b. GOE CARD OF TARGET HAS NO SHAPE CODING PUNCHES, CHECK IF THIS IS A NEW TARGET

c. <u>Program Logic</u>. FD U61

(1) Steps 1-3. The contents of the index registers are saved and SW(70) is set ØFF. Index register 1 is initialized to read four cards from the card reader and the subprogram continues at step 4.

(2) Steps 4-7. The card reader is selected and the first (next) card is read. If there is no card in the card reader when the read select instruction is executed, an end-of-file occurs with the end-of-file indicator ON. An 11 is stored in ITYER. IFLAG is set to identification integer 2161, SW(70) is set ØN, and the subprogram continues at step 8. Otherwise the subprogram continues at step 9.

(3) Step 8. The contents of the index registers are restored and the subprogram returns to the user subprogram.

(4) Steps 9-12. A P-bit test is performed to determine if column 17 row 9 is punched. If the test fails, the subprogram continues at step 13. Otherwise the punch is a Blue punch, and a P-bit test is performed to determine if column 48 row 9 is punched. If the test fails, a Blue card (first target card) is established, and the card is stored in the proper card image area. Otherwise a Red card (fourth target card) is established, and the card is stored in the proper card image area. Thus the first target card is established

by a punch in column 17 row 9, and the fourth target card is established by a punch in both columns 17 and 48 row 9. The subprogram continues at step 15.

(5) Steps 13-14. A P-bit test is performed to determine if column 48 row 9 is punched. If the test fails, a White card (second target card) is established, and the card is stored in the proper card image area. Otherwise, a Yellow card (third target card) is established and the card is stored in the proper card image area. Thus the third target card is established by a punch in column 48 row 9, and the second target card is established by the absence of punches in both columns 17 and 48 row 9. The subprogram continues at step 15.

(6) Steps 15-26. If all four target cards have not been read (index register 1 not equal to one) the contents of index register 1 are reduced by one and the subprogram continues at step 4. Otherwise the duplexed fuzing parameters are taken from the card and saved. The tens digit and units digit of the fuzing parameters are converted and masked into IRFSG. Targets 1, 4, 7, and 10 are tested for shape coding punches. A P-bit test is used to determine that these shape coding punches exist for columns 0, 1, 6, and 7 rows 1, 1, 9, and 1 respectively. If any one punch does not exist for any target, the appropriate switch pertaining to the particular target is set ØN to indicate the lack of a shape coding punch. If all punches exist, the

appropriate target switch is set ØFF. Targets 2, 5, and 8 are tested for shape coding punches. A P-bit test is performed to determine if shape coding punches exist for columns 0, 1, and 6 rows 19, 19, and 40 respectively. If any one punch does not exist for any one target, the appropriate target switch is set ØN to indicate the lack of a shape coding punch. If all punches exist, the appropriate target switch is set ØFF. Targets 3, 6, and 9 are tested for shape coding punches. A P-bit test is performed to determine if shape coding punches exist for columns 0, 1, and 6 rows 50, 50, and 58 respectively. If any one punch does not exist for any one target, the appropriate target switch is set ØN to indicate the lack of a shape coding punch. If all punches exist, the appropriate target switch is set ØFF. The subprogram continues at step 23.

(7) Steps 28-31. If the target cards do not have shape coding punches, the subprogram continues at step 33. If the target is a dummy target (SW(91), SW(92),..., or SW(100) = ØFF) or if the target card is not to be punched (SW(161), SW(162),..., or SW(170) = ØFF), U08 prints and writes statement a. The subprogram continues at step 32.

(8) Step 32. If all target cards have been checked for legaility or absence of shape coding punches, control is transferred to step 8. Otherwise the subprogram continues at step 28.

(9) Steps 33-35. This area is entered if a target

card does not contain shape coding punches. If the target is a dummy target (SW(91), SW(92),.., or SW(100) = ØFF), the subprogram continues at step 32. If the target card is to be punched (SW(161), SW(162),.., or SW(170) = ØN), U08 prints and writes statement b. The subprogram continues at step 32.

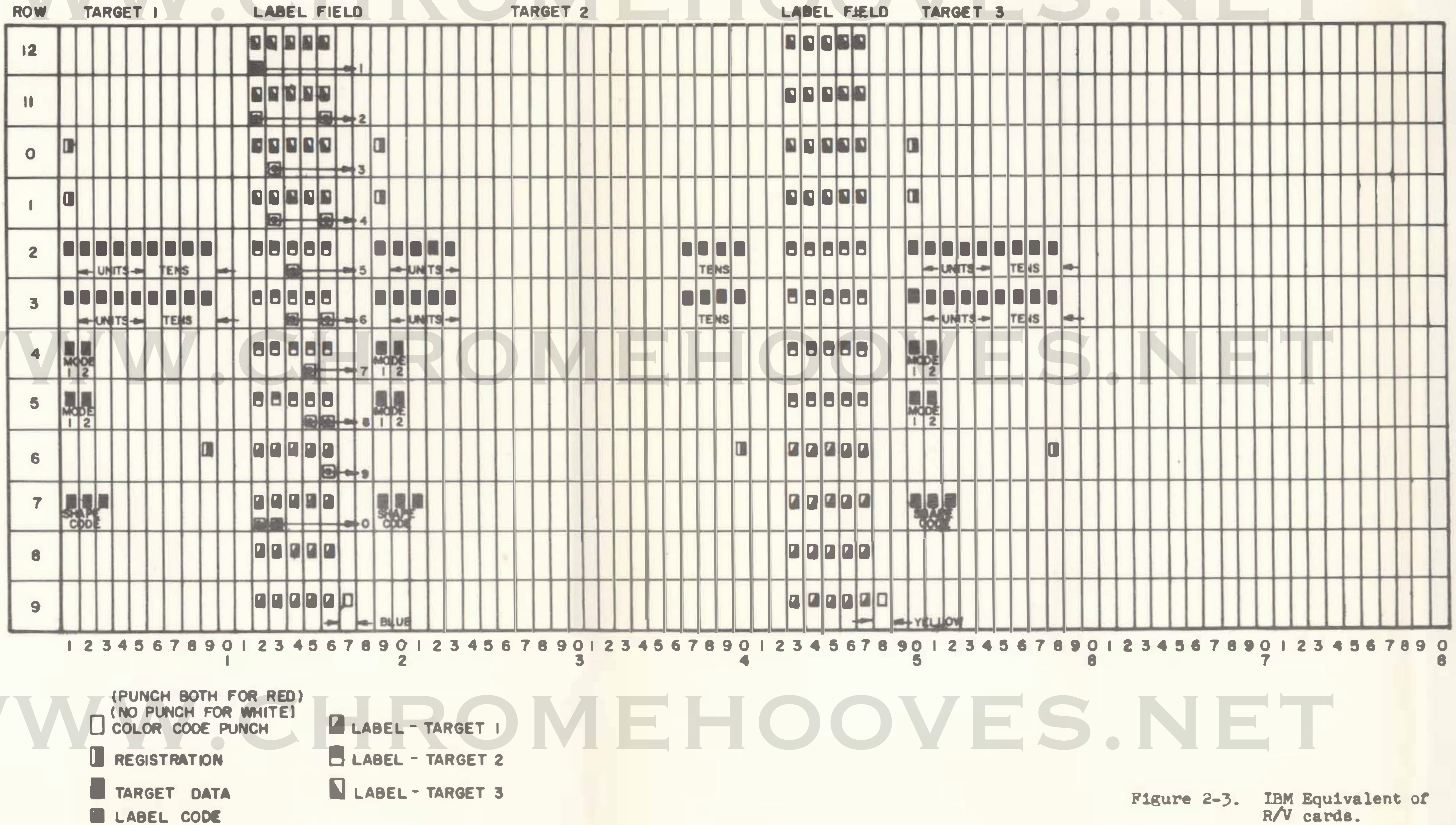

Figure 2-3. IBM Equivalent of R/V cards.

2-167. SUBPROGRAM P56 (SETUP). SETUP sets up the desired detonation point parameters pertaining to the current target slot number. The FORTRAN II reference statement is CALL SETUP.

a. Inputs. The inputs are as follows:

COMMON TAG	DIMENSION	ITEM	UNITS
TØLN	2,10	Geographic longitude of target	degrees
TØLT	2,10	Geographic latitude of target	degrees
TØGS	2,10	Geoidal separation of target	feet
TØAL	2,10	Target altitude above mean sea level	feet
TØDA	2,10	Detonation altitude above target	feet
TØID	10,1	Target data inventory number	BCD
TØGZ	10,1	Desired ground zero - numeric or alphabetic	BCD

b. Outputs. The outputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
TWTDI	1	Current target data inventory code		BCD
TWDGZ	1	Current target desired ground zero		BCD
TWLN	2,10	Geographic longitude of target		degrees

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
TWGLT	2	Current target geographic latitude north of equator		
TWGS	2	Geoidal separation at target for the current target	G_{STC}	feet
TWAL	2	Current target altitude above geoid	h_G	feet
TWDA	2	Current target desired detonation altitude above target	h_{DD}	feet
TWER	2	Current target ellipsoid radius		feet
TWCLT	2	Current target geocentric latitude		degrees
TWDRV	2	Current target distance to center of earth	r_T	feet
TWCCL	2	Current target cosine of geocentric latitude		
TWSCL	2	Current target sine of geocentric latitude		

WWW.CHROMEHOOVES.NET

c. Program Logic. FD P56. Steps 1-8. The current target desired ground zero and the target data inventory code are set to values pertaining to the current target slot number. The current target parameters are set up and the inputs to LGTØLC are initialized. LGTØLC computes the geocentric latitude north of the equator as a function of the geographic latitude. ELLRAD computes the earth ellipsoid radius r_e as a function of the geocentric latitude north of the equator. The current value of the ellipsoid

WWW.CHROMEHOOVES.NET

radius and geocentric latitude are set to the computed values. Expression (1) evaluates the current target distance to the center of the earth and SINE computes the sine of the geocentric latitude. COSINE computes the cosine of the geocentric latitude and the target sine and cosine of the geocentric latitude are set up. The subprogram returns to the user subprogram.

d. Expression.

$$r_T = (r_e + G_{STC}) + (h_G + h_{DD}) \qquad (1)$$

2-168. SUBPROGRAM P06 (TAIMPT). TAIMPT determines the difference in altitude between the specified detonation point and the desired target aim point along an ideal-earth ballistic trajectory. The FORTRAN II reference statement is CALL TAIMPT.

a. Inputs. The inputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
GCRNG	2	Cosine of inertial range to target	cos ϕ'	radians
GRESM	2,8	Re-entry angle(1,8)	γ	
GSRNG	2	Sine of inertial range to target	sin ϕ'	radians
GCBRG	2	Cosine of inertial target bearing	cos B_T	radians
PLCCL	2	Launch pad cosine of geocentric latitude	cos L_{CL}	pure no.
PLSCL	2	Sine of launch pad geocentric latitude	sin L_{CL}	radians
GRAVA	2	Gravitational parameter (=2.092569 x 10^7)		feet
GRAVJ	2	Gravitational parameter (=1.6234633 x 10^{-3})		feet

b. Outputs. The output is the altitude of the target aim point adjustment due to frequency. The output is duplexed, single precision, and stored in GTPLT. GTPLT is expressed in feet.

c. Program Logic. FD P06

(1) Steps 1-6. Expression (1) is evaluated for use in further computation. The re-entry angle is converted to radians and the tangent is computed. Expressions (2), (3), and (4) are evaluated.

(2) Steps 7-10. The altitude of the target aim point adjustment is computed by expression (5). TRAIL computes the focusing height h_f and uses this value to compute the altitude adjustment of the target by use of expression (6). CUTIE is stepped by one and TAIMPT returns to the user subprogram.

d. Expressions.

$$F = 1 - \cos \phi' \tag{1}$$

$$Q = 0.5 + \left[(\sin B_T \tan \gamma)/F\right]^2 \tag{2}$$

$$B = \cos B_T \cos L_{CL} \tag{3}$$

$$P = (2/3)\ 2\ BBF - ((\sin L_{CL})F - B \sin \phi')^2 \tag{4}$$

$$\delta y = AB\left[F\ (1 - 3 \sin L_{CL} \sin L_{CL}) - PQ\right] \tag{5}$$

$$\delta y = \delta y + h_f \tag{6}$$

mod 61

~~mod 7-2~~

2-169. SUBPROGRAM P40. (TARGET). TARGET performs the offset target computation for one target. The FORTRAN II reference statement is CALL TARGET.

a. Inputs. The inputs are as follows:

COMMON TAG	DIMENSION	ITEM	UNITS
GØMGA	2	Rate of earth rotation (= 7.2921158 x 10^{-5})	rad/sec
FRTØD	2	Conversion constant - radians to degrees ($180/\pi$ = 57.295780)	deg/rad
FINIT	2,2	Largest positive floating point number expressible in memory	
GFLRH	2	Guidance radar pulse time unit (=0.0097603)	seconds
PRWLN	6	Geographic longitudes of radars west of Greenwich	degrees
PLWR	2	Reference azimuth bearing of launch pads	degrees
UTDIN	10,1	Target data inventory number	integer
IDGZP	10,1	Desired ground zero point	integer
XDEW(702)	1	Launch azimuth adjustment factor due to yaw steering	
XDEW(496)	1	Time of flight at start of steering	
XDEW(510)	1	$\dot{\epsilon}_{cB}$	
XDEW(512)	1	$\dot{\epsilon}_{cB}$	
GØMGA	2	Rate of earth rotation (= 7.2921158 x 10^{-5})	

SW(82) is set ØN if direct print is requested for OTC and TOT.

b. Outputs. The outputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
FRFSG	2,6,10	Fuzing parameter quantization adjustment factors on aim points		sec/ft
RSLTS	I,1,ISLØT	Target aim point radius R_T	T1	feet
RSLTS	I,2,ISLØT	Target aim point geocentric latitude Φ_T'	T2	degrees
RSLTS	I,3,ISLØT	Target aim point inertial longitude $\lambda_T + \omega t_f$	T3	degrees
RSLTS	I,4,ISLØT	Maximum M_D prearm boundary limit	T4	naut mi
RSLTS	I,5,ISLØT	Minimum M_D prearm boundary limit	T5	naut mi
RSLTS	I,6,ISLØT	Maximum M_C prearm boundary limit	T6	naut mi
RSLTS	I,7,ISLØT	Minimum M_C prearm boundary limit	T7	naut mi
RSLTS	I,8,ISLØT	Δ Roll (clockwise)	T8	quanta
RSLTS	I,9,ISLØT	Kick angle	T9	degrees
RSLTS	I,10,ISLØT	Total time of flight from start of steering t_{fo}	T10	seconds
RSLTS	I,11,ISLØT	A_o	T11	degrees
RSLTS	I,12,ISLØT	E_o	T12	degrees
RSLTS	I,13,ISLØT	$\dot{\epsilon}_{cS}$	T13	ft/sec
RSLTS	I,14,ISLØT	$\dot{\epsilon}_{cS}$	T14	ft/sec
RSLTS	I,17,ISLØT	Target data inventory number	T17	integer
RSLTS	I,18,ISLØT	Designated ground zero code	T18	numeric

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
GBIPC	I,1,ISLØT	Geographic latitude of nominal booster shell impact points for all 10 targets	L_{GBI}	degrees
GBIPC	I,2,ISLØT	Geographic longitude of nominal booster shell impact points for all 10 targets	λ_{BI}	degrees
RDRUM	330	Guidance computer drum slot addresses and scaling factors to appear on output tape		addresses
RSUP	40	Supplementary data temporary storage		
XT	I,1,ISLØT	Target aim point radius R_T	T1	feet
XT	I,2,ISLØT	Target aim point geocentric latitude ϕ_T	T2	degrees
XT	I,3,ISLØT	Target aim point inertial longitude $\lambda_T + \omega t_f$	T3	degrees
XT	I,4,ISLØT	Maximum MD prearm boundary limit	T4	naut mi
XT	I,5,ISLØT	Minimum MD prearm boundary limit	T5	naut mi
XT	I,6,ISLØT	Maximum MC prearm boundary limit	T6	naut mi
XT	I,7,ISLØT	Minimum MC prearm boundary limit	T7	naut mi
XT	I,8,ISLØT	Δ Roll (clockwise)	T8	quanta
XT	I,9,ISLØT	Kick angle δ	T9	degrees
XT	I,10,ISLØT	Total time of flight from start of steering t_{fo}	T10	seconds
XT	I,11,ISLØT	A_o	T11	degrees
XT	I,12,ISLØT	E_o	T12	degrees

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
XT	I,13,ISLØT	$\epsilon_{\dot{c}R}$	T13	ft/sec
XT	I,14,ISLØT	$\epsilon_{\dot{c}S}$	T14	ft/sec
XT	I,17,ISLØT	Target data inventory number	T17	integer
XT	I,19,ISLØT	Target flag	T19	

The following statements are also outputs:

a. TARGET_____.

b. THIRD PASS WAS NECESSARY. ISLOT = _____.

c. EXCESSIVE MISS DISTANCE AFTER THIRD PASS. ISLOT = ___.

d. ADJUSTMENTS DUE TO FUSING QUANTIZATION FOR TARGET ALTITUDE ADJUSTMENT = _____ FEET (+ BELOW TARGET). TIME ADJUSTMENT = _____ SECONDS.

e. TARGET _____ LAT. DEG. _____ LON. DEG. _____ RANGE MILES _____ BEARING DEG.

f. AIM POINT _____ LAT. DEG. _____ LON. DEG. _____ LAUNCH AZIMUTH DEG. _____.

g. OFFSET ADJUSTMENT _____ DOWN RANGE FT _____ CROSS RANGE FT.

h. T CONSTANTS FOR FIRST PASS _____.

i. END POINT _____ LAT. DEG. _____ LON. DEG. _____ TOTAL MISS DISTANCE FT.

j. THE AIM POINT HAS BEEN ADJUSTED FOR OPTIMUM TRAJECTORY FOCUSING. THE ALTITUDE ADJUSTMENT WAS _____ FEET.

k. HERGET SOLUTION DIFFERENCES _____ LAT. DEG. _____ LON. DEG. _____.

l. T CONSTANTS FOR SECOND PASS _____.

m. GROUND BURST NO FUSING PARAMETERS COMPUTED.

n. TARGETING FOR TARGET _____ HAS BEEN COMPLETED.

o. T1____ T2____ T3____ T8____ T9____ T10____
T11____ T12____

p. FUSING PARAMETER = ____

c. Program Logic. FD P40

(1) Steps 1-7. SAVE4 establishes the subprogram return path. Targeting pass counter KØUNT is set to zero. The output page title (statement a) is printed on-line and written on tape A7. SETUP performs setup operation for the particular target. PREARM computes the target prearm boundaries. If a constraint was exceeded control is transferred to step 119. TRGTRB determines the inertial target range and bearing based on zero time of flight. TRJPAR computes trajectory parameters, including an estimate of the total time of flight, consistent with the estimated inertial target range. The input azimuth is set equal to the estimated inertial target bearing computed in step 6.

(2) Steps 8-20. TRGTRB computes a new estimate of the inertial target range and bearing based on a new estimate of time of flight; LAZMTH computes an estimated launch azimuth based on a new estimated target bearing; and TRJPAR computes trajectory parameters, including time of flight, based on new estimates of the inertial target range and launch azimuth. If the two latest estimates of time of flight do not agree within 25 seconds, control is returned to step 8. Otherwise, out of complex reference azimuth is adjusted if necessary. CNSTRN computes and tests the inertial launch azimuth for

exceeded constraint. If the inertial launch azimuth constraint was exceeded by more than one degree, the subprogram continues at step 119. Otherwise ØFFSET computes the estimated down range and cross range corrections for oblateness, and down range correction for atmosphere. HDTTAP computes the first estimate of target aim point using two values each of the negative of the cross range correction, a zero time of flight, and the geocentric latitude and longitude of the target. The target constants are set to the following values:

T1	Distance of target aim point from center of earth
T2	Geocentric latitude of estimated target aim point
T3	Difference between the longitude of the radar and longitude of estimated target aim point with the radar longitude adjusted for total estimated time of flight less 90 seconds
T8	Current estimate of launch azimuth times one twelfth of the guidance radar pulse unit
T9	Kick angle
T10	Current estimate of total time of flight minus 90 seconds
T11	Current value of A_o which is less than 180
T12	Current value of E_o
T19	Zero

Constant T3 is adjusted to be within the limits $-180 \leq T3 \leq 180$ by adding or subtracting 360, if necessary. The current value of A_o is adjusted so that $|A_o| \leq |180°|$.

(3) Steps 21-35. The matrix is cleared of missile parameter sigma levels. Switches used in the program are

initialized. INITAL initializes data tables for flight simulation. Statements e, f, g, h, and o are written on tape A7. SWAP controls the time sharing of subprograms in core and performs a complete simulation. If a constraint was exceeded, the subprogram continues at step 119. Otherwise, targeting pass counter KØUNT is increased by one. INGAIN sets up next address in AGAIN. CKPTCK establishes a check point and MISDIS computes miss distance. A_0 is tested for overflow. If overflow occurred, values of geocentric target aim point latitude are set to zero and TAIMPT is skipped; otherwise, TAIMPT computes the difference in altitude between the specified detonation point and the desired target aim point. The re-entry vehicle separation parameters are set up for solution of the Herget equations and SW(69) is set ØFF.

(4) Steps 36-42. Herget computes the target aim point altitude, and the geographic latitude and longitude in the inertial coordinate system. The change in the target aim point geocentric latitude and longitude west of Greenwich, ΔL_A and ΔL_G, as affected by the Herget solution, is computed. HERGET computes a new target aim point using the distance of the target from the center of the earth adjusted for focusing by the value obtained in step 33. XYZGEØ converts detonation point position vectors into geocentric latitude, longitude, and altitude of the detonation point above the earth ellipsoid. If the current target lon-

gitude is negative, 360 degrees is added to the difference between the longitude of the current target and the detonation point and the subprogram continues at step 43. Otherwise the difference between the current target longitude and the longitude of the detonation point is computed.

(5) Steps 43-47. The difference between the current target geocentric latitude and the geocentric latitude of the detonation point is computed. The target aim point longitude is corrected for down range offset and the target aim point latitude is corrected for crosswise offset. The launch azimuth is adjusted to compensate for yaw steering. If launch azimuth is less than zero, 360 is added; if greater than or equal to 360, 360 is subtracted.

(6) Steps 48-50. The target constants are set as follows: T1 to the adjusted distance from the center of the earth to the new target aim point altitude, T2 to the geocentric latitude of the new target aim point, T3 to the inertial target longitude computed by adding the time of flight since the start of steering multiplied by the rotation rate of the earth to the difference between the longitude of the new target aim point, T8 to the launch azimuth times one-twelfth of the guidance radar pulse time unit, T10 to the total time of flight since start of steering. T11 to A_o, T12 to E_o after SECO. Constant T3 is adjusted to lie within the limits $-180 \leq T3 \leq 180$ by adding or subtracting 360, if necessary. Constant T11 is tested and, if greater than 180

after SECO, is adjusted by ±360, so that |T11| ≤ |180|.

(7) Steps 51-62. CNSTRN tests for exceeded constraints. The following switches are set ØN to give the action indicated:

Switch No.	Action
9 172	RSDØRE is initialized for data recordings (USRS)
10	RSDØRE is initialized for data recordings of new target
131	CLØØP is called by SWAP
132	ØLØØP is called by SWAP
159	Ballistic simulation is executed
160	Re-entry simulation is executed

INITAL initializes data tables for flight simulation. If the longitude is negative, 360 is added to obtain positive values for the second and third pass printouts. Statements i, j, h, l, m, and o are written. If a constraint was exceeded, control is transferred to step 119. Otherwise SWAP controls the time sharing of subprograms in core and performs a complete simulation. If a constraint was exceeded, control is transferred to step 119. Otherwise the targeting pass counter KØUNT is increased by one. XYZGEØ converts second pass detonation point inertial position vectors into geocentric latitude, longitude, and altitude of detonation point above earth ellipsoid. The second pass detonation point geocentric latitude and altitude above the earth ellipsoid are placed in final targeting pass registers. If current target longitude is negative, it is increased by 360.

(8) Steps 63-67. The difference between the current target longitude and the longitude of the second pass detonation point and the difference between the current target geocentric latitude and geocentric latitude of the second pass detonation point are computed. The new target aim point longitude and latitude are computed. The initial launch azimuth is adjusted to reduce yaw steering. If launch azimuth is less than zero, 360 is added; if greater than or equal to 360, 360 is subtracted.

(9) Steps 68-73. MISDIS computes the second pass miss distance. Statement i is written. If the current miss distance is less than or equal to 1500 feet, the subprogram continues at step 77. If greater than 1500 feet, the number of targeting passes is examined. If three targeting passes have not been made, the subprogram continues at step 74. Otherwise statement c is printed and written. SW(151) is set ØN, and the subprogram continues at step 119.

(10) Steps 74-76. Statement b is printed and written. INGAIN sets up the next address in AGAIN and CKPTCK establishes a check point. Control is transferred to step 48.

(11) Steps 77-88. If a constraint was exceeded in CNSTRN, the subprogram continues at step 119. Otherwise, A_0 is compared to 999.9. If equal, FUSING is entered; otherwise the current target desired detonation altitude is

examined. If zero, the subprogram continues at step 88, where statement m is written. If not zero, FUSING determines the fuzing parameter. Statement d is printed and written; statement p is written. ELLRAD computes the earth ellipsoid radius at target aim point from final geocentric latitude. The following final target aim point information is inserted in input registers for GEØXYZ:

a. Geocentric latitude

b. Longitude west of Greenwich

c. Geoidal separation from earth ellipsoid equal to zero

d. Altitude equal to distance from center of earth to target aim point minus radius of earth at target aim point

e. Total time of flight from liftoff to detonation point

f. Earth ellipsoid radius at target aim point

g. Distance of missile center of gravity above launch pad at liftoff equal to zero

GEØXYZ converts the final target aim point geocentric position into the inertial XYZ coordinate system. These inertial XYZ coordinates are adjusted by the horizontal target aim point adjustment due to the fuzing quantization. XYZGEØ converts the adjusted inertial XYZ coordinates of the final target aim point to geocentric position. The following re-entry deceleration data about this target are saved in the re-entry deceleration table:

a. Detonation point time of error due to fuzing quantization

b. Detonation point altitude adjustment due to fuzing quantization

c. 25g point detection time adjustment factor

d. Horizontal target aim point adjustment due to fuzing quantization

(12) Steps 89-92. The minimum and maximum prearm limits are saved in RSLTS. The usable stage II LOX at liftoff is saved in RDRUM. The target geocentric latitude, the target longitude, and the sine and cosine of the target geocentric latitude are saved so that the range computations for the different points may use these registers as inputs for the required subroutines. The time from liftoff to booster cutoff is saved in RSUP. XYZGEØ converts the inertial position vectors of the booster cutoff point into the geocentric position of the booster cutoff point. The longitude of the booster cutoff point is saved in RSUP. LCTØLG converts the booster cutoff point geocentric latitude to geographic latitude. The geographic latitude of the booster cutoff point is saved in RSUP. GEØXYZ converts the geocentric position of the booster cutoff point into inertial XYZ coordinates. VCDØTP computes the vector dot product and included angle from the inertial coordinates. This output is used in computing the BECO range, which is then converted to miles and saved in RSUP. This method of computation avoids an error due to a target with a zero bearing.

(13) Steps 93-95. XYZGEØ converts the inertial position vectors of the sustainer cutoff point into geocentric position of the sustainer cutoff point. The longitude of the sustainer cutoff is saved in RSUP. LCTØLG converts the sustainer cutoff point geocentric latitude to geographic latitude, which is saved in RSUP. GEØXYZ converts the geocentric position of the sustainer cutoff point into inertial XYZ coordinates. VCDØTP computes the vector dot product and included angle from the inertial coordinates. This output is used in computing the SECO range, which is then converted to miles and saved in RSUP.

(14) Steps 96-99. ARCCØS computes the vernier cutoff look angle. The time from liftoff to vernier cutoff and the look angle at vernier cutoff are saved. XYZGEØ converts the inertial XYZ position vectors of the vernier cutoff point. The longitude of the vernier cutoff is saved in RSUP. LCTØLG converts the vernier cutoff point geocentric latitude to geographic latitude which is saved in RSUP. GEØXYZ converts the geocentric position of the vernier cutoff point into inertial XYZ coordinates. VCDØTP computes the vector dot product and included angle from the inertial coordinates. This output is used in computing the VECO range, which is then converted to miles and saved in RSUP.

(15) Steps 100-101. The time from liftoff to apogee and the apogee altitude are saved. XYZGEØ converts the apogee inertial position vectors into the geocentric posi-

tion of the apogee. GEØXYZ converts the geocentric position of the apogee into inertial XYZ coordinates. VCDØTP computes the vector dot product and included angle from the inertial coordinates. This output is used in computing the apogee range which is then converted to miles and saved in RSUP. The time from liftoff to the re-entry point, the angle of the re-entry velocity vector relative to the earth above the local horizontal, and the magnitude of the re-entry velocity vector are saved in RDRUM.

(16) Steps 102-106. XYZGEØ converts the inertial position vectors of the re-entry point into the geocentric position of the re-entry point. The longitude of the re-entry point is saved in RDRUM. LCTØLG converts the re-entry point geocentric latitude to geographic latitude. The geographic latitude of the re-entry point and the time from liftoff to the detonation point are saved in RDRUM. XYZGEØ converts the inertial position vectors of the detonation point into the geocentric position of the detonation point. The longitude of the detonation point is saved in RDRUM. LCTØLG converts the detonation point geocentric latitude to geographic latitude. The geographic latitude of the detonation point is saved in RDRUM. The total time of flight is set to zero so that TRGTRB will produce an earth-fixed range, and not an inertial range.

(17) Steps 107-109. The detonation point geocentric latitude is converted from degrees to radians. SINE and

CØSINE compute the sine and cosine of the geocentric latitude of the detonation point. RØUND rounds the double-precision outputs of SINE and CØSINE to single precision. TRGTRB computes the earth-fixed angular range and bearing to the detonation point. This range is converted to miles and saved in RSUP. The target geocentric latitude, the target longitude, and the sine and cosine of the target geocentric latitude are restored.

(18) Steps 110-113. INGAIN sets up the next address in AGAIN, and CKPTCK establishes a check point. BØØSHL controls open loop computations for booster shell impact point determination. The booster shell impact point geographic latitude and longitude for this target is saved in the booster shell impact table. The target geocentric latitude, longitude, and the sine and cosine of the target geocentric latitude are saved so that the booster shell impact point information can be inserted in these registers to compute the booster shell impact range. GEØXYZ converts the geocentric position of booster impact to inertial XYZ coordinates. VCDØTP computes the vector dot product and the included angle. This output is used in computing the booster impact range, which is converted to miles and saved in RSUP. The target geocentric latitude, longitude, and sine and cosine of the target geocentric latitude are restored to the target registers.

(19) Steps 114-118. The following information is saved in RSLTS for this target:

T1 = R_T	Target aim point radius
T2 = ϕ'_T	Target aim point geocentric latitude
T3 = $\lambda_T - \lambda_R + \Omega\, t_{fo}$	Target aim point inertial longitude
T8 = A_L	Launch azimuth
T9 = δ	Kick angle
T10 = t_{fo}	Total time of flight from start of closed loop yaw steering
T11 = A_o	Azimuth angle of radar five seconds before VECO
T12 = E_o	Elevation angle of radar five seconds before VECO
T13 = $\bar{\dot{\varepsilon}}_C$	Nominal crosswise velocity error at start of booster yaw steering
T14 = $\bar{\dot{\varepsilon}}_C$	Nominal crosswise velocity error at start of sustainer yaw steering
T17 = TDI	Target data inventory number

During the saving process, T3 and T11 are examined. Both constants are adjusted to lie within the limits $-180 \leq (T3 \text{ or } T11) \leq 180$ by adding or subtracting 360, if necessary. FAPSTR stores the designated ground zero point T18 in RSLTS. Statement n is printed and written.

(20) Step 119. RTRN4 returns the subprogram to the user subprogram.

mod F4

2-170. SUBPROGRAM P42 (VERIFY). VERIFY verifies the target-data generated in OTC. The FORTRAN II reference statement is CALL VERIFY.

a. Inputs. The inputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
RSLTS	2,9,10	Matrix - output data for all ten targets		
IRFSG	2,1,10	Output fuzing parameters for all ten targets		
FRFSG	I,5, ISLØT	Detonation point altitude adjustment due to fuzing quantization		feet
FRFSG	I,6, ISLØT	25g point detection time adjustment factor	τ	seconds
RBIAS	2,3,10	Output aim point bias vectors for all 10 targets (Δlat, Δlon, Δalt)		feet
ØTAPE	2,25,10	RSLTS table image from old paper tape		
TWGS	2	Geoidal separation at target for the current target	G	feet
TWAL	2	Current target altitude above geoid	C	feet
TWDA	2	Current target derived detonation altitude above target	g	feet
TWCCL	2	Current target cosine of geocentric latitude	cos Lg	
TWDRV	2	Current target distance to center of earth	R	feet

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
TWGLT	2	Current target geographic latitude north of equator	L_{GTC}	degrees
GDPSM	2,9	Final detonation point summary data table		
ISLØT	1	Current target slot number		integer
SW(82)	1	If ØN, direct print requested		
FRTØD		Conversion constant: radians to degrees ($180/\pi$ = 57.295780)		
TWLN	2,10	Geographic longitude of target	L_T	degrees
GMSDX	2	Maximum allowable miss distance of final flight		feet

b. Outputs. The outputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
GMLAT	2	Altitude of point above earth ellipsoid	a	feet
GMLØN	2	Longitude of point west of Greenwich	L_G	degrees
GMALT	2	Geocentric latitude of point	L_a	degrees
SW(152)	1	ØN - improper dummy target, or target with excess horizontal miss distance		
XT	2,19	Matrix - constants table		
GLAZM(I)	2	Launch azimuth	A_L	degrees

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
SW(8)	1	ØN - monitor radar slew rate A		
SW(9)	1	ØN - RSDØRE is requested to perform data recordings		
SW(20)	1	ØFF - omit D term in gravity computations		
SW(21)	1	ØFF - omit J term in gravity computations		
SW(32)	1	ØN - CLØØP on re-entry to stop at air burst time		
SW(41)	1	ØN - input MET data to be used for pressure density deviations		
SW(43)	1	ØN - suppress $\dot{\epsilon}_{co}$ maximum initial value gate logic		
SW(44)	1	ØN - suppress yaw steering		
SW(47)	1	ØN - suppress noise		
SW(50)	1	ØN - $\dot{A}$ exceeded maximum antenna slew rate		
SW(54)	1	ØN - booster shell impact point to be determined		
SW(55)	1	ØN - IIP or fuel exhaustion impact point to be determined		
SW(62)	1	ØN - check only inputs of GGDSIM to CLØØP		
SW(64)	1	ØN - do not compute time to go to GGDSIM		
SW(67)	1	ØN - gimbal angle has been excessive at least once		

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
SW(68)	1	ØN - angle of attack angle has been excessive at least once		
SW(131)	1	ØN - SWAP to call CLØØP		
SW(132)	1	ØN - SWAP to call CLØØP		
SW(133)	1	ØN - open loop guidance to be used (TAA control)		
SW(159)	1	ØN - do ballistic simulation		
SW(160)	1	ØN - do re-entry simulation		

The following printed and/or written statements are also outputs:

a. HORIZONTAL MISS DISTANCE EXCESSIVE FOR TARGET_____.

b. TARGET IDENTIFICATION SYMBOLS ON MYLAR TAPE ARE IN ERROR FOR TARGET_______.

c. A CONSTRAINT WAS EXCEEDED DURING THIS SIMULATION OF TARGET_____.

c. Program Logic. FD P42

(1) Steps 1-11. SAVE4 saves the return path of the user subprogram and SW(152) is set ØFF. If there are no duplicate targets, the subprogram continues at step 23. Otherwise IFLAG is set to identification integer 1642. INTRØG interrogates the target switches. If ØN, the subprogram continues at step 15. If ØFF, and if all the T constants, all the output fuzing parameters, the detonation