$

$VCV_{cm} = VCV_{lm}$

$T_{ccm} = T_{clm}$

$XKEP_{cm} = XKEP_{lm}$

Proceed to "TACHEXIT"

TACHEXIT

Release interrupts

Perform "MOVEPCSM"

If bit 1(AVEMIDSW) of FLAGWRD9 = 0:

$\underline{R} = RC\underline{V} + TDELTA\underline{V}$

$\underline{V} = VC\underline{V} + TNU\underline{V}$

$T_{pptm} = T_{et}$

$\underline{R}_{other} = RC\underline{V} + TDELTA\underline{V}$

$\underline{V}_{other} = VC\underline{V} + TNU\underline{V}$

Perform "INTWAKE" (starting at 3rd from last line, awaken jobs)

Proceed to "PINBRNCH"

DAPDISP (verb 48)

See Digital Autopilot Interface Routines

CREWMANU (verb 49)

If MODREG $\neq$ 0, proceed to "ALM/END"

Perform "TESTXACT"

Establish "R62DISP" (priority 10_8)

End of job

GOLOADLV (verbs 50, 51, 53, 97, and 99)

Set bit 6(Flash) of channel 11 = 0

Proceed to "LOADLV"

GOTOR23 (verb 54)

Perform "TESTXACT"

If bit 7(RNDVZFLG) of FLAGWRD0 = 1:

If bit 5(TRACKFLG) of FLAGWRD1 = 1:

Establish "R23CSM" (priority 16_8)

End of job

Perform "ALARM" (pattern 0406_g)

Proceed to "ENDEXT"

ALINTIME (verb 55)

Perform "TESTXACT"

TS = 2524_{vn}

Proceed to "GOXDSPF": if terminate, proceed to "ENDEXT"
if proceed, proceed to "ENDEXT"
otherwise, proceed

If MPAC+0 $\neq$ 23: (final verb not 23)

Proceed to "ENDEXT"

Inhibit interrupts

Set TS = T_{now} and T_{now} = 0

TS = TS + DSPTEM2+1_{dp}, with sign agreement forced (DSPTEM2+1_{dp} contents destroyed)

$T_{now} = T_{now} + TS$

Release interrupts

Proceed to "ENDEXT"

TRACKTRM (verb 56) Also entered from R60 or V34E to V06N49 in P20

Set bit 2(R67FLAG) of FLAGWRD8 = 0

If bit 9(UTFLAG) of FLAGWRD8 = 0:

If bit 7(RNDVZFLG) of FLAGWRD0 = 0:

Proceed to "PINBRNCH"

Set bit 7(AUTOSEQ) of FLGWRD10 = 0

Set bit 14(R21MARK) of FLAGWRD2 = 0

Set bit 7(UPDATFLG) of FLAGWRD1 = 0

If bit 5(TRACKFLG) of FLAGWRD1 = 0:

Set bit 7(RNDVZFLG) of FLAGWRD0 = 0

Set bit 9(UTFLAG) of FLAGWRD8 = 0

Proceed to "PINBRNCH"

Set bit 5(TRACKFLG) of FLAGWRD1 = 0

Set bit 8(IMUSE) and bit 7(RNDVZFLG) of FLAGWRD0 = 0

Set bit 9(UTFLAG) of FLAGWRD8 = 0

Perform "INITSUBA"

Perform "INTSTALL"

Make restart groups 1 and 2 inactive

Inhibit interrupts

Proceed to "ENEMA"

V57CALL (verb 57)

Perform "TESTXACT"

OPTIONX = 4

TS = 0

If bit 2(FULTKFLG) of FLGWRD10 = 1: (bit used in "AUTOW", zero means have VHF and optics)

TS = 1

OPTIONX+1 = TS

TS = 0412_{vn}

Proceed to "GOXDSPF": if terminate, proceed to "ENDEXT"
if proceed, proceed to "ENDEXT"
otherwise, proceed

If OPTIONX+1 = 0:

Set bit 2(FULTKFLG) of FLGWRD10 = 0

NOTE that action taken based on data enter, not PRO, giving difficulty for display interruptions

If OPTIONX+1 ≠ 0:

Set bit 2(FULTKFLG) of FLGWRD10 = 1

Proceed to third line of "V57CALL"

ENATMA (verb 58)

Set bit 15(V50N18FL) of FLAGWRD3 = 1

Set bit 14(STIKFLAG) of FLAGWRD1 = 0

Proceed to "PINBRNCH"

DKJTENBL (verb 59)

CH5FAIL = 0

CH6FAIL = 0

Proceed to "PINBRNCH"

V60 (verb 60)

CPHIX = CDU

Proceed to "PINBRNCH"

V61 (verb 61)

Set bit 9(NEEDLFLG) of FLAGWRD0 = 0 (display DAP error)

Proceed to "PINBRNCH"

V62 (verb 62)

Set bit 6(N22ERNDS) of FLAGWRD9 = 1 (display N22 error)

Set bit 9(NEEDLFLG) of FLAGWRD0 = 1

Proceed to "PINBRNCH"

V63 (verb 63)

Set bit 6(N22ERNDS) of FLAGWRD9 = 0 (display N17 error)

Set bit 9(NEEDLFLG) of FLAGWRD0 = 1

Proceed to "PINBRNCH"

R64 (verb 64)

Perform "TESTXACT"

Establish "V64PERF" (priority 10_8)

End of job

CKOPTVB (verb 65)

If MODREG ≠ 02:

Proceed to "ALM/END"

Establish "GCOMPVER" (priority 16_8)

Proceed to "PINBRNCH"

ATTACHED (verb 66)

Establish "ATTACHIT" (priority 10_8)

End of job

ATTACHIT

Perform "INTSTALL"

Inhibit interrupts

$\underline{R}_{rectlm} = \underline{R}_{rectcm}$

$\underline{V}_{rectlm} = \underline{V}_{rectcm}$

$T_{etlm} = T_{etcm}$

$DELTAV_{lm} = DELTAV_{cm}$

$NUV_{lm} = NUV_{cm}$

$RCV_{lm} = RCV_{cm}$

$VCV_{lm} = VCV_{cm}$

$T_{clm} = T_{ccm}$

$XKEP_{lm} = XKEP_{cm}$

Proceed to "TACHEXIT" (NOTE that Average-G cells R and V overwritten)

V67 (verb 67)

Perform "TESTXACT"

Establish "V67CALL" (priority 05_8)

End of job

V70UPDAT (verb 70)

See Uplink Processing

V71UPDAT (verb 71)

See Uplink Processing

V72UPDAT (verb 72)

See Uplink Processing

V73UPDAT (verb 73)

See Uplink Processing

DNEDUMP (verb 74)

DNTMGOTO = "DNDUMPI"

Proceed to "PINBRNCH"

LFTFLGON (verb 75)

Set bit 5(BKUPLO) of FLAGWRD5 = 1

Proceed to "PINBRNCH"

V76CALL (verb 76)

Perform "TESTXACT"

TS = 0672_{vn}

Proceed to "GOXDSPF": if terminate, proceed to "ENDEXT"
if proceed, proceed
otherwise, proceed to previous line

Inhibit interrupts

Set bit 2(R27UP2) and bit 1(R27UP1) of FLGWRD11 = 0

Set bit 12(R27FLAG) of FLGWRD11 = 1

Proceed to "ENDEXT"

V77CALL (verb 77)

Set bit 12(R27FLAG) of FLGWRD11 = 0 (contrary to "RESETVHF", Tracker light left alone)

Proceed to "PINBRNCH"

CHAZFOGC (verb 78)

If MODREG $\neq$ 02:

Proceed to "ALM/END"

Establish "AZMTHCG1" (priority 16_8)

Proceed to "PINBRNCH"

LEMVEC (verb 80)

Set bit 8(CSMUPDAT) of FLAGWRD1 = 0

Proceed to "PINBRNCH"

CSMVEC (verb 81)

Set bit 8(CSMUPDAT) of FLAGWRD1 = 1

Proceed to "PINBRNCH"

V82PERF (verb 82)

Perform "TESTXACT"

Change priority of present job to 07_8

Proceed to "V82CALL"

V83PERF (verb 83)

Perform "TESTXACT"

Set bit 4(R31FLAG) of FLAGWRD9 = 1

Establish "R31CALL" (priority 05_8)

End of job

V85PERF (verb 85)

Perform "TESTXACT"

Set bit 4(R31FLAG) of FLAGWRD9 = 0

Establish "R31CALL" (priority 05_8)

End of job

V86PERF (verb 86)

If MRKBUF1 $\geqslant$ 0: (see notes with "MKREJECT")

MRKBUF1 = -1

Proceed to "PINBRNCH"

If bit 7(R22CAFLG) of FLAGWRD9 = 1:

Set bit 12(REJCTFLG) of FLGWRD10 = 1

Proceed to "PINBRNCH"

SETVHFLG (verb 87)

Set bit 9(VHFRFLAG) of FLAGWRD9 = 1

Proceed to "PINBRNCH"

RESETVHF (verb 88)

Inhibit interrupts

Set bit 8(Tracker) of DSPTAB+11 = 0, and flag for output at next opportunity

Set bit 7(OCDUFBIT) of OPTMODES = 1

Release interrupts

Set bit 9(VHFRFLAG) of FLAGWRD9 = 0

Proceed to "PINBRNCH"

V89PERF (verb 89)

If MODREG $\neq$ 0, proceed to "ALM/END"

Perform "TESTXACT"

Establish "V89CALL" (priority 10_8)

End of job

V90PERF (verb 90)

Perform "TESTXACT"

Establish "R36" (priority 07_8)

End of job

GOSHOSUM (verb 91)

If MODREG $\neq$ 0, proceed to "ALM/END"

Perform "TESTXACT"

Proceed to "SHOWSUM+2"

WMATRXNG (verb 93)

Set bit 1(RENDWFLG) of FLAGWRD5 = 0

Proceed to "PINBRNCH"

VERB96 (verb 96)

Set bit 5(QUITFLAG) of FLAGWRD9 = 1

Set bit 3(V96ONFLG) of FLAGWRD8 = 1 (reset in "STATINT1")

TS = 0

Proceed to "V37"

See also list of major variables and list of routines

AK: See Digital Autopilot Interface Routines.

CADRSTOR: See Data Input/Output.

CH5FAIL, CH6FAIL: See Digital Autopilot Docked Jet Selection.

CPHIX: See Digital Autopilot RCS Routines.

DELTAV$_{cm}$, DELTAV$_{lm}$: See Orbital Integration.

DESOPTS, DESOPTT: See Optics Computations.

DNTMGOTO: See Telemetry.

EXTVBACT: Single precision cell several of whose bits are used to control various performance features of the extended verbs. If the cell is non-zero, this means that the "extended verb display system" is busy, and other users are locked out (hence the cell can be set non-zero deliberately to lock out other users). The individual bits are used as follows:

Bit	Use
14	Set at start of "P61" and in "NEWRNVN" to inhibit extended verbs from using coding also used by P61 computations (e.g. R30 time-of-flight information).
12	Bit set 1 at end of "COMPDISP", used in "R31CALL" (for R31 and R34) to delay start of first display until completion of loading of data cells.
5	Bit set 1 by "TESTXACT", and used in "COMPDISP", "TICKTEST", and "V82GON1" to indicate, if 0, that a display response has been received (hence the cycling for updating of display should be stopped).
3	Set 1 in "TESTXACT" to indicate that display system extended verb portion is "busy" (not in general set for those extended verbs that do not require use of the display system "mark/extended verb" priority).
2	Set 1 in "TESTMARK" to indicate that optics marking system is in use.

IMODES30, IMODES33: See IMU Computations.

IMUCADR: See IMU Computations.

K_{shots}: Single precision constant, program notation "SHOLTS", scale factor B14, units centi-seconds. Value is 500 x 2^{-14}, corresponding to 5 seconds.

K_{st2cn}: Single precision constant, program notation "LST2CON", scale factor B14, value 40. VERBREG values of this amount or greater are considered "extended verbs".

K_{st2fn_i}: Table of single precision addresses, program notation "LST2FAN", giving (in the form of TC/TCF orders) starting addresses for processing of VERBREG values between K_{st2cn} and 99. See table below.

K_{vbt_i}: Table of single precision addresses, program notation "VERBTAB", giving (in the form of CADR addresses) the starting addresses for processing of VERBREG values less than K_{st2cn}. See table below.

MONSAVE1: See Data Input/Output.

MPAC+0: See Display Interface Routines (loaded with verb that is received).

MRKBUF1: See Optics Computations.

NOUNREG: See Data Input/Output.

NOUT: See General Program Control.

$NU\underline{V}_{cm}$, $NU\underline{V}_{lm}$: See Orbital Integration.

OGC: See Coordinate Transformations (scale factor could be considered B21 in units of gyro pulses as well as B0 revolutions): is loaded by N93.

OPTIND: See Optics Computations.

OPTIONX: See Display Computations.

OPTMODES: See Optics Computations.

PAC: See Coordinate Transformations (loaded by N92).

$\underline{R}_{other}$: See Orbital Integration.

$\underline{R}_{rectcm}$, $\underline{R}_{rectlm}$: See Orbital Integration.

$RC\underline{V}$, $RC\underline{V}_{cm}$, $RC\underline{V}_{lm}$: See Orbital Integration.

SAC: See Coordinate Transformations (loaded by N92).

SWSAMPLE: See Optics Computations.

T_{ccm}, T_{clm}: See Orbital Integration.

T_{et}, T_{etcm}, T_{etlm}: See Orbital Integration.

TDELTA$\underline{V}$, TNU$\underline{V}$: See Orbital Integration.

$\underline{V}_{other}$: See Orbital Integration.

$\underline{V}_{rectcm}$, $\underline{V}_{rectlm}$: See Orbital Integration.

VC$\underline{V}$, VC$\underline{V}_{cm}$, VC$\underline{V}_{lm}$: See Orbital Integration.

VERBREG: See Data Input/Output.

$XKEP_{cm}$, $XKEP_{lm}$: See Orbital Integration.

Verb Table Information

Verb	Starts at	Function
00	"DSPALARM"	Not assigned.
01	"DSPA"	Display in octal first component.
02	"DSPB"	Display in octal second component.
03	"DSPC"	Display in octal third component.
04	"DSPAB"	Display in octal first and second components.
05	"DSPABC"	Display in octal all three components.
06	"DECDSP"	Decimal display.
07	"DSPDPDEC"	Double precision decimal display.
08-10	"DSPALARM"	Not assigned.
11-17	"MONITOR"	Perform monitor function of type specified by least significant digit of verb (e.g. verb 13 does "DSPC" periodically).
18-20	"DSPALARM"	Not assigned.
21	"ALOAD"	Load first component.
22	"BLOAD"	Load second component.
23	"CLOAD"	Load third component.
24	"ABLOAD"	Load first and second components.
25	"ABCLOAD"	Load all three components.
26	"DSPALARM"	Not assigned.
27	"DSPFMEM"	Display contents of fixed memory.
28-29	"DSPALARM"	Not assigned.
30	"VBRQEXEC"	Request executive system (for a job).
31	"VBRQWAIT"	Request waitlist system (for a task).
32	"VBRESEQ"	Recycle verb (same calling-routine return as a data enter).
33	"VBPROC"	Proceed (without data) verb.

Verb	Starts at	Function
34	"VBTERM"	Terminate (activity or function) verb.
35	"VBTSTLTS"	Test lights of display system.
36	"SLAP1"	Perform a (manually initiated) fresh start.
37	"MMCHANG"	Change program ("major mode").
38-39	"DSPALARM"	Not assigned.
40	"VBZERO"	Zero IMU CDU's.
41	"VBCOARK"	Coarse align (IMU CDU N20; optics CDU N91).
42	"IMUFINEK"	Fine align IMU (optional pulse torquing).
43	"IMUATTCK"	Load FDAI attitude error needles.
44	"DKDISP"	Load Docked DAP (R04).
45	"DKDAPON"	Activate Docked DAP.
46	"STABLISH"	Activate Undocked DAP.
47	"LMTOCMSV"	Move OWS state vector into CSM state vector.
48	"DAPDISP"	Load Undocked DAP (R03).
49	"CREWMANU"	Start crew-defined maneuver (R62).
50	"GOLOADLV"	Please perform.
51	"GOLOADLV"	Please mark.
52	"ALM/END"	Not assigned.
53	"GOLOADLV"	Please mark alternate LOS.
54	"GOTOR23"	Start rendezvous backup sighting mark routine (R23).
55	"ALINTIME"	Increment computer clock (decimal input data).
56	"TRACKTRM"	Terminate tracking (P20).
57	"V57CALL"	Select FULTKFLG option (number of sensors).
58	"ENATMA"	Enable automatic attitude maneuvers in R61 and R67.
59	"DKJTENBL"	Enable all jets in Docked DAP.

Verb	Starts at	Function
60	"V60"	Set CPHIX (N17) = CDU (N20).
61	"V61"	Display autopilot following error (the DAP "Mode I") on FDAI error needles.
62	"V62"	Display autopilot total attitude error with respect to N22 (the DAP "Mode II") on FDAI error needles.
63	"V63"	Display autopilot total attitude error with respect to N17 (the DAP "Mode III") on FDAI error needles.
64	"R64"	Start Optics Angle Transform calculation (R64).
65	"CKOPTVB"	Optical verification of prelaunch alignment (P03).
66	"ATTACHED"	Move CSM state vector into OWS state vector.
67	"V67"	Start W-matrix RSS error display.
68	"ALM/END"	Not assigned.
69	(one-step loop)	Cause a hardware restart (one-step loop).
70	"V70UPDAT"	Liftoff time update (P27).
71	"V71UPDAT"	Block address update (P27).
72	"V72UPDAT"	Single address update (P27).
73	"V73UPDAT"	Increment computer clock (P27). Input is in units of centi-seconds, as contrasted with the hours, minutes, seconds input of V55.
74	"DNEDUMP"	Initialize downlink erasable memory dump.
75	"LFTFLGON"	Set liftoff flag (for backup liftoff).
76	"V76CALL"	Enable R27 in R22.
77	"V77CALL"	Disable R27 in R22.
78	"CHAZFOGC"	Change gyrocompass launch azimuth.
79	"ALM/END"	Not assigned.

Verb	Starts at	Function
80	"LEMVEC"	Cause OWS state vector to be updated by navigation measurements (in P20).
81	"CSMVEC"	Cause CSM state vector to be updated by navigation measurements (in P20).
82	"V82PERF"	Request orbital parameter display (R30).
83	"V83PERF"	Request rendezvous parameter display #1 (R31, +X axis angle).
84	"ALM/END"	Not assigned.
85	"V85PERF"	Request rendezvous parameter display #2 (R34, optics angle).
86	"V86PERF"	Reject rendezvous backup sighting mark.
87	"SETVHFLG"	Set VHF range flag (for R22).
88	"RESETVHF"	Reset VHF range flag (for R22).
89	"V89PERF"	Start rendezvous final attitude routine (R63).
90	"V90PERF"	Request rendezvous out-of-plane display (R36).
91	"GOSHOSUM"	Compute banksums (of each fixed memory bank).
92	"ALM/END"	Not assigned.
93	"WMATRXNG"	Reset W matrix flag (RENDWFLG) to enable initialization.
94	"ALM/END"	Not assigned.
95	"ALM/END"	Not assigned.
96	"VERB96"	Terminate integration and go to P00.
97	"GOLOADLV"	Please perform engine-fail logic (R40).
98	"ALM/END"	Not assigned.
99	"GOLOADLV"	Please enable engine ignition.

Index of Routines

The routines listed below are those included by specific tag in this document. They are arranged in the "alphabetical" order of the listing of symbols at the end of the program: "+", "*", "-", "/", A-Z, and 0-9.

Symbol	Document
CHARALRM	DATA-3
CHARIN	DATA-2
CHAZFOGC	VBDF-12
CHECKNJ	GENP-27
CHKCOMED	PREL-3
CHKLINUS	ATTM-3
CKLFTBTS	VBDF-6
CKMID2	ORBI-6
CKOPTVB	VBDF-10
CLEANDSP	DINT-1
CLEAR	DATA-7
CLEARMRK	DINT-1
CLOAD	DATA-18
CLOCKJOB	BURN-25
CLOCPLAY	DINT-1
CLOKTASK	BURN-24
CLUPDATE	BURN-16
CM/DAPON	DPEN-1
CM/FDAIR	DPEN-10
CM/POSE	ENRY-1
CMDSOUT	DPTV-11
CNTRCOPY	DPTV-6
COARS	IMUC-12
COARS2	IMUC-12
COE	REND-24
COMADRS	TEST-4
COMMNOUT	CONC-10
COMMONLM	CONC-8
COMPDISP	DISP-11
COMPMATX	ATTM-5
COMPTEST	DATA-13
COM52	OPTC-14
CONSTD	ENRY-12
COUPLE	DPDS-2
CPLMATH	DPDS-10
CREWMANU	VBDF-7
CRS61.1	ATTM-16
CRS61.2A	ATTM-17
CSCTOMAN	OPTC-7
CSCTOZOP	OPTC-7
CSMCONIC	ORBI-2
CSMPREC	ORBI-2
CSMVEC	VBDF-13
CTASK	ORVN-9
CYCVHF	MEAS-14
C13STALL	MEAS-22
C33TEST	IMUC-6
DAPDISP	DPIR-2
DAPINIT	DPTV-3

Symbol	Document
DCMTOCDU	ATTM-9
DCOMPTST	DATA-13
DECDSP	DATA-14
DECDSP3	NNDF-2
DECEND	DATA-4
DEGINSF	NNDF-7
DEGINSF2	NNDF-7
DEGOUTSF	NNDF-2
DELAYJOB	GENP-23
DELCOMP	ATTM-8
DELRSPL	DISP-7
DELTIME	CONC-5
DIFEQ+2	ORBI-16
DISPALM	BURN-4
DISP45A	REND-21
DKDAPON	DPIR-3
DKDISP	DPIR-3
DKJSLECT	DPDS-1
DKJTENBL	VBDF-10
DKT6	DPDS-16
DNDUMP	TELE-5
DNDUMP1	TELE-4
DNDUMP1	TELE-4
DNEDUMP	VBDF-12
DNPHASE1	TELE-1
DNPHASE2	TELE-1
DODOWNTM	TELE-1
DOFSTART	GENP-8
DONOUN46	DPIR-2
DONOUN89	DPIR-4
DONTPULS	IMUC-18
DOV6N78	ORVN-1
DOW..	ORBI-17
DPFRACIN	NNDF-10
DPFRACOT	NNDF-7
DPINSF	DATA-22
DPINSF2	NNDF-8
DPINSF4	NNDF-10
DPOUT	DATA-15
DPTEST	NNDF-1
DP1OUTSF	NNDF-2
DP2OUTSF	NNDF-3
DP3OUTSF	NNDF-3
DSPA	DATA-12
DSPAB	DATA-12
DSPABC	DATA-12
DSPALARM	DATA-28
DSPB	DATA-12
DSPC	DATA-12

Symbol	Document
DSPCOM2	DATA-13
DSPDCEND	DATA-15
DSPDCPUT	DATA-14
DSPDCWD1	DATA-25
DSPDC2NR	DATA-25
DSPDECVN	DATA-26
DSPDECWD	DATA-25
DSPDPDEC	DATA-15
DSPFMEM	DATA-24
DSPIN	DATA-27
DSPIN1	DATA-27
DSPMMJB	DATA-33
DSPOCTWD	DATA-26
DSPOUTSB	GENP-7
DSPSIGN	DATA-24
DSP2DEC	DATA-25
DUMMYJOB	GENP-27
DXCOMP	CONC-3
DYNDISP	BURN-25
DZ1	DPEN-6
EARROT1	COOR-7
EARROT2	COOR-7
EARTHMX	COOR-10
EARTHR*	PREL-6
ECENAB	OPTC-7
EHCOMP	REND-17
ELCALC	REND-2
ENABL2	DPTV-10
ENATMA	VBDF-9
ENDEXIT	ENRY-16
ENDEXT	DINT-1
ENDFIX	MEAS-21
ENDIDLE	DINT-16
ENDIMU	IMUC-18
ENDINT	ORBI-2
ENDMANU	ATTM-13
ENDMANUV	ATTM-3
ENDMANU1	ATTM-2
ENDP76	BURN-33
ENDRET	DINT-18
ENDRET2	DINT-18
ENDTFF	DISP-17
ENDTNON	IMUC-9
ENDZOPT	OPTC-6
ENEMA	GENP-12
ENGINOFF	BURN-28
ENGOFF	STER-9
ENTANSWR	OPTC-8
ENTER	DATA-8
ENTPASO	DATA-8
ERASCHK	TEST-2
ERASLOOP	TEST-2
ERROR	DATA-6
ERRORS	TEST-1
ESTIMS	PREL-2
$ESTIMS_d$	TEST-9
EXDAP	DPEN-11
EXDAPIN	DPEN-5
EXDSPRET	DINT-2
EXRSTRT	DPTV-11
FAZAB3	MEAS-2
FETCH2WD	TELE-3
FINDGIMB	ATTM-5
FIXCW	DPIR-10
FIXDB	ATTM-14
FLASHSUB	DINT-15
FORCE	DPDS-3
FREEFUNC	DPRC-9
FWDFLTR	DPTV-9
FXADRS	TEST-5
FXFX	TEST-7
GAMCOMP	ORBI-13
GCOMPVER	PREL-7
GCOMP5	PREL-9
GEOIMUTT	TEST-8
GEOM	CONC-13
GET.LVC	DISP-1
GET+MGA	DISP-1
GETERAD	COOR-8
GETINREL	DATA-4
GETON2	DPEN-7
GETRJ	REND-23
GETUM	MEAS-4
GETX	CONC-13
GLIMITER	ENRY-15
GLOCKMON	IMUC-7
GOBAQUE	ORBI-14
GOCYCLE	DPDS-11
GODSP	DINT-2
GODSPR	DINT-2
GODSPRET	DINT-2
GODSPRS+1	DINT-7
GOESTIMS	PREL-2
$GOESTIMS_d$	TEST-8
GOFLASH	DINT-2
GOFLASHR	DINT-2
GOFLASH2+1	DINT-7
GOLOADLV	VBDF-7
GOMARKFR	DINT-3
GOMARK2	DINT-3
GOMARK4	DINT-3

Symbol	Document
LOADLV	DATA-19
LOCSKIRT	ATTM-9
LODNNTAB	NNDF-1
LOG	MATH-5
LONGPASS	ORVN-11
LSPOS	COOR-7
L355	ENRY-16
M/SOUT	NNDF-4
MAKEPLAY	DINT-7
MANUSTOP	ATTM-13
MARKDIF	OPTC-10
MARKDONE	OPTC-10
MARKMONR	DINT-5
MARKPLAY	DINT-12
MARKRUPT	OPTC-9
MASSPROP	DPIR-10
MERGMATH	DPDS-7
MIDTOAV1	ORBI-5
MIDTOAV2	ORBI-6
MINKDISP	MINK-1
MKREJECT	OPTC-11
MKRELEAS	OPTC-8
MKVBDSP	OPTC-8
MKVB5X	OPTC-8
MKVB50	OPTC-9
MKVB51	OPTC-8
MMCHANG	DATA-28
MONDO	DATA-23
MONITOR	DATA-22
MONREQ	DATA-23
MOVEACSM	ORBI-7
MOVEALEM	ORBI-7
MOVEPCSM	ORBI-8
MOVEPLEM	ORBI-8
MXM3	ATTM-9
NBDONLY	IMUC-2
NBD2	IMUC-2
NBRANCH	ORBI-15
NCCDMP7+17	BURN-6
NCCLOOP	REND-15
NC1LOOP	REND-14
NC2LOOP	REND-14
NC12D	REND-15
NC12DH	BURN-3
NC12F	REND-13
NC12G	REND-16
NC12H	BURN-4
NC12J	REND-12
NC12OUT	REND-18
NC12ST	REND-10
NDUTINPT	ORVN-3
NEEDLER	DPIR-9
NEEDLER2	DPIR-9
NEEDLES	DPIR-9
NEEDLE11	DPIR-9
NEGRDIST	DPDS-4
NEGSGN	DATA-5
NEGTESTS	ENRY-11
NEWANGL	ATTM-11
NEWDELHI	ATTM-11
NEWMODEX	DATA-33
NEWRNVN	ENTP-1
NEWSTATE	CONC-10
NEXTCOL	ORBI-17
NEXTINSL	TELE-4
NOBYPASS	DPDS-2
NODSPOUT	GENP-7
NODSPY	GENP-6
NOGO	ATTM-10
NOPOLYM	BOOS-4
NORMLIZE	GENP-1
NORMRET	DINT-17
NOROLL	DPTV-14
NOROLL1	DPTV-15
NOTIMING	DPDS-14
NOUN	DATA-5
NOUNTEST	DATA-19
NO5PT	DPDS-13
NO61PT	DPDS-12
NO62PT	DPDS-13
NUM	DATA-3
NVDSP	DINT-13
NVMONOPT	DATA-30
NVSUB	DATA-30
NVSUBEND	DATA-32
NVSUBUSY	DINT-15
NV50DSP	DINT-13
NXTBNK	TEST-6
N72STUFF	ORVN-9
N81DISP	BURN-7
OASTAR	INFA-17
OAVECS	INFA-16
OBLATE	ORBI-14
OCCULT	INFA-6
OCDUFTST	OPTC-6
OHWELL1	UPLK-2
OHWELL2	UPLK-2
OKTOCOPY	DINT-10

Symbol	Document
OKTOENT	DINT-17
OLPALRM	BURN-5
ONCEMORE	TEST-12
ONROLL	DPTV-12
OPDEGOUT	NNDF-3
OPTDEGIN	NNDF-9
OPTMON	OPTC-3
OPTTEST	OPTC-1
P/YWALRM	DPDS-5
P/YWCOMP	DPDS-2
PARAM	CONC-12
PASSOUT	ORVN-13
PASS2	DPDS-6
PASTEVB	DATA-24
PCOPY	DPTV-8
PERF20	MINK-7
PERIODCH	CONC-2
PFAILOK	IMUC-9
PHICALC-11	DISP-9
PHICOMP	ORVN-10
PHIJOB	ORVN-10
PHSMCH	REND-11
PICEND	INFA-7
PIC1	INFA-5
PIC3	INFA-5
PIKUP20	ORVN-4
PINBRNCH	DINT-5
PIPACHK	TEST-13
PIPASR	IMUC-1
PIPFREE	IMUC-14
PIPUSE	IMUC-14
PITCHDAP	DPTV-6
PITCHTIM	DPRC-22
PLANET	INFA-8
PLAYJUM1	DINT-13
POLYCOEF	CONC-16
POODOO	GENP-26
POSGN	DATA-5
POSN17C	PREL-2
POSTAND	GENP-23
POSTBURN	BURN-28
POST41	BURN-29
PREANGLE	ORVN-10
PRECOMP	DPTV-10
PRECSET	REND-1
PREDICT3	ENRY-13
PREFINAL	ENRY-12
PREREAD	GENP-1
PREREAD1	GENP-1

Symbol	Document
PRERRORS	TEST-1
PRE40.6	DPIR-5
PRIODSP	DINT-5
PRIODSPR	DINT-5
PRIOLARM	GENP-25
PROCEEDE	GENP-8
PROCKEY	DATA-30
PROG20	ORVN-1
PROG21	ORVN-5
PROG52	INFA-3
PUTCOM	DATA-20
PUTDCSF2	NNDF-7
PUTDECSF	DATA-21
PUTMARK	OPTC-10
P06	GENP-22
P11	BOOS-1
P20OPT	ORVN-2
P20TRACK	ORVN-5
P21PROG1	ORVN-5
P21PROG2	ORVN-6
P25CSM	ORVN-7
P29	ORVN-11
P30	BURN-1
P31	BURN-2
P32	BURN-5
P33	BURN-6
P33/P73B	BURN-8
P34	BURN-8
P34/P74C	BURN-12
P35	BURN-11
P35/P75B	BURN-16
P36	BURN-16
P37	BURN-17
P38	BURN-18
P38RECYC	BURN-19
P40BLNKR	BURN-22
P40CSM	BURN-20
P40RCS	BURN-28
P40S/F	BURN-20
P40S/SV	BURN-22
P40SXTY	BURN-20
P41CSM	BURN-29
P47BODY	BURN-30
P47CSM	BURN-29
P48CSM	BURN-30
P50	INFA-15
P51	INFA-1
P51A	INFA-1
P51B	INFA-2

Symbol	Document
R52H	OPTC-13
R53	OPTC-15
R53C	OPTC-16
R53CHK	OPTC-15
R53C1	OPTC-15
R53DISP	OPTC-17
R53JOB	OPTC-15
R54	INFA-13
R55	INFA-14
R55RET	INFA-14
R56	OPTC-17
R60CSM	ATTM-2
R61CSM	ATTM-15
R61TEST	ATTM-3
R62DISP	ATTM-1
R63	ATTM-19
R63COM1	ATTM-20
R64	VBDF-10
R66CSM	ATTM-23
R67	ATTM-23
R67RSTRT	ATTM-24
R67START	ATTM-23
SATSTICK	BOOS-6
SATSTKON	BOOS-5
SBTASK	STER-8
SCALEPOP	ENRY-3
SCALPREP	GENP-23
SELFCHK	TEST-1
SENDID	TELE-4
SEPMIN	NNDF-6
SEPSECNR	NNDF-6
SERVEXIT	GENP-3
SERVICER	GENP-4
SETCOARS	IMUC-13
SETGWLST	PREL-4
SETINTG	MEAS-13
SETISSW	IMUC-8
SETJTAG	DPIR-1
SETMAXDB	DPIR-4
SETMINDB	DPIR-4
SETRE	COOR-9
SETUP.9	STER-4
SETUPT6	DPDS-15
SETVHFLG	VBDF-14
SETWO+2	TELE-2
SFRUTMIX	DATA-19
SHOW	TEST-14
SHOWSUM+2	TEST-4
SICOM	MATH-1

Symbol	Document
SIDE+	DPDS-8
SIDE-	DPDS-9
SLAP1	GENP-8
SLEEPIE	PREL-3
SLEEPIE$_d$	TEST-9
SMCDURES	COOR-4
SMODECHK	TEST-1
SNAPLOOP	TELE-3
SNAPPY	MEAS-18
SOMEERRR	TEST-13
SOMERR2	PREL-2
SOPTION	TEST-5
SPNDXCK	DPDS-5
SPSICOM	MATH-2
SPSOFF	DPIR-5
SQRTTASK	STER-10
SR30.1	DISP-6
STABLISH	DPIR-1
STARTAUT	MINK-2
STARTENT	ENRY-2
STARTEN1	ENRY-3
STARTFIX	MEAS-21
STARTSB2	GENP-13
STARTSUB	GENP-13
STATEINT	ORBI-1
STATINT1	ORBI-1
STICKCHK	DPIR-4
STKTEST	ATTM-18
STOPRATE	ATTM-13
STOR77	MEAS-21
STRTGYRO	IMUC-15
STRTGYR2	IMUC-15
SUFFCHEK	CONC-8
SUPERJOB	DPDS-1
SVCT3	GENP-26
SWICHOVR	DPTV-3
SXTANG	COOR-6
SXTANG1	COOR-6
SXTMARK	OPTC-7
SXTNB	COOR-5
SXTSM	INFA-9
S33/34.1	REND-1
S34/35.2	REND-5
S34/35.3	REND-6
S3435.25	REND-5
S40.1	STER-1
S40.1B	STER-1
S40.13	STER-7
S40.14	DPIR-7

Symbol	Document	Symbol	Document
S40.15	DPIR-7	TRCOMP	DPDS-4
S40.2,3	STER-2	TRG*NBSM	COOR-4
S40.2,3B	STER-3	TRKSTAR	INFA-18
S40.6	DPIR-5	TRMATTCK	VBDF-5
S40.8	STER-5	TSTFORDP	DATA-14
S40.81	BURN-27	TSTLTS3	VBDF-2
S40.9	STER-6	TTG/0	BURN-29
S41.1	STER-8	TVCDAPON	DPTV-1
S41.2	DPIR-7	TVCEXEC	DPTV-4
S50	INFA-7	TVCINIT1	DPTV-1
S52.2	INFA-7	TVCINIT4	DPTV-3
S61.1	ENTP-4	TVCZAP	DPIR-5
S61.1A	ENTP-5	TWOPULSE	IMUC-18
S61.1C	ENTP-5	T4RUPT	GENP-5
S61.2	DISP-13	T5IDLOC	DPIR-1
TACHEXIT	VBDF-7	T5PHASE2	DPRC-7
TAKE2	MEAS-15	T5RUPT	DPIR-1
TARGDRVE	PREL-9	T6LOOP	DPDS-15
TARGETNG	ENRY-3	T6RUPT	DPIR-1
TEMPSET	DPTV-6	T6SETUP	DPRC-24
TERMSXT	OPTC-9	T6START	DPRC-25
TERM52	OPTC-15	UNZ2	IMUC-9
TESTLOOP	ORBI-10	UPCONTRL	ENRY-10
TESTMARK	OPTC-8	UPDATEOK	UPLK-1
TESTNN	DATA-9	UPDATEVG	STER-4
TESTTAU	DPDS-8	UPDATNN	DATA-11
TESTXACT	VBDF-1	UPDATVB	DATA-11
TFFCONIC	DISP-15	UPDTCALL	ATTM-12
TFFELL	DISP-17	UPJOB	UPLK-3
THISJAZZ	MEAS-20	UPOUT	UPLK-5
TICKTEST	DISP-5	UPRUPT	DATA-1
TIG-0	BURN-23	UPSTORE	UPLK-3
TIG-5	BURN-23	UPVERIFY	UPLK-3
TIGAVEG	BURN-22	UTAREA1	ATTM-20
TIGBLNK	BURN-22	UTOPT45	ATTM-21
TIGNOW	BURN-29	VAC5STOR	TELE-5
TIGON	BURN-30	VALMIS	TEST-13
TIMEDIDL	UPLK-5	VARALARM	GENP-25
TIMERAD	CONC-11	VBCOARK	VBDF-3
TIMESTEP	ORBI-12	VBPROC	DATA-29
TIMETASK	ORVN-9	VBRELDSP	DATA-30
TIMETHET	CONC-10	VBRESEQ	DATA-29
TIMEUPDT	DPDS-16	VBRQEXEC	DATA-29
TIMING	DPDS-14	VBRQWAIT	DATA-29
TNONTEST	IMUC-5	VBTERM	DATA-30
TOBALL	ATTM-2	VBTSTLTS	VBDF-1
TORQUE	TEST-13	VBZERO	VBDF-3
TRACKTRM	VBDF-8	VECPOINT	ATTM-4
TRANSPOS	ATTM-7	VERB	DATA-4

Symbol	Document
VERBFAN	VBDF-1
VERB96	VBDF-15
VGCOMP	STER-4
VHFMOD	MEAS-16
VHFREAD	MEAS-6
VHHDOT	BOOS-2
VNFLASH	DINT-6
VNFLASHR	DINT-6
VNRSTART	BURN-16
VN1645	BURN-15
V1N70DSP	ORVN-3
V37	GENP-15
V37XEQ	GENP-22
V57CALL	VBDF-9
V60	VBDF-10
V61	VBDF-10
V62	VBDF-10
V63	VBDF-10
V64PERF	DISP-2
V67	VBDF-11
V67CALL	MEAS-3
V70UPDAT	UPLK-1
V71UPDAT	UPLK-1
V72UPDAT	UPLK-1
V73UPDAT	UPLK-1
V76CALL	VBDF-12
V77CALL	VBDF-12
V82CALL	DISP-3
V82GOFF1	DISP-4
V82GOFLP	DISP-3
V82GON1	DISP-4
V82PERF	VBDF-13
V83CALL	DISP-10
V83PERF	VBDF-13
V85PERF	VBDF-13
V86PERF	VBDF-13
V89CALL	ATTM-1
V89PERF	VBDF-14
V89RECL	ATTM-1
V90PERF	VBDF-14
V97E	BURN-26
V97P	BURN-26
V97T	BURN-26
V99E	BURN-27
V99P	BURN-27
V99T	BURN-27
WAITONE	MEAS-13
WAKEP62	ENTP-4
WAKER	GENP-24
WDAGAIN	DATA-26
WLOOP	CONC-14
WMATRXNG	VBDF-14
WOTNOW	MEAS-18
WUNZMOR	MEAS-16
XCHSLEEP	DINT-12
XTRAP	MEAS-19
YAWDAP	DPTV-8
YAWTIME	DPRC-23
YCOPY	DPTV-9
ZEROJET	DPRC-6
1/CHECK	IMUC-3
1/GYRO	IMUC-3
1/PIPA	IMUC-1
1/WLOOP	CONC-15
1SHOTCHK	DPTV-5
11DSPIN	DATA-27
2BLANK	DATA-8
2INTOUT	NNDF-6
217ALARM	INFA-15
5BLANK	DATA-7
72SHOWX	ORVN-8
8192AUG	IMUC-17