AD-A079 489

AMSAA

TECHNICAL REPORT NO. 272

A TRANSPORT AND DIFFUSION MODEL FOR SMOKE MUNITIONS

ROBERT M. MARCHETTI

AUGUST 1979

U. S. ARMY MATERIEL SYSTEMS ANALYSIS ACTIVITY
ABERDEEN PROVING GROUND, MARYLAND

DISPOSITION

Destroy this report when no longer needed. Do not return it to the originator.

DISCLAIMER

The findings in this report are not to be construed as an official Department of the Army position unless so specified by other official documentation.

WARNING

Information and data contained in this document are based on the input available at the time of preparation. The results may be subject to change and should not be construed as representing the DARCOM position unless so specified.

TRADE NAMES

The use of trade names in this report does not constitute an official endorsement or approval of the use of such commercial hardware or software. The report may not be cited for purposes of advertisement.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Technical Report No. 272	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A TRANSPORT AND DIFFUSION MODEL FOR SMOKE MUNITIONS		5. TYPE OF REPORT & PERIOD COVERED
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert M. Marchetti		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS US Army Materiel Systems Analysis Activity Aberdeen Proving Ground, Maryland		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS DA Project No. 1R765706M541
11. CONTROLLING OFFICE NAME AND ADDRESS Commander, US Army Materiel Development & Readiness Command, 5001 Eisenhower Avenue Alexandria, VA 22333		12. REPORT DATE AUGUST 1979
		13. NUMBER OF PAGES 168
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Smoke
Transport
Diffusion
Gaussian
Obscuration
Aerosol
Modeling
White phosphorus
Hexachlorethane
Detection

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The US Army Materiel Systems Analysis Activity (AMSAA) smoke model was developed to provide smoke effectiveness data for the tactical planner. Its purpose is to predict the size, duration, and effectiveness of an obscuring screen in a battlefield situation. Either burning pyrotechnics or explosive munitions may be handled. Starting with the weapon's or battery aimpoints, the screening cloud is developed, diffused, and transported in accordance with environmental conditions. Aiming and precision errors developed in the firings create impact "displacements which can produce "holes" in the smoke screen. The screen is

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. ABSTRACT (CONTINUED).

continuously examined to evaluate its effectiveness as determined by the obscuration along various lines of sight originating from an observer's position. The model is adaptable to a wide spectral range and various visual aids depending on the input data employed.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Next page is blank.

A TRANSPORT AND DIFFUSION MODEL FOR SMOKE MUNITIONS

1. INTRODUCTION

The AMSAA smoke model, referred to as the Smoke Effectiveness Manual Model (SEMM), as described here was derived from the original JTCG Smoke Obscuration Model No. 1 (SOM-1) whose principal author was M. C. Johnson. In order to make the model more manageable, Mr. G. Hanna of AMSAA and the Oklahoma State University Engineering Field Office at Eglin Field, Florida, revised the SOM-1 by removing the more doubtful transmission and brightness calculations leaving this information to be supplied by calibrated test results. Also, additional features were included such as the effects of weapon delivery errors and a comprehensive analysis of the density distribution of the smoke cloud.

The AMSAA smoke model forms the basis for constructing the obscuration tables given in the Smoke Effectiveness Manual published by the Joint Technical Coordinating Group for Munitions Effectiveness. It considers the delivery of white phosphorus (WP) or hexachlorethane (HC) muntions by indirect fire weapons to selected aimpoints located at a given range from the delivery weapons. Single or multiple volleys may be fired. After impact, the computer program determines the amount of obscurant at various time intervals along numerous lines of sight. The obscuring screen is transported and diffused as a function of local meteorological conditions during which time a criterion is applied to determine if target detection can be achieved with the particular visual aid employed. The model is adaptable to a number of spectral ranges and visual aids depending on the input data used. The following distinctive features are noteworthy:

- The smoke model is a transport and diffusion model and requires transmission data.
- The model assumes an uncorrelated guassian trivariate distribution for each obscuring burst.
- The model produces "holes" or discontinuities in the smoke screen due to the aiming and precision errors of indirect firing weapons.
- The model is used for detection but with the proper data can be used for recognition and identification.

According to Johnson, Reference 2, the gaussian trivariate distribution of aerosols, chosen for this model, evolved from the works of Sutton, Calder and Milly. Milly compared the model against numerous test data on munitions of various types in order to establish the values of the model's constants. Johnson states that the model is usually selected by analysts for short to medium cloud travels because of the simplicity of parameters required for input information.

A recent adjustment has been made to the WP portion of the model and is currently being evaluated. This change will improve the correlation of the model output with field data from smoke tests. Documentation will be published at a later date.

2. MODEL DESCRIPTION

2.1 General.

The AMSAA model can be used for both WP and HC munitions. Some model changes are required when employing either agent but a basic similarity exists between both applications.

When bursted, a WP munition develops an initial size which proceeds to diffuse and transport according to meteorological conditions. Similarly, the smoke produced by HC munition is treated as a large number of small individual continuous bursts or puffs which form an elongated screen with a fixed origin. In both cases, the bursts are assumed to have a trivariate gaussian distribution with standard deviations σ_x, σ_y, σ_z at any time t given by Sutton as:

$$\sigma_x = 0.1522 \left(\frac{Ut + A}{1.0}\right)^{.9294}$$

$$\sigma_y = 3.41 \left(\frac{Ut + B}{100}\right)^{\alpha} \qquad (2.1)$$

$$\sigma_z = 1.35 \left(\frac{Ut + C}{20}\right)^{\beta}$$

The quantity U represents the mean wind velocity near the ground and t measures the age of each burst. The exponents α and β are functions of the temperature gradient in the vicinity of the ground. The quantities A, B, and C account for the size of the initial burst of the munition. These values are zero for HC rounds but have the following values for WP rounds:

$$A = 1.0 \left(\frac{\sigma_{xs}}{.1522}\right)^{1/.9294}$$

$$B = 100 \left(\frac{\sigma_{ys}}{3.41}\right)^{1/\alpha} \qquad (2.2)$$

$$C = 20 \left(\frac{\sigma_{zs}}{1.35}\right)^{1/\beta}$$

where σ_{xs}, σ_{ys}, and σ_{zs} are the source values of the standard deviations and depend on the WP fill weight of the munition. The constants in Equations (2.1) and (2.2) were established by Milly, Reference 1, in accordance with test data but have been modified somewhat by past users. The growth of these sigmas with time establishes the diffusion qualities of the bursts and consequently that of the smoke screen.

For calculating convenience, the model employs three coordinate systems; these are (a) an earth fixed system, (b) a weapon axis system with the y axis extended in the direction of the line of fire, and (c) an axis system so oriented that its x axis is always along the wind direction. The latter facilitates the integration of smoke density along the selected lines of sight.

The burst points of the munitions differ from the intended aim points by the aiming and precision errors in firing. Figure 1 shows the intended deployment of munitions in a lazy W pattern along with the actual bursts as delivered by a battery of six weapons, firing at 2/3 their maximum range, on one occasion. An estimate of the range and deflection deviations of the actual centroid of burst points from the intended centroid of bursts due to the aiming error is estimated by sampling the random variable x from a uniform source and proceeding as shown in Figure 2 to arrive at the corresponding miss distances R and D. These distances are laid out along the weapon system axis.

A similar procedure is followed to find the contribution of the precision error at each burst point. Components of the precision and aiming error are summed to arrive at the total error between the intended and actual burst points. Range and deflection standard deviations for both errors must be known.

The inclusion of munition delivery errors is essential to the development of "holes" or periods of relatively short visibility through the smoke screen. Precision errors are mainly instrumental in creating "holes" in predominently cross wind cases whereas aiming errors can produce significant smoke screen displacements in obscuring an enemy front in prevailing head or tail wind situations.

During the development of the smoke screen from the individual bursts, the density of the screen at any point is the sum of the densities of the individual burst taken at that point. With a mean ground wind velocity U in a direction x, the cloud density at point x, y, z in the wind axes system is given by:

$$\mathrm{DEN} = \frac{2}{(2)^{3/2}} \sum_{1}^{N} \frac{Q\lambda\Omega}{\sigma_x\sigma_y\sigma_z} \exp\left\{ -\frac{1}{2}\left[\frac{(x-Ut)^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} + \frac{(z-z(t))^2}{\sigma_z^2} \right] \right\} \quad (2.3)$$

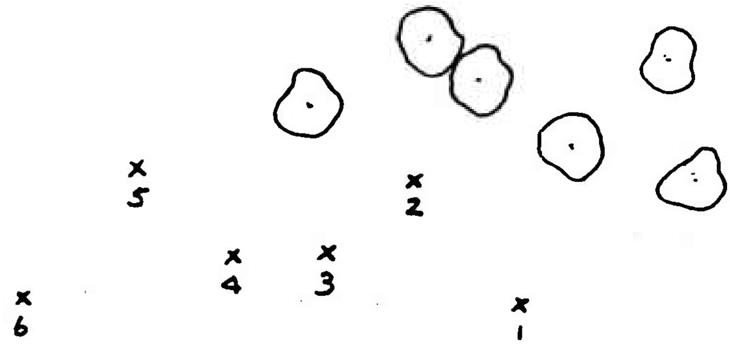
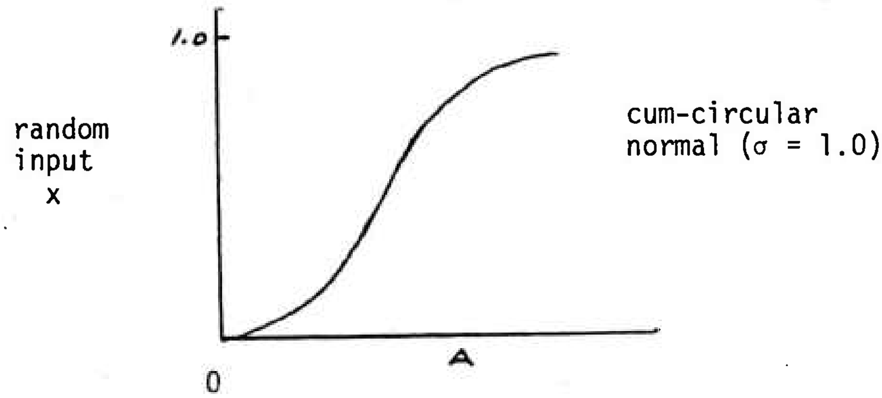


FIGURE 1 - MUNITION DISPERSION DUE TO PRECISION AND AIMING ERRORS

σ_R = range std deviation

σ_D = deflection std deviation



$$B = 2\pi x$$

$$R = A \sin B\sigma_{\beta}$$

$$D = A \cos B\sigma_D$$

FIGURE 2 - DEVELOPMENT OF PRECISION AND AIMING ERRORS

where λ is the munition efficiency with which the smoke producing material is used.

Q is the weight of the smoke producing material prior to dissemination, excluding weight lost in the plume.

Ω is the yield factor associated with the physio-chemical reaction process which converts smoke producing material into smoke. For hydroscopic agents this quantity is mainly a function of relative humidity.

$z(t)$ is the function which describes the vertical motion of the smoke due to updrafts from heat released dur-production.

N is the number of bursts of WP or puffs of HC forming the smoke screen.

The integration of the density Equation (2.3) is carried out along various lines of sight L_i originating at the observer's position and passing throught the screen. The mass per unit area of the obscurant along line L_i is therefore:

$$\text{MASS (L)} = \sum_{1}^{N} \int_{0}^{L} \text{(DEN) dL} \tag{2.4}$$

The line of sight L_i is an obscured path if and when:

$$\text{MASS } (L_i) > \text{CLTHRS} \tag{2.5}$$

where CLTHRS is the threshold value for the combined smoke agent and optical aid employed. That is, it is the mass per unit area through which the probability of target detection is derived due to the attenuation of the obscurant.

The obscuration characteristics of the smoke cloud is determined by exploring various lines of sight (LOS) through the developing screen. Figure 3 shows a screen formed by the diffusion of three bursts. An enclosing rectangle is given by the four sigma values of those bursts which define the rectangle's maximum dimensions. The size of the rectangle gives the limiting locations of the exploratory LOS's and assumes that these LOS's form unobscured paths. Figure 4 expands this idea indicating the individual LOS's along with their respective separating lengths, Δx. The screen length is defined by the first and last LOS which satisfies the condition given in Equation (2.5). Those interior Δx's which correspond to those LOSs which fail to meet this criterion are summed to give the effective size of the "hole" in the screen. The effective cloud length is given by the total length less the "hole" size. Replications of the above procedure, repeatedly sampling the precision and aiming errors, provide the average values of the total and effective cloud lengths.

The model makes provisions to explore the characteristics of the smoke cloud in greater depth. Thus the densities at numerous points within the smoke cloud, at a given height above ground level, are presented as an array of data thereby making it possible to follow the time history of high and low density areas. Another feature is the calculation of the smoke concentration along numerous lines of sight originating at points along the near, or observer edge, of the cloud and terminating at points along the far edge. In this way multi observer-target LOS's are explored for each cloud time history.

3. MODEL INPUT DATA

The model inputs needed to produce obscuration results may be divided into two parts, selected inputs and fixed inputs. The selected inputs are those which the user can choose in battlefield planning in order to employ the JTCG/ME Smoke Manual. They are also the most significant parameters determining the history of the smoke cloud. The fixed inputs are those, which have a lesser influence on the dissemination of the smoke cloud and which are not easily discernible or calculable on the battle field. They are inputs that are either held constant in developing the smoke tables or values that are dependent on those of the selected inputs.

3.1 Selected Inputs.

3.1.1 Meteorological Conditions. Select meteorological conditions have been reduced to five combinations of atmospheric stability and wind velocities in accordance with the Pasquill grouping of these parameters. These are in Table 1.

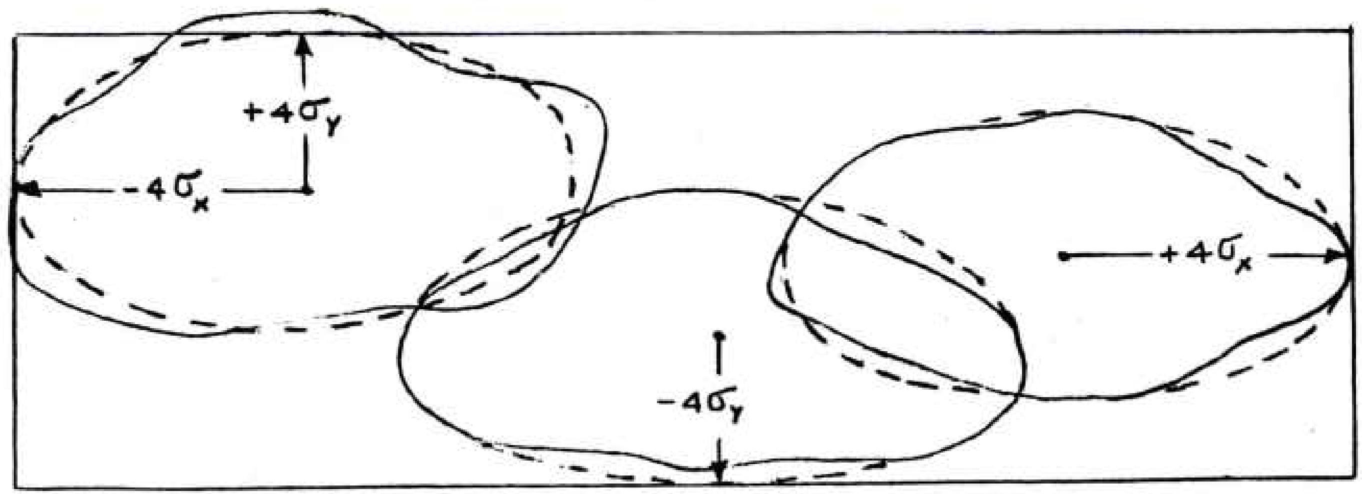


FIGURE 3 - SMOKE SCREEN EXPLORATORY REGION

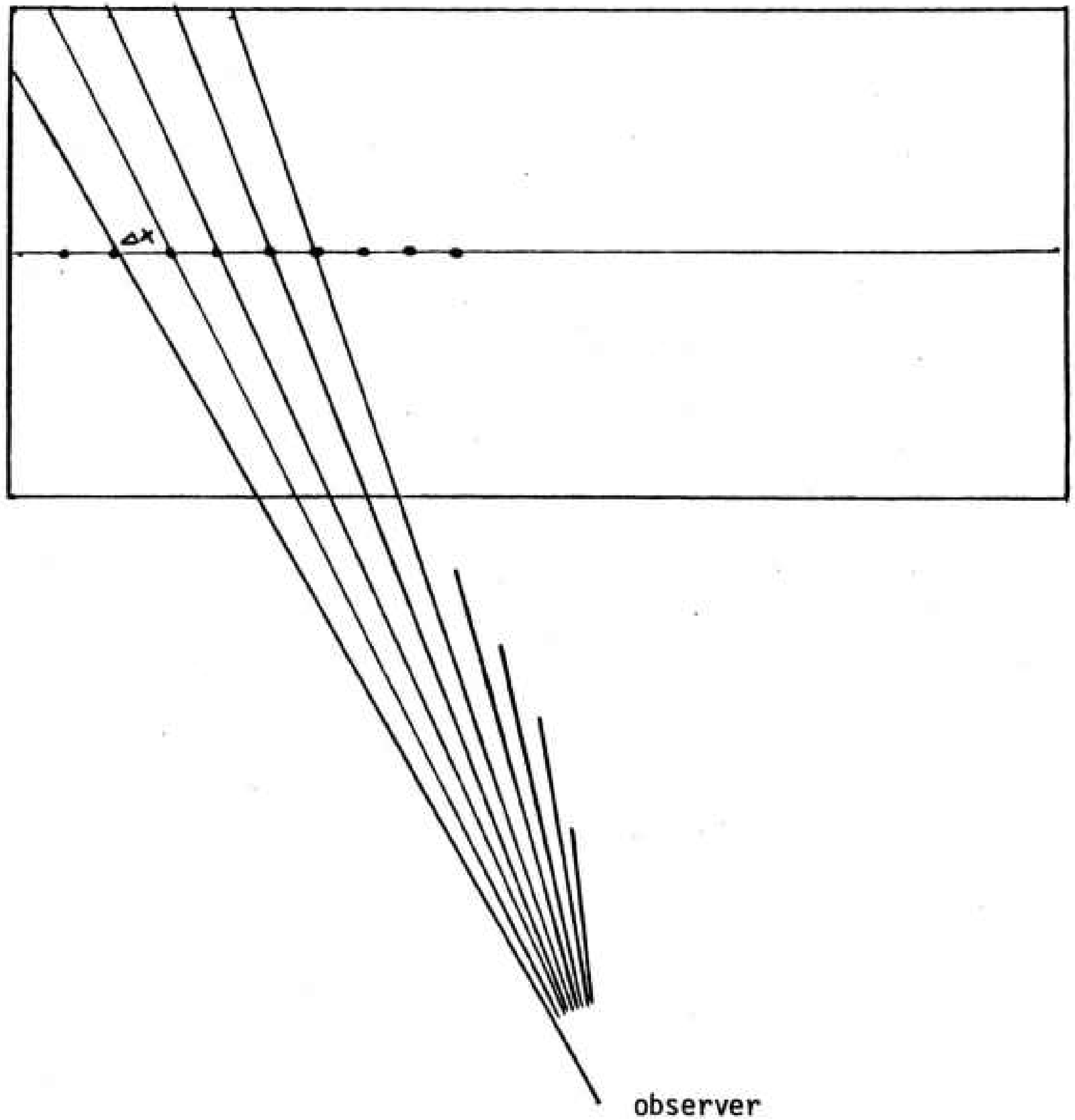


FIGURE 4 - LINE UP SIGHT EXPLORATORY REGION

TABLE 1. METEOROLOGICAL CONDITIONS

Stability	Wind Speed, knots	Approx. Pasquill Category
Ideal (Inversion)	5	E to F
Favorable (Neutral)	5	D to C
Favorable (Neutral)	10	D to C
Favorable (Neutral)	15	D to C
Marginal (Lapse)	5	A to B

3.1.2 Wind Direction. The wind directions are chosen with reference to the observer-to-target LOS. These are crosswind, head or tail wind, and quartering wind.

3.1.3 Munition Type. The model will produce obscuration results for currently available WP and HC munitions. Those to be included in the Smoke Manual are shown in Table 2.

TABLE 2. SELECTED SMOKE MUNITIONS

105mm, M84B1, HC

105mm, M60A2, WP

155mm, M110E2, WP

155mm, M116B1, HC

4.2", M328A1, WP

81mm, M375A2, WP

5", 5.54 CAL., WP

3.1.4 Volley Size. One or more multi-round volleys may be fired at selected times between volleys. To avoid lengthy tables in the Smoke Manual, only the effects of single round or multi-round volleys will be listed and the effects of additional volleys approximated by simple manual calculations.

3.1.5 Spectral Bands. The model is adaptable to any spectral band for which the sensor's capability against a particular smoke agent is available. This capability is represented by the concentration-length of smoke which the sensor can defeat. For the Smoke Manual, results will be listed for the visible spectrum, for an anti-tank guided missile and for the currently available night sight operating against the smoke munitions.

3.2 Fixed Inputs.

As stated in paragraph 3, the fixed inputs are those which are either held constant in preparing the smoke tables or whose values are average values corresponding to the range of selected inputs.

3.2.1 Yield Factor. Most military screening smokes consist chiefly of hydroscopic products of chemical reactions which form dilute solutions in the presence of atmospheric water vapor. The resulting smoke clouds are actually suspensions of small liquid droplets. Both WP and HC munitions are in this class and the mass of smoke they produce is greater than the mass of the original dry agent. Fog oil smoke, on the other hand, is an atomization of a liquid and the smoke mass is equal to the mass of oil consumed. The yield factor is the ratio of the wet smoke mass produced to the mass of dry agent consumed. Yield factors for various smoke agents, taken from Reference 2, are shown in Figure 5 plotted as a function of relative humidity. A constant relative humidity of 50 percent was used in developing the smoke tables. Table 3 gives values of yeild factor used in developing the smoke tables.

TABLE 3. YIELD FACTORS FOR 50 PERCENT RELATIVE HUMIDITY

Type Munition	Yield Factor
WP	5.2
HE	1.85

3.2.2 Diffusion Parameters α and β. According to Milly, Reference 1, the diffusion of the smoke bursts (or puffs in the case of HC) are controlled by the meteorologically dependent quantities α and β which are determined by the local temperature gradient. A reproduction of Milly's curve is shown in Figure 6. α and β develop the standard deviations for the trivariate smoke density distributions in directions perpendicular to the wind direction. Along the wind directions, the diffusion parameter δ has the constant value 0.9294. The specific temperature gradients and corresponding α and β values are

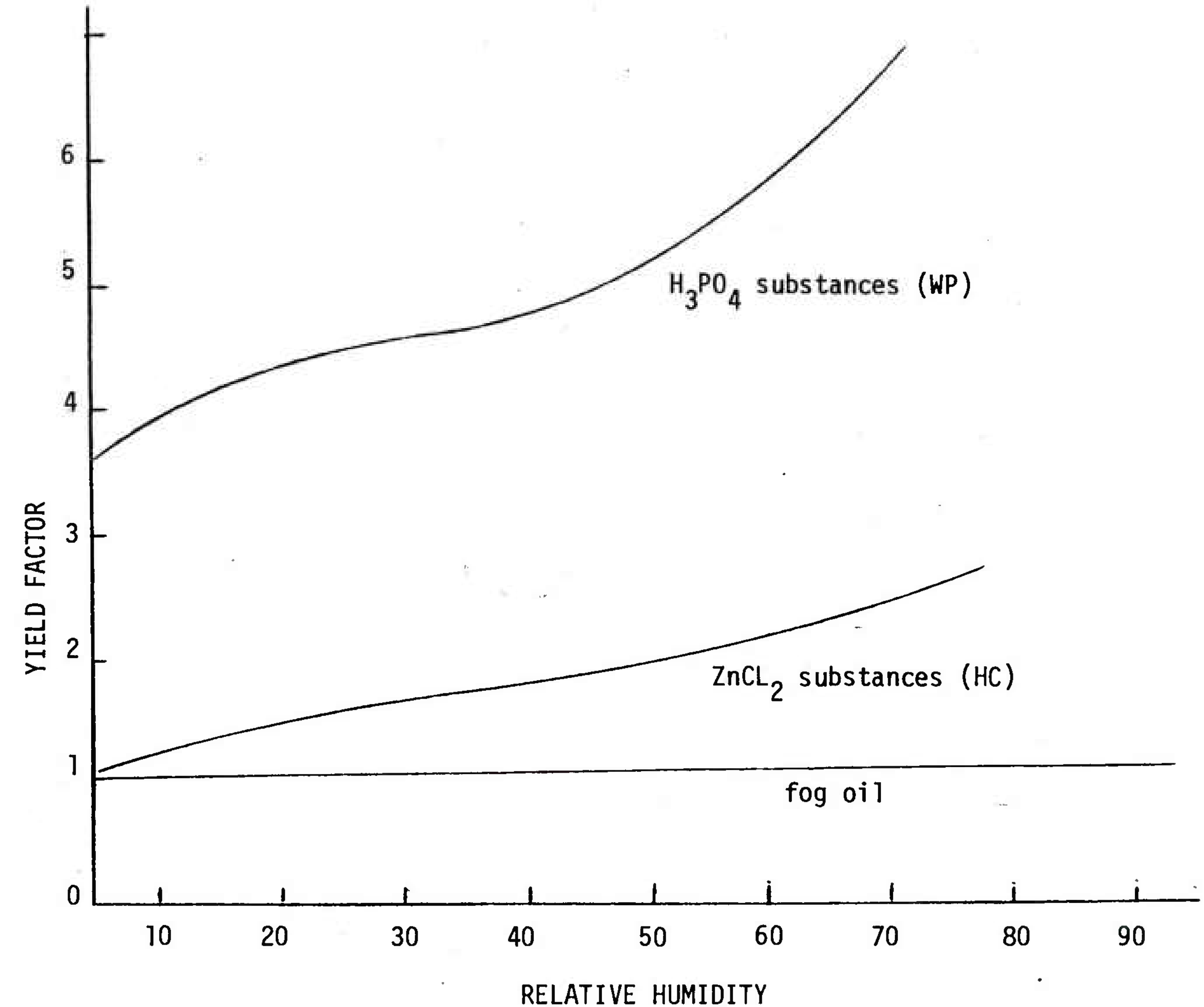


FIGURE 5 - YIELD FACTOR AS A FUNCTION OF RELATIVE HUMIDITY FOR VARIOUS SMOKE PRODUCING SUBSTANCES

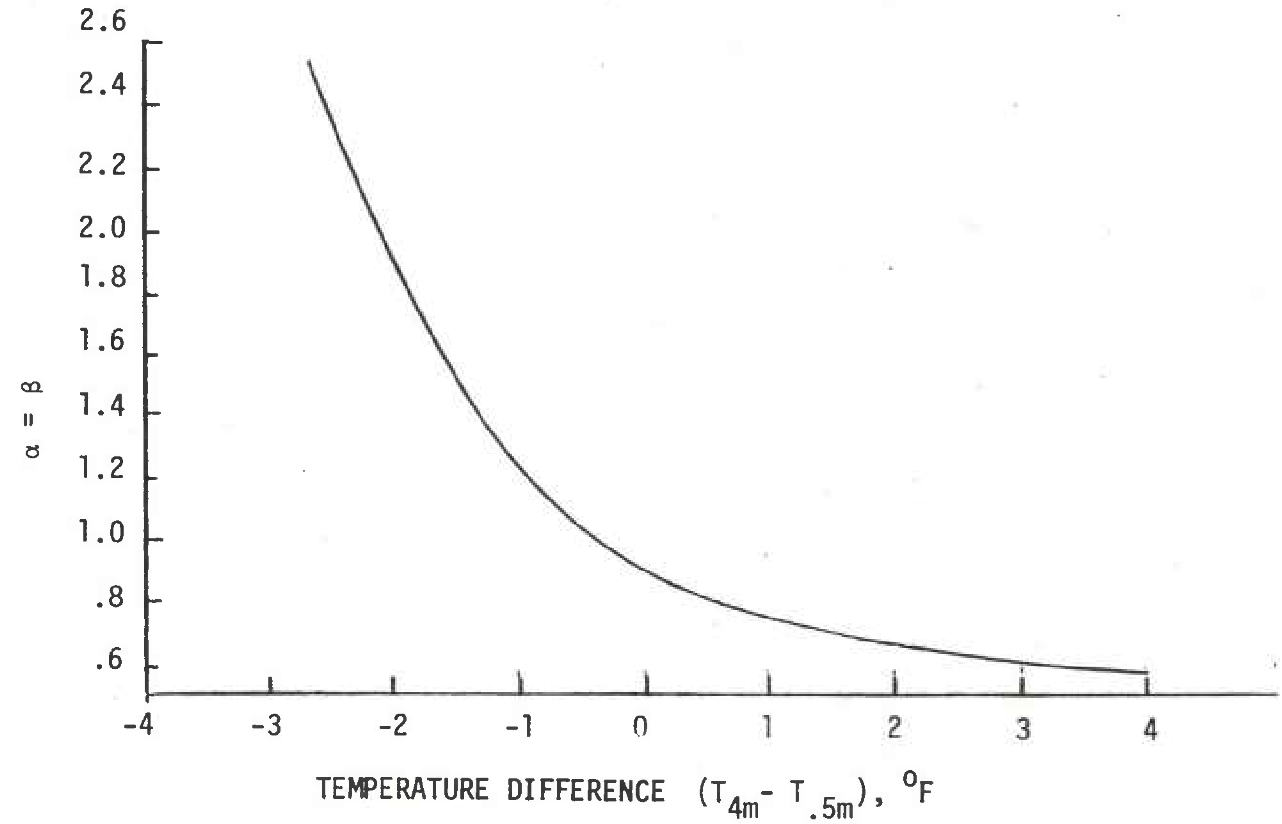


FIGURE 6 - ($\alpha = \beta$) vs TEMPERATURE DIFFERENCE

given in Table 4 for the three atmospheric stability conditions covered in the smoke tables. The gradient is represented by the average temperature difference between a height of 4 meters and 0.5 meters along the local terrain.

TABLE 4. α AND β VALUES FOR THREE STABILITY CONDITIONS

ΔT, ^{o}F	$\alpha = \beta$	Stability Conditions
- 1.5	1.5	Lapse
0	0.88	Neutral
+ 1.5	0.70	Inversion

The stability conditions are selected inputs.

3.2.3 Munition Fill Weight. The AMSAA Model uses the munition fill weight modified by an efficiency factor to account for two effects; (1) the amount of fill actually converted to dry smoke and (2) the amount of smoke which is lost in the rising plume. The former is determined from test results while the latter is dependent on atmospheric stability and is employed instead of using a smoke rise function. The product of fill weight and efficiency is called the effective fill weight for the munition and its values are given in Table 5 for the munitions listed in Section 3.1.3.

TABLE 5. EFFICIENCIES AND FILL WEIGHTS FOR VARIOUS MUNITIONS

		Efficiency			Effective Fill Wt. (lb)		
Munition	Fill Wt. (lb)	Lapse	Neutral	Inversion	Lapse	Neutral	Inversion
105 WP	3.85	0.10	0.10	0.30	0.38	0.38	1.15
105 HC	7.50	0.75	0.75	0.75	5.60	5.60	5.60
155 WP	15.6	0.10	0.10	0.30	1.56	1.56	4.68
155 HC	25.8	0.75	0.75	0.75	19.4	19.4	19.4
4.2 in. WP	7.5	0.40	0.40	0.60	3.0	3.0	4.5
81mm WP	1.75	0.10	0.10	0.30	0.175	0.175	0.53
5 in. WP	8.34	-	-	-	-	-	-

3.2.4 Source Sigmas, σ_{xs}, σ_{ys}, σ_{zs}. While HC smoke originates from a small area which may be considered a point source, WP smoke originates as the result of a bursting charge which ignites and separates the filler material. The distribution of this material is assumed to have the same gaussian characteristics as the subsequently diffused cloud. The source sigmas, defined as σ_{xs}, σ_{ys}, σ_{zs} in Equation (2.2), characterize the initial WP burst and have values strongly dependent on the munition's effective fill weight. Their growth with time defines the diffusion process of the smoke screen. Under calm wind conditions σ_{xs} is equal to σ_{ys} so that the cloud obtains a circular symmetry in planes parallel to the ground. In strong winds, the down wind sigma, σ_{xs} may be skewed in the wind direction. For HC munitions the source sigmas are given a value of zero. Figure 7, taken from Reference 2, shows the calm wind sigmas plotted against the munition's effective fill weight as determined by limited test results. For the WP munitions listed in Section 3.2.3, the source sigmas were used in preparing the Smoke tables.

TABLE 6. SOURCE SIGMAS FOR VARIOUS WP MUNITIONS

	$\sigma_{xs} = \sigma_{ys}$, meters			σ_{zs}, meters		
Munition	Lapse	Neutral	Inver.	Lapse	Neutral	Inver.
105 WP	2.79	2.79	3.79	0.93	0.93	1.26
155 WP	4.13	4.13	5.62	1.37	1.37	1.87
4.2 in. WP	4.96	4.96	5.56	1.65	1.65	1.87
81mm WP	2.24	2.24	3.05	0.74	0.74	1.01
5 in. WP	-	-	-	-	-	-

3.2.5 Light Attenuation and Extinction Coefficient. The fractional loss of light intensity and the extinction coefficient are assumed to be adequately related by the Beer-Lambert Law. When the threshold level of attenuation and the extinction coefficient for a particular smoke agent are used in this relationship, the threshold value of smoke concentration times length is obtained. This value, noted as CLTHRS in the model, is the model's obscuration criterion and is determined as a function of the selected inputs. In fact, the ability of the sensor to "see" through the cloud of a particular obscurant depends on its ability to defeat its corresponding concentration - length threshold. The fixed inputs of attenuation threshold and extinction coefficients used in the model are given in Table 7 for the visual range, for an anti-tank guided missile (ATGM) and for an operating set of night sights.

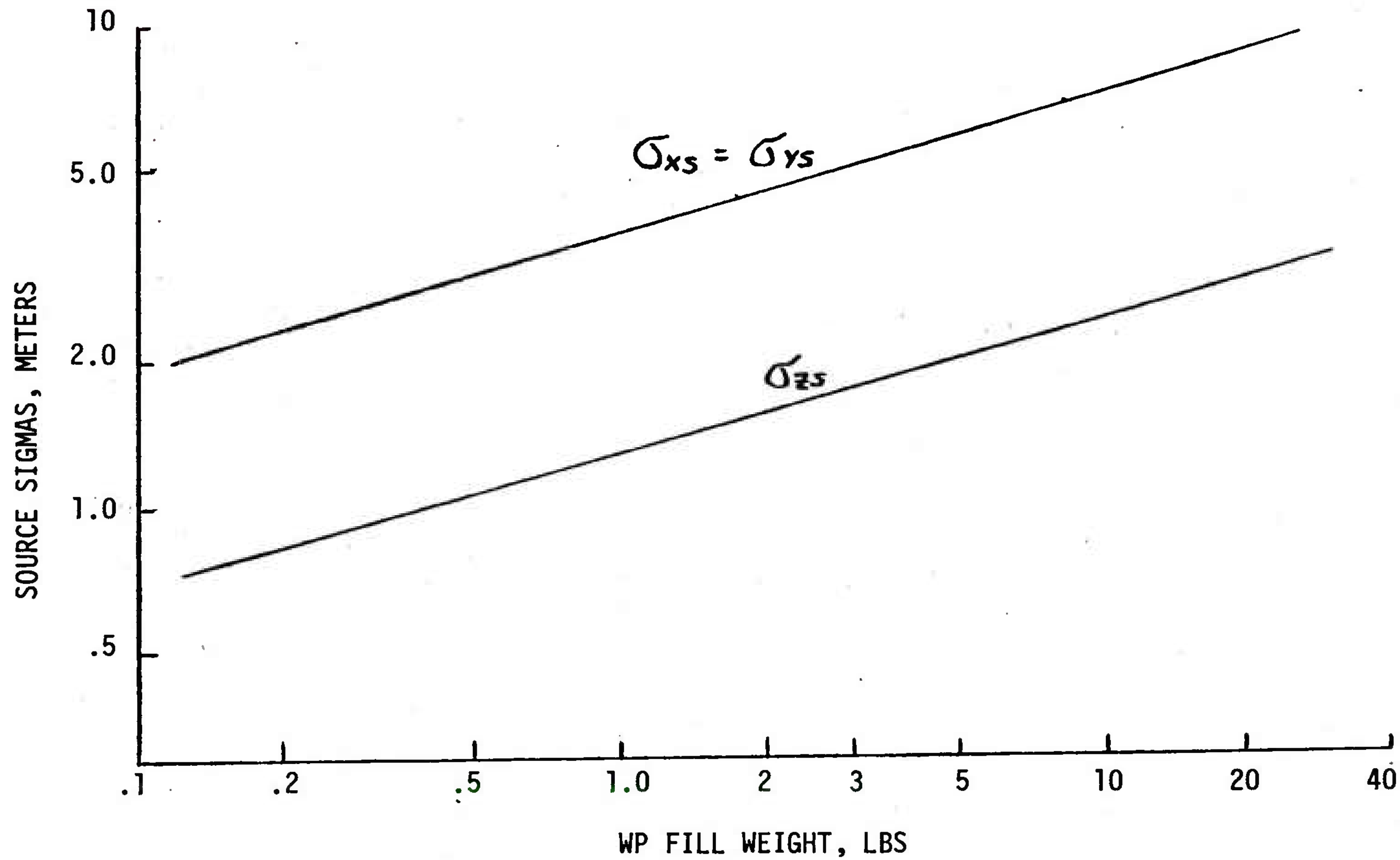


FIGURE 7 - SOURCE SIGMAS VS FILL WEIGHT

TABLE 7. ATTENUATION THRESHOLD AND EXTINCTION COEFFICIENTS

	VISIBLE	ATGM	NIGHT SIGHT
ATTEN. THRES.	0.90	0.99	0.95
WP	0.00332	0.00267	0.00028
HC	0.0045	0.0029	0.000061

3.2.6 Munition Delivery Accuracy. The smoke tables are constructed for munition delivery accuracies corresponding to two-thirds of the delivery weapon's maximum range. Representative values of the aiming and precision errors for the cannon and mortar weapons covered in the manual are listed in Table 8. The aiming errors represent adjusted fire.

TABLE 8. AIMING AND PRECISION ERRORS

	Range (m)	Precision, Std. Dev.		Aiming, Std. Dev.	
		Range	Defl.	Range	Defl.
105 How.	7000	23	6	36	27
155 How.	12000	44	10	46	15
4.2 In. Mort.	4000	43	5	49	27
81mm Mort.	5000	37	6	45	27
5 In. Gun	16000	-	-	-	-

3.2.7 Aimpoints. The intended center of impact for the volley located from the center of the firing weapons as the origin of coordinates, is designated as YT in the range direction and XT in the lateral direction. The values for YT correspond to the weapon ranges quoted in Section 3.2.6 while XT is taken as zero. The location of the intended impact points about the intended center of impact taken as the origin, are noted as X1DEAL or X(I) in the lateral direction and YIDEAL or Y(I) in the range direction. These points also establish the desired impact pattern. Table 9 gives the intended impact points for some of the volley sizes used in constructing the tables in the Manual.

TABLE 9. INTENDED IMPACT POINTS FOR VARIOUS MUNITIONS AND VOLLEY SIZES

	Firing Pattern	Volley Size	XIDEAL (m)	YIDEAL (m)
105 How.	Lazy W	6	-75	-15
			-45	15
			-15	0
			15	0
			45	15
			75	-15
155 How.	Lazy W	6	-125	-25
			- 75	25
			- 25	0
			25	0
			75	25
			125	-25
4.2 In. Mort.	Line	4	- 60	20
			- 20	0
			20	20
			60	0
81mm Mort.	Line	3	- 30	0
			0	0
			30	0
5 In. Gun	Converge Sheaf	2	0	0
			0	0

3.2.8 Reliability. The reliability of all munitions is assumed to be 100 percent. For a round reliability other than 100 percent, the model would produce an average cloud length. For small volley sizes, say a one round volley, an average value would not be realized since a full cloud either would or would not occur.

3.2.9 Burn Time. The burn time for all HC rounds is assumed to be 2 minutes. This is in accordance with field test data. (Reference 4).

4. SAMPLE RESULTS

Sample calculations were made for a battery of 155mm weapons firing in a cross wind of 5 knots under neutral atmospheric conditions. Table 10 shows the results of firing a single volley of 2, 4, and 6 rounds of WP and HC ammunition. The battery fire has been adjusted thereby minimizing the aiming errors. The cloud characteristics are defined as follows:

(a) Total cloud length - The included distance between the extreme left and right obscured LOS from the observer's position.

(b) Effective cloud length - The total cloud length less the sum of the included "hole" lengths.

(c) Upwind edge of cloud - The distance of the upwind edge of the cloud from the intended volley aimpoint (centroid). This distance becomes negative when the upwind edge crosses the aimpoint.

(d) Downwind edge of cloud - The distance of the downwind edge of the cloud from the intended volley aimpoint. This distance is positive when the downwind edge is downwind of the aimpoint.

The two-round volleys are fired at aimpoints 3 and 4 of the Lazy W shown in Figure 1 while the four-round volleys use the outer aimpoints 1, 2, 5 and 6. These aimpoints were chosen to illustrate the results obtained using close and separated bursts and are not necessarily those to be used in actual firings. The lateral spacing between each sequential aimpoint is 50 meters, all of which are located at two-thirds of the weapons' maximum range from the firing position.

Referring to the WP two-round volley data in Table 10, the closeness of the aimpoints results in a cloud with an initially small hole between the two bursts. This is indicated by the difference between the total and effective lengths. The total obscuring length increases to 202 meters in 3.5 minutes after which the cloud thins out and the obscuring length starts to shorten.

The 150-meter spacing between the number 2 and 5 aimpoints for the four-round volley creates a larger separation for the corresponding bursts. Results in Table 10 show that the accumulated hole size starts off at about 100 meters in size at 30 seconds after impact and closes almost completely in 4 minutes. Meanwhile, the effects of diffusion causes the ends to thin out so that at 3 minutes after round impact a shortening of the effective length occurs.

The six-round volley increases the effects of the four-round volley by the addition of the two middle weapons of the battery. Since the outer weapons are fired in both cases, a similar cloud length would be expected but with a smaller accumulated hole size due to the filling effect of weapons 3 and 4. Table 10 shows this to be the case, the hole size being only a few meters in length throughout the cloud's time history. The end point displacements, or distances covered by the screens, are expected to be the same whether four or six guns are firing.

TABLE 10 RESULTS OF FIRING VARIOUS 155MM SMOKE ROUNDS

RANGE: 12,000 METERS

BATTERY FORMATION: LAZY W

WIND DIRECTION: CROSS WIND

METEOROLOGICAL CONDITION: NEUTRAL - 5 KNOTS

OPTICAL DEVICE: VISUAL

RATE OF FIRE: SINGLE VOLLEY

	(TIME AFTER IMPACT (SECONDS)							
	30	60	90	120	150	180	210	240
WP: 2 ROUNDS								
TOTAL CLOUD LENGTH	104	129	148	164	179	193	202	197
EFFECTIVE CLOUD LENGTH	102	129	140	164	179	193	202	197
UPWIND EDGE OF CLOUD	-26	191	-160	-230	-302	-373	-446	-521
DOWNWIND EDGE OF CLOUD	129	220	308	394	481	566	010	718
WP: 4 ROUNDS								
TOTAL CLOUD LENGTH	331	356	375	391	404	405	387	336
EFFECTIVE CLOUD LENGTH	227	279	321	356	379	387	373	333
UPWIND EDGE OF CLOUD	84	9	- 65	-141	-215	-291	-366	-443
DOWNWIND EDGE OF CLOUD	246	346	439	632	620	696	753	779
WP: 6 ROUNDS								
TOTAL CLOUD LENGTH	317	343	362	380	399	409	399	360
EFFECTIVE CLOUD LENGTH	301	337	358	377	397	407	397	359
UPWIND EDGE OF CLOUD	78	6	- 66	-138	-209	-232	-353	-426
DOWNWIND EDGE OF CLOUD	239	335	427	510	600	691	752	786
HC: 2 ROUNDS								
TOTAL CLOUD LENGTH	187	246	325	404	416	415	413	411
EFFECTIVE CLOUD LENGTH	187	246	325	404	416	415	413	411
UPWIND EDGE OF CLOUD	39	39	39	39	- 29	-112	-196	-276
DOWNWIND EDGE OF CLOUD	126	206	204	363	445	527	609	686
HC: 4 ROUNDS								
TOTAL CLOUD LENGTH	395	478	561	643	652	640	616	573
EFFECTIVE CLOUD LENGTH	369	478	561	643	652	640	616	573
UPWIND EDGE OF CLOUD	148	148	148	148	76	- 14	-106	-190
DOWNWIND EDGE OF CLOUD	245	329	411	493	576	654	721	763
HC: 6 ROUNDS								
TOTAL CLOUD LENGTH	405	491	5786	661	673	662	634	591
EFFECTIVE CLOUD LENGTH	404	491	576	661	673	662	634	591
UPWIND EDGE OF CLOUD	152	152	152	152	78	- 9	- 90	-102
DOWNWIND EDGE OF CLOUD	251	337	423	508	594	671	732	773

Minus sign indicates upwind edge of cloud has passed through the aimpoint.

The HC burst points are a continuous source of smoke until burnout occurs at the end of 2 minutes. At this time the upwind edge of the cloud starts to move downwind. No holes are indicated after 30 seconds of burn time. Since the four- and six-round volleys fire the same outer weapons of the battery, little or no advantage is gained by firing the six rounds for the 240 seconds indicated in the Table.

NOTE: Data used in calculating results for the HC agent have been updated since this table was prepared and are included in Section 3 of this report.

5. COMPARISON BETWEEN MODEL AND TEST RESULTS

Between September and November 1977, Dugway Proving Ground personnel tested certain US inventory smoke munitions. These tests were requested by AMSAA and funded by the Office of the Project Manager for Smoke and Obscurants. Dugway's report on these tests is given in Reference 3. These tests included most of the smoke munitions listed in the JMEM/ME Smoke Manual in addition to burn time data for HE munitions taken from wind tunnel tests. For these reasons they were used as a basis for comparison with results obtained from the Smoke Effectiveness Manual Model. Table 11 is a summary of the munitions and the test conditions from which data were selected for the comparative analysis.

5.1 Test Data.

In order to cover a wide range of volume of delivered smoke, test data from single- and multi-round volleys of 155mm and 105mm WP smoke projectiles were selected from Reference 3. The choice of WP rather than HC munitions for comparison between test and predicted results represents a more severe test of simulation due to the definitive nature of the WP burst and the individual burst histories of transport and diffusion which the model must consider. Table 12 includes test data required by the model as input for simulating the test results. These data are identified in the table which also lists computer instructions appropriate for handling this information. Finally, the table includes input parameters that are functions of the input conditions. The latter are given in Figures 5, 6, and 7 and are taken from Reference 2.

5.2 Model Changes for Comparative Analysis.

A minor alteration of the WP program of the smoke model was necessary to produce results for comparison with test data. This change is shown in Appendix D Flow charts titled, Subroutine CALC, Modification for Comparative Analysis. Essentially, this change determines the value of concentration-length, CL, of the smoke cloud along a fixed observer-to-target line-of-sight. These CL's are recorded versus time from burst.

TABLE 11: SUMMARY OF TESTS FOR MODEL VALIDATION*

Trial No.	Type Smoke	Caliber (mm)	Type Munition	No. Fired	Wind Direction Deg.	Temp. Diff. ΔT °F	Rel. Humidity (%)	Pasquill Cat.
1R1	HC	105	M84A1	1	183	+1.35	17	D
2	HC	105	M84A1	3	152	+1.35	25	D
3	HC	105	M84A1	6	123	+1.35	19	C
5	WP	105	M60A2	1	271	+2.35	18	C
6	WP	105	M60A2	3	183	+3.1	22	C
7	WP	105	M60A2	6	353	+2.2	41	C
21	WP	155	M110E2	1	200	+3.2	25	C
22	WP	155	M110E2	3	181	+3.1	24	C
23	WP	155	M110E2	6	318	0	54	C
25	HC	155	M116B1	1	325	+1.5	17	B
26	HC	155	M116B1	3	173	+1.5	18	C

*REF: Inventory Smoke Munition Test (Phase IIa) Final Test Report, Report No. DPG-FR-77-314, June 1978.

TABLE 12. MODEL INPUT PARAMETERS

WP MUNITIONS

Trial No.	5	6	7	21	22	23
● U m/sec	-3.9	4.3	-4.4	4.6	5.0	-3.3
ALPHA=BETA	.62	.60	.65	.60	.60	.90
WVU1	1.0	1.0	1.0	1.0	1.0	1.0
WVU2	0	0	0	0	0	0
YIELD	4.2	4.4	4.8	4.4	4.4	5.3
CATTN(1R)	.00332	.00332	.00332	.00332	.00332	.00332
THRES(1R)	.90	.90	.90	.90	.90	.90
● QMUN (MG)	$.182\times10^6$	$.182\times10^6$	$.182\times10^6$	$.706\times10^6$	$.706\times10^6$	$.706\times10^6$
SIGXS(m)	5.3	5.3	5.3	7.9	7.9	7.9
SIGYS(m)	5.3	5.3	5.3	7.9	7.9	7.9
SIGZS(m)	1.9	1.9	1.9	2.62	2.62	2.62
● XIDEAL(1)	76.2	-76.2	30.5	-76.2	-76.2	76.2
● YIDEAL(1)	0	10	25	0	10	25
● ZIDEAL(1)	0	0	0	0	0	0
● XIDEAL(2)	-	-76.2	30.5	-	-76.2	76.2
● YIDEAL(2)	-	0	15	-	0	15
● ZIDEAL(2)	-	0	0	-	0	0
● XIDEAL(3)		-76.2	30.5	-	-76.2	76.2
● YIDEAL(3)	-	-10	5	-	-10	5

°Notes Test Data

TABLE 12. (CONTINUED)

TRIAL NO.	5	6	7	21	22	23
● ZIDEAL(3)	-	0	0	-	0	0
● XIDEAL(4)	-	-	30.5	-	-	76.2
● YIDEAL(4)	-	-	-5	-	-	-5
● ZIDEAL(4)	-	-	0	-	-	0
● XIDEAL(5)	-	-	30.5	-	-	76.2
● YIDEAL(5)	-	-	-15	-	-	-15
● ZIDEAL(5)	-	-	0	-	-	0
● XIDEAL(6)	-	-	30.5	-	-	76.2
● YIDEAL(6)	-	-	-25	-	-	-25
● ZIDEAL(6)	-	-	0	-	-	0
NDEVIC	1	1	1	1	1	1
● NVOL	1	1	1	1	1	1
● NRPV	1	3	6	1	3	6
NT	30	30	30	40	40	40
DT (SEC)	3	3	3	5	5	5
DL	10	10	10	10	10	10
NSAMP	1	1	1	1	1	1
VT (SEC)	400	400	400	400	400	400
SNAP (SEC)	3	3	3	5	5	5

°Notes Test Data

TABLE 12. (CONTINUED)

TRIAL NO.	5	6	7	21	22	23
● SIGBR(m)	0	0	0	0	0	0
● SIGBR(m)	0	0	0	0	0	0
● SIGAR(m)	0	0	0	0	0	0
● SIGAD(m)	0	0	0	0	0	0
● REL	1.0	1.0	1.0	1.0	1.0	1.0
YTLINE	1000	1000	1000	1000	1000	1000
CUTOFL	-1000	-1000	-1000	-1000	-1000	-1000
CUTOFR	1000	1000	1000	1000	1000	1000
YT	500	500	500	500	500	500
XT	0	0	0	0	0	0
XINC	10	10	10	10	10	10

●Notes Test Data

5.3 Comparisons Between Model and Test Results.

The WP portion of the smoke model was run using test data obtained from the Smoke Inventory Tests, Reference 3. In order to get a broad view of the model's predictive capabilities, the tests selected for comparative analysis consisted of those employing one and three rounds of 105mm WP M60A2 munitions and three and six rounds of 155mm WP M110E2 munitions. Each group of rounds were statically fired simultaneously in a linear array parallel to the observer's line of sight. In this way the smoke cloud passing the observer's LOS varied considerably in concentration-length from test to test rather than in screen length. A time history of smoke concentration was recorded as the cloud passed the observer's LOS and a comparison of this quantity was made with the corresponding predicted value obtained from the smoke model. Figure 8 shows the predicted and test results when the smoke from one round of 105mm WP is released and crosses the observers LOS. The concentration length represents the mass of smoke occupying a volume one square meter in cross section and extending along the observer's LOS. The dashed and solid markers indicate the level of concentrations needed to defeat the naked eye and an anti-tank guided missile (ATGM) sensor respectively. The corresponding defeat level for night sight is above the concentration levels shown. Comparisons between model and test results at the threshold levels indicated is quite favorable. A characteristic of most test and model comparisons is the tail-off of smoke that persists after the main body of smoke has passed downwind. This effect is apparently caused by the filler material containing solid chunks which do not form the initial flash but continue to burn on contact with the ground. This effect does not significantly degrade the fidelity of the model until larger volleys of the larger 155mm munition are involved. In such cases, the model gives conservative results. Figure 9 shows a similar presentation for three rounds of 105mm WP. A favorable comparison between model and test results is also apparent at the visual and ATGM thresholds. Here again, the concentration level to defeat the night sight is not reached. The threshold level for the night sight is finally exceeded with three rounds of 155mm WP. This is shown in Figure 10 along with the visual and ATGM threshold levels. The correlation is favorable, with the model being a little conservative at the lower concentrations due to the tail-off. Figure 11 shows comparative results for 6 rounds of 155mm WP. In this particular test the tail-off persists for a considerable length of time after the maximum smoke concentration level has passed.

From the comparisons made to date between model and test results, it appears that the model predicts the duration of obscuration for the three sensors quite well. Difficulties arise where the WP model predicts higher peak values of CL's and shorter duration times at lower CL's. Both effects are apparently due to some of the filler material not being consumed in the initial burst and burning on the ground in the manner of an HC munition. A model change has been prepared to correct this deficiency. This change will be documented at a later date.

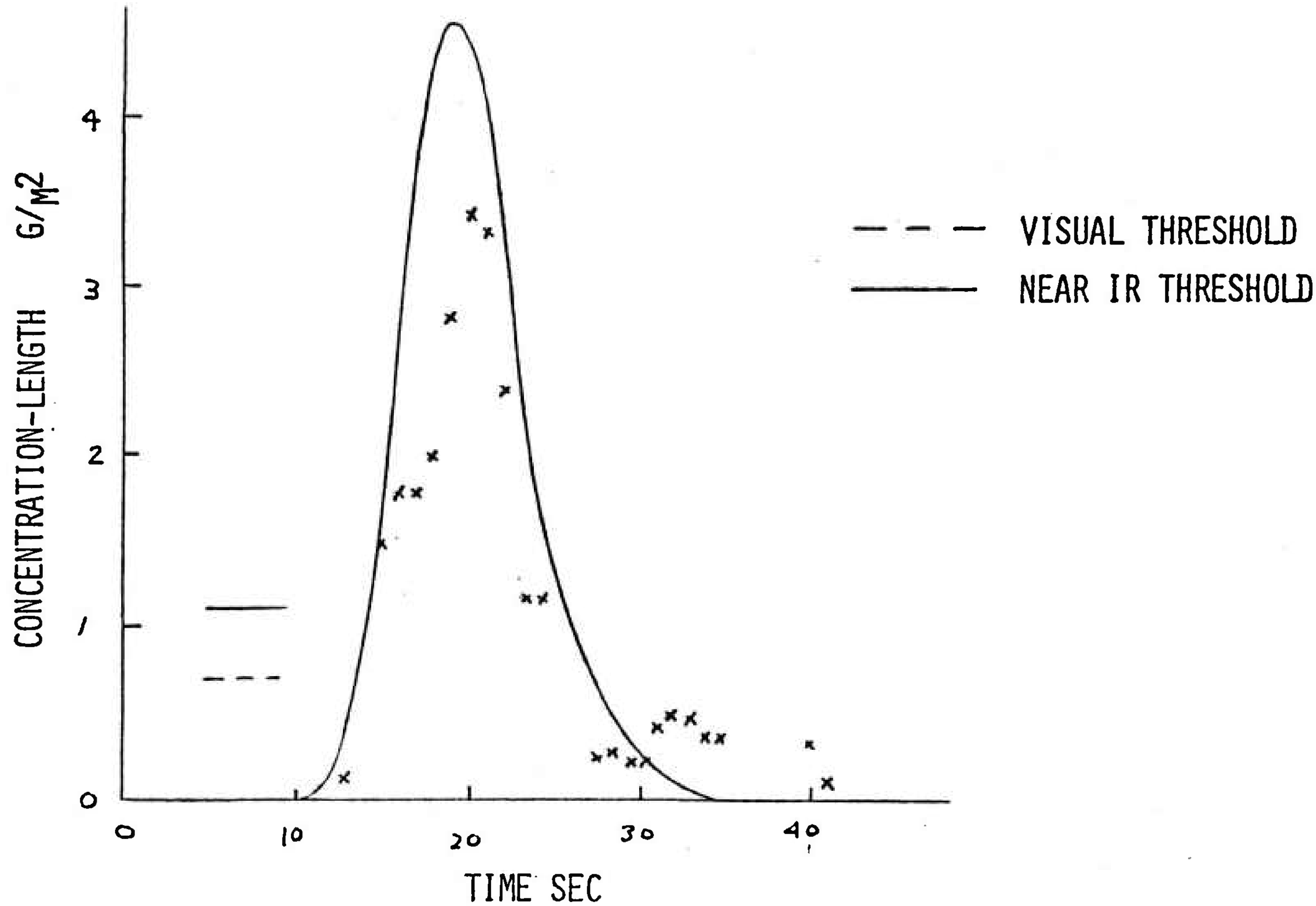


FIGURE 8 - PREDICTED VS. TEST RESULTS, ONE ROUND 105mm, WP

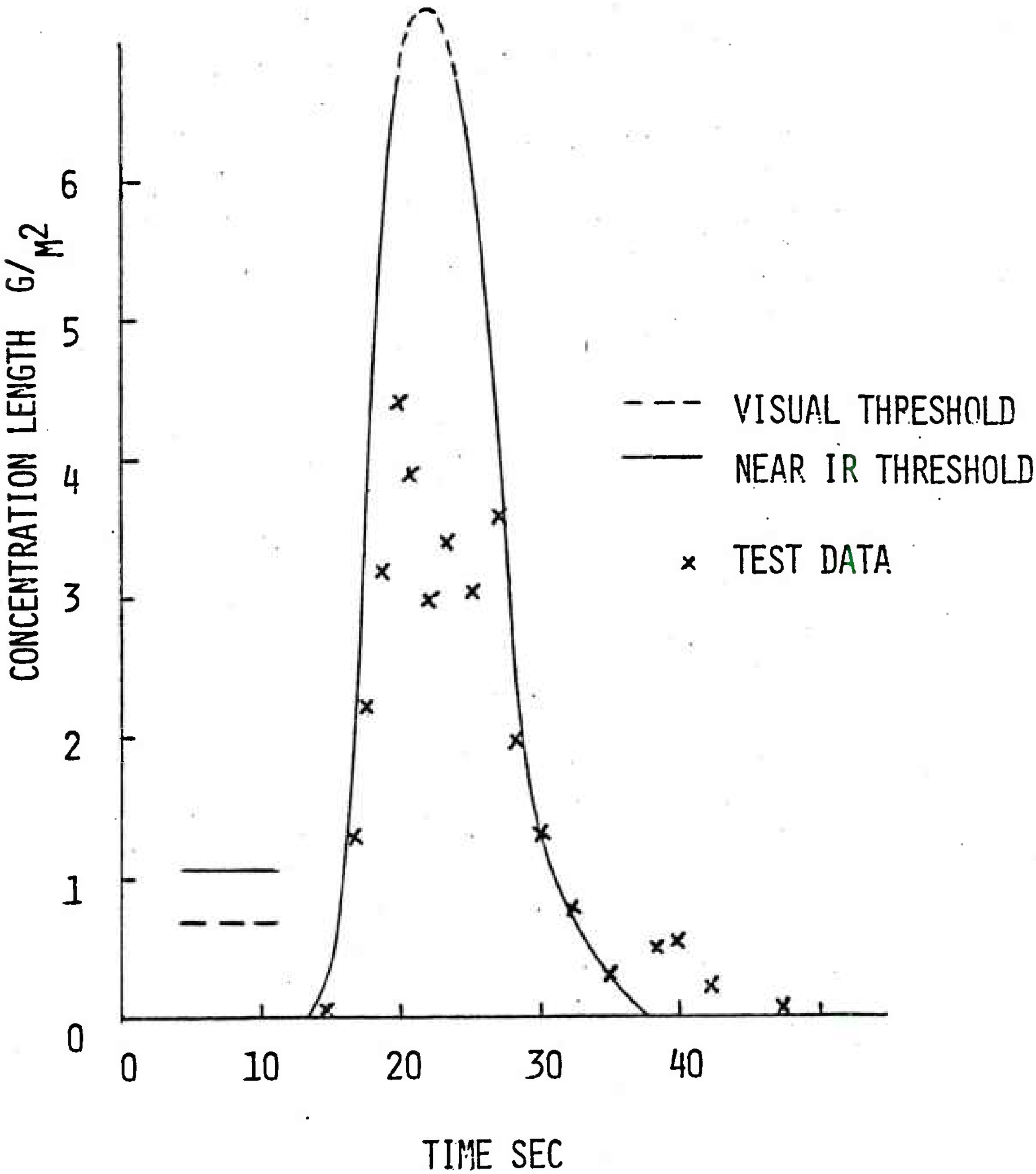


FIGURE 9 - PREDICTED VS TEST RESULTS,
3 ROUNDS 105mm, WP

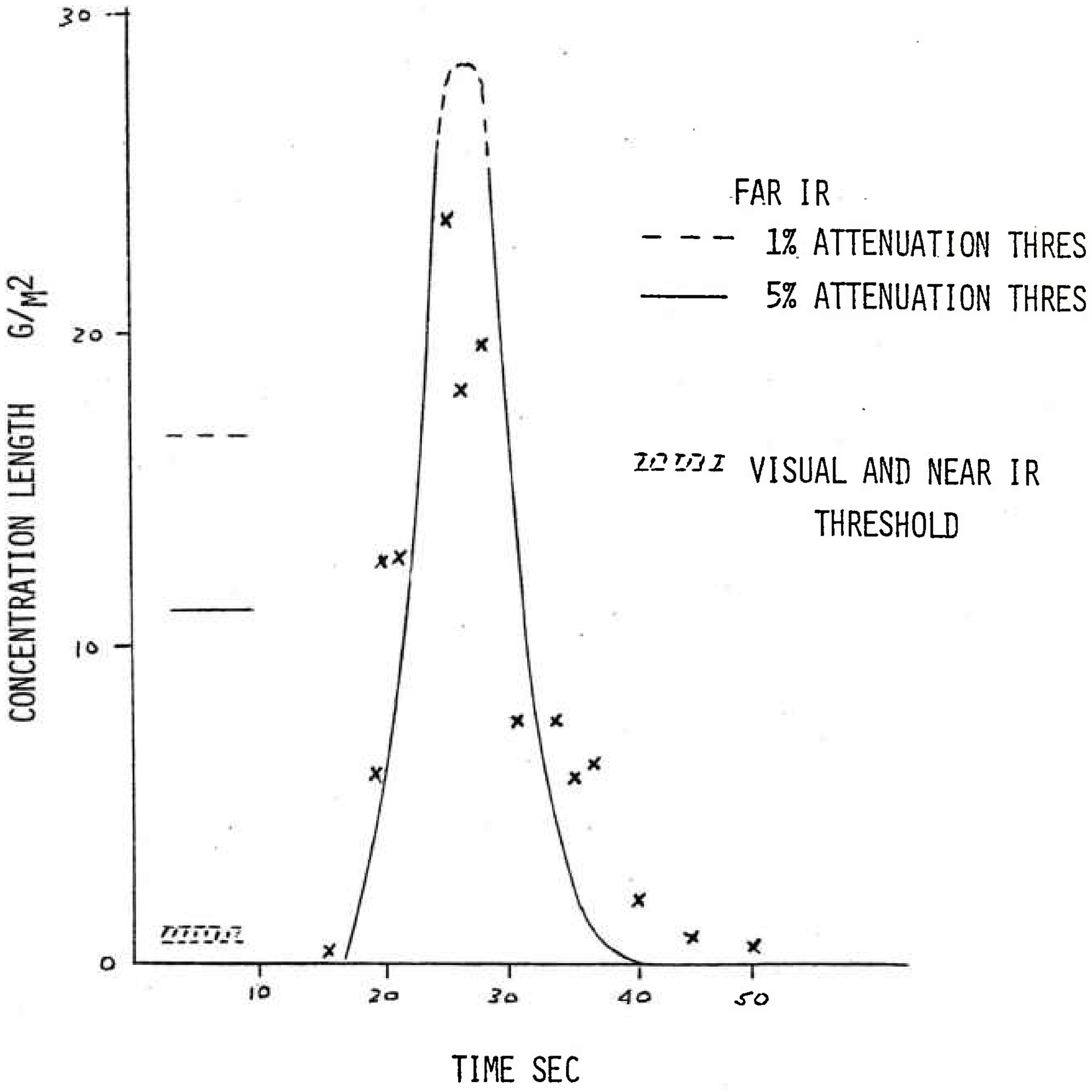


FIGURE 10 - PREDICTED VS TEST RESULTS

3 ROUNDS 155mm WP

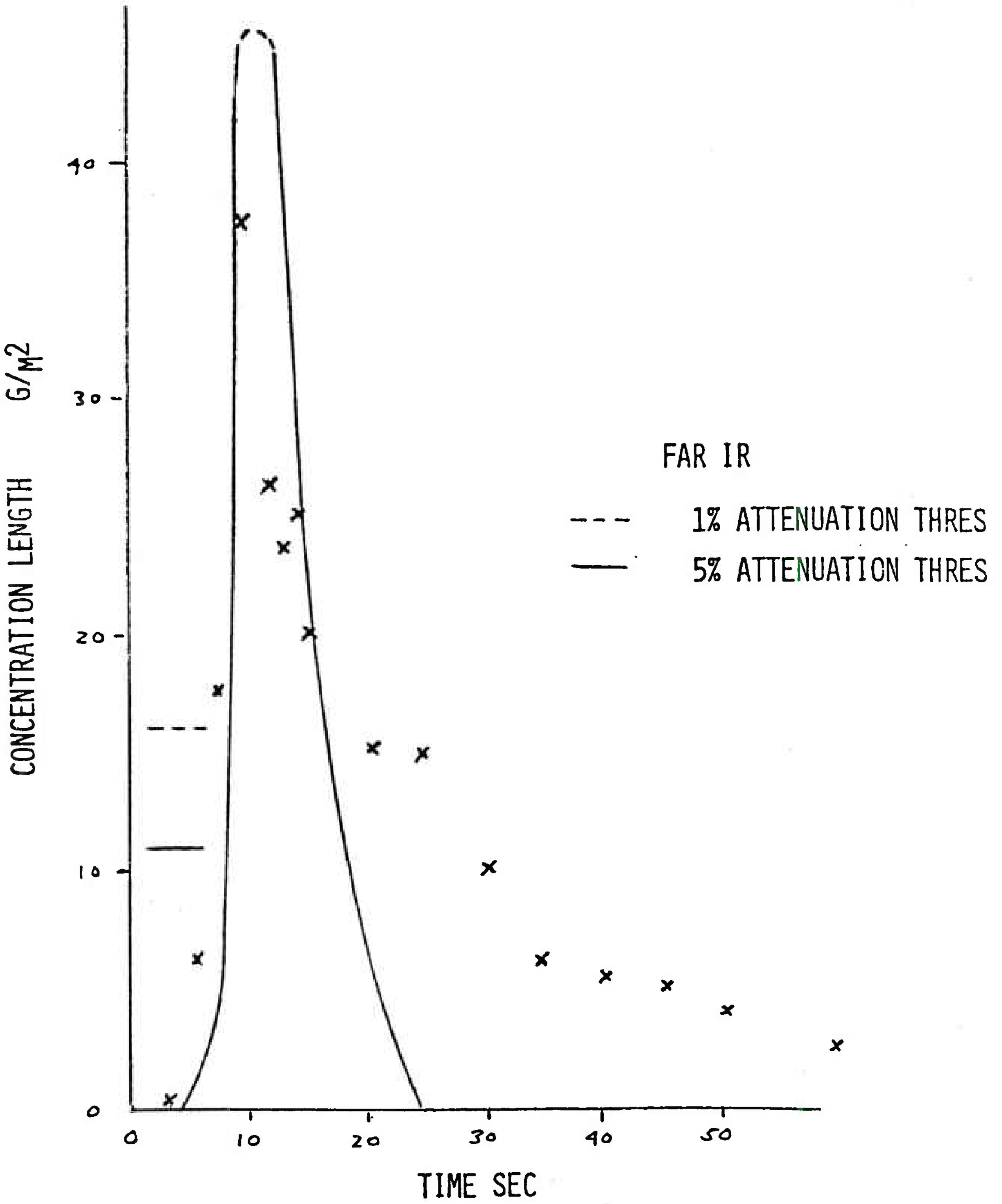


FIGURE 11 - PREDICTED VS TEST RESULTS,
6 ROUNDS 155mm WP

REFERENCES

1. Milly, George H. Atmospheric Diffusion and Generalized Munitions Expenditures, Operations Research Group Study No. 17, 1 May 1958. US Army Chemical Corps, Operations Research Group, Army Chemical Center, MD.

2. Johnson, Morris C. and Forney, Paul D., The Effectiveness of Obscuring Smokes, Operations Research Group Report, September 1972, US Army Munitions Command, Operations Research Group, Edgewood Arsenal, MD.

3. Inventory Smoke Munition Test (Phase IIa), Final Test Report, Volume I, DPG-FR-77-315, June 1978.

4. Basic Smoke Characterization Test, Final Report, DPG-TP-77-311, March 1978.

Next page is blank.

APPENDIX A

DEFINITION OF SYMBOLS

The input quantities required to operate the smoke model are listed and defined below.

ANG	The angle between the earth axis and the weapon axes system, degrees. (Taken as zero in the program listing).
CATTN	The attenuation coefficient used in the Beer-Lambert Law. Function of the smoke agent and transmitted wavelength.
DL	Line increment, meters.
DT	DT is the time between computer computations or steps, seconds.
EFF	Defined in Section 3.3.2, Munition Fill Weight.
INSNAP	Time between printouts (SNAP) divided by time between computations (DT).
NRPV	The number of rounds per volley to be fired.
NS	The total number of computer time steps in the program.
NSAMM	The number of "holes" or see through locations within the smoke screen.
NSAMP	Total number of replications for averaging results.
NT, MT	NT is the total number of times the cloud history is examined, or looked at each time noted as MT, and separated by SNAP seconds.
NVOL	The number of volleys to be fired.
QMUN	Defined in Section 3.2.3, Munition Fill Weight.
SNAP	Time increment between printouts or "looks", i.e., between MT and MT+1, seconds.
THRES	The percent of light leaving the target which is required to accomplish detection.
U	The wind velocity in meters per second.

VT	Time between volleys, seconds.
WVU1, WVU2	The direction cosines between the earth axes and the wind axes.
X_{OB}, Y_{OB}	The observer location in the earth axes system (usually 0,0) meters.
XIDEAL(J) YIDEAL(J) ZIDEAL(J)	The aimpoints of all J rounds to be fired, measured from the volley aimpoint or desired pattern center.
XT, YT	The location of the aimpoint for the volley in the weapon axes system, meters.
YIELD	Defined in Section 3.2.1.
α, β, σ	The exponents defining the standard deviations of the smoke cloud. Functions of temperature gradient.
σ_{AR}, σ_{AD}	The occasion-to-occasion aiming error in range and deflection, meters.
σ_{BR}, σ_{BD}	The round-to-round precision error in range and deflection, meters.
σ_{xs}, σ_{sy} σ_{zs}	The source sigmas, meters. Establish the initial size of the smoke burst; a function of fill weight.

APPENDIX B

DESCRIPTIONS OF SUBROUTINES FOR WP MUNITIONS

B.1 GENERAL

The subroutines for the WP program are described in this Appendix using a combination of algebraic and FORTRAN notation. The order of presentation is essentially that in which they are called for in the MAIN listing. Flow charts for the WP program are in Appendix D.

B.2 WP PROGRAM, MAIN

The function of MAIN is to call sixteen subroutines which make up the WP smoke model. In addition, the calculation of the threshold value of smoke CL for a given visual aid is computed here in accordance with the Beer-Lambert Law. Simply:

$$dI/dCL = CATTN \times I$$

$$TRANS = 1 - ATTN = I/I_0 = \exp(-CATTN * CL)$$

$$ATTN = 1 - \exp(-CATTN \times CL) \quad (6)$$

where I/I_0 is the ratio of the transmitted to the incident radiation intensity.

CATTN is the extinction coefficient for the smoke agent.

TRANS, ATTN are the transmission and attenuation of radiation through the smoke, respectivity.

B.3 SUBROUTINE STCL

Subroutine STCL calculates the virtual offset distances A, B, C which are needed to define the initial burst sigmas to be used in Subroutine FRMCLD. It also establishes the weight of smoke produced from the selected smoke agent through the calculation of the quantity FACTOR.

B.4 SUBROUTINE FRMCLD

Subroutine FRMCLD computes the sigma time histories for all the bursts which make up the smoke cloud. The increase of these sigmas with time is the diffusion mechanism which creates the cloud growth. The quantity FACT is the smoke density written in the form necessary for inclusion in the gaussian formulation.

B.5 SUBROUTINE CLEAR

Subroutine CLEAR is employed to set to zero the initial values of the statistical variables which are to be evaluated in Subroutine VEVAL. This step forms part of the Monte Carlo averaging process to be preformed in Subroutine CALPRT which follows VEVAL. The quantities to be initially set to zero are:

Cloud length	SSL
Cloud length, squared	SSLS
Left end of cloud	SLY
Left end of cloud, squared	SLYS
Right end of Cloud	SRY
Right end of cloud, squared	SRYS
Hole size	SSH
Hole size, squared	SSHS

B.6 SUBROUTINE MPLACE.

Subroutine MPLACE locates each burst or impact point of the battery volley. The aiming and precision errors are computed and taken from Subroutines NORAN. The sequence of burst placement is as follows: For volley 1 at time T_1, weapon 1 fires round 1 at aimpoint 1; weapon 2 fires round 2 at aimpoint 2; and so on until all weapons have fired. For volley 2 at time t_2, the weapon firing is repeated. This sequence continues until all volleys are fired. The volley number uses the index I, the weapon or round number uses the index J. Each of the NSAMP replications called in the MAIN program constitutes a different firing occasion.

Figure 12 shows the development of the impact points diagramatically. The weapon axes can be displaced from the reference earth axes by an angle ANG. In the weapon axes system, the target is located at point X_T, Y_T, which is also the centroid of the volley aimpoints. On one occasion, the centroid displacement due to the aiming error is

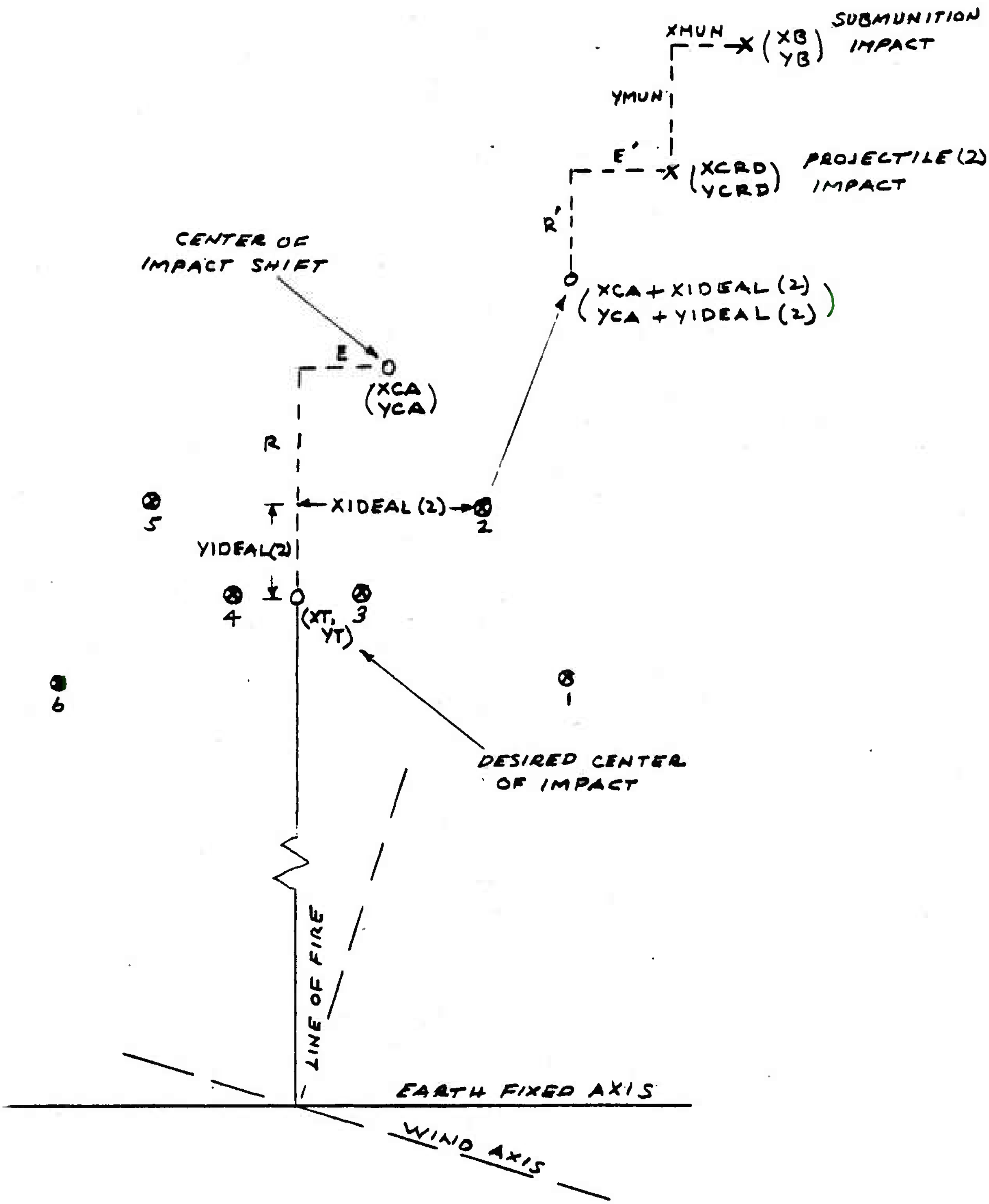


FIGURE 12 - DEVELOPMENT OF AIMING, PRECISION AND SUBMUNITION IMPACT ERRORS

determined by Subroutine NORAN and located at position XCA, YCA. The impacts of each of the volley rounds are displaced from XCA, YCA by their aimpoint shifts and by their individual precision errors. The latter is also determined by Subroutine NORAN. The final locations of each round is XBA(1), YBA(1); XBA(2), YBA(2); etc. Normally ZBA is zero. The time each volley is delivered is given by

$$BRSTTM = (I-1) * VT$$

where I is the volley number and VT the time between volleys.

The weapon axes, which define the line of fire, are usually selected coincident with the earth axes in which case ANG is zero as in Figure 12.

B.7 SUBROUTINE NORAN

Subroutine NORAN determines the aiming error for each occasion or program replication and the precision error for each round delivered.

B.8 SUBROUTINE COMBST

Subroutine COMBST transforms the impact or burst points XBA(I), YBA(I) computed in subroutine MPLACE into the wind coordinate system with an origin coincident with that of the earth fixed system and an X-axis in the wind direction. In this subroutine the index I covers all rounds fired, WVU1 and WVU2 are the direction cosines between axes. The choice of wind axes is made to facilitate the integration of smoke densities along selected lines of sight in Subroutine LCON.

B.9 SUBROUTINE TIME

Subroutine TIME is executed at each successive period MT, selected in MAIN, for a total of NT periods. At each period, the quantity AGE represents the age of all volleys or bursts in the program. The function MUNGRP is the chronological order of these volleys of bursts and is used as an index for identifying successive values of burst sigmas. The location of each burst at period MT is calculated in the earth axes system and given as XCENT(I), YCENT(I), where the index I is the burst number. XCENTP(I) is the location of the burst point in the wind axes system. Finally, the maximum component $\sigma_{XA}(I)$, $\sigma_{YA}(I)$ of each burst after resolution into the wind axes system is calculated for each program time interval, MUNGRP.

B.10 SUBROUTINE SIZE

Subroutine SIZE establishes the boundaries of a rectangle, in the earth axes system, which encloses all bursts to be delivered. The rectangle dimensions are determined by examining the $\pm 4\sigma_{XA}$ and $\pm 4\sigma_{YA}$ values of these bursts and selecting maximum values. This rectangular area serves as an exploratory region for determining smoke concentrations, (see Figure 3).

B.11 SUBROUTINE MXMIN

Subroutine MXMIN determines a set of lateral limits y_1, y_2, which are used for establishing the array of observer "look" lines in Subroutine CALC. These "look" lines pass through the $\pm 3\sigma_{XA}$ coordinates of the extreme bursts of the cloud. Since the slope of these "look" lines is needed, the lateral points y_1, y_2, are calculated at a selected distance YTLINE downrange in the earth axes system.

B.12 SUBROUTINE MATCON

Subroutine MATCON takes the $\pm 4\sigma$ dimensions of the smoke rectangle from Subroutine SIZE and determines a 40 X 40 array of x-y coordinates within these limits. These coordinates are then resolved into the wind axes system. By calling Subroutine DENSTY, the concentration of smoke in milligrams per cubic meter at each coordinate is computed. This concentration is designated by the symbol CONMAT(I, J) which is the same as CON used in subroutine DENSTY.

B.13 SUBROUTINE DENSTY

Subroutine DENSTY determines the total concentration of obscurant in milligrams per cubic meter due to all bursts at each of the 40 x 40 array of points calculated in Subroutine MATCOM. The Mass concentration at each point is calculated by the expression given in Equation (2.3). The transport of the cloud is determined by the expression

$$X = XP - XCENTP$$

where XCENTP is the burst's centroid location measured in the wind axes system and obtained from Subroutine TIME. XP, YP, and ZP refer to the array of points obtained from Subroutine MATCON. No transport is developed in the YCENTP and ZCENTP directions which are normal to the wind velocity.

B.14 SUBROUTINE MATCL

Subroutine MATCL is a supplementary procedure for examining the concentration of smoke agent along numerous lines of sight passing through the exploratory rectangular area established in Subroutine SIZE. Two functions are performed in this Subroutine, they are:

a. Along the lower boundary of the rectangle, 40 points are selected each of which will be used as the source of a family of "look" lines extending to all of the 40 points on the upper boundary.

b. The source coordinates of each line and the line's direction cosines are transformed from the earth axes to the wind axes. After this, Subroutine LCON is called to determine the integrated concentration of the obscurant along each line. This is noted as CONLIN (I,J) in milligrams per square meter.

B.15 SUBROUTINE LCON

Subroutine LCON integrates the mass of obscurants along each "look" line, through the gaussian density distributions of each burst for each time period MT. The "look" lines are determined in Subroutine MATCL. Mass concentrations along specific "look" lines originating from a fixed observer position and determined in Subroutine CALC also employ LCON. The line concentration is given in milligrams per square meter and is computed in accordance with the following equation:

$$\text{TOTLNC} = \sum_{1}^{\text{NBURST}} \frac{Q\lambda\Omega}{\pi(\sigma_x(I)\ \sigma_y(I)\ \sigma_z(I)} * \frac{1}{A^{1/2}} \times \exp\left\{-\frac{1}{2}\left[\left(\frac{x+S*Dx-xcent}{\sigma_x(I)}\right)^2 + \left(\frac{y+S*Dy-ycent}{\sigma_y(I)}\right)^2 + \left(\frac{z+S*Dz-zB}{\sigma_z(I)}\right)^2\right]\right\}$$

where DX, DY and DZ are direction cosines and A and S are functions of the burst centroid location and its standard deviations and are defined in the program listing. Wind axes are used.

B.16 SUBROUTINE CALC

Subroutine CALC determines the visibility through all parts of the smoke cloud from an observer location XOBS, YOBS. Lines of sight are selected to various points incrementally spaced a distance ZINC apart and located at a distance YTLINE from the observer. Along each line, the mass of smoke agent is computed by calling Subroutine LCON. Each line of sight which permits observation through the cloud contributes a length of XINC to a "hole" size in that cloud. The total length of the cloud is defined as the distance between the extreme point through which obscurance first occurs and last occurs. Figure 13 shows the construction of the observer-cloud geometry treated in Subroutine CALC.

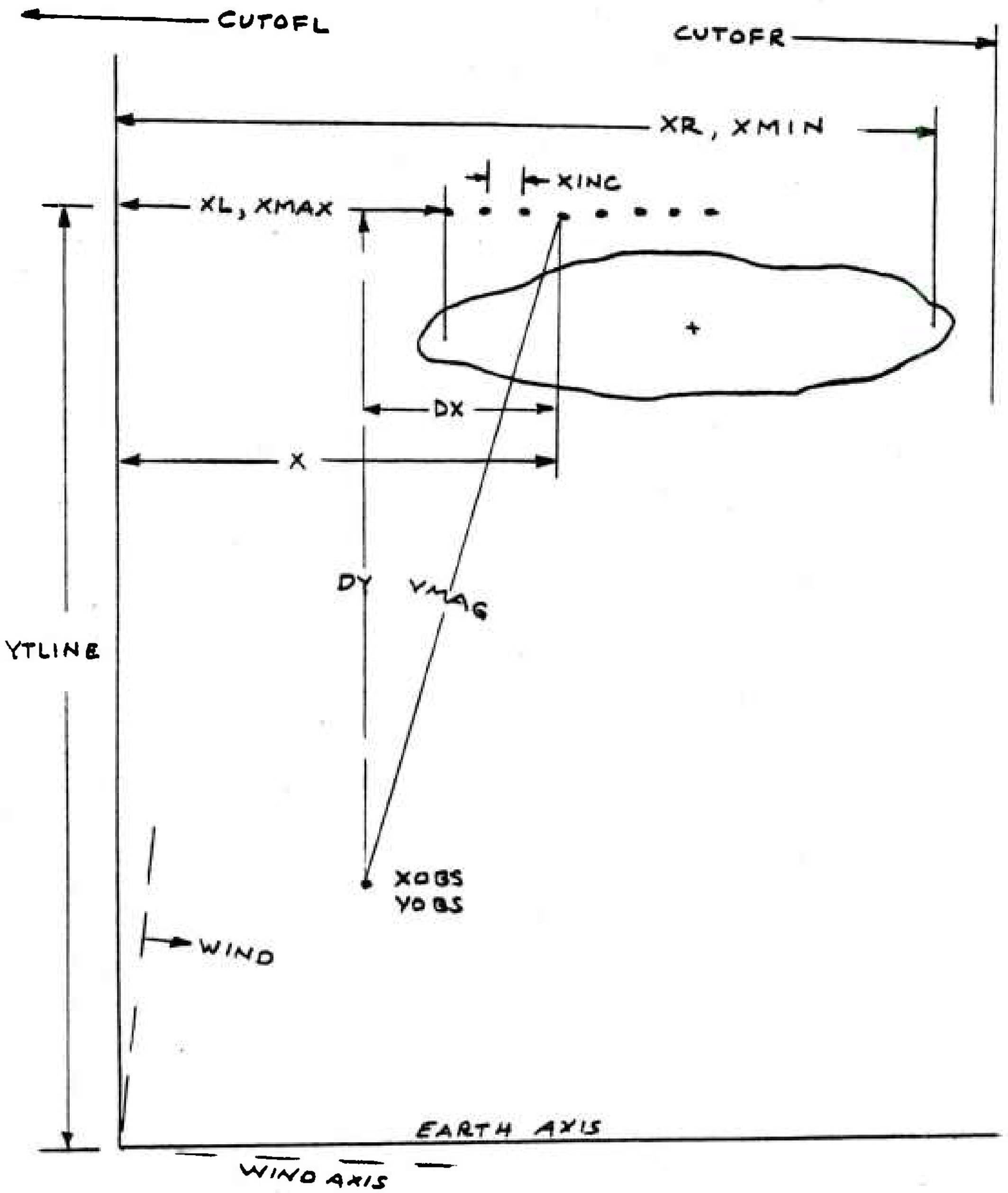


FIGURE 13 - ANALYSIS GEOMETRY AS USED IN SUBROUTING CALC

B.17 SUBROUTINE VEVAL

Subroutine VEVAL computes the left and right extremes of the obscuring screen LY, RY; the screen length, SLY; and the accumulated hole size, SH across the screen. In preparation for statistical averaging, the summation of the square of these quantities is determined for specific periods MT for each replication of the program KI. The initialization of these quantities has already been established in Subroutine CLEAR. The quantity NSAMM, which defines the number of times no screen is detected, is also calculated.

B.18 SUBROUTINE CALPRT

Subroutine CALPRT sums up the main results of the program by averaging the screen length, SSL; "hole" size, SSH; and the left and right coordinates of the cloud, SLY and SRY respectively, at each specific period MT for a total of KI replications of the program.

APPENDIX C

DESCRIPTION OF SUBROUTINES FOR HC MUNITIONS

C.1 GENERAL

The subroutine for HC munitions are similar to those for WP munitions. There are two significant differences. First, the HC smoke cloud is defined as a series of small periodic bursts or puffs originating at the point of the munition's impact. The number of bursts and the elapsed time between them is defined as NPUFTL and TINCR respectively. Representative values for these quantities used in the past are 120 bursts at 1 second intervals. Secondly, for those cases where the munition contains a number of submunitions, the latter are dispersed radially in accordance with a selected non-linear distribution and dispersed directionally according to a uniform distribution. Flow charts for the HC program are given in Appendix D.

C.2 HC PROGRAM, MAIN

MAIN HC contains essentially the same subroutines as MAIN WP. These similarities include provisions for calculating the concentration length thresholds for various sensors (NDEVIC) so that more than one device can be handled in one run. Also the earth fixed axes is assumed to be aligned along the line of fire of the delivery weapon thereby eliminating the weapon axes system.

C.3 SUBROUTINE STCL

This subroutine is also similar to that of the WP program except that the fill weight per puff, rather than fill weight per munition, is considered. Also the time between puffs, TINCR, is identified.

C.4 SUBROUTINE CLEAR

As in the corresponding WP subroutine, CLEAR sets initial values of statistical variables to zero.

C.5 SUBROUTINE MPLACE

In addition to placing the munitions in accordance with firing accuracy, the HC subroutine also places the submunitions in accordance with the radial and directional distribution of these quantities. This subroutine calls NORAN and MUNDSP to identify all errors.

C.6 SUBROUTINE NORAN

As in the WP program, NORAN determines the aiming error and precision errors of each round delivered.

C.7 SUBROUTINE MUNDSP

Scatterable submunitions contained in certain projectiles are placed in accordance with radial and directional distribution functions. As already stated, Subroutine MUNDSP determines these locations using a nonlinear and uniform distribution respectively. These displacements are used in Subroutine MPLACE along with the aiming and precision errors from Subroutine NORAN to define the impact points for all submunitions.

C.8 SUBROUTINE CONBST

This subroutine is identical to that of the WP program. The impact points of the submunitions are transformed from the earth axes to the wind axes system.

C.9 SUBROUTINE TIME

Subroutine TIME for HC munitions has been expanded over the corresponding Subroutine for WP smoke. As before, the displacement and diffusion of the puffs are computed as a function of time. In addition, the location and size of the first and last puff of each submunitions are determined in order to define the extremeties of the cloud in Subroutine SIZE.

C.10 SUBROUTINE CON

Subroutine CON, the location of the first and last puff from subroutine TIME are transformed to the earth axes system.

C.11 SUBROUTINE SIZE

Subroutine SIZE establishes the boundaries of an exploratory rectangle in the wind axis system. The rectangle is determined by the $\pm 4\sigma$ values of the extreme puffs considering all of the submunitions of the delivered load.

C.12 SUBROUTINE MXMIN

As in the corresponding subroutine for WP smoke, this program determines the lateral limits, now called XMIN and XMAX, used for establishing an array of observer look lines. These "look" lines are bounded by the $\pm 4\sigma$ coordinates of the extreme puffs determined from Subroutine SIZE. These lateral limits are calculated using their slopes to the origin, RATMN and RATMX.

C.13 SUBROUTINE MATCON

This Subroutine is identical to that for WP smoke. A 40 x 40 array of x-y coordinates are located within the limits of the exploratory rectangle taken from Subroutine SIZE. These points are then resolved into the wind axes system. By calling Subroutine DENSTY, the concentration of smoke in milligrams per cubic meter at each coordinate is computed and noted by the symbol CONMAT.

C.14 SUBROUTINE DENSTY

DENSTY calculates the displacement of any point, XP, YP, ZP in the wind axes system from the munition burst points. DENSTY then calls Subroutine DENM to determine the density of smoke at each of the desired points due to the individual effects of all puffs in the field.

C.15 SUBROUTINE DENM

The integration of the individual densities of all puffs in the field is performed by Subroutine DENM at the points of interest. The densities of all puffs at all points are computed in Subroutine DENPF.

C.16 SUBROUTINE DENPF

The density of smoke at any point x, y, z for each puff of each submunition is calculated by this subroutine.

C.17 SUBROUTINE MATCL

This subroutine is identical to the corresponding subroutine for the WP program. MATCL is a supplementary procedure for examining concentration of smoke agent along numerous lines of sight passing through the exploratory rectangular area established in Subroutine SIZE. Two functions are performed in this Subroutine, they are:

a. Along the lower boundary of the rectangle, 40 points are selected each of which will be used as the source of a family of "look" lines extending to all of the 40 points on the upper boundary.

b. The source coordinates of each line and the line's direction cosines are transformed from the earth axes to the wind axes. After this, Subroutine LCON is called to determine the integrated concentration of the obscurant along each line. This is noted as CONLIN (I,J) in milligrams per square meter.

C.18 SUBROUTINE LCON

LCON computes the displacement of the exploratory point x, y, z (called POX, POY, POZ) from the submunition burst point. Using this information and the slope of the LOS from the observer to that point, integration of all smoke densities along the designated line is performed by calling Subroutine LCONM. The concentration along the LOS is called TOTLNC in these Subroutines.

C.19 SUBROUTINE LCONM

Subroutine LCONM integrates through the puffs of each submunition or burst along the LOS defined by a point (POX, POY, POZ) and the slope through that point to the observers position. LCONM calls Subroutine LCONPF for supplementary calculations in carrying out the integration.

C.20 SUBROUTINE LCONPF

The total mass of smoke along the observer's LOS is determined by integrating through the Gaussian trivariate distribution of each puff in turn. LCONPF performs supplementary calculations for use in Subroutine LCONM.

C.21 SUBROUTINE CALC

Subroutine CALC for the HC program is identical to that for the WP program. CALC determines the visibility through all parts of the smoke cloud from an observer location XOBS, YOBS. Lines of sight are selected to various points incrementally spaced a distance XINC apart and located at a distance YTLINE from the observer. Along each line, the mass of smoke agent is computed by calling Subroutine LCON. Each line of sight which permits observation through the cloud contributes a length XINC to a "hole" size in that cloud. The total length of the cloud is defined as the distance between the extreme points through which obscurance first occurs and last occurs. Figure 13 shows the construction of the observer-cloud geometry treated in Subroutine CALC.

C.22 SUBROUTINE ENDPTS

ENDPTS compiles statistical information on the left and right extremes ($\pm 4\sigma$ values) of the exploratory rectangle defined in Subroutine SIZE. This information is used in Subroutine CALPRT to determine the average location and standard deviation of the upward and downwind boundaries of this rectangle at each time the smoke screen is examined (SNAP time). This information was meant for future use of the program and contributes little to any results as it is presently listed.

C.23 SUBROUTINE VEVAL

Subroutine VEVAL for the HC program is identical to that for the WP program. This subroutine computes the left and right extremes of the obscuring screen LY, RY; the screen length, SLY; and the accumulated hole size, SH across the screen. In preparation for statistical averaging, the summation of the square of these quantities is determined for specific periods MT for each replication of the program, KI. The initial quantities have already been established in Subroutine CLEAR. The quantity NSAMM, which defines the number of times no screen is detected, is also calculated and used in adjusting statistical computations in Subroutine CALPRT.

C.24 SUBROUTINE CALPRT

The main results of the program are given in CALPRT. These are obtained by averaging the screen length, SSL; "hole" size, SSH; and the left and right coordinates of the cloud, SLY and SRY respectively, at each specific period MT for a total of KI replications of the program. In addition, CALPRT uses information from Subroutine ENDPTS to determine the probablelocations of the exploratory rectangle established in Subroutine SIZE.

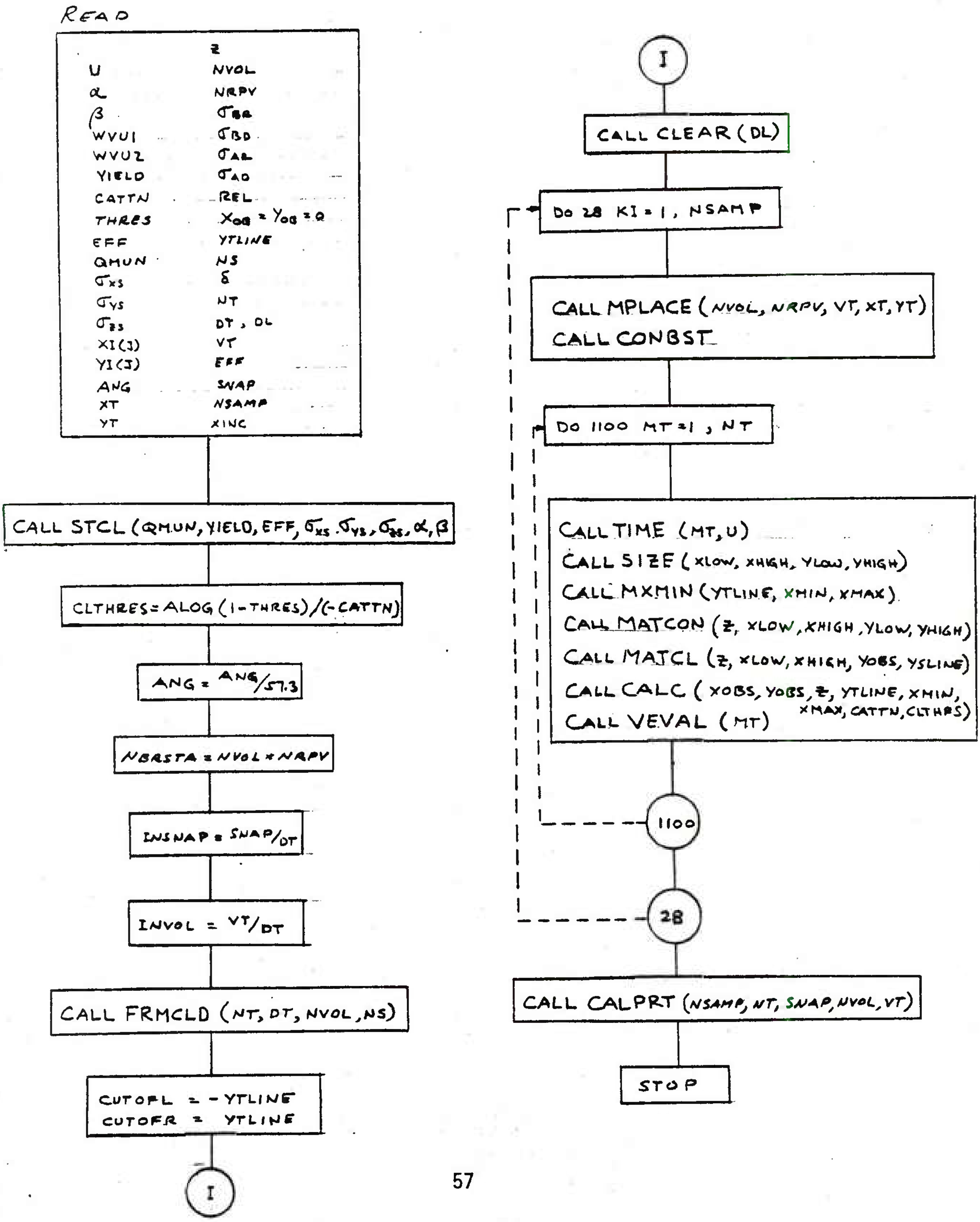
Next page is blank.

APPENDIX D

MODEL FLOW CHARTS

Next page is blank.

MAIN WP


READ
Z
U
NVOL
α
NRPV
β
σBR
WVU1
σBD
WVU2
σAL
YIELD
σAD
CATTN
REL
THRES
XOB = YOB = 0
EFF
YTLINE
QMUN
NS
σXS
δ
σYS
NT
σZS
DT, DL
XI(J)
VT
YI(J)
EFF
ANG
SNAP
XT
NSAMP
YT
XINC
CALL STCL (QMUN, YIELD, EFF, σXS, σYS, σZS, α, β
CLTHRES = ALOG (1-THRES)/(-CATTN)
ANG = ANG/57.3
NBRSTA = NVOL * NRPV
INSNAP = SNAP/DT
INVOL = VT/DT
CALL FRMCLD (NT, DT, NVOL, NS)
CUTOFL = - YTLINE
CUTOFR = YTLINE
I
I
CALL CLEAR (DL)
DO 28 KI = 1, NSAMP
CALL MPLACE (NVOL, NRPV, VT, XT, YT)
CALL CONBST
DO 1100 MT = 1, NT
CALL TIME (MT, U)
CALL SIZE (XLOW, XHIGH, YLOW, YHIGH)
CALL MXMIN (YTLINE, XMIN, XMAX)
CALL MATCON (Z, XLOW, XHIGH, YLOW, YHIGH)
CALL MATCL (Z, XLOW, XHIGH, YOBS, YSLINE)
CALL CALC (XOBS, YOBS, Z, YTLINE, XMIN, XMAX, CATTN, CLTHRS)
CALL VEVAL (MT)
1100
28
CALL CALPRT (NSAMP, NT, SNAP, NVOL, VT)
STOP

SUBROUTINE STCL $(Q, \Omega, \lambda, \sigma_{xs}, \sigma_{ys}, \sigma_{zs}, \alpha, \beta)$

$$\text{FACTOR} = \frac{2}{(2\pi)^{3/2}} * \text{EFF} * \text{QMUN} * \text{YIELD}$$

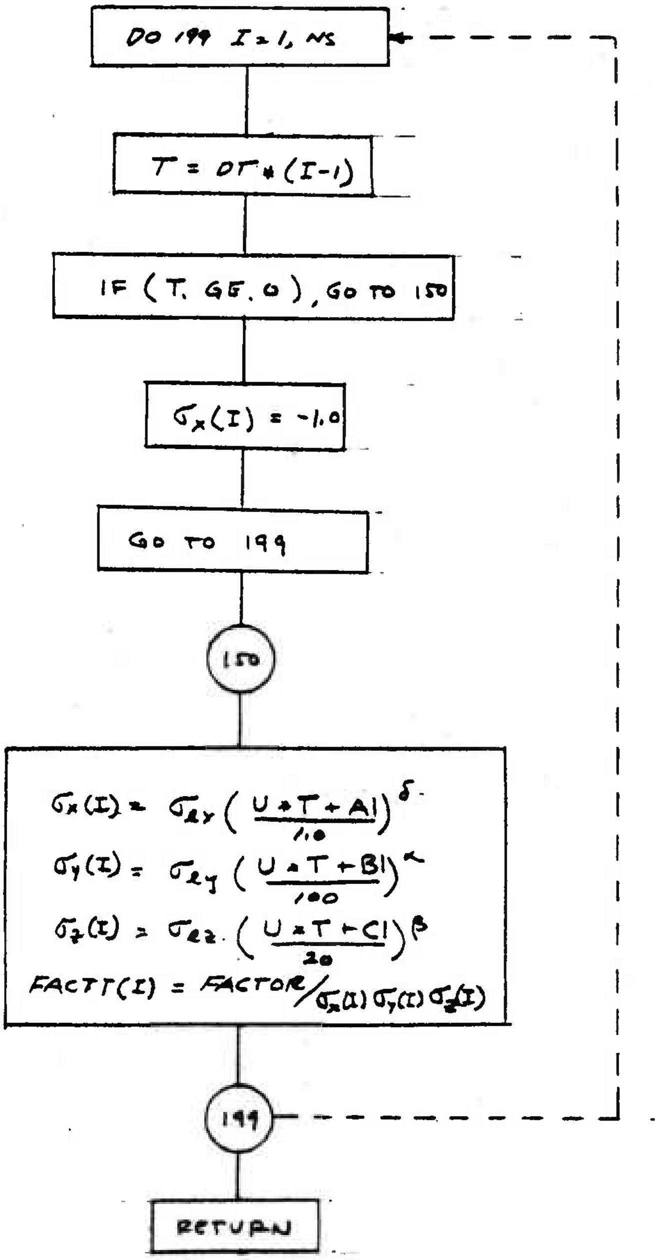
$$A1 = 1.0 \left(\frac{\sigma_{xs}}{.1522} \right)^{\frac{1}{.929}}$$

$$B1 = 100 \left(\frac{\sigma_{ys}}{3.41} \right)^{1/\alpha}$$

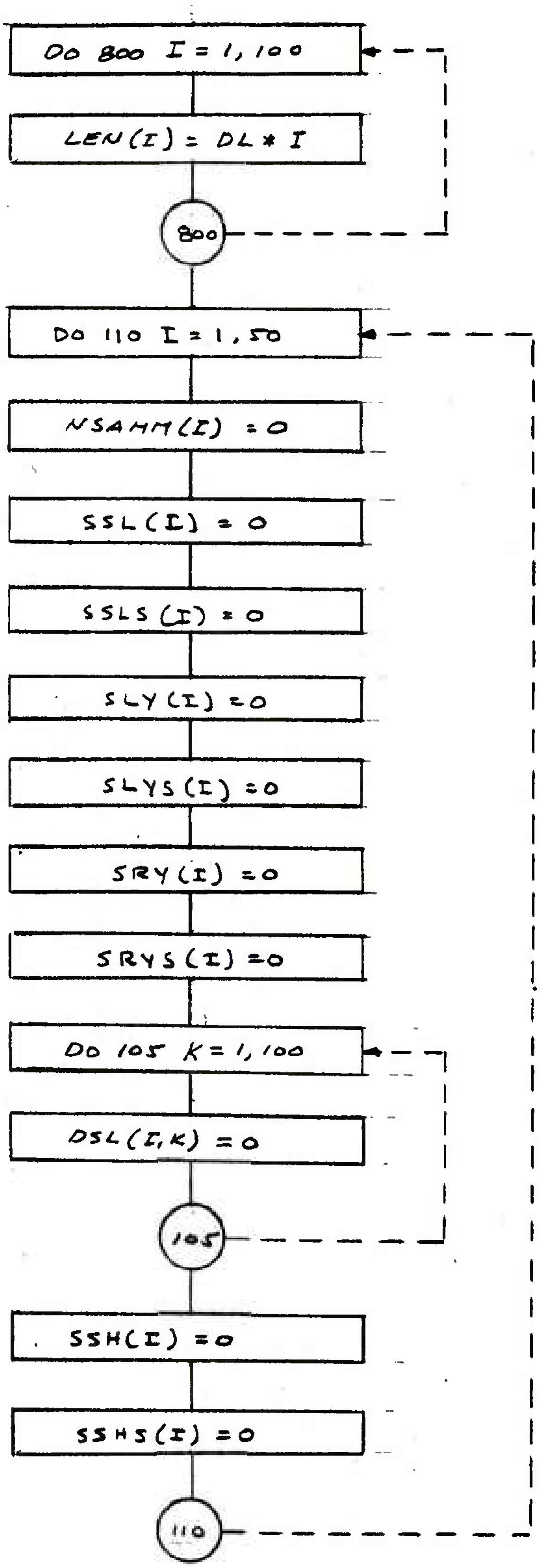
$$C1 = 20 \left(\frac{\sigma_{zs}}{1.35} \right)^{1/\beta}$$

RETURN

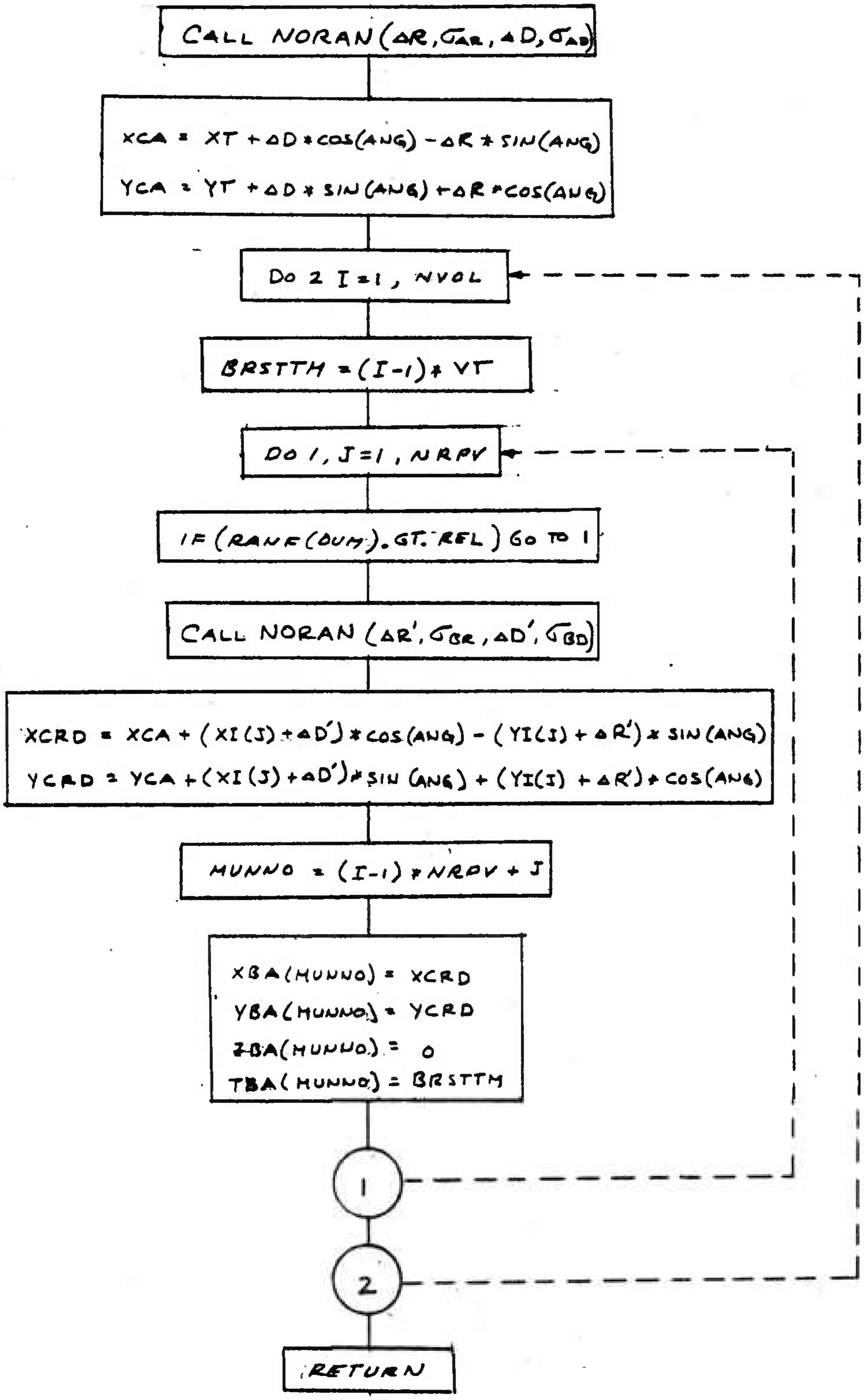
SUBROUTINE FRMCLD (NT, DT, NVOL, NS)



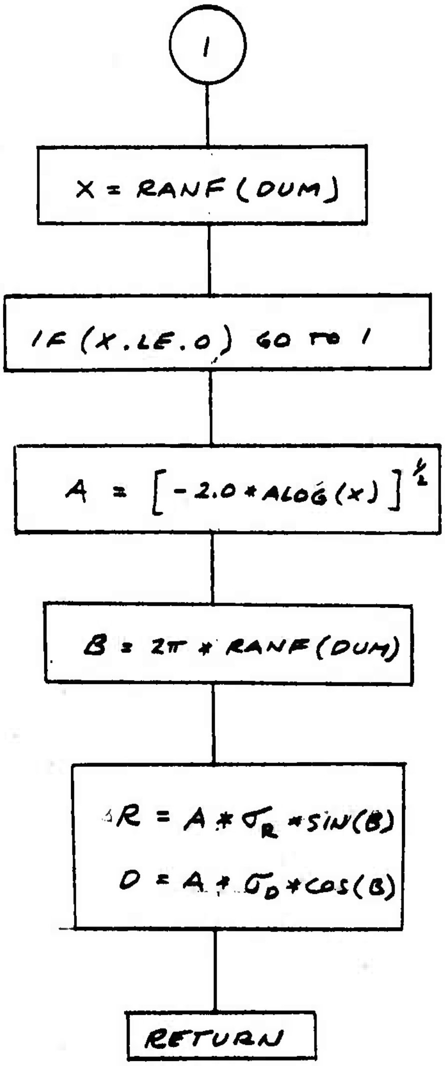
SUBROUTINE CLEAR (DL)



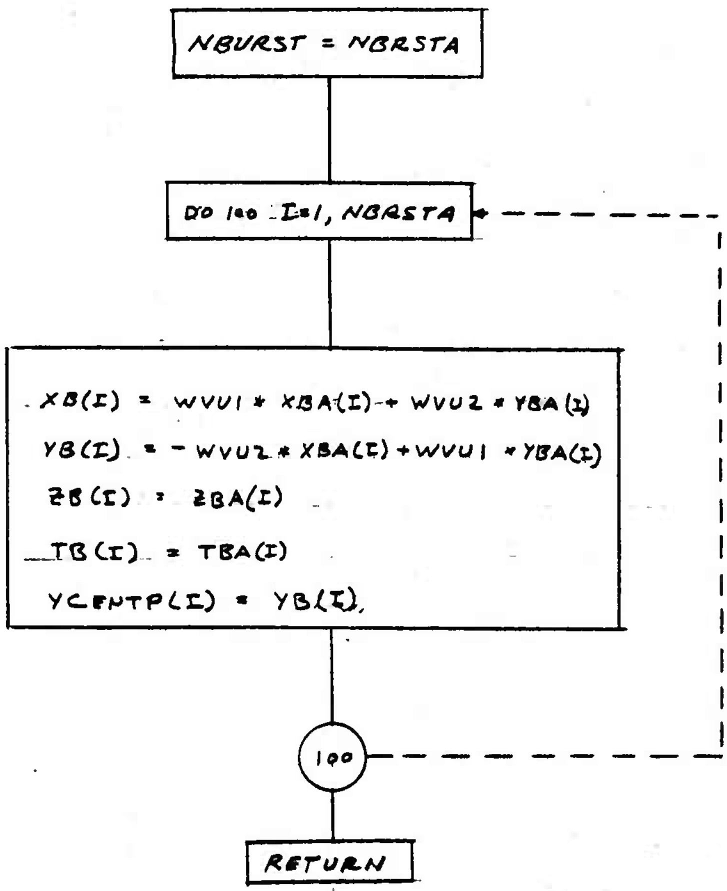
SUBROUTINE MPLACE (NVOL, NRPV, VT, XT, YT)



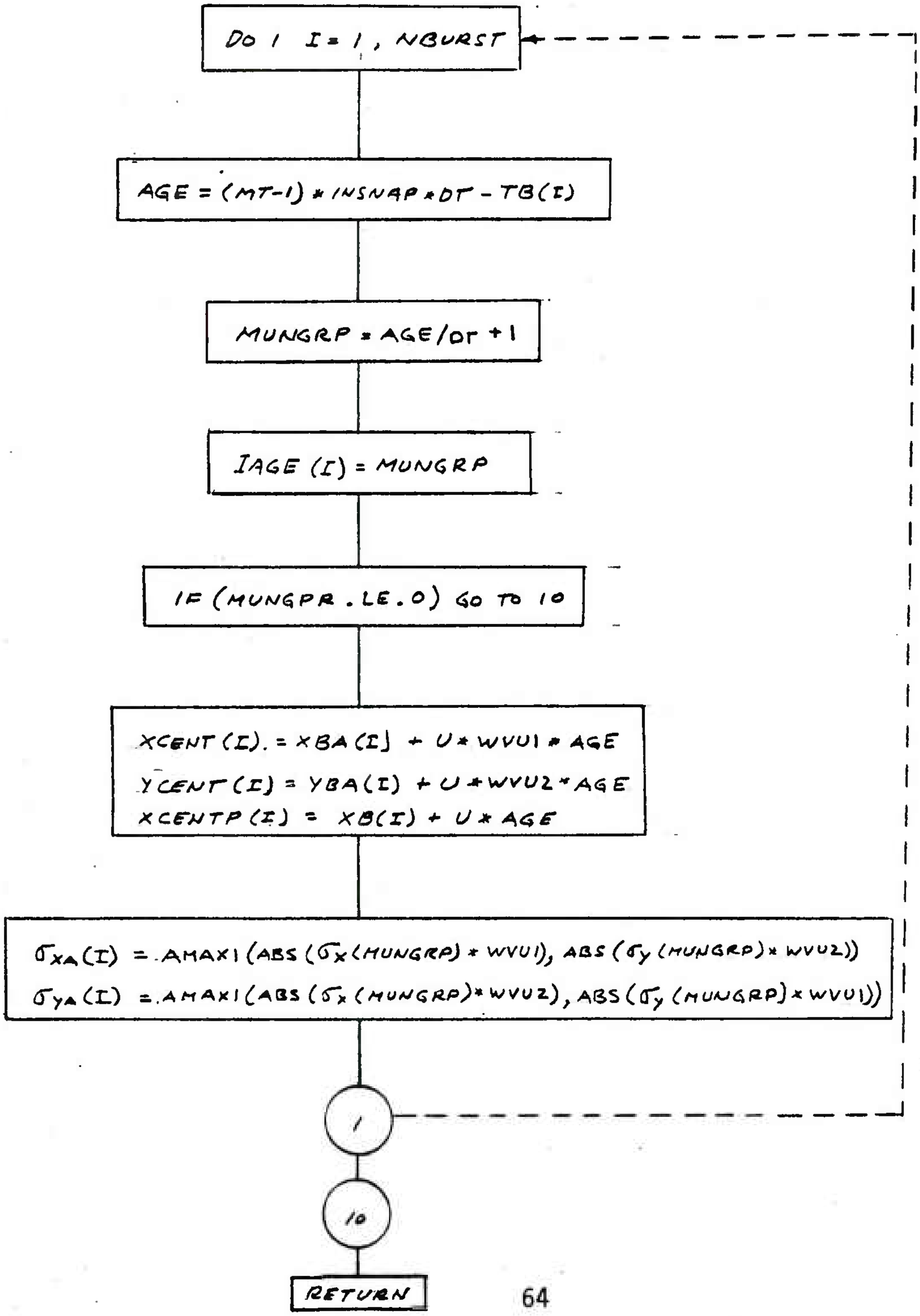
Subroutine NORAN (R, σ_R, D, σ_D)



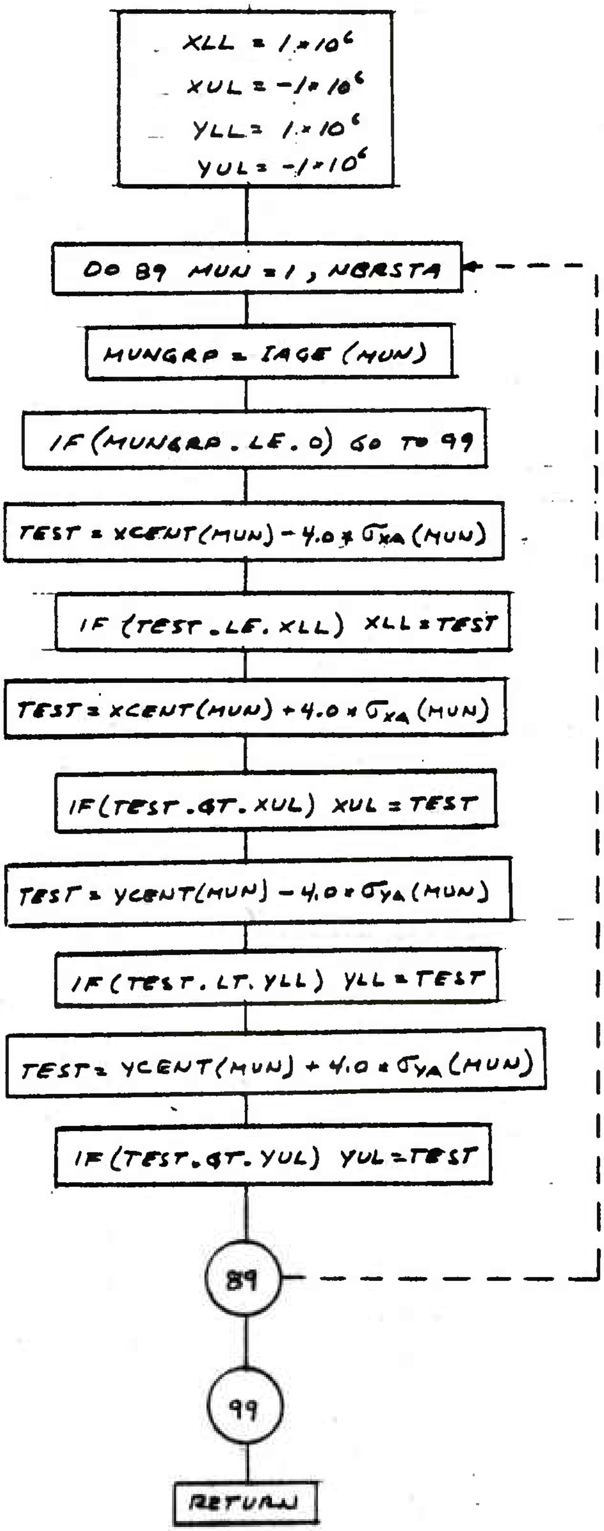
SUBROUTINE CONBST



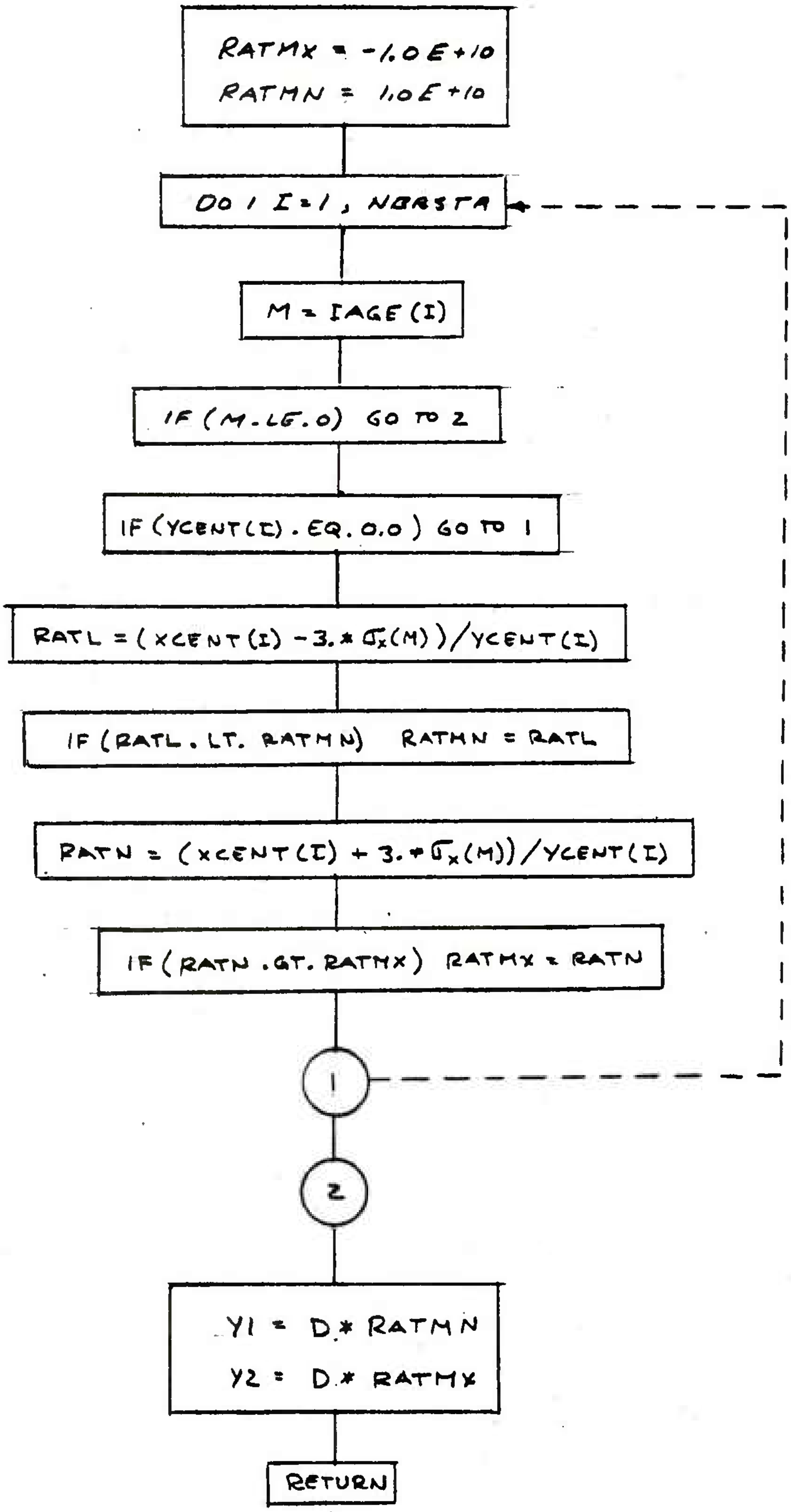
SUBROUTINE TIME (MT, U)



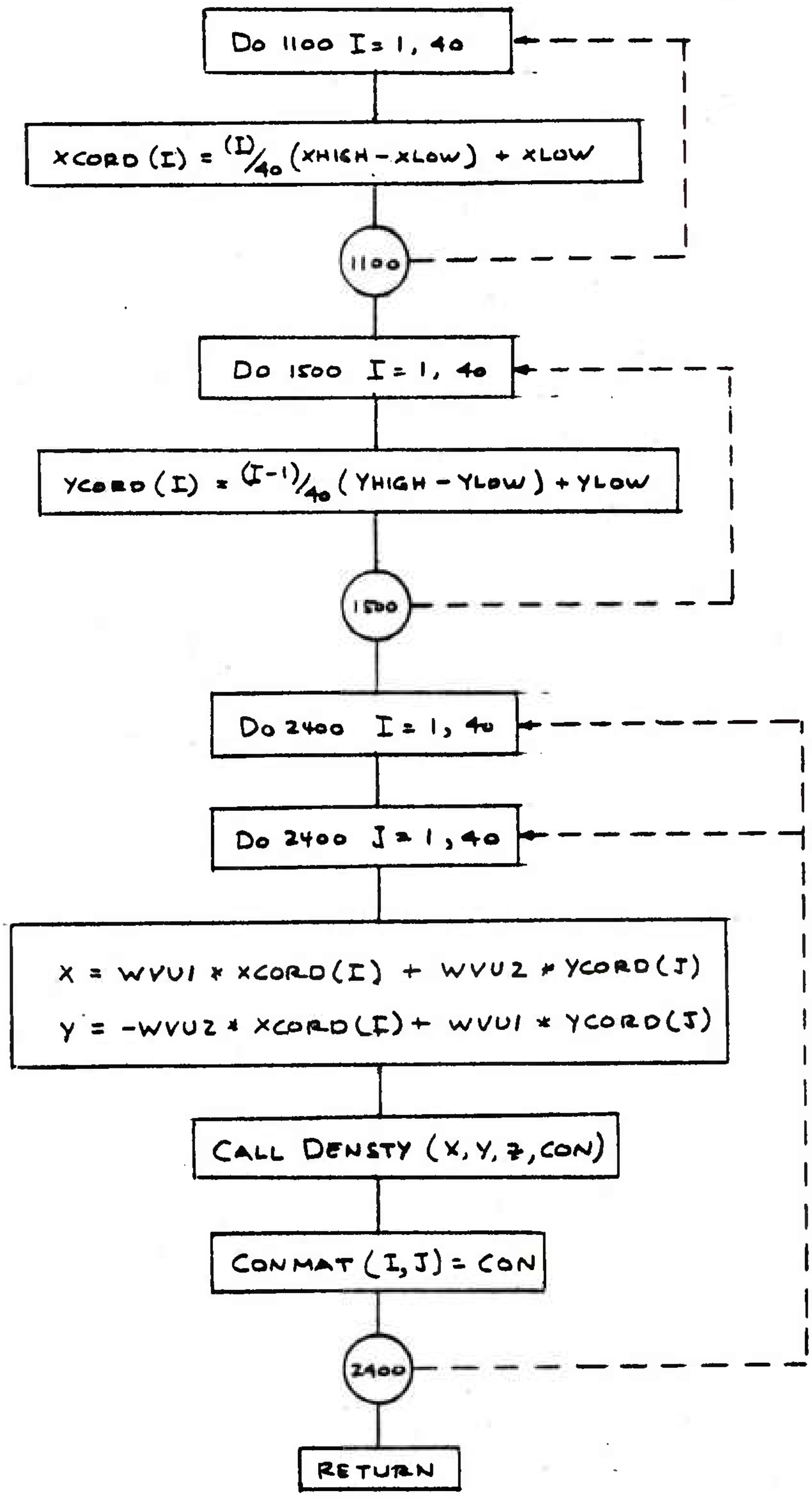
SUBROUTINE SIZE (XLL, XUL, YLL, YUL)



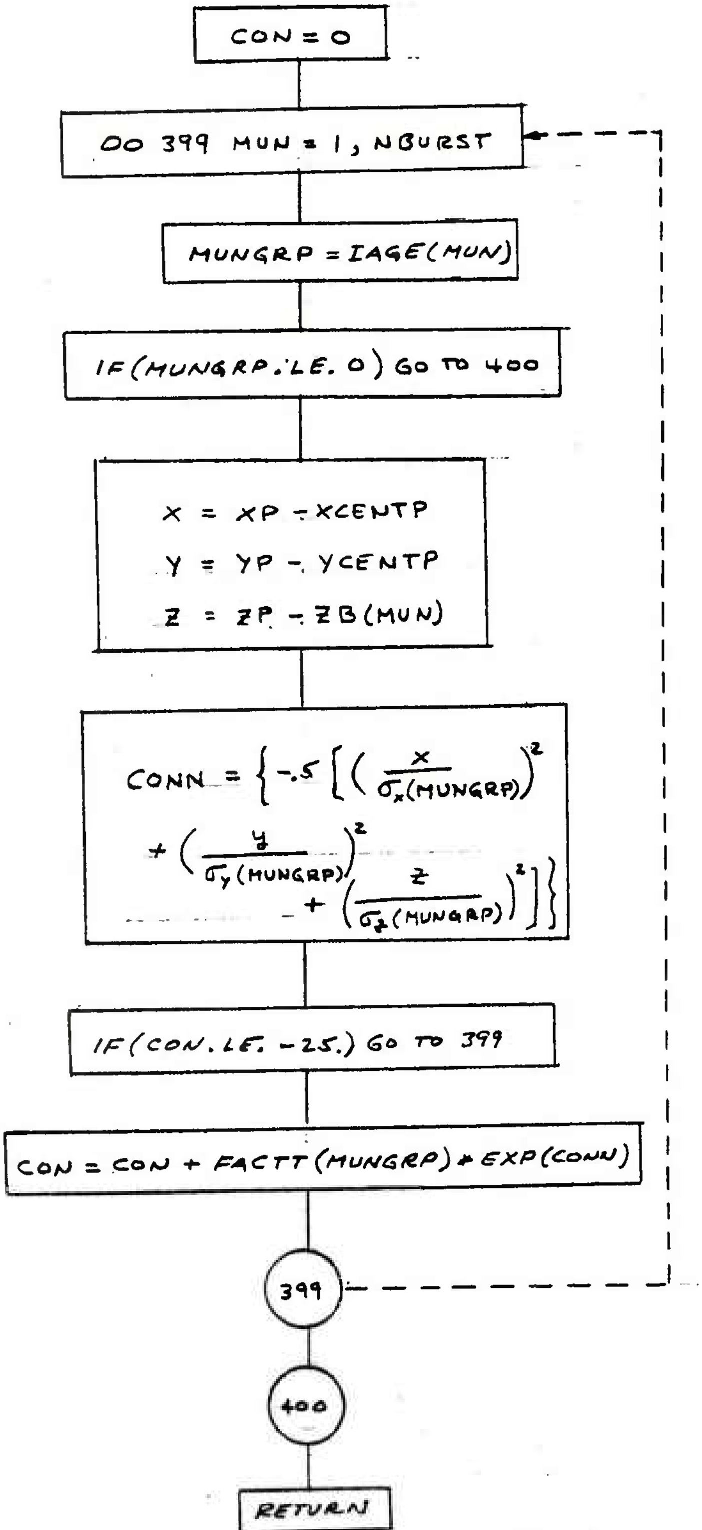
SUBROUTINE MXMIN (D, Y1, Y2)



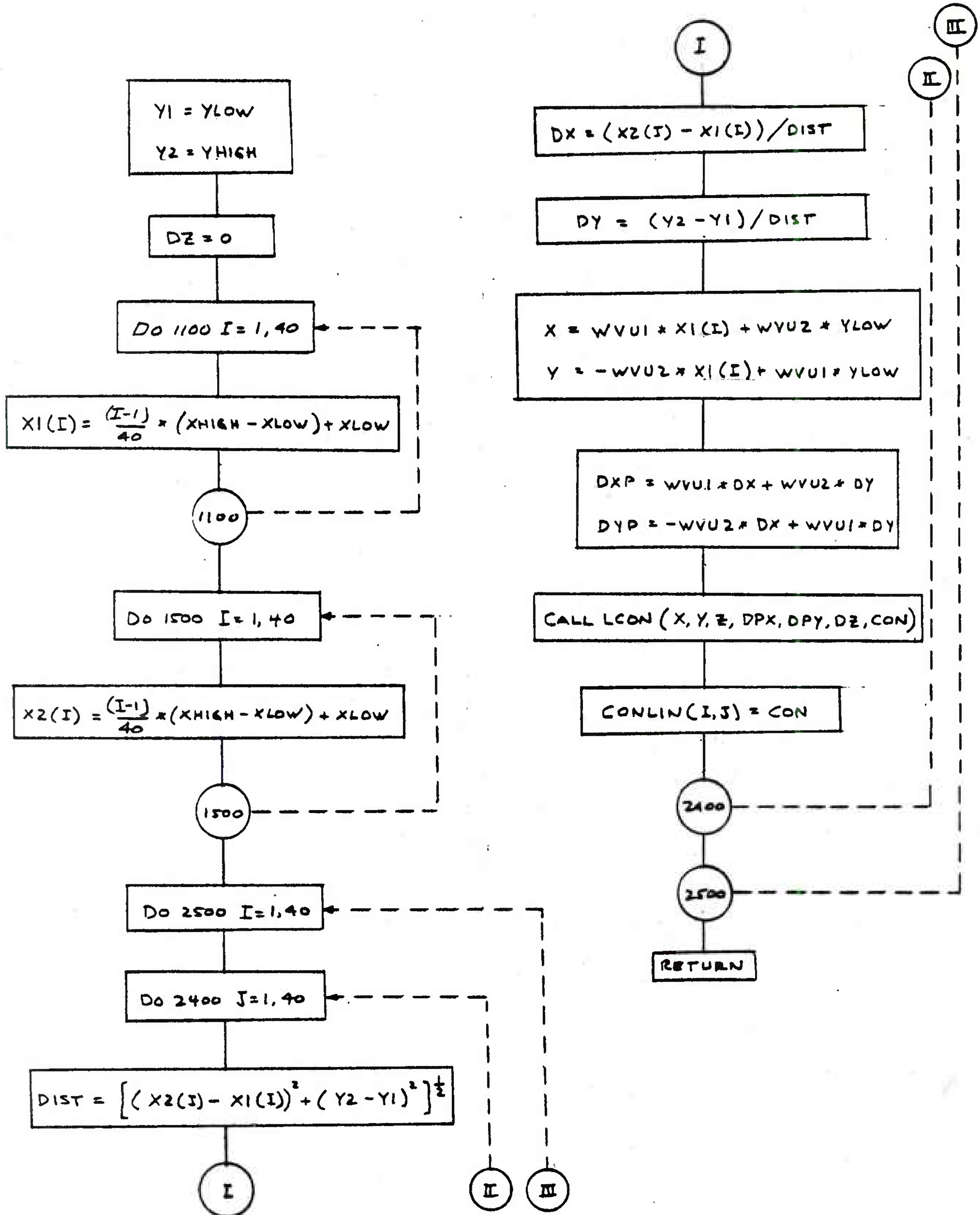
SUBROUTINE MATCON (Z, XLOW, XHIGH, YLOW, YHIGH)



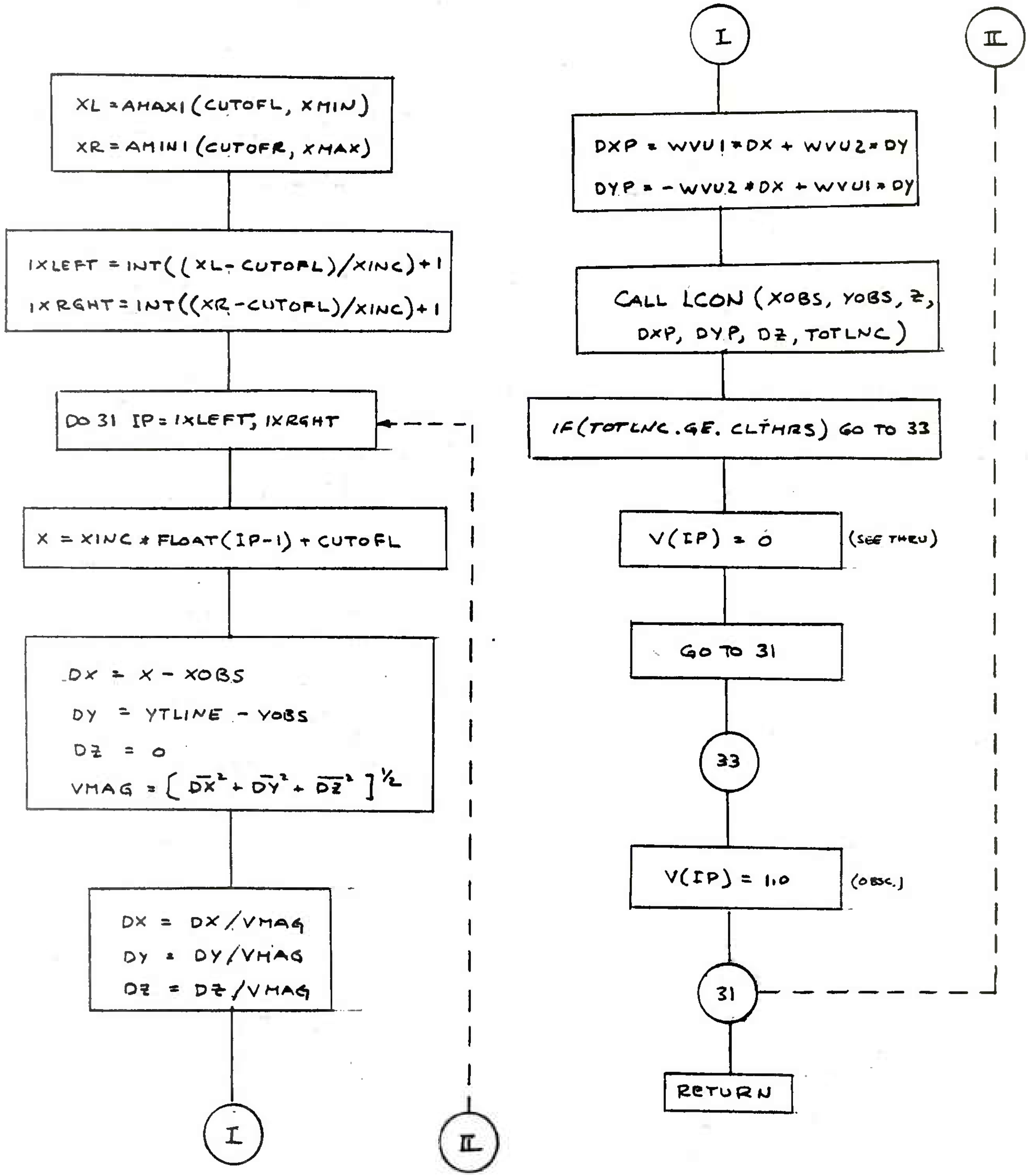
Subroutine DENSTY (XP, YP, ZP, CON)



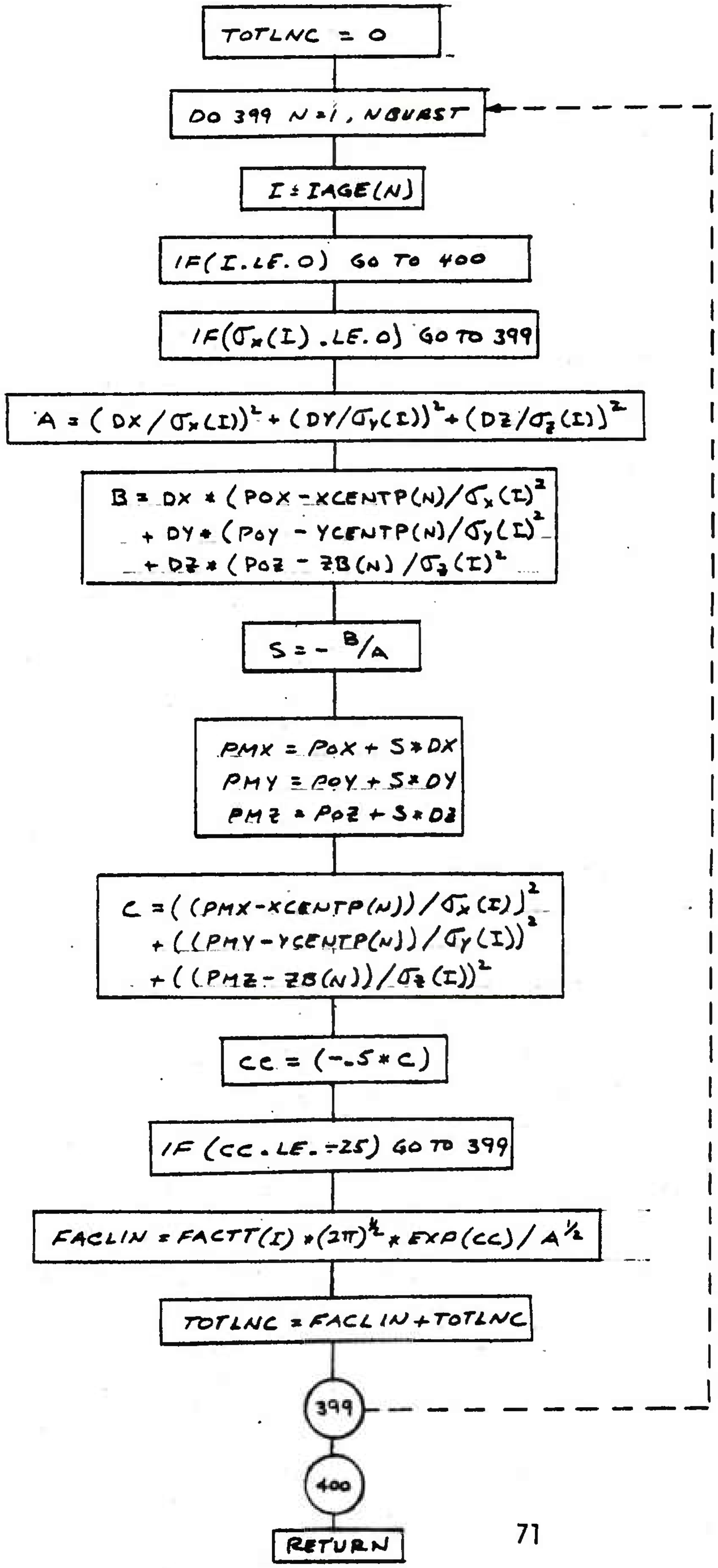
Subroutine MATCL (Z, XLOW, XHIGH, YLOW, YHIGH)



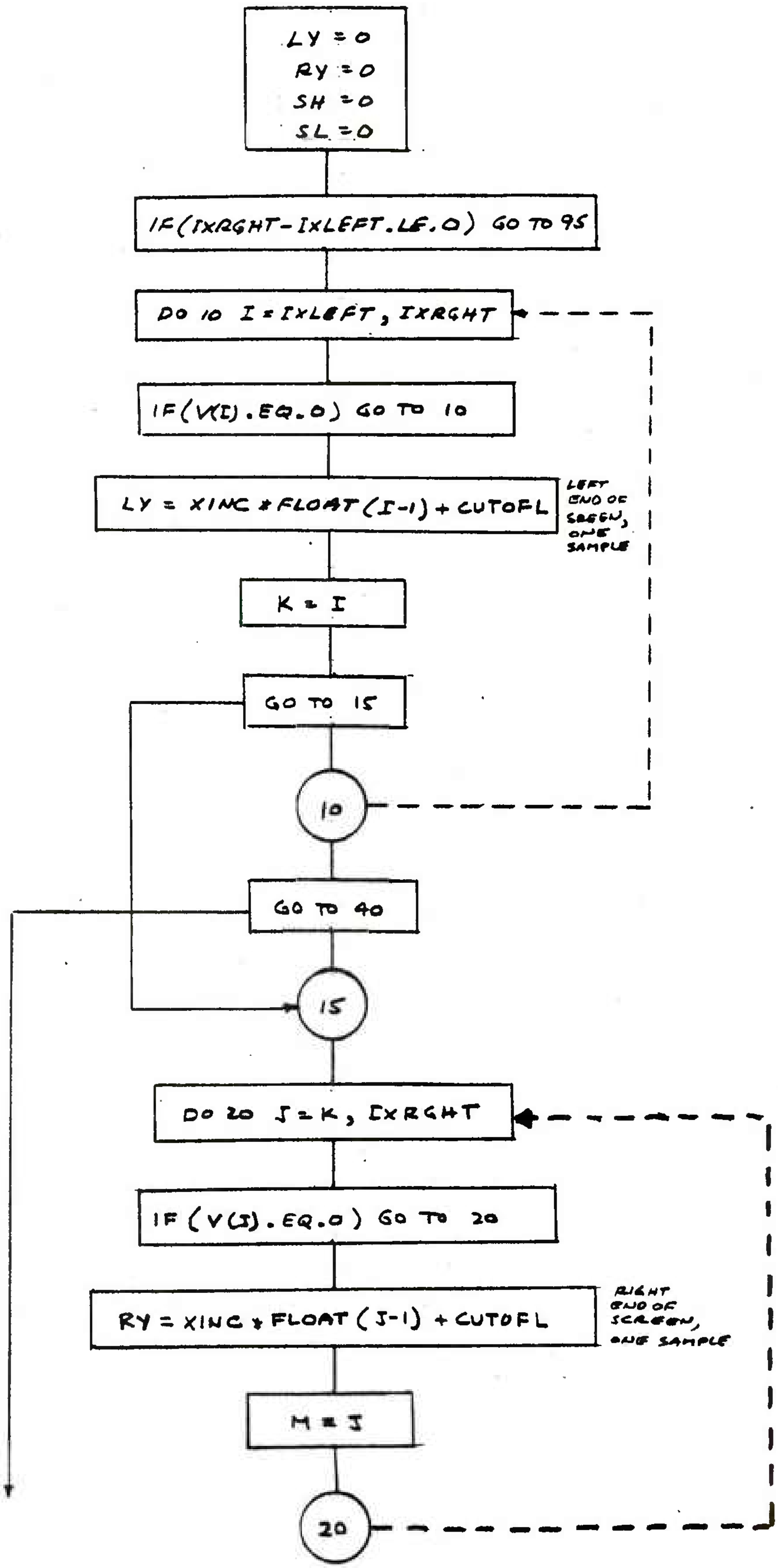
SUBROUTINE CALC (XOBS, YOBS, Z, YTLINE, XMIN, XMAX, CATTN, CLTHRS)



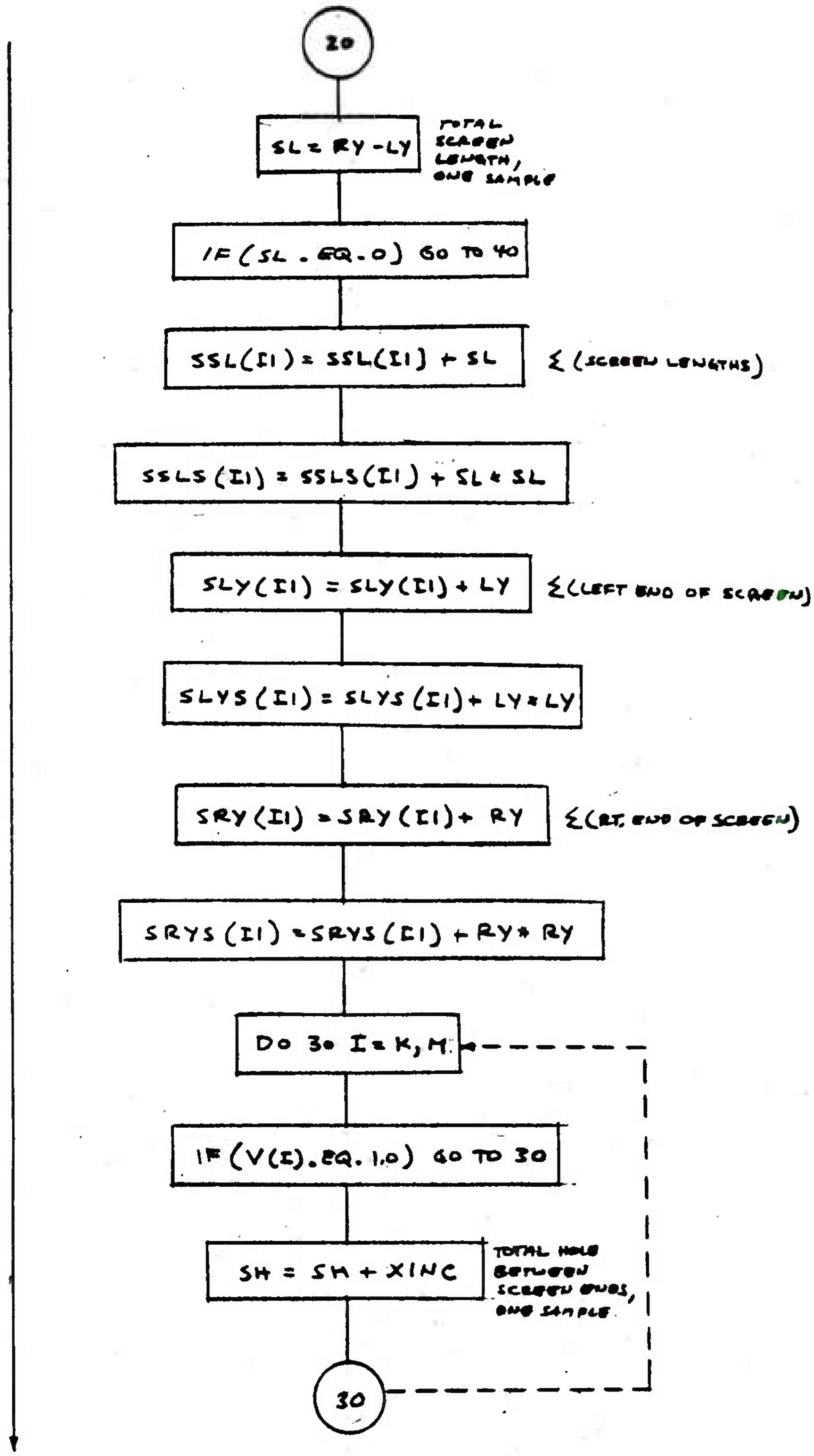
SUBROUTINE LCON (POX, POY, POZ, DX, DY, DZ, TOTLNC)



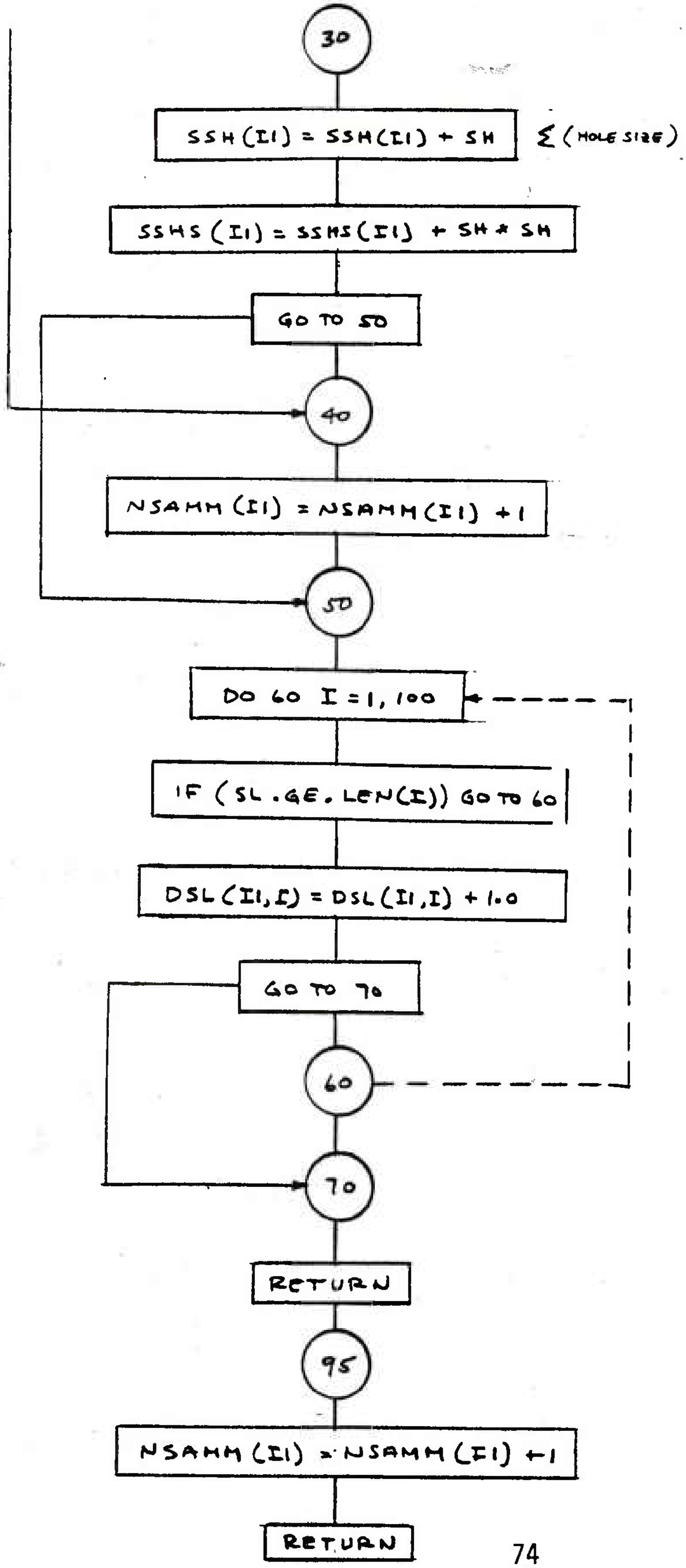
SUBROUTINE VEVAL (I1)



SUBROUTINE VEVAL (CONT.)



SUBROUTINE VEVAL (CONT.)



SUBROUTINE CALPRT (NSAMP, NT, SNAP, NVOL, VT)

DO 520 I = 1, NT

NSAMPI = NSAMP - NSAMH(I)

IF (NSAMPI.LE.1) GO TO 515

$SLYS(I) = [(SLYS(I) - SLY(I)^2/NSAMPI)/(NSAMPI-1)]^{1/2}$ — σ, LEFT COORD.

$SRYS(I) = [(SRYS(I) - SRY(I)^2/NSAMPI)/(NSAMPI-1)]^{1/2}$ — σ, RT. COORD.

$SSHS(I) = [(SSHS(I) - SSH(I)^2/NSAMPI)/(NSAMPI-1)]^{1/2}$ — σ, HOLE SIZE

SLY(I) = SLY(I)/NSAMPI — AVE. LEFT COORD.

SRY(I) = SRY(I)/NSAMPI — AVE. RT. COORD.

SSH(I) = SSH(I)/NSAMPI — AVE Σ(HOLE SIZE)

$SSLS(I) = [(SSLS(I) - SSL(I)^2/NSAMP)/(NSAMP-1)]^{1/2}$ — σ, SCREEN LTH.

515

SSL(I) = SSL(I)/NSAMP — AVE SCREEN LTH.

520

DO 530 I = 1, NT

STIME = (I-1) * SNAP

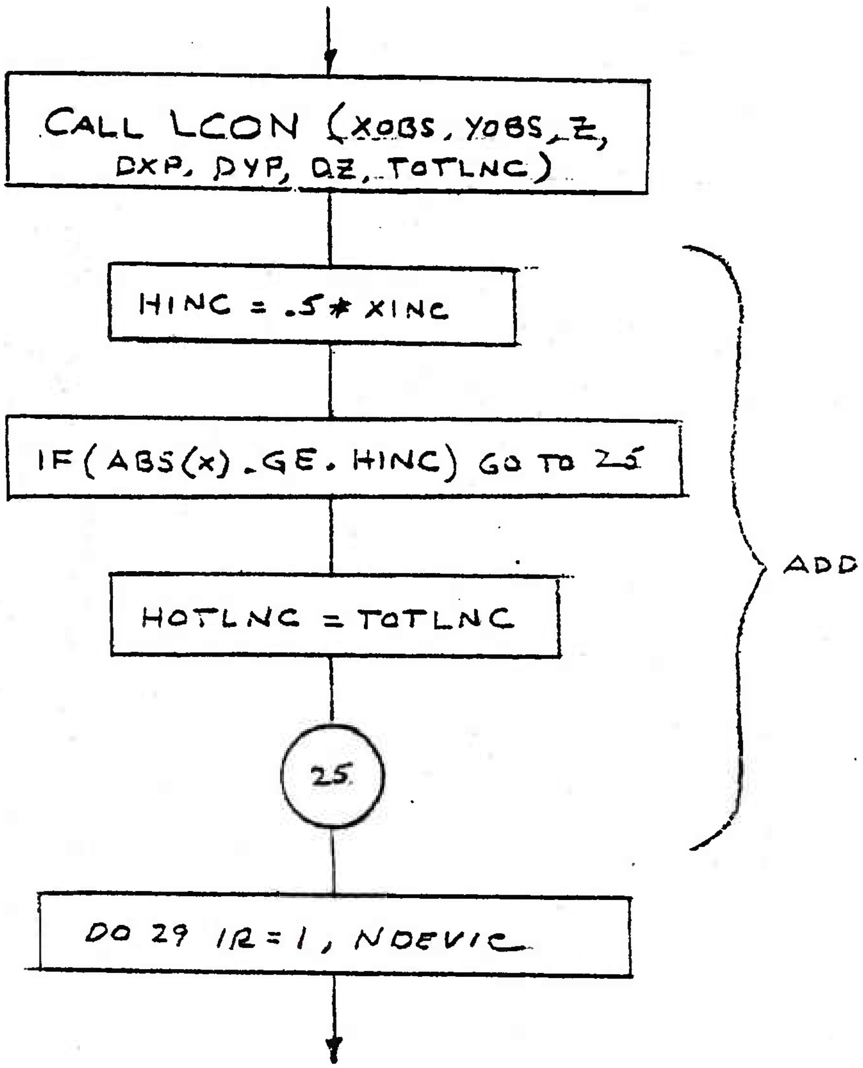
J = MINO(NVOL, 1 + INT(STIME/NT)) — ACT. VOLLIES FIRED

530

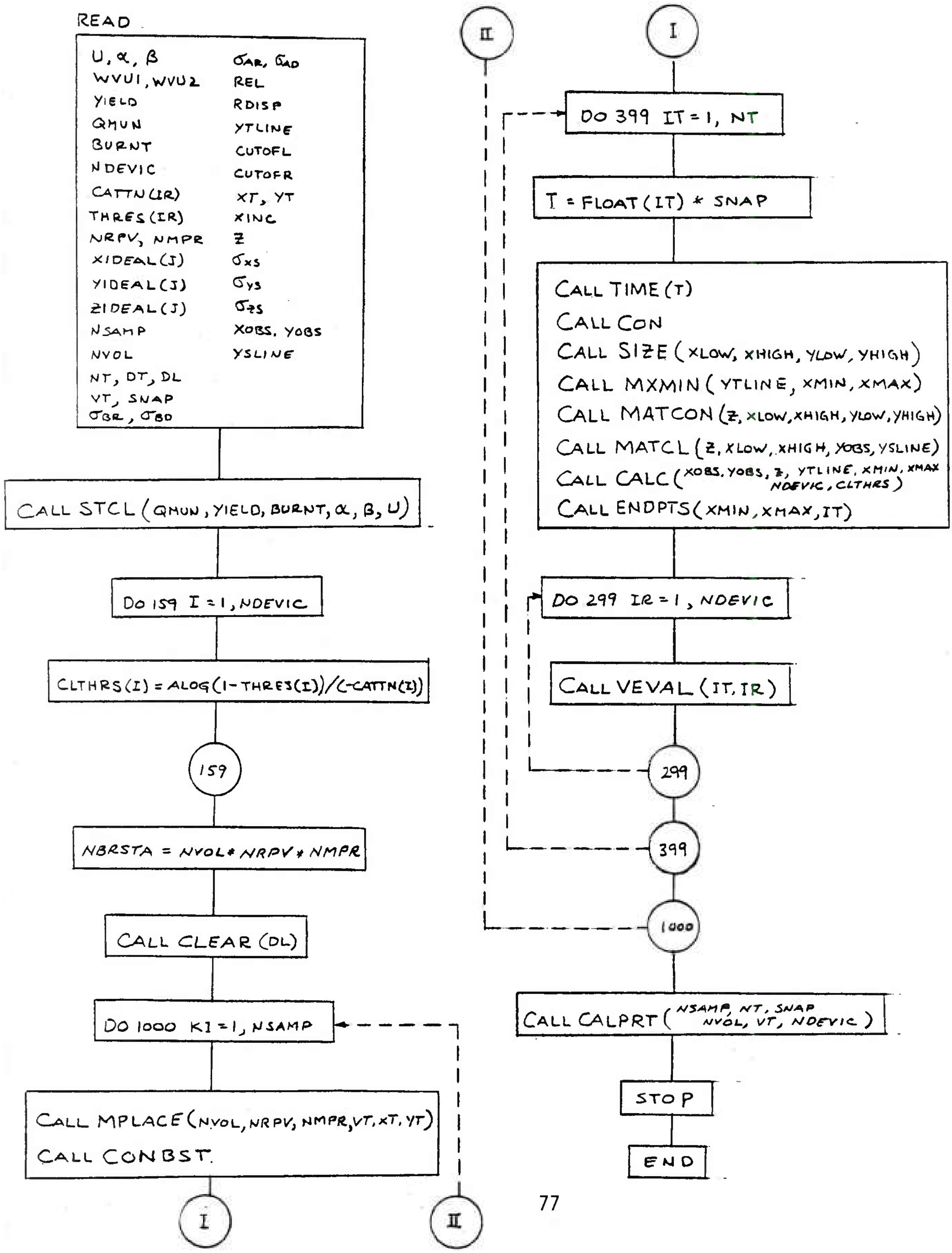
RETURN

SUBROUTINE CALC

MODIFICATION FOR COMPARATIVE ANALYSIS



MAIN HC


READ
U, α, β
WVU1, WVU2
YIELD
QMUN
BURNT
NDEVIC
CATTN(IR)
THRES(IR)
NRPV, NMPR
XIDEAL(J)
YIDEAL(J)
ZIDEAL(J)
NSAMP
NVOL
NT, DT, DL
VT, SNAP
σBR, σBD
σAR, σAD
REL
RDISP
YTLINE
CUTOFL
CUTOFR
XT, YT
XINC
Z
σxs
σys
σzs
XOBS, YOBS
YSLINE
CALL STCL (QMUN, YIELD, BURNT, α, β, U)
DO 159 I = 1, NDEVIC
CLTHRS(I) = ALOG(1-THRES(I))/(-CATTN(I))
159
NBRSTA = NVOL*NRPV*NMPR
CALL CLEAR (DL)
DO 1000 KI = 1, NSAMP
CALL MPLACE (NVOL, NRPV, NMPR, VT, XT, YT)
CALL CONBST
I
II
II
I
DO 399 IT = 1, NT
T = FLOAT (IT) * SNAP
CALL TIME (T)
CALL CON
CALL SIZE (XLOW, XHIGH, YLOW, YHIGH)
CALL MXMIN (YTLINE, XMIN, XMAX)
CALL MATCON (Z, XLOW, XHIGH, YLOW, YHIGH)
CALL MATCL (Z, XLOW, XHIGH, YOBS, YSLINE)
CALL CALC (XOBS, YOBS, Z, YTLINE, XMIN, XMAX, NDEVIC, CLTHRS)
CALL ENDPTS (XMIN, XMAX, IT)
DO 299 IR = 1, NDEVIC
CALL VEVAL (IT, IR)
299
399
1000
CALL CALPRT (NSAMP, NT, SNAP, NVOL, VT, NDEVIC)
STOP
END

SUBROUTINE STCL (QMUN, YIELD, BTIME, α, β, UP)

NPUFTL = 120

PUFT = FLOAT(NPUFTL)

WNDSPD = UP

$$A1 = 100\left(\frac{\sigma_{xs}}{3.41}\right)^{1/\alpha}$$

$$B1 = 100\left(\frac{\sigma_{ys}}{3.41}\right)^{1/\alpha}$$

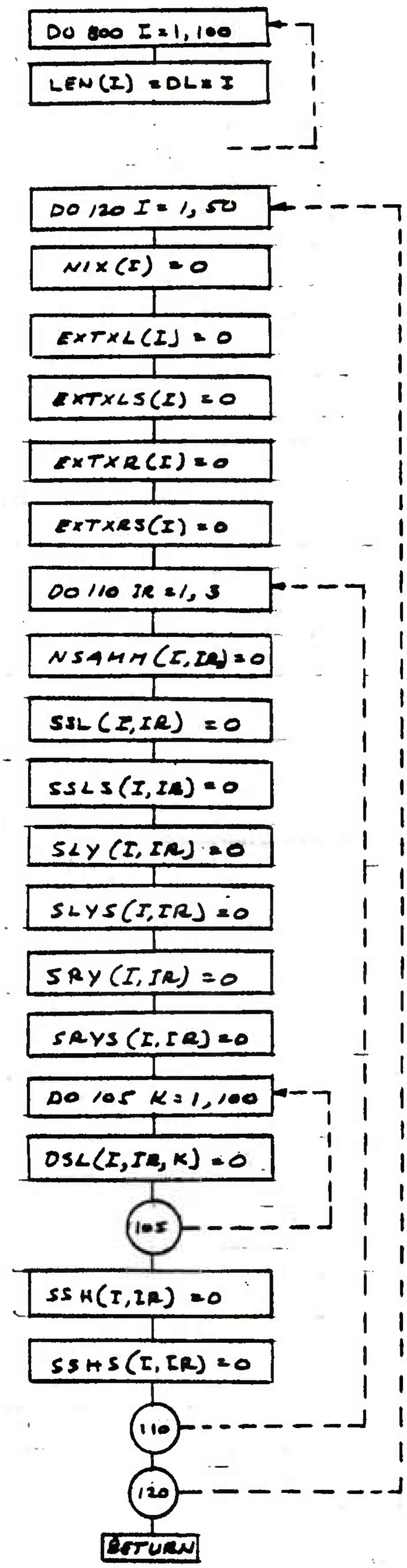
$$C1 = 20\left(\frac{\sigma_{zs}}{1.35}\right)^{1/\beta}$$

$$FACTOR = \frac{2}{(2\pi)^{3/2}} * QMUN * YIELD * \frac{1}{PUFT}$$

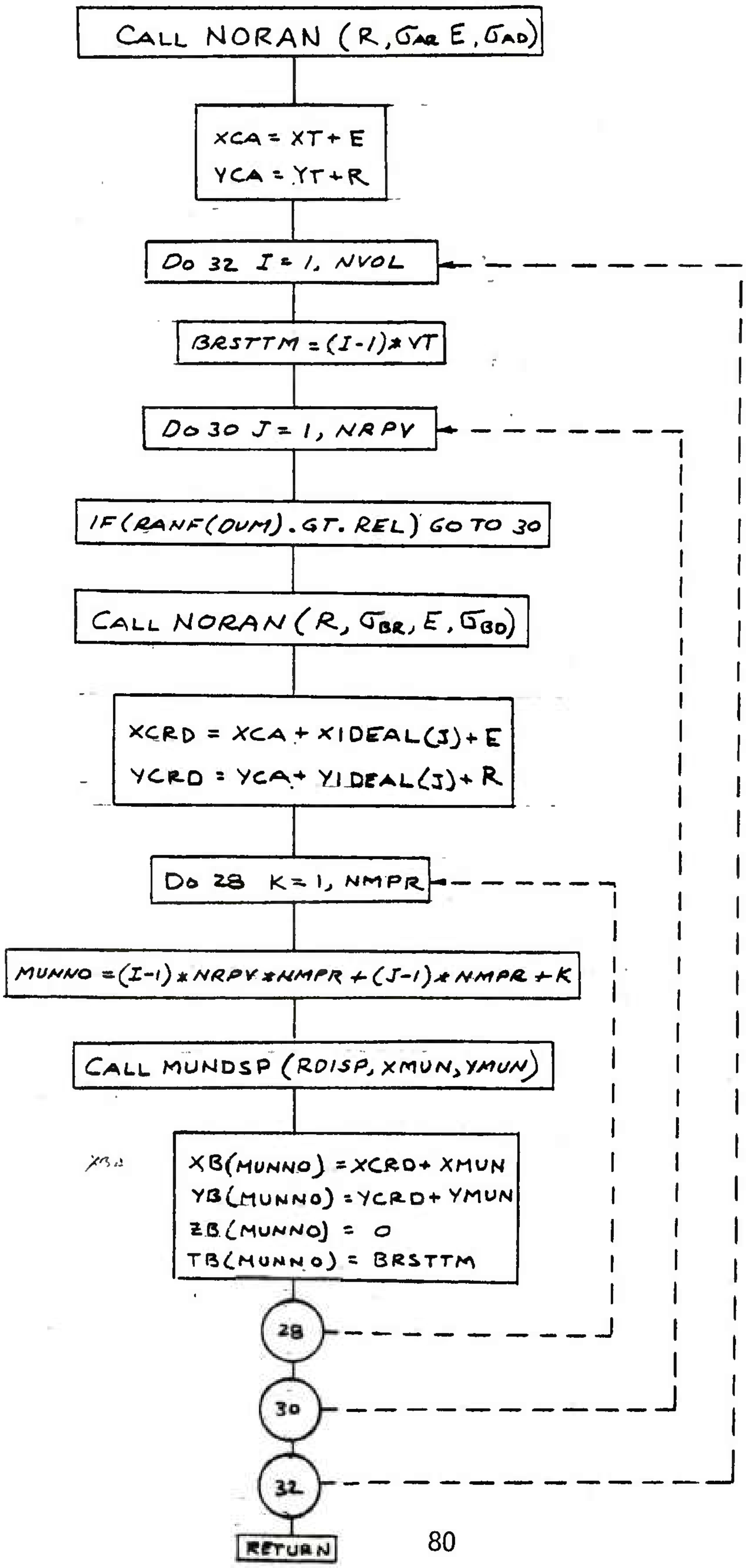
TINCR = BTIME/PUFT

RETURN

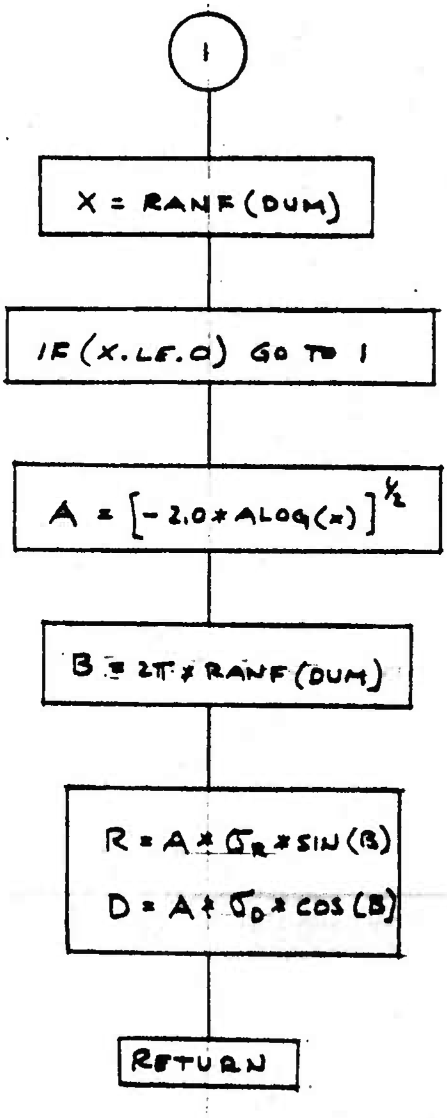
SUBROUTINE CLEAR (DL)



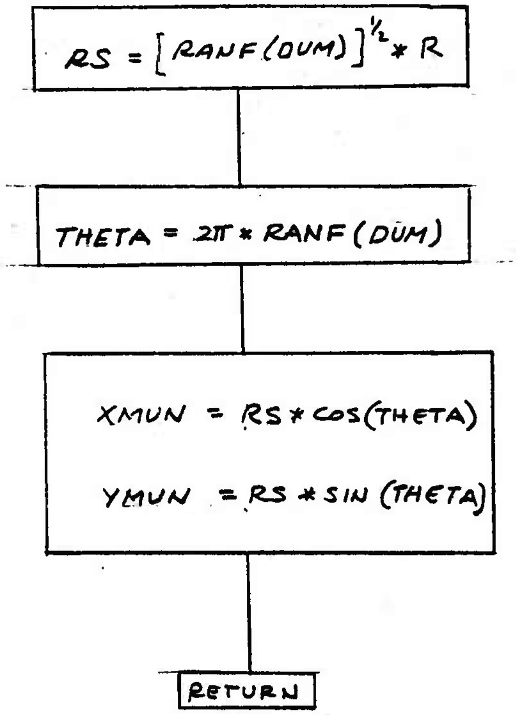
SUBROUTINE MPLACE (NVOL, NRPV, NMPR, VT, XT, YT)



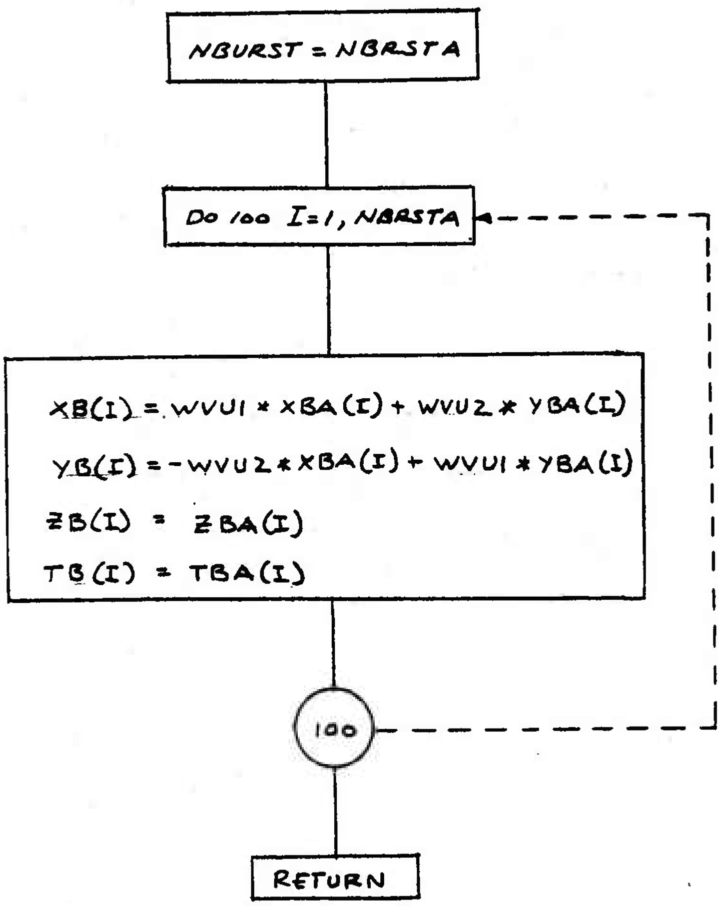
SUBROUTINE NORAN$(R, \sigma_R, D, \sigma_D)$



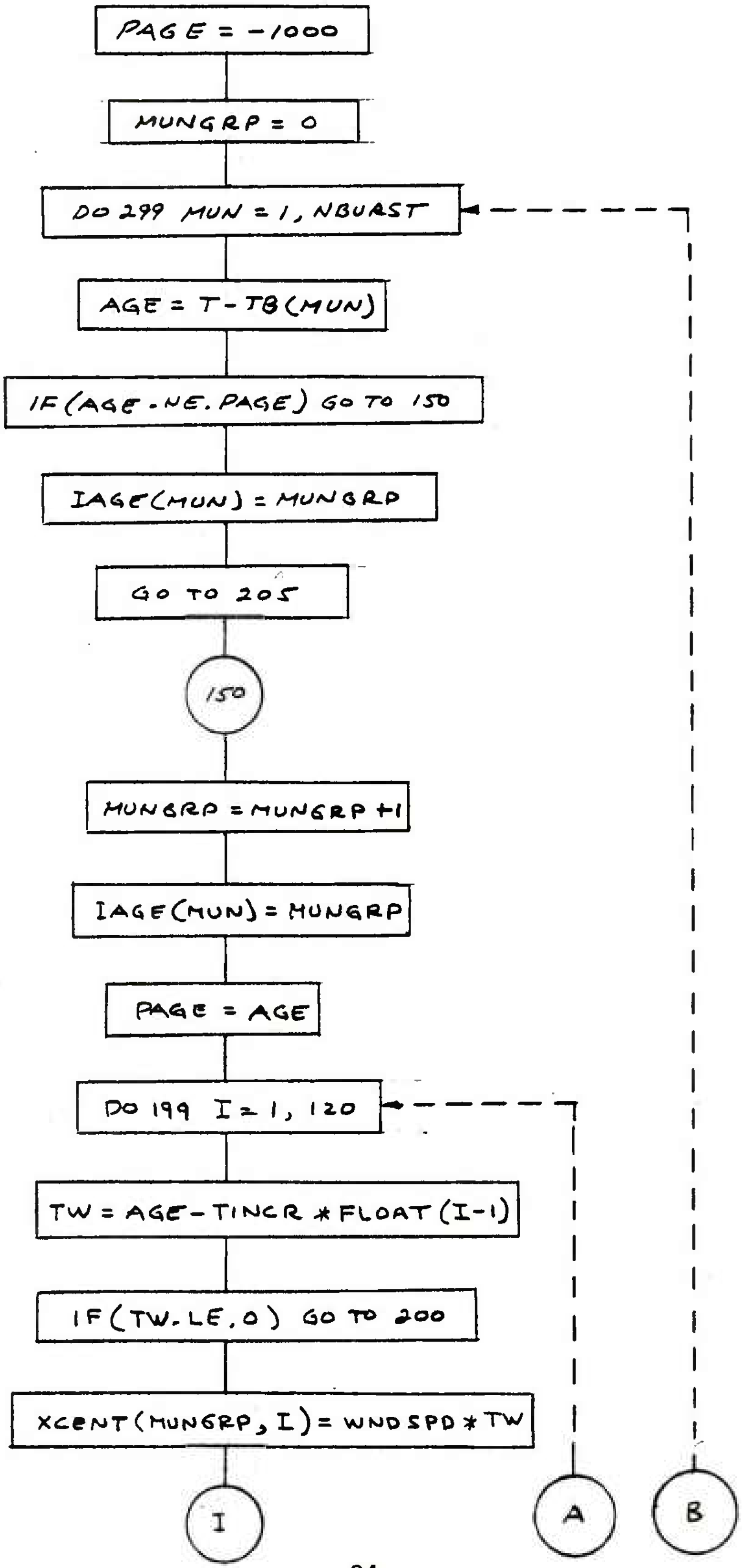
SUBROUTINE MUNDSP (R, XMUN, YMUN)



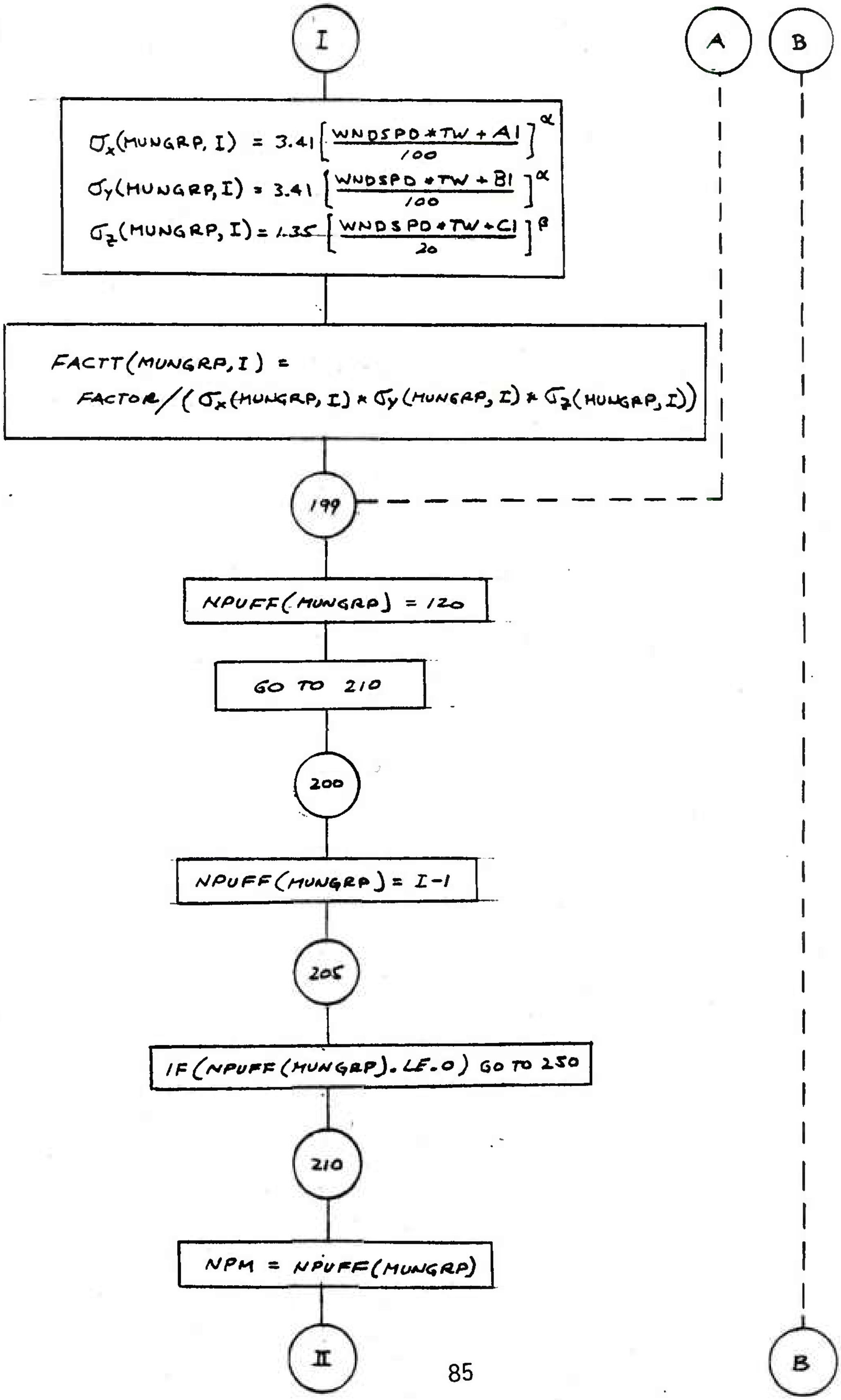
SUBROUTINE CONBST



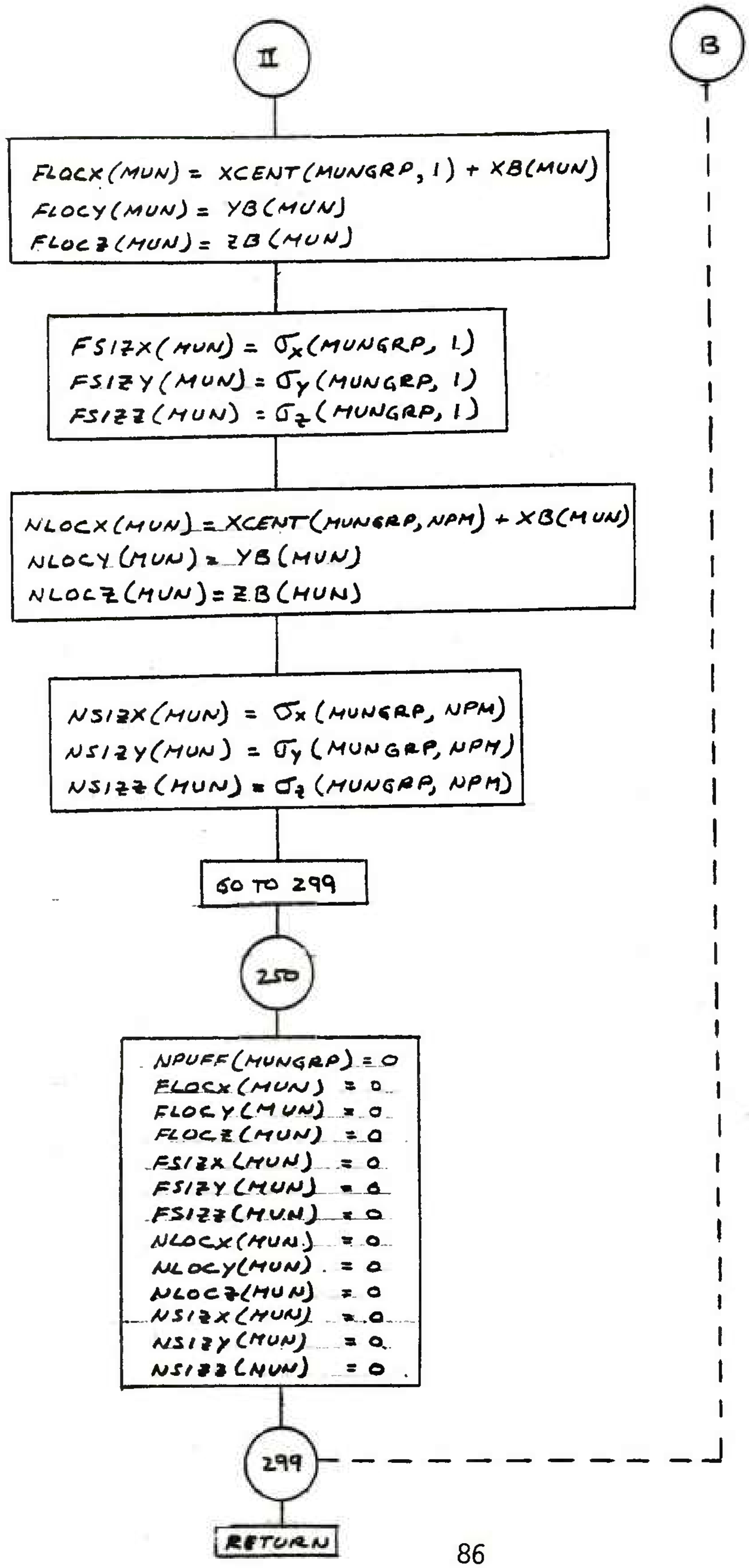
SUBROUTINE TIME (T)



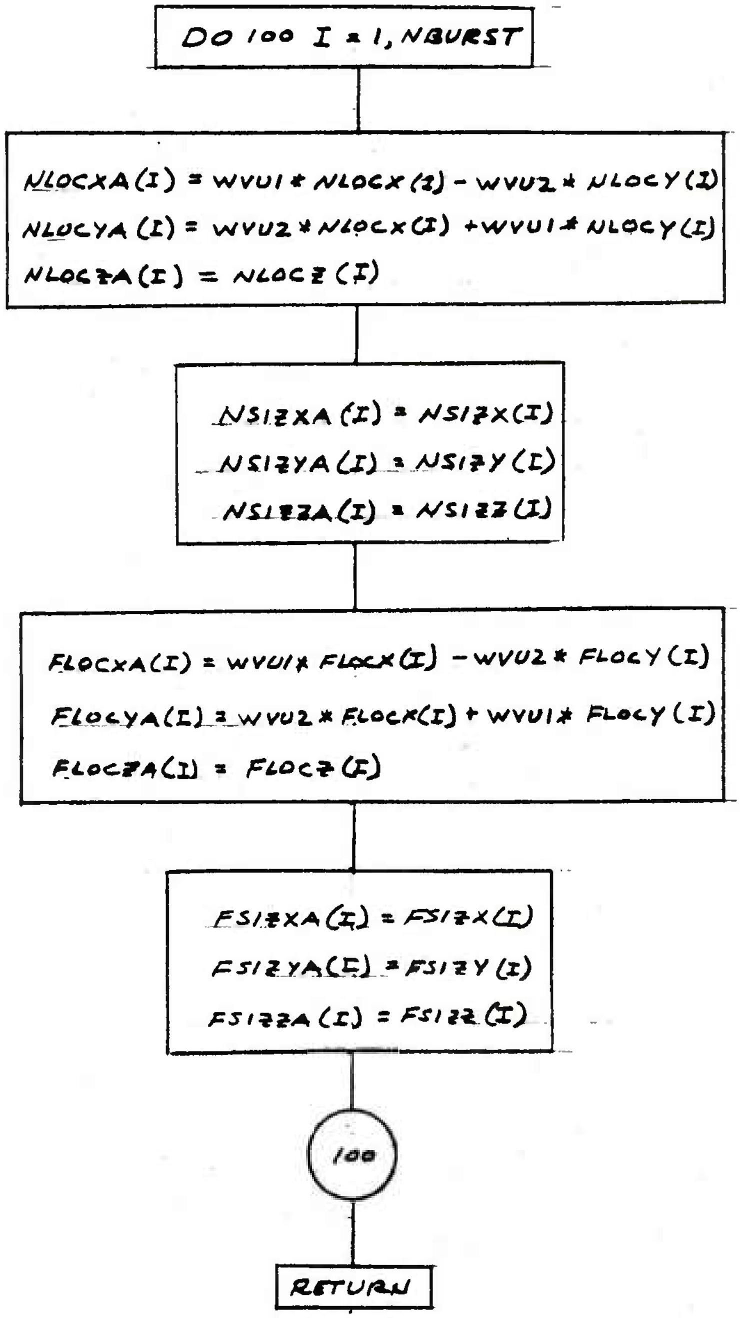
SUBROUTINE TIME (CONTINUED)



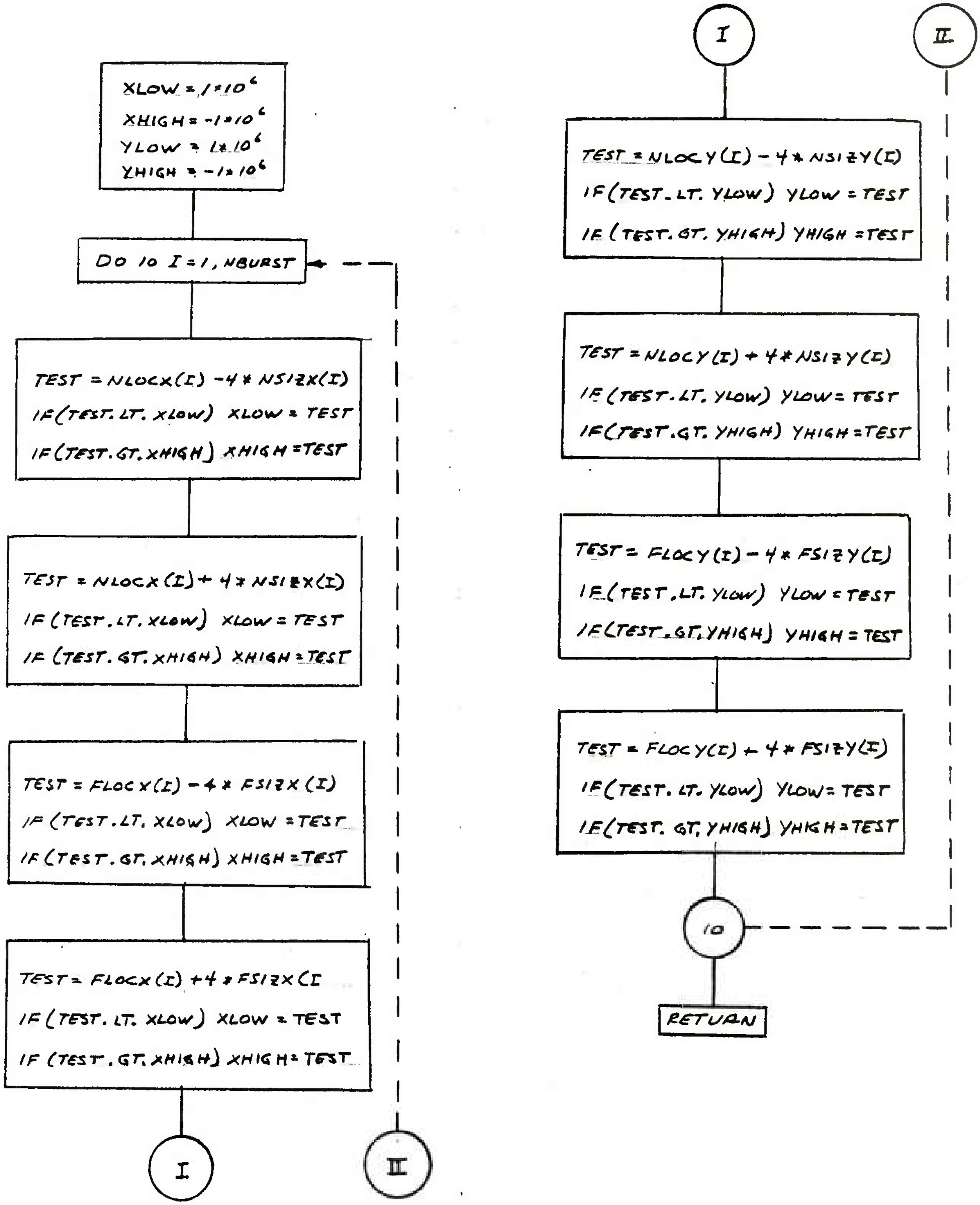
SUBROUTINE TIME (CONTINUED)



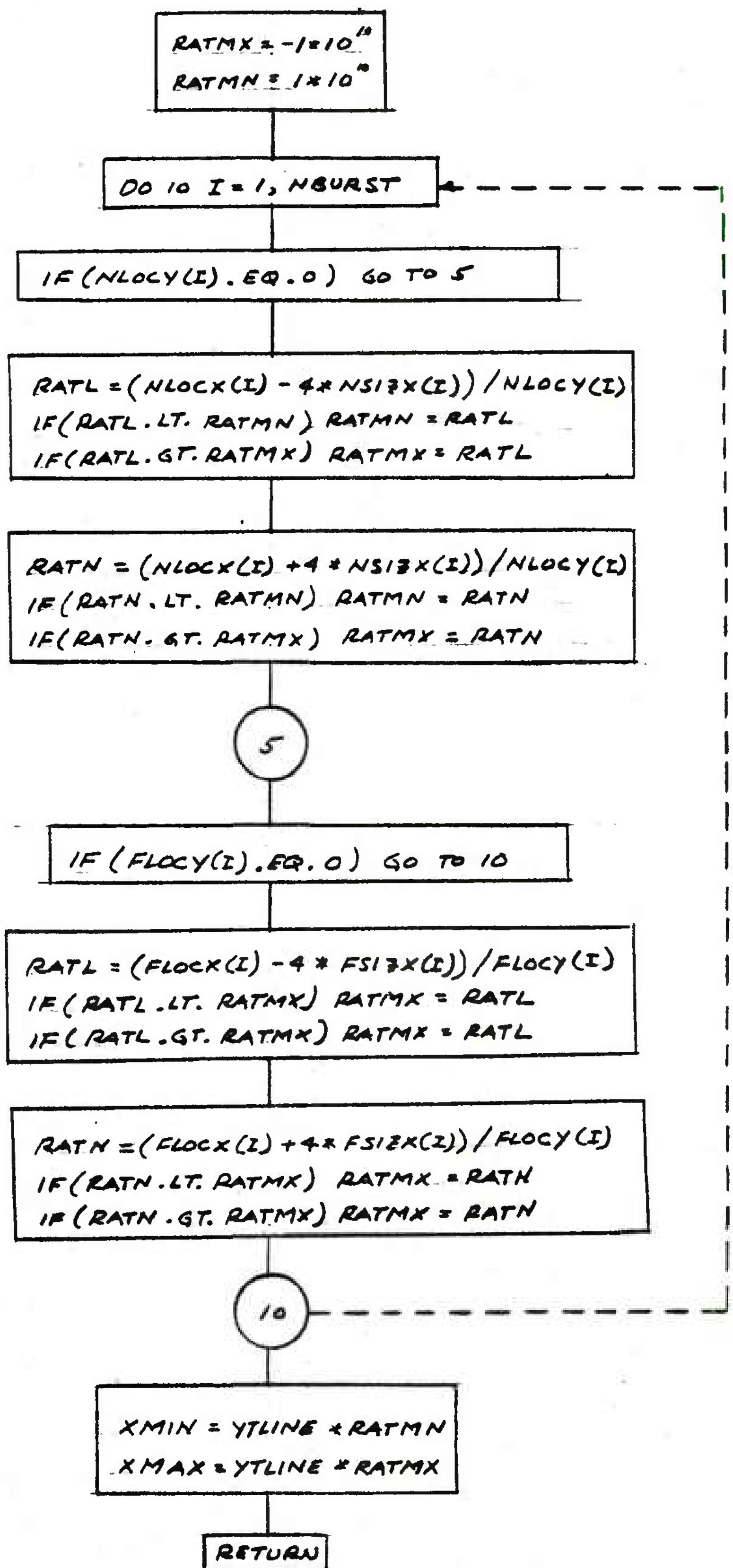
SUBROUTINE CON



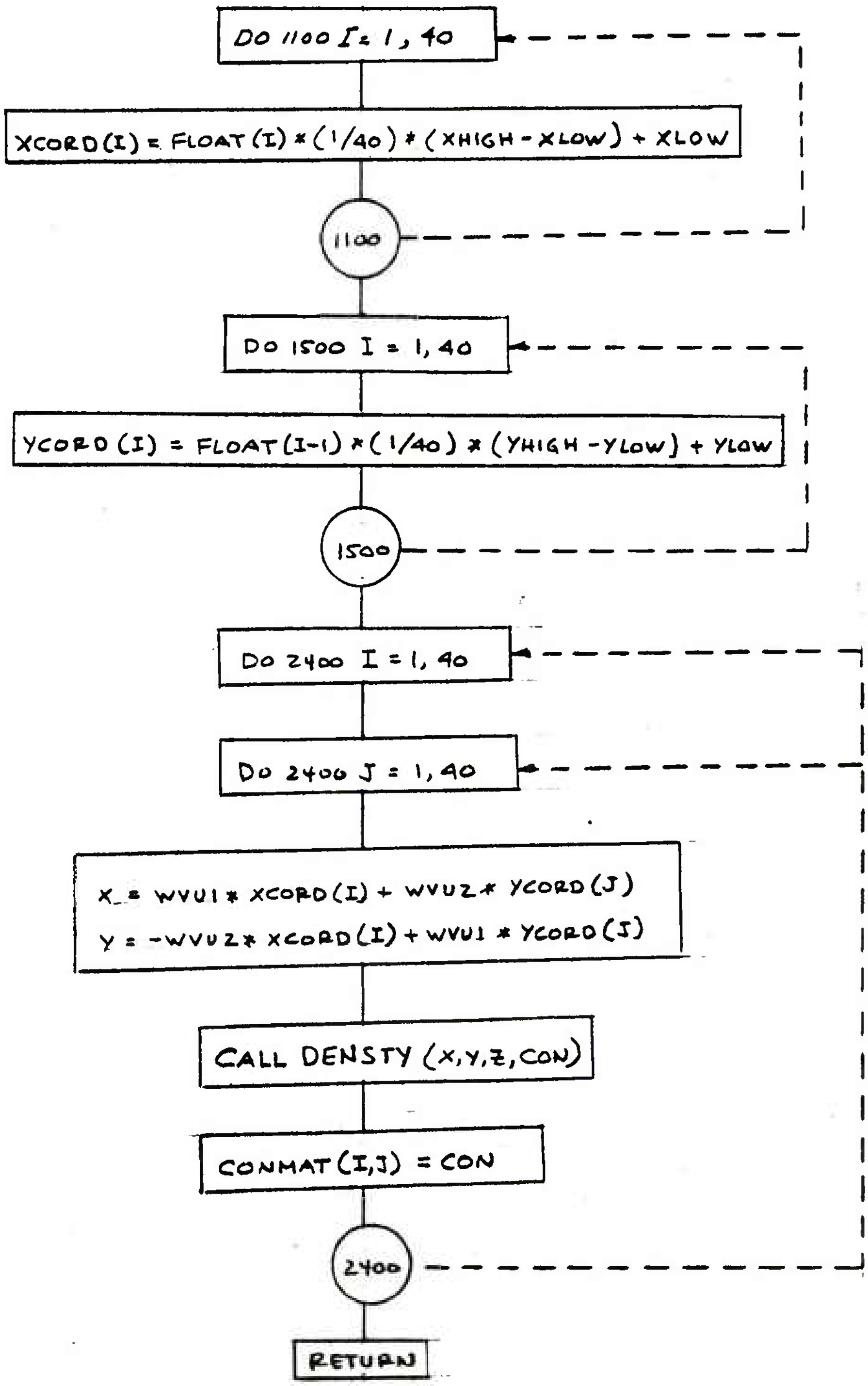
SUBROUTINE SIZE (XLOW, XHIGH, YLOW, YHIGH)



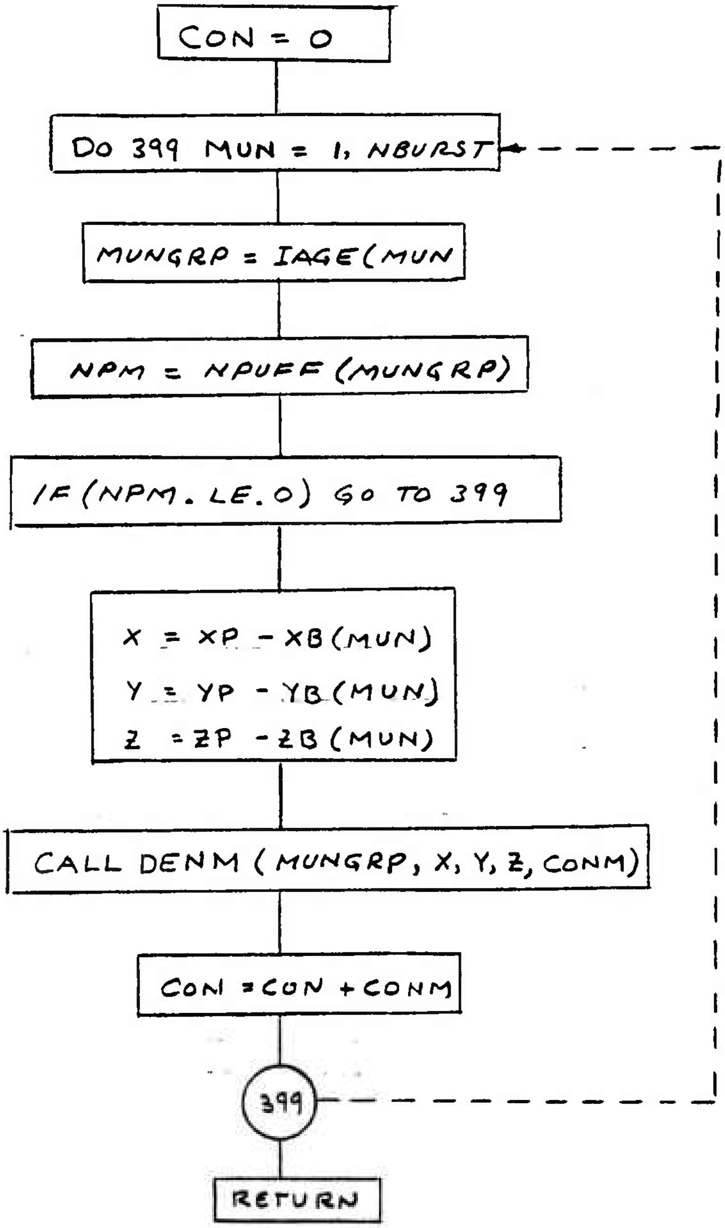
SUBROUTINE MXMIN (YTLINE, XMIN, XMAX)



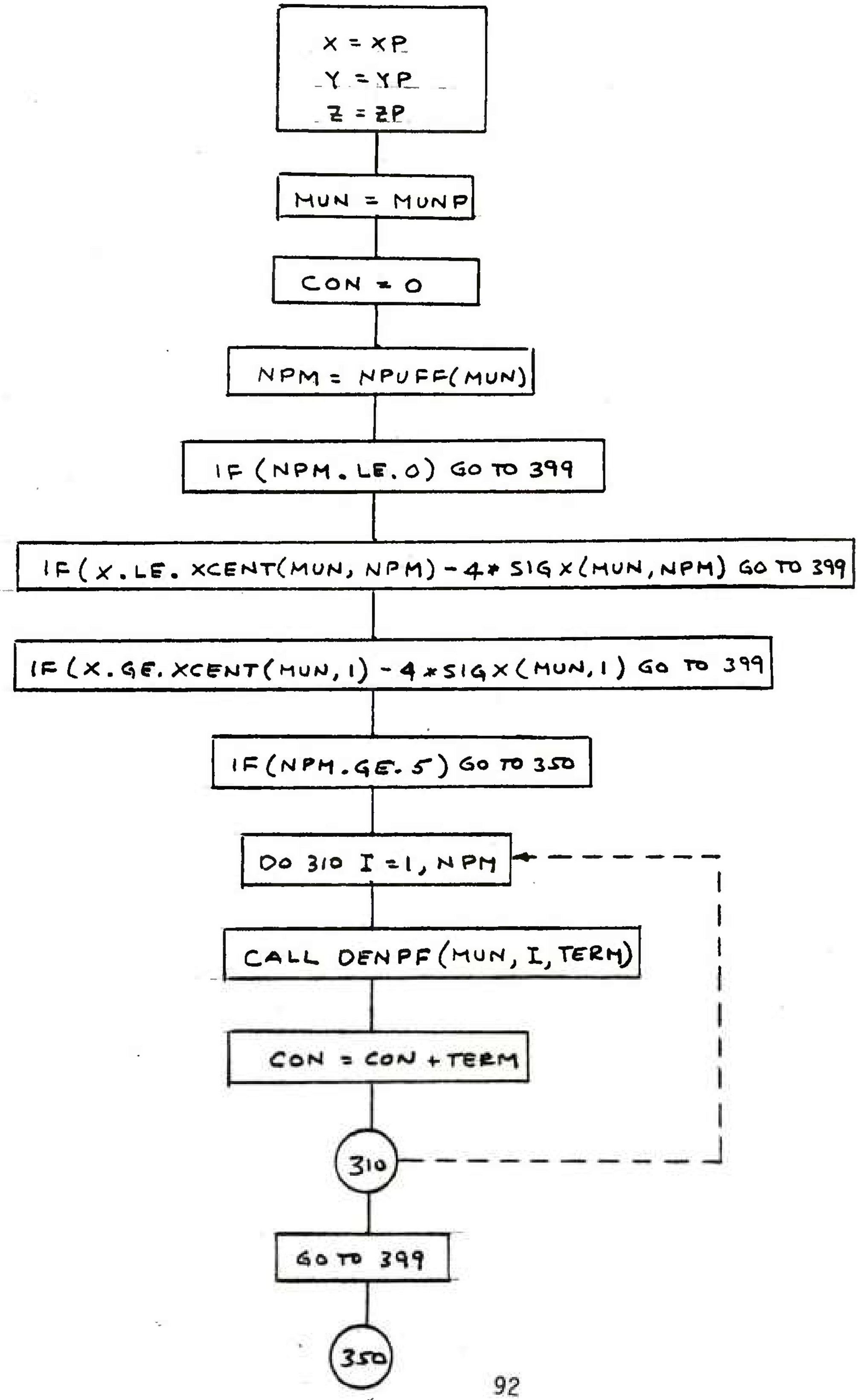
SUBROUTINE MATCON (Z, XLOW, XHIGH, YLOW, YHIGH)



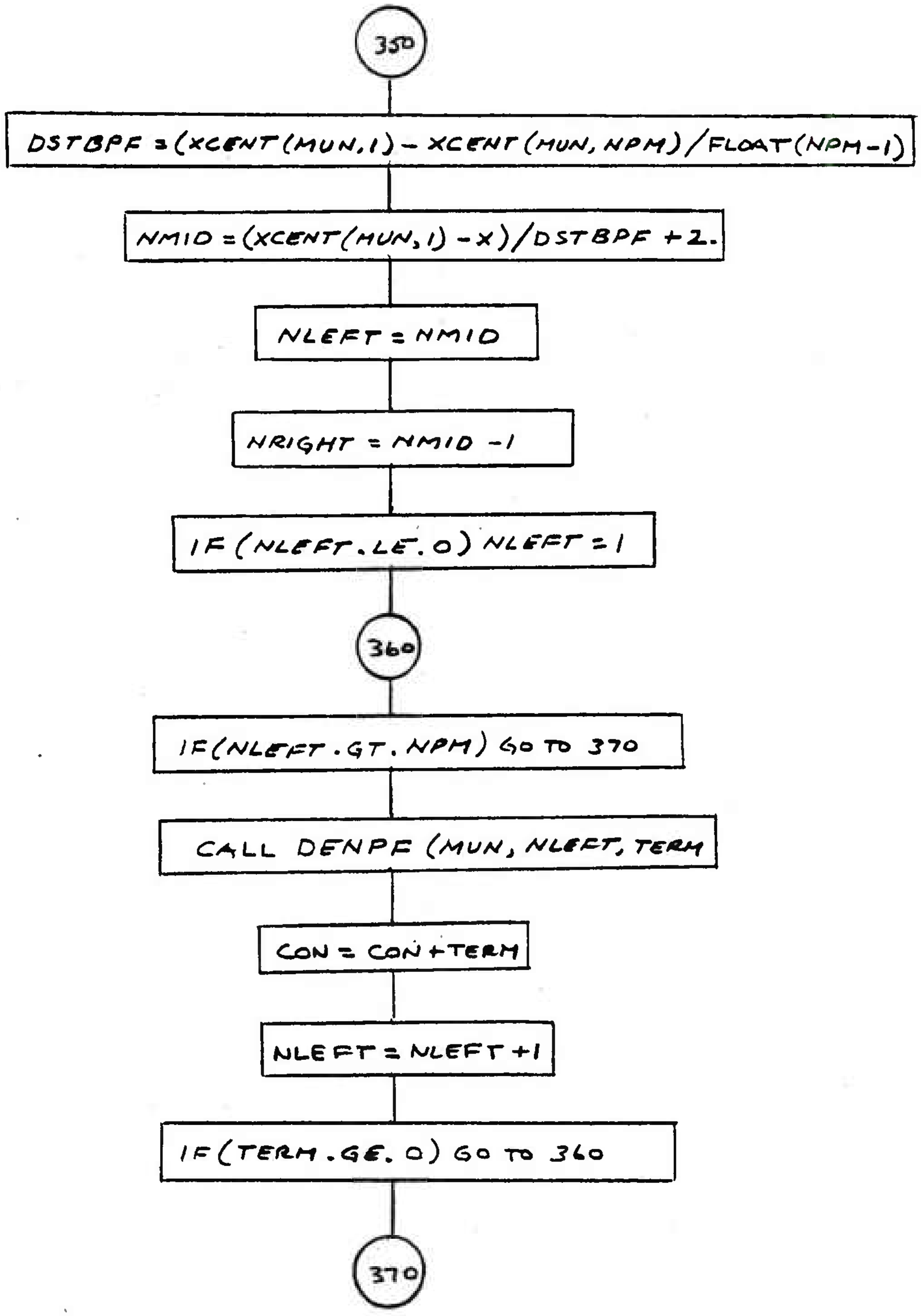
SUBROUTINE DENSTY (XP, YP, ZP, CON)



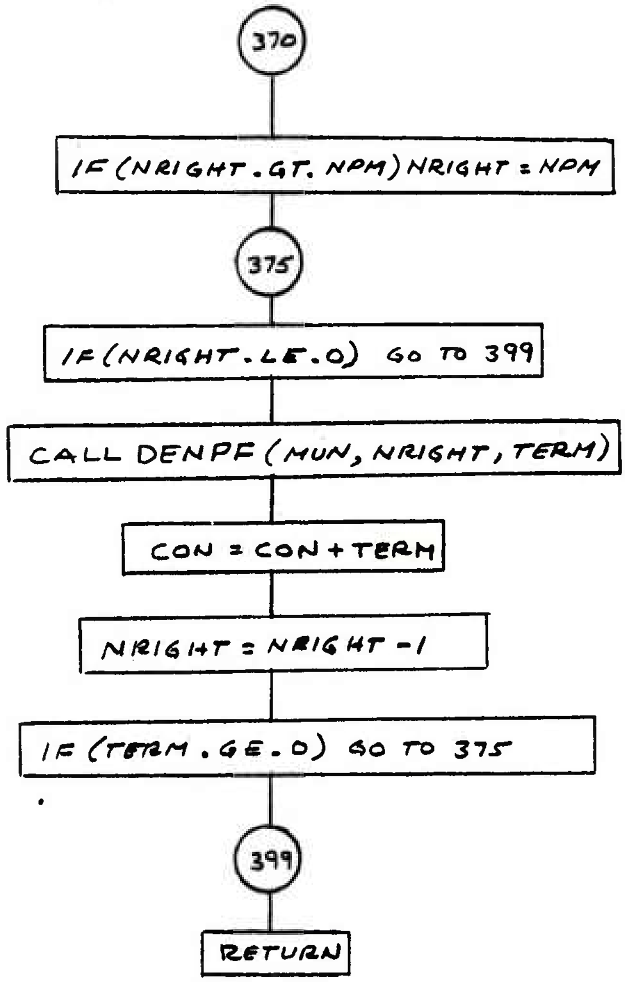
SUBROUTINE DENM (MUNP, XP, YP, ZP, CON)



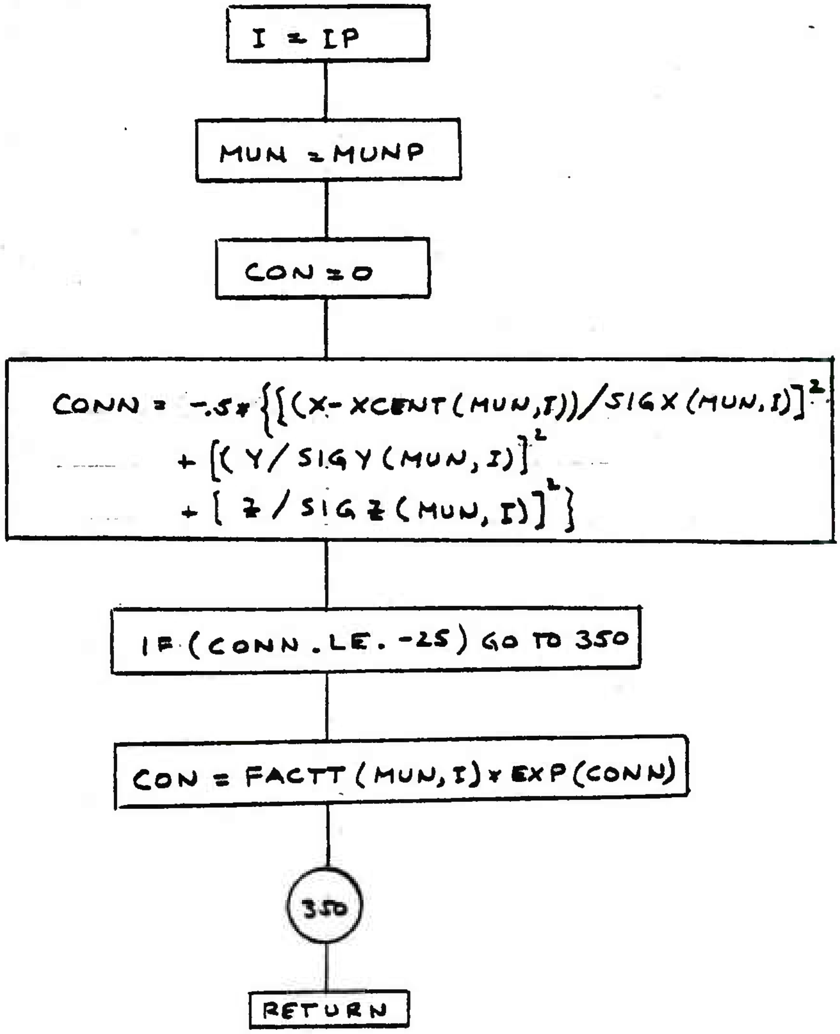
SUBROUTINE DENM (CONTINUED)



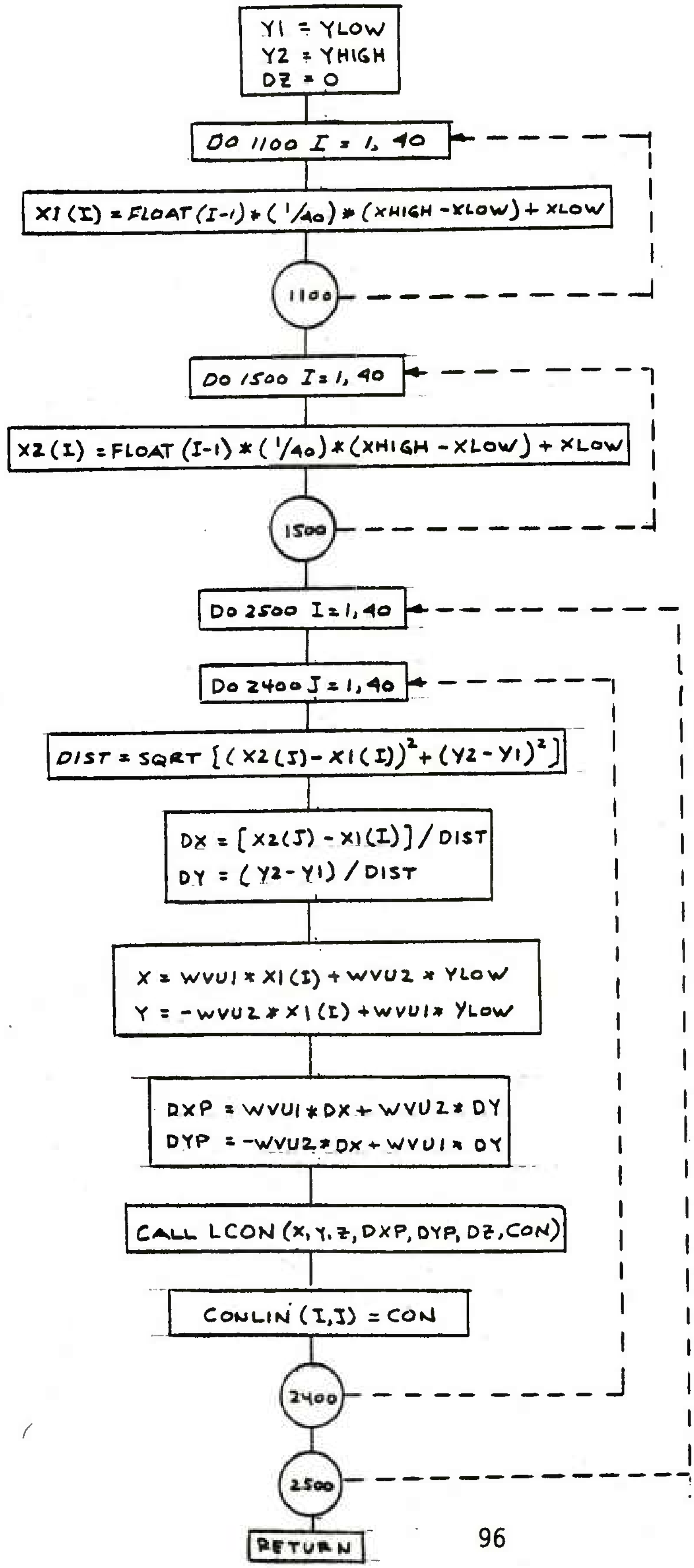
SUBROUTINE DENM (CONTINUED)



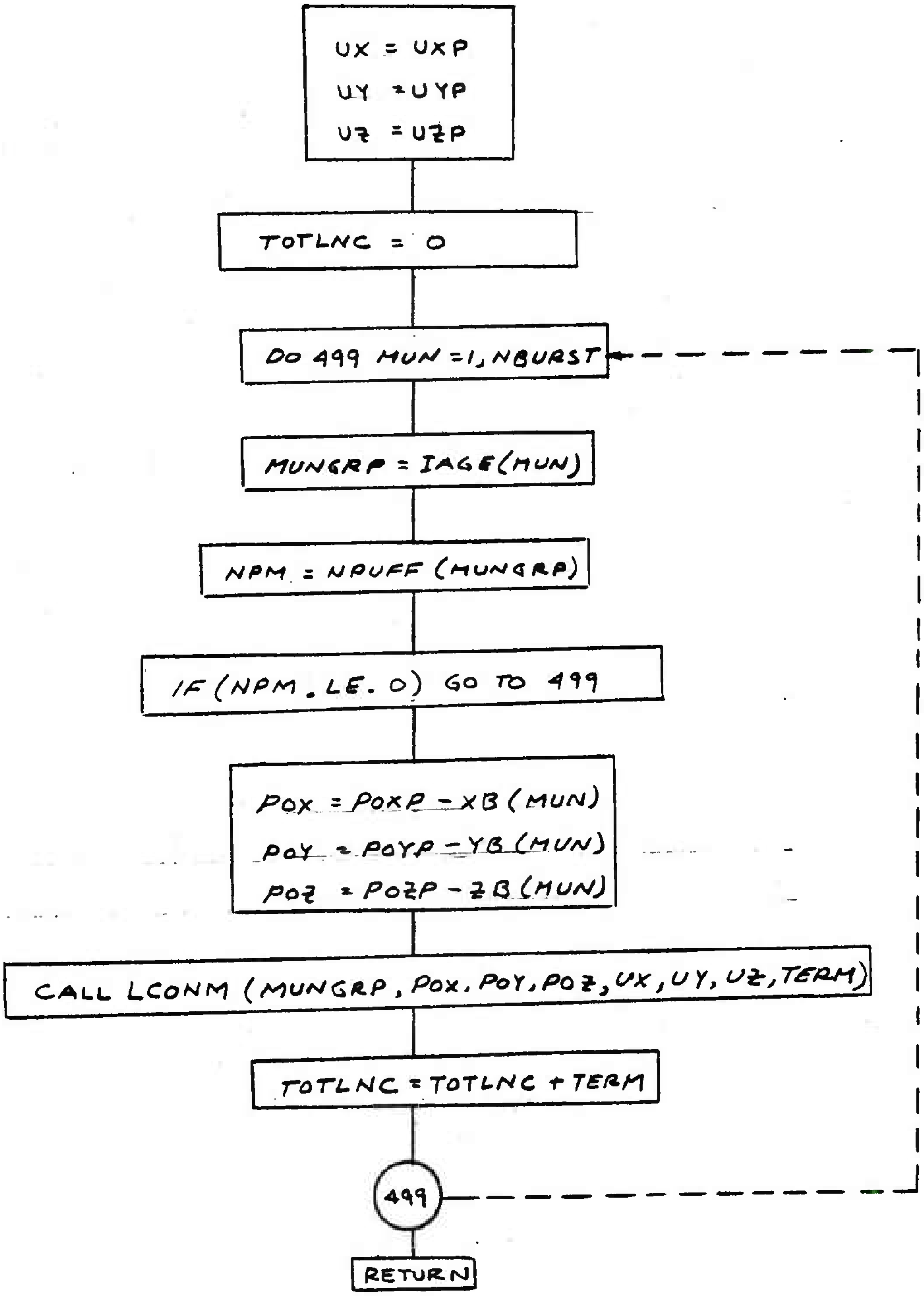
Subroutine DENPF (MUNP, IP, CON)



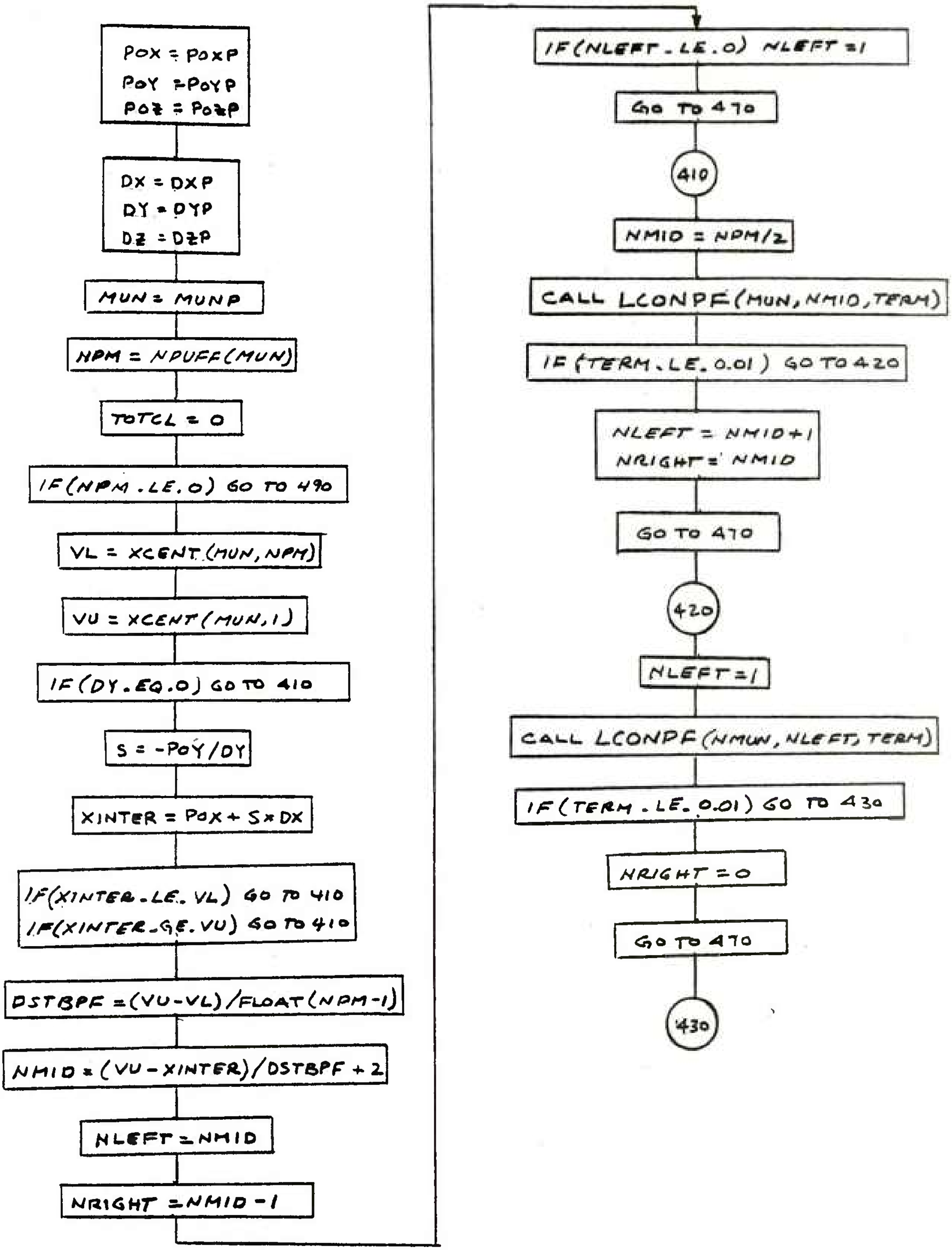
Subroutine MATCL (Z, XLOW, XHIGH, YLOW, YHIGH)



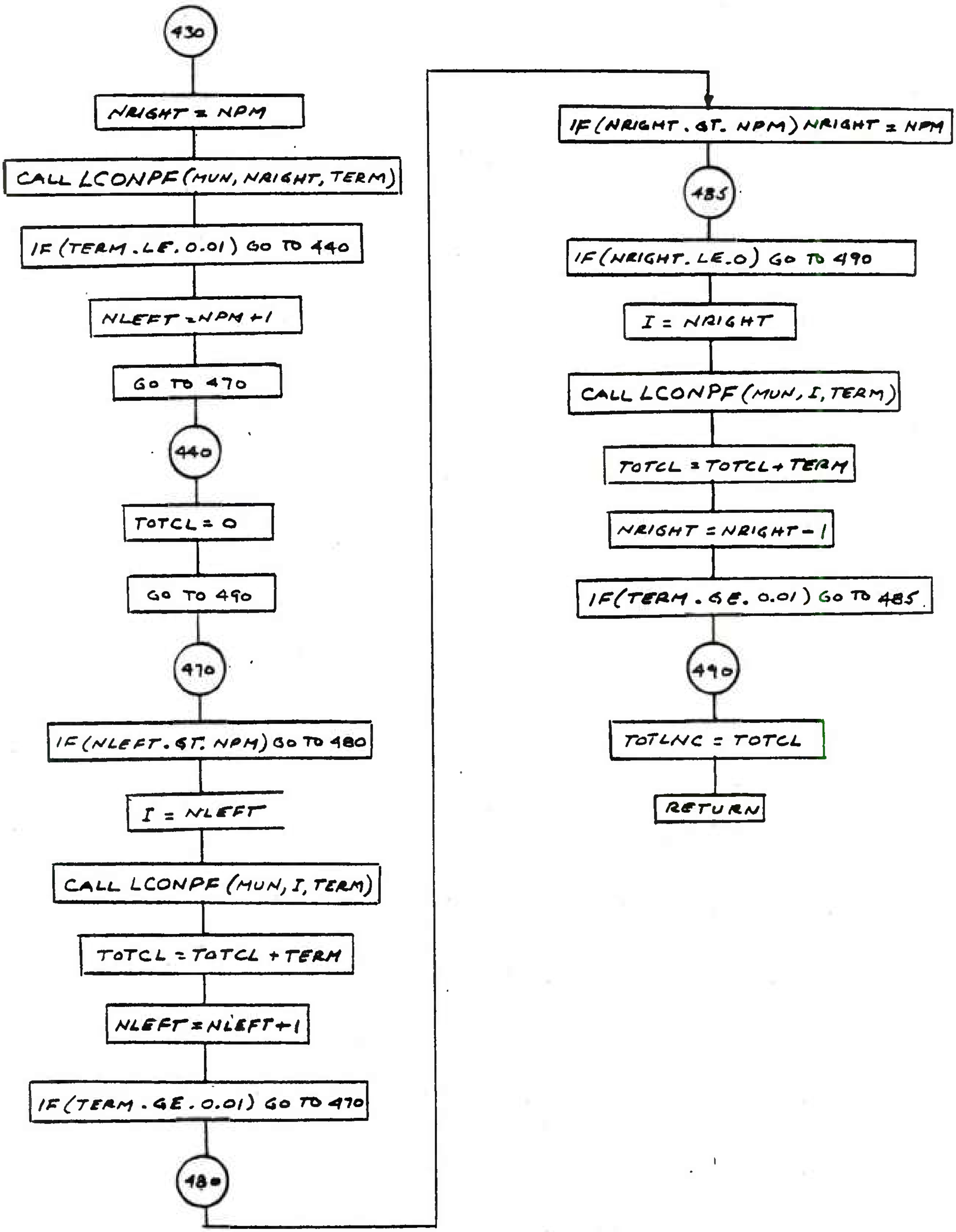
SUBROUTINE LCON (POXP, POYP, POZP, UXP, UYP, UZP, TOTLNC)



SUBROUTINE LCONM (MUNP, POXP, POYP, POZP, DXP, DYP, DZP, TOTLNC)



SUBROUTINE LCONM (CONTINUED)



SUBROUTINE LCONPF (MUNP, IP, PFLNC)

I = IP

MUN = MUNP

PFLNC = 0

$$A = [DX/SIGX(MUN,I)]^2 + [DY/SIGY(MUN,I)]^2 + [DZ/SIGZ(MUN,I)]^2$$

$$B = DX*(POX-XCENT(MUN,I)/SIGX(MUN,I)^2 + DY*POY/SIGY(MUN,I)^2 + DZ*POZ/SIGZ(MUN,I)^2$$

S = - B/A

PMX = POX + S * DX

PMY = POY + S * DY

PMZ = POZ + S * DZ

$$C = [(PMX-XCENT(MUN,I))/SIGX(MUN,I)]^2 + [PMY/SIGY(MUN,I)]^2 + [PMZ/SIGZ(MUN,I)]^2$$

CC = -.5 * C

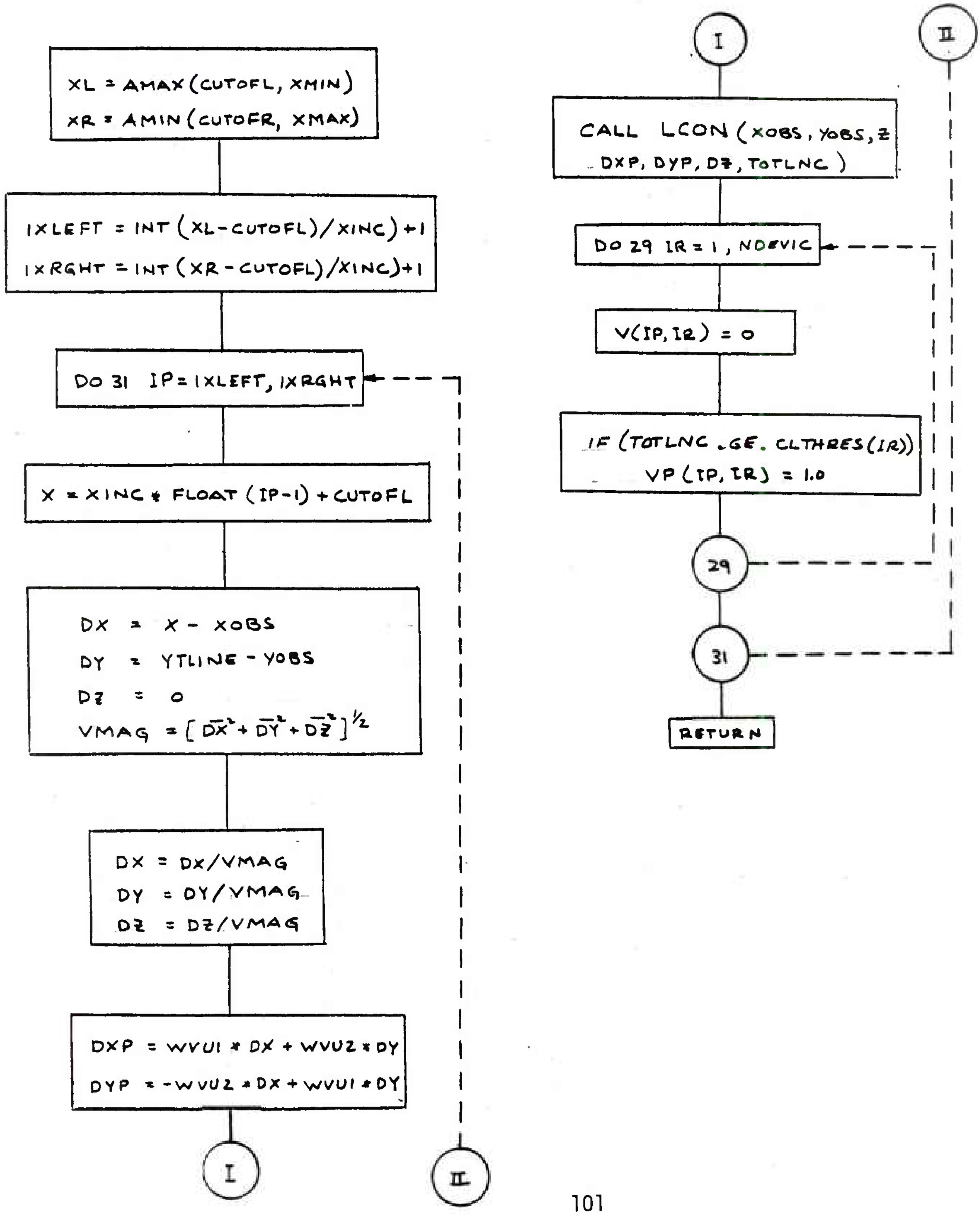
IF (CC .LE. -25) GO TO 399

$$PFLNC = FACTT(MUN,I) * 2.5066 * EXP(CC)/A^{1/2}$$

399

RETURN

SUBROUTINE CALC (XOBS, YOBS, Z, YTLINE, XMIN, XMAX, NDEVIC, CLTHRS)



SUBROUTINE ENDPTS (XMIX, XMAX, IT)

IF (XMAX-XMIN . LE. 1.0) GO TO 25

EXTXL(IT) = EXTXL(IT) + XMIN

EXTXLS(IT) = EXTXLS(IT) + XMIN * XMIN

EXTXR (IT) = EXTXR(IT) + XMAX

EXTXRS (IT) = EXTXRS (IT) + XMAX * XMAX

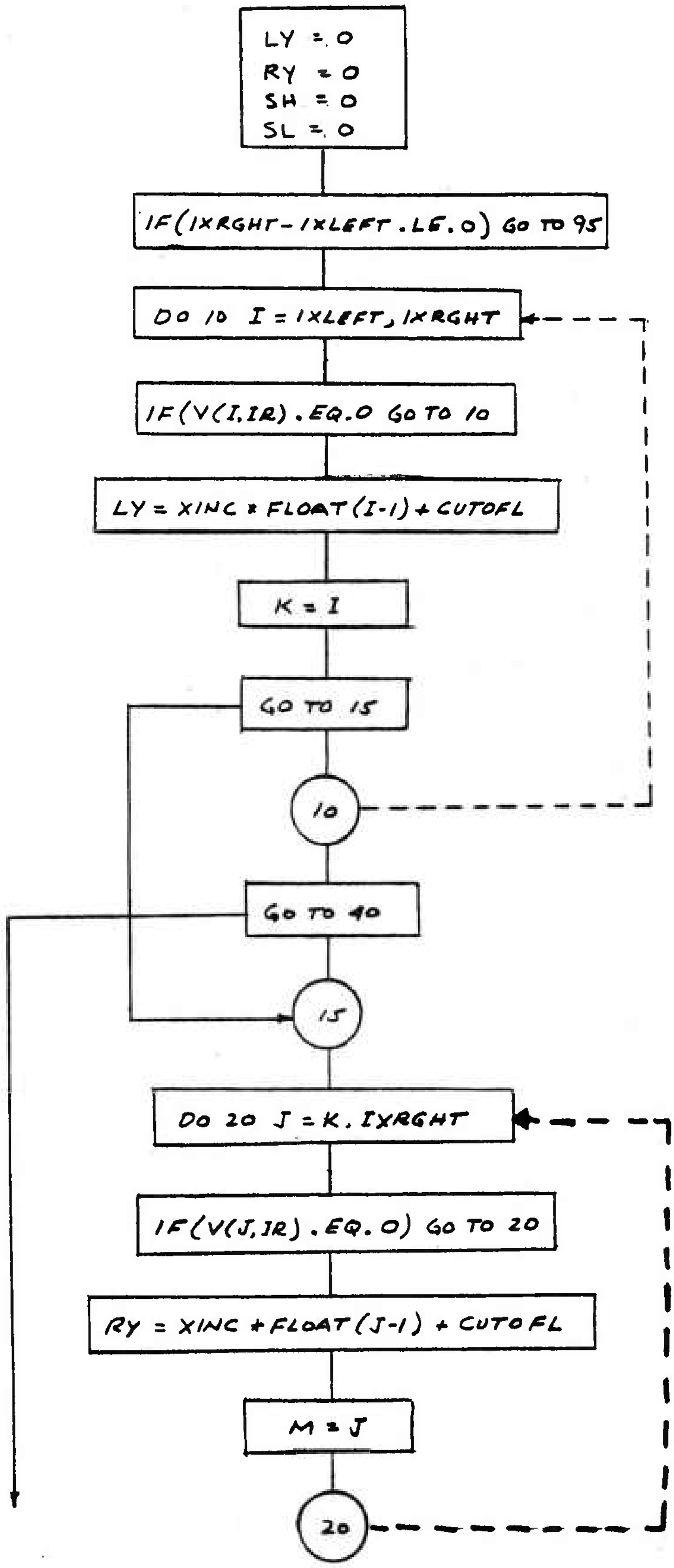
RETURN

25

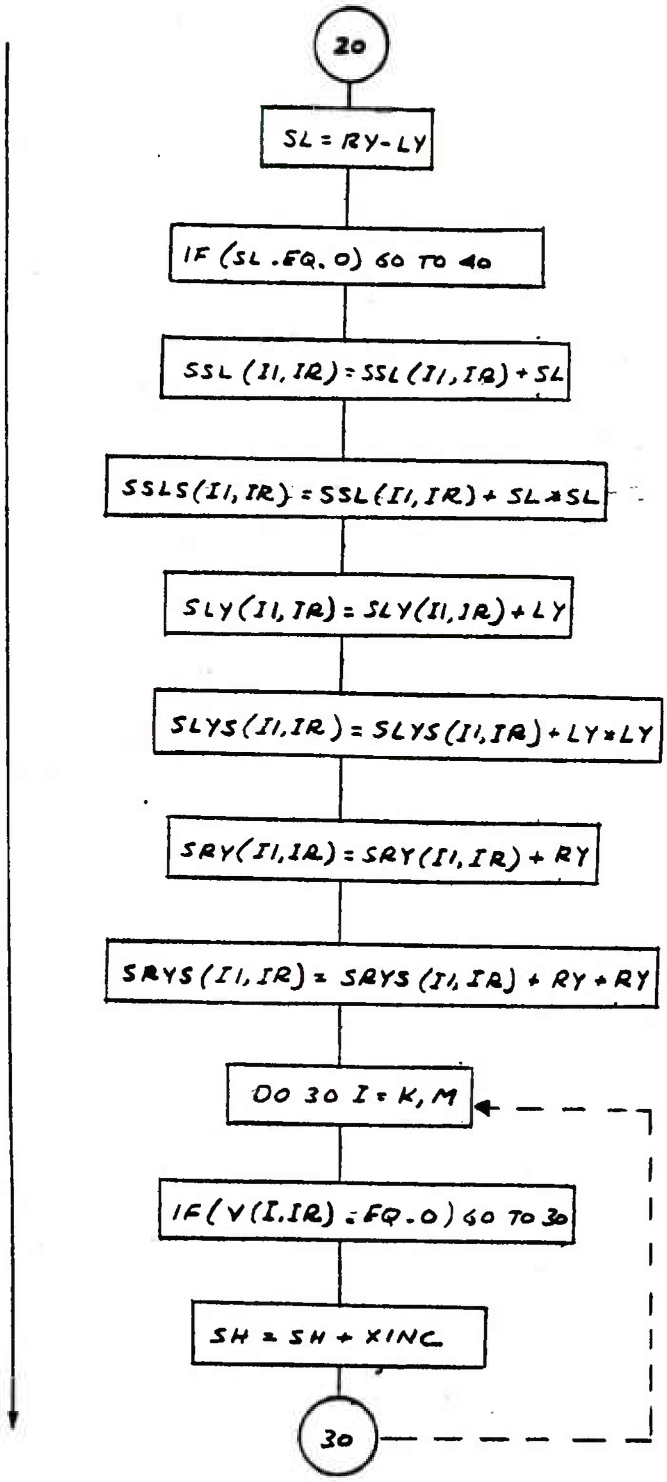
NIX(IT) = NIX(IT)+1

RETURN

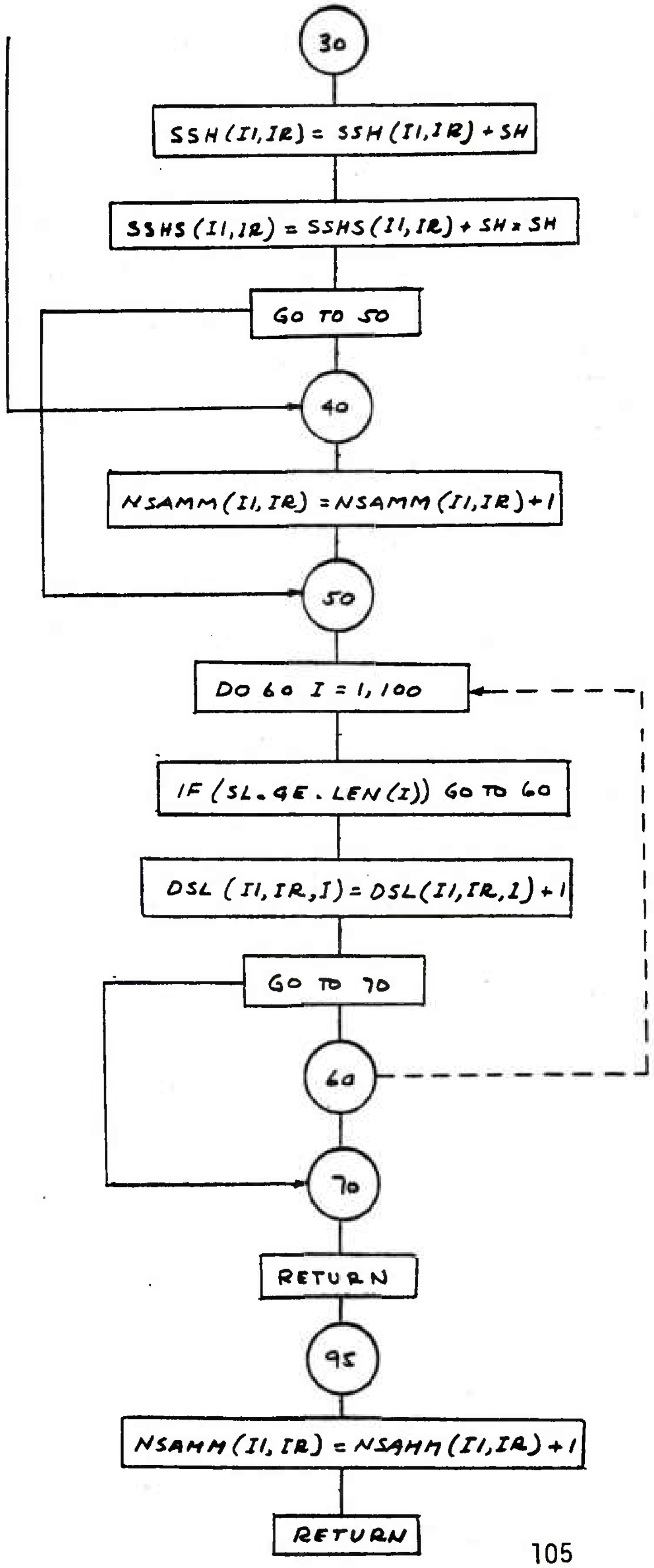
SUBROUTINE VEVAL (I1, IR)



SUBROUTINE VEVAL (CONT.)



SUBROUTINE VEVAL (CONT.)



SUBROUTINE CALPRT (NSAMP, NT, SNAP, NVOL, VT, NDEVIC)

```
DO 505 I = 1, NT
DO 520 IR = 1, NDEVIC
NSAMP1 = NSAMP - NSAMM (I,IR)
IF (NSAMP1 .LE. 1) GO TO 515

SLYS(I,IR) = [(SLYS (I,IR) - SLY (I,IR)^2/NSAMP1) /(NSAMP1 - 1)]^1/2
SRYS(I,IR) = [(SRYS (I,IR) - SRY (I,IR)^2/NSAMP1) /(NSAMP1 - 1)]^1/2
SSHS(I,IR) = [(SSHS (I,IR) - SSH (I,IR)^2/NSAMP1) /(NSAMP1 - 1)]^1/2
SLY (I,IR) = SLY (I,IR) / NSAMP1
SRY (I,IR) = SRY (I,IR) / NSAMP1
SSH (I,IR) = SSH (I,IR) / NSAMP1
SSLS(I,IR) = [(SSLS(I,IR) - SSL (I,IR)^2/NSAMP) /(NSAMP - 1)]^1/2

(515)
SSL(I,IR) = SSL(I,IR)/NSAMP
(520)
NIX1 = NSAMP - NIX (I)
IF(NIX1 .LE. 1) GO TO 505
(I)
```

(I)

$EXTXLS(I) = \left[\left(EXTXLS(I) - EXTXL(I)^2 / NIXI \right) / (NIXI - 1) \right]^{1/2}$

$EXTXRS(I) = \left[\left(EXTXRS(I) - EXTXR(I)^2 / NIXI \right) / (NIXI - 1) \right]^{1/2}$

EXTXL(I) = EXTXL(I) / NIXI

EXTXR(I) = EXTXR(I)/NIXI

(505)

DO 530 I = 1, NT

STIME = I * SNAP

J = MIN0 (NVOL, 1+INT (STIME / VT))

(530)

RETURN

Next page is blank.

APPENDIX E

SOURCE LISTING OF PROGRAM WP MUNITIONS

Next page is blank.

```
      PROGRAM SMOKEWP(INPUT,OUTPUT,TAPE4=INPUT)

      DIMENSION TITLE(8),CATTN(3),THRES(3),CLTHRS(3)

      COMMON /REL/ XB(60),YB(60),ZB(60),TB(60),NBURST
      COMMON /REL2/XCENTP(60),YCENTP(60)

      COMMON /WPSCR/ SIGX(50),SIGY(50),SIGZ(50)
      COMMON /WPSCR/ FACTT(50)

      COMMON /WPCLD/ FSIGX,FSIGY,FSIGZ
      COMMON /WPCLD/ DSIGX,DSIGY,DSIGZ
      COMMON /WPCLD/ ASIGX,ASIGY,ASIGZ
      COMMON /WPCLD/ ESIGX,ESIGY,ESIGZ
      COMMON /WPCLD/ WNDSPD,FACTOR

      COMMON /ABS/XBA(60),YBA(60),ZBA(60),TBA(60),NBRSTA
      COMMON /ABS2/ XCENT(60),YCENT(60),ZCENT(60)
      COMMON /ABS3/ SIGXA(50),SIGYA(50)

      COMMON /TMSCLE/INSNAP,INVOL,DT
      COMMON /PLACE/ SIGBR,SIGBD,SIGAR,SIGAD,REL
      COMMON /PLACE/ XIDEAL(6),YIDEAL(6),ZIDEAL(6)
      COMMON /WNDDIR/WVU1,WVU2
      COMMON /AGEIND/ IAGE(60)
      COMMON /FLDSIT/ CUTOFL,CUTOFR,XINC

      DATA FSIGX/.1522/,FSIGY/3.41/,FSIGZ/1.35/
      DATA DSIGX/1.0/,DSIGY/100.0/,DSIGZ/20.0/
      DATA ZCENT/50*0.0/

  500 CONTINUE

      READ(4,2) TITLE
    2 FORMAT(8A10)
      PRINT 10,TITLE
   10 FORMAT(1H1,8A10)

      READ(4,1)U,ALPHA,BETA
    1 FORMAT(3F10.5)

      PRINT 4,U,ALPHA,BETA
    4 FORMAT(1H0,* WIND SPEED(X DIRECT) =*,F10.5,
     *       *ALPHA =*,F10.5,*BETA =*,F10.5)

      READ(4,11) WVU1,WVU2
   11 FORMAT(2F10.5)
      PRINT 12,WVU1,WVU2
   12 FORMAT(1H0,*WIND VECTOR DIRECTION COSINES X =*,F10.5,*Y =*,F10.5)

      READ(4,6) YIELD,QMUN,SIGXS,SIGYS,SIGZS
    6 FORMAT(F10.5,E10.5, 3F10.5)
      PRINT 3,YIELD,QMUN, SIGXS,SIDYS,SIGZS
    3 FORMAT(*0*,* YIELD =*,F10.5,*QUAN OF MUN (ADJ) =*,E10.5,
     1           * SOURCE SIGMAS =*,3F10.5)

      READ(4,5) NDEVIC
    5 FORMAT(I5)
      READ(4,7) ((CATTN(IR),THRES(IR)),IR=1,NDEVIC)
    7 FORMAT(6F10.5)

      CALL STCL(QMUN,YIELD,SIGXS,SIGYS,SIGZS,ALPHA,BETA,U)

      DO 159 I=1,NDEVIC
      CLTHRS(I)=ALOG(1.0-THRES(I))/(-CATTN(I))
      PRINT 95,I,CATTN(I),THRES(I),CLTHRS(I)
   95 FORMAT(*0*,I3,* COEF ATTN =*,F10.5,
     1              * THRESHOLD =*,F10.2,
```

```
      2                 ' CL THRES  =',F10.2)
  159 CONTINUE
C ******************** MUNITIONS PLACEMENT HERE
      READ(4,50) NRPV
   50 FORMAT(I10)
      PRINT 14,NRPV
   14 FORMAT(1H0,'NUMBER OF ROUNDS PER VOLLEY ',I4/
     2       1H0,'IDEAL IMPACT POINTS '/
     *           '0',' NO.',6X,'X',11X,'Y',11X,'Z')

      DO 2500 J=1,NRPV
      READ(4,55) XIDEAL(J),YIDEAL(J),ZIDEAL(J)
   55 FORMAT(3F10.5)
      PRINT 16,J,XIDEAL(J),YIDEAL(J),ZIDEAL(J)
   16 FORMAT(' ',I4,6(1X,F10.2,1X))
 2500 CONTINUE

      READ(4,21) NSAMP,NVOL,NT,DT,DL
   21 FORMAT(3I5,2F5.0)
      PRINT 23,NSAMP,NVOL,NT,DT,DL
   23 FORMAT(1H0,'TOTAL NUMBER OF SAMPLES =',I5,/
     1        1X,'TOTAL NUMBER OF VOLLEYS =',I5,/
     2        1X,'TOTAL NUMBER OF TIMES   =',I5,/
     3        1X,'TIME INCREMENTS =',F5.0,'SEC.',/
     4        1X,'LINE INCREMENTS =',F5.0)

      READ(4,18) VT,SNAP
   18 FORMAT(2F10.1)
      PRINT 19,VT,SNAP
   19 FORMAT(1H0,'TIME BETWEEN VOLLEYS =',F10.1,/
     1        1X,'SNAP TIME INCREMENT =',F10.1)

      READ(4,22) SIGBR,SIGBD,SIGAR,SIGAD,REL
   22 FORMAT(5F10.4)
      PRINT 24,SIGBR,SIGBD,SIGAR,SIGAD,REL
   24 FORMAT(1H0,'BR =',F5.0,5X,'BD =',F5.0,/
     1        1X,'AR =',F5.0,5X,'AD =',F5.0,/
     2        1X,'REL =',F5.0)

      NBRSTA=NVOL*NRPV
      INSNAP=SNAP/DT
      INVOL=VT/DT
      CALL FRMCLD(NT,DT,NVOL,NS)
      XOBS=0.0
      YOBS=0.0
      READ(4,20) YTLINE,CUTOFL,CUTOFR
   20 FORMAT(3F10.2)
      READ(4,20) XT,YT
      READ(4,20) XINC
      Z=1.5

      PRINT 73,XOBS,YOBS
   73 FORMAT(1H0,'OBSERVER COORD =',2F10.1)
      PRINT 71,XT,YT
   71 FORMAT(1H0,'AIMPOINT COORD =',2F10.1)
      PRINT 72,YTLINE
   72 FORMAT(1H0,'DISTANCE TO LINE =',F10.1)
      PRINT 74,XINC
   74 FORMAT(1H0,' X-INCREMENT =',F5.1)
      PRINT 75,CUTOFL,CUTOFR
   75 FORMAT(1H0,' 90 DEGREE SECTOR =',F7.0,' , ',F7.0)
      PRINT 35,Z
   35 FORMAT('0','HEIGHT ABOVE CENTROID ',F10.5)

      CALL CLEAR (DL)

      DO 28 KI=1,NSAMP
```

```
      PRINT 8,KI
    8 FORMAT('1',' DATA FOR SAMPLE NUMBER ',I5)

      CALL MPLACE(NVOL,NRPV,VT,XT,YT)
      CALL CONBST

      PRINT 42
   42 FORMAT('0',' MUNITION BURSTS AT ')
      PRINT 43
   43 FORMAT('0',11X,'LOCATION',11X,5X,7X,'TIME')

      I=0
      DO 199 IV=1,NVOL
      DO 199 IRD=1,NRPV
      I=I+1

      PRINT 48,I,IV,IRD,XBA(I),YBA(I),ZBA(I),TBA(I)
   48 FORMAT('0', 3(1X,I3,1X),3(1X,F8.2,1X),6X,F8.2)
  199 CONTINUE

      DO 1100 MT=1,NT

      CALL TIME(MT,U)

C     PRINT 32
   32 FORMAT('0',' MUNITION CHARACTERISTICS    ')
C     PRINT 33
   33 FORMAT('0',11X,'LOCATION',11X,5X,14X,'SIZE')
      DO 350 IMUN=1,NBRSTA
      MG=IAGE(IMUN)
      IF(MG.LE.0) GO TO 360
C     PRINT 38,IMUN,XCENT(IMUN),YCENT(IMUN),YCENT(IMUN),
C    *          SIGXA(IMUN),SIGYA(IMUN),SIGZ(MG)
   38 FORMAT('0',1X,I2,2X,3(1X,F8.2,1X),5X,3(1X,F8.2,1X))
  350 CONTINUE
  360 CONTINUE

      CALL SIZE(XLOW,XHIGH,YLOW,YHIGH)

      PRINT 86,XLOW,XHIGH,YLOW,YHIGH
   86 FORMAT(1H0,' SCREEN DIMENSIONS  X=',2F7.2,2X,'Y=',2F7.2)

      CALL MXMIN(YTLINE,XMIN,XMAX)

      PRINT 9,XMIN,XMAX
    9 FORMAT(1H ,'XMIN =',F10.2,2X,'XMAX =',F10.2)

      YSLINE=500.0

C     CALL MATCON(Z,XLOW,XHIGH,YLOW,YHIGH)
C     CALL MATCL(Z,XLOW,XHIGH,YOBS,YSLINE)
      CALL CALC(XOBS,YOBS,Z,YTLINE,XMIN,XMAX,NDEVIC,CLTHRS)
      CALL ENDPTS(XMIN,XMAX,MT)

      DO 299 IR=1,NDEVIC
      CALL VEVAL (MT,IR)
  299 CONTINUE
 1102 CONTINUE
   28 CONTINUE
      CALL CALPRT (NSAMP,NT,SNAP,NVOL,VT,NDEVIC)
      STOP
      END
```

```
      SUBROUTINE CLEAR(DL)

      COMMON /STATS/ SSL(50,3),SSLS(50,3),SLY(50,3),SLYS(50,3),
     1 SRY(50,3),SRYS(50, 3),SSH(50,3),SSHS(50,3),NSAMM(50,3),LEN(100),
     2 DSL(50,3,100)
      COMMON /STAT2/ EXTXL(50),EXTXLS(50),EXTXR(50),EXTXRS(50),NIX(50)

      DO 800 I=1,100
  800 LEN(I)=DL*I
      DO 120 I=1,50
      NIX(I)=0
      EXTXL(I)=0.0
      EXTXLS(I)=0.0
      EXTXR(I)=0.0
      EXTXRS(I)=0.0
      DO 110 IR=1,3
      NSAMM(I,IR)=0
      SSL(I,IR)=0.0
      SSLS(I,IR)=0.0
      SLY(I,IR)=0.0
      SLYS(I,IR)=0.0
      SRY(I,IR)=0.0
      SRYS(I,IR)=0.0
      DO 105 K=1,100
  105 DSL(I,IR,K)=0.0
      SSH(I,IR)=0.0
  110 SSHS(I,IR)=0.0
  120 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MPLACE(NVOL,NRPV,VT,XT,YT)

      COMMON /ABS/  XB(60),YB(60),ZB(60),TB(60),NBURST
      COMMON /PLACE/ SIGBR,SIGBD,SIGAR,SIGAD,REL
      COMMON /PLACE/ XIDEAL(6),YIDEAL(6),ZIDEAL(6)

      CALL NORAN(R,SIGAR,E,SIGAD)
      XCA=XT+E
      YCA=YT+R
      DO 32 I=1,NVOL
      BRSTTM= (I-1)*VT
      DO 30 J=1,NRPV
      IF(RANF(DUM).GT.REL) GO TO 30
      CALL NORAN(R,SIGBR,E,SIGBD)
      XCRD=XCA+XIDEAL(J)+E
      YCRD=YCA+YIDEAL(J)+R

      MUNNO=(I-1)*NRPV + J

      XB(MUNNO)=XCRD
      YB(MUNNO)=YCRD
      ZB(MUNNO)=0.0
      TB(MUNNO)=BRSTTM
   30 CONTINUE
   32 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONBST

      COMMON /REL/ XB(60),YB(60),ZB(60),TB(60),NBURST
      COMMON /REL2/XCENTP(60),YCENTP(60)

      COMMON /ABS/XBA(60),YBA(60),ZBA(60),TBA(60),NBRSTA
      COMMON /WNDDIR/HVU1,HVU2

      NBURST=NBRSTA
      DO 100 I=1,NBRSTA
      XB(I)=HVU1*XBA(I)+HVU2*YBA(I)
      YB(I)=-HVU2*XBA(I)+HVU1*YBA(I)
      ZB(I)=ZBA(I)
      TB(I)=TBA(I)
      YCENTP(I)=YB(I)
100   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE NORAN(R,SR,O,SD)
    1 X=RANF(DUM)
      IF(X.LE.0.0) GO TO 1
      A=SQRT(-2.0*ALOG(X))
      B=6.283185307 18*RANF(DUM)
      R=A*SR*SIN(B)
      D=A*SD*COS(B)
      RETURN
      END
```

```
      SUBROUTINE SIZE(XLL,XUL,YLL,YUL)
      COMMON /ABS/XBA(60),YBA(60),ZBA(60),IBA(60),NBRSTA
      COMMON /ABS2/ XCENT(60),YCENT(60),ZCENT(60)
      COMMON /ABS3/ SIGXA(50),SIGYA(50)
      COMMON /AGEIND/ IAGE(60)
      XLL= 1.E6
      XUL=-1.E6
      YLL= 1.E6
      YUL=-1.E6
      DO 89 MUN=1,NBRSTA
      MUNGRP=IAGE(MUN)
      IF(MUNGRP.LE.0) GO TO 99
      TEST=XCENT(MUN)-4.0*SIGXA(MUN)
      IF(TEST.LT.XLL) XLL=TEST
      TEST=XCENT(MUN)+4.0*SIGXA(MUN)
      IF(TEST.GT.XUL) XUL=TEST
      TEST=YCENT(MUN)-4.0*SIGYA(MUN)
      IF(TEST.LT.YLL) YLL=TEST
      TEST=YCENT(MUN)+4.0*SIGYA(MUN)
      IF(TEST.GT.YUL) YUL=TEST
   89 CONTINUE
   99 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MXMIN(YTLINE,XMIN,XMAX)

      COMMON /WPSCR/ SIGX(50),SIGY(50),SIGZ(50)
      COMMON /ABS/XBA(60),YBA(60),ZBA(60),TBA(60),NBRSTA
      COMMON /ABS2/ XCENT(60),YCENT(60),ZCENT(60)
      COMMON /AGEIND/ IAGE(60)

      RATMX=-1.0E+10
      RATMN=1.0E+10
      DO 1 I=1,NBRSTA
      M=IAGE(I)
      IF(M.LE.0) GO TO 2
      IF(YCENT(I).EQ.0.0) GO TO 1
      RATL=(XCENT(I)-4.*SIGX(M))/YCENT(I)
      IF(RATL.LT.RATMN) RATMN=RATL
      RATN=(XCENT(I)+4.*SIGX(M))/YCENT(I)
      IF(RATN.GT.RATMX) RATMX=RATN
    1 CONTINUE
    2 CONTINUE
      XMIN=YTLINE*RATMN
      XMAX=YTLINE*RATMX
      RETURN
      END
```

```
      SUBROUTINE CALC(XOBS,YOBS,Z,YTLINE,XMIN,XMAX,NDEVIC,CLTHRS)

      COMMON /WNDDIR/WVU1,WVU2
      COMMON /FLDSIT/ CUTOFL,CUTOFR,XINC
      COMMON /LINES/ V(1001,3),IXLEFT,IXRGHT

      DIMENSION CLTHRS(3)

      XL=AMAX1(CUTOFL,XMIN)
      XR=AMIN1(CUTOFR,XMAX)
      IXLEFT=INT((XL-CUTOFL)/XINC) + 1
      IXRGHT=INT((XR-CUTOFL)/XINC) + 1
      DO 31 IP=IXLEFT,IXRGHT
      X=XINC * FLOAT(IP-1) + CUTOFL
      DX=X-XOBS
      DY=YTLINE-YOBS
      DZ=0.0
      VMAG=SQRT(DX**2+DY**2+DZ**2)
      DX=DX/VMAG
      DY=DY/VMAG
      DZ=DZ/VMAG
      DXP=WVU1*DX+WVU2*DY
      DYP=-WVU2*DX+WVU1*DY
      CALL LCON(XOBS,YOBS,Z,DXP,DYP,DZ,TOTLNC)
      DO 29 IR=1,NDEVIC
      V(IP,IR)=0.0
      IF(TOTLNC.GE.CLTHRS(IR)) V(IP,IR)=1.0
   29 CONTINUE
   31 CONTINUE
      DO 35 IR=1,NDEVIC
C     PRINT 34,(V(I,IR),I=IXLEFT,IXRGHT)
   34 FORMAT('0',(25F4.1,/))
   35 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE VEVAL(II,IR)

      COMMON /FLDSIT/ CUTOFL,CUTOFR,XINC
      COMMON /LINES/ V(1001,3),IXLEFT,IXRGHT
      COMMON /STATS/ SSL(50,3),SSLS(50,3),SLY(50,3),SLYS(50,3),
     1 SRY(50,3),SRYS(50,3),SSH(50,3),SSHS(50,3),NSAMM(50,3),LEN(100),
     2 DSL(50,3,100)

      INTEGER RY,SH
      LY=0
      RY=0
      SH=0
      SL=0.0

C     LEFT Y-COORD = LY

      IF(IXRGHT-IXLEFT.LE.0) GO TO 95
      DO 10 I=IXLEFT,IXRGHT
      IF(V(I,IR).EQ.0.0) GO TO 10
      LY=XINC*FLOAT(I-1) + CUTOFL
      K=I
      GO TO 15
   10 CONTINUE
      GO TO 40

C     RIGHT Y-COORD = RY

   15 DO 20 J=K,IXRGHT
      IF(V(J,IR).EQ.0.0) GO TO 20
      RY=XINC*FLOAT(J-1) + CUTOFL
      M=J
   20 CONTINUE

C     TOTAL SCREEN LENGTH = SL

      SL=RY-LY
      IF(SL.EQ.0.) GO TO 40
      SSL(II,IR)=SSL(II,IR)+SL
      SSLS(II,IR)=SSLS(II,IR)+SL*SL
      SLY(II,IR)=SLY(II,IR)+LY
      SLYS(II,IR)=SLYS(II,IR)+LY*LY
      SRY(II,IR)=SRY(II,IR)+RY
      SRYS(II,IR)=SRYS(II,IR)+RY*RY

C     INTERIOR HOLE LENGTH = SH
C

      DO 30 I=K,M
      IF(V(I,IR).EQ.1.0) GO TO 30
      SH=SH+XINC
   30 CONTINUE
      SSH(II,IR)=SSH(II,IR)+SH
      SSHS(II,IR)=SSHS(II,IR)+SH*SH
      GO TO 50

C     NSAMM TO BE USED TO MODIFY NSAMP SUCH THAT
C     AVE LY,RY,AND SH WILL BE CALC. GIVEN
C     NON-ZERO SCREEN LENGTH.

   40 NSAMM(II,IR)=NSAMM(II,IR)+1

C     DISTRIBUTION OF SCREEN LENGTH

   50 DO 60 I=1,100
      IF(SL.GE.LEN(I)) GO TO 60
      DSL(II,IR,I)=DSL(II,IR,I)+1.0
      GO TO 70
   60 CONTINUE
```

```
70 CONTINUE
   RETURN
95 CONTINUE
   NSAMM(II,IR)=NSAMM(II,IR) + 1
   RETURN
   END
```

```
      SUBROUTINE ENDPTS(XMIN,XMAX,IT)

      COMMON /STAT2/ EXTXL(50),EXTXLS(50),EXTXR(50),EXTXRS(50),NIX(50)
      IF(XMAX-XMIN.LE.1.0) GO TO 25
      EXTXL(IT)=EXTXL(IT) + XMIN
      EXTXLS(IT)=EXTXLS(IT) + XMIN*XMIN
      EXTXR(IT)=EXTXR(IT) + XMAX
      EXTXRS(IT)=EXTXRS(IT) + XMAX*XMAX
      RETURN
   25 CONTINUE
      NIX(IT)=NIX(IT) + 1
      RETURN
      END
```

```
      SUBROUTINE MATCON(Z,XLOW,XHIGH,YLOW,YHIGH)

      COMMON /WNDDIR/WVU1,WVU2
      DIMENSION XCORD(40),YCORD(40)
      DIMENSION CONMAT(40,40)

      DO 1100 I=1,40
      XCORD(I)=FLOAT (I)*(1.0/40.0)*(XHIGH-XLOW)+XLOW
 1100 CONTINUE
      DO 1500 I=1,40
      YCORD(I)=FLOAT(I-1)*(1.0/40.0)*(YHIGH-YLOW)+YLOW
 1500 CONTINUE

      PRINT 22
   22 FORMAT(' ',' X ',5X,20X,'            Y               ')

      PRINT 26,(YCORD(IP),IP=1,20)
   26 FORMAT('0',7X,20(F6.1))
      DO 2400 I=1,40
      DO 2400 J=1,40
      X=WVU1*XCORD(I)+WVU2*YCORD(J)
      Y=-WVU2*XCORD(I)+WVU1*YCORD(J)
      CALL DENSTY(X,Y,Z,CON)
      CONMAT(I,J)=CON
 2400 CONTINUE
      DO 2500 I=1,40
      PRINT 28,XCORD(I),(CONMAT(I,IP),IP=1,20)
      IBLANK=5*(I/5)-I
      IF (IBLANK.EQ.0)PRINT 32
 2500 CONTINUE
      PRINT 22
      PRINT 26,(YCORD(IP),IP=21,40)
      DO 3000 I=1,40
      PRINT 28,XCORD(I),(CONMAT(I,IP),IP=21,40)
      IBLANK=5*(I/5)-I
      IF (IBLANK.EQ.0)PRINT 32
 3000 CONTINUE

   28 FORMAT(' ',1X,F6.1,20(F6.0))

   32 FORMAT(' ')
      RETURN
      END
```

```
      SUBROUTINE MATCL(Z,XLOW,XHIGH,YLOW,YHIGH)

      COMMON /WNDDIR/WVU1,WVU2
      DIMENSION X1(40),X2(40)
      DIMENSION CONLIN(40,40)

      Y1=YLOW
      Y2=YHIGH
      DZ=0.0
      DO 1100 I=1,40
      X1(I)=FLOAT(I-1)*(1.0/40.0)*(XHIGH-XLOW)+XLOW
 1100 CONTINUE
      DO 1500 I=1,40
      X2(I)=FLOAT(I-1)*(1.0/40.0)*(XHIGH-XLOW)+XLOW
 1500 CONTINUE

      PRINT 15,Y1,Y2
   15 FORMAT('0',' Y1 ',F8.2,' Y2 ',F8.2)
      PRINT 22
   22 FORMAT(' ',' X1  ',5X,20X,'               X2               ')

      DO 2500 I=1,40
      DO 2400 J=1,40
      DIST=SQRT( (X2(J)-X1(I))**2+(Y2-Y1)**2)
      DX=(X2(J)-X1(I))/DIST
      DY=(Y2-Y1)/DIST
      X=WVU1*X1(I)+WVU2*YLOW
      Y=-WVU2*X1(I)+WVU1*YLOW
      DXP=WVU1*DX+WVU2*DY
      DYP=-WVU2*DX+WVU1*DY
      CALL LCON(X,Y,Z,DXP,DYP,DZ,CON)
      CONLIN(I,J)=CON
 2400 CONTINUE
 2500 CONTINUE

      PRINT 26,(X2(I),I=1,20)
      DO 29 K=1,40
      PRINT 28,X1(K),(CONLIN(K,I),I=1,20)
   29 CONTINUE

      PRINT 15,Y1,Y2
      PRINT 22
      PRINT 26,(X2(I),I=21,40)
      DO 27 K=1,40
      PRINT 28,X1(K),(CONLIN(K,I),I=21,40)
   27 CONTINUE
   26 FORMAT('0',7X,20(F6.1))
   28 FORMAT(' ',F6.1,20(F6.0))
      RETURN
      END
```

```
      SUBROUTINE CALPRT(NSAMP,NT,SNAP,NVOL,VT,NDEVIC)
C     CALCULATE AND PRINT STATISTICS
      COMMON /STATS/ SSL(50,3),SSLS(50,3),SLY(50,3),SLYS(50,3),
     1 SRY(50,3),SRYS(50,3),SSH(50,3),SSHS(50,3),NSAMM(50,3),LEN(100),
     2 OSL(50,3,100)
      COMMON /STAT2/ EXTXL(50),EXTXLS(50),EXTXR(50),EXTXRS(50),NIX(50)

      PRINT 510,NSAMP
  510 FORMAT(1H1,*TOTAL NUMBER OF SAMPLES =*,I5)

      DO 505 I=1,NT
      DO 520 IR=1,NDEVIC
      NSAMPI=NSAMP-NSAMM(I,IR)
      IF(NSAMPI.LE.1)GO TO 515
      SLYS(I,IR)=SQRT((SLYS(I,IR)-SLY(I,IR)**2./NSAMPI)/(NSAMPI-1))
      SRYS(I,IR)=SQRT((SRYS(I,IR)-SRY(I,IR)**2./NSAMPI)/(NSAMPI-1))
      SSHS(I,IR)=SQRT((SSHS(I,IR)-SSH(I,IR)**2./NSAMPI)/(NSAMPI-1))
      SLY(I,IR)=SLY(I,IR)/NSAMPI
      SRY(I,IR)=SRY(I,IR)/NSAMPI
      SSH(I,IR)=SSH(I,IR)/NSAMPI
  515 SSLS(I,IR)=SQRT((SSLS(I,IR)-SSL(I,IR)**2./NSAMP)/(NSAMP-1))
      SSL(I,IR)=SSL(I,IR)/NSAMP
  520 CONTINUE
      NIXI=NSAMP-NIX(I)
      IF(NIXI.LE.1) GO TO 505
      EXTXLS(I)=SQRT((EXTXLS(I)-EXTXL(I)**2./NIXI)/(NIXI-1))
      EXTXRS(I)=SQRT((EXTXRS(I)-EXTXR(I)**2./NIXI)/(NIXI-1))
      EXTXL(I)=EXTXL(I)/NIXI
      EXTXR(I)=EXTXR(I)/NIXI
  505 CONTINUE

      DO 530 I=1,NT
      STIME=(I-1)*SNAP
      PRINT 560,STIME
  560 FORMAT(1H0,*SNAP TIME =*,F10.1)
      J=MIN0(NVOL,1+INT(STIME/VT))
      PRINT 570,J
  570 FORMAT(1H0,*VOLLEYS FIRED =*,I5)
      PRINT 540,NIX(I),(NSAMM(I,IR),IR=1,NDEVIC)
  540 FORMAT(*0*,*NUMBER TIMES NO SCREEN *,I5,3(5X,* IN SECTOR *,I5))
      PRINT 600,((SSL(I,IR),SSLS(I,IR)),IR=1,NDEVIC)
      PRINT 601,((SSH(I,IR),SSHS(I,IR)),IR=1,NDEVIC)
      PRINT 602,((SLY(I,IR),SLYS(I,IR)),IR=1,NDEVIC)
      PRINT 603,((SRY(I,IR),SRYS(I,IR)),IR=1,NDEVIC)
      PRINT 604,EXTXL(I),EXTXLS(I)
      PRINT 605,EXTXR(I),EXTXRS(I)
  600 FORMAT(*0*,* SCREEN LENGTH *,2X,3(* AVE  *,F8.1,2X,* SIGMA *,F8.1))
  601 FORMAT(*0*,* INTERIOR HOLE *,2X,3(* AVE  *,F8.1,2X,* SIGMA *,F8.1))
  602 FORMAT(*0*,* LEFT COORD    *,2X,3(* AVE  *,F8.1,2X,* SIGMA *,F8.1))
  603 FORMAT(*0*,* RIGHT COORD   *,2X,3(* AVE  *,F8.1,2X,* SIGMA *,F8.1))
  604 FORMAT(*0*,* EXT LEFT COORD*,2X,* AVE *,F8.1,2X,* SIGMA *,F8.1)
  605 FORMAT(*0*,* EXT RGHT COORD*,2X,* AVE *,F8.1,2X,* SIGMA *,F8.1)
  530 CONTINUE
      PRINT 650
  650 FORMAT(1H0,18HDISTRIBUTION OF SL)
      DO 680 I=1,NT
      STIME=(I-1)*SNAP
      PRINT 560,STIME
      J=MIN0(NVOL,1+INT(STIME/VT))
      PRINT 570,J
      DO 678 IR=1,NDEVIC
      DO 675 L=1,4
      K2=25*L
      K1=K2-24
      PRINT 660,(LEN(K),K=K1,K2)
  660 FORMAT(1H0,25I5)
```

```
      PRINT 670,(DSL(I,IR,K),K=K1,K2)
670   FORMAT(1H0,2X,25F5.0)
      PRINT 672
672   FORMAT(1H0)
675   CONTINUE
678   CONTINUE
680   CONTINUE
      RETURN
      END
```

```
SUBROUTINE STCL(QMUN,YIELD,SIGXS,SIGYS,SIGZS,ALPHAP,BETAP,UP)

COMMON /WPCLD/ FSIGX,FSIGY,FSIGZ
COMMON /WPCLD/ DSIGX,DSIGY,DSIGZ
COMMON /WPCLD/ ASIGX,ASIGY,ASIGZ
COMMON /WPCLD/ ESIGX,ESIGY,ESIGZ
COMMON /WPCLD/ WNDSPD,FACTOR

WNDSPD=UP
ESIGX=0.9294
ESIGY=ALPHAP
ESIGZ=BETAP
FACTOR=QMUN*YIELD*.126987
ASIGX=DSIGX*((SIGXS/FSIGX)**(1.0/ESIGX))
ASIGY=DSIGY*((SIGYS/FSIGY)**(1.0/ESIGY))
ASIGZ=DSIGZ*((SIGZS/FSIGZ)**(1.0/ESIGZ))

RETURN

END
```

```
      SUBROUTINE FRMCLG(NT,DT,NVOL,NS)

      COMMON /WPSCR/ SIGX(50),SIGY(50),SIGZ(50)
      COMMON /WPSCR/ FACTT(50)

      COMMON /WPCLG/ FSIGX,FSIGY,FSIGZ
      COMMON /WPCLG/ DSIGX,DSIGY,DSIGZ
      COMMON /WPCLD/ ASIGX,ASIGY,ASIGZ
      COMMON /WPCLD/ ESIGX,ESIGY,ESIGZ
      COMMON /WPCLD/ WNDSPD,FACTOR

      PRINT 14
   14 FORMAT('0','NO.',15X,'SIZE')
      NS=50
      DO 199 I=1,NS
      T=DT*(I-1)
      IF (T.GE.0.0) GO TO 150
      SIGX(I)=-1.0
      GO TO 199
  150 CONTINUE
      SIGX(I)=FSIGX*((WNDSPD*T+ASIGX)/DSIGX)**ESIGX
      SIGY(I)=FSIGY*((WNDSPD*T+ASIGY)/DSIGY)**ESIGY
      SIGZ(I)=FSIGZ*((WNDSPD*T+ASIGZ)/DSIGZ)**ESIGZ
      FACTT(I)=FACTOR/(SIGX(I)*SIGY(I)*SIGZ(I))
      PRINT 16,I,SIGX(I),SIGY(I),SIGZ(I)
   16 FORMAT(' ',I4,6(1X,F10.2,1X))
  199 CONTINUE
      RETURN

      END
```

```
      SUBROUTINE TIME(MT,U)

      COMMON /REL/ XB(60),YB(60),ZB(60),TB(60),NBURST
      COMMON /REL2/XCENTP(60),YCENTP(60)

      COMMON /WPSCR/ SIGX(50),SIGY(50),SIGZ(50)

      COMMON /ABS/XBA(60),YBA(60),ZBA(60),TBA(60),NBRSTA
      COMMON /ABS2/ XCENT(60),YCENT(60),ZCENT(60)
      COMMON /ABS3/ SIGXA(50),SIGYA(50)

      COMMON /TMSCLE/INSNAP,INVOL,DT
      COMMON /WNDDIR/WVU1,WVU2
      COMMON /AGEIND/ IAGE(60)

      DO 1 I=1,NBURST
      AGE=(MT-1)*INSNAP*DT-TB(I)
      MUNGRP=AGE/DT+1
      IAGE(I)=MUNGRP
      IF(MUNGRP.LE.0) GO TO 10
      XCENT(I)=XBA(I)+ U *WVU1*AGE
      YCENT(I)=YBA(I)+ U *WVU2*AGE
      XCENTP(I)=XB(I)+ U *AGE
      SIGXA(I)=AMAX1(ABS(SIGX(MUNGRP)*WVU1),ABS(SIGY(MUNGRP)*WVU2))
      SIGYA(I)=AMAX1(ABS(SIGX(MUNGRP)*WVU2),ABS(SIGY(MUNGRP)*WVU1))
    1 CONTINUE
      RETURN
   10 CONTINUE
      END
```

```
      SUBROUTINE DENSTY(XP,YP,ZP,CON)

      COMMON /REL/ XB(60),YB(60),ZB(60),TB(60),NBURST
      COMMON /REL2/XCENTP(60),YCENTP(60)

      COMMON /WPSCR/ SIGX(50),SIGY(50),SIGZ(50)
      COMMON /WPSCR/ FACTT(50)

      COMMON /AGEIND/ IAGE(60)

      CON=0.0

      DO 399 MUN=1,NBURST

      MUNGRP=IAGE(MUN)
      IF(MUNGRP.LE.0) GO TO 400

      X=XP-XCENTP(MUN)
      Y=YP-YCENTP(MUN)
      Z=ZP-ZB(MUN)

      CONN=(-.5*((X/SIGX(MUNGRP))**2
     .          +(Y/SIGY(MUNGRP))**2
     .          +(Z/SIGZ(MUNGRP))**2))

      IF(CONN.LE.-25.) GO TO 399
      CON=CON+FACTT(MUNGRP)*EXP(CONN)

  399 CONTINUE

  400 CONTINUE
      RETURN

      END
```

```
      SUBROUTINE LCON(POX,POY,POZ,DX,DY,DZ,TOTLNC)

      COMMON /REL/ XB(60),YB(60),ZB(60),TB(60),NBURST
      COMMON /REL2/XCENTP(60),YCENTP(60)

      COMMON /WPSCR/ SIGX(50),SIGY(50),SIGZ(50)
      COMMON /WPSCR/ FACTT(50)

      COMMON /AGEIND/ IAGE(60)

      TOTLNC=0.0

      DO 399 N=1,NBURST
      I=IAGE(N)
      IF(I.LE.0) GO TO 400
      IF(SIGX(I).LE.0.0)GO TO 399

      A=(DX/SIGX(I))**2+(DY/SIGY(I))**2+(DZ/SIGZ(I))**2
      B=DX*(POX-XCENTP(N))/SIGX(I)**2+
     *  DY*(POY-YCENTP(N))/SIGY(I)**2+
     *  DZ*(POZ-ZB(N))/SIGZ(I)**2
      S=-B/A
      PMX=POX+S*DX
      PMY=POY+S*DY
      PMZ=POZ+S*DZ
      C=((PMX-XCENTP(N))/SIGX(I))**2+
     *  ((PMY-YCENTP(N))/SIGY(I))**2+
     *  ((PMZ-ZB(N))/SIGZ(I))**2
      CC=(-.5*C)
      IF(CC.LE.-25.) GO TO 399
      FACLIN=FACTT(I)*2.5066283*EXP(CC)/SQRT(A)
      TOTLNC=FACLIN+TOTLNC
  399 CONTINUE
  400 CONTINUE
      RETURN

      END
```

APPENDIX F

SOURCE LISTING OF PROGRAM HC MUNITIONS

Next page is blank.

```
      PROGRAM SMOKEHC(INPUT,OUTPUT,TAPE4=INPUT)

      REAL NLOCX,NLOCY,NLOCZ
      REAL NSIZX,NSIZY,NSIZZ
      REAL NLOCXA,NLOCYA,NLOCZA
      REAL NSIZXA,NSIZYA,NSIZZA

      COMMON XB(100),YB(100),ZB(100),TB(100),NBURST
      COMMON FLOCX(100),FLOCY(100),FLOCZ(100)
      COMMON FSIZX(100),FSIZY(100),FSIZZ(100)
      COMMON NLOCX(100),NLOCY(100),NLOCZ(100)
      COMMON NSIZX(100),NSIZY(100),NSIZZ(100)

      COMMON /ABS/XBA(100),YBA(100),ZBA(100),TBA(100),NBRSTA
      COMMON /ABS/FLOCXA(100),FLOCYA(100),FLOCZA(100)
      COMMON /ABS/FSIZXA(100),FSIZYA(100),FSIZZA(100)
      COMMON /ABS/NLOCXA(100),NLOCYA(100),NLOCZA(100)
      COMMON /ABS/NSIZXA(100),NSIZYA(100),NSIZZA(100)

      COMMON /PLACE/ SIGBR,SIGBD,SIGAR,SIGAD,REL,RDISP
      COMMON /PLACE/ XIDEAL(6),YIDEAL(6),ZIDEAL(6)
      COMMON /WNDDIR/WVU1,WVU2
      COMMON /AGEIND/ IAGE(100)
      COMMON /FLDSIT/ CUTOFL,CUTOFR,XINC

      COMMON /HCSCR/ XCENT(20,150)
      COMMON /HCSCR/ SIGX(20,150),SIGY(20,150),SIGZ(20,150)
      COMMON /HCSCR/ FACTT(20,150),NPUFF(20)

      COMMON /HCCLD/ FSIGX,FSIGY,FSIGZ
      COMMON /HCCLD/ DSIGX,DSIGY,DSIGZ
      COMMON /HCCLD/ ASIGX,ASIGY,ASIGZ
      COMMON /HCCLD/ ESIGX,ESIGY,ESIGZ
      COMMON /HCCLD/ WNDSPD,FACTOR,TINCR

      DIMENSION TITLE(8),CATTN(3),THRES(3),CLTHRS(3)

      DATA FSIGX/3.41/,FSIGY/3.41/,FSIGZ/1.35/
      DATA DSIGX/100.0/,DSIGY/100.0/,DSIGZ/20.0/
      DATA ASIGX/0.0/,ASIGY/0.0/,ASIGZ/0.0/

      READ(4,2) TITLE
    2 FORMAT(8A10)
      PRINT 1,TITLE
    1 FORMAT(1H1,8A10)

      READ(4,3) U,ALPHA,BETA
    3 FORMAT(3F10.5)
      PRINT 4,U,ALPHA,BETA
    4 FORMAT('0',' WIND SPEED ',F10.5,
     *       ' ALPHA ',F10.5,' BETA ',F10.5)

      READ(4,11) WVU1,WVU2
   11 FORMAT(2F10.5)
      PRINT 12,WVU1,WVU2
   12 FORMAT(1H0,'WIND VECTOR DIRECTION COSINES X =',F10.5,'Y =',F10.5)

      READ(4,6) YIELD,QMUN,BURNT
    6 FORMAT(F10.5,E10.5,F10.5)
      PRINT 10,YIELD,QMUN,BURNT
   10 FORMAT('0',' YIELD =',F10.5,
     *         ' QUAN OF MUN ADJ =',E10.5,' BURN TIME =',F10.5)

      CALL STCL(QMUN,YIELD,BURNT,ALPHA,BETA,U)

      READ(4,8) NDEVIC
    8 FORMAT(I5)
      READ(4,15) ((CATTN(IR),THRES(IR)),IR=1,NDEVIC)
```

```
   15 FORMAT(6F10.5)
      DO 159 I=1,NDEVIC
      CLTHRS(I)=ALOG(1.0-THRES(I))/(-CATIN(I))
      PRINT 20,I,CATTN(I),THRES(I),CLTHRS(I)
  159 CONTINUE
   20 FORMAT('0',I3,' COEF ATTN =',F10.5,
     1           ' THRESHOLD =',F10.2,
     2           ' CL THRES  =',F10.2)

C READ IN PLACEMENT INFORMATION

      READ(4,50) NRPV,NMPR
   50 FORMAT(2I10)
      PRINT 14,NRPV,NMPR
   14 FORMAT(1H0,'NUMBER OF ROUNDS PER VOLLEY ',I4/
     1        1H0,'NUMBER OF MUNITIONS PER ROUND', I4/
     2        1H0,'IDEAL IMPACT POINTS '/
     3        1H0,' NO. ',6X,'X',11X,'Y',11X,'Z')

      DO 2500 J=1,NRPV

      READ(4,55) XIDEAL(J),YIDEAL(J),ZIDEAL(J)
   55 FORMAT(4F10.5)
      PRINT 16,J,XIDEAL(J),YIDEAL(J),ZIDEAL(J)

 2500 CONTINUE
   16 FORMAT(' ',I4,6(1X,F10.2,1X))

      READ(4,21) NSAMP,NVOL,NT,DT,DL
   21 FORMAT(3I5,2F5.0)
      PRINT 23,NSAMP,NVOL,NT,DT,DL
   23 FORMAT(1H0,'TOTAL NUMBER OF SAMPLES =', I5,/
     1        1X,'TOTAL NUMBER OF VOLLEYS =', I5,/
     2        1X,'TOTAL NUMBER OF TIMES   =', I5,/
     3        1X,'TIME INCREMENTS =',F5.0,'SEC.',/
     4        1X,'LINE INCREMENTS =',F5.0)

      READ(4,19) VT,SNAP
   19 FORMAT(2F10.1)
      PRINT 25,VT,SNAP
   25 FORMAT(1H0,'TIME BETWEEN VOLLEYS =',F10.1,/
     1        1X,'SNAP TIME INCREMENT =',F10.1)

      READ(4,62) SIGBR,SIGBD,SIGAR,SIGAD,REL,RDISP
   62 FORMAT(6F10.4)
      PRINT 24,SIGBR,SIGBD,SIGAR,SIGAD,REL,RDISP
   24 FORMAT(1H0,'BR =',F5.0,5X,'BD =',F5.0,/
     1        1X,'AR =',F5.0,5X,'AD =',F5.0,/
     2        1X,'REL =',F5.0,/
     3        1X,'RADIAL DISP =',F5.0)

      NBRSTA=NVOL*NRPV*NMPR
      XOBS=0.0
      YOBS=0.0
      READ(4,15) YTLINE,CUTOFL,CUTOFR
      READ(4,15) XT,YT
      READ(4,15) XINC
      Z=1.5

      PRINT 74,XINC
   74 FORMAT(1H0,'X-INCREMENT =',F5.1)
      PRINT 75,CUTOFL,CUTOFR
   75 FORMAT(1H0,'90 DEGREE SECTOR =',F7.0,' , ',F7.0)
      PRINT 73,XOBS,YOBS
   73 FORMAT(1H0,'OBSERVER COORD =',2F10.1)
      PRINT 71,XT,YT
   71 FORMAT(1H0,'AIMPOINT COORD =',2F10.1)
      PRINT 72,YTLINE
```

```
  72 FORMAT(1H0,'DISTANCE TO LINE =',F10 ))
     PRINT 35,Z
  35 FORMAT('0','HEIGHT ABOVE CENTROID ',F10 5)

     CALL CLEAR (DL)

     DO 1000 KI=1,NSAMP

     PRINT 7,KI
   7 FORMAT(1H1,'DATA FOR SAMPLE NUMBER ',I5)

     CALL MPLACE(NVOL,NRPV,NMPR,VT,XT,YT)
     CALL CONBST
     PRINT 42
  42 FORMAT('0',' MUNITION BURSTS AT ')
     PRINT 43
  43 FORMAT('0',11X,'LOCATION',11X,5X,7X,'TIME')

     I=0
     DO 199 IV=1,NVOL
     DO 199 IRD=1,NRPV
     DO 199 IS=1,NMPR
     I=I+1

     PRINT 48,I,IV,IRD,IS,XBA(I),YBA(I),ZBA(I),TBA(I)
 199 CONTINUE

  48 FORMAT(' ',4(1X,I3,1X),3(1X,F8.2,1X),6X,F8.2)

     DO 399 IT=1,NT
     T=FLOAT(IT)*SNAP

     PRINT 22,T
  22 FORMAT('0','TIME AFTER EMISSION ',F10.0)

     CALL TIME(T)
     CALL CON

C    PRINT 32
  32 FORMAT('0',' MUNITION CHARACTERISTICS   ')
C    PRINT 33
  33 FORMAT('0',11X,'LOCATION',11X,5X,14X,'SIZE')

     DO 350 IMUN=1,NBRSTA
C    PRINT 38,IMUN,NLOCXA(IMUN),NLOCYA(IMUN),NLOCZA(IMUN),
C   *         NSIZXA(IMUN),NSIZYA(IMUN),NSIZZA(IMUN)
C    PRINT 39,IMUN,FLOCXA(IMUN),FLOCYA(IMUN),FLOCZA(IMUN),
C   *         FSIZXA(IMUN),FSIZYA(IMUN),FSIZZA(IMUN)

 350 CONTINUE

  38 FORMAT('0',1X,I2,1X,'NEAR',1X,3(1X,F8.2,1X),5X,3(1X,F8.2,1X))
  39 FORMAT('0',1X,I2,1X,' FAR',1X,3(1X,F8.2,1X),5X,3(1X,F8.2,1X))

     CALL SIZE(XLOW,XHIGH,YLOW,YHIGH)
     CALL MXMIN(YTLINE,XMIN,XMAX)
     PRINT 5,XMIN,XMAX
   5 FORMAT(1H ,'XMIN =',F10.2,2X,'XMAX =',F10.2)

     YSLINE=500.0

C    CALL MATCON(Z,XLOW,XHIGH,YLOW,YHIGH)
C    CALL MATCL(Z,XLOW,XHIGH,YOBS,YSLINE)
     CALL CALC(XOBS,YOBS,Z,YTLINE,XMIN,XMAX,NDEVIC,CLTHRS)
     CALL ENDPTS(XMIN,XMAX,IT)

     DO 299 IR=1,NDEVIC
     CALL VEVAL (IT,IR)
```

```
  299 CONTINUE
  399 CONTINUE
 1000 CONTINUE
      CALL CALPRT (NSAMP,NT,SNAP,NVOL,VT,NDEVIC)
      STOP
      END
```

```
      SUBROUTINE CLEAR(DL)

      COMMON /STATS/ SSL(50,3),SSLS(50,3),SLY(50,3),SLYS(50,3),
     1 SRY(50,3),SRYS(50,3),SSH(50,3),SSHS(50,3),NSAMM(50,3),LEN(100),
     2 DSL(50,3,100)
      COMMON /STAT2/ EXTXL(50),EXTXLS(50),EXTXR(50),EXTXRS(50),NIX(50)

      DO 800 I=1,100
  800 LEN(I)=DL*I
      DO 120 I=1,50
      NIX(I)=0
      EXTXL(I)=0.0
      EXTXLS(I)=0.0
      EXTXR(I)=0.0
      EXTXRS(I)=0.0
      DO 110 IR=1,3
      NSAMM(I,IR)=0
      SSL(I,IR)=0.0
      SSLS(I,IR)=0.0
      SLY(I,IR)=0.0
      SLYS(I,IR)=0.0
      SRY(I,IR)=0.0
      SRYS(I,IR)=0.0
      DO 105 K=1,100
  105 DSL(I,IR,K)=0.0
      SSH(I,IR)=0.0
  110 SSHS(I,IR)=0.0
  120 CONTINUE
      RETURN

      END
```

```
      SUBROUTINE MPLACE(NVOL,NRPV,NMPR,VT,XT,YT)

      COMMON /ABS/  XB(100),YB(100),ZB(100),TB(100),NBURST
      COMMON /PLACE/ SIGBR,SIGBD,SIGAR,SIGAD,REL,RDISP
      COMMON /PLACE/ XIDEAL(6),YIDEAL(6),ZIDEAL(6)
      CALL NORAN(R,SIGAR,E,SIGAD)
      XCA=XT+E
      YCA=YT+R
      DO 32 I=1,NVOL
      BRSTTM= (I-1)*VT
      DO 30 J=1,NRPV
      IF(RANF(DUM).GT.REL) GO TO 30
      CALL NORAN(R,SIGBR,E,SIGBD)
      XCRD=XCA+XIDEAL(J)+E
      YCRD=YCA+YIDEAL(J)+R
      DO 28 K=1,NMPR
      MUNNO=(I-1)*NRPV*NMPR +(J-1)*NMPR +K
      CALL MUNDSP(RDISP,XMUN,YMUN)
      XB(MUNNO)=XCRD+XMUN
      YB(MUNNO)=YCRD+YMUN
      ZB(MUNNO)=0.0
      TB(MUNNO)=BRSTTM
   28 CONTINUE
   30 CONTINUE
   32 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONBST

      COMMON XB(100),YB(100),ZB(100),TB(100),NBURST

      COMMON /ABS/XBA(100),YBA(100),ZBA(100),TBA(100),NBRSTA
      COMMON /WNDDIR/WVU1,WVU2

      NBURST=NBRSTA
      DO 100 I=1,NBRSTA
      XB(I)=WVU1*XBA(I)+WVU2*YBA(I)
      YB(I)=-WVU2*XBA(I)+WVU1*YBA(I)
      ZB(I)=ZBA(I)
      TB(I)=TBA(I)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CON

      REAL NLOCX,NLOCY,NLOCZ
      REAL NSIZX,NSIZY,NSIZZ
      REAL NLOCXA,NLOCYA,NLOCZA
      REAL NSIZXA,NSIZYA,NSIZZA

      COMMON XB(100),YB(100),ZB(100),TB(100),NBURST
      COMMON FLOCX(100),FLOCY(100),FLOCZ(100)
      COMMON FSIZX(100),FSIZY(100),FSIZZ(100)
      COMMON NLOCX(100),NLOCY(100),NLOCZ(100)
      COMMON NSIZX(100),NSIZY(100),NSIZZ(100)

      COMMON /ABS/XBA(100),YBA(100),ZBA(100),TBA(100),NBRSTA
      COMMON /ABS/FLOCXA(100),FLOCYA(100),FLOCZA(100)
      COMMON /ABS/FSIZXA(100);FSIZYA(100),FSIZZA(100)
      COMMON /ABS/NLOCXA(100),NLOCYA(100),NLOCZA(100)
      COMMON /ABS/NSIZXA(100),NSIZYA(100),NSIZZA(100)

      COMMON /WNDDIR/WVU1,WVU2

      DO 100 I=1,NBURST
      NLOCXA(I)=WVU1*NLOCX(I)-WVU2*NLOCY(I)
      NLOCYA(I)=WVU2*NLOCX(I)+WVU1*NLOCY(I)
      NLOCZA(I)=NLOCZ(I)
      NSIZXA(I)=NSIZX(I)
      NSIZYA(I)=NSIZY(I)
      NSIZZA(I)=NSIZZ(I)
      FLOCXA(I)=WVU1*FLOCX(I)-WVU2*FLOCY(I)
      FLOCYA(I)=WVU2*FLOCX(I)+WVU1*FLOCY(I)
      FLOCZA(I)=FLOCZ(I)
      FSIZXA(I)=FSIZX(I)
      FSIZYA(I)=FSIZY(I)
      FSIZZA(I)=FSIZZ(I)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MUNDSP(R,XMUN,YMUN)

      RS=SQRT(RANF(DUM))*R
      THETA=RANF(DUM)*6.28318
      XMUN=RS*COS(THETA)
      YMUN=RS*SIN(THETA)
      RETURN
      END
```

```
      SUBROUTINE NORAN(R,SR,D,SD)
    1 X=RANF(DUM)
      IF(X.LE.0.0) GO TO 1
      A=SQRT(-2.0*ALOG(X))
      B=6.2831853071 8*RANF(DUM)
      R=A*SR*SIN(B)
      D=A*SD*COS(B)
      RETURN
      END
```

```
      SUBROUTINE SIZE(XLOW,XHIGH,YLOW,YHIGH)

      REAL NLOCX,NLOCY,NLOCZ
      REAL NSIZX,NSIZY,NSIZZ

      COMMON /ABS/XB(100),YB(100),ZB(100),TB(100),NBURST
      COMMON /ABS/FLOCX(100),FLOCY(100),FLOCZ(100)
      COMMON /ABS/FSIZX(100),FSIZY(100),FSIZZ(100)
      COMMON /ABS/NLOCX(100),NLOCY(100),NLOCZ(100)
      COMMON /ABS/NSIZX(100),NSIZY(100),NSIZZ(100)

      XLOW=1.0E+6
      XHIGH=-1.0E+6
      YLOW=1.0E+6
      YHIGH=-1.0E+6
      DO 10 I=1,NBURST
      TEST=(NLOCX(I)-4.*NSIZX(I))
      IF(TEST.LT.XLOW) XLOW=TEST
      IF(TEST.GT.XHIGH) XHIGH=TEST
      TEST=(NLOCX(I)+4.*NSIZX(I))
      IF(TEST.LT.XLOW) XLOW=TEST
      IF(TEST.GT.XHIGH) XHIGH=TEST
      TEST=(FLOCX(I)-4.*FSIZX(I))
      IF(TEST.LT.XLOW) XLOW=TEST
      IF(TEST.GT.XHIGH) XHIGH=TEST
      TEST=(FLOCX(I)+4.*FSIZX(I))
      IF(TEST.LT.XLOW) XLOW=TEST
      IF(TEST.GT.XHIGH) XHIGH=TEST
      TEST=(NLOCY(I)-4.*NSIZY(I))
      IF(TEST.LT.YLOW) YLOW=TEST
      IF(TEST.GT.YHIGH) YHIGH=TEST
      TEST=(NLOCY(I)+4.*NSIZY(I))
      IF(TEST.LT.YLOW) YLOW=TEST
      IF(TEST.GT.YHIGH) YHIGH=TEST
      TEST=(FLOCY(I)-4.*FSIZY(I))
      IF(TEST.LT.YLOW) YLOW=TEST
      IF(TEST.GT.YHIGH) YHIGH=TEST
      TEST=(FLOCY(I)+4.*FSIZY(I))
      IF(TEST.LT.YLOW) YLOW=TEST
      IF(TEST.GT.YHIGH) YHIGH=TEST
   10 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE MXMIN(YTLINE,XMIN,XMAX)

      REAL NLOCX,NLOCY,NLOCZ
      REAL NSIZX,NSIZY,NSIZZ

      COMMON /ABS/XB(100),YB(100),ZB(100),TB(100),NBURST
      COMMON /ABS/FLOCX(100),FLOCY(100),FLOCZ(100)
      COMMON /ABS/FSIZX(100),FSIZY(100),FSIZZ(100)
      COMMON /ABS/NLOCX(100),NLOCY(100),NLOCZ(100)
      COMMON /ABS/NSIZX(100),NSIZY(100),NSIZZ(100)

      RATMX=-1.0E+10
      RATMN=1.0E+10
      DO 10 I=1,NBURST
      IF(NLOCY(I).EQ.0.0) GO TO 5
      RATL=(NLOCX(I)-4.*NSIZX(I))/NLOCY(I)
      IF(RATL.LT.RATMN) RATMN=RATL
      IF(RATL.GT.RATMX) RATMX=RATL
      RATN=(NLOCX(I)+4.*NSIZX(I))/NLOCY(I)
      IF(RATN.LT.RATMN) RATMN=RATN
      IF(RATN.GT.RATMX) RATMX=RATN
    5 CONTINUE
      IF(FLOCY(I).EQ.0.0) GO TO 10
      RATL=(FLOCX(I)-4.*FSIZX(I))/FLOCY(I)
      IF(RATL.LT.RATMN) RATMN=RATL
      IF(RATL.GT.RATMX) RATMX=RATL
      RATN=(FLOCX(I)+4.*FSIZX(I))/FLOCY(I)
      IF(RATN.LT.RATMN) RATMN=RATN
      IF(RATN.GT.RATMX) RATMX=RATN
   10 CONTINUE
      XMIN=YTLINE*RATMN
      XMAX=YTLINE*RATMX
      RETURN
      END
```

```
      SUBROUTINE CALC(XOBS,YOBS,Z,YTLINE,XMIN,XMAX,NDEVIC,CLTHRS)

      COMMON /FLDSIT/ CUTOFL,CUTOFR,XINC
      COMMON /LINES/ V(1001,3),IXLEFT,IXRGHT
      COMMON /WNDDIR/WVU1,WVU2
      DIMENSION CLTHRS(3)

      XL=AMAX1(CUTOFL,XMIN)
      XR=AMIN1(CUTOFR,XMAX)
      IXLEFT=INT((XL-CUTOFL)/XINC) + 1
      IXRGHT=INT((XR-CUTDFL)/XINC) + 1
      DO 31 IP=IXLEFT,IXRGHT
      X=XINC * FLDAT(IP-1) +CUTOFL
      DX=X-XOBS
      DY=YTLINE-YOBS
      DZ=0.0
      VMAG=SQRT(DX**2+DY**2+DZ**2)
      DX=DX/VMAG
      DY=DY/VMAG
      DZ=DZ/VMAO
      DXP=WVU1*DX+WVU2*DY
      DYP=-WVU2*DX+WVU1*DY
      CALL LCON(XOBS,YOBS,Z,DXP,DYP,DZ,TDTLNC)
      DO 29 IR=1,NDEVIC
      V(IP,IR)=0.0
      IF(TDTLNC.OE.CLTHRS(IR)) V(IP,IR)=1.0
   29 CONTINUE
   31 CONTINUE
      DO 35 IR=1,NDEVIC
C     PRINT 34,(V(I,IR),I=IXLEFT,IXRGHT)
   35 CONTINUE
   34 FORMAT(1H0,(25F4.1,/))
      RETURN
      END
```

```
      SUBROUTINE VEVAL(II,IR)

      COMMON /FLDSIT/ CUTOFL,CUTOFR,XINC
      COMMON /LINES/ V(1001,3),IXLEFT,IXRGHT
      COMMON /STATS/ SSL(50,3),SSLS(50,3),SLY(50,3),SLYS(50,3),
     1 SRY(50,3),SRYS(50,3),SSH(50,3),SSHS(50,3),NSAMM(50,3),LEN(100),
     2 OSL(50,3,100)

      INTEGER RY,SH
      LY=0
      RY=0
      SH=0
      SL=0.0

C     LEFT Y-COORD = LY

      IF(IXRGHT-IXLEFT.LE.0) GO TO 95
      DO 10 I=IXLEFT,IXRGHT
      IF(V(I,IR).EQ.0.0) GO TO 10
      LY=XINC*FLOAT(I-1) + CUTOFL
      K=I
      GO TO 15
   10 CONTINUE
      GO TO 40

C     RIGHT Y-COORD = RY

   15 DO 20 J=K,IXRGHT
      IF(V(J,IR).EQ.0.0) GO TO 20
      RY=XINC*FLOAT(J-1) + CUTOFL
      M=J
   20 CONTINUE

C     TOTAL SCREEN LENGTH = SL

      SL=RY-LY
      IF(SL.EQ.0.) GO TO 40
      SSL(II,IR)=SSL(II,IR)+SL
      SSLS(II,IR)=SSLS(II,IR)+SL*SL
      SLY(II,IR)=SLY(II,IR)+LY
      SLYS(II,IR)=SLYS(II,IR)+LY*LY
      SRY(II,IR)=SRY(II,IR)+RY
      SRYS(II,IR)=SRYS(II,IR)+RY*RY

C     INTERIOR HOLE LENGTH = SH
C

      DO 30 I=K,M
      IF(V(I,IR).EQ.1.0) GO TO 30
      SH=SH+XINC
   30 CONTINUE
      SSH(II,IR)=SSH(II,IR)+SH
      SSHS(II,IR)=SSHS(II,IR)+SH*SH
      GO TO 50

C     NSAMM TO BE USED TO MODIFY NSAMP SUCH THAT
C     AVE LY,RY,AND SH WILL BE CALC. GIVEN
C     NON-ZERO SCREEN LENGTH.

   40 NSAMM(II,IR)=NSAMM(II,IR)+1

C     DISTRIBUTION OF SCREEN LENGTH

   50 DO 60 I=1,100
      IF(SL.GE.LEN(I)) GO TO 60
      OSL(II,IR,I)=OSL(II,IR,I)+1.0
      GO TO 70
   60 CONTINUE
```

```
70 CONTINUE
   RETURN
95 CONTINUE
   NSAMM(II,IR)=NSAMM(II,IR)+ 1
   RETURN
   END
```

```
      SUBROUTINE ENDPTS(XMIN,XMAX,IT)
      COMMON /STAT2/ EXTXL(50),EXTXLS(50),EXTXR(50),EXTXRS(50),NIX(50)
      IF(XMAX-XMIN.LE.1.0) GO TO 25
      EXTXL(IT)=EXTXL(IT) + XMIN
      EXTXLS(IT)=EXTXLS(IT) + XMIN*XMIN
      EXTXR(IT)=EXTXR(IT) + XMAX
      EXTXRS(IT)=EXTXRS(IT) + XMAX*XMAX
      RETURN
   25 CONTINUE
      NIX(IT)=NIX(IT) + 1
      RETURN
      END
```

```
      SUBROUTINE MATCON(Z,XLOW,XHIGH,YLOW,YHIGH)

      COMMON /WNDDIR/WVU1,WVU2
      DIMENSION XCORD(40),YCORD(40)
      DIMENSION CONMAT(40,40)

      DO 1100 I=1,40
      XCORD(I)=FLOAT (I)*(1.0/40.0)*(XHIGH-XLOW)+XLOW
 1100 CONTINUE
      DO 1500 I=1,40
      YCORD(I)=FLOAT(I-1)*(1.0/40.0)*(YHIGH-YLOW)+YLOW
 1500 CONTINUE

      PRINT 22
   22 FORMAT(' ',' X ',5X,20X,'         Y            ')

      PRINT 26,(YCORD(IP),IP=1,20)
   26 FORMAT('0',7X,20(F6.1))
      DO 2400 I=1,40
      DO 2400 J=1,40
      X=WVU1*XCORD(I)+WVU2*YCORD(J)
      Y=-WVU2*XCORD(I)+WVU1*YCORD(J)
      CALL DENSTY(X,Y,Z,CON)
      CONMAT(I,J)=CON
 2400 CONTINUE
      DO 2500 I=1,40
      PRINT 28,XCORD(I),(CONMAT(I,IP),IP=1,20)
      IBLANK=5*(I/5)-I
      IF (IBLANK.EQ.0)PRINT 32
 2500 CONTINUE
      PRINT 22
      PRINT 26,(YCORD(IP),IP=21,40)
      DO 3000 I=1,40
      PRINT 28,XCORD(I),(CONMAT(I,IP),IP=21,40)
      IBLANK=5*(I/5)-I
      IF (IBLANK.EQ.0)PRINT 32
 3000 CONTINUE

   28 FORMAT(' ',1X,F6.1,20(F6.0))

   32 FORMAT(' ')
      RETURN
      END
```

```
      SUBROUTINE MATCL(Z,XLOW,XHIGH,YLOW,YHIGH)

      COMMON /WNDDIR/WVUI,WVU2
      DIMENSION XI(40),X2(40)
      DIMENSION CONLIN(40,40)

      YI=YLOW
      Y2=YHIGH
      DZ=0.0
      DO 1100 I=1,40
      XI(I)=FLOAT(I-1)*(1.0/40.0)*(XHIGH-XLOW)+XLOW
 1100 CONTINUE
      DO 1500 I=1,40
      X2(I)=FLOAT(I-1)*(1.0/40.0)*(XHIGH-XLOW)+XLOW
 1500 CONTINUE

      PRINT 15,YI,Y2
   15 FORMAT('0',' YI ',F8.2,' Y2 ',F8.2)
      PRINT 22
   22 FORMAT(' ',' XI   ',5X,20X,'               X2                  ')

      DO 2500 I=1,40
      DO 2400 J=1,40
      DIST=SQRT( (X2(J)-XI(I))**2+(Y2-YI)**2)
      DX=(X2(J)-XI(I))/DIST
      DY=(Y2-YI)/DIST
      X=WVUI*XI(I)+WVU2*YLOW
      Y=-WVU2*XI(I)+WVUI*YLOW
      DXP=WVUI*DX+WVU2*DY
      DYP=-WVU2*DX+WVUI*DY
      CALL LCON(X,Y,Z,DXP,DYP,DZ,CON)
      CONLIN(I,J)=CON
 2400 CONTINUE
 2500 CONTINUE

      PRINT 26,(X2(I),I=1,20)
      DO 29 K=1,40
      PRINT 28,XI(K),(CONLIN(K,I),I=1,20)
   29 CONTINUE

      PRINT 15,YI,Y2
      PRINT 22
      PRINT 26,(X2(I),I=21,40)
      DO 27 K=1,40
      PRINT 28,XI(K),(CONLIN(K,I),I=21,40)
   27 CONTINUE
   26 FORMAT('0',7X,20(F6.1))
   28 FORMAT(' ',F6.1,20(F6.0))
      RETURN
      END
```

```
      SUBROUTINE CALPRT(NSAMP,NT,SNAP,NVOL,VT,NDEVIC)
C     CALCULATE AND PRINT STATISTICS
      COMMON /STATS/ SSL(50,3),SSLS(50,3),SLY(50,3),SLYS(50,3),
     1 SRY(50,3),SRYS(50,3),SSH(50,3),SSHS(50,3),NSAMM(50,3),LEN(100),
     2 DSL(50,3,100)
      COMMON /STAT2/ EXTXL(50),EXTXLS(50),EXTXR(50),EXTXRS(50),NIX(50)
      PRINT 510,NSAMP
  510 FORMAT(1H1,'TOTAL NUMBER OF SAMPLES =',I5)
      DO 505 I=1,NT
      DO 520 IR=1,NDEVIC
      NSAMPI=NSAMP-NSAMM(I,IR)
      IF(NSAMPI.LE.1)GO TO 515
      SLYS(I,IR)=SQRT((SLYS(I,IR)-SLY(I,IR)**2./NSAMPI)/(NSAMPI-1))
      SRYS(I,IR)=SQRT((SRYS(I,IR)-SRY(I,IR)**2./NSAMPI)/(NSAMPI-1))
      SSHS(I,IR)=SQRT((SSHS(I,IR)-SSH(I,IR)**2./NSAMPI)/(NSAMPI-1))
      SLY(I,IR)=SLY(I,IR)/NSAMPI
      SRY(I,IR)=SRY(I,IR)/NSAMPI
      SSH(I,IR)=SSH(I,IR)/NSAMPI
  515 SSLS(I,IR)=SQRT((SSLS(I,IR)-SSL(I,IR)**2./NSAMP)/(NSAMP-1))
  520 SSL(I,IR)=SSL(I,IR)/NSAMP
      NIXI=NSAMP-NIX(I)
      IF(NIXI.LE.1) GO TO 505
      EXTXLS(I)=SQRT((EXTXLS(I)-EXTXL(I)**2./NIXI)/(NIXI-1))
      EXTXRS(I)=SQRT((EXTXRS(I)-EXTXR(I)**2./NIXI)/(NIXI-1))
      EXTXL(I)=EXTXL(I)/NIXI
      EXTXR(I)=EXTXR(I)/NIXI
  505 CONTINUE
      DO 530 I=1,NT
      STIME=(I)*SNAP
      PRINT 560,STIME
  560 FORMAT(1H0,'SNAP TIME =',F10.1)
      J=MIN0(NVOL,1+INT(STIME/VT))
      PRINT 570,J
  570 FORMAT(1H0,'VOLLEYS FIRED =',I5)
      PRINT 540,NIX(I),(NSAMM(I,IR),IR=1,NDEVIC)
  540 FORMAT(1H0,'NUMBER TIMES NO SCREEN ',I5,3(2X,'IN SECTOR ',I5))
      PRINT 600,((SSL(I,IR),SSLS(I,IR)),IR=1,NDEVIC)
      PRINT 601,((SSH(I,IR),SSHS(I,IR)),IR=1,NDEVIC)
      PRINT 602,((SLY(I,IR),SLYS(I,IR)),IR=1,NDEVIC)
      PRINT 603,((SRY(I,IR),SRYS(I,IR)),IR=1,NDEVIC)
      PRINT 604,EXTXL(I),EXTXLS(I)
      PRINT 605,EXTXR(I),EXTXRS(I)
  600 FORMAT('0',' SCREEN LENGTH ',2X,3(' AVE ',F8.1,2X,' SIGMA ',F8.1))
  601 FORMAT('0',' INTERIOR HOLE ',2X,3(' AVE ',F8.1,2X,' SIGMA ',F8.1))
  602 FORMAT('0',' LEFT COORD    ',2X,3(' AVE ',F8.1,2X,' SIGMA ',F8.1))
  603 FORMAT('0',' RIGHT COORD   ',2X,3(' AVE ',F8.1,2X,' SIGMA ',F8.1))
  604 FORMAT('0',' EXT LEFT COORD',2X,' AVE ',F8.1,2X,' SIGMA ',F8.1)
  605 FORMAT('0',' EXT RGHT COORD',2X,' AVE ',F8.1,2X,' SIGMA ',F8.1)
  530 CONTINUE
      PRINT 650
  650 FORMAT(1H0,18HDISTRIBUTION OF SL)
      DO 680 I=1,NT
      STIME=(I)*SNAP
      PRINT 560,STIME
      J=MIN0(NVOL,1+INT(STIME/VT))
      PRINT 570,J
      DO 678 IR=1,NDEVIC
      DO 675 L=1,4
      K2=25*L
      K1=K2-24
      PRINT 660,(LEN(K),K=K1,K2)
  660 FORMAT(1H0,25I5)
      PRINT 670,(DSL(I,IR,K),K=K1,K2)
  670 FORMAT(1H0,2X,25F5.0)
```

```
      PRINT 672
672   FORMAT(1H0)
675   CONTINUE
678   CONTINUE
680   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE STCL(QMUN,YIELD,BTIME,ALPHAP,BETAP,UP)

      COMMON /HCCLD/ FSIGX,FSIGY,FSIGZ
      COMMON /HCCLD/ DSIGX,DSIGY,DSIGZ
      COMMON /HCCLD/ ASIGX,ASIGY,ASIGZ
      COMMON /HCCLD/ ESIGX,ESIGY,ESIGZ
      COMMON /HCCLD/ WNDSPD,FACTOR,TINCR

      NPUFTL=120
      PUFT=FLOAT(NPUFTL)
      WNDSPD=UP
      ESIGX=ALPHAP
      ESIGY=ALPHAP
      ESIGZ=BETAP
      FACTOR=QMUN*YIELD*.126987/PUFT
      TINCR=BTIME/PUFT
      RETURN
      END
```

```
      SUBROUTINE TIME(T)

      REAL NLOCX,NLOCY,NLOCZ
      REAL NSIZX,NSIZY,NSIZZ

      COMMON XB(100),YB(100),ZB(100),TB(100),NBURST
      COMMON FLOCX(100),FLOCY(100),FLOCZ(100)
      COMMON FSIZX(100),FSIZY(100),FSIZZ(100)
      COMMON NLOCX(100),NLOCY(100),NLOCZ(100)
      COMMON NSIZX(100),NSIZY(100),NSIZZ(100)

      COMMON /AGEIND/ IAGE(100)

      COMMON /HCSCR/ XCENT(20,150)
      COMMON /HCSCR/ SIGX(20,150),SIGY(20,150),SIGZ(20,150)
      COMMON /HCSCR/ FACTT(20,150),NPUFF(20)

      COMMON /HCCLG/ FSIGX,FSIGY,FSIGZ
      COMMON /HCCLG/ DSIGX,DSIGY,DSIGZ
      COMMON /HCCLG/ ASIGX,ASIGY,ASIGZ
      COMMON /HCCLG/ ESIGX,ESIGY,ESIGZ
      COMMON /HCCLG/ WNDSPD,FACTOR,TINCR


      PAGE=-1000.
      MUNGRP=0


      DO 299 MUN=1,NBURST

      AGE=T-TB(MUN)
      IF(AGE.NE.PAGE) GO TO 150
      IAGE(MUN)=MUNGRP
      GO TO 205

  150 CONTINUE
      MUNGRP=MUNGRP+1
      IAGE(MUN)=MUNGRP
      PAGE=AGE

      DO 199 I=1,120
      TW=AGE-TINCR*FLOAT(I-1)
      IF(TW.LE.0.0)GO TO 200
      XCENT(MUNGRP,I)=WNDSPD*TW
      SIGX(MUNGRP,I)=FSIGX*((WNDSPD*TW+ASIGX)/DSIGX)**ESIGX
      SIGY(MUNGRP,I)=FSIGY*((WNDSPD*TW+ASIGY)/DSIGY)**ESIGY
      SIGZ(MUNGRP,I)=FSIGZ*((WNDSPD*TW+ASIGZ)/DSIGZ)**ESIGZ
      FACTT(MUNGRP,I)=
     I              FACTOR/(SIGX(MUNGRP,I)*SIGY(MUNGRP,I)*SIGZ(MUNGRP,I))

  199 CONTINUE

      NPUFF(MUNGRP)=120
      GO TO 210

  200 CONTINUE
      NPUFF(MUNGRP)=I-1

  205 CONTINUE
      IF(NPUFF(MUNGRP).LE.0)GO TO 250

  210 CONTINUE
      NPM=NPUFF(MUNGRP)
      FLOCX(MUN)=XCENT(MUNGRP,I)+XB(MUN)
      FLOCY(MUN)=YB(MUN)
      FLOCZ(MUN)=ZB(MUN)
      FSIZX(MUN)=SIGX(MUNGRP,I)
      FSIZY(MUN)=SIGY(MUNGRP,I)
```

```
      FSIZZ(MUN)=SIGZ(MUNGRP,I)
      NLOCX(MUN)=XCENT(MUNGRP,NPM)+XB(MUN)
      NLOCY(MUN)=YB(MUN)
      NLOCZ(MUN)=ZB(MUN)
      NSIZX(MUN)=SIGX(MUNGRP,NPM)
      NSIZY(MUN)=SIGY(MUNGRP,NPM)
      NSIZZ(MUN)=SIGZ(MUNGRP,NPM)
      GO TO 299

  250 CONTINUE
      NPUFF(MUNGRP)=0
      FLOCX(MUN)=0.0
      FLOCY(MUN)=0.0
      FLOCZ(MUN)=0.0
      FSIZX(MUN)=0.0
      FSIZY(MUN)=0.0
      FSIZZ(MUN)=0.0
      NLOCX(MUN)=0.0
      NLOCY(MUN)=0.0
      NLOCZ(MUN)=0.0
      NSIZX(MUN)=0.0
      NSIZY(MUN)=0.0
      NSIZZ(MUN)=0.0

  299 CONTINUE

      RETURN

      END
```

```
      SUBROUTINE DENSTY(XP,YP,ZP,CON)

      COMMON XB(100),YB(100),ZB(100),TB(100),NBURST

      COMMON /AGEIND/ IAGE(100)

      COMMON /HCSCR/ XCENT(20,150)
      COMMON /HCSCR/ SIGX(20,150),SIGY(20,150),SIGZ(20,150)
      COMMON /HCSCR/ FACTT(20,150),NPUFF(20)

      CON=0.0

      DO 399 MUN=1,NBURST

      MUNGRP=IAGE(MUN)
      NPM=NPUFF(MUNGRP)
      IF(NPM.LE.0)GO TO 399
      X=XP-XB(MUN)
      Y=YP-YB(MUN)
      Z=ZP-ZB(MUN)

      CALL DENM(MUNGRP,X,Y,Z,CONM)
      CON=CON+CONM
  399 CONTINUE

      RETURN

      END
```

```
      SUBROUTINE DENM(MUNP,XP,YP,ZP,CON)

      COMMON /HCSCR/ XCENT(20,150)
      COMMON /HCSCR/ SIGX(20,150),SIGY(20,150),SIGZ(20,150)
      COMMON /HCSCR/ FACTT(20,150),NPUFF(20)

      COMMON /POINT/ X,Y,Z

      X=XP
      Y=YP
      Z=ZP
      MUN=MUNP

      CON=0.0
      NPM=NPUFF(MUN)
      IF(NPM.LE.0)GO TO 399
      IF(X.LE.XCENT(MUN,NPM)-4.0*SIGX(MUN,NPM))GO TO 399
      IF (X.GE.XCENT(MUN,1)+4.0*SIGX(MUN,1))GO TO 399

      IF(NPM.GE.5)GO TO 350

      DO 310 I=1,NPM
      CALL DENPF(MUN,I,TERM)
      CON=CON+TERM
  310 CONTINUE
      GO TO 399

  350 CONTINUE
      DSTBPF=(XCENT(MUN,1)-XCENT(MUN,NPM))/FLOAT(NPM-1)
      NMID=(XCENT(MUN,1)-X)/DSTBPF+2.0
      NLEFT=NMID
      NRIGHT=NMID-1
      IF (NLEFT.LE.0)NLEFT=1

  360 CONTINUE
      IF(NLEFT.GT.NPM)GO TO 370
      CALL DENPF(MUN,NLEFT,TERM)
      CON=CON+TERM
      NLEFT=NLEFT+1
      IF(TERM.GE.0.01)GO TO 360

  370 CONTINUE
      IF (NRIGHT.GT.NPM)NRIGHT=NPM
  375 CONTINUE
      IF(NRIGHT.LE.0)GO TO 399
      CALL DENPF(MUN,NRIGHT,TERM)
      CON=CON+TERM
      NRIGHT=NRIGHT-1
      IF(TERM.GE.0.01)GO TO 375

  399 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DENPF(MUNP,IP,CON)

      COMMON /HCSCR/ XCENT(20,150)
      COMMON /HCSCR/ SIGX(20,150),SIGY(20,150),SIGZ(20,150)
      COMMON /HCSCR/ FACTT(20,150),NPUFF(20)

      COMMON /POINT/ X,Y,Z

      I=IP
      MUN=MUNP
      CON=0.0

      CONN=(-.5*(((X-XCENT(MUN,I))/SIGX(MUN,I))**2
     *                  +(Y/SIGY(MUN,I))**2
     *                   +(Z/SIGZ(MUN,I))**2))

      IF(CONN.LE.-25.) GO TO 350
      CON=FACTT(MUN,I)*EXP(CONN)
  350 CONTINUE
      RETURN

      END
```

```
      SUBROUTINE LCON  (POXP,POYP,POZP,UXP,UYP,UZP,TOTLNC)

      COMMON XB(100),YB(100),ZB(100),TB(100),NBURST

      COMMON /AGEIND/ IAGE(100)

      COMMON /HCSCR/ XCENT(20,150)
      COMMON /HCSCR/ SIGX(20,150),SIOY(20,150),SIGZ(20,150)
      COMMON /HCSCR/ FACTT(20,150),NPUFF(20)

      UX=UXP
      UY=UYP
      UZ=UZP
      TOTLNC=0.0

      DO 499 MUN=1,NBURST

      MUNGRP=IAGE(MUN)
      NPM=NPUFF(MUNGRP)
      IF(NPM.LE.0)GO TO 499

      POX=POXP-XB(MUN)
      POY=POYP-YB(MUN)
      POZ=POZP-ZB(MUN)

      CALL LCONM(MUNGRP,POX,POY,POZ,UX,UY,UZ,TERM)
      TOTLNC=TOTLNC+TERM

  499 CONTINUE

      RETURN

      END
```

```
      SUBROUTINE LCONM (MUNP,POXP,POYP,POZP,DXP,DYP,DZP,TOTLNC)

      COMMON /HCSCR/ XCENT(20,150)
      COMMON /HCSCR/ SIGX(20,150),SIGY(20,150),SIGZ(20,150)
      COMMON /HCSCR/ FACTT(20,150),NPUFF(20)

      COMMON /LINE/ POX,POY,POZ,DX,DY,DZ

      POX=POXP
      POY=POYP
      POZ=POZP
      DX=DXP
      DY=DYP
      DZ=DZP
      MUN=MUNP
      NPM=NPUFF(MUN)
      TOTCL=0.0
      IF(NPM.LE.0)GO TO 490
      VL=XCENT(MUN,NPM)
      VU=XCENT(MUN,1)

C         PUFF CHOOSING LOGIC
C         DETERMINE INTERSECTION WITH Y,Z PLANE
      IF (DY.EQ.0.0)GO TO 410
      S=-POY/DY
      XINTER=POX+S*DX
      IF(XINTER.LE.VL)GO TO 410
      IF(XINTER.GE.VU)GO TO 410
      DSTBPF=(VU-VL)/FLOAT(NPM-1)
      NMID=(VU-XINTER)/DSTBPF+2.0
      NLEFT=NMID
      NRIGHT=NMID-1
      IF (NLEFT.LE.0)NLEFT=1
      GO TO 470
  410 CONTINUE
      NMID=NPM/2
      CALL LCONPF(MUN,NMID,TERM)
      IF(TERM.LE.0.01)GO TO 420
      NLEFT=NMID+1
      NRIGHT=NMID
      GO TO 470
  420 CONTINUE
      NLEFT=1
      CALL LCONPF(MUN,NLEFT,TERM)
      IF(TERM.LE.0.01)GO TO 430
      NRIGHT=0
      GO TO 470
  430 CONTINUE
      NRIGHT=NPM
      CALL LCONPF(MUN,NRIGHT,TERM)
      IF (TERM.LE.0.01) GOTO 440
      NLEFT=NPM+1
      GO TO 470
  440 CONTINUE
      TOTCL=0.0
      GOTO 490

  470 CONTINUE
      IF(NLEFT.GT.NPM)GO TO 480
C         COMPUTE LINE CONCENTRATION CONTRIBUTION
      I=NLEFT
      CALL LCONPF(MUN,I,TERM)
      TOTCL=TOTCL+TERM
      NLEFT=NLEFT+1
      IF(TERM.GE.0.01)GO TO 470

  480 CONTINUE
      IF (NRIGHT.GT.NPM)NRIGHT=NPM
```

```
 485 CONTINUE
     IF(NRIGHT.LE.0)GO TO 490
     I=NRIGHT
     CALL LCONPF(MUN,I,TERM)
     TOTCL=TOTCL+TERM
     NRIGHT=NRIGHT-1
     IF(TERM.GE.0.01)GO TO 485
 490 CONTINUE
     TOTLNC=TOTCL
     RETURN
     END
```

```
      SUBROUTINE LCONPF (MUNP,IP,PFLNC)

      COMMON /HCSCR/ XCENT(20,150)
      COMMON /HCSCR/ SIGX(20,150),SIGY(20,150),SIGZ(20,150)
      COMMON /HCSCR/ FACTT(20,150),NPUFF(20)

      COMMON /LINE/ POX,POY,POZ,DX,DY,DZ

      I=IP
      MUN=MUNP
      PFLNC=0.0
      A=(DX/SIGX(MUN,I))**2+(DY/SIGY(MUN,I))**2+(DZ/SIGZ(MUN,I))**2
      B=DX*(POX-XCENT(MUN,I))/SIGX(MUN,I)**2+
     * DY*(POY               )/SIGY(MUN,I)**2+
     * DZ*(POZ               )/SIGZ(MUN,I)**2
      S=-B/A
      PMX=POX+S*DX
      PMY=POY+S*DY
      PMZ=POZ+S*DZ
      C=((PMX-XCENT(MUN,I))/SIGX(MUN,I))**2+
     * ((PMY              )/SIGY(MUN,I))**2+
     * ((PMZ              )/SIGZ(MUN,I))**2
      CC=(-.5*C)
      IF(CC.LE.-25.) GO TO 399
      PFLNC=FACTT(MUN,I)*2.5066283*EXP(CC)/SQRT(A)
  399 CONTINUE
      RETURN

      END
```

DISTRIBUTION LIST

No. of Copies	Organization
2	Commander Defense Documentation Center ATTN: TCA Cameron Station Alexandria, VA 22314
2	Commander US Army Armament Research & Development Command ATTN: DRDAR-SEA Technical Library Dover, NJ 07801
1	Commander Rock Island Arsenal ATTN: Tech Lib Rock Island, IL 61299
2	Commander US Army Test & Evaluation Command ATTN: STEDP-MT-L STEDP-MT (Dr. L. Solomon) Dugway Proving Ground, UT 84022
1	Commander US Army Electronics R&D Command ATTN: DRDEL-SA Fort Monmouth, NJ 07703
2	Commander US Army Electronics R&D Command ATTN: DRDEL-AP-OA DRDEL-AP-CCM (Mr. D. Giglio) 2800 Powder Mill Road Adelphi, MD 20783
9	Commander US Army Materiel Development & Readiness Command ATTN: DRCCP DRCDE-F DRCRE-I DRCPA-S DRCQA DRCDE-R DRCDE-D DRCBSI-L DRCBSI-D 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander Harry Diamond Laboratories ATTN: DELHD-SAB 2800 Powder Mill Road Adelphi, MD 20783
1	Commander US Army Aviation R&D Command ATTN: DRDAV-BC PO Box 209 St. Louis, MO 63166
1	Commander US Army Electronics R&D Command Atmospheric Sciences Lab ATTN: DELAS-EO (Dr. Richard Gomez) White Sands Missile Range NM 88002
3	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-SL ATAA-T ATAA-TDB (Mr. L. Dominguez) White Sands Missile Range NM 88002

DISTRIBUTION LIST

No. of Copies	Organization
2	Commander US Army Missile Command ATTN: DRSMI-C (R&D) DRSMI-CGA(R&D)(Mr. B. Fowler) Redstone Arsenal, AL 35809
2	Commander US Army Tank-Automotive Research & Develop. Command ATTN: DRDTA-UL (Tech Lib) DRDTA-V Warren, MI 48090
1	Commander US Army Natick R&D Command ATTN: DRDNA-O Natick, MA 01760
2	Commander US Army Concepts Analysis Agency ATTN: MOCA-SMC (Mr. H.Hock) (1cy) 8120 Woodmont Avenue Bethseda, MD 20014
1	Commander US Army Training and Doctrine Command ATTN: ATCD-Z (MAJ I.J. Licata) Fort Monroe, VA 23651
1	Commandant US Army Infantry School ATTN: ATSH-CD-CS (CPT D. Geroge) Fort Benning, GA 31905
1	Director Night Vision Laboratories ATTN: DELNV-VI (Ms. Luanne Obert) Fort Belvoir, VA 22060
1	Naval Air Systems Command ATTN: Code Air-310C (Dr. H. Rosenwasser) Washington, DC 20361
1	Commander US Army Troop Support & Aviation Materiel Readiness Command ATTN: DRSTS-BA 4300 Goodfellow Blvd St. Louis, MO 63120
2	Commander US Army Mobility Equipment R&D COMMAND ATTN: DRDME-O DRDME-RT (Mr. O.F. Kezer) Fort Belvoir, VA 22060
2	Chief Defense Logistics Studies Information Exchange US Army Logistics Management Center ATTN: DRXMC-D Fort Lee, VA 23801
2	Commander US Army Combined Arms Center ATTN: ATZLCA-CAT (Mr. R.E. DeKinder, Jr.) ATZLCA-TM-K (LTC D. Bacon) Fort Leavenworth, KS 66027
1	Commandant US Army Field Artillery School ATTN: ATSF-G-RA (MAJ John Kehres) Fort Sill, OK 73503
1	Marine Corps Development Center ATTN: (FPR) MEDEC (MAJ Clara Tucker) Quantico, VA 22134
1	LCDR William E. Major NAVSEASYSOM (Code SEA-62Y21) Washington, DC 20350

DISTRIBUTION LIST

No. of Copies	Organization
1	Naval Intelligence Support Center ATTN: NISC (Code 43) (H.F. St. Aubin) 4301 Suitland Road Suitland, MD 20390
1	Naval Weapons Support Center ATTN: Dr. Duane Johnson Crane, IN 47522
1	Commander Tactical Air Command ATTN: DRAR (MAJ D. Nix) Langley Air Force Base, VA 23665
1	Commander USAFAWS/DNDP ATTN: CPT J. Kahler Scott Air Force Base, IL 62225
1	Oklahoma State Field Office ATTN: Dr. C.K. Arpke P.O. Box 1925, Bldg 230 Eglin Air Force Base, FL 32542
1	Naval Surface Weapons Center ATTN: Diane L. Shamblin (Code DG-302) Dahlgren, VA 22448
1	Naval Weapons Center ATTN: Dr. Alex Shlanta (Code 3173) China Lake, CA 93555
1	Reliability Analysis Center ATTN: Mr. I.L. Krulac Griffiss AFB, NY 13441
1	Commander ASD/ENAMB ATTN: Mr. Larry J. Beasley Wright-Patterson AFB, OH 45433
1	Commander USAFTAWC/THL ATTN: MAJ J. T. McNeely,Jr Eglin Air Force Base, FL 32542

DISTRIBUTION LIST

Aberdeen Proving Ground

No. of Copies	Organization
3	CDR, USATECOM ATTN: DRSTE DRSTE-CS-A DRSTE-AD-M (Mr. W. Baity) Bldg 314
2	Dir, BRL ATTN: DRDAR-BLB (Mr. A. LaGrange)(1 cy) Bldg 328
1	Dir, BRL ATTN: DRDAR-TSB-S (STINFO Branch) Bldg 305
1	Dir, HEL, Bldg 520
1	Cdr, USA O&CC&S ATTN: ATSL-CLC-C (LT A. Slavinski) Bldg 3071
1	Cdr/Dir, CSL ATTN: DRDAR-CLB-PS (Dr. Stuebing) APG-EA, MD 21010
1	PM, Smoke/Obscurants ATTN: DRCPM-SMK (Mr. J. Steedman) Bldg 324
17	Director, USAMSAA ATTN: DRXSY-GI (Mr. S. Gerard) DRXSY-LM (Mr. P. Frosell) DRXSY-LM (Mr. Marchetti) (15 cys) Bldg 392