2-156. B3 SUBPROGRAMS.

2-157. The subprograms described in this area are mainly of a control nature. These subprograms are required during control operations and are specifically concerned with the offset target computations.

2-158. Subprogram LO2 (BENTRY) enables the loader to establish linkage between the B1 and B3 subprograms. This version of BENTRY will be in core only when the OTC and TOT mode of operation is requested. The return path of the user subprogram is saved by SAVE4 and ØTCNT is called. After OTC and TOT have been completed, the subprogram exits to DØCNT through RTRN4.

a.	BENTRY	*LO2	Establish B3 Control Area
b.	HDTTAP	P31	Horizontal Adjustment of Target Aim Point in Down Range and Cross Range
c.	IDCØMP	U31	Compare Target Data Inventory Number and Desired Ground Zero with SAC ID
d.	ØFFSET	P32	Determine Offset Down Range and Cross Range Corrections for Oblateness and Down Range Corrections for Atmosphere
e.	ØTC5	U56	Old Targeting Kit
f.	ØTCNT	P50	Control for Offset Target Computations
g.	PREARM	P23	Adjust Prearm Boundaries
h.	RDTGTP	U66	Read Target Tape ~~Read Mylar Tape Cards~~
i.	RVCDRD	U61	Read in Output Cards from Computation Step of OTC

j.	SETUP	P56	Setup OTC Parameters for this Target
k.	TAIMPT	P06	Determine Attitude Adjustment to be made to Target Aim Point Due to Focusing
l.	TARGET	P40	Compute Offset Target Computations
m.	VERIFY	P42	Verify Targeting Data Generated in OTC

* Subprogram description is in the introductory paragraph of this area.

Mod ~~1~~ F2

2-159. SUBPROGRAM P31 (HDTTAP). HDTTAP adjusts the aim point when the point of impact differs from the desired impact point. The FORTRAN II reference statement is CALL MDTTAP.

a. Inputs. The inputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
FRTØD	2	Conversion constant radians to degrees ($\frac{180}{\pi} = 57.295780$)		deg/rad
PLLØN	2	Geographic lattitude of this launcher	λ_L'	degrees
~~PRWLN~~	~~2~~	~~Longitude of guidance west of Greenwich~~	~~λ_R~~	~~degrees~~
TAPLN	2	Longitude of aim point at time of launch	λ_{TAP}	degrees
TAPLT	2	Latitude of aim point at time of launch	L_{CTAP}	degrees
GDRMS	2	Down range miss distance	IQF	feet
GCWMS	2	Crosswise miss	d_{cwms}	feet
GØMGA	2	Rotational rate of earth	Ω	rad/sec
GTMFL	2	Time of flight	t_{fp}	seconds
PLSCL	2	Radar sine of geocentric latitude	$\sin L_{CL}$	pure no.
PLCCL	2	Radar cosine of geocentric latitude	$\cos L_{CL}$	pure no.
TAPAL	2	Target aim point altitude above real-earth ellipsoid		feet
FHFPI	2	Dual precision $\pi/2$		pure no.
FPI	2	Dual precision π		pure no.
GTRNG	2	Inertial range to target general	$\emptyset'$	degrees

b. Outputs. The outputs are the adjusted aim point parameters as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
TAPLN	2	Longitude of new aim point	X_{TAP}	degrees
TAPLT	2	Latitude of new aim point	L_{CTAP}	degrees
IFLAG	1	Identification integer		

c. Program Logic. FD P31

(1) In the computations of the adjusted target aim point, a conversion constant is introduced wherever conversion is required from degrees to radians or radians to degrees.

(2) Steps 1-6. SINE and CØSINE compute the sine and cosine L_{CTAP}. ELLRAD computes the earth ellipsoid radius at the target aim point. The current target parameters in registers defined as input to TRGTRB are moved to a temporary storage block SAVE. The corresponding aim point parameters are placed in the input registers of TRGTRB. TRGTRB computes the angular range and bearing to the target aim point. The target parameters are then restored.

(3) Steps 7-18. The down range and crosswise miss distances are converted from feet to radians by dividing by the earth radius at the target aim point. Expression (1) determines range P'. SINE and CØSINE compute the sine and cosine P'. SINE is called again to obtain the sine of the

crosswise miss distance. Expression (2) determines the sine of the arc defining the crosswise miss distance. The aim point is adjusted by obtaining the increment of the bearing angle from the arcsine evaluation in ARCSIN. Expression (3) determines the new bearing of aim point γ'. SINE and CØSINE compute the sine and cosine γ'. Expression (4) determines sine L_A', latitude of new aim point. ARCSIN computes the arcsine to obtain L_A'.

(4) Steps 19-28. Expression (5) determines the denominator of expression (6). Constants B, BK1, and BK2 of expression (7) are set to the numbers indicated depending upon the values obtained in testing N and sine γ'.

Sine γ'	N	Action
$\gamma' = 0$ or 180	< 0	B is set to 180
	≥ 0	B is set to 0
$180 < \gamma' < 360$	0	B is set to -90
$0 < \gamma' < 180$	0	B is set to 90

After each of the above four actions, control is transferred to step 35 for evaluation of the new target aim point λ_{TAP}

Sine	N	Action
< 0	< 0	BK1 is set to -180 BK2 is set to 1
> 0	< 0	BK1 is set to +180 BK2 is set to -1
< 0	> 0	BK1 is set to 0 BK2 is set to -1

Sine	N	Action
>0	>0	BK1 is set to 0 BK2 is set to 1

After each of the above four actions, control is transferred to step 33 for evaluation of tangent B. ARCTAN then computes B. If N is positive and sine γ' is negative, BK1 and BK2 are set to 0 and -1; if sine γ' is positive, BK1 and BK2 are set to 0 and 1. At this point expression (6) determines tangent B and ARCTAN computes B.

(5) Steps 29-30. Expression (7) determines the longitude of the new aim point λ_{TAP}. CUTIE is stepped by one and control is returned to the user subprogram.

d. Expressions.

$$P' = \phi' - (IQF/r) \quad (1)$$

$$M = \sin\left[\left({}^{d}cwms/r\right)\right] / \sin P' \quad (2)$$

$$\gamma' = B_T - \text{arcsine } M \quad (3)$$

$$\sin L_A' = \cos P' \sin L_{CL} + \sin P' \cos L_{CL} \cos \gamma' \quad (4)$$

$$N = \cot P' \cos L_{CL} - \sin L_{CL} \cos \gamma' \quad (5)$$

$$\tan B = (1/N)(\sin \gamma') \quad (6)$$

$$\lambda_{TAP} = \Omega t_{fp} + \lambda_R' - B \cdot BK2 + BK1 \quad (7)$$

2-160. SUBPROGRAM U31 (IDCØMP). IDØOMP compares the ID data of the input tape with the ID data on the mylar tape or output data tape in the verification pass through OTC. The FORTRAN II reference statement is CALL IDCØMP.

a. Inputs. The inputs are as follows:

COMMON TAG	DIMENSION	ITEM	UNITS
UTDIN	10,1	Target data inventory number TDI	integer
IDGZP	10,1	Target data desired ground zero DGZ	integer
RSLTS	2,17,10	SAC TDI number for 10 targets	pure no.
RSLTS	2,18,10	SAC DGZ number for 10 targets	pure no.
ISLØT	1	Current target slot number	integer

b. Outputs. The output is the setting of SW(51), which is set ØN if TDI or DGZ data do not agree.

c. Program Logic. FD U31

(1) Steps 1-7. The contents of the index registers are saved and the subprogram is initialized for the target corresponding to the target slot number ISLØT. The target data inventory number is compared with the SAC target data inventory number in the RSLTS table of output data. If both values of the target data inventory number do not agree, SW(51) is set ØN. and the subprogram continues at step 8. Otherwise the desired ground zero point is compared with the SAC desired ground zero

point in the RSLTS table of output data. If both values of the desired ground zero point agree, SW(51) is set ØFF and the subprogram continues at step 8. If both values do not agree, SW(51) is set ØN and the subprogram continues at step 8.

(2) Step 8. The index registers are restored to their original contents and the subprogram returns to the user subprogram.

2-161. SUBPROGRAM P32 (ØFFSET). ØFFSET computes the approximate down range and cross range correction for oblateness, and the approximate down range correction for atmosphere. This aim point offset is computed when the approximate time of flight and inertial range have been determined by table look-up. The FORTRAN II reference statement is CALL ØFFSET.

a. Inputs. The inputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNIT
GK2	2	0.59206995		
GK3	2	0.15142828		
GLAZM	2	Launch azimuth		degrees
TWLN	2	Detonation point longitude west of Greenwich	λ_T	degrees
PRWLN	2	Geographic longitude of guidance radar west of Greenwich	λ_R	degrees
GØMGA	2	Rotation rate of earth ($\approx 7.29211 \times 10^{-5}$)	Ω	rad/sec
GDTTM	2	Total real-earth trajectory time of flight from liftoff to impact		
FRTØD	2	Conversion factor		deg/rad
PRSCL	4	Sine of radar geocentric latitude	$\sin L_{CR}$	
PRCCL	4	Cosine of radar geocentric latitude	$\cos L_{CR}$	radians

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNIT
TWSCL	4	Sine of detonation point geocentric latitude		radians
TWCCL	4	Cosine of detonation point geocentric latitude		radians
GSRNG	4	Sine of inertial range to detonation point		radians
GCRNG	4	Cosine of inertial range to detonation point		radians
GTRNG	2	Inertial range to detonation point	Ø	radians

b. Outputs. The outputs are as follows:

COMMON TAG	DIMENSION	ITEM	UNITS
GCWMS	2	Cross range correction	feet
GDRMS	2	Down range correction	feet

c. Program Logic. FD P32

(1) Steps 1-2. Expression (1) evaluates the relative longitude (radar to impact) in simplexed computation algorithm.

(2) Steps 3-4. SINE computes the sine of the relative longitude. The launch azimuth is converted to radians in simplexed computation algorithm.

(3) Steps 5-8. CØSINE computes the cosine of the launch azimuth. The simplexed values are used in evaluating

expression (2). These results are used in expression (3) to obtain a simplexed value for the cross range. The down range is computed in simplexed algorithm using expression (4).

(4) Step 9. CUTIE is stepped by one and the subprogram returns to the user subprogram.

d. Expressions.

$$C = \lambda_T - \lambda_R + \Omega^\circ t_f \qquad (1)$$

$$B = \cos a \cos d \qquad (2)$$

$$GCWMS = \left[\frac{-2 J_a}{f \sin^2 g} \quad \frac{\sin c \cos L_{CR} \cos b}{\sin \phi}\right] \left[A(1 - \cos \phi) + B(\phi - \sin \phi)\right] \qquad (3)$$

$$GDRMS = \frac{- J_a (1 - \cos \phi)}{f^2 \sin^3 g \cos g} \left\{ 2(1/3 - A^2) + (f \sin^2 g - 2/3) \left[1 - A^2 (\cos \phi + 2) - B^2 (1 - \cos \phi) - 2 AB \sin \phi\right]\right\} - 15190 \qquad (4)$$

where

$-J_a = -0.34256 \times 10^5$

$f \sin^2 g = 0.59206995$

$f^2 \sin^3 g \cos g = 0.15142828$

$f = r_o v^2/GM$

$r_o = a + h$ (2.21×10^7 feet)

a = earth radius (2.0926×10^7 feet)

h = missile altitude at burnout (estimated) ($.1174 \times 10^7$ feet)

v = missile velocity at burnout (estimated) (2.12085×10^4 ft^3/sec^2)

GM = gravitational product of earth (1.4077×10^{16} ft^3/sec^2)

g = estimated burnout angle from vertical (66.5 degrees)

J = 0.001637

d = launch azimuth to time-of-flight advance-impact point

2-162. SUBPROGRAM U56 (ØTC5). ØTC5 reads the input for the target. The FORTRAN II reference statement is CALL ØTC5.

a. Inputs. No inputs are defined.

b. Outputs. The output is IFLAG set to identification integer 2166.

c. Program Logic. FD U56.

(1) Steps 1-7. RVCDRD reads the target R/V cards, and INTRØG interrogates SW(70) to determine if an error occurred in reading the target R/V cards. If ØN, the subprogram continues at step 8. If ØFF, IFLAG is set to identification integer 2166. RDTGTP reads the tape containing the target constants. INTROG interrogates SW(70) to determine if an error occurred. If ØN, the subprogram continues at step 8. If ØFF, the subprogram returns to the user subprogram.

(2) Step 8. ERRPRT prints the identification integer and a statement describing the type of error. The subprogram returns to the user subprogram.

 Mod F3

2-163. SUBPROGRAM P50 (ØTCNT). ØTCNT controls the offset target computations. It also controls the production and verification of a complete targeting kit. The FORTRAN II reference statement is CALL ØTCNT.

a. Inputs. The inputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
RSLTS	2,19,10	Output data, including T_1-T_{15}, for all 10 targets	array	
IRFSG	2,1,10	Output fuzing parameters for all 10 targets		
FRFSG	2,6,10	Fuzing parameter quantization adjustment factors on aim points		
RBIAS	2,3,10	Output aim point bias vectors for all 10 targets	Δ lat Δ lon Δ alt	
IDPTG	10,1,1	Duplicate target slot number storage		
TWGS	2	Current target - geoidal separation	G_{STC}	
TWAL	2	Current target - altitude above geoid	h_G	
TWDA	2	Current target - desired detonation altitude (above target)	h_{DD}	feet
GMALT	2	Altitude of point above earth ellipsoid	h_p'	
GMLØN	2	Longitude of point west of Greenwich	λ_p'	
GMLAT	2	Geocentric latitude of point	L_{CP}	

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
SW(158)	1	If ØN, first pass through ØTCNT		
SW(31)	1	If ØN, ØTC operation called		
SW(83)	1	If ØN, new target indicator		
SW(91)-SW(100)	1	Target switch for target slots 1-10. If ØFF, dummy target indicator		
SW(141)-SW(150)	1	If ØN, current target data is same as from an old tape		
SW(82)	1	If ØN, print statements		
SW(152)	1	If ØN, dummy target with excess horizontal miss distance has been detected		

b. Outputs. The outputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
ØTAPE	2,25,10	RSLTS table image from old paper tape		
RSLTS	2,19,10	Output data, including T_1-T_{15}, for all 10 targets		
IRFSG	2,1,10	Output fuzing parameters for all 10 targets		
FRFSG	2,6,10	Fuzing parameter quantization adjustment factors on aim points		
ISLØT	1	Current target slot number		
UTDIN	10,1	Target data inventory number		
IDGZP	10,1	Desired ground zero point		

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
TØGZ	10,1	Desired ground zero - numeric and alphabetic		
TØID	10,1	Target data inventory number		
VEXTR	50,1	Spare block		
TWGLT	2	Current target - geographic latitude north of equator	ϕ_T	degrees
TWLN	2	Current target - longitude west of Greenwich	λ_T	degrees

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

a. DUMMY TARGET INSERTED IN TARGET SLOT __

b. TARGET __ DUMMY AS INDICATED

c. TARGET __ IMPROPERLY ESTABLISHED AS A DUMMY

d. VERIFICATION RESULTS FOR TARGET SLOT __ WERE UNSUCCESSFUL

e. VERIFICATION RESULT FOR TARGET SLOT __ WERE SUCCESSFUL

f. DESIRED DETONATION POINT/LAT. ______ DEGREES. LON. ______ DEGREES. ALT. ______ FEET./LAT. ______ DEGREES. LON. ______ DEGREES. ALT ______/ACTUAL DETONATION POINT/LAT. ______ DEGREES. LON. ______ DEGREES. ALT. ______ LAT. ______ DEGREES. LON. ______ DEGREES. ALT. ______/MISS DISTANCE COMPONENTS IN FEET-/ NORTH ______ EAST ______ UP ______/NORTH ____ EAST ______ UP ______

g. VERIFICATION RESULTS FOR TARGET SLOT __ WERE SUCCESSFUL. T CONSTANTS AND FUSING PARAMETERS FROM TARGET SLOT __ COMPARE WITH THOSE FOR TARGET SLOT __

h. VERIFICATION RESULTS FOR TARGET SLOT __ WERE UN-SUCCESSFUL. T CONSTANTS AND FUSING PARAMETERS FROM TARGET SLOT __ DO NOT COMPARE WITH THOSE FOR TARGET SLOT __

i. VERIFICATION RESULTS FOR TARGET SLOT __ WERE SUC-CESSFUL. T CONSTANTS AND FUSING PARAMETERS FROM TARGET SLOT __ COMPARE WITH THOSE FOR OLD TAPE TARGET SLOT __

j. VERIFICATION RESULTS FOR TARGET SLOT __ WERE UN-SUCCESSFUL. T CONSTANTS AND FUSING PARAMETERS FROM TARGET SLOT __ DO NOT COMPARE WITH THOSE FOR OLD TAPE TARGET SLOT __

k. TARGETING KIT VERIFICATION HAS BEEN COMPLETED BY COMPUTER. RETURNING TO DOC CONTROL

l. VERIFICATION RESULTS FOR SAME AS TARGET SLOT WERE UNSUCCESSFUL/TARGET SLOT __ DID NOT VERIFY

c. Program Logic. FD P50

(1) Steps 1-16. SAVE4 saves the return path to the user subprogram in IFØUR. IFLAG is set to identification integer 1650. SW(151) is examined to determine if a constraint has been exceeded. If ØN, the subprogram continues at step 48. If ØFF, INTRØG interrogates SW(158) to determine if this is the first pass through ØTCNT (SW(158) = ØN). If ØFF, the sub-program continues at step 52. If ØN, IFLAG is set to identi-fication integer 1650. INTRØG interrogates SW(31) to deter-mine whether the OTC operator is to be called. If ØFF, the subprogram continues at step 52. If ØN, INTRØG interrogates SW(83) to determine if a new tape is being processed (SW(83) = ØN). If ØN, the subprogram continues at step 17; if ØFF, ØTC5 reads the old targeting inputs. The RSLTS table image is ob-tained from the old paper tape. The output data T constants, the fuzing parameter quantization adjustment factors on aim

point, the detonation point altitude adjustment due to fuzing quantization, and the 25g point detection time adjustment factors are set to zero. The subprogram continues at step 17.

(2) Steps 17-19. INGAIN establishes AGAIN for use in CKPTCK. CKPTCK establishes a check point within the program for rollback control. The current target slot number is initialized to zero and the subprogram continues at step 20.

(3) Steps 20-23. ISLØT is stepped by one and IFLAG is set to identification integer 1650. INTRØG interrogates the target switch pertaining to the current target slot number (SW(ISLØT + 90)) for a dummy target. If ØFF, indicating a dummy target, the subprogram continues at step 34; if ØN, the subprogram continues at step 24.

(4) Steps 24-33. If IDPTG indicates that this is a new target, the subprogram continues at step 39. Otherwise this is a SAME AS target and INTROG interrogates SW(ISLØT + 140) to determine if the target is from an old tape. If ØN, the output fuzing parameter and fuzing parameter quantization adjustment factors, the output aim point bias vector, and the output T constants are set equal to the values from the old paper tape. FAPSTR sets the target data inventory number and desired ground zero equal to SAC TDI and SAC DGZ. FAPDEC converts the binary DGZ and TDI to BCD. TAPDGZ converts the binary DGZ and TDI to BCD in the decrement. The subprogram continues at step 38. If OFF, INTROG interrogates SW(M + 90), M taking values from 1 to 10, to determine if the target is the same as

the previously calculated new target (SW(M+90) = ØFF). This step causes a dummy to be inserted for a SAME AS target if the original was made a dummy during an OTC run. If SW(M+90) is ØFF, the subprogram continues at step 48. If ØN, the output fuzing parameter and fuzing parameter quantization adjustment factors, the output aim point bias vector, the output T constants, the target data inventory number, and desired ground zero values are set equal to values computed for previous new target. The subprogram continues at step 38.

(5) Step 37. The target flag is set to minus one.

(6) Step 38. If ISLØT is less than 10, the subprogram continues at step 20. Otherwise the subprogram continues at step 99.

(7) Steps 39-46. This is a new target, SW(151) is set ØFF. U04 reads record ISLØT, file 1, of tape A4. If an error occurred in reading the tape, the subprogram continues at step 101. Otherwise GCØNST processes the ground guidance complex data, TARGET performs the offset target computations, and IFLAG is set to identification integer 1650. INTRØG interrogates SW(151) to determine if a constraint has been exceeded. If ØFF, the control is transferred to step 35. If ØN, the subprogram continues at the next step.

(8) Steps 48-51. The target flag is set to minus one. Switches SW(ISLØT+90) and SW(ISLØT+160) are set to ØFF, the target data inventory and desired ground zero numbers are set to zero, and statement a is written and printed. The

subprogram continues at step 35.

(9) Steps 52-55. ØTC5 reads the target R/V cards and the tape containing the target constants. INGAIN establishes AGAIN for use in CKPTCK. CKPTCK establishes a check point. ISLØT is set to zero.

(10) Steps 56-63. ISLØT is stepped by one. INTRØG interrogates SW(ISLOT+90) to determine if this is supposed to be a dummy target (SW(ISLØT+90) = ØFF). If ØN, IFLAG is set to identification integer 1650 and the subprogram continues at step 64. If ØFF, statement b is written and, if SW(82) is ØN, it is printed. The subprogram continues at step 97.

(11) Steps 64-68. If this is not a dummy target, the subprogram continues at step 69. Otherwise statement c is written and, if SW(82) is ØN, it is printed. The subprogram continues at step 97.

(12) Steps 69-72. SW(151) is set ØFF to remove on-line data printing in the verify mode. VERIFY verifies the targeting data. If the target being processed is the same as a target on an old tape, the subprogram continues at step 78; if it is the same as a target on a new paper tape, the subprogram continues at step 73. If the target being processed is a new target, the subprogram continues at step 77.

(13) Steps 73-76. JERK is set equal to the duplicate target slot number. If verification results of the target

slot are not successful, statement l is written and printed and the subprogram continues at step 97. Otherwise the subprogram continues at step 77.

(14) Step 77. If verification of T constants and fuzing parameters is successful, statement h is written. Otherwise statement g is written. The subprogram continues at step 87.

(15) Steps 78-89. JERK is set equal to the duplicate target slot number. If verification of T constants and fuzing parameters are successful, statement i is written. Otherwise statement j is written. The subprogram continues at step 87.

(16) Steps 81-86. IFLAG is set to identification integer 1650. If an improper dummy target is found with excess horizontal miss distance, statements d and f are written. Otherwise statements e and f are written.

(17) Steps 87-96. The statements (d, e, g, h, i, j), which were written, are printed.

(18) Steps 97-99. If ISLØT is less than 10, the subprogram continues at step 52. Otherwise statement k is written and printed. RTRN4 causes return to the user subprogram along the path established by SAVE4. Tape A4 is rewound.

(19) Steps 100-102. SW(151) is set ØFF and IFLAG is set to identification integer 2104. ERRPRT determines the type of error that has occurred, and writes and prints the appropriate error statement.

2-164. SUBPROGRAM P23 (PREARM). PREARM computes the current target prearm boundaries for cross range and down range distances, and crosswise-left and crosswise-right boundary distances. The FORTRAN II reference statement is CALL PREARM.

a. Inputs. The inputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
ISLØT	1	Current target slot number		
TWGLT	2	Current target geographic latitude north of equator	ϕ_T	
TOPD	2	Maximum latitude of prearm boundary circle	ϕ_{max}	
TOPU	2	Minimum latitude of prearm boundary circle	ϕ_{min}	
TWLN	2	Current target longitude west of Greenwich	λ_T	degrees
TOPR	2,10	Maximum longitude of prearm boundary circle	λ_{max}	degrees
TOPL	2,10	Minimum longitude of prearm boundary circle	λ_{min}	degrees
TWDRV	2	Current target distance to center of earth	r_T	feet
GNAUT	2	Conversion factor nautical miles to feet (=6076.1033)		ft/naut mile
FRTØD	2	Conversion factor degrees to radians		deg/rad
GØMGA	2	Rate of earth rotation (=7.2921158×10^{-5})	Ω	rad/sec

b. Outputs. The outputs are as follows:

COMMON TAG	DIMENSION	ITEM	SYMBOL	UNITS
XW(151)	1	If ØN, prearm boundaries have been exceeded		
RSLTS	2,19,10	Target flag output for ten targets		
XT(1,19)- XT(2,19)	2	T constant target flag	T19	
XT(1,4)- XT(2,4)	2	Maximum northern latitude prearm boundary limit	T4	naut miles
XT(1,5)- XT(2,5)	2	Minimum southern latitude prearm boundary limit	T5	naut miles
XT(1,6)- XT(2,6)	2	Maximum western longitude prearm boundary limit	T6	naut miles
XT(1,7)- XT(2,7)	2	Minimum eastern longitude prearm boundary limit	T7	naut miles
IFLAG	1	Subprogram identification		integer

The following statements are also outputs:

a. PREARM BOUNDARY GREATER THAN 2000 MILES FROM TARGET. DUMMY TARGET INSERTED.

b. TARGET LESS THAN 50 MILES FROM CLOSEST ARMING BOUNDARY. DUMMY TARGET INSERTED.

c. Program Logic. FD P23

(1) Steps 1-6. The subprogram is initialized for the current target using ISLØT. The maximum and minimum latitude differences $\Delta\phi_{max}$ and $\Delta\phi_{min}$, are computed by use of expressions (1) and (2). The real-world target longitude λ_T is examined. If negative, 360 degrees are added to give λ_T'; otherwise λ_T itself becomes λ_T'. The maximum longitude boundary λ_{max} is examined. If negative, 360 degrees are added to give λ_{max}'; otherwise λ_{max} itself becomes λ_{max}'. The maximum

longitude difference $\Delta\lambda_{max}$ is computed by use of expression (3) and examined. If greater than or equal to 180 degrees, λ_T is subtracted from λ'_{max} to give $\Delta\lambda'_{max}$; otherwise $\Delta\lambda_{max}$ itself becomes $\Delta\lambda'_{max}$.

(2) Steps 7-13. The minimum longitude boundary is examined. If negative, 360 degrees are added to give λ'_{min}; otherwise λ'_{min} itself becomes λ_{min}. The minimum longitude difference is computed by use of expression (4) and examined. If greater than or equal to 180 degrees, λ_T is subtracted from λ'_{min} to give $\Delta\lambda'_{min}$; otherwise $\Delta\lambda_{min}$ itself becomes $\Delta\lambda'_{min}$. The absolute values of $\Delta\phi_{max}$, $\Delta\phi_{min}$, $\Delta\lambda'_{max}$, and $\Delta\lambda'_{min}$ are obtained. The maximum angular distance between the target and the closest arming boundary is the largest Δ_{max} in the following group: $|\Delta\phi_{max}|$, $|\Delta\phi_{min}|$, $|\Delta\lambda'_{max}| \cos\phi_T$, and $|\Delta\lambda'_{min}| \cos\phi_T$.

(3) Steps 14-22. The distance D_{max} from the target to the closest arming boundary is computed using expression (7). If D_{max} is greater than or equal to 2000 miles, SW(151) is set ØN, statement a is printed and written, and control is transferred to step 36. Otherwise the smallest value Δ_{min} is selected from the longitude and latitude differences previously computed. The distance D_{min} is computed using expression (8). If D_{min} is greater than 50 miles, SW(151) is set ØN, statement b is printed and written, and control is transferred to step 36. Otherwise the subprogram continues at step 23.

(4) Steps 23-32. The value D_{min} is compared again with 50. This step is included to provide for a future comparison with 25 miles. Obviously, D_{min} is less than or equal to 50 (from the previous comparison), and the target flag output and T19 are set to one. Otherwise both would be set to zero.

The final Δ_{max} and Δ_{min} are computed, including the 4 mile correction, by use of expressions (9) and (11). Constants T4, T5, T6, and T7 are computed by use of expressions (10), (12), (13), and (14). CUTIE is stepped by one.

(5) Step 33. IFLAG is set to identification integer 1623. Control is returned to the user subprogram.

d. Expressions.

$$\Delta \phi_{max} = \phi_{max} - \phi_T \qquad (1)$$

$$\Delta \phi_{min} = \phi_{min} - \phi_T \qquad (2)$$

$$\Delta \lambda_{max} = \left| \lambda'_{max} - \lambda'_T \right| \qquad (3)$$

$$\Delta \lambda_{min} = \left| \lambda'_{min} - \lambda'_T \right| \qquad (4)$$

$$\Delta \lambda''_{max} = \left| \Delta \lambda'_{max} \right| \cos \phi_T \qquad (5)$$

$$\Delta \lambda''_{min} = \left| \Delta \lambda'_{min} \right| \cos \phi_T \qquad (6)$$

$$D_{max} = \frac{r_t \, \Delta_{max}}{6076.1033} \qquad (7)$$

$$D_{min} = \frac{r_t \, \Delta_{min}}{6076.1033} \qquad (8)$$

$$\Delta'_{max} = \frac{\left| \Delta \phi_{max} \right| \, r_t}{6076.1033} - 4 \qquad (9)$$

$$T4 = \frac{r_t \sin \left(\frac{\Delta'_{max} \; 6076.1033}{r_t} \right)}{6076.1033} \qquad (10)$$

$$\Delta'_{min} = \frac{\left| \Delta \phi_{min} \right| \, r_t}{6076.1033} - 4 \qquad (11)$$

$$T5 = - \frac{r_t \sin \left(\frac{\Delta'_{min} \; 6076.1033}{r_t} \right)}{6076.1033} \qquad (12)$$

$$T6 = \frac{1}{6076.1033 \; \Omega} \sin \left[\frac{r_t \Delta \lambda''_{max} - 4 \times 6076.1033}{r_t \cos \phi_T} \right] \qquad (13)$$

$$T7 = - \frac{1}{6076.1033 \; \Omega} \sin \left[\frac{r_t \Delta \lambda''_{min} - 4 \times 6076.1033}{r_t \cos \phi_T} \right] \qquad (14)$$

The angles are expressed in degrees where required for the computations.

2-165. SUBPROGRAM U60 (RDTGTP). RDTGTP reads the target tape constants T, M, and G. The subprogram assembles the constants, converts them to floating point form, and stores them in the proper storage blocks. The FORTRAN II reference statement is CALL RDTGTP.

a. Inputs. The inputs are as follows:

COMMON TAG	DIMENSION	ITEM
SW(127)	1	If ØN, old tape run If ØFF, new tape run
RDRUM	2,19,10	Guidance computer drum slot addresses and scaling factors to appear on output paper tape

b. Outputs. The outputs are as follows:

COMMON TAG	DIMENSION	ITEM	UNITS
RSLTS	2,19,10	Matrix - output data for all ten targets, T	pure no.
XGI	2,24	Matrix - G constants table	pure no.
XM	2,30	M constants table used by guidance computer	pure no.
ICØMP	1	Compatability integer	integer
ICØNT	1	Control number for target kit	integer
IFLAG	1	Identification integer	integer
SW(75)	1	DOC/OTC card indicator	
SW(90)	1	Series simulation indicator	
SW(158)	1	If ØN, TOT/1st pass thru ØTCNT If ØFF, Verify pass thru ØTCNT	

The following printed and written statements also are outputs:

a. UNABLE TO READ RECORD ON TARGET TAPE

b. VALUE _____ IN XGI TABLE DOES NOT AGREE
WITH VALUE _____ ON TAPE

c. ADDRS _____ IN RDRUM TABLE DOES NOT AGREE WITH
ADDRS _____ ON TAPE

c. Program Logic. FD U66

(1) Steps 1-8. The contents of the index registers are saved, and IFLAG is set to identification integer 2166. U04 reads the record indicated by the first of the series simulator switches (SW(91)-SW(100)) which is ØN. Tape A4 is rewound. If an error occurred in U04, control is transferred to step 65.

(2) Steps 9-14. If the old tape is to be used (SW(127) = ØN), all following tape instructions are initialized to address the old tape B6. Otherwise, they are initialized to address the new tape B2. The proper tape then is rewound, the first 12 records are skipped. and the redundancy and end-of-file indicators are turned ØFF.

(3) Steps 15-25. The number of read attempts is set to zero. One record is read from the tape. If a redundancy persists after ten attempts to read the record. U08 prints and writes statement a, ITYER is set to eight, SW(70) is set ØN, and the subprogram exits to ERRPRT U13. If an end-of-file was read, control is transferred to step 26 to assemble the T constants that were read. Otherwise, control is trans-

ferred to step 15 to read another record.

(4) Steps 26-35. The first set of octal T constants are assembled. Constants T1 through T14 and T19 are converted to floating point. All of the constants (T1 through T14 and T17 through T19) are stored in the RSLTS table. If T20 and T21 were assembled previously, the subprogram continues at step 37. Otherwise, the subprogram assembles T20 and sets it equal to ICØMP. If either SW(158) or SW(75) is ØFF, T21 is assembled and set equal to ICØNT and the subprogram continues at step 37. Otherwise, the subprogram continues at step 36.

(5) Step 36. Assemble the (10-RECRD)th set of T constants.

(6) Steps 37-38. Counter RECRD is incremented by one and compared to ten. If less than 10, the program continues at step 36. If equal to 10, the program transfers control to step 39.

(7) Steps 39-52. M constants are assembled in octal form and tested one at a time. If the addresses of the assembled M constants are not equal to the addresses of the corresponding M constants in core, the subprogram enters U08 and prints and writes statement c. The octal M constants on tape are converted to floating point form. Tests then are performed to determine if the values of the assembled M constants agree with those in core. If not, the subprogram

enters U08 and prints and writes statement b. When all M constants are assembled, the subprogram continues at step 53.

(8) Steps 53-63. G constants are assembled in octal form. converted to floating point, and tested one at a time. If the addresses of the assembled G constants are not equal to the addresses of the corresponding G constants in core, the subprogram enters U08 and prints and writes statement c. Tests then are performed to determine if the values of the assembled G constants agree with those in core. If not, the subprogram enters U08 and prints and writes statement b. When all G constants are assembled, the subprogram continues at step 64.

(9) Steps 64-65. The tape in use is rewound and the contents of the index registers restored. Control is returned to the user subprogram.

 mod F1

2-166. SUBPROGRAM U61 (RVCDRD). RVCDRD reads the four IBM equivalent of the R/V cards into their respective card image areas. The card images are converted to the integer form of the output fuzing parameter. The FORTRAN II reference statement is CALL U61. The FAP reference instruction is TSX U61, 4.

a. Inputs. The inputs are the four target cards. Figure 2-3 illustrates the contents of the IBM equivalent of the R/V cards. The following registers are also inputs:

COMMON TAG	DIMENSION	ITEM
SW(91)-SW(100)	10	If ØFF, dummy target
SW(101)-SW(110_	10	If ØN, handover target
SW(161)-SW(170)	10	If ØN, punch target R/V card

b. Outputs. The outputs are as follows:

COMMON TAG	DIMENSION	ITEM	UNITS
IRFSG	2,1,10	Output fuzing parameter for all ten targets	integer
IFLAG	1	Identification integer	
SW(70)	1	If ØN, error occurred	
ITYER	1	Error indicator	

The following written and printed statements are also outputs:

a. GOE CARD OF TARGET _____ HAS COMPLETE SHAPE CODING PUNCHES, BUT SHOULD NOT