$$Z'' = Z' - Z_R \tag{6}$$

$$X''' = X'' \sin L_{GR} - Z'' \cos L_{GR} \tag{7}$$

$$Y''' = Y'' \tag{8}$$

$$Z''' = Z'' \sin L_{GR} + X'' \cos L_{GR} \tag{9}$$

$$D = \sqrt{(X''')^2 + (Y''')^2 + (Z''')^2} \tag{10}$$

$$E = \arcsin \frac{Z'''}{D} \tag{11}$$

$$A = C_1 + C_2 \arctan \frac{X'''}{Y'''} \tag{12}$$

where C_1 and C_2 depend on the signs of X''' and Y''' as follows:

X'''	Y'''	C_1	C_2
+	+	$\pi/2$	1
-	-	$3\pi/2$	1
-	+	$\pi/2$	-1
+	-	$3\pi/2$	-1

2-258. SUBPROGRAM C76 (EXAUST). EXAUST maintains a log of usable propellants and the time of their exhaustion. The FORTRAN II reference statement is CALL EXAUST.

a. Inputs. The duplexed inputs are as follows:

COMMON TAG	DIMENSION	UNITS	SYMBOL	UNITS
LSEQ	1	Flight stage and substage indicator, normal sequencing		positive integer
FUFB	2	Usable booster fuel remaining	F_{BU}	slugs
FWBG	2	Booster fuel rate	$\dot{F}_B$	slugs/sec
FDLT	2	Length of current integration interval	t_{CIi}	seconds
FTFSP	2	Current time of flight since liftoff	t_f	seconds
FULB	2	Usable booster LOX	L_{BU}	slugs
FWBL	2	Booster LOX flow rate	$\dot{L}_B$	slugs/sec
FULS	2	Usable stage II LOX remaining	L_{SU}	slugs
FWLB	2	Stage II LOX bleed rate	$\dot{L}_{SBL}$	slugs/sec
FPRGL	2	Stage II gas generator bypass LOX, flow rate	$\dot{L}_{ggb}$	slugs/sec
FUFS	2	Usable stage II fuel remaining	F_{SU}	slugs
FPRGG	2	Stage II gas generator bypass fuel flow rate	$\dot{F}_{ggb}$	slugs/sec
FPSGL	2	Stage II gas generator non-bypass LOX flow rate	$\dot{L}_{ggnb}$	slugs/sec
FPRSL	2	Sustainer thrust build-up total LOX consumption	L_{STB}	slugs
FPSGG	2	Stage II gas generator non-bypass fuel flow rate	$\dot{F}_{ggnb}$	slugs/sec

COMMON TAG	DIMENSION	UNITS	SYMBOL	UNITS
FPRSG	2	Sustainer thrust build-up total fuel consumption	F_{STB}	slugs
FPRFL	2	Stage II sustainer LOX flow rate	$\dot{L}_S$	slugs/sec
FPRFG	2	Stage II sustainer fuel flow rate	$\dot{F}_S$	slugs/sec
FPSSL	2	Sustainer thrust decay total LOX consumption	L_{STD}	slugs
FPSSG	2	Sustainer thrust decay total fuel consumption	F_{STD}	slugs
FT54	2	$t_5 - t_4$		seconds
FT74	2	$t_7 - t_4$		seconds
FT75	2	$t_7 - t_5$		seconds
FT98	2	$t_9 - t_8$		seconds
LEVEL	1	Number of intervals per simulation interval		
FDELT	2	Length of output intervals		seconds

b. Outputs. The outputs are newly computed values of usable propellants and the time of their exhaustion as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
FUFB	2	Usable booster fuel remaining	F_{BU}	slugs
FULB	2	Usable booster LOX remaining	L_{BU}	slugs
FTFEB	2	Expected time of booster usable fuel exhaustion from liftoff	t_{BFE}	seconds
FTLEB	2	Expected time of booster usable LOX exhaustion from liftoff	t_{BLE}	seconds

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
FUFS	2	Usable stage II fuel	F_{SU}	slugs
FULS	2	Usable stage II LOX remaining	L_{SU}	slugs
FTFES	2	Expected time of stage II fuel exhaustion from liftoff	t_{FE}	seconds
FTLES	2	Expected time of stage II LOX exhaustion from liftoff	t_{LE}	seconds
SW(1)	1	Turned ØN if booster fuel exhaustion is expected to occur before booster LOX exhaustion		
SW(2)	1	Turned ØN if stage II fuel exhaustion is expected to occur before stage II LOX exhaustion		

c. Program Logic. FD C76

(1) Steps 1-8. A log of usable propellants and the time of their exhaustion is monitored by a flight stage and substage indicator for normal sequencing in LSEQ. At liftoff, if the number of intervals per simulation interval is equal to three, the length D of the current integrated interval is multiplied by three. Prior to booster cutoff, booster LOX is computed and the parameters are determined by the expressions in group one. Otherwise the subprogram proceeds at step 9. If the booster fuel exhaustion is expected to occur before the booster LOX exhaustion, SW(1) is set ØN. Otherwise SW(1) is set ØFF.

(2) Steps 9-21. The following chart gives the conditions and expression group used to compute the stage II LOX parameters.

MISSILE STATE	EXPRESSION GROUP	SUBPROGRAM ACTION
Liftoff	2	SW(2) is set ØFF and subprogram continues at step 25
Prior to booster jettison	3	Subprogram continues at step 22
Prior to sustainer full thrust	4	Subprogram continues at step 22
Sustainer full thrust	5	Subprogram continues at step 22
Sustainer cut-off time	6	Subprogram continues at step 22
Prior to vernier thrust decay	7	Subprogram continues at step 22

In all other cases, the subprogram returns to the user subprogram.

(3) Steps 22-24. If the stage II fuel exhaustion is expected to occur before the stage II LOX exhaustion, SW(2) is set ØN. Otherwise SW(2) is set ØFF and the subprogram proceeds at step 25.

(4) Step 25. CUTIE is stepped by one and control is returned to the user subprogram.

(5) The following chart gives the time sequence and expression group used to compute the new values of usable propellants and the time of exhaustion.

LSEQ	TIME LESS THAN	STAGE OF FLIGHT	EXPRESSION GROUP
1	t_1	Liftoff	1 and 2
2	t_2	Stage II gas generator start	1 and 3
3	t_4	Prior to booster jettison	3
4	t_7	Prior to sustainer full thrust	4
5	t_8	Sustainer full thrust	5
6	t_9	Sustainer cutoff command	6
7	t_{10}	Sustainer thrust decay	7
8	t_{11}	Vernier thrust decay	7
9	t_{12}	Vernier cutoff command	

d. Expressions. Dual computations are performed for all expressions. The alphabetic characters used in the expressions designate the terms defined in the Inputs or Outputs paragraphs.

GROUP 1

$$F_{BU} = F_{BU} + (\dot{F}_B \cdot t_{CI1}) \qquad L_{BU} = L_{BU} + (\dot{L}_B \cdot t_{CI1})$$

$$t_{BFE} = t_f - (F_{BU}/\dot{F}_B) \qquad t_{BLE} = t_f - (L_{BU}/\dot{L}_B)$$

GROUP 2

$$L_{SU} = L_{SU} + (\dot{L}_{SBL} \cdot t_{CI1})$$

$$t_{LE} = t_f - (L_{SU}/\dot{L}_{SBL})$$

GROUP 3

$$L_{SU} = L_{SU} + (L_{ggb} \cdot t_{CD1}) \qquad t_{LE} = t_f - (L_{SU}/\dot{L}_{ggb})$$

$$F_{SU} = F_{SU} + (F_{ggb} \cdot t_{CD1}) \qquad t_{FE} = t_f - (F_{SU}/\dot{F}_{ggb})$$

GROUP 4

$$\text{where } Y = \frac{I\,(t_5 - t_4) + L\,(t_7 - t_5)}{t_7 - t_4} + \frac{M}{t_7 - t_4}\;D$$

$$\text{where } Z = \frac{K\,(t_5 - t_4) + N\,(t_7 - t_5)}{t_7 - t_4} + \frac{A}{t_7 - t_4}\;D$$

$$L_{SU} = L_{SU} + Y \qquad t_{LE} = t_f - t_{CI1}\,(F_{SU}/Y)$$

$$F_{SU} = F_{SU} + Z \qquad t_{FE} = t_f - t_{CI1}\,(F_{SU}/Z)$$

GROUP 5

$$L_{SU} = L_{SU} + (\dot{L}_{ggnb} + \dot{L}_S)\,t_{CI1}$$

$$F_{SU} = F_{SU} + (\dot{F}_{gnnb} + \dot{F}_S)\,t_{CI1}$$

$$t_{LE} = t_f - \left[L_{SU}/(\dot{L}_{ggnb} + L_S)\right]$$

$$t_{FE} = t_f - \left[F_{SU}/(\dot{F}_{ggnb} + \dot{F}_S)\right]$$

GROUP 6

$$L_{SU} = L_{SU} + \left[\dot{L}_{ggb} + \frac{L_{STD}}{t_9 - t_8}\right] t_{CI1}$$

$$F_{SU} = F_{SU} + \left[F_{ggb} + \frac{F_{STD}}{t_9 - t_8}\right] t_{CI1}$$

$$t_{LE} = t_f - \frac{F_{SU}}{L_{STD}\,(t_9 - t_8) + \dot{L}_{ggb}}$$

$$t_{FE} = t_f - \frac{F_{SU}}{\left[F_{STD}/(t_9 - t_8)\right] + \dot{F}_{ggb}}$$

GROUP 7

$$L_{SU} = L_{SU} + \dot{L}_{ggnb} \; t_{CI1}$$

$$F_{SU} = F_{SU} + \dot{F}_{ggnb} \; t_{CI1}$$

$$t_{LE} = t_f - \frac{L_{SU}}{\dot{L}_{ggnb}}$$

$$t_{FE} = t_f - \frac{F_{SU}}{\dot{F}_{ggnb}}$$

2-259. SUBPROGRAM P10 (GGDSIM). GGDSIM produces cyclic steering orders and discrete commands for use by the flight simulator. GGDSIM is used as the guidance portion of closed loop flight simulation and is called every cycle during powered flight by CLØØP. The FORTRAN II reference statement is CALL GGDSIM.

a. Inputs. For each operational cycle DASIM supplies the ground guidance simulator with missile coordinates expressed in range D, elevation E, and azimuth A. Other inputs include the constants T, G, M, and S. The system S constants are used by all subprograms and are a permanent part of OTC. The T, G, and M constants involve data sensitive to target-launch site locations and are set up for every OTC run. The phase counters (P, P_2, P_3), the subphase counter M, the phase sequencer PFLAG, and the coast flag also are supplied to GGDSIM. These inputs are summarized as follows:

COMMON TAG	ITEM
XD(1), XD(2) XD(3), XD(4) XD(5), XD(6) XD(7), XD(8)	D_1^k (i = 1..4) four values of range
XE(1), XE(2) XE(3), XE(4) XE(5), XE(6) XE(7), XE(8)	E_1^k (i = 1..4) four values of elevation
XA(1), XA(2) XA(3), XA(4) XA(5), XA(6) XA(7), XA(8)	A_1^k (i = 1..4) four values of azimuth
SW(42)	Coast flag

COMMON TAG	ITEM
NFLAG(2)	M
NFLAG(8)	PFLAG
NFLAG(10)	P
NFLAG(12)	P_2
NFLAG(14)	P_3
XS(1)-XS(308)	System constants S_1-S_{154}
XG(1)-XG(24)	Ground guidance site constants G_1-G_{12}
XM(1)-XM(66)	Missile (pitch table and impulse adjustment) constants M_1-M_{33}
XT(1)-XT(30)	Target constants T_1-T_{15}

b. Outputs. GGDSIM generates yaw, pitch, roll, abort, prearm delay, and time-to-go commands depending on the outcome of the current cycle computations. These outputs are summarized as follows:

COMMON TAG	ITEM	SYMBOL
XSTØR(1), XSTØR(2)	1st pitch channel	$\dot{\theta}_A$
XSTØR(3), XSTØR(4)	2nd pitch channel	$\dot{\theta}_B$
XSTØR(5), XSTØR(6)	3rd pitch channel	$\dot{\theta}_C$
XSTØR(7), XSTØR(8)	4th pitch channel	$\dot{\theta}_D$
XSTØR(9), XSTØR(10)	1st yaw channel	$\dot{\Psi}_A$
XSTØR(11), XSTØR(12)	2nd yaw channel	$\dot{\Psi}_B$
XSTØR(13), XSTØR(14)	3rd yaw channel	$\dot{\Psi}_C$
XSTØR(15), XSTØR(16)	4th yaw channel	$\dot{\Psi}_D$
XDEW(657), XDEW(658)	Time-to-go to cutoff	t_g

COMMON TAG	ITEM	SYMBOL
NFLAG(10)	1st value of P	P
NFLAG(12)	2nd value of P	P_2
NFLAG(14)	3rd value of P	P_3
SW(45)	Prearm delay flag	
SW(46)	Abort flag	
XDBIT	XDBIT(1)-XDBIT(13)	
NFLAG(3), NFLAG(4)	Substage cycle counter	q
NFLAG(5), NFLAG(6)	Cycle counter	K
NFLAG(1), NFLAG(2)	Sub-P phase cycle counter	M
XDEW(709), XDEW(710)	Net pitch	$\theta_A^{k-1} + \dot{\theta}_2^k$
XDEW(711), XDEW(712)	Net yaw attitude deviation	$\Psi_A^k + \dot{\Psi}_2^k$
XDEW(585), XDEW(586)	Slant range acceleration	$\ddot{D}$

c. Program Logic. FD P10

(1) Steps 1-22. SW(60) is set ØFF. P is stepped by PFLAG and P_2 and P_3 are set equal to P.

P VALUE	ACTION
P = 40	Exit to ILLCMD to print error statement.
P changed	M is set to one to indicate the first pass through GGDSIM for that phase.
P unchanged	M is stepped by one.
P = 0	PFLAG is set to four. The XDEW area is initialized. STUP1, STUP2, and STUP3 perform setup computations. NP is set to one. XDEW(714) and XDEW(728) are set to three and the subprogram continues at step 245.

$P \neq 0$	PFLAG is cleared and CYCLE sets up certain registers for this pass through GGDSIM.
$P = 7$	If $M = 5$, current time of flight is stored for future use. Otherwise, the subprogram continues at step 21.
$P = 6$ or 10	If $M = 1$, counter q is cleared and the subprogram continues at step 23. Otherwise, the subprogram goes directly to step 23.

(2) Steps 23-38.

P VALUE	ACTION
$P < 38$	XDBIT(3) is set to one.
$P = 2, 3, 38,$ or 39	Counters t and t_s are cleared.
$3 < P < 38$	Counter t is stepped by one.
$P = 10$ or 16 or $22 < P < 30$	Counter t_s is cleared.
$P \neq 10$ or 16 and $3 < P < 23$ or $29 < P < 38$	Counter t_s is stepped by one
$6 < P < 38$	TFLYT computes current time of flight since liftoff.
$29 < P < 38$	MSDST computes miss distance.

(3) Steps 39-49.

P VALUE	ACTION
$1 < P < 40$	Radar coast counter is set to minus one. DASMB performs data assembly. CØCØN converts radar data to guidance simulation coordinates.
$P = 38$	PFLAG is set to one and the subprogram continues at step 245.

P = 39	XDBIT(9) and XDBIT(13) are set to one and the subprogram continues at step 245.
1 < P < 38	MSPØS computes current missile position. The current target position, crossplane vector and its magnitude, rotational velocity, and gravitational acceleration are computed.

(4) Steps 50-73.

P VALUE	ACTION
P > 5	First and second differences are computed.
22 < P < 30	PFLAG is set to one.
P > 29	If NFLAG(4) is greater than zero, the evaluation phase filters are computed, NFLAG(4) is stepped by one, and the subprogram continues at step 157.
P = 26 or 29	V is cleared and the subprogram continues at step 245.
P = 28	XDBIT(11) is set to one and the subprogram continues at step 245.
P = 27	The subprogram continues at step 245.
P = 25	XDBIT(10) is set to one and the subprogram continues at step 245.
P = 24	IFLAG is set to 1610. The tangential velocity of target aim point and the square of the tangential velocity at any point are computed. INTRØG interrogates SW(45) to determine if the prearm delay flag is set. If ØN, PFLAG is set to six and the subprogram continues at step 245. Otherwise, the subprogram goes directly to step 245.
P = 23	If $\left\|\dot{\Psi}_A\right\| < S_{24}$, V is cleared and the subprogram continues at step 245. Otherwise, SW(45) is set ØN and the subprogram continues at step 245.

(5) Steps 74-83.

P VALUE	ACTION
$5 < P < 23$	If radar coast counter is greater than S_{22}, PFLAG is set to 40 - P and the subprogram continues at step 245. If $t > S_{163}$, the first-order filter common to pre-vernier and vernier phases is computed.
$P = 18$	If $t_s > S_{14}$, PFLAG is set to one and the subprogram continues at step 243. Otherwise, the subprogram goes directly to step 243.

(6) Steps 84-92.

P VALUE	ACTION
$18 < P < 23$	NFLAG(4) is stepped by one and the vernier filter is computed. If $D^{*k} > S_{25}$, XDBIT(10) is set to one. PFLAG is set to 40 - P and the subprogram continues at step 245. If $\tau_c \geq 0$, SW(45) is set ØN and the subprogram continues at step 90. Otherwise, the integrals of the vernier dead-reckoned acceleration are updated and the vernier Z and vernier Y velocities are computed.

(7) Steps 93-113.

P VALUE	ACTION
$P = 22$	The subprogram continues at step 157.
$P = 19$	If $t_s > S_{15}$, PFLAG is set to one and the subprogram continues at step 157. Otherwise, the subprogram goes directly to step 157.
$P < 18$	If $t > S_{163}$, the second-order filter used before sustainer cutoff is computed. Otherwise, the subprogram continues at step 99.

$P < 6$	The subprogram continues at step 114.
$P = 6$	The subprogram continues at step 142.
$P = 10$	If $\ddot{Y} < S_8$, SUSIC performs sustainer initialization, PFLAG is set to one, and the subprogram continues at step 142. Otherwise, the subprogram goes directly to step 142.
$P < 15$	The subprogram continues at step 121.
$P = 15$	The subprogram continues at step 157.
$P > 15$	PFLAG is set to one.
$P = 16$	VRNIC performs vernier initialization and V is cleared. If $\lvert\dot{\Psi}_A\rvert$ or $\lvert\dot{\theta}_A\rvert > S_{23}$, SW(45) is set ØN and the subprogram continues at step 243. Otherwise, the subprogram goes directly to step 243.
$P = 17$	Vernier pitch-over is computed and the subprogram continues at step 243.

(8) Steps 114-120.

P VALUE	ACTION
$P = 5$	If $t > S_3$, BSTIC performs booster initialization, PFLAG is set to one, and the subprogram continues at step 243. Otherwise, the subprogram goes directly to step 243.
$P = 4$	If $t > S_2$, initial roll is set up, PFLAG is set to one, and the subprogram continues at step 243. Otherwise, the subprogram goes directly to step 243.
$P < 4$	The subprogram continues at step 245.

(9) Step 121.

P VALUE	ACTION
P = 7, 8, 9, 11, 12, 13, 14, 20, or 21	Crosswise velocity errors are computed.

(10) Steps 122-140.

P VALUE	ACTION
P = 7	If $t_s > S_{141}$, the constant attitude wires prior to booster and sustainer steering are initialized. Otherwise, the subprogram continues at step 142. If $t > S_5$, WIRES initializes the constant attitude wire components, PFLAG is set to one, and the subprogram continues at step 142. Otherwise, the subprogram goes directly to step 142.
P = 8	If $t > S_6$, PFLAG is set to one and the subprogram continues at step 142. Otherwise, the subprogram goes directly to step 142.
P = 9	If $t > S_7$, PFLAG is set to one, and the subprogram continues at step 140. Otherwise, the subprogram goes directly to step 140.
P = 11	If $t_s \leq S_{141}$, the subprogram continues at step 142. If $t_s > S_0$, WIRES initializes the constant attitude wire components, PFLAG is set to one, and the subprogram continues at step 142. Otherwise, the subprogram goes directly to step 142.
P = 12	If $t_s > S_{10}$, PFLAG is set to one, and the subprogram continues at step 142. Otherwise, the subprogram goes directly to step 142.
P = 13	STEER computes the steering filters and gain adjustment. The subprogram continues at step 141.

WWW.CHROMEHOOVES.NET

(11) Steps 141-147.

P VALUE	ACTION
P = 6, 7, 8, 9, 11, 12, 13, or 14	The pitch computation is performed. Threshold acceleration and nominal pitch wire are computed. If radar coast counter ≥ 0, or $\ddot{Y}_1 \geq \ddot{Y}_N$, or $t \leq S_{163}$, the subprogram continues at step 154. Otherwise the subprogram continues at step 148.

WWW.CHROMEHOOVES.NET

(12) Steps 148-153.

P VALUE	ACTION
$P = 13$ or $5 < P < 10$	PFLAG is set to 40-P and the subprogram continues at step 245.
$10 < P < 13$	The subprogram continues at step 154.
$P = 14$	XDBIT(10) is set to one, SW(45) is set ØN, and PFLAG is set to 16-P.

(13) Steps 154-157.

P VALUE	ACTION
$P = 6$	If $t > S_4$, PFLAG is set to one, and the subprogram continues at step 157. Otherwise, the subprogram continues at step 243.
$P = 7$	Ballistic equations are computed by BALEQ.

(14) Steps 158-163.

P VALUE	ACTION
$P < 13$	The subprogram continues at step 170.
$P = 13$	If $M > 70$, the subprogram continues at step 164. Otherwise, the subprogram continues at step 170.
$P = 14$	The subprogram continues at step 164.
$14 < P < 20$	XDEW(730), XDEW(732), and XDEW(734) are cleared and the subprogram continues at step 170.
$P = 20$	If $t_s > S_{76}$, the subprogram continues at step 164. Otherwise, the subprogram continues at step 170.
$P = 21$	The subprogram continues at step 164.

(15) Steps 164-171.

P VALUE	ACTION
P = 21	XDEW(730) and XDEW(732) are initialized. If XDEW(730) = 0, the subprogram continues at step 170. Otherwise, the program interpolates for missile speed as a function of current time of flight since liftoff. VECMAG computes the magnitude of the current missile velocity vector. If the interpolated missile speed minus the magnitude of the current velocity vector is greater than 5.0 feet/sec., the subprogram continues at step 259. Otherwise, the subprogram continues at step 170.
$P < 30$	Constant attitude wire components are computed.
P = 30	The subprogram continues at step 212.

(16) Steps 172-195.

P VALUE	ACTION
$P > 18$	The subprogram continues at step 196.
$P < 13$	The subprogram continues at step 243.
$12 < P < 19$	XDBIT(5) is set to one. The estimated vernier acceleration and the velocity magnitude error during sustainer are computed. If radar coast counter $> S_{21}$, SW(45) is set ØN.
P = 13	If $t_s < S_{22}$, PFLAG is set to one and the subprogram continues at step 243.
P = 14, 15, 21, or 22	IFLAG is set to identification integer 1610. INTRØG interrogates SW(64) to determine if time to go is to be computed. If ØN, the time to go is computed.
P = 21	If $t_g < S_{17}$, XDBIT(10) and PFLAG are set to one and the subprogram continues at step 243. Otherwise, the subprogram goes directly to step 243.

P = 14	If $t_g < S_{12}$, the subprogram continues at step 188. Otherwise, the subprogram continues at step 243.
P = 15 or 22	XDBIT(10) is set to one. If $t_g < S_{13}$, register V is set to t_g, PFLAG is set to one, and the subprogram continues at step 243. Otherwise, the subprogram goes directly to step 243.

(17) Steps 196-211.

P VALUE	ACTION
$18 < P < 23$	XDBIT(6) is set to one. The velocity error magnitude during vernier and vernier threshold are computed. If $D^{*}k > S_{25}$, XDBIT(10) is set to one, PFLAG is set to 40 - P, and the subprogram continues at step 243. If $\tau_c \geq 0$, SW(45) is set ØN.
P = 21 or 22	If $\varepsilon_{vt} \geq \varepsilon_v$, the subprogram continues at step 182. Otherwise, XDBIT(10) is set to one, SW(45) is set ØN, PFLAG is set to 23 - P, and the subprogram continues at step 243.
P = 20	If $t_s \leq S_{16}$, the subprogram continues at step 243. Otherwise, if $\varepsilon_v \geq 0$, PFLAG is set to one and the subprogram continues at step 243.

(18) Steps 212-220.

P VALUE	ACTION
P = 30	If $t_s > S_{19}$, PFLAG is set to one and the subprogram continues at step 245. Otherwise, if $\tau_c < 0$ and $D^{*}k \leq S_{25}$, the subprogram continues at step 245. If $\tau_c \geq 0$ and/or $D^{*}k > S_{25}$, IFLAG is set to identification integer 1610. If the prearm delay flag is set (SW(45) = ØN), PFLAG is set to 40 - P; otherwise, PFLAG is set to 38 - P. In either case, the subprogram continues at step 245.

(19) Steps 221-230.

P VALUE	ACTION
P = 31	IFLAG is set to identification integer 1610. If SW(45) is ØFF, PFLAG is set to 37 - P and the subprogram continues at step 245. If SW(45) is ØN, $C_5 < M_{\emptyset} < C_4$, and $C_7 < M_{\lambda}/\cos \emptyset_H < C_6$, PFLAG is set to one, and the subprogram continues at step 245. Otherwise, PFLAG is set to 40 - P and the subprogram continues at step 245.

(20) Steps 231-242.

P VALUE	ACTION
P = 32	XDBIT(10) and PFLAG are set to one. The subprogram continues at step 245.
P = 33	Register V is cleared and PFLAG is set to one.
P = 34	PFLAG is set to one.
P = 35	XDBIT(11) and PFLAG are set to one.
P = 36	Register V is cleared and PFLAG is set to one.
P = 37	If $t_s > S_{20}$, radar coast counter ≥ 0, and/or $C_9 > S_{25}$, PFLAG is set to one.

(21) Steps 243-245.

P VALUE	ACTION
$3 < P < 23$	PRSCØ processes steering orders, SGSEP separates the signals for use by the flight simulator, and SW(60) is set ØN.

(22) Steps 246-247.

P VALUE	ACTION
For all P	YAWCØ computes the correction for initial value of launch azimuth. XV(1) is set equal to XV(2).

(23) Steps 248-263.

P VALUE	ACTION
P = 4	Equivalence is forced in register NP. If M < 4 or NP > 5, CUTIE is stepped by one and control is returned to the user subprogram. Otherwise, XDEW registers are initialized and VECMAG computes the magnitude of the missile position vector. If NP < 5, the subprogram continues at step 262. If NP = 5, the program interpolates for missile position as a function of time of flight since liftoff. VECMAG computes the magnitude of the current missile position vector. If the interpolated value of missile position minus the magnitude of the current missile position vector is greater than 10 feet, IFLAG is set to identification integer 1610, XDEW registers are cleared, RLLBCK effects return to the previous checkpoint, and the subprogram continues at step 262 where NP is stepped by two. If NP > 0, CUTIE is stepped by one and control is returned to the user subprogram.

2-261. SUBPROGRAM G07 (MSDST). MSDST computes miss distance. The FORTRAN II reference statement is CALL MSDST.

a. Inputs. The inputs are as follows:

COMMON TAG	SYMBOL		UNITS
XDEW(38)	R_d^{k-1}	Aged upon entry	$feet^3$
XDEW(2)	a^k		pure no.
XDEW(4)	b^k		$feet^{-1/2}$
XDEW(8)	c^k		pure no.
XDEW(14)	f^k		pure no.
XDEW(18)	h^k		pure no.
XDEW(20)	i^k		pure no.
XDEW(170)	X_M^{k-1}		feet
XDEW(174)	Y_M^{k-1}		feet
XDEW(178)	Z_M^{k-1}		feet
XDEW(598)	$\dot{X}_M^{k-1}$		ft/sec
XDEW(602)	$\dot{Y}_M^{k-1}$		ft/sec
XDEW(606)	$\dot{Z}_M^{k-1}$		ft/sec
XDEW(750)	X_E^k		ft/sec
XDEW(752)	Y_E^k		ft/sec
XDEW(754)	Z_E^k		ft/sec
XC(30)	C_{15}		ft^2/sec^2
XC(32)	C_{16}		rad/sec
XC(34)	C_{17}		rad/sec

COMMON TAG	SYMBOL	UNITS
XC(36)	C_{18}	rad/sec
XC(2)	C_1	feet
XC(96)	C_{48}	pure no.
XC(98)	C_{49}	pure no.
XS(80)	S_{40}	$sec/ft^{3/2}$
XS(86)	S_{43}	naut-mi/ft

b. Outputs. The outputs are as follows:

COMMON TAG	SYMBOL	UNITS
XDEW(28)	j^k	pure no.
XDEW(30)	k^k	naut mi
XDEW(32)	x_ϵ^k	naut mi
XDEW(34)	y_ϵ^k	naut mi
XDEW(36)	z_ϵ^k	naut mi
XDEW(38)	R_d^k	$feet^3$
XDEW(40)	R_d^{k-1}	$feet^3$
XDEW(42)	$(R_d^k)^2$	$feet^6$
XDEW(44)	M_c^k	naut mi
XDEW(46)	M_d^k	naut mi

c. Program Logic. IFLAG is set to identification integer 707 and the value R_d is aged. The miss distance is computed as follows:

$$k^k = \left[(i^k + c^k - f^k)/(b^k)^3\right] S_{40}$$

$$j^k = (h^k - a^k)/(1 - a^k)$$

$$X_\epsilon^k = (j^k X_M^{k-1} + k^k \dot{X}_M^{k-1})$$

$$Y_\epsilon^k = (j^k Y_M^{k-1} + k^k \dot{Y}_M^{k-1})$$

$$Z_\epsilon^k = (j^k Z_M^{k-1} + k^k \dot{Z}_M^{k-1})$$

$$(R_d^k)^2 = C_{16} X_\epsilon^k + C_{17} Y_\epsilon^k + C_{18} Z_\epsilon^k$$

$$\text{CALL SQDEW} \left[C_{15} - (R_d^k)^4, R_d^{k-1}, R_d^k\right]$$

$$M_c^k = S_{43} \left[C_{49} (R_d^k)^2 - C_{48} R_d^k\right]$$

$$M_d^k = S_{43} \left[X_\epsilon^k X_E^k + Y_\epsilon^k Y_E^k + Z_\epsilon^k Z_E^k / C_1 + C_{49}\right]$$

SQDEW performs the square root function in the preceding expressions. CUTIE is stepped by one and control is returned to the user subprogram.

2-262. SUBPROGRAM 012 (MSPØS). MSPØS computes earth-centered missile position. The FORTRAN II reference statement is CALL MSPØS.

a. Inputs. The inputs are as follows:

COMMON TAG	ITEM	UNITS
XDEW(94)	X_1^{k}	feet
XDEW(98)	X_1^{k-2}	feet
XDEW(102)	Z_1^{k}	feet
.	.	.
.	.	.
XDEW(116)	Z_1^{k-7}	feet
XDEW(120)	Y_1^{k}	feet
XDEW(124)	Y_1^{k-2}	feet
XDEW(182)	X_3^{k-1}	feet
XDEW(186)	Y_3^{k-1}	feet
XDEW(190)	Z_3^{k-1}	feet
XDEW(246)	$\ddot{Y}_g^{k-1}$	ft/sec-cy
XDEW(250)	$\ddot{Z}_g^{k-1}$	ft/sec-cy
XDEW(464)	$\dot{Z}_1^{k-1}$	ft/sec
XC(38)	C_{19}	feet
XC(40)	C_{20}	feet
XC(42)	C_{21}	feet
XS(108)	S_{54}	cycles
XS(110)	S_{55}	cycles
XS(112)	S_{56}	seconds
XS(114)	S_{57}	sec/cy
NFLAG(4)	q	cycles

COMMON TAG	ITEM	UNITS
NFLAG(10)	P	pure no.
SW(52)	Switch 52	

b. Outputs. The outputs are as follows:

COMMON TAG	ITEM	UNITS
XDEW(170)	X_M^k	feet
XDEW(174)	Y_M^k	feet
XDEW(178)	Z_M^k	feet
XDEW(182)	X_3^k	feet
XDEW(184)	X_3^{k-1}	feet
XDEW(186)	Y_3^k	feet
XDEW(188)	Y_3^{k-1}	feet
XDEW(190)	Z_3^k	feet
XDEW(192)	Z_3^{k-1}	feet
NFLAG(4)	q	cycles
SW(52)	Switch 52	

c. Program Logic.

(1) IFLAG is set to identification integer 712. The earth-centered missile position is computed depending on the stage of flight.

If $P \leq 15$ (missile prior to vernier)

$$X_M^k = X_1^k + S_{54} (X_1^k - X_1^{k-2}) + C_{19}$$

$$Y_M^k = Y_1^k + S_{54} (Y_1^k - Y_1^{k-2}) + C_{20}$$

$$Z_M^k = Z_1^k + S_{54} (Z_1^k - Z_1^{k-2}) + C_{21}$$

If $16 \leq P \leq 29$ (missile in vernier)

$$X_M^k = X_1^k + S_{55}(X_1^k - X_1^{k-2}) + C_{19}$$

$$Y_M^k = Y_1^k + S_{55}(Y_1^k - Y_1^{k-2}) + C_{20}$$

$$Z_M^k = \frac{1}{8} \sum_{i=0}^{7} Z_1^{k-i} + \dot{Z}_1^{k-1} S_{56} + C_{21}$$

If $P \geq 30$ (missile after vernier) and SW(52) is ØFF, set

SW(52) = ØN

$q = 0$

$X_3 = Y_3 = Z_3 = 0$

Proceed to age X_3, Y_3, Z_3. If SW(52) is ØN, proceed to age X_3, Y_3, Z_3 as follows:

$$X_3^k = X_3^{k-1} + \frac{1}{q+1}(X_1^k - X_3^{k-1})$$

$$Y_3^k = Y_3^{k-1} + \frac{1}{q+1}(Y_1^k - Y_3^{k-1})$$

$$Z_3^k = Z_3^{k-1} + \frac{1}{q+1}(Z_1^k - Z_3^{k-1})$$

The earth-centered missile position coordinates after vernier are computed as follows:

$$X_M^k = X_3^{k-1} + C_{19}$$

$$Y_M^k = Y_3^{k-1} - q^2 \ddot{Y}_g^{k-1} S_{57} + C_{20}$$

$$Z_M^k = Z_3^{k-1} - q^2 \ddot{Z}_g^{k-1} S_{57} + C_{21}$$

(2) ARROR checks the status of indicators. CUTIE is stepped by one and control is returned to the user subprogram.

2-263. SUBPROGRAM D02 (ØLGSIM). ØLGSIM enters from SWAP with SW(58) ØN and produces pitch, yaw, and roll turning rates in quanta per cycle. It is used in open loop flight simulation during the sustainer and vernier phases. The FORTRAN II reference statement is CALL ØLGSIM.

a. Inputs. The inputs are supplied by the XDEW and VPP matrices. Vernier engine cutoff and sustainer engine cutoff are supplied by XDEW(799) and XDEW(800).

b. Outputs. The outputs consist of pitch, yaw, and roll turning rates in quanta per cycle stored in XSTØR(1,1)-XSTOR(2,8); time-to-go to sustainer cutoff and vernier cutoff; and NFLAG(10) set to 6 for end of roll commands, 15 for sustainer engine cutoff, 22 for vernier engine cutoff, and 40 for error.

c. Program Logic. FD D02.

(1) Steps 1-28. Using time of flight since liftoff, this sequence of steps computes the current missile velocity, attitude, and gravitational acceleration vectors together with the missile gravity turn.

(2) Steps 29-64. Output matrix XSTØR is cleared. The input table is searched for values which are tested by comparison with time of flight since liftoff. The table search stops when time of flight is less than the value in the table. The remainder of this section determines the value of I to establish what commands are present.

(3) Steps 65-73. Test for roll is performed. The program continues in a sequence dependent on the value of I.

(4) Steps 74-99. Computations are performed for pitch orders.

(5) Steps 100-108. Computations are performed to determine the times to sustainer and vernier engine cutoff.

(6) Steps 109-122. Computations are performed for yaw orders.

(7) Steps 123-136. Computations are performed for roll orders.

d. Expressions.

$$DELTD = \frac{\delta}{t_2 - t_1} \qquad (1)$$

$$DLZ = Z \qquad (2)$$

$$= PLWR - XDEW(8) \cdot (XDEW(9) - XDEW(5)$$

$$= A_L - (\text{Roll turning rate})\ (t)$$

$$SC16 = -\omega_o \cos \phi_R \sin A_L \cdot \frac{180}{\pi} \qquad (3)$$

$$DLTAVA = \frac{XM(1,34) + (V_{a2} - V_{a1}) \cdot \frac{0.390412}{FLAMB}}{V_{a2}} \qquad (4)$$

$$VDR = FVA \cdot DLTAVA \qquad (5)$$

$$ROOT3 = (X^2 + Y^2 + Z^2)^3 \qquad (6)$$

where X, Y, and Z are components of missile position vector.

$$RSI = r_{s1} \qquad (7)$$

$$= r_1 \cos \theta - Y_1 \sin \theta$$

$$RSJ = r_{sj} \tag{8}$$
$$= r_j \cos \theta - Y_j \sin \theta$$

$$RSK = r_{sk} \tag{9}$$
$$= r_k \cos \theta - Y_k \sin \theta$$

$$YSI = Y_{si} \tag{10}$$
$$= Y_i \cos \theta + r_i \sin \theta$$

$$YSJ = Y_{sj} \tag{11}$$
$$= Y_j \cos \theta + r_j \sin \theta$$

$$YSK = Y_{sk} \tag{12}$$
$$= Y_k \cos \theta + r_k \sin \theta$$

$$GP = \omega_p \tag{13}$$
$$= \frac{k_o (XY_{si} + YY_{si} + ZZ_{si})}{(X^2 + Y^2 + Z^2)^3 (VDR)} \frac{180}{\pi} - SC16$$

WWW.CHROMEHOOVES.NET

$$\omega_p^{\ k} = \frac{3}{2} \omega_p^{\ k} - \frac{1}{2} \omega_p^{\ k-1}$$

$$\omega_p^{\ k-1} = \omega_p^{\ k} \text{ at } t = t_1$$

$$B = (D - |XDEW(K)|)/FLAMB(1) \tag{14}$$
$$AA = (D - |XDEW(K+4)|)/FLAMB(1) \tag{15}$$
$$BB = (D - |XDEW(K+8)|)/FLAMB(1) \tag{16}$$
$$AA = (|XDEW(K+8)| - |XDEW(K+4)|)/FLAMB(1) \tag{17}$$
$$B = (|XDEW(K+4)| - |XDEW(K)|)/FLAMB(1) \tag{18}$$
$$A = (|XDEW(K)| - TF)/FLAMB(1) \tag{19}$$

WWW.CHROMEHOOVES.NET